Method	Performance(dB)
Mean value	0.00
First order bivariate polynomials	3.48
Second order bivariate polynomials	5.46
Dirichlet using original boundary pixels	5.21
Dirichlet using suboptimally modified boundary pixels	6.75
Dirichlet using optimally modified boundary pixels	7.11

Table 1: Performance comparison

4. EXPERIMENT RESULTS

The performance of the proposed methods are evaluated on square regions of sizes ranging from 10x10 to 25x25. Some of the regions contain edges, so that robustness of the methods can be tested. The mean square prediction error is taken as the measure of performance. The performances are also compared to the performance of approximating the regions by 2-D polynomials of up to second order. Table 1 shows the average performances with reference to prediction by the mean value of the region. It can be seen from the table that the performance of the suboptimal method is close to that of the optimal one, and that both methods can bring significant improvement over using the original values. The latter observation is especially true for regions containing edges.

Figure 1 depicts the prediction of a 25x25 region for various methods.

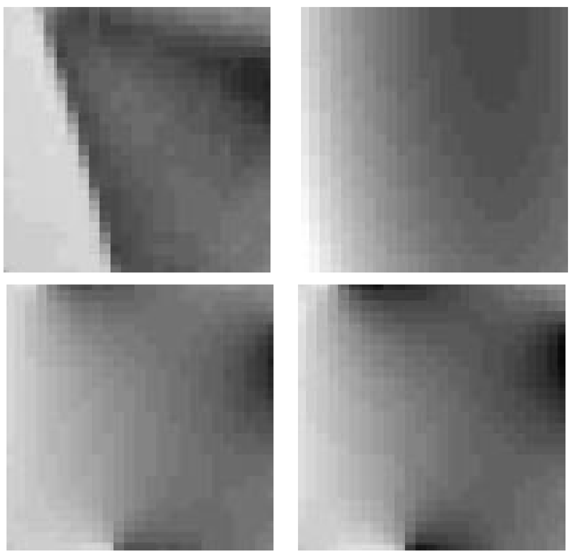

Figure 1: *Upper left:* original 25x25 region, *upper right:* approximation by a quadratic polynominal, *lower left:* prediction from original boundary pixel values, *lower right:* prediction from optimally selected boundary pixel values

5. CONCLUSIONS

Experimental results support our expectation that the pixel values in a region can be successfully predicted from boundary values, if there are no edges or abrupt changes in the region. For some applications, such as segmentation based image coding, modification of the actual values of the boundary pixels is possible. For these kind of applications, the proposed methods bring significant improvements over using original boundary pixel values, and increase the robustness of the prediction. The first one of the proposed methods is optimal in least squares sense, whereas the second one is computationally simple. The method of modifying the boundary pixel values can be used for other optimality criterions as well, but in this case the optimal solutions would require the employment of nonlinear techniques.

6. REFERENCES

[1] "roifill" function of Image Processing Toolbox, Matlab^{TM} Technical Computing Software Version 5.2, The MathWorks Inc., Jan 17, 1998.

[2] W. H. Press, S. A. Teukolsky, W. T. Vetterling, B. P. Flannery, "Numerical Recipes in C, The Art of Scientific Computing", Second Edition, Cambridge University Press, 1992, pp. 863-871

[3] M. Kunt, M. Benard, R. Leonardi, "Recent Results in High-Compression Image Coding", IEEE Transactions on Circuits and Systems, Vol. 34., No. 11, pp. 1306-1336, November 1987

[4] S. Carlsson, "Image Reconstruction from Coded Edge Data", Proc. Premier Colloque Image, Biarritz, France, June 1984, pp 71-77

[5] S. Carlsson, "Sketch Based Representations in Image Coding", Ph. D. Thesis, Department of Electrical Engineering, Royal Institute of Technology, Stockholm, Sweden, December 1986.

overhead, because this 1-D data is expected to be slowly varying. Such a coding system is mentioned in [3] with reference to [4] and [5]. Although formulated differently, the prediction method in the mentioned system is equivalent to solving the Dirichlet problem.

One potential problem associated with the prediction by solving Equation (1) is that it is too sensitive to the pixel values on the boundary. It will not perform very well if the boundary pixels are noisy, or if the boundary is a bit distorted due to lossy coding, or simply if the boundary pixels do not represent the region pixels very well. This problem can be overcome by using more robust values instead of actual x_b values. x_b is supposed to be sent anyway, so it will not make a difference to use values different than the original for boundary pixels.

3.1. Optimal Selection of Boundary Pixel Values

Let us reformulate the problem as follows: let the vector of predicted values for all pixels of the region be denoted as $\widehat{x}_a = \left[\widehat{x}_b^T \;\; \widehat{x}_i^T\right]^T$. Equation (1) now takes the following form:

$$\widehat{x}_a = \begin{bmatrix} I \\ A^{-1}B \end{bmatrix} \cdot \widehat{x}_b \tag{3}$$

Then, we can choose $\widehat{x}_b$ so as to minimize the sum of square of prediction error $(x_a - \widehat{x}_a)^T\,(x_a - \widehat{x}_a)$. Let the matrix in right-hand side of Equation (3) be denoted by $\mathbf{C}$. In this case, the minimizing $\widehat{x}_b$ is given by

$$\widehat{x}_b = \left(\mathbf{C}^\mathbf{T}\mathbf{C}\right)^{-1}\mathbf{C}^\mathbf{T} \cdot x_a \tag{4}$$

and the final prediction is given by

$$\widehat{x}_a = \mathbf{C}\;\left(\mathbf{C}^\mathbf{T}\mathbf{C}\right)^{-1}\mathbf{C}^\mathbf{T} \cdot x_a \tag{5}$$

This selection of boundary pixel values is optimal in least square sense, but it is a computationally complex operation. Therefore, suboptimal solutions with significantly less computational requirements might be desirable.

3.2. A Robust Method with Low Complexity

The basic idea of the low-complexity method is modifying the original boundary pixel values in a minimal amount so that predicted region would satisfy some simple constraints defined by the original pixel intensity values vector x_a.

The constraints can qualitatively be defined as follows:

- The mean value of the predicted region $\widehat{x}_a$ should be equal to that of the original region x_a.
- The first order moments of the predicted region in both directions should be equal to those of the original region.
- Some horizontal and vertical symmetry measures of $\widehat{x}_a$ must be as close to those of x_a.

In order to describe the imposition of these constraints in matrix notation, we need to define an orthonormal set of five vectors $\{b_1, b_2, b_3, b_4, b_5\}$. These vectors have the same size as x_a, and obtained by the following procedure:

- For each $(i,j) \in$ *interior* define
  $\tilde{b_1}(i,j) = 1$,
  $\tilde{b_2}(i,j) = i - \bar{i}$,
  $\tilde{b_3}(i,j) = j - \bar{j}$,
  $\tilde{b_4}(i,j) = |i - \bar{i}|$,
  $\tilde{b_5}(i,j) = |j - \bar{j}|$,

  where $\bar{i}$ and $\bar{j}$ are the mean values of i and j respectively, and $|.|$ denotes absolute value.

- Construct the set of vectors $\{\tilde{b_1}, \tilde{b_2}, \tilde{b_3}, \tilde{b_4}, \tilde{b_5}\}$ by rearranging the elements of the above-defined functions, exactly the same way as the vector x_a is obtained from $x(i,j)$ values.

- Obtain $\{b_1, b_2, b_3, b_4, b_5\}$ by orthonormalizing $\{\tilde{b_1}, \tilde{b_2}, \tilde{b_3}, \tilde{b_4}, \tilde{b_5}\}$

Now,
$b_1^T x_a$ gives the mean value of x_a,
$b_2^T x_a$ is the first order vertical moment of x_a,
$b_3^T x_a$ is the first order horizontal moment of x_a,
$b_4^T x_a$ is a measure of vertical symmetry of x_a, and
$b_5^T x_a$ is a measure of horizontal symmetry of x_a.

The low complexity method for selecting the values of boundary pixels $\widehat{x}_b$ goes as follows:

1. Compute $\widehat{x}_i$ by solving Equation (1) with some low-complexity iterative technique such as Gauss-Seidel iterations
2. Set $\tilde{x}_a = \left[x_b^T \;\; \widehat{x}_i^T\right]^T$
3. Set $d = x_a - \tilde{x}_a$
4. Set $x_a^c = \tilde{x}_i + b_1(b_1^T d) + b_2(b_2^T d) + b_3(b_3^T d) + b_4(b_4^T d) + b_5(b_5^T d)$
5. Compose $\widehat{x}_b$ by taking first B elements of x_a^c

Note that the total computational complexity of this method is marginally more than that of iteratively solving Equation (1).

PREDICTION OF PIXELS IN AN ARBITRARY REGION FROM PIXELS ON THE REGION BOUNDARY

Levent Öktem and Jaakko Astola

Signal Processing Laboratory,
Tampere University of Technology,
P.O.B. 527, FIN-33101 Tampere, Finland
loktem@cs.tut.fi jta@cs.tut.fi

ABSTRACT

Solution of Dirichlet problem is proposed as a method for predicting the pixel values in an arbitrary region from the pixel values on the boundary of the region. Using the actual boundary pixel values as input to the predictor does not necessarily yield the best prediction. A computationally complex method is proposed for 'selecting' the pixel values on the boundary so as to yield an optimal prediction in a least squares sense. An iterative low complexity approximation to this method is also proposed. Experimental results show that both methods provide significant improvements over using original boundary pixel values for prediction.

1. INTRODUCTION

Interpolation of pixel values in a region from the values on the region boundary might be useful in various applications such as image editing and segmentation based image coding. One of the ways [1] for interpolation is solving the Dirichlet problem, i.e. finding the field that satisfies Laplace equation and conforms to the boundary value constraints. Assuming that the region does not contain edges or abrupt changes, interpolation by solving the Dirichlet problem can be expected to approximate the region content well.

In this work, we present the application of this method to prediction in image compression context. In our context, a region is a connected set of pixels grouped together according to a similarity measure, and computed by a segmentation process. We further assume that the boundary pixels belong to the region, i.e. they conform to the similarity measure of the region. Our initial problem can be stated as follows: "given the pixel values along the boundary, predict the pixel values in the region so as to yield small prediction error".

In the next section, the prediction method is formulated in matrix notation. We discuss the application of this method to segmentation based image coding in Section 3, and propose several variants. Section 4 presents the experiment results, which is followed by the final section, Conclusions.

2. PREDICTION BY SOLVING DIRICHLET PROBLEM

Let x_b denote the $B \times 1$ vector of pixels on the boundary, and let $\widehat{x}_i$ denote the $I \times 1$ vector of predicted values for pixels in the interior. The Dirichlet problem on discrete domain can then be approximated as

$$\mathbf{A} \cdot \widehat{x}_i = \mathbf{B} \cdot x_b \tag{1}$$

where each row of $\mathbf{A}$, together with the corresponding row of $\mathbf{B}$, impose an approximation of Laplace's equation at a certain interior pixel. The approximation we use for a pixel $x(i,j)$ with coordinates (i,j) is

$$4\,x(i,j) - x(i-1,j) - x(i+1,j) - x(i,j-1) - x(i,j+1) = 0 \tag{2}$$

Note that $\mathbf{A}$ is a sparse matrix, but its size is huge (even for a 65x65 square region, the size of $\mathbf{A}$ is 4096x4096). Hence, direct solution of Equation (1) can be computationally complex. Instead of direct solution, iterative solution methods such as Gauss-Seidel iterations, or successive overrelaxation can be employed [2]. Gauss-Seidel iterations can be summarized as follows in pseudocode:

```
for k := 1 to MAX_ITER
  for i := MIN_Y to MAX_Y
    for j := MIN_X to MAX_X
      if (i,j) ∈ interior
        x(i,j) ← (x(i−1,j)+x(i+1,j)+x(i,j−1)+x(i,j+1))/4
```

3. APPLICATION TO SEGMENTATION BASED IMAGE CODING

The prediction of interior pixels from boundary pixels can be used in a segmentation based image coding scheme, where the pixel values along the boundaries are supposed to be coded as side information. This typically will not bring too much