IMAGE INTERPOLATION USING A SIMPLE GIBBS RANDOM FIELD MODEL

Nicos Herodotou, A.N. Venetsanopoulos

University of Toronto
Dept. of Elect. & Computer Eng.
10 Kings College Road,
Toronto, Ont. M5S 1A4, CANADA

Levent Onural

Bilkent University,
Dept. of Elect. & Electronics Eng.
Bilkent, Ankara, TR-06533, TURKEY

ABSTRACT

Spatial interpolation is an important technique that is often used to recover an image from its downsampled version, or to simply perform image expansion. Many conventional linear techniques exist, however, these often perform rather poorly in a subjective manner. In this paper, image interpolation is performed using a binary-based Gibbs Random field (GRF) model. Images are interpolated from their downsampled versions along with a number of texture parameters that are estimated within smaller image blocks. These iterative GRF methods are subsequently approximated by a non-iterative nonlinear filtering operation, thereby reducing the computational complexity of the interpolation process. Experimental results indicate that the statistical GRF approaches adapt to textured regions as well as the smooth areas within an image, and thus, can achieve better results than the conventional linear schemes.

1. INTRODUCTION

The process of subsampling or decimation is frequently used to compress the spatial information in an image for transmission or storage purposes as in multi-resolution decompositions (MRD). In order to obtain the original high resolution image, the reverse procedure of decimation, termed interpolation must be performed on the downsampled version. These two operations find their way into many practical applications such as progressive image transmission systems, aspect ratio conversion for different display devices, optical scanning devices, and image/video zooming.

Many conventional interpolation techniques exist [1, 2, 3], and most of these are implemented according to linear theory by lowpass multirate filtering. Some of these include pixel replication, bilinear interpolation, and B-splines. However, the above methods are not the best from a perceptual point of view.

In this paper, the problem of interpolation is approached under the statistical framework of Gibbs random field (GRF) models. The application of GRF models to problems in texture modelling and classification [4, 5], segmentation and restoration of degraded images [6], has been studied quite extensively. Here, we follow a similar approach by formulating the problem of image interpolation as an image restoration problem. Different parametric forms of Gibbs random fields have been applied to this problem [7, 8, 9].

GRF modelling allows the joint probability distribution of the image pixels to be expressed in terms of an energy function which provides a powerful mechanism for modelling spatial continuity. Through the simple choice of a binary-based energy function, several parameters are determined in different regions of the image. These parameters characterize the bonding strength of the pixels in the region in which they are estimated. Parameter estimation is a computationally expensive procedure and several techniques have been used in modelling noisy or textured models [6], [10]. An iterative approach based on a gradient ascent method [11] is used here and adapted to the problem of interpolation. Having determined the parameters of the Gibbs distribution, the interpolated image is determined from its downsampled version along with these estimated "texture" parameters.

A maximum a posteriori (MAP) estimate of the original image given the subsampled image is desirable in regeneration, however, iterative schemes which are computationally prohibitive are necessary. An iterative deterministic relaxation algorithm, ICM, which does not necessarily converge to the MAP estimate, is used to obtain the results of the interpolated image [12]. This iterative ICM method is implemented using binary logic-type filters and subsequently approximated by a non-iterative 3-pass nonlinear filtering operation. The use of binary logic operations allows for a fast and efficient implementation, both in hardware and software. This is compared to the method of simulated annealing (SA) [13], [14] which is known to converge to the MAP estimate. Experimental results are presented using the different methods outlined above.

2. MODELLING IMAGES USING GRF'S

In the GRF representation of images, each pixel is assigned to a random variable and the collection of all of these is the random field X. The joint probability distribution of X is given by

$$P(X=x) = \frac{1}{Z} e^{-U(x)/T} \quad (1)$$

$$U(x) = \sum_{c \in C} V_c(x) \quad (2)$$

where U is the energy function, C is the set of cliques, V_c is the clique potential, Z is the normalizing constant, and

0-8186-7310-9/95 $4.00 © 1995 IEEE

Proceedings of the 1995 International Conference on Image Processing (ICIP '95)
0-8186-7310-9/95 $10.00 © 1995 IEEE

T is the temperature constant.

A simple binary valued energy function over a second order neighbourhood is chosen here for image regeneration.

$$V_1 = \begin{cases} -\beta & \mid x(i,j) - x(i+1,j) \mid < t \\ \beta & \text{otherwise} \end{cases}$$

$$V_2 = \begin{cases} -\gamma & \mid x(i,j) - x(i,j+1) \mid < t \\ \gamma & \text{otherwise} \end{cases}$$

$$V_3 = \begin{cases} -\delta & \mid x(i,j) - x(i+1,j-1) \mid < t \\ \delta & \text{otherwise} \end{cases}$$

$$V_4 = \begin{cases} -\zeta & \mid x(i,j) - x(i+1,j+1) \mid < t \\ \zeta & \text{otherwise} \end{cases} \quad (3)$$

where t is the pixel threshold value of similarity. The estimation of the model parameters $\beta, \gamma, \delta, \zeta$ of the original image is the first step in the interpolation problem. The above parameters can be represented by the parameter vector $s = [\beta, \gamma, \delta, \zeta]$ and are determined by an iterative parameter estimation technique using a gradient ascent method [11]. An objective function $Q = \log[P(X \mid s')/P(X \mid s)]$ is maximized through an iterative process until convergence is attained (s is the model parameter vector estimated from the previous iteration while s' is the parameter vector yet to be estimated). The parameter vector s can be updated by the parameter update equation

$$s'^{(k+1)} = s'^{(k)} + \lambda^{(k)} \nabla^{s'^{(k)}} Q \quad (4)$$

where $\lambda^{(k)}$ is the step size for the update of s' at the kth iteration and ∇ is the gradient function. The iterative scheme described above admits a unique estimate of s' when Q is strictly concave [11]. This is the case for the energy function chosen above.

3. IMAGE REGENERATION

In image regeneration, or interpolation, we begin with a downsampled image along with the estimated parameter set of the Gibbs distribution. The first step requires the insertion of randomly chosen pixel values in every other row and every other column (EORC) of the downsampled image. The result of this process we call the "upsampled" image. In this paper, we consider EORC (i.e. rectangular) downsampling/upsampling and also note that pixel insertion does not necessarily require that random values be chosen. The interpolation problem requires choosing acceptable values for these inserted pixels. A MAP criterion can be used to select the maximum likely pixel values which are to replace these random ones in the upsampled image. In other words, a maximization of the probability of the image given the known pixels in the upsampled image and the texture parameters is desired. However, MAP estimation presents a formidable problem since the number of possible intensity images is enormous, ruling out any type of explicit solution. The iterative technique of simulated annealing is known to converge to the MAP estimate, however, it is computationally expensive as it requires a large number of scans over the image [13]. The ICM process is another iterative scheme that is much less computationally demanding than SA, however, it still requires a large number of passes over the image. Due to this complexity, here in this paper, the ICM technique is converted to a nonlinear filtering process where binary logic-type filters are used. This also allows for a simple hardware implementation. The iterative process is also eliminated by implementing these filters over three passes of the image. Let us examine the simple binary valued energy function over a second order neighbourhood as in equation (3). At any pixel site only the 4 surrounding pixels have any effect on the conditional probability function. Clearly, the pixel $x(i,j)$ chosen must result in the smallest value for the sum $V = V_1 + V_2 + V_3 + V_4$ in order to have the highest probability of occurrence, as in ICM. If for example, the pixel $x(i,j)$ falls within t of the pixel $x(i+1,j)$ then $-\beta$ is contributed towards the sum. If at the same time this value also is within t of $x(i,j+1)$ then $-\gamma$ is added towards the sum, and so forth. Not all pixel values in the range of grey levels will produce a different value for the sum. Thus, only pixels that produce different sums need to be checked. These pixels can be determined as follows. Each of the four neighbouring pixels $x(i+1,j), x(i,j+1), x(i+1,j-1), x(i+1,j+1)$ has an associated region of similarity R_1, R_2, R_3, R_4 , respectively. The region or the set R_1 consists of the pixels in the range $[x(i+1,j)-(t-1)]$ to $[x(i+1,j)+(t-1)]$ and R_1' , the complement of R_1 , consists of all other pixel values in the range of grey level values. The set R_1 has an associated weight attached to it which is $-\beta$ and R_1' is β . Regions R_2, R_3, R_4 are formed in a similar fashion. The sum V is computed only for different overlapping regions. If there is an overlap of all four regions then $V = -\beta - \gamma - \delta - \zeta$ and clearly this results in the smallest possible value for V . However, if there is no intersection of these 4 regions then the sums must be computed for different combinations of 3 overlapping regions (i.e. $R_2 \cap R_3 \cap R_4$), 2 overlapping regions, and single regions. In total, 15 combinations must be checked. The smallest sum is chosen. This reduces the number of checks if all 256 grey level values must be tested. Furthermore, these set operations can be implemented via binary logic-type operations shown in the next section.

4. NONLINEAR FILTERING USING BINARY LOGIC-TYPE OPERATIONS

The set operations indicated in the previous section can be performed using binary logic-type filters as will be described below. Every pixel value has an associated region, R_i which can be stored as a bit pattern in a 256 bit-wide register where the least significant bit represents the pixel value 0, and the most significant, the value 255. The bits which correspond to the pixel values in the region R_i are marked with 1's and all other bits are set to 0's. Therefore, a specific bit pattern is generated for a particular region. If there are 256 grey levels, then there are 256 possible bit patterns (i.e. 256 possible regions) for any chosen threshold t . These patterns can be stored in the form of a look-up table (LUT) which can consist of 256 different registers each 256 bits wide. In hardware this can be 64K of non-volatile memory. The set operations described earlier can now be implemented by bitwise logical AND and OR operations. Thus, for each of the 4 neighbourhood pixel values

Proceedings of the 1995 International Conference on Image Processing (ICIP '95)
0-8186-7310-9/95 $10.00 © 1995 IEEE

(i.e. $x(i+1,j),\ldots$) the corresponding bit patterns are found from the LUT and a series of bitwise logical AND operations are performed in place of the set operations shown earlier. The output value of each AND operation is taken and a bitwise OR operation is subsequently performed on this. If the result is 1 then there is region overlap otherwise a 0 is encountered which indicates no overlap. For each combination of overlapping regions there is an associated sum of the parameter values (i.e. for $R_1 \cap R_2 \cap R_3$, $V = -\beta-\gamma-\delta+\zeta$). These sums can be precomputed and stored for each combination of overlapping regions and stored in increasing order for a particular parameter set. Thus, at the interpolation stage the bitwise AND and OR operations are performed, and as a result the probabilities of pixel values are checked in decreasing order. If a 1 is found in the output then the current pixel replaces the previous one, and the algorithm proceeds to the next site for pixel replacement. A simple example may help the above explanation. For the purposes of this example, let us assume that there are only 8 grey level values, and the threshold of similarity, t is 2. In addition to this, let the 4 neighbouring pixels x_1, x_2, x_3, x_4 have pixel values of $5, 6, 1, 6$, respectively, and the model parameters in equation (3) be as follows $\beta = \gamma = 1$, $\delta = 3$, and $\zeta = 10$. In this simple example just described, the LUT will consist of 8 registers, each 8 bits wide $b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0$ since there are 8 possible grey level values. The first step in the procedure is to retrieve the bit patterns that correspond to the region of each neighbouring pixel. For x_1, its corresponding region R_1 has a bit pattern

$$B_1 = 01110000 = b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0$$

since $x_1 = 5$. For $x_2 = 6, x_3 = 1$, and $x_4 = 6$ the associated bit patterns B_2, B_3, and B_4 are

$$B_2 = 11100000 = b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0$$

$$B_3 = 00000111 = b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0$$

$$B_4 = 11100000 = b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0$$

respectively. We first start with the smallest possible sum, which is -15, and this is obtained if all 4 regions overlap $B_1 \cdot B_2 \cdot B_3 \cdot B_4$ (B_i are the registers and the symbol $\cdot$ indicates a bitwise AND operation). However, a bitwise logical AND with all four registers shown above yields

$$Y = B_1 \cdot B_2 \cdot B_3 \cdot B_4 = 00000000$$

and thus, a logical OR of the individual bits $b_0, \ldots, b_7$ results in 0. Therefore, there is no intersection. As we follow this procedure with bit patterns that give increasingly larger sums, the result of $B_1 \cdot B_2 \cdot B_3' \cdot B_4$ (with a sum of -9) yields

$$Y = B1 \cdot B_2 \cdot B_3' \cdot B_4 = 01100000$$

and a logical bitwise OR on Y gives a 1. Since a 1 is obtained upon the logical OR, the search is terminated. The most probable pixel value can be any one of the bit positions that holds the 1. In this case, the pixel value of 5, or 6 (i.e. b_5, or b_6) is the most probable value and an exhaustive search was not necessary in determining this. This example can very easily be extended to 256 grey levels and any threshold value, t.

The filtering operations just described duplicate the ICM process in an efficient manner. However, a further simplification is possible by eliminating the iterative procedure and reducing the number of operations per pixel. A 3-pass filter that implements the logic-type operations described above can be used to simplify and approximate the ICM method. The three stage filtering process operating on the subset of pixels of the upsampled image is described below.

$$\begin{matrix} X & N_2 & X & N_2 & X \\ N_2 & N_1 & N_2 & N_1 & N_2 \\ X & N_2 & X & N_2 & X \end{matrix}$$

In the first pass of the image, pixels N_1 are determined using each pixel's south-east and south-west neighbours. In the second pass, the pixels N_2 are found using each ones' east and south neighbours. In the final pass all of the pixels N_1, N_2 are recomputed from each ones' corresponding four neighbours (i.e. east, south-east, south, south-west). In the first two passes, only one logical AND and one OR operation are required per pixel and in the third pass only 42 operations of each type, at most, are required per pixel. This reduces the computational burden of the ICM method and yields visually similar results as found in the experiments. Further to this, a very fast hardware implementation is possible for these logic-type operations.

5. EXPERIMENTAL RESULTS

All of the techniques mentioned in this paper were applied to the interpolation problem and the results are provided for comparison. A 64×64 textured region (i.e. whiskers) of the original "baboon" image of Figure 1a) was used in order to compare the results. Parameters were determined for each 8×8 block over this 64×64 region. This block size is small enough to adapt to smaller textured regions yet large enough to keep the computational complexity relatively low. The results of pixel replication, bilinear interpolation, SA, ICM, and the 3-pass filter shown in Figure 1 indicate that the former two conventional techniques suffer from artifacts in the form of blockiness (i.e. in pixel replication) and blurring (i.e. in bilinear interpolation) while the latter three statistical techniques yield similar interpolated images that improve upon the conventional methods. The original 64×64 textured region is shown in Figure 1b).

6. CONCLUSIONS

Statistical techniques using Gibbs random fields models were used to determine an interpolated image from its downsampled version and a number of texture parameters. Estimation of these parameters was carried out in smaller blocks over the image. The adaptive nature of this procedure yields more accurate parameter estimates in the different regions of an image. Consequently, good results may be achieved over smooth or constant areas as well as differently textured regions. Two iterative techniques, that of SA and ICM as well as a non-iterative 3-pass filter based on logic-type operations were implemented. It was found experimentally that the latter produced visually similar results to the former two methods which are both computationally expensive. The 3-pass filter also permits a simple

Proceedings of the 1995 International Conference on Image Processing (ICIP '95)
0-8186-7310-9/95 $10.00

hardware solution for real-time applications. Experimental results also indicate that the statistical techniques yield better results than the conventional linear schemes.

7. REFERENCES

[1] A.J. Parker, R.V. Kenyon, D.E. Troxel, 'Comparison of interpolating methods for image resampling', IEEE Trans. on Medical Imaging, vol. 2, no. 1, pp. 31-39, 1983.

[2] H.H. Hou, H.C. Andrews, 'Cubic splines for image interpolation and digital filtering', IEEE Trans. on Acoustics, Speech, and Signal Processing, vol. 26, no. 6, pp. 508-517, 1978.

[3] R.G. Keys, 'Cubic convolution interpolation for digital image processing', IEEE Trans. on Acoustics, Speech, and Signal Processing, vol. 29, no. 6, pp. 1153-1160, 1981.

[4] G.R. Cross, A.K. Jain, 'Markov random field texture models', IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 5, pp. 25-39, 1983.

[5] M.I. Gurelli, L. Onural, 'Feature-based Analysis of Textured Images Using Relative Frequencies of Gibbs Random Field Clique Realizations', accepted for publication in IEEE Trans. on Systems, Man, and Cybernetics.

[6] H. Derin, H. Elliot, 'Modeling and Segmentation of Noisy and Textured Images using Gibbs Random Fields', IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 9, no. 1, pp. 39-55, 1987.

[7] R.R. Schultz, R.L. Stevenson 'A Bayesian Approach to Image Expansion for Improved Definition', IEEE Trans. on Image Processing, vol. 3 ,no. 3, pp. 233-242, 1994.

[8] S. Lakshmanan, A.K. Jain, Y. Zhong, 'Multi-Resolution Image Representation Using Markov Random Fields', IEEE Proc. ICIP, Vol. I, pp. 855-860, 1994.

[9] L. Onural, N. Herodotou, A.N. Venetsanopoulos, 'Interpolation by Regeneration Using Gibbs Random Field Models', IEEE Workshop on Nonlinear Signal and Image Processing, Halkidiki, Greece, June 1995.

[10] J. Besag, 'Spatial interaction and the statistical analysis of lattice systems', J. Roy. Statist. Soc. B, vol. 36, pp. 192-236, 1974.

[11] Y. Zhao, X. Zhuang, L. Atlas, L. Anderson, 'Parameter Estimation and Restoration of Noisy Images using Gibbs Distributions in Hidden Markov Models', CVGIP: Graphical Models and Image Processing, vol. 54, no. 3, pp. 187-197, 1992.

[12] J.E. Besag 'On the statistical analysis of dirty pictures', J. R. Stat. Soc. Ser. B, vol. 48, pp. 259-302, 1986.

[13] S. Geman, D. Geman, 'Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images', IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 6, no. 6, pp. 721-741, 1984.

[14] S. Kirkpatrick, C.D. Gellatt Jr., M.P. Vecchi, 'Optimization by simulated annealing', IBM Thomas J. Watson Research Center, Yorktown Heights, NY, 1982.

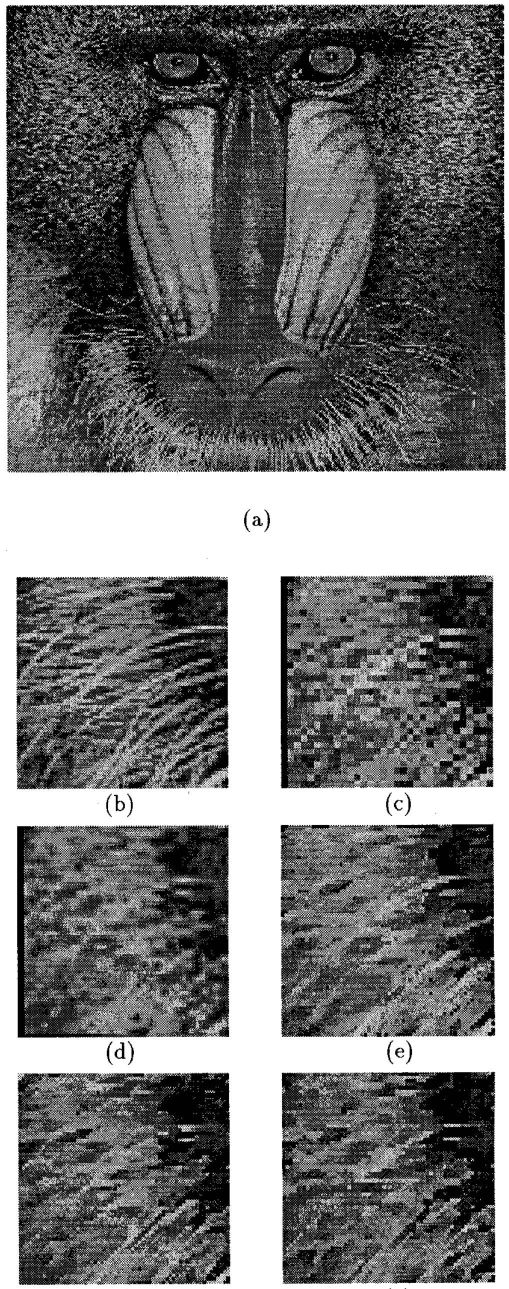

Figure 1: a) Original "baboon" image, b) A section of the "baboon" image, c) Pixel-replication, d) Bilinear interpolation, e) SA f) ICM, g) 3-pass nonlinear filter.

Proceedings of the 1995 International Conference on Image Processing (ICIP '95)
0-8186-7310-9/95 $10.00