Correspondence

Adaptive Image Restoration Using a Generalized Gaussian Model for Unknown Noise

Wai Ho Pun and Brian D. Jeffs

***Abstract*—A model adaptive method is proposed for restorating blurred and noise corrupted images. The generalized p-Gaussian family of probability density functions is used as the approximating parametric noise model. Distribution shape parameters are estimated from the image, and the resulting maximum likelihood optimization problem is solved. An iterative algorithm for data-directed restoration is presented and analyzed.**

I. Introduction

Restoring a degraded image is one of the fundamental problems in image processing. Some of the classical approaches include such widely used algorithms as constrained least squares [1], maximum entropy [2], minimum norm [3], [4], and other related methods. These problems, like the one presented here, are readily solved with simple iterative algorithms that minimize some objective function of the image [3], [7], [10]. However, any given restoration method usually performs best for a particular class of images only. This limitation usually arises from the implicit or explicit assumptions regarding image formation and degradation which are associated with a particular algorithm. The choice of a deterministic algorithm is often equivalent to applying a specific prior model to the noise process or image itself. For example, a least squares restoration is optimal (in the maximum likelihood sense) in the presence of i.i.d. Gaussian noise. In practice, however, these implied image model assumptions are often ignored, and an algorithm is chosen for its convenience. In this correspondence, we propose a method that relaxes these constraining assumptions by adapting a parametric noise model to fit the observed data. This yields improved restorations even when noise statistics are unknown.

Consider the following commonly used linear model for an image degradation process

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{n} \tag{1}$$

where

- $\mathbf{H}$ doubly block Toeplitz system matrix corresponding to a 2-D shift invariant convolutional blur [4];
- $\mathbf{n}$ additive noise vector;
- $\mathbf{x}$ original image;
- $\mathbf{y}$ observed image of length N.

The vectors $\mathbf{x}$, $\mathbf{y}$, and $\mathbf{n}$ are obtained from row or column scanning the original 2-D images.

Given this model, the well-known maximum likelihood (ML) estimate of $\mathbf{x}$ is $\hat{\mathbf{x}}_{\mathrm{ML}} = \mathrm{Arg}\{\max_{\mathbf{x}} f_{\mathbf{y}}(\mathbf{y}|\mathbf{x})\}$, where $f_{\mathbf{y}}(\mathbf{y}|\mathbf{x})$ is the the conditional probability density of $\mathbf{y}$ given true image $\mathbf{x}$. The proposed adaptive method uses information from the observed image to form a constrained parametric estimate of $f_{\mathbf{y}}(\mathbf{y}|\mathbf{x})$ as a member of the generalized p-Gaussian (gpG) distribution family. The ML solution is then found to be the l_p norm minimization of the error residual, where p is the shape parameter associated with $f_{\mathbf{y}}(\mathbf{y}|\mathbf{x})$ [6]. The gpG family of symmetric distributions includes many commonly encountered ones, and even when the noise is not gpG, an approximate gpG model can often be found that yields improved restoration.

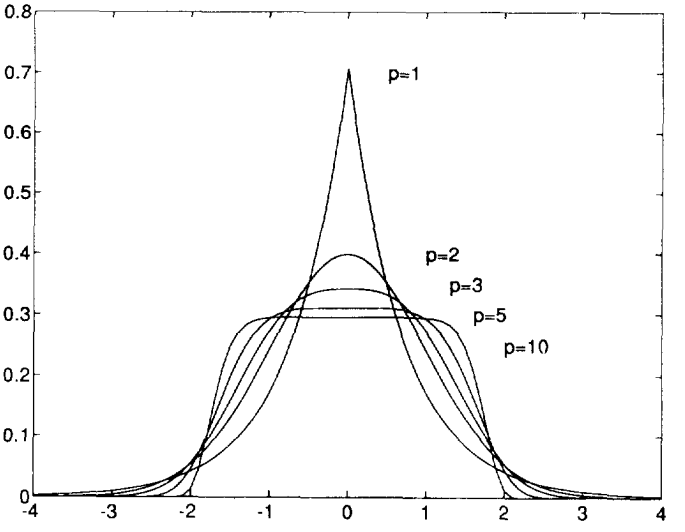


Fig. 1. PDF of the gpG distribution family as a function of p. $\beta = 1$.

The gpG family has been used successfully as a noise model in robust detection and estimation [9] and as an image prior to control edge and region structure in Bayesian estimation of images [8]. The proposed adaptive approach is more closely related to the work of McDonald in partially adaptive estimation for overdetermined regression problems [5]. We extend the approach to the underdetermined, high-dimensionality case of image restoration and introduce a new efficient iterative algorithm to solve the associated optimization problem. This work is also related to the problem of regularized robust image restoration in the presence of outliers, which has been addressed by Zervakis and Kwon and others [11].

II. Theoretical Development

The probability density function of the generalized Gaussian distribution family is defined as [6]

$$f(x; p, \beta) = \frac{p}{2\beta\Gamma\left(\frac{1}{p}\right)} \exp\left\{-\left(\frac{|x|}{\beta}\right)^p\right\}, \quad p > 0 \tag{2}$$

where $\Gamma(\cdot)$ is the standard gamma function. This is a two-sided symmetric density with two distributional parameters p and β that control the shape and standard deviation of the density, respectively. As shown in Fig. 1, this family is very flexible. For example, with $p = 2, \beta = \sqrt{2}$, it becomes a standard normal distribution. For $p = 1$, we have a double exponential, and for $0 < p < 1$, we have heavy tailed distributions, while as $p \to \infty$, the uniform distribution is approximated. Since this distribution family has only two parameters, their estimation can be carried out without extreme difficulties, as shown in Section III-A.

Manuscript received January 16, 1994; revised January 17, 1995. The associate editor coordinating the review of this paper and approving it for publication was Dr. Reginald L. Lagendijk.

The authors are with Brigham Young University, Provo, UT 84602 USA.

IEEE Log Number 9413844.

Assuming the image data $\mathbf{x}$ are deterministic and the noise $\mathbf{n}$ comes from an i.i.d. generalized Gaussian process with unknown parameters, the maximum likelihood solution to (1) is given by

$$\hat{\mathbf{x}}_{\mathrm{ML}} = \mathrm{Arg}\{\max_{\mathbf{x},p,\beta} f_{\mathbf{y}}(\mathbf{y}|\mathbf{x}; p, \beta)\} \tag{3}$$

where the posterior distribution of $\mathbf{y}$ is given by

$$f_{\mathbf{y}}(\mathbf{y}|\mathbf{x}; p, \beta) = \left[\frac{p}{2\beta\Gamma\left(\frac{1}{p}\right)}\right]^N \exp\left\{-\sum_{i=1}^{N}\left[\frac{|n_i|}{\beta}\right]^p\right\}. \tag{4}$$

The noise vector element n_i can be replaced by $y_i - \mathbf{h}_i^T\mathbf{x}$, where $\mathbf{h}_i$ is the vector corresponding to the ith column of $\mathbf{H}$. In addition, taking the logarithm of (4), (3) can be expressed as a minimization of the negative log-likelihood function in terms of $\mathbf{x}$, β, and p

$$\hat{\mathbf{x}}_{\mathrm{ML}} = \mathrm{Arg}\left\{\min_{\mathbf{x},p,\beta}\left\{-N\ln p + N\ln 2\beta\Gamma\left(\frac{1}{p}\right) + \frac{1}{\beta^p}\sum_{i=1}^{N}|y_i - \mathbf{h}_i^T\mathbf{x}|^p\right\}\right\}, \quad p > 0. \tag{5}$$

If both p and β are known, then the maximum likelihood estimate of the original image can be found as [6]

$$\hat{\mathbf{x}}_{\mathrm{ML}} = \mathrm{Arg}\left\{\min_{\mathbf{x}}\sum_{i=1}^{N}|y_i - \mathbf{h}_i^T\mathbf{x}|^p\right\}, \quad p > 0. \tag{6}$$

Notice that the term $\sum_{i=1}^{N}|y_i - \mathbf{h}_i^T\mathbf{x}|^p$ is monotonically related to the l_p norm $\|\mathbf{y} - \mathbf{H}\mathbf{x}\|_p$. Thus, l_p norm minimization yields the maximum likelihood solution to the image restoration problem under the assumption that the observation error can be modeled as a generalized Gaussian distribution for some value of p [6]. Note that if $p = 2$, which corresponds to a Gaussian noise case, the maximum likelihood solution becomes the least squares solution.

III. Algorithm Implementation

A. Parameter Estimation

The maximum likelihood estimate of the image depends on shape parameter p and standard deviation parameter β. For data-directed restoration, p must be estimated from the image, and the ideal approach would be a joint maximum likelihood estimation of $\mathbf{x}$, p, and β as in (5). However, (5) poses a difficult nonlinear optimization problem. We have found that in image restoration, the high dimensionality of the problem makes joint optimization intractable.

The practical approach presented here separates the problems, forming a prior estimate of p from image data, then computing the maximum likelihood restoration of $\mathbf{x}$ given that p. In practice, β can be absorbed into the restoration process as a scale factor. Note that when it is assumed p and β are known (or separately estimated), β does not appear as part of the ML solution of (6). The method proposed in this section normalizes the noise data sample to unit variance so that estimates of p are independent of β. On the other hand, knowing the value of p is fundamental to the model adaptive approach. In fact, choosing a value for p is equivalent to choosing a particular restoration method.

$\mathbf{x}$ and p are coupled in the optimization problem of (5). Therefore, this separate estimation is a suboptimal approach that is, in general, not equivalent to joint estimation, but that has proven to be effective. The algorithm attempts to reduce interaction between $\mathbf{x}$ and p by estimating p from a noise sample $\hat{\mathbf{n}}$ extracted from $\mathbf{y}$. A pixel masking operation is used to reduce influence from significant features of $\mathbf{Hx}$. In images with moderate signal-to-noise ratios (SNR's), the low-variance regions often correspond to constant image values where fluctuations are due to additive noise rather dominant edge activity. A localized sample estimate of the variance is formed, and pixels whose local variance falls below a threshold are extracted from the image. The difference between these pixels and the corresponding local means of their constant surround serves as the extracted noise estimate, which is computed as follows:

$$\hat{\mathbf{n}} = \{y_i - \hat{\mu}_{y_i} | \forall\, i \in M\}, \tag{7}$$

$$\hat{\mu}_{y_i} = \frac{1}{K}\sum_{j\in W_i} y_j, \quad M = \{i | \forall\, i \text{ s.t. } \hat{\sigma}^2_{y_i} < T\},$$

$$\hat{\sigma}^2_{y_i} = \frac{1}{K-1}\sum_{j\in W_i}\left[y_j^2 - \hat{\mu}^2_{y_i}\right]$$

where

- W_i set of indices for all pixels contained in a local 2-D, K pixel window centered on the ith pixel y_i;
- T variance threshold;
- M noise mask set containing the indices of all pixels to be included in $\hat{\mathbf{n}}$, which is the extracted vector of approximate noise samples.

W_i is usually chosen to be a 3×3 or 5×5 window centered on the ith pixel, and T is set to be some percentage of the median pixel variance in $\mathbf{y}$.

An estimate $\hat{f}(k)$ of the probability density for $\hat{\mathbf{n}}$ is formed as follows:

1) $\hat{\mathbf{n}}$ is normalized to unit variance and zero mean to remove dependence on β.
2) A histogram $b(k)$ of the normalized $\hat{\mathbf{n}}$ is computed.
3) Each bin k of the histogram is normalized so that the total area under the curve is 1.

$$\hat{f}'(k) = \frac{b(k)\Big/\left(\sum_{j=1}^{J} b(j)\Delta - \mu_{\hat{n}}\right)}{\sigma_{\hat{n}}} \tag{8}$$

where

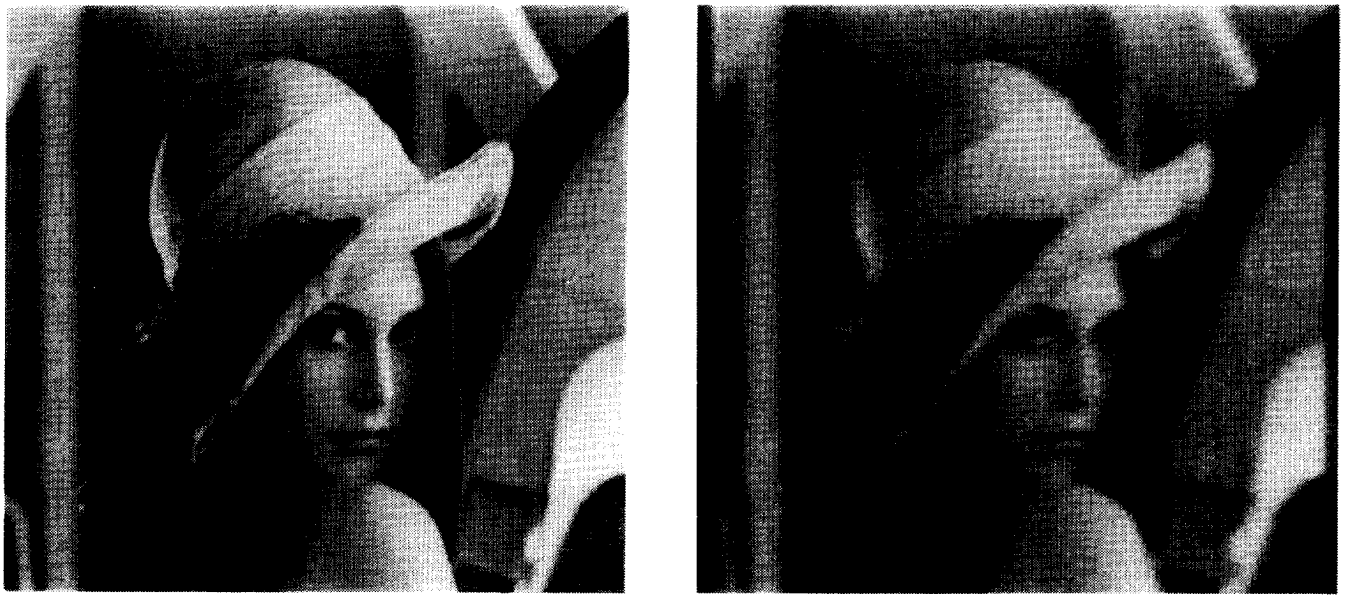
- $b(k)$ number of counts in the kth bin of the histogram;
- J length of $\mathbf{n}$;
- Δ width of the bin;
- $\mu_{\hat{n}}$ sample mean of $\mathbf{n}$;
- $\sigma_{\hat{n}}$ sample standard deviation of $\hat{\mathbf{n}}$;
- $\hat{f}'(k)$ kth normalized bin.

4) Since $\hat{\mathbf{n}}$ may not be gpG and could be nonzero mean or skewed, the gpG density function provides a better approximation if the mode of the sample distribution $\hat{f}'(k)$ is aligned to the mode of the gpG model as follows: $\hat{f}(k) = \hat{f}'(k - k_0)$, where k_0 is the index of the maximum bin in $\hat{f}'(k)$.

A minimum distance estimator is used to estimate the shape parameter p.

$$\hat{p} = \mathrm{Arg}\left\{\min_{p}\sum_{k=1}^{N}|f(x_k; p, \beta) - \hat{f}(k)|^2\right\} \tag{9}$$

where $f(x_k; p, \beta)$ is the gpG density function evaluated at the N histogram bin midpoints x_k with $\beta = \sqrt{\frac{\Gamma(1/p)}{\Gamma(3/p)}}$ (yielding unit variance.) Since (9) is convex in p and N is typically small, $\hat{p}$ can be computed efficiently with any standard nonlinear optimization code. It can be shown that if in (7), $\hat{\mu}_{y_i} = [\mathbf{Hx}]_i$ (i.e., the window mean is equal to the noise free blurred pixel), then (9) provides an unbiased estimate of p. Clearly, this will not be the case in general, so the estimate is biased. To minimize this effect, we seek pixels where $[\mathbf{Hx}]_i$ is constant over the local window. To address this problem, T is kept as small as possible while still retaining enough pixels in mask M to provide a low-variance estimate of p.



(a) (b)

(c)

Fig. 2. (a) Original 256 × 240 unblurred image; (b) 1-D horizontal motion blurred image with additive gpG noise ($p = 8$) at 15 dB SNR; (c) pixel mask used for extraction of $\hat{\mathbf{n}}$ with threshold $T = 1.6$; (d) best result obtained using adaptive restoration method on Fig. 2(b) with estimated $\hat{p} = 5.0$; (e) best result obtained using least squares method ($p = 2$) on Fig. 2(b).

B. Iterative Solution

Using the steepest descent approach, the objective function $\Phi(\mathbf{x}) = \sum_{i=1}^{N} |y_i - \mathbf{h}_i^T \hat{\mathbf{x}}_k|^{\hat{p}}$, as given in (6), can be minimized by the following iterative procedure:

$$\begin{aligned}\hat{\mathbf{x}}_{k+1} &= \hat{\mathbf{x}}_k - \frac{\alpha_k}{\hat{p}} \nabla_{\mathbf{x}} \Phi(\mathbf{x})|_{\hat{\mathbf{x}}_k} \\ &= \hat{\mathbf{x}}_k + \alpha_k \sum_{i=1}^{N} (y_i - \mathbf{h}_i^T \hat{\mathbf{x}}_k) |y_i - \mathbf{h}_i^T \hat{\mathbf{x}}_k|^{\hat{p}-2} \mathbf{h}_i, \quad \hat{p} \geq 1 \\ &= \hat{\mathbf{x}}_k + \alpha_k \mathbf{H}^T \mathbf{U}_k (\mathbf{y} - \mathbf{H}\hat{\mathbf{x}}_k) \quad \hat{p} \geq 1 \end{aligned} \tag{10}$$

where α_k is a constant that regulates the step size, and $\mathbf{U}_k$ is a diagonal matrix of the same size as $\mathbf{H}$ with diagonal elements $u_{k_{ii}} = |y_i - \mathbf{h}_i^T \hat{\mathbf{x}}_k|^{\hat{p}-2}$. The algorithm may be initialized with $\hat{\mathbf{x}}_0 = \mathbf{H}^T \mathbf{y}$.

The iterative algorithm of (10) is much better suited to image restoration than general-purpose nonlinear optimization codes or the deconvolution approach presented in the work of Pham and De Figueiredo [6]. Their method optimizes with respect to one element of $\mathbf{x}$ at a time and performs a 1-D line search at each iteration. Each element is visited repeatedly until reduction in the objective is no longer realized. This approach is effective but entirely intractable for the high-dimensional problem of image restoration where there are often on the order of 10^5 pixels in $\mathbf{x}$. Equation (10) operates on the entire vector in each iteration.

C. Computational Efficiency

Assuming $\mathbf{x}$ and $\mathbf{y}$ correspond to square images of size $M \times M$, each iteration of (10) involves $O\{2(M^4 + M^2)\}$ operations. An

(d)

(e)

Fig. 2. *(Cont'd).*

improvement in computational efficiency and a reduction in memory usage are possible if we recognize that $\mathbf{H}$ is not an arbitrary matrix and that multiplication by $\mathbf{H}$ and $\mathbf{H}^T$ represent 2-D convolution and correlation, respectively. Equation (10) may be rewritten as

$$\hat{\mathcal{X}}_{k+1} = \hat{\mathcal{X}}_k + \alpha \mathcal{H} \otimes |\mathcal{Y} - \mathcal{H} * \mathcal{X}_k|^{\hat{p}-1} \otimes \text{signum}\{\mathcal{Y} - \mathcal{H} * \mathcal{X}_k\} \quad (11)$$

where $\mathcal{X}$, $\mathcal{Y}$, and $\mathcal{H}$ are the 2-D image matrices corresponding to $\mathbf{x}$, $\mathbf{y}$, and $\mathbf{H}$, respectively; $\otimes$ and $*$ indicate 2-D deterministic space domain correlation and convolution and $\cdot$, $|\cdot|^{\hat{p}-1}$, and signum{ } are element-by-element matrix operations of multiplication, absolute exponentiation, and sign retrieval, respectively. Assuming $\mathcal{H}$ is $N \times N$, (11) involves $O\{2(M^2N^2 + M^2)\}$ operations. Since the point-spread function usually has a small finite region of support, we note that $N \ll M$; therefore, the computational load for (11) can be many orders of magnitude less than for a direct implementation of (10). Results presented in Section IV were computed using (11), and we have found this algorithm to be sufficiently efficient for restoration of 1024 × 1024 pixel images on a modest desktop workstation computer (e.g., a Macintosh Quadra 700 running MATLAB) in a few minutes.

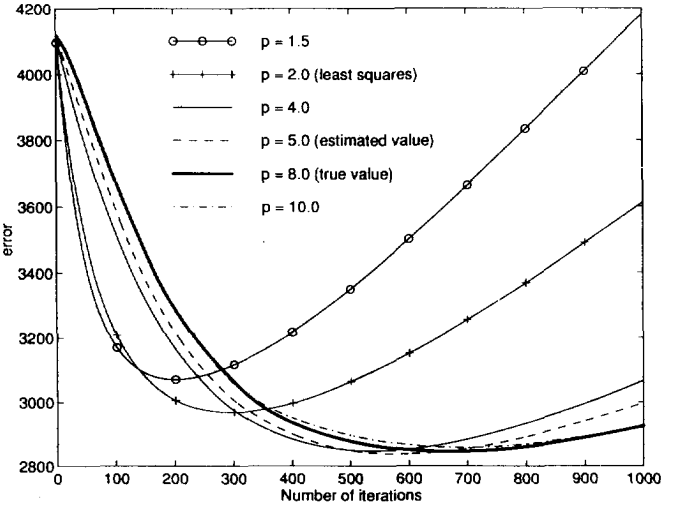


Fig. 3. Error curves (Frobenius norm of the error image $\|\mathbf{x} - \hat{\mathbf{x}}_k\|_F$) as a function of the number of iterations for restoring Fig. 2(b) using different values of p.

D. Convergence Analysis

Consider the following mapping based on the l_2 vector-matrix of the iteration from (10) $g(\mathbf{x}) = \mathbf{x} + \alpha(\mathbf{H}^T\mathbf{U}\mathbf{y} - \mathbf{H}^T\mathbf{U}\mathbf{H}\mathbf{x})$.

$$\begin{aligned}\|g(\mathbf{x}_1) - g(\mathbf{x}_2)\| &= \|\mathbf{x}_1 + \alpha(\mathbf{H}^T\mathbf{U}\mathbf{y} - \mathbf{H}^T\mathbf{U}\mathbf{H}\mathbf{x}_1) \\ &\quad - \mathbf{x}_2 - \alpha(\mathbf{H}^T\mathbf{U}\mathbf{y} - \mathbf{H}^T\mathbf{U}\mathbf{H}\mathbf{x}_2)\| \\ &\leq \|I - \alpha\mathbf{H}^T\mathbf{U}\mathbf{H}\|\,\|\mathbf{x}_1 - \mathbf{x}_2\|. \end{aligned} \quad (12)$$

If $0 \leq \|I - \alpha\mathbf{H}^T\mathbf{U}\mathbf{H}\| \leq 1$, then $g(\mathbf{x})$ is a contraction mapping, and thus (10) is convergent under this constraint. Applying the definition of the matrix norm to $\|I - \alpha\mathbf{H}^T\mathbf{U}\mathbf{H}\|$ yields an equivalent condition for convergence

$$0 \leq \max_i\{|1 - \alpha\lambda_i|\} \leq 1 \quad (13)$$

where λ_i's are the eigenvalues of the matrix $\mathbf{H}^T\mathbf{U}\mathbf{H}$. Furthermore, the inequalities in (13) are always satisfied if $0 < \alpha < \frac{2}{\max_i |\lambda_i|}$. We may obtain a more useful bound on α by noting that $(\max_i |\lambda_i|) = \|\mathbf{H}^T\mathbf{U}\mathbf{H}\| \leq \|\mathbf{H}^T\|^2\|\mathbf{U}\|$. Recalling that α and $\mathbf{U}$ are updated each iteration and that $\mathbf{U}$ is diagonal so that its norm is equal to the largest diagonal term, we have assured convergence for α_k in the range

$$0 < \alpha_k < \frac{2}{\|\mathbf{H}^T\|^2 \max_i |u_{k_{ii}}|}. \quad (14)$$

We now consider the form of the solution at convergence. At this point, the gradient term of (10) is equal to zero, yielding $\mathbf{H}^T\mathbf{U}(\mathbf{y} - \mathbf{H}\hat{\mathbf{x}}) = 0$, which leads to

$$\hat{\mathbf{x}} = (\mathbf{H}^T\mathbf{U}\mathbf{H})^{-1}\mathbf{H}^T\mathbf{U}\mathbf{y}. \quad (15)$$

We note that for $p = 2$, $\mathbf{U} = \mathbf{I}$, and the solution is the familiar inverse filter $\hat{\mathbf{x}} = \mathbf{H}^{\dagger}\mathbf{y}$. This is known to suffer from noise amplification problems due to small eigenvalues of $\mathbf{H}$. For $p \neq 2$, the elements of $\mathbf{y}$ are weighted by $\mathbf{U}$, but the $(\mathbf{H}^T\mathbf{U}\mathbf{H})^{-1}$ term still leads to noise amplification.

(a) (b)

Fig. 4. (a) One-dimensional horizontal motion blurred image of Fig. 2(a) with additive χ^2 shape noise at 15 dB SNR; (b) best result obtained using adaptive restoration method on Fig. 4(a) with estimated $\hat{p} = 1.2$.

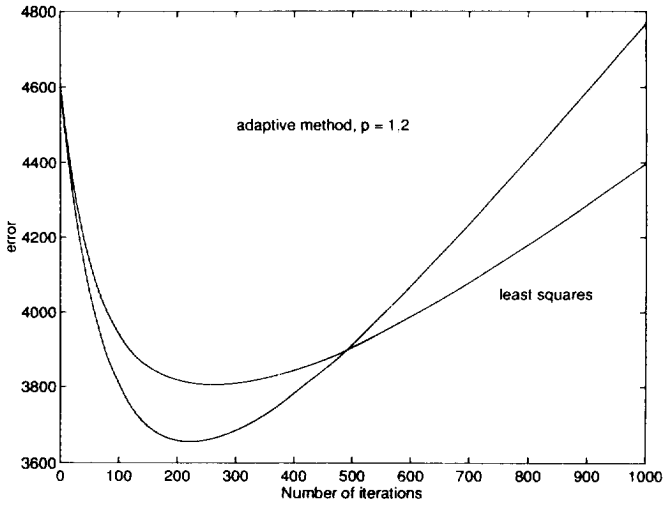


Fig. 5. Error curves (Frobenius norm of the error image $\|\mathbf{x} - \hat{\mathbf{x}}_k\|_F$) as a function of the number of iterations for restoring Fig. 4(a) using adaptive method and least squares method, respectively.

As in other iterative restoration techniques, termination of the sequence prior to convergence yields the best compromise between minimizing the cost function and regularizing the noise amplification effect of the inverse filter [7]. The desired solution is at the minimum error point, but unfortunately, we do not usually have a copy of the true image to permit computation of the error as a function of k. We find that a simple manual procedure works well. The operator observes $\hat{\mathbf{x}}_k$ at regular intervals until the onset of subjective image degradation by increased "graininess" is visually noted. This manual stopping point is usually close to the minimum error point solution.

IV. Results

In this section, we illustrate algorithm performance by presenting restoration results of several synthetic examples. Fig. 2(a) shows the original 256 × 240 pixel uncorrupted image. In all of the examples presented here, this image was blurred by the point spread function $h[m,n] = \frac{1}{16}[1\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 1]$, which corresponds to a 1-D horizontal motion blur.

Fig. 2(b) shows the blurred image with gpG noise $p = 8$ at an SNR of 15 dB. A threshold setting of $T = 1.6$ yields $\hat{p} = 5.0 \pm 0.1$. Fig. 2(c) shows noise mask M and illustrates how (7) ensures that $\hat{\mathbf{n}}$ consists only of pixels from relatively flat intensity regions. Fig. 2(d) is the resulting adaptive ($\hat{p} = 5.0$) restoration of Fig. 2(b), whereas 2(e) is a least squares result presented here for comparison. Both approaches reduce blur and noise level, but the adaptive algorithm (Fig. 2(d)) exhibits lower noise level in flat regions and some improvement in edge preservation.

Though the visual differences between Fig. 2(d) and 2(e) are subtle, Fig. 3 presents a quantitative performance comparison that more clearly shows the improvement obtained by the adaptive approach. Fig. 2(b) was used as the image data source in computing each of the curves. Fig. 2(d) and 2(e) correspond to the minimum error points of the $p = 5$ and $p = 2$ curves, respectively. Terminating the algorithms near the minimum provides the needed regularization to avoid noise amplification.

Fig. 3 also illustrates algorithm sensitivity to estimation error in $\hat{p}$ and suggests the possible range of improvement over a simple least squares approach. It is notable that though the true value was $p = 8$, the minimum point on the $p = 5.0$ curve is the lowest error result. The small difference between it and results for $p = 8$ or $p = 10$ indicates less sensitivity in this region and agrees with the observation that the shape of the gpG density function changes little between large values of p (see Fig. 1). We note that in our experiments with Gaussian noise, p was always estimated to be $\hat{p} = 2, \pm 0.1$; therefore, the adaptive method had no performance penalty if the unknown noise happened to be Gaussian.

The following results are from a number of experiments designed to evaluate parameter estimation performance. Tables I and II show how the estimated value $\hat{p}$ is affected by changes in SNR and mask threshold level for the cases of gpG noise with $p = 4$ and $p = 8$. These results suggest that very high SNR levels hamper estimating noise distribution parameters. This is likely due to the difficulty in

TABLE I
ESTIMATED VALUES OF P USING (7)–(9) FOR VARIOUS SNR AND THRESHOLD T, TRUE $P = 4$

$p = 4$	SNR			
T	3dB	10dB	20dB	30dB
0.4	1.8	1.6	2.8	2.4
0.8	2.8	3.0	3.2	2.2
1.2	3.4	3.4	3.2	2.0
1.6	3.6	3.4	3.0	1.8

TABLE II
ESTIMATED VALUES OF P USING (7)–(9) FOR VARIOUS SNR AND THRESHOLD T, TRUE $P = 8$

$p = 8$	SNR			
T	3dB	10dB	20dB	30dB
0.4	2.0	1.4	3.4	1.6
0.8	3.6	3.8	4.6	1.2
1.2	5.2	4.8	4.2	1.2
1.6	5.6	5.0	3.8	1.2

extracting a representative noise sample $\hat{\mathbf{n}}$, which is not dominated by image components. There is also a bias in the estimates, but for this image, a threshold level between 1.2 and 1.6 provides the best performance. Considering the implications of Fig. 3, this bias is not serious for p larger than about 3.

Another important consideration is whether improvement is possible when the noise distribution is not a member of the gpG family. The following example illustrates that flexibility provided by the shape parameter enables the gpG model to approximate the noise distribution. In this case, the noise was distributed χ^2 (Chi squared) with three degrees of freedom, which is neither gpG nor symmetric. Noise data was scaled such that the resulting corrupted image had a 15-dB SNR. The estimated value for p using the method outlined in Section III-A is 1.2. Fig. 4(a) shows the blurred image with χ^2 noise, whereas Fig. 4(b) shows the restored image. Fig. 5 presents a comparison of the error curves for the adaptive restoration and least squares. Note that the minimum error point for $p = 1.2$ is significantly below that of $p = 2$. The encouraging observation here is that even in the presence of noise clearly not well modeled by a gpG distribution, the adaptive approach offers an improvement over methods that implicitly assume a strict Gaussian model.

REFERENCES

[1] B. R. Hunt, "The application of constrained least squares estimation to image restoration by digital computer," *IEEE Trans. Comput.*, vol. C-22, no. 9, Sept. 1973.

[2] S. F. Gull and J. Skilling, "Maximum entropy method in image processing," *Proc. IEEE*, vol. 131, Pt. F, no. 6, pp. 646–659, 1984.

[3] Y. Censor, "Finite series-expansion reconstruction methods," *Proc. IEEE*, vol. 71, pp. 409–418, 1983.

[4] A. K. Jain, *Fundamentals of Digital Image Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1989.

[5] J. B. McDonald, "Partially adaptive estimation of ARMA time series models," *Int. J. Forecasting*, vol. 5, pp. 217–230, 1989.

[6] T. T. Pham and R. J. P. deFigueiredo, "Maximum likelihood estimation of a class of non-Gaussian densities with application to l_p deconvolution," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, no. 1, pp. 73–82, Jan. 1989.

[7] R. L. Lagendijk and J. Biemond, *Iterative Identification and Restoration of Images*. Boston: Kluwer, 1991.

[8] C. Bouman and K. Sauer, "A generalized Gaussian model for edge-preserving MAP estimation," *IEEE Trans. Image Processing*, vol. 2, pp. 296–310, July 1993.

[9] S. A. Kassam, *Signal Detection in Non-Gaussian Noise*. New York: Springer-Verlag, 1988.

[10] A. K. Katsaggelos, "Iterative image restoration algorithms," *Opt. Eng.*, vol. 28, no. 7, pp. 735–748, July 1989.

[11] M. E. Zervakis and T. M. Kwon, "Robust estimation techniques in regularized image restoration," *Opt. Eng.*, vol. 31, no. 10, pp. 2174–2192, Oct. 1992.

Scalable Data Parallel Algorithms for Texture Synthesis Using Gibbs Random Fields

David A. Bader, Joseph JáJá, and Rama Chellappa

***Abstract*—This correspondence introduces scalable data parallel algorithms for image processing. Focusing on Gibbs and Markov random field model representation for textures, we present parallel algorithms for texture synthesis, compression, and maximum likelihood parameter estimation, currently implemented on Thinking Machines CM-2 and CM-5. Use of fine-grained, data parallel processing techniques yields real-time algorithms for texture synthesis and compression that are substantially faster than the previously known sequential implementations. Although current implementations are on Connection Machines, the methodology presented here enables machine-independent scalable algorithms for a number of problems in image processing and analysis.**

I. INTRODUCTION

Random fields have been successfully used to sample and synthesize textured images [4]–[7], [9]. Texture analysis has applications in image segmentation and classification, biomedical image analysis, and automatic detection of surface defects. Of particular interest are the models that specify the statistical dependence of the gray level at a pixel on those of its neighborhood. There are several well-known algorithms describing the sampling process for generating synthetic textured images and algorithms that yield an estimate of the parameters of the assumed random process given a textured image. Impressive results related to real-world imagery have appeared in the literature [3], [5]–[8]. However, all these algorithms are quite computationally demanding because they typically require on the order of Gn^2 arithmetic operations per iteration for an image of size

Manuscript received October 9, 1993; revised December 17, 1994. This work was supported by NASA Graduate Student Researcher Fellowship NGT-50951, by the National Science Foundation under Grant CCR-9103135 and NSF HPCC/GCAG Grant BIR-9318183, and by the U.S. Air Force under Grant F49620-92-J0130. The associate editor coordinating the review of this paper and approving it for publication was Dr. Hsueh-Ming Hang.

The authors are with the Department of Electrical Engineering and the Institute for Advanced Computer Studies, University of Maryland, College Park, MD 20742 USA.

IEEE Log Number 9413842.