

A Signomial Programming Approach for Binary Image Restoration by Penalized Least Squares

Yijiang Shen, Edmund Y. Lam, *Senior Member, IEEE*, and Ngai Wong, *Member, IEEE*

Abstract—The authors present a novel optimization approach, using signomial programming (SP), to restore noise-corrupted binary and grayscale images. The approach requires the minimization of a penalized least squares functional over binary variables, which has led to the design of various approximation methods in the past. In this brief, we minimize the functional as a SP problem which is then converted into a reversed geometric programming (GP) problem and solved using standard GP solvers. Numerical experiments show that the proposed approach restores both degraded binary and grayscale images with good accuracy, and is over 20 times faster than the positive semidefinite programming approach.

Index Terms—Binary image restoration, geometric programming (GP), optimization, positive semidefinite (PSD) programming, signomial programming (SP).

I. INTRODUCTION

DIGITAL images are imperfect representations of scenes in the real world due to degradations such as blur, noise and rounding errors. One special case is when the true scenery is binary but is degraded by an additive noise, such as the Gaussian or salt-and-pepper noise. Restoration of these noisy binary images is necessary in various applications, such as stellar astronomy, fingerprint recognition, automated document handling etc. Numerous attempts have been made to deal with binary image restoration. Some restoration schemes require prior knowledge about the images, such as the weighted mean-square error (WMSE) method by Boyd and Meloche [1] developed to restore binary images containing thin objects. This is a generalization of the average mean-square error (AMSE) method by Meloche and Zamar [2]. In the case where the object of interest occupies only a small part of the image, AMSE is not a good restoration criterion. The inferential procedure developed by Hitchcock and Glasbey [3], which identifies a statistical model for digital image data, works for images that contain blob-like and filamentous objects. Meanwhile, Neifeld *et al.* [4] included prior knowledge concerning local correlations among pixel values into the Viterbi-based restoration process. Another algorithm that restores binary images degraded by white Gaussian noise, is the pulse-coupled neural network (PCNN) proposed by Gu *et al.* [5]. Common to many

of these methods is that the noisy image formation can be modeled as

$$g(x_1, x_2) = f(x_1, x_2) + n(x_1, x_2) \quad (1)$$

with x_1 and x_2 being the two dimensional coordinates. $f(x_1, x_2)$ and $g(x_1, x_2)$ represent the true and the degraded images, respectively, and $n(x_1, x_2)$ is the additive noise. This is also the model we use in our work.

The growing interest of convex optimization techniques, however, provides other alternatives to restore degraded binary images, including, for example, a convergent method provided by Chan *et al.* [6], which finds a minimizer of the total-variational functional to restore binary images and an iterative quadratic programming algorithm [7], as a special case of convex optimization, to restore blurry and noisy binary images. A well understood criterion for image restoration is to minimize a penalized least-squares (PLS) problem

$$z_p = \sum_i (f_i - g_i)^2 + \lambda \sum_{j \in \Omega_i} (f_i - f_j) \quad (2)$$

in which f_i and g_i carry corresponding pixel values in $f(x_1, x_2)$ and $g(x_1, x_2)$, and Ω_i is a bounded neighborhood of pixel i . The penalty term incorporates nearest neighbor interactions aiming at smoothing the image. The combinatorial nature of binary image denoising has been noted, and is incorporated in the PLS problem to restore degraded binary images in [8], [9]. Basically, for each pixel position i of an image, the true pixel value y_i can take on either of two known prototype values u_1 and u_2 , while in practice, the observed g_i 's take on a range of grayscale values due to blurring and noise. To restore a discrete-valued image function represented by the vector $y \in \{-1, +1\}^n$ from the measurement $g = \{g_i\}$, where n is the number of pixels in the image, we minimize the functional

$$z(y) = \frac{1}{4} \sum_i ((u_2 - u_1)y_i + u_2 + u_1 - 2g_i)^2 + \frac{\lambda}{2} \sum_{\langle i, j \rangle} (y_i - y_j)^2 \quad (3)$$

in which λ is the smoothness parameter, and u_1 and u_2 are the binary values representing black and white pixels, respectively. The second term sums over all pairwise adjacent pixels on the regular image grid in both vertical and horizontal directions, that is, over the 4-neighborhood of every pixel. The first term in (3) is a data fitting term and is minimized when y_i takes either $+1$ or -1 , whichever is closer to g_i . The data fitting term becomes $\sum_i (u_2 - g_i)^2$ for those y_i 's that take value $+1$, and $\sum_i (u_1 - g_i)^2$

Manuscript received May 15, 2007; revised August 10, 2007. This work was supported in part by the Research Grants Council of the Hong Kong SAR, China, under Project HKU 7164/03E and Project 7139/06E. This paper was recommended by Associate Editor E. A. B. da Silva.

The authors are with the Department of Electrical and Electronic Engineering, The University of Hong Kong, Hong Kong (e-mail: elam@eee.hku.hk).

Digital Object Identifier 10.1109/TCSII.2007.907751

otherwise. The second term is a smoothness term which guarantees that the minimization of (3) will identify those pixels surrounded by their binary counterparts as noise and assign them the same pixel value as their neighbors.

Equation (3) is a combinatorial problem, but can be relaxed to a convex optimization problem, which is then solved using positive semidefinite (PSD) programming [10]. The computation time however quickly grows with the number of variables in the problem such that the restoration of binary images with size around 100×100 is impractical. The optimization process is further explored in [11] to restore binary images degraded by blur and noise, where an overlapping method over image blocks is applied to decrease computation complexity. Yet, it still may take several hours to restore a binary image with a size around 100×100 .

The development of geometric programming (GP) has made a much faster restoration of binary images possible. New solution methods of GP can solve even large-scale problems extremely efficiently and reliably with reasonable speed. More information about GP can be found in [12], [13]. In this brief, we aim to deal with restoration of noisy binary images using the closely-related signomial programming (SP) within much faster time than when the PSD programming is applied. Section II describes the restoration problem and makes a brief introduction to GP and SP. In Section III, the conversion of the SP problem into a reversed GP is discussed, and thus the SP problem is solved using available GP solvers. Experimental results are given in Section IV. Finally, Section V draws some concluding remarks and provides suggestions for future work.

II. BINARY IMAGE RESTORATION USING GP

GP is a mathematical optimization problem characterized by objective and constraint functions that have a special form. The basic approach in GP modeling is to attempt to express a practical problem in its standard format. Let X_1, \dots, X_n denote n real positive variables, and $X = (X_1, \dots, X_n)$ a vector with components X_i , a generalized geometric program (GGP) is an optimization problem of the form

$$\begin{aligned} & \text{minimize} && f_0(X) \\ & \text{subject to} && f_i(X) \leq 1, \quad i = 1, \dots, m \\ & && h_i(X) = 1, \quad i = 1, \dots, p \end{aligned} \quad (4)$$

where h_1, \dots, h_p are monomials and f_0, \dots, f_m are generalized posynomials [13]. Since any posynomial is also a generalized posynomial, any GP is also a GGP. A signomial is a function with the same form as a posynomial, but the coefficients are allowed to be negative. SP is a generalization of a geometric program, which has the form of a GGP in (4), but the objective and constraint functions can be signomials.

Equation (3) can be rewritten as

$$\begin{aligned} z(y) = \sum_i \left(\left(\frac{u_2 - u_1}{2} \right) \left(y_i + \frac{u_2 + u_1}{u_2 - u_1} \right) - g_i \right)^2 \\ + \frac{\lambda}{2} \sum_{\langle i,j \rangle} (y_i - y_j)^2. \end{aligned}$$

Without loss of generality, we assign $u_1 = 1$ and $u_2 = 3$ to represent the binary values indicating white and black pixels respectively. By scaling g_i accordingly, the data fitting term can be simplified as $\sum_i (y_i + 2 - g_i)^2$. If we replace $y_i + 2$ with x_i , (3) now becomes

$$\begin{aligned} z(x) &= \sum_i (x_i - g_i)^2 + \frac{\lambda}{2} \sum_{\langle i,j \rangle} (x_i - x_j)^2 \\ &= \sum_i x_i^2 + \sum_i g_i^2 + \frac{\lambda}{2} \sum_{\langle i,j \rangle} (x_i^2 + x_j^2) \\ &\quad - 2 \sum_i x_i g_i - \lambda \sum_{\langle i,j \rangle} x_i x_j. \end{aligned} \quad (5)$$

Because $y \in \{-1, +1\}^n$ and $x_i = y_i + 2$, now $x \in \{1, 3\}^n$, ensuring the positivity of the variables in the GP problem. Let

$$\begin{aligned} f_{01}(x) &= \sum_i x_i^2 + \sum_i g_i^2 + \frac{\lambda}{2} \sum_{\langle i,j \rangle} (x_i^2 + x_j^2) \\ f_{02}(x) &= 2 \sum_i x_i g_i + \lambda \sum_{\langle i,j \rangle} x_i x_j \end{aligned}$$

the minimization of $z(x)$ in (5) thus requires the computation of the following optimization problem:

$$\begin{aligned} & \text{minimize} && f_{01}(x) - f_{02}(x) \\ & \text{subject to} && x_i = 1 \text{ or } 3 \end{aligned} \quad (6)$$

with $x_i = 1$ being a black pixel and $x_i = 3$ a white one. This optimization problem is not in the standard GGP format, which means it cannot be solved readily with available GP solvers.

We presented some preliminary results in [14] to restore noisy images by solving a simplified version of (5) with

$$z'(x) = \sum_i (x_i - g_i)^2$$

which ignores the second term in (5). It should be noted that $z'(x)$ is in the form an ordinary least squares problem with binary variables. This approach gives fast and accurate restoration results when the degraded images are grayscale. Since x_i has to take binary values 1 or 3, the numerical value of $z'(x)$ will be minimized when x_i takes on values that are nearest to one of the binary values of g_i . Therefore, when the noise is not large enough to cross the midpoint between the two values of g_i , the proposed simplified SP approach will produce satisfactory restoration results. However, when it restores degraded images that are still binary, (which often is the case, when salt-and-pepper noise degrades a binary image), according to the analysis above, after the minimization of $z'(x)$, the restoration will fail because the restored pixel values would essentially be those from the noisy binary image. This failure is caused by ignoring the correlation between neighboring pixels when the smoothness term in (5) is not included in the restoration process. In this brief, we minimize (6) to restore both binary and grayscale noise-corrupted binary images. By including the smoothness term in (5), when a black pixel is surrounded by white pixels, or

vice versa, it should be recognized as noise. Thus, minimizing $z(x)$ will still be effective because the increase in the data fitting term will be compensated by the decrease in the smoothness term, thus restoring the degraded binary image by penalizing isolated pixels. It should be noted that the minimization of $z(x)$ instead of $z'(x)$ will also help enhance the quality of the restored binary image out of a degraded grayscale image by including not only the data fitting term but also the smoothness term.

III. SOLVING THE OPTIMIZATION PROBLEM

The optimization problem in (6) has the format of a mixed integer SP and we first relax the integer constraints in (6) to inequality constraints $x_i \leq 3$, $x_i \geq 1$ and (6) is turned to a SP problem. This relaxation of the integer constraints will not adversely affect the solution since minimizing $z(x)$ in (5) requires x_i to be near 1 or 3, and the integer value of x_i can be obtained by thresholding. There are several ways to solve a SP. The first approach is a standard branch and bound technique. This technique does not utilize the special structure of SP, and thus is not widely used. Another approach is to convert it into a complementary GP, which allows upper bound constraints on the ratio between two posynomials, and then applies a monomial approximation iteratively. In this brief, we convert the SP into a Reversed GP, and then apply a monomial approximation to solve the problem. Reversed GP refers to the minimization of a posynomial subject to both upper and lower bound inequality constraints. More information about the branch and bound technique, the complementary GP and the Reversed GP can be found in [15].

In (6), the monomial terms with negative multiplicative coefficients $f_{02}(x)$ are separated from those monomial terms with positive multiplicative coefficients $f_{01}(x)$. We introduce an auxiliary variable $t \geq 0$ and convert the objective to the minimization of t , with an additional constraint

$$f_{01}(x) - f_{02}(x) \leq t$$

which may be written as

$$f_{01}(x) \leq s \leq f_{02}(x) + t$$

where $s \geq 0$ is another auxiliary variable. The SP problem is now converted into a Reversed GP, and can be further relaxed as

$$\begin{aligned} & \text{minimize} && t \\ & \text{subject to} && s^{-1}f_{02}(x) + s^{-1}t \geq 1 \\ & && s^{-1}f_{01}(x) \leq 1 \\ & && x_i \leq 3 \\ & && x_i \geq 1. \end{aligned} \quad (7)$$

The first and the third term in $f_{01}(x)$ can be combined. For the four corner pixels, the exponential index of x is $2 + \lambda$, while for the other boundary pixels, the exponential index of x is $2 + 1.5\lambda$. The exponential index of the rest of x is $2 + 2\lambda$. Thus, there are $i + 2$ terms in the second constraint in (7). If we can approximate the posynomial $f_{02}(x) + t$ with a monomial, then a lower

bound on $f_{02}(x) + t$ becomes an upper bound on a monomial, which is allowed in standard form GP. We can use a simple approximation based on the geometric inequality that leads to the development of GP, that the arithmetic mean is greater than or equal to the geometric mean. Therefore

$$\begin{aligned} f_{02}(x) + t &= 2 \sum_i x_i g_i + \lambda \sum_{\langle i,j \rangle} x_i x_j + t \\ &\geq \prod_i \left(\frac{2g_i x_i}{\alpha_i} \right)^{\alpha_i} \times \left(\frac{t}{\alpha_t} \right)^{\alpha_t} \times \prod_{\langle i,j \rangle} \left(\frac{\lambda x_i x_j}{\alpha_{\langle i,j \rangle}} \right)^{\alpha_{\langle i,j \rangle}} \\ &= f'_{02}(x) \end{aligned} \quad (8)$$

where $\sum_i \alpha_i + \alpha_t + \sum \alpha_{\langle i,j \rangle} = 1$. One possibility of computing α_i , $\alpha_{\langle i,j \rangle}$ and α_t is to let

$$\begin{aligned} \alpha_i &= 2g_i x_i / (f_{02}(x) + t), \quad \forall i \\ \alpha_{\langle i,j \rangle} &= \lambda x_i x_j / (f_{02}(x) + t), \quad \forall \langle i,j \rangle \\ \alpha_t &= t / (f_{02}(x) + t) \end{aligned}$$

where x_i, t can take any feasible values which satisfy (7), and we choose the starting value of x_i to be g_i . After combining related terms in $f'_{02}(x)$, the number of variables of the first constraint in (7) will be the size of the variable x plus two, one for s and one for t ; and the SP in (7) can now fit in a standard GGP format

$$\begin{aligned} & \text{minimize} && t \\ & \text{subject to} && s^{-1}f'_{02}(x)^{-1} \geq 1 \\ & && s^{-1}f_{01}(x) \leq 1 \\ & && x_i \leq 3 \\ & && x_i \geq 1. \end{aligned} \quad (9)$$

For a binary image with size $u \times v$, the optimization problem in (9) has an objective function with 1 variable, together with first constraint with $u \times v + 2$ variables, second constraint with $u \times v + 1$ variables and third, fourth constraints both with $u \times v$ variables.

Several software packages are available to solve GPs, including MOSEK [16], TOMLAB [17] and YALMIP [18]. In our implementation, we use the GP software package GPCVX [19] to solve the optimization problem in (9). It should be noted that after a solution is obtained, it is not as straightforward as in [14] to assign binary values to the pixels. Due to the approximation of $f'_{02}(x)$ over $f_{02}(x) + t$, the computation of the optimization problem in (9) will increase or decrease the pixel value of black or white pixels which should be recognized as noise by a bigger amount than those which should not. So we apply a threshold method to assign pixel values: for a pixel whose value is originally 1, if its value increase in the solution is bigger than a certain threshold T , we assume the pixel is surrounded by white pixels, and thus is recognized as white, with a restored pixel value 3. Similarly, for a pixel whose value is originally 3, if its value decrease in the solution is bigger than threshold T , we assume the pixel's neighbors are black pixels, and thus is recognized as black, with a restored pixel value 1. When selecting T , we first apply the SP approach to a small image and then observe the value increase or decrease of those pixels who should be recognized as noise and set this value

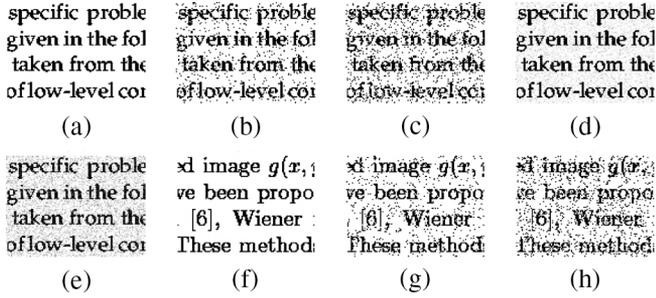


Fig. 1. (a), (f) Two clean images. (b), (c), (g), (h) Degraded images of Fig. 1 (a), (f) by salt-and-pepper noise with noise density of 0.1 and 0.2, respectively. (d), (e) Degraded images of Fig. 1 (a) by Gaussian noise with zero mean value and noise variance of 0.01 and 0.02, respectively.

as T . We also simulate with different values of T s to restore images in our experiments.”

IV. EXPERIMENTAL RESULTS

We apply the image restoration algorithm to noisy binary images, which are degraded by salt-and-pepper noise. Although GPCVX can solve GP reasonably fast, for our case it is still time-consuming to attempt the restoration process of the binary image as a whole. Instead, the degraded binary image is first segmented into small image blocks before the SP is applied to each individual image block. Afterwards, the restored images blocks are regrouped to form the denoised binary image. Image segmentation has also been used to speed up the restoration process in [11], which proves to be very effective. Fig. 1(a) and (f) shows two clean binary images with size of 124×91 , while Fig. 1(b) and (c) and Fig. 1(g) and (h) both show the noisy binary images of Fig. 1(a) and Fig. 1(f) degraded by salt-and-pepper noise with noise density 0.1 and 0.2, respectively. Fig. 1(c) and (d) shows the noisy grayscale images of Fig. 1(a) degraded by Gaussian white noise with zero mean value and variance 0.01 and 0.02, respectively. We set $\lambda = 1$ and $T = 0.4$ in our experiments. Fig. 2(a) and (b) and Fig. 2(c) and (d) both show the restoration results of Fig. 1(b) and Fig. 1(c) with segmented image block size of 10 and 20, respectively. Fig. 2(g) and (h) shows the restoration results of Fig. 1(g) and Fig. 1(h) with segmented image block size of 10, respectively. Fig. 2(d), (e), (i), and (j) shows the restored images of Fig. 1(b), (c), (g), and (h) with image block size of 10, respectively, using the PSD approach in [8], [9]. Fig. 2(k) and (l) shows the restored images of Fig. 1(d) and (e) using the proposed SP approach with image block size 10. We also present in Fig. 2(m) and (n) the restoration results of Fig. 1(b) with image block size 10, setting T as 0.3 and 0.5, respectively. The experiments were conducted on a 3.2-GHz PC with 512-MB RAM, and Table I shows the processing time of the restoration experiments in Fig. 2 in seconds. Table II shows the root mean-square error of the degraded images and the restored images versus the true image, respectively.

In Fig. 1, (b), (c), (g), and (h) are binary, and (d) and (e) are grayscale demonstrating that the SP approach can handle both degraded binary and grayscale images. From Table I and the RMSE data in Table II, the PSD approach is more accurate than the SP approach because there is no such approximation in the PSD approach as converting the SP problem to Reversed GP. The RMSE data in Fig. 1 (a), (m), and (n) shows that T should

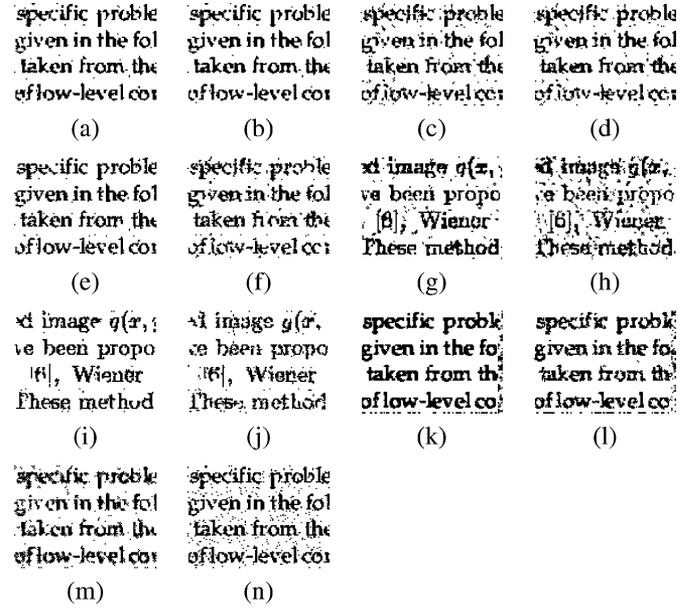


Fig. 2. (a), (b), (c), (d) Restoration results of Fig. 1(b), (c) using SP approach with image block size 10 and 20, respectively. (e), (f), (i), (j) Restoration results of Fig. 1(b), (c), (g), (h) using PSD approach with image block size 10, respectively. (g), (h), (k), (l) Restoration results of Fig. 1(g), (h), (d), (e) using SP approach with image block size 10, respectively. (m), (n) Restoration results of Fig. 1(b) using SP approach setting $T = 0.3$ and $T = 0.5$ with image block size 10, respectively.

TABLE I
CPU TIME (SECONDS)

Fig. 2	(a)	(b)	(c)	(d)	(e)	(f)	(g)
Time	168.6	430.1	149.3	413.5	3483.5	3641.6	173.2
Fig. 2	(h)	(i)	(j)	(k)	(l)	(m)	(n)
Time	177.1	3900.0	3933.1	159.6	167.3	233.4	172.6

TABLE II
ROOT MEAN-SQUARE ERROR VERSUS THE TRUE IMAGE

Fig.	1(b)	2(a)	2(b)	2(c)	1(c)	2(c)
RMSE	3.2521	2.4472	2.3385	1.8534	4.4721	3.3174
Fig.	2(d)	2(f)	1(g)	2(g)	2(i)	1(h)
RMSE	3.1533	2.7101	3.2827	2.7975	1.8609	4.5457
Fig.	2(h)	2(j)	1(d)	2(k)	1(e)	2(l)
RMSE	3.6644	2.5526	8.3488	4.0669	10.0254	3.9273
Fig.	1(m)	2(n)				
RMSE	3.8365	3.0253				

be set as 0.4 with the best restoration results. It is also noted that processing time increases when the image block size is larger, but no obvious restoration improvement is observed. This can be explained by the fact that although with larger image block size the number of image blocks is smaller, the size of the optimization problem corresponding to each image block also increases. The variable number of the first constraint in (7) will be 402 when the image block size is 20 as compared with 102 when the image block size is 10. There is no obvious quality improvement when the image block size increases, because error caused by the boundary effect in (7) is not serious. In (3), $z(x)$ requires the 4-neighborhood of x_i , and the segmentation of image into blocks will cause some neighborhood pixels of x_i s on the block

boundaries to be ignored, which otherwise would be preserved without segmentation. However, only 2 neighborhood pixels of the 4 corners, and 1 neighborhood pixel of the boundary pixels are ignored, and thus the effect caused by ignoring these neighboring pixels is not obvious. It can also be seen that while the RMSE of the restored images is smaller when the PSD approach in [8], [9] is applied, the processing time of restoring a degraded binary image with size around 100×100 using the proposed SP approach is much less, which is over 20 times faster than the PSD approach, when the size of segmented image block is 10. It has also been noticed from [11] that the restoration time grows exponentially with the variable number in the image block, when the PSD approach is applied to restore blurred and noisy images. Although the SP approach proposed in this brief only deals with noisy binary images, it has the potential to restore blurred and noisy images at a much faster speed.

By transforming the SP to a Reversed GP, the optimization problem in (7) fits the standard GP format and therefore can be solved by common GP solvers. On the other hand, it is also where the intrinsic error lies in solving the transformed GP. The geometric inequality used in (8) guarantees that $f'_{02}(x)$ is not a perfect representation of $f_{02}(x) + t$, which is why λ should be carefully selected. Often, when λ is too big, instead of recognizing a pixel neighboring by binary counterparts as noise, the computation of the SP problem will have so small a value of the neighboring pixels that they are recognized as noise incorrectly, and this has to be avoided by selecting an appropriate λ .

The approximation of $f'_{02}(x)$ over $f_{02}(x) + t$ also requires an appropriate selection of an initiation t . It can be seen that a smaller t should enable a more accurate restoration, but too small a t will also render the optimization problem infeasible; in our experiment, we define t to be one twentieth of $f_{02}(x) + t$. From (7), if t is small enough, the approximation of $f'_{02}(x)$ over $f_{02}(x) + t$ will be more accurate when the values of $2g_i x_i, \forall i$ and $\lambda x_i x_j, \forall \langle i, j \rangle$ are drawn together. This could be shown by a simple experiment: for a white binary image of size 5×5 , if we set λ to be 2, the restored pixel values will get closer to u_2 with the decreasing of t so long as the restoration is still feasible; in this case, the approximation of $f'_{02}(x)$ over $f_{02}(x) + t$ is very accurate because $2g_i x_i, \forall i$ and $\lambda x_i x_j, \forall \langle i, j \rangle$ are equal. However, when restoring a noisy binary image, the values of $2g_i x_i, \forall i$ and $\lambda x_i x_j, \forall \langle i, j \rangle$ are far from being equal or close which makes the selection of t and T not robust, and that is why we select t and T empirically. Although we do not come up with a mathematical way of selecting the parameters, extensive experiments show that the current empirically selected parameters λ , t and T work well when restoring degraded binary text images.

V. CONCLUSION

This brief has presented a novel optimization approach to restore degraded binary images by penalized least-squares over binary values. The SP problem is transformed to a Reversed GP to fit into the standard format before it is solved by available

GP solvers. Experiments have demonstrated the superior performance of the approach compared with an existing one using PSD programming, as the former is over 20 times faster and is an accurate method capable of restoring noisy binary images as well as noisy grayscale images. Future investigation in the problem will include robust selection of the parameters such as λ , t and T , and restoration of binary images that are degraded by both noise and blur.

REFERENCES

- [1] J. E. Boyd and J. Meloche, "Binary restoration of thin objects in multi-dimensional imagery," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 20, no. 6, pp. 647–651, Jun. 1998.
- [2] J. Meloche and R. H. Zamar, "Binary image restoration," *Canad. J. Statistics*, vol. 22, pp. 335–355, 1994.
- [3] D. Hitchcock and C. A. Glasbey, "Binary image restoration at subpixel resolution," *Biometrics*, vol. 53, pp. 1010–1053, Sep. 1997.
- [4] M. Neifeld, R. Xuan, and M. Marcellin, "Communication theoretic image restoration for binary-valued imagery," *Appl. Opt. IP*, vol. 39, pp. 269–276, Jan. 2000.
- [5] X. D. Gu, H. M. Wang, and D. H. Yu, "Binary image restoration using pulse-coupled neural network," in *Proc. 8th Int. Conf. Neural Inf. Process.*, Shanghai, China, Nov. 2001, pp. 922–927.
- [6] T. F. Chan, S. Esedoglu, and M. Nikolova, "Finding the global minimum for binary image restoration," in *Proc. IEEE Int. Conf. Image Processing*, Genova, Sep. 2005, vol. 1, pp. 121–124.
- [7] E. Y. Lam, "Blind bi-level image restoration with iterated quadratic programming," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 54, no. 1, pp. 52–56, Jan. 2007.
- [8] J. Keuchel, C. Schellewald, D. Cremers, and C. Schnörr, "Convex relaxations for binary image partitioning and perceptual grouping," *Lecture Notes in Computer Science*, vol. 2191, pp. 353–360, 2001.
- [9] J. Keuchel, C. Schellewald, D. Cremers, and C. Schnörr, "Binary partitioning, perceptual grouping, and restoration with semidefinite programming," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 25, no. 11, pp. 1364–1379, Nov. 2003.
- [10] Y. Shen, E. Y. Lam, and N. Wong, "Robust binary image deconvolution with positive semidefinite programming," in *Recent Advances in Engineering and Computer Science*. Hong Kong: Newswood, 2007, pp. 159–166.
- [11] Y. Shen, E. Y. Lam, and N. Wong, "Binary image restoration by positive semidefinite programming," *Opt. Lett.*, vol. 32, pp. 121–123, Jan. 2007.
- [12] S. Boyd, S. J. Kim, L. Vandenberghe, and A. Hassibi, *A Tutorial on Geometric Programming, Optimization and Engineering*. New York: Springer, 2007 [Online]. Available: <http://www.stanford.edu/~boyd/ggplab/>
- [13] S. Boyd and L. Vandenberghe, *Convex Optimization*. New York: Cambridge University Press, 2004.
- [14] Y. Shen, E. Y. Lam, and N. Wong, "Binary image restoration by signal recovery and synthesis," *Signal Recovery and Synthesis* Jun. 2007, p. SMA3.
- [15] M. Chiang, *Geometric Programming for Communication Systems*. Boston, MA: Now Publishers Inc, 2005.
- [16] "The MOSEK Optimization Tools Version 2.5. User's Manual and Reference" MOSEK ApS, 2002 [Online]. Available: <http://www.mosek.com>
- [17] "User's Guide for TOMLAB/GP," TOMLAB, 2005 [Online]. Available: http://www.tomlab.biz/docs/TOMLAB_GP.pdf
- [18] J. Löfberg, YALMIP. Yet Another LMI Parser. Version 2.4, Linköping Univ., Linköping, Sweden, 2003 [Online]. Available: <http://www.control.ee.ethz.ch/~joloef/yalmip.php>
- [19] gpconv, A MATLAB Solver for Geometric Programs in Convex Form, Stanford Univ., Stanford, CA, 2006 [Online]. Available: <http://www.stanford.edu/~boyd/ggplab/gpconv.pdf>