BFGS method based variable projection approach for image restoration

Qiong-Ying Chen¹ | Yun-Zhi Huang² | Min Gan³,⁴ | C. L. Philip Chen⁵ | Guang-Yong Chen⁶

¹ Fujian Meteorological Information Center, Fuzhou, China
² College of Electrical Engineering and Automation, Hefei University of Technology, Hefei, China
³ College of Computer Science and Technology, Qingdao University, Qingdao, China
⁴ College of Mathematics and Computer Science, Fuzhou University, Fuzhou, China
⁵ College of Computer Science and Engineering, South China University of Technology, Guangzhou, Guangdong, China
⁶ College of Mathematics and Computer Science, Fuzhou University, Fuzhou, China

Abstract

In this paper, a variable projection approach based on the BFGS (Broyden–Fletcher–Goldfarb–Shanno) method for image reconstruction problems is proposed, which is an alternative to the common alternating minimisation scheme. The image restoration is expressed as a nonlinear least-squares problem with reduced parameter space. To improve the efficiency of the algorithm, the BFGS method is proposed to be used to optimise the reduced objective function. The large-scale problem considered in this paper is projected on to a small Krylov subspace using Lanczos bidiagonalisation. The regularisation parameter is selected by a weighted generalised cross validation criterion. Numerical examples demonstrate the efficiency and effectiveness of the proposed algorithm.

1 | INTRODUCTION

1.1 | Background

Image degradation usually occurred in practice due to noise and blurring. Reconstructing original images from degraded images, which is a typical inverse problem, plays an important role in many applications, e.g. medical, biological and astronomical imaging [1–4]. In this paper, we consider recovering a clear image from a blurred and noisy image by a nonlinear least squares framework. We assume that the observed image follows the linear formulation model equation

$$g(a, b) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} k(a - s, b - t) f(s, t) ds dt + \eta(a, b)$$  \hspace{1cm} (1)

where \(g(a, b)\) is the observed, noisy and blurred image, \(a, b\) represent the position of the pixel in space coordinates, \(f(s, t)\) is the original image, \(\eta(a, b)\) is the additive noise, and the kernel function \(k(\cdot, \cdot)\) is the point spread function (PSF).

Since we are interested in digital images, we discretise Equation (1) and obtain the following matrix-vector form

$$y = B(\gamma)x + \eta$$  \hspace{1cm} (2)

where \(y \in \mathbb{R}^n\) is the observed image, \(x \in \mathbb{R}^n\) represents the original image, \(\eta \in \mathbb{R}^n\) is unknown additive noise and \(\gamma \in \mathbb{R}^p\) represents nonlinear parameters. \(B(\gamma) \in \mathbb{R}^{n \times n}\) is a blur operator.

How to recover the original image \(x\) from the degraded image \(y\) is referred to as the image deconvolution. If the blurring operator is known, the process is called non-blind deconvolution; while the blurring operator is unknown, the process is
called blind deconvolution. In this paper, we focus on the blind restoration task.

1.2 Related work

There has been an extensive literature on the image blind restoration. We only mention some of them according to our interests. Sondhi [3] first noticed that image restoration is an ill-posed problem by showing the equivalence of image restoration to the solution of the first kind Fredholm equation. Thus, image restoration is usually performed by minimising a cost function consisting a data fidelity term and certain regularisation terms, which results in the following minimisation problem [6]

\[
\min_{\gamma, x} \Phi(y - B(\gamma)x) + \lambda^2 \Theta(x, \gamma)
\]  

(3)

where \(\Phi(y - B(\gamma)x)\) is the fidelity term, \(\lambda\) is the regularisation parameter and \(\Theta(x, \gamma)\) is the regularisation term, respectively.

In most situations, the fidelity term in (3) is \(\ell_2\)-norm. Designing regularisation term \(\Theta(x, \gamma)\) is an active research field. Besides the commonly used Tikhonov regularisation [7, 8, 10, 11], there are many other regularisation methods, such as, the variational approach [12], total variation [13], and methods based on transform domains [14, 15]. Recently, Cai et al. [6] introduced new sparsity-based regularisation terms on both images and motion-blur kernels; Bharadwaj et al. [16] added two additional constraints (maximally white and maximally front-loaded) to the blind deconvolution framework. The advantage of using the Tikhonov regularisation is that it may lead to very efficient iterative solvers [7].

The optimisation of (3) is nontrivial, because it is a nonconvex problem. The common approach used in the field of image processing is the alternating minimisation (AM) scheme [6, 11, 13, 17], i.e. minimising (3) with respect to \(x\) with \(B\) fixed and minimising (3) with respect to \(B\) with \(x\) fixed. The AM scheme generates a sequence of image and PSF pair. The drawback of this approach is that a good initial guess is required and it may have uncertain convergence properties. If the regulariser in (3) is Tikhonov, an alternative approach to the AM scheme is the variable projection (VP) method [7–9]. For the case of Tikhonov regularisation, the minimisation of (3) is a linear least squares problem if we fix \(\gamma\). Utilising this property, the parameter \(x\) can be eliminated from the optimisation problem, resulting in a reduced objective function that depends only on \(\gamma\). This is the basic idea of the VP method. Apparently, the VP approach takes advantage of the fact that the function is strongly convex in \(x\) when \(\gamma\) is fixed. There are many works demonstrating its efficiency [18–21].

There have been several attempts to handle the image deconvolution problem by deep neural networks. In [23], Xu et al. proposed a deep convolutional neural network to capture the characteristics of degradation. In [33], Jim et al. proposed a deep convolutional network for inverse problems in biomedical imaging. Despite their high quality of deconvolution, those methods generally are time-consuming.

1.3 Motivation and contributions

In this paper, we consider to use a least-squares fit-to-data term with Tikhonov regularisation on \(x\):

\[
\min_{\gamma, x} f(\gamma, x) = \|B(\gamma)x - y\|_2^2 + \lambda^2 \|x\|_2^2
\]

(4)

where \(\| \cdot \|\) denotes vector 2-norm, \(I\) is a regularisation operation and is usually chosen as the identity matrix, and \(\lambda\) is unknown which should be estimated from the given data. For (4), after eliminating \(x\) from the problem, Chung and Nagy [7] proposed a Gauss–Newton method to solve the non-linear least-squares problem. More specifically, rather than solve (4), they instead work with the following reduced objective function

\[
\tilde{f}(\gamma) = \min_x f(\gamma, x)
\]

(5)

The above VP approach is justified by the following theorem, adopted from [22]. Note that the VP method does not alternate between \(\gamma\) and \(x\). Instead, \(x\) is projected out from the problem.

**Theorem 1.** Suppose that \(\mathcal{U} \subset \mathbb{R}^p\) and \(\mathcal{V} \subset \mathbb{R}^q\) are open, and \(f(\gamma, x)\) is twice continuously differentiable on \(\mathcal{U} \times \mathcal{V}\). Suppose that \(\tilde{\gamma} \in \mathcal{U}^*\) and \(\tilde{x} \in \mathcal{V}\) are such that \(\nabla_x f(\tilde{\gamma}, \tilde{x}) = 0\) and \(\nabla^2_x f(\tilde{\gamma}, \tilde{x})\) is positive definite. Then there exist neighborhoods of \(\tilde{\gamma}\) and \(\tilde{x}\) and a twice continuously differentiable function \(\tilde{x}(\gamma)\) is the unique minimiser of \(f(\gamma, \cdot)\) on \(\mathcal{V}\). Then \(\tilde{f}(\gamma)\) is twice continuously differentiable with

\[
\nabla_{\gamma} \tilde{f}(\gamma) = \nabla_{\gamma} f(\gamma, \tilde{x}(\gamma))
\]

(6)

\[
\nabla^2_{\gamma} \tilde{f}(\gamma) = \nabla^2_{\gamma} f(\gamma, \tilde{x}(\gamma)) + \nabla^2_{\gamma, x}(f(\gamma, x))\nabla_x \tilde{x}(\gamma).
\]

(7)

The Gauss–Newton method requires the Jacobian of the reduced function (5). In [7], the authors used the finite-difference approach to approximate the Jacobian numerically. However, the numerical computation of Jacobian is time-consuming.

Since the image blind deconvolution we considered is a large-scale problem and \(x\) is approximated by an iterative method, the analytical expression of the Jacobian is difficult to obtain. On the contrary, the gradient of \(\tilde{f}(\gamma)\) can be easily obtained. From Theorem 1, we can minimise (5) using only the information (6) based on the quasi-Newton method, such as BFGS (Broyden–Fletcher–Goldfarb–Shanno) method.

Thus, in this paper we propose a VP approach based on BFGS method for the image restoration. The BFGS method
has been proven the advantages in solving large-scale problems [24, 25]. Specially, Thompson et al. [26] studied several optimisation methods for blind deconvolution by comparing their speed and accuracy, and the results indicated that the BFGS algorithm has the best performance.

Regularisation parameter selection is another important problem. The generalised cross validation (GCV) is a popular approach to regularisation parameter estimation in blind deconvolution [17, 27]. However, Kim and Gu [28] empirically found that the standard GCV may produce too large or too small regularisation parameters. To overcome the defect, Chung et al. [29] proposed a weighted-GCV (WGCV) method for choosing the regularisation parameter for large scale ill-posed problems. Thus, in this paper we adopt the WGCV method for the regularisation parameter selection.

Specifically, the main contribution of this paper is the introduction of BFGS method to the VP algorithm in the large scale problem which make the algorithm easy to implement in image blind deconvolution. The proposed approach only uses the gradient of the reduced objective function, which makes the algorithm more efficient.

The remainder of this article is organised as follows. The iterative regularisation based on Lanczos bidiagonalisation and WGCV is introduced in Section 2. The proposed BFGS method based VP algorithm is presented in Section 3. Numerical examples are carried out in Section 4. The conclusions are given in Section 5.

2 | ITERATIVE REGULARISATION BASED ON LANCZOS BIDIAGONALISATION AND WGCV

In this paper, we deal with the reduced objective function (5). This is achieved by eliminating the linear parameters $x$ in (4).

For fixed $\lambda$, the problem (5) is convex, and $x$ can be determined by solving the following linear least squares problem

$$
\min_x \| \begin{bmatrix} B & \lambda I \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} - \begin{bmatrix} y \\ 0 \end{bmatrix} \|_2^2,
$$

(8)

where we omit $y$ in $B(y)$ for convenience.

If $\lambda$ is known in (8), the optimal value of $x$ can be expressed as

$$
\hat{x} = B^*_\lambda y
$$

(9)

$$
= (B^T B + \lambda^2 I)^{-1} B^T y
$$

### 2.1 | WGCV

Unfortunately, $\lambda$ is unknown and has to be estimated from the observation data. An efficient way to estimating $\lambda$ is to minimise the GCV function

$$
G_{B,y}(\lambda) = \frac{n \| (I - BB^*_\lambda)y \|_2^2}{(\text{trace}(I - BB^*_\lambda))^2}
$$

(10)

where trace(.) represents sum of diagonal elements.

However, the standard GCV may not perform well for some large-scale problems. Chung et al. [27] showed that the GCV tends to over-estimate the regularisation parameters and thus proposed a WGCV method to overcome the semi-convergence behaviour. The WGCV takes the form

$$
G_{B,y}(\mu, \lambda) = \frac{n \| (I - BB^*_\lambda)y \|_2^2}{(\text{trace}(I - \mu BB^*_\lambda))^2}
$$

(11)

where $\mu$ controls the value of $\lambda$. Note that $\mu=1$ results in the standard GCV (10), and $\mu > 1$ results in smoother solutions. The method of choice of $\mu$ shall be given in Section 2.3.

### 2.2 | Lanczos bidiagonalisation

The minimisation of (11) requires the singular value decomposition (SVD) of the matrix $B$. When $B$ is too large, evaluation of (11) can be computational impractical, because the SVD is not feasible. This difficulty can be relieved by Lanczos bidiagonalisation which projects the large-scale problem on to a small Krylov subspace.

Given the vector $y$ and matrix $B$, the Lanczos bidiagonalisation is described as follows [30]. Let

$$
ev_i m_i = y, \quad w_i n_i = B^T m_i,
$$

(12)

and for $i = 1, 2...$ compute

$$
\begin{cases}
v_{i+1} m_{i+1} = B n_i - w_i m_i, \\
w_{i+1} n_{i+1} = B^T m_{i+1} - v_{i+1} n_i,
\end{cases}
$$

(13)

where $v_i \geq 0, v_i \geq 0$ are chosen so that $\| m_i \|_2 = \| n_i \|_2 = 1$.

Define

$$
A_k = \begin{bmatrix} v_1 & w_1 \\
v_2 & w_2 \\
\vdots & \vdots \\
v_k & w_k \\
v_{k+1} & w_{k+1} \end{bmatrix},
$$

(14)

$$
M_k = (m_1, m_2, ..., m_k), \\
N_k = (n_1, n_2, ..., n_k),
$$

where $k$ is the $k$th iteration of Golub–Kahan bidiagonalisation.

The recurrence relations (13) and (14) can be written as

$$
M_{k+1} (v_i e_i) = y,
$$

(15)
where \( e_{k+1} \) denotes the \((k+1)\) standard unit vector.

At \( k \)th iteration, we get an \( A_k \). Then, an approximate solution can be obtained from the projected least squares problem

\[
\min_{\hat{x}} \frac{1}{2} \| B \ast \hat{x} - y \|^2 = \min_{\hat{x}} \| A_k \hat{x} - y \|_2^2
\]

where \( \varepsilon = \| y \|_2 \), and the approximate solution is \( x_k = N_i \hat{x} \).

### 2.3 Iterative regularisation

Since the original problem is ill posed, \( A_k \) may become ill-conditioned as the iteration proceeds. Thus, we compute a regularised projected least squares problem

\[
\min_{\hat{x}} \frac{1}{2} \| A_k \hat{x} - y \|_2^2 + \lambda^2 \| \hat{x} \|_2^2
\]

Note that the dimension of \( A_k \) is very small compared to \( B \), therefore the SVD can be applied.

In this paper, as described in Section 2.1, we use the WGCV (11) to find an approximate \( \lambda \). The WGCV function for problem (18) is given as

\[
G_{A_k,s}(\lambda, \mu) = \frac{k \| (I - A_k A_k^+ s) \varepsilon \|_2^2}{\text{trace}(I - \mu A_k A_k^+(I + \mu A_k A_k^+))}
\]

At early iterations, the \( \mu \) is obtained by solving

\[
\frac{\partial}{\partial \lambda} (G_{A_k,s}(\lambda, \mu))|_{\lambda=\lambda_{k,\text{opt}}} = 0,
\]

where \( \lambda_{k,\text{opt}} \) is the smallest singular value of \( A_k \). In later iterations, \( \mu \) is taken as the average of the previous values.

According to Equation (19), we choose \( \lambda \) based on \( A_k \). \( A_k \) is obtained by Lanczos bidiagonalisation of \( B \) (see Section 2.2). That is, the selection of \( \lambda \) depends on \( B(\gamma) \) in (4).

### 3 BFGS METHOD BASED VP APPROACH FOR LARGE-SCALE IMAGE RESTORATION

In this section, we return to the nonlinear least-squares problem (4). We can treat the problem (4) as a general nonlinear least-squares problem, and solve it using some standard optimisation algorithms regardless its special structure. The disadvantage of this approach is that the algorithm generally converges very slowly. Or, we can use the block coordinate descent approach, i.e. the AM scheme mentioned in Section 1, which alternates between fixing one set of the variables. The problem with this approach is that it may have uncertain convergence properties and good initial guesses are required.

It will be very beneficial to utilise the fact that the problem is convex in \( x \). Thus we consider a reduced parameter space method. Denote the reduced cost function as

\[
\tilde{f}(\gamma) = f(\gamma, x(\gamma))
\]

where \( x(\gamma) \) is the solution of

\[
\min_{x} \left[ \frac{B(\gamma)}{\lambda} x - y \right]_2^2
\]

We have described in detail how to obtain \( x(\gamma) \) in Section 2.

According to (6),

\[
\nabla_{\gamma} \tilde{f}(\gamma) = \nabla_{\gamma} f(\gamma, x(\gamma)) = \frac{\partial}{\partial \gamma} f(\gamma, x(\gamma)) x(\gamma) = \frac{\partial}{\partial \gamma} f(\gamma, x(\gamma)) + \frac{\partial}{\partial x} f(\gamma, x(\gamma)) \frac{\partial x}{\partial \gamma}
\]

where \( D \) denotes the Fréchet derivative of the matrix.

Then, the update of \( \gamma \) at \((k+1)\)th iteration based on BFGS method is given by

\[
\gamma_{k+1} = \gamma_k - \beta_k (H_k)^{-1} \nabla_{\gamma} \tilde{f}(\gamma_k)
\]

where \( \beta_k \) is the step length,

\[
H_{k+1} = H_k + \frac{\gamma_k P_k^T}{\gamma_k P_k} - \frac{H_k s_k s_k^T (H_k)^T}{s_k^T H_k s_k}
\]

\[
s_k = \gamma_{k+1} - \gamma_k,
\]

\[
z_k = \nabla_{\gamma} \tilde{f}(\gamma_{k+1}) - \nabla_{\gamma} \tilde{f}(\gamma_k).
\]

For clarity, we give the pseudocode of BFGS method based VP approach for large-scale image restoration as follows.

### 4 NUMERICAL EXAMPLES

In order to verify the efficiency and effectiveness of proposed BFGS method based VP approach (denoted as BFGS-VP), we give examples from image deblurring, also called blind deconvolution in the field of image processing.

In the experiments, we use the blurred signal-to-noise ratio (BSNR) to measure the noise contained in the observed image, and the improved signal-to-noise ratio (ISNR) to measure the quality of the recovered image. They are defined as follows:
The image formation process is modelled as (2). The matrix

\[ B(y) = B(P(y)) \]  

(28)

where \( P(y) \) is the PSF. PSF in this article mainly considers the spatial invariance and parametric type of blurring function. Two common types of blur are used in this article: Gaussian blur (PSF form see Equation (29)) and motion blur (PSF form see Equation (33)).

In any blind deconvolution problem, some assumptions about the blur are needed. Here we assume a normal Gaussian blur, where the PSF, centered at \((s_1, s_2)\), has the form

\[ p_{ij} = \exp \left( \frac{-1}{2} \begin{bmatrix} i - s_1 \\ j - s_2 \end{bmatrix}^T \begin{bmatrix} \sigma_1^2 & \rho \\ \rho & \sigma_2^2 \end{bmatrix}^{-1} \begin{bmatrix} i - s_1 \\ j - s_2 \end{bmatrix} \right) \]

\[ = \exp \left( \frac{-(i - s_1)^2 \sigma_1^2}{2\sigma_1^2} - \frac{(j - s_2)^2 \sigma_2^2}{2\sigma_2^2} + 2(i - s_1)(j - s_2)\rho \right). \]

(29)

Let

\[ y = \begin{bmatrix} c_1 \\ c_2 \\ l \end{bmatrix} \]

(30)

and let \( B(y) \) be the matrix defined by the Gaussian PSF with parameters \( y \).

**Case 1.** To make a fair comparison with the results of [7], we use a 512 \( \times \) 512 “Grain” image convolved with a Gaussian PSF defined by the parameter \( y_{true} = [3, 4, 0.5]^T \) to create a blurred image, and then crop the center of the blurred image to create a size of 256 \( \times \) 256 blurred image. We also added 1% Gaussian white noise to the image, and define the relative error of the blur parameter as

\[ \Delta y = \frac{\| y_{true} - y \|_2}{\| y_{true} \|_2} \]

(31)

and define the relative error of the image as

\[ \Delta x = \frac{\| x_{true} - x \|_2}{\| x_{true} \|_2}. \]

(32)

Using the same initial guess \( y_0 = [5, 6, 1]^T \) as in [7], the relative objective function value (rel obj), the relative gradient value (rel grad), the relative error of the blur parameter (\( \Delta y \)), the relative error of the image (image error), the calculated regularisation parameters (\( \lambda \)) and the running time are shown in Table 1. Table 1 also lists the results in [7] (denoted as GN-VP) for comparison. From Table 1, it can be seen that proposed method achieves slightly smaller \( \Delta y \) and image error than those in [7]. However, the running time of the proposed method is much shorter than that of GN-VP, which indicates the proposed method is more efficient. To verify the effectiveness of the WGCV, we list the results of both methods using GCV for comparison in Table 1. It can be observed that the methods using the WGCV obtain better results.

Figure 1 shows the original and blurred images, and the reconstructed images obtained by GN-VP and BFGS-VP methods. Figure 2 shows the relative errors of the blur parameters \( \Delta y \) and the errors of the image \( \Delta x \) during the iteration using the GN-VP and BFGS-VP. We note that \( \Delta y \) and \( \Delta x \) begin increasing after eight iterations for the GN-VP method. This is because there is no good stopping criterion. However, this phenomenon does not occur in the BFGS-VP method.

**Case 2.** In this case, we use another image (cameraman) to test the efficiency and effectiveness of proposed BFGS-VP. Similar to case 1, the 512 \( \times \) 512 cameraman image is convolved with a Gaussian PSF using the parameter \( y_{true} = [3, 4, 0.5]^T \), and then a size of 256 \( \times \) 256 blurred image is obtained by cropping the 512 \( \times \) 512 image. 1% Gaussian white noise is added to the image. Table 2 lists the rel obj, rel grad, \( \lambda \), \( \Delta y \), image error and running time of the GN-VP method and BFGS-VP method. The initial value is \( y_0 = [5, 6, 1]^T \). Again, from Table 2, we can see that the BFGS-VP obtains similar image error and smaller \( \Delta y \) compared to the results of GN-VP. Moreover, the BFGS-VP spends much less time than the GN-VP. Figure 3 shows the original and blurred images, and the reconstructed images obtained by the GN-VP and BFGS-VP methods. Figure 4 shows the relative errors of the blur parameters \( \Delta y \) and the errors of the image \( \Delta x \) during the iteration using the GN-VP and BFGS-VP. In this case, the \( \Delta y \) obtained by the GN-VP
| Method                  | Iter | Rel obj | Rel grad | λ      | Δy     | Image Error | Elapsed time |
|------------------------|------|---------|----------|--------|--------|-------------|--------------|
| GN-VP + WGCV           | 1    | 1.0000  | 1.0000   | 0.1684 | 0.5716 | 0.4082      | 95.6878 s    |
|                        | 2    | 0.3309  | 0.6411   | 0.1224 | 0.3347 | 0.3768      |              |
|                        | 3    | 0.1493  | 0.4342   | 0.0988 | 0.2192 | 0.3542      |              |
|                        | 4    | 0.0762  | 0.2953   | 0.0807 | 0.1469 | 0.3395      |              |
|                        | 5    | 0.0479  | 0.2262   | 0.0718 | 0.0998 | 0.3281      |              |
|                        | 6    | 0.0355  | 0.1862   | 0.0678 | 0.0639 | 0.3220      |              |
|                        | 7    | 0.0288  | 0.1615   | 0.0661 | 0.0345 | 0.3184      |              |
|                        | 8    | 0.0242  | 0.1419   | 0.0651 | 0.0139 | 0.3169      |              |
|                        | 9    | 0.0213  | 0.1265   | 0.0645 | 0.0240 | 0.3164      |              |
|                        | 10   | 0.0190  | 0.1151   | 0.0645 | 0.0449 | 0.3166      |              |
|                        | 11   | 0.0171  | 0.1042   | 0.0641 | 0.0657 | 0.3176      |              |
|                        | 12   | 0.0159  | 0.0947   | 0.0638 | 0.0853 | 0.3190      |              |
|                        | 13   | 0.0146  | 0.0863   | 0.0634 | 0.1038 | 0.3207      |              |
|                        | 14   | 0.0137  | 0.0800   | 0.0635 | 0.1210 | 0.3224      |              |
|                        | 15   | 0.0131  | 0.0731   | 0.0629 | 0.1375 | 0.3244      |              |
|                        | 16   | 0.0123  | 0.0668   | 0.0622 | 0.1530 | 0.3265      |              |
| BFGS-VP + WGCV         | 1    | 1.0000  | 1.0000   | 0.1684 | 0.5716 | 0.4082      | 52.7791 s    |
|                        | 2    | 0.2591  | 0.5727   | 0.1148 | 0.3086 | 0.3699      |              |
|                        | 3    | 0.2289  | 0.5383   | 0.1111 | 0.3119 | 0.3680      |              |
|                        | 4    | 0.0721  | 0.2835   | 0.0810 | 0.2133 | 0.3372      |              |
|                        | 5    | 0.0471  | 0.2205   | 0.0715 | 0.1115 | 0.3282      |              |
|                        | 6    | 0.0386  | 0.1939   | 0.0692 | 0.1065 | 0.3235      |              |
|                        | 7    | 0.0294  | 0.1616   | 0.0664 | 0.0769 | 0.3187      |              |
|                        | 8    | 0.0236  | 0.1373   | 0.0650 | 0.0516 | 0.3165      |              |
|                        | 9    | 0.0196  | 0.1168   | 0.0644 | 0.0492 | 0.3163      |              |
| GN-VP + GCV            | 1    | 1.0000  | 1.0000   | 0.2036 | 0.5716 | 0.4091      | 91.1822 s    |
|                        | 2    | 0.1854  | 0.5076   | 0.1185 | 0.2645 | 0.3673      |              |
|                        | 3    | 0.0701  | 0.3055   | 0.0861 | 0.1650 | 0.3436      |              |
|                        | 4    | 0.0408  | 0.2246   | 0.0743 | 0.1116 | 0.3306      |              |
|                        | 5    | 0.0287  | 0.1796   | 0.0686 | 0.0730 | 0.3235      |              |
|                        | 6    | 0.0226  | 0.1523   | 0.0622 | 0.0429 | 0.3197      |              |
|                        | 7    | 0.0190  | 0.1347   | 0.0653 | 0.0214 | 0.3174      |              |
|                        | 8    | 0.0165  | 0.1209   | 0.0649 | 0.0234 | 0.3163      |              |
|                        | 9    | 0.0147  | 0.1084   | 0.0645 | 0.0422 | 0.3165      |              |
|                        | 10   | 0.0131  | 0.0978   | 0.0641 | 0.0625 | 0.3174      |              |
|                        | 11   | 0.0122  | 0.0888   | 0.0638 | 0.0821 | 0.3187      |              |
|                        | 12   | 0.0112  | 0.0808   | 0.0634 | 0.1006 | 0.3204      |              |
|                        | 13   | 0.0105  | 0.0749   | 0.0635 | 0.1178 | 0.3219      |              |
|                        | 14   | 0.0100  | 0.0684   | 0.0629 | 0.1344 | 0.3239      |              |
|                        | 15   | 0.0093  | 0.0624   | 0.0622 | 0.1499 | 0.3260      |              |
|                        | 16   | 0.0090  | 0.0568   | 0.0613 | 0.1645 | 0.3281      |              |
| BFGS-VP + GCV          | 1    | 1.0000  | 1.0000   | 0.2036 | 0.5716 | 0.4091      | 46.3715 s    |
|                        | 2    | 0.2269  | 0.5621   | 0.1261 | 0.3067 | 0.3722      |              |
|                        | 3    | 0.1679  | 0.4867   | 0.1130 | 0.3025 | 0.3666      |              |
|                        | 4    | 0.1183  | 0.4041   | 0.1014 | 0.2573 | 0.3573      |              |

(Continues)
TABLE 1 (Continued)

| Method | Iter | Rel obj | Rel grad | λ   | Δy   | Image Error | Elapsed time |
|--------|------|---------|----------|-----|------|-------------|--------------|
| 5      | 0.0923 | 0.3510 | 0.0935 | 0.2299 | 0.3518 |
| 6      | 0.0478 | 0.2408 | 0.0770 | 0.1537 | 0.3354 |
| 7      | 0.0290 | 0.1764 | 0.0687 | 0.0894 | 0.3244 |
| 8      | 0.0196 | 0.1351 | 0.0655 | 0.0670 | 0.3179 |
| 9      | 0.0136 | 0.0994 | 0.0644 | 0.1354 | 0.3170 |
| 10     | 0.0101 | 0.0685 | 0.0628 | 0.1446 | 0.3234 |

FIGURE 1 (a) The original grain image, (b) blurred image, (c) reconstructed image by GN-VP and (d) reconstructed image by BFGS-VP

has a jump at the 14th iteration and the Δy obtained by the BFGS-VP increases a little at the last iteration. However, this does not severely affect the reconstructed image.

Case 3. In this case, we compare the proposed method with the method in [17] (denoted as GCV-BD) and [32] (denoted as VB-TV1 and VB-TV2) using the cameraman image and Lena image of size 256×256. The blurred, noisy images are obtained by blurring the original images with a Gaussian PSF with variance five and adding additive Gaussian noise (BSNR = 20 db and BSNR = 40 db). Table 3 lists the comparison results of these methods. It can be observed that the proposed BFGS-VP method achieves very competitive results, which demonstrates its effectiveness. Figure 5 shows the original and blurred image, and the reconstructed image of cameraman obtained by the BFGS-VP method. Figure 6 shows the original and blurred image, and the reconstructed image of Lena obtained by the BFGS-VP method.

Case 4. As our method can naturally be extended to deal with non-uniform blur, we report results on an image (shown in Figure 7) degraded by spatially-variant motion blur. This case restores the image degradation caused by uniform linear motion. The specific form of PSF that causes image degradation by uniform linear motion is:

\[
p_{ij} = \begin{cases} 
\frac{1}{\sigma}, & j = \tan \theta, \ 0 \leq j \leq \sigma \cos \theta \\
0, & j \neq \tan \theta, \ -\infty \leq i \leq +\infty, 
\end{cases}
\]  

(33)
FIGURE 2  The relative errors of blur parameters and relative image errors during the iteration for grain image

TABLE 2  Comparison results of cameraman image restoration ($\gamma_0 = [5, 6, 1]$)

| Method       | Iter | Rel obj | Rel grad | HYBR  | $\Delta y$ | Image error | Elapsed time |
|--------------|------|---------|----------|-------|------------|-------------|--------------|
| GN-VP + WGCV | 1    | 1.0000  | 1.0000   | 0.1264| 0.5716     | 0.2278      | 276.9038 s   |
|              | 2    | 0.3964  | 0.6224   | 0.0894| 0.4193     | 0.2191      |              |
|              | 3    | 0.2028  | 0.3857   | 0.0645| 0.3684     | 0.2186      |              |
|              | 4    | 0.1500  | 0.2870   | 0.0541| 0.3322     | 0.2156      |              |
|              | 5    | 0.1234  | 0.2311   | 0.0489| 0.2920     | 0.2078      |              |
|              | 6    | 0.1103  | 0.2004   | 0.0468| 0.2601     | 0.1982      |              |
|              | 7    | 0.1019  | 0.1788   | 0.0494| 0.2327     | 0.1888      |              |
|              | 8    | 0.1002  | 0.1744   | 0.0455| 0.2266     | 0.1866      |              |
|              | 9    | 0.0998  | 0.1733   | 0.0454| 0.2251     | 0.1861      |              |
|              | 10   | 0.0996  | 0.1728   | 0.0454| 0.2244     | 0.1858      |              |
|              | 11   | 0.2181  | 0.4101   | 0.0941| 0.2236     | 0.1659      |              |
|              | 12   | 0.1727  | 0.3330   | 0.0906| 0.1631     | 0.1560      |              |
|              | 13   | 0.1451  | 0.2718   | 0.0875| 0.1348     | 0.1502      |              |
|              | 14   | 0.1306  | 0.2291   | 0.0883| 0.2131     | 0.1464      |              |
|              | 15   | 0.1156  | 0.1928   | 0.0830| 0.1711     | 0.1443      |              |
|              | 16   | 0.1093  | 0.1624   | 0.0803| 0.1307     | 0.1437      |              |
| BFGS-VP + WGCV| 1    | 1.0000  | 1.0000   | 0.1261| 0.5716     | 0.2279      | 51.9953 s    |
|              | 2    | 0.3168  | 0.5226   | 0.0930| 0.3932     | 0.1931      |              |
|              | 3    | 0.2290  | 0.5407   | 0.0922| 0.2984     | 0.1888      |              |
|              | 4    | 0.2889  | 0.5303   | 0.0914| 0.2932     | 0.1878      |              |
|              | 5    | 0.1862  | 0.3871   | 0.0850| 0.2010     | 0.1649      |              |
|              | 6    | 0.1695  | 0.3411   | 0.0906| 0.1363     | 0.1547      |              |
|              | 7    | 0.1132  | 0.1990   | 0.0828| 0.0701     | 0.1438      |              |
|              | 8    | 0.1043  | 0.1605   | 0.0797| 0.0865     | 0.1430      |              |
FIGURE 3  (a) The original cameraman image, (b) blurred image, (c) reconstructed image by GN-VP and (d) reconstructed image by BFGS-VP

FIGURE 4  The relative errors of blur parameters and relative image errors during the iteration for the cameraman image
where \( \sigma \) is the blur length, \( \theta \) is the direction of motion. Let

\[
\gamma = \begin{bmatrix} \sigma \\ \theta \end{bmatrix}
\]  

(34)

and let \( B(\gamma) \) be the matrix defined by the uniform linear motion PSF with parameters \( \gamma \). We use a 512 \( \times \) 512 “license plate” image convolved with a Gaussian PSF defined by the parameter \( \gamma_{\text{true}} = [12, 7]^T \) to create a blurred image. We also added 1\% Gaussian white noise to the image. Using the initial guess \( \gamma_0 = [11, 6]^T \). Figure 7. shows the original image, blurred image and the restored image by the proposed method. We can see that our method generated image with clear textures.

5 | CONCLUSION

We have proposed a BFGS-VP algorithm for the large-scale ill-posed inverse problems with applications to image deblurring. The VP approach, which projects out the \( x \) from the original objective function, is different from the commonly used AM scheme. The WGCV, which is an improved version of GCV, is adopted to select the Tikhonov regularisation parameter. The Lanczos bidiagonalisation method is used to project the large-scale problem on to a small Krylov subspace. The main selling point of the proposed algorithm is that it avoids computing the complicated “inexact” Jacobian matrix when using the Gauss–Newton algorithm, thus improves the computational efficiency of the algorithm. Numerical experiments verify the efficiency.
and effectiveness of the proposed algorithm. Our future work will be combining some deep neural networks [33] to deal with the ill-posed inverse problems in imaging.

ACKNOWLEDGEMENT
This work was supported in part by the National Nature Science Foundation of China under Grant 62073082, the science and technology major special project of Anhui Province 18030901040, and the Taishan Scholar Program of Shandong Province.

REFERENCES
1. Hansen, P.C., Nagy, J.G., O’Leary, D.P.: Deblurring Images: Matrices, Spectra and Filtering. SIAM, Philadelphia, PA (2006)
2. Campisi, P., Egiazarian, K.: Blind Image Deconvolution: Theory and Applications. CRC Press, Boca Raton, FL (2007)
3. Chaudhuri, S., Velmurugan, R., Rameshan, R.: Blind Image Deconvolution. Springer, Berlin, Heidelberg (2014)
4. Chung, J., Roththob, L.: Computational methods for image reconstruction. NMR Biomed. 30(4), e3545 (2017)
5. Sondhi, M.M.: Image restoration: The removal of spatially invariant degradations. Proc. IEEE 60, 842–857 (1972)
6. Cai, J.-F., et al.: Framelet-based blind motion deblurring from a single image. IEEE Trans. Image Process. 21(2), 562–572 (2011)
7. Chung, J., Nagy, J.G.: An efficient iterative approach for large-scale separable nonlinear inverse problems. SIAM J. Sci. Comput. 31(6), 4654–4674 (2009)
8. Cornelio, A., Piccolomini, E.L., Nagy, J.G.: Constrained numerical optimization methods for blind deconvolution. Numer. Algorithms 65(1), 23–42 (2014)
9. Gan, M., et al.: Recursive variable projection algorithm for a class of separable nonlinear models. IEEE Trans. Neural Networks Learn. Syst. (2020). https://doi.org/10.1109/TNNLS.2020.3026482
10. Chung, J., Palmer, K.: A hybrid LSRM algorithm for large-scale Tikhonov regularization. SIAM J. Sci. Comput. 37, S652–S580 (2015)
11. You, Y., Kaveh, M.: A regularization approach to joint blur identification and image restoration. IEEE Trans. Image Process 5(3), 416–428 (1996)
12. Molina, R., Mateos, J., Katsaggelos, A.K.: Blind deconvolution using a variational approach to parameter, image, and blur estimation. IEEE Trans. Image Process. 15(12), 3715–3727 (2006)
13. Chan, T.F., Wong, C.K.: Total variation blind deconvolution. IEEE Trans. Image Process. 7(3), 370–379 (1998)
14. Easley, G., Healy, D., Berenstein, C.: Image deconvolution using a general ridgelet and curvelet domain. SIAM J. Imaging Sci. 2(1), 253–283 (2009)
15. Easley, G., Labate, D., Lim, W.: Sparse directional image representations using the discrete shearlet transform. Appl. Comput. Harmonic Anal. 25(1), 25–46 (2008)
16. Bharadwaj, P., Demanet, L., Fournier, A.: Focused blind deconvolution. IEEE Trans. Image Process. 67(12), 3168–3180 (2019)
17. Liao, H., Ng, M.K.: Blind deconvolution using generalized cross-validation approach to regularization parameter estimation. IEEE Trans. Image Process. 20(3), 670–680 (2011)
18. Gan, M., Li, H.-X., Peng, H.: A variable projection approach for efficient Estimation of RBF-ARX model. IEEE Trans. Cybern. 45(3), 476–485 (2015)
19. Gan, M., et al.: On some separated algorithms for separable nonlinear least squares problems. IEEE Trans. Cybern. 48(10), 2866–2874 (2018)
20. Chen, G.-Y., et al.: A regularized variable projection algorithm for separable nonlinear least squares problems. IEEE Trans. Autom. Control 64(2), 526–537 (2019)
21. Chen, G.-Y., et al.: Modified Gram-Schmidt method based variable projection algorithm for separable nonlinear models. IEEE Trans. Neural Netw. Learn. Syst. 30(8), 1–9 (2019)
22. Aravkin, A.Y., van Leeuwen, T.: Estimating nuisance parameters in inverse problems: Inverse Problems. 28, 115016 (2012)
23. Xu, L., et al.: Deep convolutional neural network for image deconvolution. In: Proc. of the Advanced Neural Information Processing Systems, pp. 1793–1798. IEEE, Piscataway, NJ (2014)
24. Aravkin, A.Y., Drusvyatskiy, D., Leeuwen, T-V.: Efficient quadratic penalization through the partial minimization technique. arXiv:1606.02395, (2017)
25. Fang, H., et al.: Iteratively reweighted blind deconvolution with adaptive regularization parameter estimation. IEEE Access 5, 11959–11973 (2017)
26. Thompson, D., Calef, B., Werth, M.: Performance comparison of optimization methods for blind deconvolution. Proc. Adv. Maui Opt. Space Survell. Technol. Conf. (AMOS), pp. 1–11. Springer, Berlin, Heidelberg (2016)
27. Ramani, S., et al.: Regularization parameter selection for nonlinear iterative image restoration and MRI reconstruction using GCV and SURE-based methods. IEEE Trans. Image Process. 21(8), 3659–3672 (2012)
28. Kim, Y., Gu, C.: Smoothing spline Gaussian regression: More scalable computation via efficient approximation. R. Stat. Soc. B 66(2), 337–356 (2004)
29. Chung, J., Nagy, J.G., O’Leary, D.P.: A weighted GCV method for Lanczos hybrid regularization. Electron. Trans. Numer. Anal. 149–167 (2008)
30. Paige, C.C., Saunders, M.A.: LSQR: an algorithm for sparse linear equations and sparse least squares problems. ACM Trans. Math. Soft. 8, 43–71 (1982)

FIGURE 7 (a) The original License plate image, (b) the blurred image, (c) restored image by the proposed method.
31. Schulz, T.: Multiframe blind deconvolution of astronomical images. J. Opt. Soc. Am. A 10, 1064–1073 (1993)
32. Babacan, S.D., Molina, R., Katsaggelos, A.K.: Variational Bayesian blind deconvolution using a total variation prior. IEEE Trans. Image Process. 18(1), 12–26 (2009)
33. Jin, K.H., et al.: Deep convolutional neural network for inverse problems in imaging. IEEE Trans. Image Process. 26, 4509–4522 (2017)

How to cite this article: Chen Q-Y, Huang Y-Z, Gan M, Chen CLP, Chen G-Y. BFGS method based variable projection approach for image restoration. IET Image Process. 2021−12. https://doi.org/10.1049/ipr2.12270