This is a repository copy of *The Sylvester and Bézout resultant matrices for blind image deconvolution*.

White Rose Research Online URL for this paper:
http://eprints.whiterose.ac.uk/129418/

Version: Published Version

eprints@whiterose.ac.uk
https://eprints.whiterose.ac.uk/

CrossMark

The Sylvester and Bézout Resultant Matrices for Blind Image Deconvolution

Joab R. Winkler[1] · Hanan Halawani[1]

Abstract

Blind image deconvolution (BID) is one of the most important problems in image processing, and it requires the determination of an exact image $\mathcal{F}$ from a degraded form of it $\mathcal{G}$ when little or no information about $\mathcal{F}$ and the point spread function (PSF) $\mathcal{H}$ is known. Several methods have been developed for the solution of this problem, and one class of methods considers $\mathcal{F}, \mathcal{G}$ and $\mathcal{H}$ to be bivariate polynomials in which the polynomial computations are implemented by the Sylvester or Bézout resultant matrices. This paper compares these matrices for the solution of the problem of BID, and it is shown that it reduces to a comparison of their effectiveness for greatest common divisor (GCD) computations. This is a difficult problem because the determination of the degree of the GCD of two polynomials requires the calculation of the rank of a matrix, and this rank determines the size of the PSF. It is shown that although the Bézout matrix is symmetric (unlike the Sylvester matrix) and it is smaller than the Sylvester matrix, which has computational advantages, it yields consistently worse results than the Sylvester matrix for the size and coefficients of the PSF. Computational examples of blurred and deblurred images obtained with the Sylvester and Bézout matrices are shown, and the superior results obtained with the Sylvester matrix are evident.

Keywords Blind image deconvolution · Sylvester matrix · Bézout matrix

Mathematics Subject Classification 65F99 · 12Y05

1 Introduction

The removal of blur and other degradations from an image is important because it makes its subsequent processing for, for example, feature detection and feature extraction, significantly easier and more reliable. Images that arise in practical applications may suffer from degradations, which must be removed before the images are interrogated. This practical necessity has been a major motivation for the continued research into methods for the improvement of degraded images.

✉ Joab R. Winkler
  j.r.winkler@sheffield.ac.uk

  Hanan Halawani
  hhalawani2@sheffield.ac.uk

[1] Department of Computer Science, The University of Sheffield, Regent Court, Sheffield S1 4DP, UK

If the point spread function (PSF) is spatially invariant, then the blurred image $\mathcal{G}$ is formed by the convolution of the PSF $\mathcal{H}$ and the exact image $\mathcal{F}$. If, in addition, the PSF is known, then the determination of a deblurred form of a blurred image reduces to linear deconvolution, which can be solved by, for example, methods of computational linear algebra [16]. The more difficult problem arises when there is no information, or only partial information, about the PSF, in which case the problem reduces to blind image deconvolution (BID). In this circumstance, additional information about the exact image and/or the PSF must be provided in order to obtain a deblurred image. An example of this prior information is the knowledge that the PSF is separable, that is, if $h(x, y)$ is the bivariate polynomial whose coefficients are the pixel values of the PSF, then it can be written as the product of its column component $h_c(x)$ and its row component $h_r(y)$, $h(x, y) = h_c(x)h_r(y)$.

The first set of methods that were used for the solution of the problem of BID are reviewed in [17], but more recent methods have used Bayes' theorem, for which priors are placed on the PSF and the exact image [3,23,26,37]. These

methods require that the maximum a posteriori (MAP) estimates $\hat{\mathcal{F}}$ and $\hat{\mathcal{H}}$ of, respectively, $\mathcal{F}$ and $\mathcal{H}$,

$$
\begin{aligned}
\left(\hat{\mathcal{F}}, \hat{\mathcal{H}}\right) &= \arg \max_{\mathcal{F}, \mathcal{H}} \ p\left(\mathcal{F}, \mathcal{H} | \mathcal{G}\right) \\
&= \arg \max_{\mathcal{F}, \mathcal{H}} \ p\left(\mathcal{G} | \mathcal{F}, \mathcal{H}\right) p\left(\mathcal{F}\right) p\left(\mathcal{H}\right),
\end{aligned} \tag{1}
$$

minimise the error associated with the convolution operation and have sparse derivatives, which arises because many natural images are composed of regions in which the gradients are small, with relatively few regions of high gradients. The distribution of gradients x in $\mathcal{F}$ is often modelled as a generalised Laplacian,

$$
p(x) \propto \exp \left(-\frac{1}{2}\left(\frac{|x|}{\sigma}\right)^{\alpha}\right), \qquad 0 < \alpha < 1,
$$

where σ is the standard deviation of the distribution. This distribution is frequently called a natural image prior because of its satisfaction by many natural images, and its imposition as a prior concentrates gradients at a small number of pixels. The condition $\alpha = 2$ yields the Gaussian distribution, but this prior does not yield a good deblurred image.

The simultaneous MAP estimate of $\mathcal{F}$ and $\mathcal{H}$ is ill-posed, and it is better to calculate the MAP estimate of $\mathcal{H}$ by marginalisation over all possible images $\mathcal{F}$ because this yields a better conditioned problem, and then calculate the MAP estimate of $\mathcal{F}$ [18,19]. The calculation of the MAP estimate of $\mathcal{H}$ leads, however, to an intractable integral, but it is shown that an approximation can be made. This approximation yields a solution of (1) that uses the expectation–maximisation (EM) algorithm, which consists of two steps, the E-step and the M-step, that alternate until convergence is achieved. The E-step requires that a prior be placed on $\mathcal{H}$ and a non-blind deconvolution problem be solved, which allows an estimate $\tilde{\mathcal{F}}$ of $\mathcal{F}$ to be calculated. The M-step involves the calculation of an improved estimate of $\mathcal{H}$, using the estimate $\tilde{\mathcal{F}}$ of the exact image. The E- and M-steps are then invoked again, using the most recent estimates of $\mathcal{H}$ and $\mathcal{F}$, and the process is repeated until convergence occurs. The first step in the EM algorithm requires that a prior be placed on $\mathcal{H}$, and a uniform distribution is usually assumed [18,26], but this is not realistic [19].

The image processing toolbox in MATLAB has four functions that deblur an image:

1. `deconvreg.m`: Regularisation of a linear algebraic equation
2. `deconvwnr.m`: The Wiener filter
3. `deconvlucy.m`: The Lucy–Richardson algorithm
4. `deconvblind.m`: Maximum likelihood estimation

The functions `deconvreg.m`, `deconvwnr.m` and `deconvlucy.m` solve the problem of semi-blind image deconvolution because the PSF is specified as an input argument to these functions. The edges of the deblurred image obtained with these three functions may show ringing, which can be reduced by calling the function `edgetaper.m` before the deblurring function is called. The function `edgetaper.m` requires the PSF as one of its arguments, and it can therefore only be used for solving the problem of semi-blind image deconvolution.

The function `deconvblind.m` is different because only an estimate of the PSF, rather than the exact PSF, need be specified as an input argument, but the computed PSF is very dependent on the estimate of its size, and less by the entries of the matrix that defines it. A comparison of these four functions and the Sylvester resultant matrix for the computation of a deblurred image is in [30,31], and it is noted in [15] that the best deblurred image obtained from the four functions requires visual inspection of several deblurred images that differ in the values of the input arguments, for example, the number of iterations and the noise power.

A different problem is solved in this paper because prior information on the PSF is not assumed (apart from the property of spatial invariance), and thus both the size and coefficients of the PSF are calculated. The method of image deblurring described in this paper considers $\mathcal{F}$, $\mathcal{G}$ and $\mathcal{H}$ to be bivariate polynomials, and deblurring is achieved by performing polynomial computations on $\mathcal{G}$ and $\mathcal{H}$. This approach is justified by the convolution operation, which defines the multiplication of two polynomials and the formation of a blurred image by a spatially invariant PSF. It therefore follows that if the pixel values of $\mathcal{F}$, $\mathcal{G}$ and $\mathcal{H}$ are the coefficients of their polynomial forms, $f(x, y)$, $g(x, y)$ and $h(x, y)$, respectively, then $g(x, y) = f(x, y)h(x, y)$. The polynomial computations are implemented by resultant matrices, of which there are several types, but only the Bézout matrix [12,20,21] and the Sylvester matrix [11,22,24,30,31] have been used for image deblurring. Section 2 contains a comparison of these matrices for greatest common divisor (GCD) computations, which are required for the determination of the size of the PSF and the solution of the problem of BID. The application of these matrices to image deblurring is considered in Sect. 3 and it is shown that, as for GCD computations, the Sylvester matrix yields better results than the Bézout matrix. Section 4 contains a summary of the paper.

The computations in all the examples were performed using MATLAB (64 bits).

2 Resultant Matrices and GCD Computations

It was stated in Sect. 1 that the convolution operation is common to polynomial multiplication and the formation of a

blurred image by a spatially invariant PSF. The equivalence of these operations is now summarised, and more details are in [30]. In particular, if the PSF is separable, then only one blurred image $\mathcal{G}$ is required for its determination, and this allows a deblurred form of $\mathcal{G}$ to be computed. The calculation of the PSF requires two GCD computations, one computation for the degree and coefficients of the column component $h_c(x)$ of the PSF and one computation for the degree and coefficients of the row component $h_r(y)$ of the PSF, and the PSF is equal to $h_c(x)h_r(y)$. A deblurred image is then obtained by two deconvolution computations, where the first computation yields a partially deblurred image $\tilde{\mathcal{F}}$ by deconvolving $h_c(x)\,(h_r(y))$ from the given blurred image $\mathcal{G}$, and the second computation yields a fully deblurred image by deconvolving $h_r(y)\,(h_c(x))$ from $\tilde{\mathcal{F}}$. The GCD computations can be implemented by resultant matrices, of which the Sylvester and Bézout are the most popular, and the deconvolution computation reduces to the solution of a linear algebraic equation whose coefficient matrix is Toeplitz. Both these computations are ill-posed and thus noise, which is present in all images, makes the computation of their improved forms difficult. In particular, the presence of noise implies it is necessary to consider an approximate greatest common divisor (AGCD), rather than the GCD, of two polynomials [36]. It is, however, convenient to consider the Sylvester and Bézout resultant matrices for the computation of the GCD of two exact polynomials, and to defer a discussion of their use for the computation of an AGCD of two noisy polynomials to Sect. 3, which considers the application of resultant matrices to image deblurring.

The Bézout matrix $B(f, g)$ of two polynomials $f(y)$ and $g(y)$, which are of degrees m and n, $m \geq n$, respectively,

$$f(y) = \sum_{i=0}^{m} a_i y^{m-i} \quad \text{and} \quad g(y) = \sum_{i=0}^{n} b_i y^{n-i}, \quad (2)$$

is used for GCD computations [4] and image deblurring [12, 20,21]. The matrix $B(f, g)$ is defined by

$$\frac{f(x)g(y) - f(y)g(x)}{x - y}$$

$$= \begin{bmatrix} x^{m-1} & x^{m-2} & \cdots & 1 \end{bmatrix} B(f, g) \begin{bmatrix} y^{m-1} \\ y^{m-2} \\ \vdots \\ 1 \end{bmatrix},$$

from which it follows that it is square and symmetric, and of order m. It is clear that if $f(y)$ and $g(y)$ are redefined as

$$f^*(y) = \sum_{i=0}^{m} a_i y^i \quad \text{and} \quad g^*(y) = \sum_{i=0}^{n} b_i y^i,$$

respectively, then

$$B(f, g) = J B(f^*, g^*) J,$$

where J is the reverse unit matrix of order m.

The Sylvester matrix $S(f, g)$ of $f(y)$ and $g(y)$ is square and of order $m + n$,

$$S(f, g) = \begin{bmatrix} C(f) & D(g) \end{bmatrix}, \qquad C(f) \in \mathbb{R}^{(m+n)\times n},$$
$$D(g) \in \mathbb{R}^{(m+n)\times m},$$

where $C(f)$ and $D(g)$ are Toeplitz matrices whose entries are the coefficients of $f(y)$ and $g(y)$, respectively [4],

$$C(f) = \begin{bmatrix} a_0 & & \\ a_1 & \ddots & \\ \vdots & \ddots & a_0 \\ a_{m-1} & \vdots & a_1 \\ a_m & \ddots & \vdots \\ & \ddots & a_{m-1} \\ & & a_m \end{bmatrix}, \qquad D(g) = \begin{bmatrix} b_0 & & \\ b_1 & \ddots & \\ \vdots & \ddots & b_0 \\ b_{n-1} & \vdots & b_1 \\ b_n & \ddots & \vdots \\ & \ddots & b_{n-1} \\ & & b_n \end{bmatrix}.$$
$$(3)$$

Theorem 1 shows that the degree and coefficients of the GCD of $f(y)$ and $g(y)$ can be computed from the QR decomposition of their Sylvester matrix [8, Theorems 2.5.2 and 2.5.3] and Bézout matrix [8, Theorems 2.6.7 and 2.6.11]. Theorem 2 shows that the degree and coefficients of the GCD of $f(y)$ and $g(y)$ can also be calculated from the leading principal submatrices of their Bézout matrix [5, Algorithm 9.1] and [8, Corollary 2.7.6 and Algorithm Bézout GCD].

Theorem 1 *Let the degree of the GCD of $f(y)$ and $g(y)$, which are defined in (2), be d. Let $Q^B R^B$ be the QR decomposition of $B(f, g)$ and let $Q^S R^S$ be the QR decomposition of $S(f, g)$. Then the rank loss of $B(f, g)$ and $S(f, g)$ is equal to d,*

$$\text{rank } B(f, g) = m - d \quad \text{and} \quad \text{rank } S(f, g) = m + n - d, \quad (4)$$

and the coefficients of the GCD are in the last non-zero rows of R^B and R^S.

Theorem 2 *Let $B_k(f, g)$ be the $k \times k$ leading principal submatrix of $B(f, g)$. The degree of the GCD of $f(y)$ and $g(y)$ is equal to d if and only if the following two conditions are satisfied:*

$$\det B_k(f, g) = 0, \quad k = m - d + 1, \ldots, m,$$
$$\det B_k(f, g) \neq 0, \quad k = m - d.$$

The coefficients of the GCD are obtained from computations on the solution x of the equation $B_{m-d}x = d$, where the entries of d are functions of the coefficients of $f(y)$ and $g(y)$.

Theorem 3 shows that the GCD of $f(y)$ and $g(y)$ can also be computed from their Sylvester matrix and its subresultant matrices $S_k(f, g), k = 1, \ldots, \min(m, n), S_1(f, g) = S(f, g)$, where the subresultant matrices are obtained by the deletion of rows and columns from $S(f, g)$. Examples of this computation when $f(y), g(y)$ and their GCD contain multiple roots of high degree are in [32,33].

Theorem 3 *Let $f(y)$ and $g(y)$ be real polynomials of degrees m and n, respectively, and let $S(f, g)$ be their Sylvester matrix. Let $S_k(f, g)$, which is of order $(m + n - k + 1) \times (m + n - 2k + 2)$, be the kth Sylvester subresultant matrix, $k = 1, \ldots, \min(m, n)$, where $S_1(f, g) = S(f, g)$. The degree d of their GCD is equal to the largest value of k for which $S_k(f, g)$ is rank deficient,*

$$\text{rank } S_k(f, g) < m + n - 2k + 2, \quad k = 1, \ldots, d,$$
$$\text{rank } S_k(f, g) = m + n - 2k + 2, \quad k = d + 1, \ldots, \min(m, n).$$
$$(5)$$

The coefficients of the coprime polynomials lie in the null space of $S_d(f, g)$.

Proof Since the degree of the GCD of $f(y)$ and $g(y)$ is d, there exist quotient polynomials $u_k(y)$ and $v_k(y)$, and a common divisor polynomial $t_k(y)$, such that

$$t_k(y) = \frac{f(y)}{u_k(y)} = \frac{g(y)}{v_k(y)},$$
$$\deg v_k < \deg g = n, \quad \deg u_k < \deg f = m, \quad (6)$$

for $k = 1, \ldots, d$, where

$$u_k(y) = \sum_{i=0}^{m-k} u_{k,i} y^{m-k-i}, \quad v_k(y) = \sum_{i=0}^{n-k} v_{k,i} y^{n-k-i},$$
$$t_k(y) = \sum_{i=0}^{k} d_{k,i} y^{k-i}.$$

It follows from (6) that

$$v_k(y) f(y) = u_k(y) g(y), \quad k = 1, \ldots, d, \quad (7)$$

and if $u_k(y)$ and $v_k(y)$ are equal to the zero polynomial for $k = d + 1, \ldots, \min(m, n)$, because $\deg \text{GCD}(f, g) = d$,

$$u_k(y) \equiv 0 \quad \text{and} \quad v_k(y) \equiv 0, \quad k = d + 1, \ldots, \min(m, n), \quad (8)$$

then (7) and (8) can be written in matrix form as

$$\begin{bmatrix} C_k(f) \ D_k(g) \end{bmatrix} \begin{bmatrix} v_k \\ -u_k \end{bmatrix} = S_k(f, g) \begin{bmatrix} v_k \\ -u_k \end{bmatrix} = 0,$$
$$k = 1, \ldots, \min(m, n), \quad (9)$$

where

$$u_k = \begin{bmatrix} u_{k,0} \ u_{k,1} \ \cdots \ u_{k,m-k-1} \ u_{k,m-k} \end{bmatrix}^{\mathrm{T}} \in \mathbb{R}^{m-k+1},$$
$$v_k = \begin{bmatrix} v_{k,0} \ v_{k,1} \ \cdots \ v_{k,n-k-1} \ v_{k,n-k} \end{bmatrix}^{\mathrm{T}} \in \mathbb{R}^{n-k+1},$$

$C_k(f) \in \mathbb{R}^{(m+n-k+1)\times(n-k+1)}$ and $D_k(g) \in \mathbb{R}^{(m+n-k+1)\times(m-k+1)}$ are Toeplitz matrices of the form (3), and

$$S_k(f, g) = \begin{bmatrix} C_k(f) \ D_k(g) \end{bmatrix}, \quad (10)$$

is the kth Sylvester subresultant matrix. The vectors u_k and v_k in (9) satisfy

$$u_k \neq 0, \quad v_k \neq 0, \quad k = 1, \ldots, d,$$
$$u_k = 0, \quad v_k = 0, \quad k = d + 1, \ldots, \min(m, n),$$

because the polynomials $f(y)$ and $g(y)$ possess common divisors of degrees $1, \ldots, d$, but they do not possess a common divisor of degree $k > d$. The result (5) follows from (9). □

Theorem 2 does not assign a value to, or limits on the values of, $\det B_k(f, g)$ for $k = 1, \ldots, m - d - 1$. This must be compared with Theorem 3, which distinguishes between the singular and non-singular subresultant matrices $S_k(f, g)$ for all values of k and therefore allows the value of k for which a change from singularity to non-singularity, as k increases, to be calculated. This property of the subresultant matrices is used in Sect. 3 to determine the size of the PSF that is used to blur an image.

Table 1 lists some differences between the Sylvester matrix $S(\tilde{f}, \tilde{g})$ and Bézout matrix $B(\tilde{f}, \tilde{g})$ of the polynomials $\tilde{f}(y)$ and $\tilde{g}(y)$,

$$\tilde{f}(y) = \sum_{i=0}^{m} \tilde{a}_i y^{m-i} \quad \text{and} \quad \tilde{g}(y) = \sum_{i=0}^{n} \tilde{b}_i y^{n-i}. \quad (11)$$

The properties of the Sylvester and Bézout matrices in Table 1 suggest that the Bézout matrix is preferred to the Sylvester matrix [28]. This preference is clear for the GCD computations in image deblurring, which require that $m = n$ and thus the Bézout matrix is half the size of the Sylvester matrix for this application, and it is also symmetric. Also, Boito [8, page 84] notes that the computation of the GCD of $f(y)$ and $g(y)$ using the QR decomposition of $S(f, g)$ and $B(f, g)$ may miss large common roots, and that this problem

Table 1 The properties of the Sylvester matrix $S(\tilde{f}, \tilde{g})$ and Bézout matrix $B(\tilde{f}, \tilde{g})$, where $\tilde{f}(y)$ and $\tilde{g}(y)$, which are of degrees m and n, respectively, are defined in (11)

Property	Sylvester matrix $S(\tilde{f}, \tilde{g})$	Bézout matrix $B(\tilde{f}, \tilde{g})$
Size	$(m + n) \times (m + n)$	$\max(m, n) \times \max(m, n)$
Symmetric	No	Yes
Additive	Yes: $S(\tilde{f}_1, \tilde{g}_1) + S(\tilde{f}_2, \tilde{g}_2)$ $= S(\tilde{f}_1 + \tilde{f}_2, \tilde{g}_1 + \tilde{g}_2)$	No: $B(\tilde{f}_1, \tilde{g}_1) + B(\tilde{f}_2, \tilde{g}_2)$ $\neq B(\tilde{f}_1 + \tilde{f}_2, \tilde{g}_1 + \tilde{g}_2)$
Entry (p, q)	$\bar{a}_i$ or $\bar{b}_j$ or zero	$\sum_{i,j}(\tilde{a}_i \tilde{b}_j - \tilde{a}_j \tilde{b}_i)$
Structure	$\begin{bmatrix} T_1(\tilde{f}) & T_2(\tilde{g}) \end{bmatrix}$; T_1, T_2 Toeplitz	Every entry is a bilinear term

The polynomials $\tilde{f}_1(y)$ and $\tilde{f}_2(y)$ are of the same degree, and similarly, the polynomials $\tilde{g}_1(y)$ and $\tilde{g}_2(y)$ are of the same degree

affects the Sylvester matrix more than the Bézout matrix. It is shown, however, in [33] that significantly improved results are obtained with the Sylvester matrix and its subresultant matrices $S_k(f, g)$, $k = 1, \ldots, \min(m, n)$, if $f(y)$ and $g(y)$ are processed by three operations before computations are performed on these matrices:

1. *Normalisation* Equation (10) shows that the coefficients of $f(y)$ and $g(y)$ are decoupled in $S_k(f, g)$ because they occupy, respectively, its first $n - k + 1$ columns and last $m - k + 1$ columns. If the coefficients of $g(y)$ are much larger than the coefficients of $f(y)$, then the left partition $C_k(f)$ is approximately equal to the zero matrix with respect to the right partition $D_k(g)$. This imbalance between the partitions of $S_k(f, g)$ can lead to incorrect results for GCD computations, and thus the first preprocessing operation is the normalisation of the coefficients of $f(y)$ and $g(y)$ in order to balance the left and right partitions of $S_k(f, g)$.

2. *Scale $g(y)$ by a Constant* The GCD of two polynomials is defined up to an arbitrary non-zero constant α, and thus

$$\text{GCD}(f, g) \sim \text{GCD}(f, \alpha g), \qquad \alpha \in \mathbb{R} \backslash 0,$$

where $\sim$ denotes equivalence to within an arbitrary non-zero constant. The matrices $S_k(f, g)$ are therefore written as

$$S_k(f, \alpha g) = \begin{bmatrix} C_k(f) & \alpha D_k(g) \end{bmatrix}, \qquad \|f\|_2 = \|g\|_2 = 1,$$

where α is a constant whose optimal value is determined from the solution of a linear programming problem. This issue is discussed below.

3. *Scale the Independent Variable* Computations on matrices whose entries vary widely in magnitude may cause numerical problems, and it is therefore desirable to minimise the ratio r of the maximum entry to the minimum entry of $S_k(f, \alpha g)$, for every value of k. This is accomplished by the substitution $y = \theta w$ in (2), where w is the new independent variable and θ is a parameter. It is shown in [33] that α_0 and θ_0, the optimal values of α and θ, are

chosen to minimise r, and that they are the solution of a linear programming problem. Furthermore, these optimal values are valid for all values of $k = 1, \ldots, \min(m, n)$, and thus the linear programming problem need only be solved once.

The application of these three preprocessing operations to the polynomials $f(y)$ and $g(y)$ in (2) yields the polynomials $\bar{f}(w)$ and $\alpha_0 \bar{g}(w)$ whose coefficients are $\bar{a}_i \theta_0^{m-i}$, $i = 0, \ldots, m$, and $\alpha_0 \bar{b}_j \theta_0^{n-j}$, $j = 0, \ldots, n$, respectively,

$$\bar{f}(w) = \sum_{i=0}^{m} \left(\bar{a}_i \theta_0^{m-i} \right) w^{m-i}$$

and

$$\alpha_0 \bar{g}(w) = \alpha_0 \sum_{j=0}^{n} \left(\bar{b}_j \theta_0^{n-j} \right) w^{n-j},$$

where $\bar{a}_i$ and $\bar{b}_j$ are the normalised coefficients of $f(y)$ and $g(y)$, respectively. All computations are performed on the Sylvester matrix and its subresultant matrices $S_k(\bar{f}, \alpha_0 \bar{g})$, $k = 1, \ldots, \min(m, n)$, and the importance of these operations for GCD computations with the Sylvester matrix is shown in [32,33]. They are also required for the computation of multiple roots of a polynomial, where the multiplicities of its distinct roots are obtained by a series of GCD computations and polynomial deconvolutions [29,34].

Examples 1 and 2 consider Theorems 1, 2 and 3 for the calculation of the degree of the GCD of two polynomials.

Example 1 Consider the polynomials $f(y)$ and $g(y)$ whose GCD is of degree $d = 9$,

$$f(y) = (y + 2)^2 (y + 1)^2 (y - 1)^3 (y - 3)(y - 5)^3 (y - 7),$$

and

$$g(y) = (y + 2)^4 (y - 1)^3 (y - 3)(y - 5)^3 (y - 8),$$

and for which $m = 12$. Noise δa_i and δb_j was added to the coefficients a_i and b_j of, respectively, $f(y)$ and $g(y)$,

$$\delta a_i = \varepsilon r_i a_i, \quad i = 0, \dots, 12,$$

and

$$\delta b_j = \varepsilon r_j b_j, \quad j = 0, \dots, 12,$$

where $\varepsilon = 10^{-5}$, and r_i and r_j are uniformly distributed random variables in the interval $[-1, 1]$. Figure 1 shows the normalised singular values of $B(f, g)$, with and without noise, and it is seen that, in the absence of noise, the rank of $B(f, g)$ is equal to 3, as required from (4) because $m = 12$ and $d = 9$. The figure also shows that the addition of noise causes a significant deterioration in the results because the maximum change (about four orders of magnitude) in the normalised singular values occurs between $i = 3$ and $i = 4$, but this change is significantly smaller than in the absence of noise.

Theorem 2 requires that the rank of the principal submatrices of $B(f, g)$ be considered and Fig. 2 shows the variation of the condition number $\kappa(B_k(f, g))$ of $B_k(f, g)$ with the order k of the submatrix, with and without noise. It is seen that, in the absence of noise, it decreases slowly for $k = 12, 11, \dots, 5, 4$, and that there is a significant change in $\kappa(B_k(f, g))$ between $k = 4$ and $k = 3$. Theorem 2 suggests, therefore, that the value of d is given by $m - d = 3$, that is, $d = 9$, which is correct. The condition number of $B_3(f, g)$ is approximately equal to 10^5, which is large, but many orders of magnitude smaller than the values of $\kappa(B_k(f, g))$, $k = 4, \dots, 12$. Figure 2 shows that the degree of the GCD of $f(y)$ and $g(y)$ cannot be determined in the presence of noise.

Figure 3 shows the normalised singular values of $S(\bar{f}, \alpha_0 \bar{g})$, with and without noise. In both cases, the maximum

change in the normalised singular values occurs from $i = 15$ to $i = 16$, and thus the computed degree of the GCD is $d = 9$, which is correct. Figure 4 shows the condition number of each Sylvester subresultant matrix $S_k(\bar{f}, \alpha_0 \bar{g})$ and it is seen that, in the presence of noise, the maximum change in the condition number, about four orders of magnitude, occurs from $k = 9$ to $k = 10$, and thus the computed degree of the GCD is correct.   □

Example 2 The Sylvester matrix and subresultant matrices, and the Bézout matrix and principal submatrices, of the polynomials

$$f(y) = 6y^{13} + 44y^{11} + 138y^9 + 120y^7$$
$$- 6y^6 - 39y^4 - 110y^2 - 74,$$

and

$$g(y) = 2y^{11} + 12y^9 + 30y^7 - 2y^4 - 11y^2 - 26,$$

were formed and the procedure described in Example 1 was followed, except that noise was not added to the polynomials. Figure 5 shows the normalised singular values of $B(f, g)$, and it is seen that the degree of the GCD of $f(y)$ and $g(y)$ is four, and that this rank deficiency is clearly defined. Figure 6 shows the variation of the condition number of the principal submatrices $B_k(f, g)$ with k, but a conclusion cannot be drawn from the graph.

Figure 7 shows the normalised singular values of the Sylvester matrix $S(\bar{f}, \alpha_0 \bar{g})$, and Fig. 8 shows the condition numbers of the Sylvester matrix and its subresultant matrices $S_k(\bar{f}, \alpha_0 \bar{g})$, $k = 1, \dots, 11$. These figures are consistent with Fig. 5 because they show that the degree of the GCD of $f(y)$ and $g(y)$ is four. Furthermore, the rank deficiency is clearly defined in Figs. 5, 7 and 8, which must be compared with the unsatisfactory result in Fig. 6.

Fig. 1 The normalised singular values σ_i / σ_1 of $B(f, g)$ against i, with and without noise, for Example 1
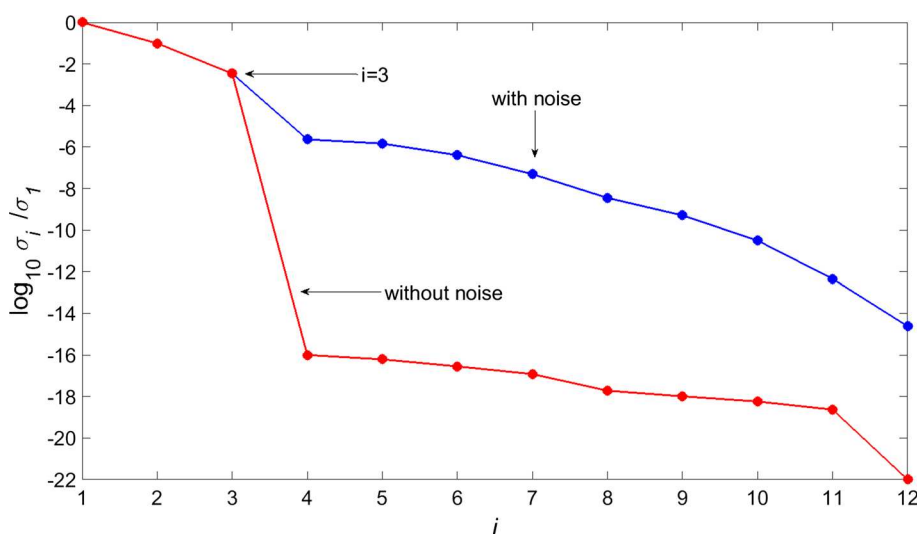
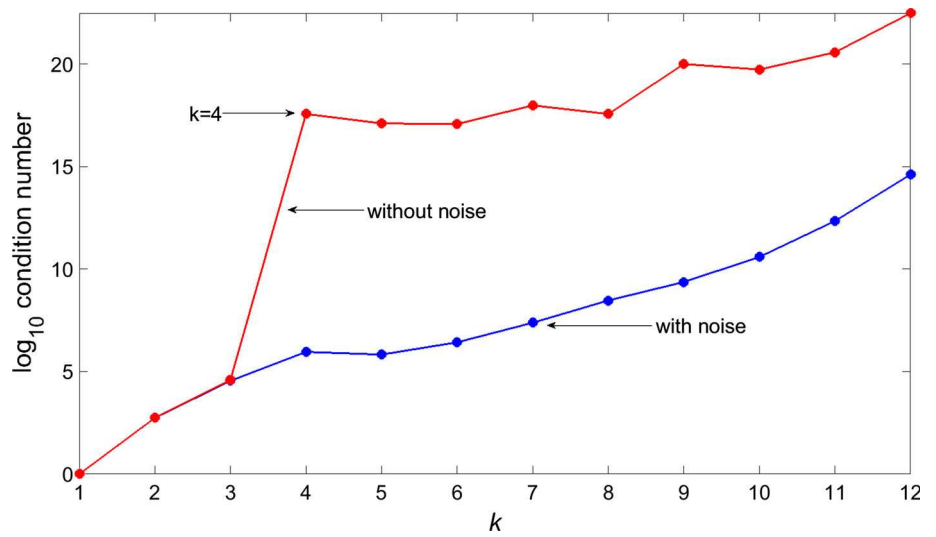**Fig. 2** The condition number of $B_k(f, g)$ against k, with and without noise, for Example 1

Fig. 3 The normalised singular values σ_i/σ_1 of $S(\bar{f}, \alpha_0\bar{g})$ against i, with and without noise, for Example 1
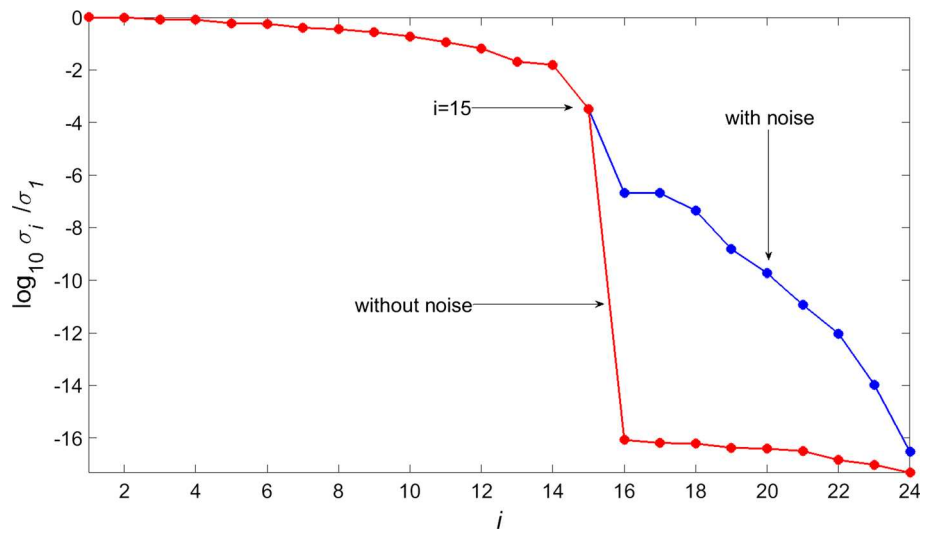

Fig. 4 The condition number of $S_k(\bar{f}, \alpha_0\bar{g})$ against k, with and without noise, for Example 1
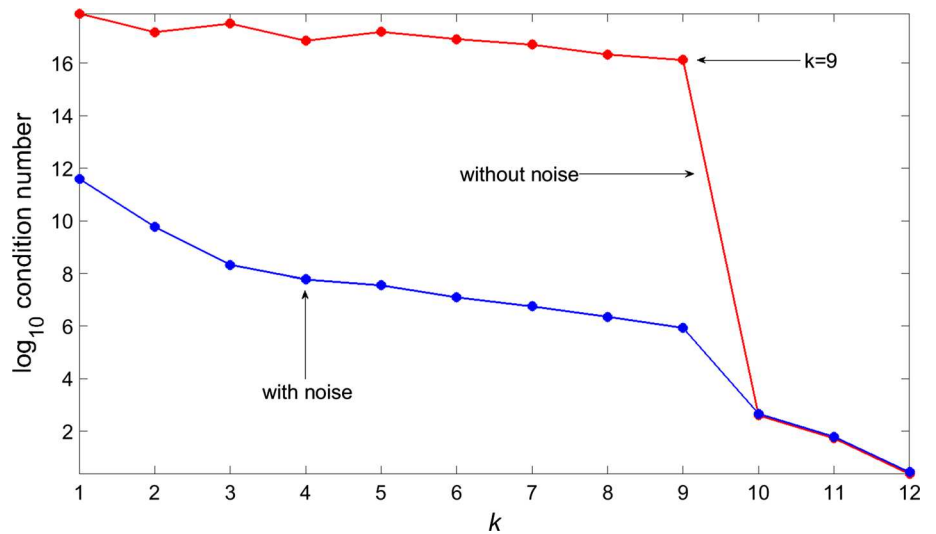
Fig. 5 The normalised singular values σ_i/σ_1 of $B(f, g)$ against i for Example 2
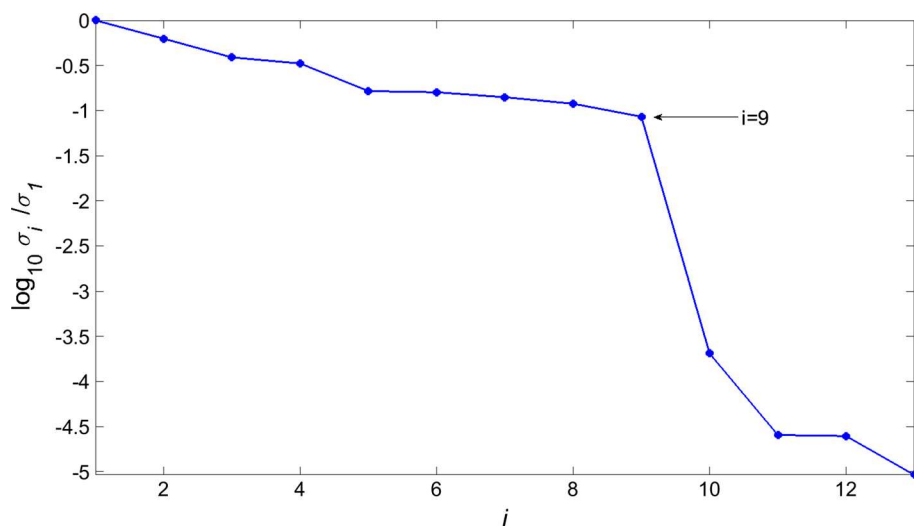

i=9

Fig. 6 The condition number of $B_k(f, g)$ against k for Example 2
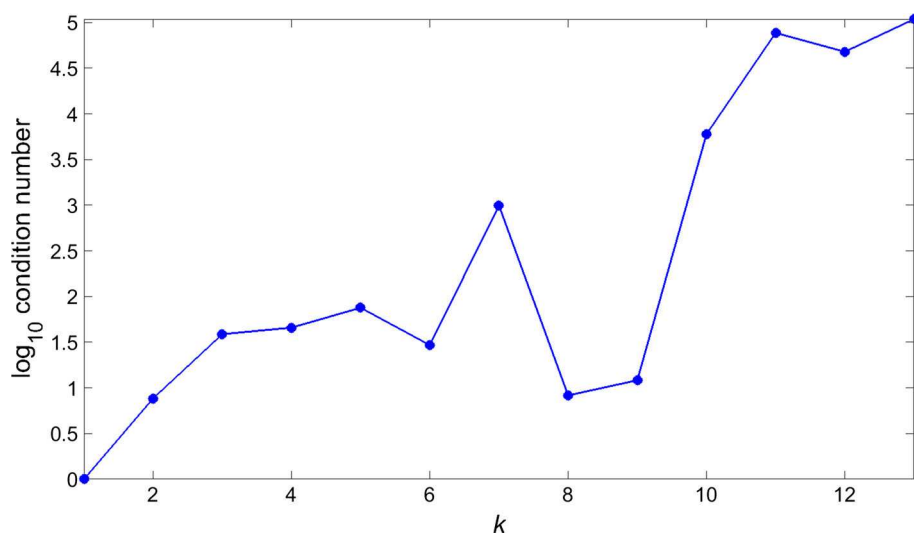
Fig. 7 The normalised singular values σ_i/σ_1 of $S(\bar{f}, \alpha_0\bar{g})$ against i for Example 2
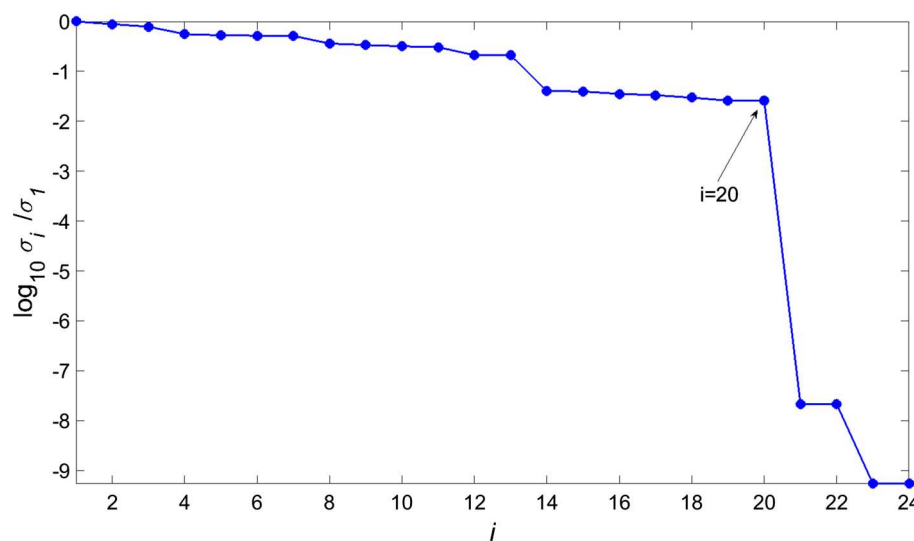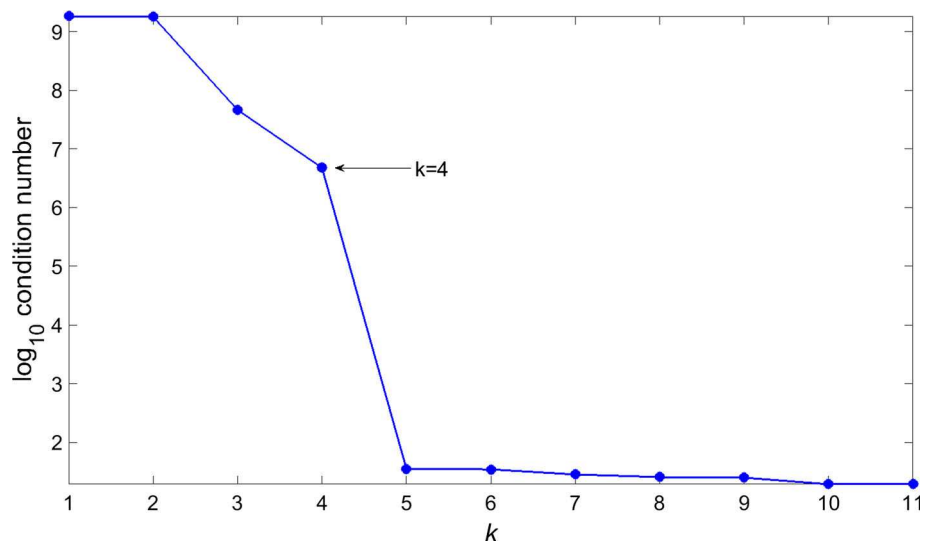

i=20

Fig. 8 The condition number of $S_k(\bar{f}, \alpha_0 \bar{g})$ against k for Example 2

The polynomials $f(y)$ and $g(y)$ are, however, coprime, and Table 2 shows they have four very close roots. It therefore follows that $B(f, g)$ and $S(f, g)$ are, from theoretical considerations, non-singular, and the proximity of four roots of $f(y)$ to four roots of $g(y)$ manifests itself in the near-singularity of these matrices, with a rank loss of four, that is, these matrices are numerically singular. This distinction between the theoretical result, $B(f, g)$ and $S(f, g)$ are non-singular, and the computational result, $B(f, g)$ and $S(f, g)$ are numerically singular with a rank loss of four, can be quantified by calculating the relative separation of the roots in Table 2. In particular, if α_1 and α_2 are the roots of $f(y)$ with positive imaginary parts,

$$\alpha_1 = -0.660248786 + 1.85420355i,$$
$$\alpha_2 = 0.661093255 + 1.85335080i,$$

and β_1 and β_2 are the roots of $g(y)$ with positive imaginary parts,

$$\beta_1 = -0.660249099 + 1.85420406i,$$
$$\beta_2 = 0.661093571 + 1.85335129i,$$

then the relative separation of α_1 and β_1, and their complex conjugates, is

$$\frac{|\alpha_1 - \beta_1|}{0.5\,|\alpha_1 + \beta_1|} = 3.04 \times 10^{-7}.$$

Similarly, the relative separation of α_2 and β_2, and their complex conjugates, is equal to 2.96×10^{-7}. □

The singular value decomposition (SVD) was used in Examples 1 and 2 to calculate the degree of the GCD of two polynomials, but it is computationally expensive. Furthermore, it will be shown in Example 4 that the calculation

Table 2 Four of the roots of $f(y)$ and four of the roots of $g(y)$, for Example 2

Four roots of $f(y)$	Four roots of $g(y)$
$-0.660248786 \pm 1.85420355i$	$-0.660249099 \pm 1.85420406i$
$0.661093255 \pm 1.85335080i$	$0.661093571 \pm 1.85335129i$

of the size of a separable PSF requires 25 GCD computations in each direction, which implies that a large number of SVDs must be computed. It is therefore desirable to consider computationally cheaper methods for the calculation of the size of the PSF. Sections 2.1 and 2.2 consider, respectively, methods for the computation of the degree and coefficients of the GCD of two polynomials.

2.1 The Degree of the GCD of Two Polynomials

The Sylvester and Bézout matrices require different methods for the calculation of the degree of the GCD of two polynomials and they are therefore considered separately.

Theorem 1 shows that the degree d of the GCD of $f(y)$ and $g(y)$ can be calculated from the rank loss of the Sylvester matrix $S(f, g)$, and Theorem 3 shows that d can also be calculated from the change from singularity to non-singularity of the Sylvester subresultant matrices. These methods require the SVD, but the presence of noise may lead to a significant deterioration in the results, as shown in Examples 1 and 2, because the small singular values of a matrix A are sensitive to perturbations in A. Also, the SVD is cubic in complexity and its application to each subresultant matrix in Theorem 3 is therefore expensive. This theorem can, however, be implemented efficiently by the QR decomposition, which is cubic in complexity, of $S(f, g)$, and the QR decomposition of each subresultant matrix is then computed using

its update formula, which is quadratic in complexity [14], because $S_{k+1}(f, g)$ is formed by the deletion of two columns and one row from $S_k(f, g)$. The calculation of the degree of the GCD therefore reduces to the calculation of the singular or non-singular nature of each subresultant matrix $S_k(f, g)$ from the upper triangular matrix R_k, where $Q_k R_k$ is the QR decomposition of $S_k(f, g)$.

Care must be exercised when the QR decomposition is used to compute the rank r of a matrix $A \in \mathbb{R}^{p \times q}$, $p \geq q$, because if the QR decomposition of $A\Pi$, where Π is a permutation matrix, is

$$A\Pi = \text{QR} = Q \begin{bmatrix} R_{11} & R_{12} \\ & R_{22} \end{bmatrix},$$

where $R_{22} \in \mathbb{R}^{r \times r}$, then $\sigma_{q-r+1} \leq \|R_{22}\|_2$ where σ_i is the ith singular value of A and the singular values are arranged in non-increasing order. It follows that if $\|R_{22}\|_2$ is small, then A has r small singular values, but the converse is not true because it does not follow that, even with column pivoting, the existence of r small singular values implies that $\|R_{22}\|_2$ is small. Although $\|R_{22}\|_2$ cannot be used to calculate the rank of $A\Pi$, Example 3 shows that the rank of $S_k(\bar{f}, \alpha_0\bar{g})$ can be calculated from the row sums and diagonal entries of R_k, where $S_k(\bar{f}, \alpha_0\bar{g}) = Q_k R_k$.

Example 3 Let the degrees of the polynomials $p(y)$ and $q(y)$ be 150 and 200, respectively, and let their coefficients be uniformly distributed random variables in the ranges $[-10, 10]$ and $[-10, 20]$, respectively. The polynomial $d(y)$ is of degree 70, its coefficients are uniformly distributed random variables in the range $[-30, 30]$, and it satisfies $d(y) = \text{GCD}(f, g)$, where $f(y)$ are $g(y)$ are defined by

$$f(y) = p(y)d(y) \quad \text{and} \quad g(y) = q(y)d(y).$$

The polynomials $\bar{f}(w)$ and $\alpha_0\bar{g}(w)$ were obtained by applying the preprocessing operations to $f(y)$ and $g(y)$, as described in Sect. 2. Figures 9 and 10 show, respectively, the diagonal entries of R_k and the row sums of R_k for $k = 1, \ldots, 220$. The points that define the subresultant matrix indexed by $k = 70$ are marked in the figures, and it is seen that they can, in principle, be used to determine the degree of the GCD of $f(y)$ and $g(y)$. The figures also show that the diagonal entries and row sums of R_k have structure. □

Example 3 shows that the degree d of the GCD of $f(y)$ and $g(y)$ (or equivalently, the degree of the GCD of $\bar{f}(w)$ and $\bar{g}(w)$) requires the calculation of λ_k, the difference between the maximum diagonal entry and the minimum diagonal entry of R_k, and μ_k, the difference between the maximum row sum and the minimum row sum of R_k,

$$\lambda_k = \max_i \log_{10} |R_{k,i,i}| - \min_i \log_{10} |R_{k,i,i}|, \tag{12}$$

$$\mu_k = \max_i \log_{10} \sum_j |R_{k,i,j}| - \min_i \log_{10} \sum_j |R_{k,i,j}|, \tag{13}$$

for $k = 1, \ldots, \min(m, n)$, where $R_{k,i,j}$ is entry (i, j) of R_k. Figures 9 and 10 show that d is defined by the maximum change in λ_k and μ_k between successive values of k,

$$d = \begin{cases} \arg\max_k(\lambda_k - \lambda_{k+1}), & k = 1, \ldots, \min(m, n) - 1, \\ \arg\max_k(\mu_k - \mu_{k+1}), & k = 1, \ldots, \min(m, n) - 1. \end{cases}$$

These methods of calculating d from the Sylvester resultant matrix and its subresultant matrices are heuristic and they are not mathematically rigorous. They are, however, justified by computational experiments, which show that they lead to good results for GCD computations. Also, Example 4 shows that good results are obtained when they are used to calculate the size of the PSF from a blurred image in the presence of added noise and uncertainty in the PSF.

Example 1 shows that the SVD cannot be used to calculate the degree of the GCD of two polynomials from their Bézout matrix $B(f, g)$ and its leading principal submatrices $B_k(f, g)$ because it yields unsatisfactory results when the polynomials are perturbed by noise. Furthermore, Example 2 shows that, even in the absence of added noise, the matrices $B_k(f, g)$ may return unsatisfactory results. These issues are addressed in Sect. 2.3.

2.2 The Coefficients of the GCD of Two Polynomials

The computation of the coefficients of the GCD of $f(y)$ and $g(y)$ from their Sylvester matrix is considered initially, after which the Bézout matrix is considered for this computation.

It is assumed that the degree d of the GCD has been calculated, and that the polynomials have been preprocessed, as discussed in Sect. 2, thereby yielding the polynomials $\bar{f}(w)$ and $\alpha_0\bar{g}(w)$. The calculation of the coefficients of $\bar{t}(w)$, the GCD of $\bar{f}(w)$ and $\alpha_0\bar{g}(w)$, is addressed for Bernstein basis polynomials in [9], and the same method can be used for power basis polynomials. In particular, Theorem 3 shows that $S_d(\bar{f}, \alpha_0\bar{g})$, which is of order $(m+n-d+1) \times (m+n-2d+2)$, has unit rank loss and (9) shows that the coprime polynomials $\bar{v}_d(w)$ and $\bar{u}_d(w)$ lie in the null space of $S_d(\bar{f}, \alpha_0\bar{g})$. There is therefore one equation that defines the linear dependence of the columns of $S_d(\bar{f}, \alpha_0\bar{g})$, and if the pth column of $S_d(\bar{f}, \alpha_0\bar{g})$ is one of these linearly dependent columns and it is denoted by b, then the homogeneous equation (9) can be written as

$$\bar{A}_d\bar{x} \approx \bar{b}, \qquad \bar{A}_d = \bar{A}_d(\bar{f}, \alpha_0\bar{g}) \in \mathbb{R}^{(m+n-d+1) \times (m+n-2d+1)}, \tag{14}$$

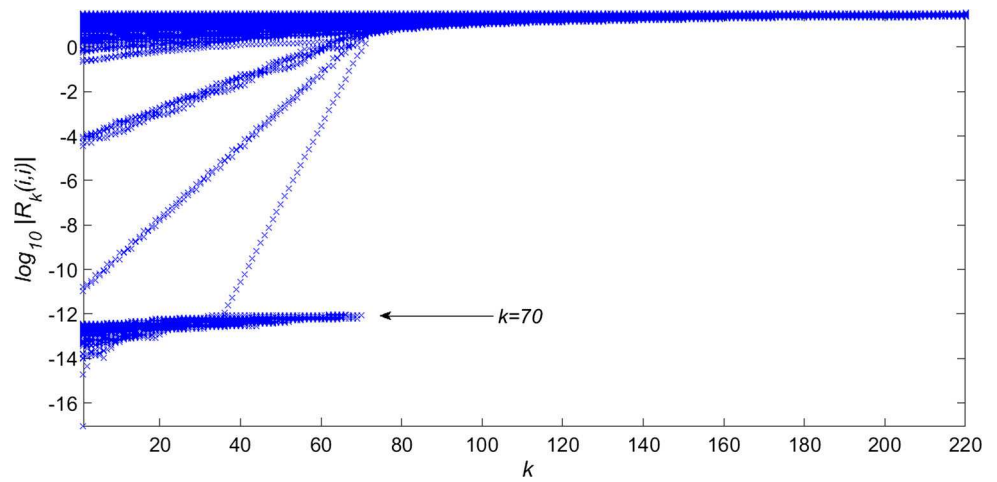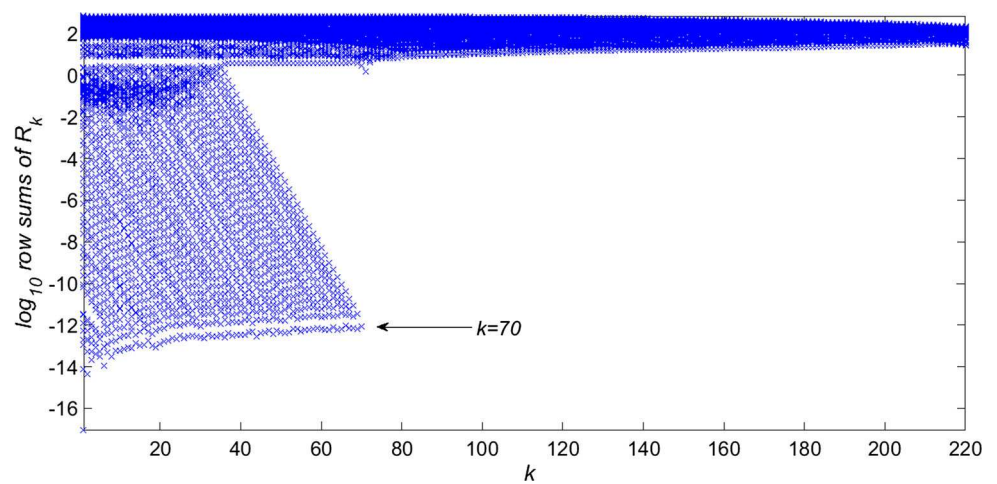**Fig. 9** The magnitudes of the diagonal entries $R_k(i,i)$ for Example 3

Fig. 10 The row sums of R_k for Example 3

where it follows from (10) that the coefficient matrix $\bar{A}_d$ is formed by the concatenation of two Toeplitz matrices and the subsequent removal of the pth column, the $=$ is replaced by an $\approx$ because it is assumed that $f(y)$ and $g(y)$ (and therefore $\bar{f}(w)$ and $\bar{g}(w)$) are corrupted by added noise and they therefore have an AGCD, not a GCD,

$$\bar{x} = \begin{bmatrix} \bar{x}_1 \\ \vdots \\ \bar{x}_{p-1} \\ \bar{x}_{p+1} \\ \vdots \\ \bar{x}_{m+n-2d+2} \end{bmatrix}, \qquad \begin{bmatrix} \bar{x}_1 \\ \vdots \\ \bar{x}_{p-1} \\ -1 \\ \bar{x}_{p+1} \\ \vdots \\ \bar{x}_{m+n-2d+2} \end{bmatrix} = \begin{bmatrix} \bar{v}_d \\ -\bar{u}_d \end{bmatrix},$$

and $\bar{v}_d$ and $\bar{u}_d$ are vectors that contain the coefficients of the coprime polynomials $\bar{v}_d(w)$ and $\bar{u}_d(w)$,

$$\bar{f}(w) \approx \bar{u}_d(w)\bar{t}(w) \qquad \text{and} \qquad \alpha_0 \bar{g}(w) \approx \bar{v}_d(w)\bar{t}(w).$$

The structure of $\bar{A}_d$ allows (14) to be replaced by

$$(\bar{A}_d + \bar{E}_d)\tilde{x} = \bar{b} + \bar{e}, \qquad \bar{E}_d \in \mathbb{R}^{(m+n-d+1)\times(m+n-2d+1)}, \tag{15}$$

where $\bar{E}_d$ has the same structure as $\bar{A}_d$, and $\bar{e}$ has the same structure as $\bar{b}$, from which it follows that a structure-preserving matrix method is used to compute its solution, that is, the matrix $\bar{E}_d$ and the vectors $\bar{e}$ and $\tilde{x}$ [25]. The solution is under-determined but it is shown in [9] that the addition of a constraint allows a unique solution to be computed.

The approximation (14) and the exact equation (15) can be interpreted in terms of polynomial computations. In particular, it follows from (7) that (14) is equivalent to the approximate equality of two deconvolutions,

$$\frac{\bar{f}(w)}{\bar{u}_d(w)} \approx \frac{\alpha_0 \bar{g}(w)}{\bar{v}_d(w)},$$

and (15) is obtained by the addition of the polynomials $\delta \bar{f}(w)$, $\alpha_0 \delta \bar{g}(w)$, $\delta \bar{u}_d(w)$ and $\delta \bar{v}_d(w)$ to $\bar{f}(w)$, $\alpha_0 \bar{g}(w)$,

$\bar{u}_d(w)$ and $\bar{v}_d(w)$, respectively, such that the approximate deconvolutions are replaced by exact deconvolutions,

$$\frac{\bar{f}(w) + \delta\bar{f}(w)}{\bar{u}_d(w) + \delta\bar{u}_d(w)} = \frac{\alpha_0\left(\bar{g}(w) + \delta\bar{g}(w)\right)}{\bar{v}_d(w) + \delta\bar{v}_d(w)},$$

where the entries of $\bar{E}_d$ and $\bar{e}$ in (15) are the coefficients of $\delta\bar{f}(w)$ and $\alpha_0\delta\bar{g}(w)$, and the entries of $\tilde{x}$ are the coefficients of the coprime polynomials $\bar{u}_d(w) + \delta\bar{u}_d(w)$ and $\bar{v}_d(w) + \delta\bar{v}_d(w)$.

The solution of (15) yields the corrected polynomials $\tilde{f}(w)$ and $\tilde{g}(w)$, and the corrected coprime polynomials $\tilde{u}_d(w)$ and $\tilde{v}_d(w)$,

$$\tilde{f}(w) = \bar{f}(w) + \delta\bar{f}(w), \qquad \tilde{g}(w) = \alpha_0\left(\bar{g}(w) + \delta\bar{g}(w)\right),$$
$$\tilde{u}_d(w) = \bar{u}_d(w) + \delta\bar{u}_d(w), \quad \tilde{v}_d(w) = \bar{v}_d(w) + \delta\bar{v}_d(w).$$

These corrected polynomials allow an AGCD $\tilde{t}(w)$ to be computed,

$$\begin{bmatrix} T(\tilde{u}_d) \\ T(\tilde{v}_d) \end{bmatrix}\tilde{t} = \begin{bmatrix} \tilde{f} \\ \tilde{g} \end{bmatrix}, \quad T(\tilde{u}_d) \in \mathbb{R}^{(m+1)\times(d+1)}, \quad T(\tilde{v}_d) \in \mathbb{R}^{(n+1)\times(d+1)},$$

where $T(p)$ is a Toeplitz matrix whose entries are the coefficients of the polynomial $p(w)$, and $\tilde{t}$, $\tilde{f}$ and $\tilde{g}$ are vectors of the coefficients of $\tilde{t}(w)$, $\tilde{f}(w)$ and $\tilde{g}(w)$, respectively.

Consider now the computation of the coefficients of the GCD $t(y)$ of $f(y)$ and $g(y)$ from their Bézout matrix $B(f, g)$. The preprocessing operations that are necessary for the Sylvester matrix are not required for the Bézout matrix and computations can therefore be performed directly on $B(f, g)$. The simplest method requires the QR decomposition of $B(f, g)$ because the coefficients of $t(y)$ are in the last non-zero row of the upper triangular matrix R [8, Theorem 2.6.11]. Another method requires the solution of the equation $Ax = b$ where A is a square symmetric matrix of order $m - d$ [5, Algorithm 9.1]. This method is used in [20,21] to calculate the size of a separable PSF from a blurred image, and these two methods (the QR decomposition and the solution of $Ax = b$) are compared in Example 5 for the computation of a deblurred image. Finally, it is noted that the Bézout matrix does not satisfy an equation that is similar to (15) because, as stated in Table 1, its entries are bilinear, not linear.

2.3 Comparison of the Sylvester and Bézout Matrices for AGCD Computations

Some properties of the Sylvester and Bézout matrices are stated in Table 1, and the smaller size and symmetry of the Bézout matrix suggest it is preferred to the Sylvester matrix for AGCD computations. Numerical experiments show, however, that the Sylvester matrix yields better results because it is less sensitive to noise than the Bézout matrix, that is, the Sylvester matrix returns the correct degree of the GCD, and coefficients with a small backward error, for much higher noise levels than the Bézout matrix. The Bézout matrix $B(f, g)$ may yield poor results because each of its entries requires one or more computations of the form $\sum_{i,j}(a_i b_j - a_j b_i)$, which results in a large error, due to floating point cancellation, when $|a_i b_j - a_j b_i| \ll 1$ [7]. Furthermore, this error is significant in the presence of additive noise, which is much larger than the error due to floating point cancellation. By contrast, each non-zero entry of the Sylvester matrix is either a_i or b_j, and thus this error does not occur. This problem also arises when computations are performed on the Bézout matrix of polynomials expressed in the Bernstein basis [35], and the results in this paper show that if $f(y)$ and $g(y)$ are processed before their Sylvester matrix is formed, as discussed in Sect. 2, then the Sylvester matrix yields significantly better results than the Bézout matrix.

Figures 2 and 6 are consistent with the results of other examples considered by the authors, and with the result of Bini and Gemignani [6], who obtain very large condition numbers of the leading principal submatrices $B_k(f, g)$. Their results also show that the ratio of the smallest singular values of two successive leading principal submatrices exhibits dramatic changes, such that computations based on these singular values are unreliable. These results imply that GCD computations that require the leading principal submatrices of $B(f, g)$ are prone to numerical instability and they are therefore not recommended.

3 Resultant Matrices and Image Deblurring

This section considers the Sylvester and Bézout matrices for the solution of the problem of BID in order to determine if the Sylvester matrix yields better deblurred images than the Bézout matrix.

The authors of [20,21] claim they successfully solved the problem of BID by computing a deblurred image using the Bézout matrix. It is assumed in these references that the same image $\mathcal{F}$ is blurred by two different PSFs, $\mathcal{H}_1$ and $\mathcal{H}_2$, thereby yielding the blurred images $\mathcal{G}_1$ and $\mathcal{G}_2$,

$$\mathcal{G}_1 = \mathcal{F} * \mathcal{H}_1 + \mathcal{N}_1 \quad \text{and} \quad \mathcal{G}_2 = \mathcal{F} * \mathcal{H}_2 + \mathcal{N}_2, \quad (16)$$

where $\mathcal{N}_1$ and $\mathcal{N}_2$ are the noise samples added to the first and second blurred images, respectively, and $*$ denotes convolution. These equations can be expressed in polynomial form,

$$g_1(x, y) = f(x, y)h_1(x, y) + n_1(x, y),$$
$$g_2(x, y) = f(x, y)h_2(x, y) + n_2(x, y),$$

from which it follows that $f(x, y)$ (the exact image, which is to be computed) is an AGCD of $g_1(x, y)$ and $g_2(x, y)$ (the given blurred images) if $h_1(x, y)$ and $h_2(x, y)$ (the PSFs) are coprime polynomials. It is shown in [20,21] that these bivariate polynomial computations can be reduced to a series of univariate polynomial AGCD computations, from which a deblurred image can be calculated, using the leading principal submatrices of the Bézout matrix. The authors claim good results for these computations, despite its numerical problems, which are discussed in Sect. 2. The MATLAB code (which is referenced in [20,21]) was therefore examined, and it is clear that the authors *hardcode* the size of the PSF as arguments of the function that deblurs the images, that is, the size of the PSF is *defined, not calculated*. It therefore follows that these authors solve the problem of semi-blind deconvolution because they calculate only the coefficients of the PSF, and they do not calculate its size. This semi-blind deconvolution problem is significantly easier than the totally blind deconvolution problem because the calculation of the size of the PSF is not trivial since it reduces to the calculation of the (numerical) rank of a matrix whose entries are corrupted by noise.

A separable Gaussian PSF is used in [20,21], and comparison of the Sylvester and Bézout matrices for BID therefore requires that a PSF that satisfies this property be used. This property is not realistic for practical problems, but it is the simplest form of a PSF because a deblurred image can be calculated from one blurred image [30,31], and the computations are simplified because the Fourier transform is not required. By contrast, two blurred images are required for the computation of a non-separable PSF, and the Fourier transform is used to reduce the two-dimensional BID problem to two sets of GCD computations on univariate polynomials. This approach is used in [22,24], but the signal-to-noise ratios (SNRs) of the images in the examples in these references are 50 dB and 45 dB, respectively, which are very large values of the SNR of an image. Experimental results by the authors of this paper showed that the solution of the BID problem for a non-separable PSF degrades rapidly in the presence of noise because of the use of the Fourier transform in the computations, and it is suggested that this deterioration is the reason for the very high SNRs in the examples in [22,24]. Since the objective of this paper is a comparison of the Sylvester and Bézout matrices for BID, it is necessary to remove all sources of error that could make the analysis and interpretation of the results difficult. It therefore follows that, for the purposes of this paper, it is necessary to consider a separable PSF. Furthermore, this comparison is included in Example 4, for which the SNR of the blurred and noisy image is 12 dB, and Example 7 shows that a weakly non-separable PSF can be approximated by a separable PSF in the absence of noise and uncertainty in the PSF.

A blurred image $\mathcal{G}$ is formed by the convolution of the exact image $\mathcal{F}$ and the PSF $\mathcal{H}$, as shown in (16) for two blurred images, and it necessarily follows from this model that $\mathcal{G}$ is larger than $\mathcal{F}$. This difference in the sizes of $\mathcal{G}$ and $\mathcal{F}$ is not satisfied in practical problems because the given blurred image and computed deblurred image are the same size. Deconvolution of $\mathcal{H}$ from $\mathcal{G}$ in computational experiments requires, therefore, that $\mathcal{G}$ be cropped to the same size as $\mathcal{F}$ after it has been formed by the convolution operation, and the borders of this cropped image must then be restored by extrapolation, thereby yielding a modified blurred image $\mathcal{G}^*$. The best extrapolation function would return $\mathcal{G}^* = \mathcal{G}$, in which case deconvolution of $\mathcal{H}$ from $\mathcal{G}^*$ would return $\mathcal{F}$. It is, however, now shown that the extrapolation of the border pixels of the cropped form of $\mathcal{G}$ introduces several complications.

The number of pixels that are extrapolated is a function of the size of the PSF because if the size of the PSF is $r \times s$ pixels, where r and s are odd and, by definition of the problem of BID, unknown, then $(s - 1)/2$ pixels must be extrapolated on the left and right borders of the cropped form of $\mathcal{G}$, and $(r - 1)/2$ pixels must be extrapolated at the top and bottom borders of the cropped form of $\mathcal{G}$, thereby yielding the image $\mathcal{G}^*$ that is the same size as $\mathcal{G}$. Different extrapolation functions give rise to different images $\mathcal{G}^*$ and, in the absence of prior information, it is necessary to compute several deblurred images that differ in the values of r and s, and it may also be necessary to consider different extrapolation functions. If a quantitative measure of the quality of a deblurred image cannot be developed, then visual inspection of the deblurred images is required to determine the best deblurred image. This visual test is simplified if, for a given extrapolation function, the difference in the deblurred images obtained with the correct values of r and s, and with incorrect values of r and s, is significant. The effect on the deblurred image of a difference in the size of the PSF that is used to blur an image, and then deblur its blurred form, is considered in Example 6.

The discussion above shows that the problem of BID requires careful consideration of the boundary conditions. It is clear that exponential extrapolation functions are a natural choice because the extrapolated pixel values are positive and they decay smoothly as the distance from the centre of the PSF increases. These boundary conditions for the solution of the problem of BID are significantly different from the boundary conditions that are imposed when the PSF is known and the deblurred image is obtained from the solution of a linear algebraic equation $Ax = b$. In particular, the entries of A are functions of the PSF and they are therefore not known when the problem of BID is to be solved. If, however, the PSF (and therefore A) are known, the simplest boundary condition is the zero boundary condition, which yields satisfactory results when the image outside the field of view is black, but artefacts, for example, ringing at the boundaries, may be obtained for images that do not satisfy this property. This has

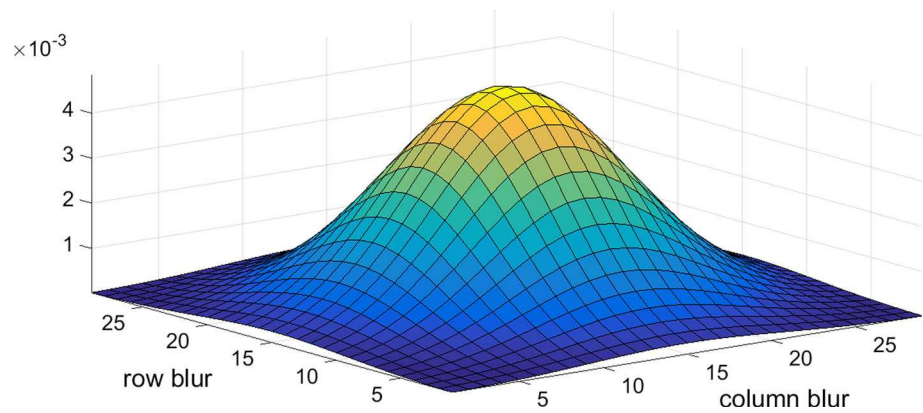**Fig. 11** The Gaussian PSF for Examples 4 and 5

Fig. 12 The exact image (left) and blurred image (right) for Examples 4 and 5, and Example 4, respectively

led to the development of anti-reflective boundary conditions [2,27], synthetic boundary conditions [13], and periodic and reflexive boundary conditions [16]. Anti-reflective boundary conditions are the preferred boundary conditions in the absence of noise because discontinuities in the image and its derivative at its edges are not imposed, thereby minimising the effects of ringing. None of these boundary conditions are physically realistic, but they are imposed in order to preserve structure in the coefficient matrix A and thus allow fast algorithms to be used [1]. They are imposed for mathematical convenience and must be compared with the more realistic boundary conditions defined by extrapolation functions, which allow the extrapolated pixels to be estimated from the pixels that lie in the border region of the cropped image.

The cropping and extrapolation procedures required for the solution of the problem of BID are not implemented in [20,21], and comparison of the Sylvester and Bézout matrices therefore requires that the blurred image $\mathcal{G}$, rather than the cropped and extrapolated image $\mathcal{G}^*$, be used. Although this is not realistic in practical problems (as noted above), it is appropriate for the work considered in this paper because the extrapolation procedure introduces errors in $\mathcal{G}^*$. Specifically, if $\mathcal{G}^*$ is used, it may not be possible to identify the cause of the differences between the deblurred images obtained from

the Sylvester and Bézout matrices, that is, differences that arise from the cropping and extrapolation procedures, and differences that arise from the properties of the Sylvester and Bézout matrices.

The coefficients of the GCD of $f(y)$ and $g(y)$ are contained in the last non-zero row of the upper triangular matrix R of the QR decomposition of $B(f, g)$, and similarly for $S(f, g)$, and Chang and Paige [10] show that an error ϵ in a matrix $A = QR$ introduces an error in R that is much smaller than $\epsilon\kappa(A)$, where $\kappa(A)$ is the condition number of A. Examples 4 and 5 show, however, that superior results for the coefficients of the GCD are obtained by using a structure-preserving matrix method to solve (15), which is derived from the Sylvester subresultant matrix $S_d(\bar{f}, \alpha_0\bar{g})$, rather than by using the QR decomposition of the Bézout matrix.

Example 4 Figure 11 shows a separable Gaussian PSF, which is 29×29 pixels, and Fig. 12 shows an exact image and the blurred image formed by convolving the exact image with the PSF. Noise was not included in the formation of the blurred image.

The calculation of the size of the PSF using the Sylvester and Bézout matrices is described in [30,31] and it requires that the same procedure be used for both matrices. Con-

sider initially the calculation of the horizontal extent of the PSF, which requires the selection of 25 random pairs of rows of the blurred image, thus forming 25 pairs of polynomials whose coefficients are the pixel values of the chosen rows. This enables 25 GCD computations to be performed, and the length of the row component of the PSF is therefore equal to one plus the mode of the degree of the GCD from these 25 computations. This procedure is repeated for the column component of the PSF by selecting 25 random pairs of columns of the blurred image. Although this method of selecting 25 pairs of rows and 25 pairs of columns was used for the Sylvester and Bézout matrices, the computation of the degrees of the GCDs (one GCD for the row component of the PSF and one GCD for the column component of the PSF) from these 25 pairs of rows and 25 pairs of columns was performed differently for these two matrices. In particular, Theorems 1 and 3, respectively, were used to compute the rank of the Bézout matrix, and the rank of the Sylvester matrix and each of its subresultant matrices.

Consider initially the Bézout matrix, whose rank loss was computed using the SVD. This matrix was formed for each of the 25 pairs of rows, and each of the 25 pairs of columns, of the blurred image $\mathcal{G}$, and the SVD was used to calculate the rank (and therefore the rank loss) of each of these 50 matrices. By contrast, the rank loss of the Sylvester matrix and its subresultant matrices was computed using the QR decomposition, as described in Sect. 2.1 and Example 3. In particular, the degree of the GCD was computed using (12), and identical results were obtained using (13).

Figures 13 and 14 show the histograms of the results for the computation of the size of the PSF obtained from the Bézout matrix and the correct degree, 28, is achieved in 24 of the 25 trials for the row component of the PSF, and in all the trials for the column component of the PSF. Slightly better results were obtained when the Sylvester matrix was used because the correct degree was achieved in all the trials for the row and column components of the PSF.

These results from the histograms for the degrees of the GCD in the horizontal and vertical directions (and there-

Fig. 13 The computation of the degree of the row component of the PSF, in the absence of noise, using the Bézout matrix, for Example 4
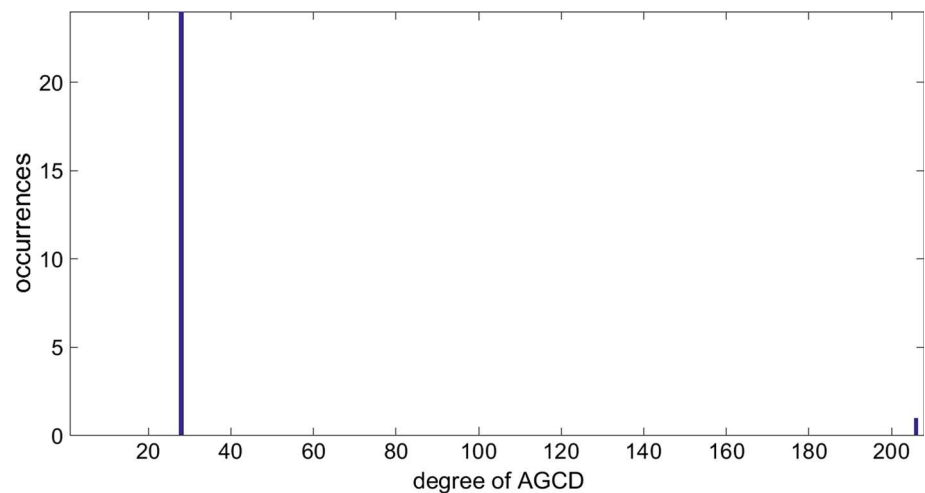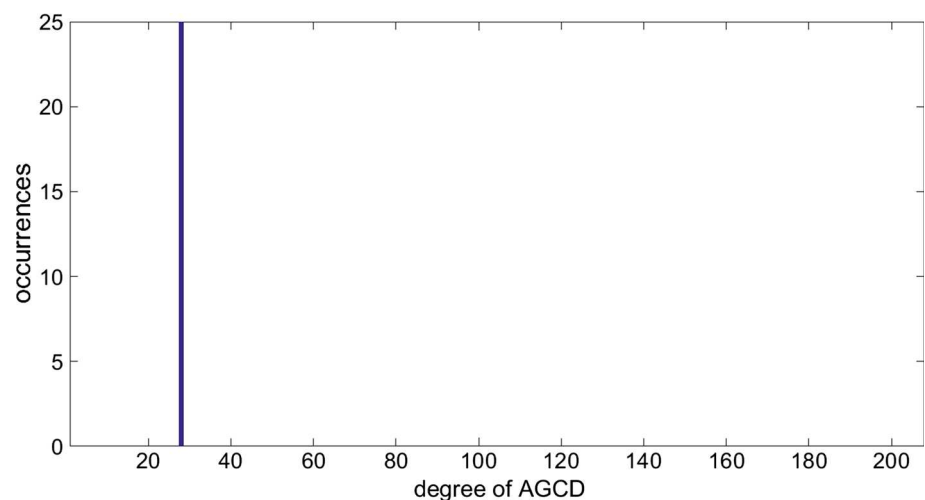

Fig. 14 The computation of the degree of the column component of the PSF, in the absence of noise, using the Bézout matrix, for Example 4

fore the size of the PSF) allow the coefficients of the PSF to be calculated, as described in Sect. 2.2 for the Sylvester matrix, or, as shown in Theorem 1, from the upper triangular matrix R of the QR decomposition of the Bézout matrix. For both methods, the deblurred image is then obtained by two deconvolution operations, one operation for the deconvolution of the row (column) component of the PSF from the given blurred image, thereby yielding a partially deblurred image $\tilde{\mathcal{F}}$, and one operation for the deconvolution of the column (row) component of the PSF from $\tilde{\mathcal{F}}$, thereby yielding a fully deblurred image.

The experiment was repeated but uncertainty $\mathcal{E}$ was included in the PSF and noise $\mathcal{N}$ was added, such that the blurred image is defined by

$$\mathcal{G} = \mathcal{F} * (\mathcal{H} + \mathcal{E}) + \mathcal{N},$$

which can be expressed in polynomial form as

$$g(x, y) = f(x, y)(h(x, y) + e(x, y)) + n(x, y).$$

If $g_{i,j}, f_{i,j}, h_{i,j}, e_{i,j}$ and $n_{i,j}$ are the coefficients of the polynomials $g(x, y), f(x, y), h(x, y), e(x, y)$ and $n(x, y)$, respectively, then the relative uncertainty in each coefficient of the PSF and the relative error due to the noise were defined by

$$0 < \frac{|e_{i,j}|}{h_{i,j}} \le \varepsilon_{i,j},$$

and

$$0 < \frac{|n_{i,j}|}{\left(f(x, y)\big(h(x, y) + e(x, y)\big)\right)_{i,j}} \le \varepsilon_{i,j},$$

respectively, where $\varepsilon_{i,j}$ is a uniformly distributed random variable in the range $\left[10^{-6}, 10^{-5}\right]$. This range of values of $\varepsilon_{i,j}$ yields a normwise relative error of the blurred image

with respect to the exact image of 0.25, which corresponds to a signal-to-noise ratio of 12 dB.

The dependence of the uncertainty $\mathcal{E}$ in the PSF and noise $\mathcal{N}$ on the pixel coordinates i and j is included for two reasons:

1. It cannot be assumed in real images that $\varepsilon_{i,j}$ is approximately constant for all values of i and j, that is, across the entire image, and thus a variation of one order of magnitude ($\left[10^{-6}, 10^{-5}\right]$) in the relative errors defines a more realistic scenario.

2. It makes the inclusion of a tolerance, which is dependent on the relative error, for the determination of the size of the PSF difficult, and it therefore provides a stringent test for the calculation of the size of the PSF from the numerical rank of a matrix.

Figures 15 and 16 show the histograms of the results obtained when the rank deficiency of the Bézout matrix was used to determine the size of the PSF. It is clear that bad results were obtained because the correct degrees (28 for both components) were not obtained in any of the 50 trials. Much better results were obtained when the Sylvester matrix and its subresultant matrices were used because, as shown in Figs. 17 and 18, the correct degrees of the row and column components of the PSF were achieved in 14 and 13 trials, respectively, and the mode of each set of results therefore returned the correct degree of each GCD, and therefore the correct size of the PSF.   □

The discussion in Sect. 2.3 and Example 4 show that the Bézout matrix cannot be used for GCD computations and image deblurring because even a modest level of noise causes unsatisfactory results to be obtained since it does not yield the correct degree of the GCD, and therefore the correct size of the PSF. Example 5 extends these examples by considering the application of this matrix to BID when neither additive noise nor uncertainty in the PSF are included in the formation of a blurred image.

Fig. 15 The computation of the degree of the row component of the PSF, in the presence of noise, using the Bézout matrix, for Example 4
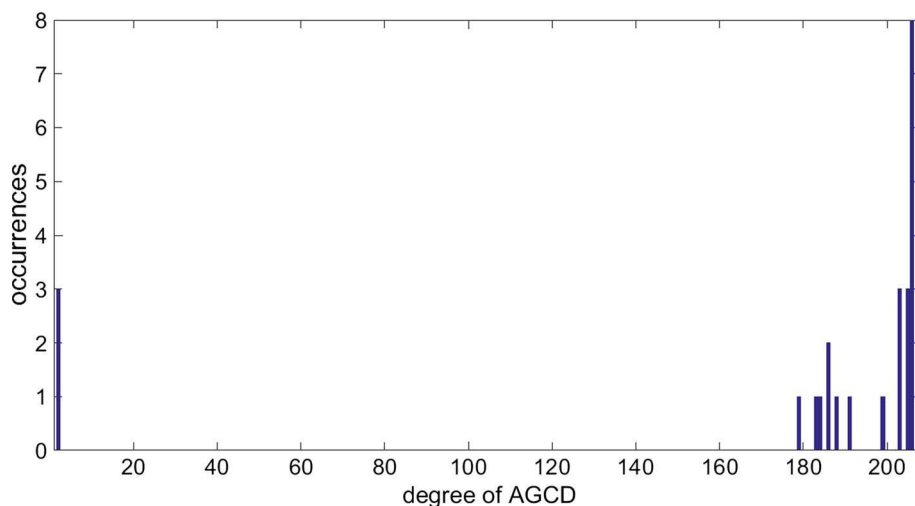
Fig. 16 The computation of the degree of the column component of the PSF, in the presence of noise, using the Bézout matrix, for Example 4
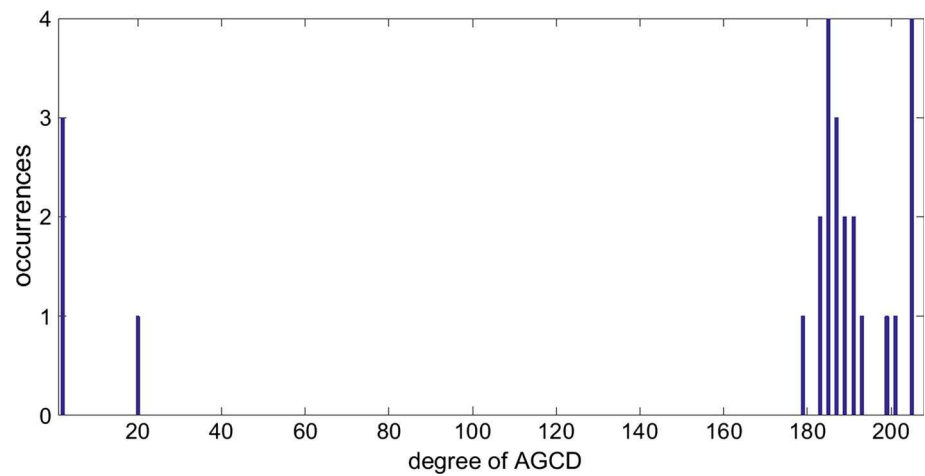

Fig. 17 The computation of the degree of the row component of the PSF in the presence of noise, using the Sylvester matrix, for Example 4
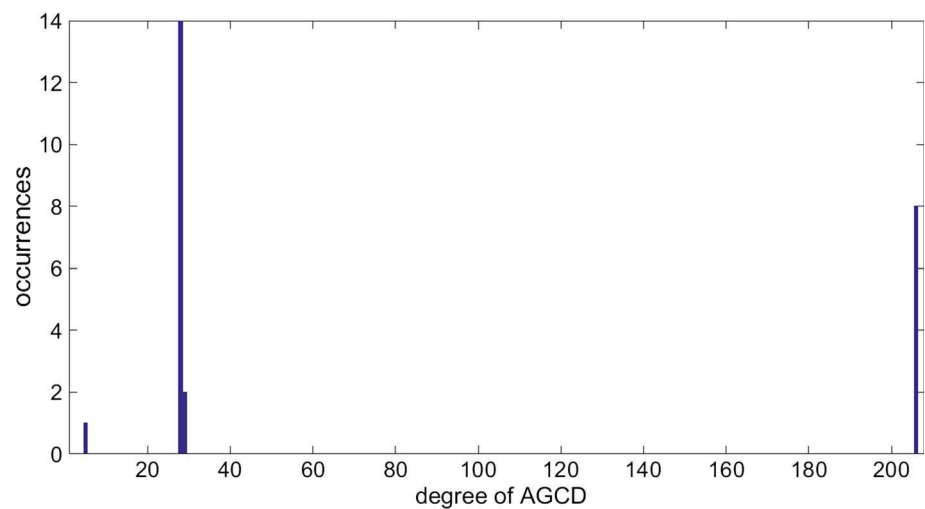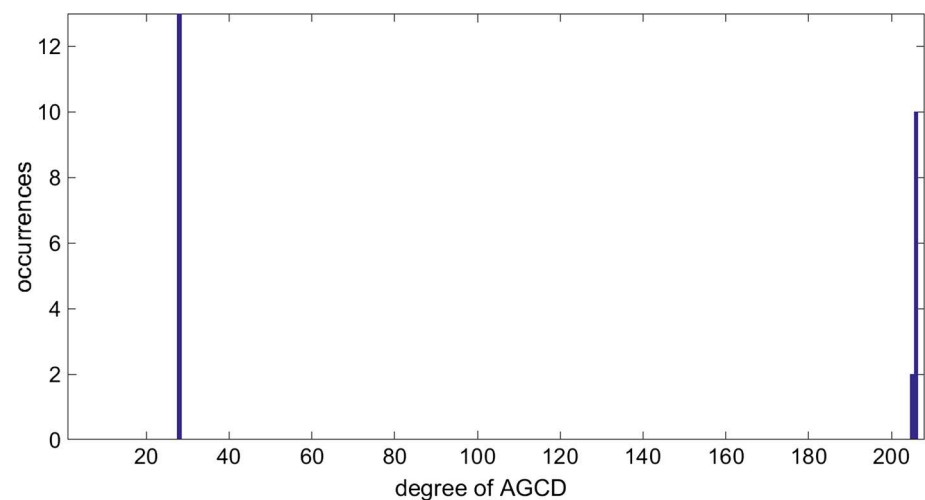

Fig. 18 The computation of the degree of the column component of the PSF in the presence of noise, using the Sylvester matrix, for Example 4

Example 5 Consider the Gaussian PSF and exact image in Figs. 11 and 12 (left), respectively. A blurred form of this image was formed by convolving it with the PSF, and neither noise nor uncertainty in the PSF were added. The Bézout matrix was used to obtain two deblurred images:

1. The size of the PSF was calculated using the method described in Example 4 and the coefficients of its column component $h_c(x)$ were computed from the last non-zero row of the upper triangular matrix R from the QR decomposition of the Bézout matrix, as stated in Theorem 1.

Fig. 19 The deblurred images using the QR decomposition of the Bézout matrix (left) and the MATLAB code in [20,21] (right), for Example 5

This procedure was repeated in order to calculate the coefficients of the row component $h_r(y)$ of the PSF. The deblurred image was then obtained by performing two deconvolutions, one deconvolution for $h_c(x)$ and one deconvolution for $h_r(y)$.

2. The MATLAB code in [20,21] was executed, for which the semi-blind deconvolution problem was solved because the size of the PSF was defined (hardcoded) and not calculated. The coefficients of the PSF in each direction were then calculated from a leading principal submatrix of $B(f, g)$, as stated in Theorem 2 and [5, Algorithm 9.1].

The deblurred images are shown in Fig. 19 and they are of poor quality because they show spurious artefacts. The relative errors in the images obtained using methods (1) and (2) are 5.64×10^{-2} and 1.04×10^{-1}, respectively. The deblurred image obtained from the Sylvester matrix is shown in Fig. 20, and it is clear that it is superior to the deblurred images in Fig. 19. Its relative error is 2.14×10^{-9}, which is many orders of magnitude smaller. □

The solution of the problem of BID requires that the size and coefficients of the PSF be computed, and the former computation is significantly harder than the latter computation because it reduces to the determination of the (numerical) rank of a matrix, which is a difficult problem. The effect on a deblurred image of a difference in the size of the PSF for blurring an image and then deblurring the blurred image, which occurs when the computed rank of the resultant matrix is incorrect, is considered in Example 6.

Example 6 The image in Fig. 21 was blurred by the separable Gaussian PSF shown in Fig. 22. The size of the PSF is 17×17 pixels, and the blurred image was deblurred by square separable Gaussian PSFs of widths 9, 11, ..., 17, ..., 23, 25 pixels using the Sylvester matrix, as described in Example 4.

Fig. 20 The deblurred image using the Sylvester matrix, for Example 5

The variation of the relative error of each deblurred image with the error in the width of the PSF is shown in Fig. 23, where the error in the width of the PSF is defined as

$$\text{error in width of PSF} = \text{width of PSF for blurring} \\ - \text{width of PSF for deblurring}.$$

The graph has a sharp minimum when the error in the width of the PSF used to deblur the blurred image is zero. If, however, the widths of the PSF used to blur the exact image and then deblur its blurred form differ, then the relative error in the deblurred image is very large. □

The result of Example 6 is consistent with the function `deconvblind.m` in MATLAB, for which the computed PSF is affected strongly by the initial estimate of its size, and less strongly by the values of the entries of its matrix form, or equivalently, by the coefficients of its polynomial representation [15].

Fig. 21 The exact image for Examples 6 and 7

The PSFs in Examples 4, 5 and 6 are separable, but, as noted above, although this restriction is appropriate for some theoretical investigations, it is not realistic because the PSF in most imaging systems is not separable. If the PSF is weakly

separable, that is, $h(x, y) \approx h_c(x)h_r(y)$, then it may be possible to approximate a non-separable PSF by a separable PSF, which would make the computation of a deblurred image simpler and faster. Example 7 considers the error in the deblurred image that results from this approximation.

Example 7 The image in Fig. 21 was blurred by a Gaussian PSF of width 15×15 pixels and whose covariance matrix was

$$R = \begin{bmatrix} \sigma_1 & \sigma_2 \\ \sigma_2 & \sigma_1 \end{bmatrix}, \qquad \sigma_1 = 2.5,$$

where the conditions $\sigma_2 = 0$ and $\sigma_2 > 0$ define a separable PSF and a non-separable PSF, respectively. Eleven blurred images, which differ in the value of σ_2, were formed,

$$\sigma_2 = \{0.000, 0.025, 0.050, 0.075, 0.100,$$
$$0.125, 0.150, 0.175, 0.200, 0.225, 0.250\},$$

and each blurred image was deblurred assuming $\sigma_2 = 0$. It follows that ten of the eleven blurred images were formed

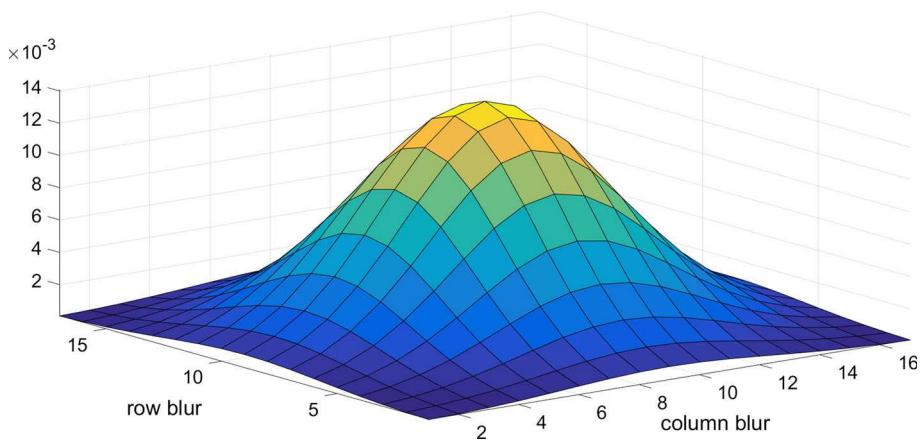**Fig. 22** The Gaussian PSF for Example 6

Fig. 23 The variation of the relative error in the deblurred image with the error in the width of the PSF used to deblur the blurred image, for Example 6
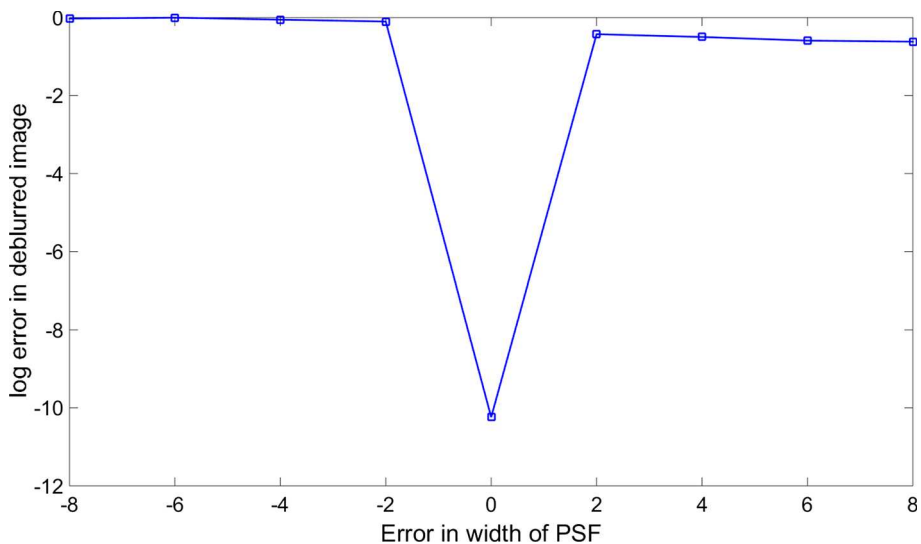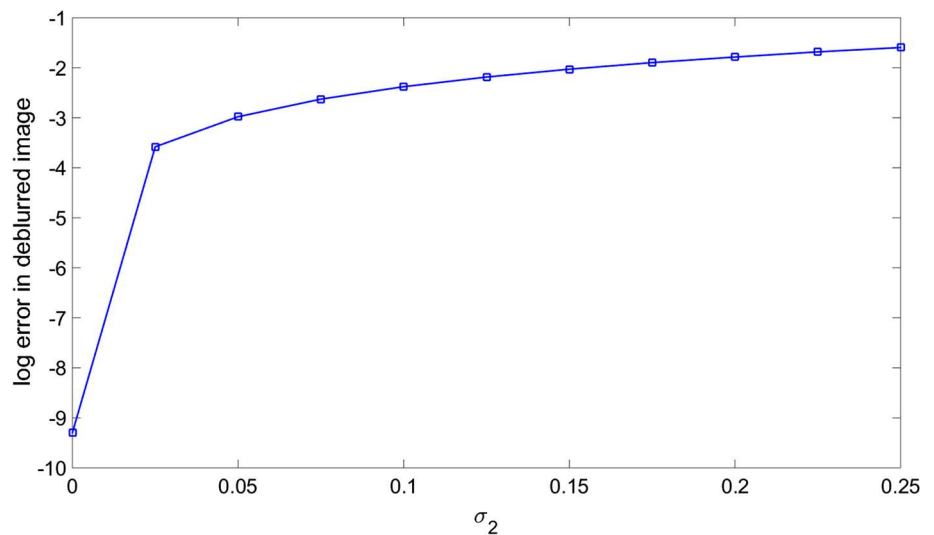
Fig. 24 The variation of the relative error in the deblurred image with σ_2, for Example 7

with a non-separable PSF ($\sigma_2 > 0$), and all these blurred images were deblurred assuming a separable PSF ($\sigma_2 = 0$). The method described in Example 4 was used to deblur each image, and thus the size of the PSF was calculated using the QR decomposition of the Sylvester matrix and histograms of the results were drawn, as shown in Figs. 17 and 18.

The effect of $\sigma_2 > 0$ is, as expected, a decrease in the number of occurrences of the mode of each histogram, that is, the correct degrees of the PSF (14 in the horizontal and vertical directions) are achieved less frequently. This causes an increase in the error in the computed PSF, and therefore an increase in the relative error in the deblurred image. This error was calculated for each deblurred image, and its variation with σ_2 is shown in Fig. 24. The minimum error occurs, as expected, at $\sigma_2 = 0$, and the error then increases, initially rapidly and then more slowly, as σ_2 increases. The relative error in the deblurred image when $\sigma_2 = 0.25$ is 2.54×10^{-2}, but this image is of high quality, as shown in Fig. 25. $\qquad \square$

Example 7 shows that a weakly non-separable PSF can be approximated by a separable PSF in the absence of noise and uncertainty in the PSF, which makes the computation of a deblurred image simpler and faster. Further investigation is, however, required to determine the validity of this approximation in the presence of noise and uncertainty in the PSF.

4 Summary

This paper has considered the Sylvester and Bézout matrices for AGCD computations and the solution of the problem of BID. It has been shown that the Bézout matrix does not yield good results because floating point cancellation may occur when its entries are computed since it is necessary to

Fig. 25 The deblurred image for $\sigma_2 = 0.25$, for Example 7

evaluate terms of the form $\sum_{i,j}(a_i b_j - a_j b_i)$ where a_i and b_j are the coefficients of the polynomials $f(y)$ and $g(y)$, respectively. This problem does not arise when the entries of the Sylvester matrix are formed because each of its non-zero entries is either the coefficient a_i or the coefficient b_j. Also, computations on the leading principal submatrices of the Bézout matrix are not recommended because they yield unstable results. Computations on the Sylvester subresultant matrices are, however, much more reliable and they can be used for AGCD computations.

Examples showed that the Bézout matrix cannot be used to compute the size of the PSF in the presence of noise and uncertainty in the PSF, which is a disadvantage of this matrix because these degradations are present in images obtained in practical problems. The Sylvester matrix yields good results because it allows high quality deblurred images to be computed, even when these degradations are present.

The improved results obtained with the Sylvester matrix justify its use, even though it is not symmetric (unlike the Bézout matrix, which is symmetric), and it is larger than the Bézout matrix.

References

1. Almeida, M.S.C., Figueiredo, M.A.T.: Frame-based image deblurring with unknown boundary conditions using the alternating direction method of multipliers. In: IEEE International Conference on Image Processing, pp. 582–585. Melbourne, Australia (2013)
2. Aricò, A., Donatelli, M., Serra-Capizzano, S.: The anti-reflective algebra: structural and computational analysis with application to image deblurring and denoising. Calcolo **45**(3), 149–175 (2008)
3. Babacan, S., Molina, R., Do, M., Katsaggelos, A.: Bayesian blind deconvolution with general sparse image priors. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part VI, Lecture Notes in Computer Science, vol. 7577, pp. 341–355. Springer, Berlin (2012)
4. Barnett, S.: Polynomials and Linear Control Systems. Marcel Dekker, New York (1983)
5. Bini, D., Pan, V.: Polynomial and Matrix Computations. Volume 1: Fundamental Algorithms. Birkhäuser, Boston (1994)
6. Bini, D.A., Gemignani, L.: Bernstein–Bezoutian matrices. Theoret. Comput. Sci. **315**, 319–333 (2004)
7. Bini, D.A., Marco, A.: Computing curve intersection by means of simultaneous iterations. Numer. Algor. **43**, 151–175 (2006)
8. Boito, P.: Structured Matrix Based Methods For Approximate Polynomial GCD. Ph.D. thesis, Department of Mathematics, The University of Pisa, Italy (2007)
9. Bourne, M., Winkler, J.R., Su, Y.: A non-linear structure-preserving matrix method for the coefficients of an approximate greatest common divisor of two Bernstein polynomials. J. Comput. Appl. Math. **320**, 221–241 (2017)
10. Chang, X., Paige, C.: A perturbation analysis for R in the QR factorization. Tech. rep., No. SOCS-95.7. School of Computer Science, McGill University, Montreal, Canada (1995)
11. Danelakis, A., Mitrouli, M., Triantafyllou, D.: Blind image deconvolution using a banded matrix method. Numer. Algor. **64**, 43–72 (2013)
12. Diaz-Toca, G., Belhaj, S.: Blind image deconvolution through Bezoutians. In: 8th International Congress on Industrial and Applied Mathematics. Beijing, China (2015)
13. Fan, Y.W., Nagy, J.G.: Synthetic boundary conditions for image deblurring. Linear Algebra Appl. **434**, 2244–2268 (2011)
14. Golub, G.H., Van Loan, C.F.: Matrix Computations. John Hopkins University Press, Baltimore (2013)
15. Gonzalez, R.C., Woods, R.E., Eddins, S.L.: Digital Image Processing Using MATLAB. Gatesmark Publishing, (2009)
16. Hansen, P.C., Nagy, J.G., O'Leary, D.P.: Deblurring Images: Matrices, Spectra, and Filtering. SIAM, Philadelphia (2006)
17. Kundur, D., Hatzinakos, D.: Blind image deconvolution. IEEE Signal Process. Mag. **13**(3), 43–64 (1996)
18. Levin, A., Weiss, Y., Durand, F., Freeman, W.: Efficient marginal likelihood optimization in blind deconvolution. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2657–2664. Colorado, USA (2011)
19. Levin, A., Weiss, Y., Durand, F., Freeman, W.: Understanding blind deconvolution algorithms. IEEE Trans. Pattern Anal. Mach. Intell. **33**(12), 2354–2367 (2011)
20. Li, F., Li, Z., Saunders, D., Yu, J.: A theory of coprime blurred pairs. In: IEEE International Conference on Computer Vision, pp. 217–224. http://fengl.org/publications/ (2011). Accessed Feb 2017
21. Li, Z., Yang, Z., Zhi, L.: Blind image deconvolution via fast approximate GCD. In: Proceeding of International Symposium Symbolic and Algebraic Computation, pp. 155–162. http://fengl.org/publications/ (2010). Accessed Feb 2017
22. Liang, B., Pillai, S.: Blind image deconvolution using a robust 2-D GCD approach. In: IEEE International Symposium on Circuits and Systems, pp. 1185–1188 (1997)
23. Perrone, D., Favaro, P.: A clearer picture of total variation blind deconvolution. IEEE Trans. Pattern Anal. Mach. Intell. **38**(6), 1041–1055 (2016)
24. Pillai, S., Liang, B.: Blind image deconvolution using a robust GCD approach. IEEE Trans. Image Process. **8**(2), 295–301 (1999)
25. Rosen, J.B., Park, H., Glick, J.: Structured total least norm for nonlinear problems. SIAM J. Mat. Anal. Appl. **20**(1), 14–30 (1998)
26. Ruiz, P., Zhou, X., Mateos, J., Molina, R., Katsaggelos, A.: Variational Bayesian blind image deconvolution: a review. Digit. Signal Proc. **47**, 116–127 (2015)
27. Serra-Capizzano, S.: A note on antireflective boundary conditions and fast deblurring models. SIAM J. Sci. Stat. Comput. **25**(4), 1307–1325 (2003)
28. Sturmfels, B.: Introduction to resultants. In: Cox, D., Sturmfels, B. (eds.) Proceedings of Symposia in Applied Mathematics, volume 53, Applications of Computational Algebraic Geometry, pp. 25–39. AMS, Rhode Island, USA (1998)
29. Winkler, J.R.: Structured matrix methods for the computation of multiple roots of a polynomial. J. Comput. Appl. Math. **272**, 449–467 (2014)
30. Winkler, J.R.: The Sylvester resultant matrix and image deblurring. In: Boissonnat, J.D., Cohen, A., Gibaru, O., Gout, C., Lyche, T., Mazure, M.L., Schumaker, L.L. (eds.) Curves and Surfaces, Lecture Notes in Computer Science, vol. 9213, pp. 461–490. Springer, Berlin (2015)
31. Winkler, J.R.: Polynomial computations for blind image deconvolution. Linear Algebra Appl. **502**, 77–103 (2016)
32. Winkler, J.R., Hasan, M.: An improved non-linear method for the computation of a structured low rank approximation of the Sylvester resultant matrix. J. Comput. Appl. Math. **237**(1), 253–268 (2013)
33. Winkler, J.R., Hasan, M., Lao, X.Y.: Two methods for the calculation of the degree of an approximate greatest common divsior of two inexact polynomials. Calcolo **49**, 241–267 (2012)
34. Winkler, J.R., Lao, X.Y., Hasan, M.: The computation of multiple roots of a polynomial. J. Comput. Appl. Math. **236**, 3478–3497 (2012)
35. Winkler, J.R., Yang, N.: Resultant matrices and the computation of the degree of an approximate greatest common divisor of two inexact Bernstein basis polynomials. Comput. Aided Geom. Des. **30**(4), 410–429 (2013)
36. Zeng, Z.: The approximate GCD of inexact polynomials. Part 1: a univariate algorithm, Preprint (2004)
37. Zhou, X., Mateos, J., Zhou, F., Molina, R., Katsaggelos, A.: Variational Dirichlet blur kernel estimation. IEEE Trans. Image Process. **24**(12), 5127–5139 (2015)

Joab R. Winkler obtained his undergraduate degree at Imperial College of Science and Technology, London, and his Ph.D. degree at University College London. He then worked in industry for a few years, before returning to academia. He is now a Reader in the Department of Computer Science at the University of Sheffield. His main research interest was in numerical problems associated with curves and surfaces in geometric modelling, which included the application of a structure-preserving matrix method to the Sylvester resultant matrix for the computation of multiple roots of a polynomial. This matrix, and more generally the computation of the greatest common divisor of two polynomials, can be used to obtain a solution of the problem of blind image deconvolution, and he has extended his research interest to this topic.

Hanan Halawani was born in Makkah, Saudi Arabia, in 1987. She received the B.Sc. degree in Computer Science from Umm Al-Qura University, Makkah, Saudi Arabia, in 2008, and the M.Sc. and Ph.D. degrees in Computer Science from the University of Sheffield in 2011 and 2017, respectively. In 2009, she joined the Department of Computer Science and Information Systems, Najran University. Her current research interests include computer vision, image processing and machine learning. She is a co-author of Fundamental of Computer Graphics (C++ and Open GL) book (in Arabic). She is a student member of the IEEE and also a member of the IEEE Signal Processing and Women in Engineering societies.