Monthly Notices
of the
ROYAL ASTRONOMICAL SOCIETY

MNRAS **457,** 542–574 (2016) doi:10.1093/mnras/stv2910

# Difference image analysis: automatic kernel design using information criteria

D. M. Bramich,[1]* Keith Horne,[2] K. A. Alsubai,[1] E. Bachelet,[1] D. Mislis[1] and N. Parley[1]

[1] *Qatar Environment and Energy Research Institute (QEERI), HBKU, Qatar Foundation, Doha, Qatar*
[2] *SUPA Physics and Astronomy, North Haugh, St Andrews, Fife KY16 9SS, UK*

Accepted 2015 December 9. Received 2015 November 22; in original form 2015 August 5

## ABSTRACT

We present a selection of methods for automatically constructing an optimal kernel model for difference image analysis which require very few external parameters to control the kernel design. Each method consists of two components; namely, a kernel design algorithm to generate a set of candidate kernel models, and a model selection criterion to select the simplest kernel model from the candidate models that provides a sufficiently good fit to the target image. We restricted our attention to the case of solving for a spatially invariant convolution kernel composed of delta basis functions, and we considered 19 different kernel solution methods including six employing kernel regularization. We tested these kernel solution methods by performing a comprehensive set of image simulations and investigating how their performance in terms of model error, fit quality, and photometric accuracy depends on the properties of the reference and target images. We find that the irregular kernel design algorithm employing unregularized delta basis functions, combined with either the Akaike or Takeuchi information criterion, is the best kernel solution method in terms of photometric accuracy. Our results are validated by tests performed on two independent sets of real data. Finally, we provide some important recommendations for software implementations of difference image analysis.

**Key words:** methods: data analysis – methods: statistical – techniques: image processing – techniques: photometric.

## 1 INTRODUCTION

In astronomy, the technique of difference image analysis (DIA) aims to measure changes, from one image to another, in the objects (e.g. stars, galaxies, etc.) observed in a particular field. Typically these changes consist of variations in flux and/or position. However, the variations in the object properties that we are interested in are entangled with the differences in the sky-to-detector (or scene-to-image) transformation between pairs of images. Therefore, the DIA method must carefully model the changes in astrometry, throughput, background, and blurring between an image pair in order to extract the required astronomical information.

The state of the art in DIA has evolved substantially over the last decade and a half. Possibly the most complicated part of DIA is the optimal modelling of the convolution kernel describing the changes in point-spread function (PSF) between images. The seminal paper by Alard & Lupton (1998) set the current framework for doing this by detailing the expansion of the kernel as a linear combination of basis functions. Alard (2000) subsequently showed how to model a spatially varying convolution kernel by modelling the coefficients of the kernel basis functions as polynomials of the image coordinates. The most important ingredient then in constructing a kernel solution in the Alard DIA framework is the definition of the set of kernel basis functions. The main developments in this area were achieved by Alard & Lupton (1998), who defined the Gaussian basis functions, Bramich (2008) and Miller, Pennypacker & White (2008) who introduced the delta basis functions (DBFs), and Becker et al. (2012, hereafter Be12) who conceived of the regularized DBFs. A detailed discussion of the kernel basis functions presented in the DIA literature may be found in Bramich et al. (2013, hereafter Br13).

The traditional Gaussian basis functions require the specification of numerous parameters while demanding precise sub-pixel image registration for optimal results, as do many other sets of kernel basis functions (e.g. the network of bicubic B-spline functions introduced by Yuan & Akerlof 2008). Consequently, the optimal choice of parameters for generating such sets of basis functions is not obvious, although some investigation into this issue has been performed (Israel, Hessman & Schuh 2007). In contrast, the DBFs have the

* E-mail: dan.bramich@hotmail.co.uk

Published by Oxford University Press on behalf of the Royal Astronomical Society

ultimate flexibility to represent a discrete kernel of any form while requiring the absolute minimal user specification; namely the kernel size and shape (or equivalently the set of 'active' kernel pixels). They may even be used to model fractional pixel offsets between images, avoiding the need for image resampling in the absence of other image misalignments (rotation, scale, shear, and distortion). Unsurprisingly then, DIA photometry for kernels employing DBFs has been shown to be better than that produced for kernels using Gaussian basis functions (Albrow et al. 2009). However, the use of DBFs yields somewhat noisier kernel solutions than is desirable due to the relatively large number of parameters in the kernel model. To tackle this weakness of the DBFs, Be12 developed the regularized DBFs through the elegant application of Tikhonov regularization to the kernel model. This refined approach produces very clean and low-noise kernel solutions at the expense of introducing an extra parameter λ into the kernel definition, where the value of λ controls the strength of the regularization. Be12 recommend values of λ between 0.1 and 1 for square kernels of size 19 × 19 pixels although they caution that the optimal value will likely be data set dependent.

The next logical step in the development of DIA is to investigate how the properties of the image pair under consideration influence the composition of the optimal kernel model (i.e. the optimal set of DBFs, the optimal values of their coefficients, and the optimal value of λ). In this context, 'optimality' refers both to the Principle of Parsimony, in that the optimal kernel model should constitute the simplest configuration of DBFs that provides a sufficiently good fit to the data, and to appropriate/relevant model performance measure(s). The proposed investigation may be accomplished both by generating and analysing a comprehensive set of simulated images, and by testing on a wide variety of real image data. Neither of these tasks have yet been attempted.

Various model selection criteria have been developed from different statistical view-points as implementations of the Principle of Parsimony (e.g. the Aikaike information criterion – Akaike 1974, the Bayesian information criterion – Schwarz 1978, etc.) and each one may be used to automatically select a parsimonious model from a set of models.[1] Due to the sheer number of possible combinations of DBFs that may constitute the kernel model, the set of models that can be considered will be limited to a set of feasible candidate kernel models defined via the adoption of an appropriate kernel design algorithm. The performance of each model selection criterion may then be assessed by measuring the quality of the corresponding kernel solution with respect to one or more desired metric(s). The final result will then be a recommendation, dependent on the properties of the image pair under consideration, as to which model selection criterion should be adopted to consistently yield the best kernel solutions for the specified kernel design algorithm.

In this paper, we report on the results of having carried out the proposed investigation for both the unregularized and regularized DBFs (Section 2) using simulated images (Section 5) and real data (Section 6). We restrict attention to the case of solving for a spatially invariant convolution kernel. The performance of three proposed kernel design algorithms (Section 4) coupled with up to eight model selection criteria (Section 3) was assessed with regards to model error (simulations only), fit quality, and photometric accuracy. In total, 19 methods were tested. The conclusions and recommendations from our investigation are detailed in Section 7.

## 2 MODELLING THE CONVOLUTION KERNEL

In this section, we briefly describe the methods used in this paper to solve for the spatially invariant convolution kernel matching the PSF between two images of the same field.

### 2.1 Solving for a spatially invariant kernel: recap

Consider a pair of registered images of the same field with the same dimensions and sampled on the same pixel grid. To avoid invalidating the assumption of a spatially invariant kernel model, the image registration should be such that at most there is a translational offset of a few pixels between the images, with no rotational (or other) image misalignments. Let the images be referred to as the reference image $R$ and the target image $I$ with pixel values $R_{ij}$ and $I_{ij}$, respectively, where $i$ and $j$ are pixel indices referring to the column $i$ and row $j$ of an image.

We model the target image $I$ as a model image $M$ formed by the convolution of the reference image $R$ with a spatially invariant discrete convolution kernel $K$ plus a spatially invariant (constant[2]) differential background $B$:

$$M_{ij} = [R \otimes K]_{ij} + B, \tag{1}$$

where the $M_{ij}$ are the pixel values of the model image. As in Alard & Lupton (1998), we model $K$ as a linear combination of basis functions:

$$K_{rs} = \sum_{q=1}^{N_\kappa} a_q \, \kappa_{qrs}, \tag{2}$$

where the $K_{rs}$ are the kernel pixel values, $r$ and $s$ are pixel indices corresponding to the column $r$ and row $s$ of the discrete kernel, $N_\kappa$ is the number of kernel basis functions, and the $\kappa_{qrs}$ are the pixel values of the $q$th discrete kernel basis function $\kappa_q$ with corresponding coefficient $a_q$. Substitution of equation (2) into equation (1) yields:

$$M_{ij} = \sum_{q=1}^{N_\kappa} a_q \, [R \otimes \kappa_q]_{ij} + B \tag{3}$$

with

$$[R \otimes \kappa_q]_{ij} = \sum_{rs} R_{(i+r)(j+s)} \kappa_{qrs}. \tag{4}$$

The image $[R \otimes \kappa_q]_{ij}$ is referred to as a *basis image*. The model image $M$ has $N_{\mathrm{par}} = N_\kappa + 1$ parameters. Note that equation (3) may be derived as a special case of equation 8 from Br13.

Assuming that the target-image pixel values $I_{ij}$ are independent observations drawn from normal (or Gaussian) distributions $\mathcal{N}(M_{ij}, \sigma_{ij})$ and that the parameters $a_q$ and $B$ of the model image have uniform Bayesian prior probability distribution functions (PDFs), then the maximum likelihood estimator (MLE) of $a_q$ and $B$ may be found by minimizing the chi-squared:

$$\chi^2 = \sum_{ij} \left( \frac{I_{ij} - M_{ij}}{\sigma_{ij}} \right)^2. \tag{5}$$

This is a general linear least-squares problem (see Press et al. 2007) with associated *normal equations* in matrix form:

$$\mathbf{H}\,\boldsymbol{\alpha} = \boldsymbol{\beta} \tag{6}$$

[1] We note that the application of a model selection criterion to model fitting may also be viewed as a regularization technique.

[2] All of the results in this paper are easily generalized to the case of a differential background that is a polynomial function of the image coordinates (e.g. see Br13).

where the symmetric and positive-definite $(N_\kappa + 1) \times (N_\kappa + 1)$ matrix $\mathbf{H}$ is the least-squares matrix, the vector $\boldsymbol{\alpha}$ is the vector of $N_\kappa + 1$ model parameters, and $\boldsymbol{\beta}$ is another vector. For the vector of parameters:

$$\alpha_q = \begin{cases} a_q & \text{for } 1 \le q \le N_\kappa \\ B & \text{for } q = N_\kappa + 1, \end{cases} \tag{7}$$

the elements of $\mathbf{H}$ and $\boldsymbol{\beta}$ are given in terms of the basis images by

$$H_{qq'} = \sum_{ij} \frac{\psi_{qij}\,\psi_{q'ij}}{\sigma_{ij}^2} \tag{8}$$

$$\beta_q = \sum_{ij} \frac{\psi_{qij}\,I_{ij}}{\sigma_{ij}^2} \tag{9}$$

$$\psi_{qij} = \frac{\partial M_{ij}}{\partial \alpha_q} = \begin{cases} [R \otimes \kappa_q]_{ij} & \text{for } 1 \le q \le N_\kappa \\ 1 & \text{for } q = N_\kappa + 1. \end{cases} \tag{10}$$

Cholesky factorization of $\mathbf{H}$, followed by forward and back substitution is the most efficient and numerically stable method (Golub & Van Loan 1996) for obtaining the solution $\boldsymbol{\alpha} = \widehat{\boldsymbol{\alpha}}$ to the normal equations (i.e. $\widehat{\boldsymbol{\alpha}}$ is the vector of MLEs of the model parameters). The inverse matrix $\mathbf{H}^{-1}$ is the covariance matrix of the parameter estimates $\mathrm{cov}\,(\widehat{\alpha}_q, \widehat{\alpha}_{q'}) = [\mathbf{H}^{-1}]_{qq'}$ and consequently the uncertainty $\sigma_q$ in each $\widehat{\alpha}_q$ is given by

$$\sigma_q = \sqrt{[\mathbf{H}^{-1}]_{qq}}. \tag{11}$$

For the spatially invariant kernel, the *photometric scale factor* $P$ between the reference and target image is a constant:

$$P = \sum_{rs} K_{rs}. \tag{12}$$

As noted by Bramich (2008), it is good practice to subtract an estimate of the sky background level from $R$ before solving for $K$ and $B$ in order to minimize any correlation between $P$ and $B$.

We adopt a noise model for the model image pixel uncertainties $\sigma_{ij}$ of

$$\sigma_{ij}^2 = \frac{\sigma_0^2}{F_{ij}^2} + \frac{M_{ij}}{G\,F_{ij}}, \tag{13}$$

where $\sigma_0$ is the CCD readout noise (ADU), $G$ is the CCD gain (e$^-$/ADU), and $F_{ij}$ is the flat-field image. The $\sigma_{ij}$ depend on the $M_{ij}$ which renders our maximum likelihood problem as a non-linear problem and also requires that the MLE of the model image parameters is obtained by minimizing $\chi^2 + \sum_{ij} \ln(\sigma_{ij}^2)$ instead of $\chi^2$. However, iterating the solution by considering the $\sigma_{ij}$ and $M_{ij}$ in turn as fixed is an appropriate linearization of the problem that still allows for the model image parameters to be determined by minimizing $\chi^2$ at each iteration as described above (since the $\sigma_{ij}$ are considered as constant whenever the model image parameters are being estimated). For the first iteration, we estimate the $\sigma_{ij}$ by approximating $M_{ij}$ in equation (13) with $I_{ij}$. A $k$-sigma-clip algorithm is employed at the end of each iteration except for the first to prevent outlier target-image pixel values from influencing the solution (e.g. cosmic rays, variable stars, etc.). The criterion for pixel rejection is $|\varepsilon_{ij}| = |(I_{ij} - M_{ij})/\sigma_{ij}| \ge k$, and we use $k = 4$. Only 3–4 iterations are required for convergence and the final solution is highly insensitive to the initial choice of $\sigma_{ij}$ (e.g. setting all of the $\sigma_{ij}$ to unity for the first iteration gives exactly the same result as setting the $\sigma_{ij}$ by approximating $M_{ij}$ in equation 13 with $I_{ij}$). Finally, it should be noted that lack of iteration introduces a bias into the kernel and differential background solution (see Br13 for a discussion and examples).

The difference image $D$ is defined by

$$D_{ij} = I_{ij} - M_{ij} \tag{14}$$

from which we may define a normalized difference image

$$\varepsilon_{ij} = D_{ij}/\sigma_{ij}. \tag{15}$$

In the absence of varying objects, and for a reliable noise model, the distribution of the $\varepsilon_{ij}$ values provides an indication of the quality of the difference image; namely, the $\varepsilon_{ij}$ should follow a Gaussian distribution with zero mean and unit standard deviation. If the $\varepsilon_{ij}$ follow a Gaussian distribution with significant bias or standard deviation greater than unity, then systematic errors are indicated, which may be due to underfitting. If they follow a Gaussian distribution with standard deviation less than unity, then overfitting may be indicated. If they follow a non-Gaussian distribution, then an inappropriate noise model may be at least part of the cause.

### 2.2 The DBFs

The final ingredient required to construct a kernel solution is the definition of the set of kernel basis functions, which in turn defines the set of basis images. In this paper, we consider only the *DBFs*, which are defined by

$$\kappa_{qrs} = \delta_{r\mu}\,\delta_{s\nu}, \tag{16}$$

where a one-to-one correspondence $q \leftrightarrow (\mu, \nu)$ associates the $q$th kernel basis function $\kappa_q$ with the discrete kernel pixel coordinates $(\mu, \nu)$, and $\delta_{ij}$ is the Kronecker delta function:

$$\delta_{ij} = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \ne j. \end{cases} \tag{17}$$

As such, each DBF $\kappa_q$ and its corresponding coefficient $a_q$ represent a single kernel pixel and its value, respectively. Note that this definition of the DBFs ignores the transformation that is required when the photometric scale factor is spatially varying (Br13).

The DBFs have a conveniently simple expression for the corresponding basis images:

$$[R \otimes \kappa_q]_{ij} = R_{(i+\mu)(j+\nu)}. \tag{18}$$

### 2.3 Regularizing the DBFs

For the DBFs, Be12 introduced a refinement to the normal equations to control the trade-off between noise and resolution in the kernel solution. They used *Tikhonov regularization* (see Press et al. 2007) to penalize kernel solutions that are too noisy by adding a penalty term to the chi-squared that is derived from the second derivative of the kernel surface and whose strength is parametrized by a tuning parameter $\lambda$. The addition of a penalty term to the chi-squared is equivalent to adopting a non-uniform Bayesian prior PDF on the model parameters. The corresponding maximum penalized likelihood estimator (MPLE) of $a_q$ and $B$ is obtained by minimizing:

$$\chi^2 + \lambda N_{\mathrm{dat}}\,\boldsymbol{\alpha}^{\mathrm{T}}\,\mathbf{L}^{\mathrm{T}}\,\mathbf{L}\,\boldsymbol{\alpha} = \sum_{ij} \left( \frac{I_{ij} - M_{ij}}{\sigma_{ij}} \right)^2 + \lambda N_{\mathrm{dat}} \sum_{q=1}^{N_\kappa} \sum_{u=1}^{N_\kappa} \sum_{v=1}^{N_\kappa} a_q\,L_{uq}\,L_{uv}\,a_v, \tag{19}$$

where $N_{\mathrm{dat}}$ is the number of data values[3] (i.e. target-image pixels) and $\mathbf{L}$ is an $(N_\kappa + 1) \times (N_\kappa + 1)$ matrix with elements:

$$L_{uv} = \begin{cases} N_{\mathrm{adj},u} & \text{for } v = u \le N_\kappa \text{, and where } N_{\mathrm{adj},u} \text{ is the number of DBFs adjacent to the DBF corresponding to } u, \\ -1 & \text{for } u \le N_\kappa,\ v \le N_\kappa,\ v \ne u \text{, and } u \text{ and } v \text{ corresponding to adjacent DBFs,} \\ 0 & \text{otherwise.} \end{cases} \quad (20)$$

We consider two DBFs to be *adjacent* if they share a common kernel-pixel edge, *connected* if they can be linked via any number of pairs of adjacent DBFs, and *disconnected* if they are not connected. Note that the elements of the last row and column of $\mathbf{L}$, corresponding to the differential background parameter $B$, are all zero.

The matrix $\mathbf{L}$ is the *Laplacian matrix* representing the connectivity graph of the set of DBFs (cf. graph theory). It is symmetric, diagonally dominant, and positive-semidefinite. All of the eigenvalues of $\mathbf{L}$ are non-negative, while $N_{\mathrm{grp}} + 1$ of them are equal to zero. Here, $N_{\mathrm{grp}}$ is the number of disconnected sets of connected DBFs within the full set of DBFs (i.e. the number of components of the connectivity graph). Consequently, the rank of $\mathbf{L}$ is $N_\kappa - N_{\mathrm{grp}}$, as is the rank of $\mathbf{L}^{\mathrm{T}}\,\mathbf{L} = \mathbf{L}\mathbf{L}$, which are facts that we will use later in Section 3.4. It is also useful to note that if all of the DBFs are connected to each other, then $\mathbf{L}$ and $\mathbf{L}\mathbf{L}$ are both of rank $N_\kappa - 1$. In Appendix A, we present a couple of example kernels with their corresponding $\mathbf{L}$ matrices.

The expression in equation (19) is at a minimum when its gradient with respect to each of the parameters $a_q$ and $B$ is equal to zero. Performing the $N_\kappa + 1$ differentiations and rewriting the set of linear equations in matrix form we obtain the *regularized normal equations*:

$$\mathbf{H}_{\mathrm{P}}\,\boldsymbol{\alpha} = \boldsymbol{\beta} \quad (21)$$

where

$$\mathbf{H}_{\mathrm{P}} = \mathbf{H} + \lambda N_{\mathrm{dat}}\,\mathbf{L}\mathbf{L}. \quad (22)$$

Obtaining the solution to the regularized normal equations now proceeds as for the normal equations in Section 2.1. The covariance matrix of the parameter estimates $\boldsymbol{\alpha} = \widehat{\boldsymbol{\alpha}}_{\mathrm{P}}$ is similarly given by $\mathrm{cov}\,(\widehat{\alpha}_{\mathrm{P},q}, \widehat{\alpha}_{\mathrm{P},q'}) = [\mathbf{H}_{\mathrm{P}}^{-1}]_{qq'}$.

## 3 MODEL SELECTION CRITERIA

Here, we describe our statistical tool-kit of model selection criteria that we will use for deciding on the best set of DBFs to be employed in the modelling of the convolution kernel. The criteria are valid for linear models, such as our model image $M$ in equation (3), and for data drawn from independent Gaussian distributions, which is a valid approximation to the Poissonian statistics of photon detection for CCD image data $I$ that only breaks down at very low signal levels ($\lesssim 16\ \mathrm{e}^-$). We direct the reader to Konishi & Kitagawa (2008) for an essential reference on the information criteria presented below.

### 3.1 Hypothesis testing for nested models

#### *3.1.1 $\Delta\chi^2$-test*

The $\Delta\chi^2$-test may be used to compare two models A and B with parameter sets $P_{\mathrm{A}}$ and $P_{\mathrm{B}}$, respectively, that are nested (i.e. $P_{\mathrm{A}} \subset P_{\mathrm{B}}$). The $\Delta\chi^2$-statistic is defined by

$$\Delta\chi^2 = \chi^2_{\mathrm{A}} - \chi^2_{\mathrm{B}}, \quad (23)$$

where $\chi^2_{\mathrm{A}}$ and $\chi^2_{\mathrm{B}}$ are the chi-squared values of models A and B, respectively (see equation 5). Under the null hypothesis that model B does not provide a significantly better fit than model A, the $\Delta\chi^2$-statistic follows a chi-squared distribution with $N_{\mathrm{par,\,B}} - N_{\mathrm{par,\,A}}$ degrees of freedom (DoF). We set our $\Delta\chi^2$ threshold for rejection of the null hypothesis at 1 per cent (e.g. $\Delta\chi^2 \gtrsim 6.63$ for DoF $= 1$). We adopt the chi-squared values of models A and B as those calculated during the first iteration of our kernel solution procedure to enable a fair comparison between models since they are both computed using the same pixel uncertainties (i.e. the $\sigma_{ij}$ estimated by approximating $M_{ij}$ in equation 13 with $I_{ij}$). However, the values of the model image parameters are still taken as those calculated in the final iteration of the kernel solution procedure.

Model selection using the $\Delta\chi^2$-test applies only to models A, B,..., Z with sequentially nested parameter sets $P_{\mathrm{A}} \subset P_{\mathrm{B}} \subset \ldots \subset P_{\mathrm{Z}}$. Starting with models A and B, the $\chi^2$ is minimized for each model and the $\Delta\chi^2$-test is used to determine whether or not model B provides a significantly better fit than model A. If it does not, then model A is accepted as the correct model and the procedure terminates, otherwise the next pair of models B and C are evaluated using the same method. The procedure continues by evaluating sequential model pairs in this fashion until either the $\Delta\chi^2$-test indicates that the next model does not provide a significantly better fit or until there are no more models to test.

#### *3.1.2* F*-test*

The $F$-test may also be used to compare two nested models A and B. The $F$-statistic is defined by

$$F = \frac{\Delta\chi^2/\left(N_{\mathrm{par,B}} - N_{\mathrm{par,A}}\right)}{\chi^2_{\mathrm{B}}/\left(N_{\mathrm{dat}} - N_{\mathrm{par,B}}\right)}, \quad (24)$$

where $N_{\mathrm{dat}}$ is the number of data values. Again, under the null hypothesis that model B does not provide a significantly better fit than model A, $F$ follows an $F$-distribution with DoF $= (N_{\mathrm{par,\,B}} - N_{\mathrm{par,\,A}}, N_{\mathrm{dat}} - N_{\mathrm{par,\,B}})$. We set our $F$ threshold for rejection of the null hypothesis at 1 per cent (e.g. $F \gtrsim 4.63$ for DoF $=$ (2,1000)) and we compute the $F$-statistic using the chi-squared values of models A and B calculated during the first iteration of our kernel solution procedure. Model selection with the $F$-test applies to models A, B,..., Z with sequentially nested parameter sets and proceeds in the same way as model selection with the $\Delta\chi^2$-test.

### 3.2 Information criteria for maximum likelihood

The principal of maximum likelihood assumes a uniform prior PDF on the model parameters. A consequence of this is that as parameters are added to a model, the maximum likelihood always increases, rendering it useless for the purpose of model selection between models with different dimensionality. Information criteria are used as an alternative for evaluating models with different numbers of

[3] Be12 accidentally omitted $N_{\mathrm{dat}}$ from their equation 12.

parameters. They may be applied regardless of whether the models under consideration are nested or non-nested.

### *3.2.1 AIC$_{\rm C}$*

The Akaike information criterion (AIC; Akaike 1974) is derived as an asymptotic approximation to the Kullback–Leibler divergence (Kullback & Leibler 1951),[4] which measures the distance of a candidate model from the true underlying model under the assumption that the true model is of infinite dimension and is therefore not represented in the set of candidate models. The aim of the AIC is to evaluate models based on their prediction accuracy.

A version of the AIC for Gaussian linear regression problems that corrects for the small-sample bias, while being asymptotically the same as the AIC for $N_{\rm dat} \gg N_{\rm par}$ was derived by Sugiura (1978):

$$\mathrm{AIC_C} = -2\ln\mathcal{L}(\widehat{\boldsymbol{\theta}}) + 2N_{\rm par}\left(\frac{N_{\rm dat}}{N_{\rm dat} - N_{\rm par} - 1}\right), \tag{25}$$

where $\mathcal{L}(\boldsymbol{\theta})$ is the likelihood function for the vector of model parameters $\boldsymbol{\theta}$, and $\widehat{\boldsymbol{\theta}}$ is a vector of MLEs for the model parameters. Model selection with the AIC$_{\rm C}$ is performed by minimizing $-2\ln\mathcal{L}(\boldsymbol{\theta})$ for each model, and then minimizing AIC$_{\rm C}$ over the full set of models under consideration.

### *3.2.2 TIC*

The Takeuchi information criterion (TIC; Takeuchi 1976) is a generalization of the AIC (Konishi & Kitagawa 2008) given by

$$\mathrm{TIC} = -2\ln\mathcal{L}(\widehat{\boldsymbol{\theta}}) + 2\,\mathrm{tr}\left(\mathbf{I}(\widehat{\boldsymbol{\theta}})\;\mathbf{J}^{-1}(\widehat{\boldsymbol{\theta}})\right), \tag{26}$$

where tr is the matrix trace operator. The matrices $\mathbf{I}$ and $\mathbf{J}$ are defined as

$$\mathbf{I}(\boldsymbol{\theta}) = \frac{1}{N_{\rm dat}}\sum_{i=1}^{N_{\rm dat}} \frac{\partial \ln l_i(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\;\frac{\partial \ln l_i(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}^{\rm T}} \tag{27}$$

$$\mathbf{J}(\boldsymbol{\theta}) = -\left(\frac{1}{N_{\rm dat}}\right)\frac{\partial^2 \ln\mathcal{L}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}\;\partial \boldsymbol{\theta}^{\rm T}} \tag{28}$$

$$\ln\mathcal{L}(\boldsymbol{\theta}) = \sum_{i=1}^{N_{\rm dat}} \ln l_i(\boldsymbol{\theta}), \tag{29}$$

where $l_i(\boldsymbol{\theta})$ is the likelihood function for the $i$th (single) data point. Model selection with the TIC proceeds as for the AIC$_{\rm C}$.

### *3.2.3 BIC*

The Bayesian approach to model selection is to choose the model with the largest Bayesian posterior probability. By approximating the posterior probability of each model, Schwarz (1978) derived the Bayesian information criterion (BIC) for model selection:

$$\mathrm{BIC} = -2\ln\mathcal{L}(\widehat{\boldsymbol{\theta}}) + N_{\rm par}\ln N_{\rm dat} - N_{\rm par}\ln 2\pi. \tag{30}$$

The BIC generally includes a heavier penalty than the AIC$_{\rm C}$ for more complicated models (e.g. in the regime $N_{\rm par} < 20$ and $N_{\rm dat} > 100$), therefore favouring models with fewer parameters than those favoured by the AIC$_{\rm C}$. Model selection with the BIC proceeds as for the AIC$_{\rm C}$.

### *3.2.4 BIC$_{\rm I}$*

Konishi, Ando & Imoto (2004) performed a deeper Bayesian analysis to derive an improved BIC:

$$\begin{aligned}\mathrm{BIC_I} = {} & -2\ln\mathcal{L}(\widehat{\boldsymbol{\theta}}) + N_{\rm par}\ln N_{\rm dat} \\ & + \ln(\det(\mathbf{J}(\widehat{\boldsymbol{\theta}}))) - N_{\rm par}\ln 2\pi.\end{aligned} \tag{31}$$

Model selection with the BIC$_{\rm I}$ proceeds as for the AIC$_{\rm C}$.

It is worth mentioning that the BIC and BIC$_{\rm I}$ are *consistent* model selection criteria in that they select with high probability the true model from the set of candidate models whenever the true model is represented in the set of candidate models.

## 3.3 Information criteria for maximum penalized likelihood

The AIC, AIC$_{\rm C}$, TIC, BIC, and BIC$_{\rm I}$ apply only to models estimated by maximum likelihood.

### *3.3.1 GIC$_{\rm P}$*

Konishi & Kitagawa (1996) derived a further generalization of the AIC and TIC, called the generalized information criterion (GIC), that can be applied to model selection for models with parameters estimated by maximum penalized likelihood:

$$\mathrm{GIC_P}(\lambda) = -2\ln\mathcal{L}(\widehat{\boldsymbol{\theta}}_{\rm P}) + 2\,\mathrm{tr}\left(\mathbf{I}_{\rm P}(\widehat{\boldsymbol{\theta}}_{\rm P})\;\mathbf{J}_{\rm P}^{-1}(\widehat{\boldsymbol{\theta}}_{\rm P})\right), \tag{32}$$

where $\widehat{\boldsymbol{\theta}}_{\rm P}$ is a vector of MPLEs for the model parameters, and

$$\mathbf{I}_{\rm P}(\boldsymbol{\theta}) = \mathbf{I}(\boldsymbol{\theta}) - \frac{\lambda}{N_{\rm dat}}\mathbf{L}^{\rm T}\,\mathbf{L}\,\boldsymbol{\theta}\,\frac{\partial \ln\mathcal{L}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}^{\rm T}} \tag{33}$$

$$\mathbf{J}_{\rm P}(\boldsymbol{\theta}) = \mathbf{J}(\boldsymbol{\theta}) + \lambda\mathbf{L}^{\rm T}\,\mathbf{L}. \tag{34}$$

Here $\mathbf{L}^{\rm T}\,\mathbf{L}$ is an $N_{\rm par} \times N_{\rm par}$ matrix and we have used the fact that it is symmetric to slightly simplify the Konishi & Kitagawa (1996) expressions for $\mathbf{I}_{\rm P}(\boldsymbol{\theta})$ and $\mathbf{J}_{\rm P}(\boldsymbol{\theta})$ (their equation 21). Model selection with the GIC$_{\rm P}(\lambda)$ is performed by minimizing GIC$_{\rm P}(\lambda)$ over $\lambda$ for each model, and then selecting the model for which GIC$_{\rm P}(\lambda)$ is minimized over the full set of models under consideration.

### *3.3.2 BIC$_{\rm P}$*

Using the same Bayesian analysis as for the derivation of the BIC$_{\rm I}$, Konishi et al. (2004) also extended the BIC$_{\rm I}$ to apply to model selection for models with parameters estimated by maximum penalized likelihood. For $\mathbf{L}^{\rm T}\,\mathbf{L}$ of rank $N_{\rm par} - d$, and denoting the product of the $N_{\rm par} - d$ non-zero eigenvalues of $\mathbf{L}^{\rm T}\,\mathbf{L}$ by $\Lambda_+$, they derived:

$$\begin{aligned}\mathrm{BIC_P}(\lambda) = {} & -2\ln\mathcal{L}(\widehat{\boldsymbol{\theta}}_{\rm P}) + d\ln N_{\rm dat} + \ln(\det(\mathbf{J}_{\rm P}(\widehat{\boldsymbol{\theta}}_{\rm P}))) \\ & - d\ln 2\pi + \lambda N_{\rm dat}\,\widehat{\boldsymbol{\theta}}_{\rm P}^{\rm T}\,\mathbf{L}^{\rm T}\,\mathbf{L}\,\widehat{\boldsymbol{\theta}}_{\rm P} - \ln\Lambda_+ \\ & - (N_{\rm par} - d)\ln\lambda.\end{aligned} \tag{35}$$

Model selection with the BIC$_{\rm P}(\lambda)$ proceeds as for the GIC$_{\rm P}(\lambda)$.

[4] Use of the AIC as a model selection criterion is also equivalent to assuming a prior PDF on the model parameters that is proportional to $\exp(-N_{\rm par})$, hence favouring models with smaller numbers of parameters.

### 3.4 Information criteria for DIA

We may adapt the various information criteria from Sections 3.2 and 3.3 to our problem of solving for the kernel and differential background in DIA. The model image $M$ has $N_{\mathrm{par}} = N_\kappa + 1$ parameters and we use the notation $\boldsymbol{\theta} \equiv \boldsymbol{\alpha}$, $\widehat{\boldsymbol{\theta}} \equiv \widehat{\boldsymbol{\alpha}}$ and $\widehat{\boldsymbol{\theta}}_{\mathrm{P}} \equiv \widehat{\boldsymbol{\alpha}}_{\mathrm{P}}$.

First, we compute the log-likelihood function for data drawn from Gaussian distributions $\mathcal{N}(M_{ij}, \sigma_{ij})$ as

$$-2 \ln \mathcal{L}(\boldsymbol{\alpha}) = \chi^2 + \sum_{ij} \ln\left(\sigma_{ij}^2\right) + N_{\mathrm{dat}} \ln 2\pi. \tag{36}$$

For model selection purposes, the last term $N_{\mathrm{dat}} \ln 2\pi$ is constant and can be ignored. Secondly, we note that since the $\sigma_{ij}$ are considered as constant at each iteration of the maximum likelihood problem in Section 2.1, the matrices $\mathbf{I}(\boldsymbol{\alpha})$ and $\mathbf{J}(\boldsymbol{\alpha})$ evaluated at $\boldsymbol{\alpha} = \widehat{\boldsymbol{\alpha}}$ are given by

$$[\mathbf{I}(\widehat{\boldsymbol{\alpha}})]_{qq'} = \frac{1}{N_{\mathrm{dat}}} \sum_{ij} \varepsilon_{ij}^2 \, \frac{\psi_{qij} \, \psi_{q'ij}}{\sigma_{ij}^2} \tag{37}$$

$$\mathbf{J}(\widehat{\boldsymbol{\alpha}}) = \mathbf{H}/N_{\mathrm{dat}}. \tag{38}$$

For computational purposes it is useful to note that $\mathbf{I}(\widehat{\boldsymbol{\alpha}})$ is symmetric. From these two expressions, we may derive the following results:

$$\mathrm{tr}\left(\mathbf{I}(\widehat{\boldsymbol{\alpha}}) \; \mathbf{J}^{-1}(\widehat{\boldsymbol{\alpha}})\right) = N_{\mathrm{dat}} \sum_{q=1}^{N_\kappa+1} \sum_{q'=1}^{N_\kappa+1} [\mathbf{I}(\widehat{\boldsymbol{\alpha}})]_{qq'} \, [\mathbf{H}^{-1}]_{qq'} \tag{39}$$

$$\ln(\det(\mathbf{J}(\widehat{\boldsymbol{\alpha}}))) = \ln(\det(\mathbf{H})) - (N_\kappa + 1) \ln N_{\mathrm{dat}}. \tag{40}$$

Finally, we consider that the solution of the normal equations requires the computation of the Cholesky factorization $\mathbf{H} = \mathbf{G}\mathbf{G}^{\mathrm{T}}$, where $\mathbf{G}$ is a lower triangular matrix with positive diagonal entries $g_{qq}$, from which we may immediately calculate the determinant of $\mathbf{H}$ as $\det(\mathbf{H}) = \prod_{q=1}^{N_\kappa+1} g_{qq}^2$. Hence, with minimal extra computation, the Cholesky factorization of $\mathbf{H}$ yields

$$\ln(\det(\mathbf{H})) = 2 \sum_{q=1}^{N_\kappa+1} \ln g_{qq}. \tag{41}$$

Therefore, using equations (36) and (39)–(41) for the maximum likelihood problem in Section 2.1, we have the following formulae for the relevant information criteria from Section 3.2:

$$\mathrm{AIC_C} = \chi^2 + \sum_{ij} \ln\left(\sigma_{ij}^2\right) + 2(N_\kappa + 1)\left(\frac{N_{\mathrm{dat}}}{N_{\mathrm{dat}} - N_\kappa - 2}\right) \tag{42}$$

$$\mathrm{TIC} = \chi^2 + \sum_{ij} \ln\left(\sigma_{ij}^2\right) + 2N_{\mathrm{dat}} \sum_{q=1}^{N_\kappa+1} \sum_{q'=1}^{N_\kappa+1} [\mathbf{I}(\widehat{\boldsymbol{\alpha}})]_{qq'} \, [\mathbf{H}^{-1}]_{qq'} \tag{43}$$

$$\mathrm{BIC} = \chi^2 + \sum_{ij} \ln\left(\sigma_{ij}^2\right) + (N_\kappa + 1)(\ln N_{\mathrm{dat}} - \ln 2\pi) \tag{44}$$

$$\mathrm{BIC_I} = \chi^2 + \sum_{ij} \ln\left(\sigma_{ij}^2\right) + 2 \sum_{q=1}^{N_\kappa+1} \ln g_{qq} - (N_\kappa + 1) \ln 2\pi. \tag{45}$$

Considering now the maximum penalized likelihood problem, for constant $\sigma_{ij}$ we have $\partial \ln \mathcal{L}(\boldsymbol{\alpha})/\partial \boldsymbol{\alpha}^{\mathrm{T}} = \boldsymbol{\beta}^{\mathrm{T}} - \boldsymbol{\alpha}^{\mathrm{T}}\mathbf{H}$, which is equal to $\lambda N_{\mathrm{dat}} \, (\mathbf{L}\mathbf{L}\widehat{\boldsymbol{\alpha}}_{\mathrm{P}})^{\mathrm{T}}$ when evaluated at $\boldsymbol{\alpha} = \widehat{\boldsymbol{\alpha}}_{\mathrm{P}}$ (using equations 21 and 22). Then, using $\mathbf{L}^{\mathrm{T}}\,\mathbf{L} = \mathbf{L}\mathbf{L}$, the matrices $\mathbf{I}_{\mathrm{P}}(\boldsymbol{\alpha})$ and $\mathbf{J}_{\mathrm{P}}(\boldsymbol{\alpha})$ evaluated at $\boldsymbol{\alpha} = \widehat{\boldsymbol{\alpha}}_{\mathrm{P}}$ are given by

$$\mathbf{I}_{\mathrm{P}}(\widehat{\boldsymbol{\alpha}}_{\mathrm{P}}) = \mathbf{I}(\widehat{\boldsymbol{\alpha}}_{\mathrm{P}}) - \lambda^2 \, (\mathbf{L}\,\mathbf{L}\,\widehat{\boldsymbol{\alpha}}_{\mathrm{P}})(\mathbf{L}\,\mathbf{L}\,\widehat{\boldsymbol{\alpha}}_{\mathrm{P}})^{\mathrm{T}} \tag{46}$$

$$\mathbf{J}_{\mathrm{P}}(\widehat{\boldsymbol{\alpha}}_{\mathrm{P}}) = \mathbf{H}_{\mathrm{P}}/N_{\mathrm{dat}}. \tag{47}$$

Writing $\Omega_q = \sum_{u=1}^{N_\kappa} \sum_{v=1}^{N_\kappa} L_{qu} \, L_{uv} \, \widehat{\alpha}_{\mathrm{P},v}$, then, from these two expressions, we may derive the following results:

$$\begin{aligned} \mathrm{tr}\left(\mathbf{I}_{\mathrm{P}}(\widehat{\boldsymbol{\alpha}}_{\mathrm{P}}) \; \mathbf{J}_{\mathrm{P}}^{-1}(\widehat{\boldsymbol{\alpha}}_{\mathrm{P}})\right) = \; & N_{\mathrm{dat}} \sum_{q=1}^{N_\kappa+1} \sum_{q'=1}^{N_\kappa+1} \\ & \times \left([\mathbf{I}(\widehat{\boldsymbol{\alpha}}_{\mathrm{P}})]_{qq'} - \lambda^2 \, \Omega_q \, \Omega_{q'}\right) \, [\mathbf{H}_{\mathrm{P}}^{-1}]_{qq'} \end{aligned} \tag{48}$$

$$\ln(\det(\mathbf{J}_{\mathrm{P}}(\widehat{\boldsymbol{\alpha}}_{\mathrm{P}}))) = \ln(\det(\mathbf{H}_{\mathrm{P}})) - (N_\kappa + 1) \ln N_{\mathrm{dat}} \tag{49}$$

Also, the Cholesky factorization of $\mathbf{H}_{\mathrm{P}} = \mathbf{G}_{\mathrm{P}}\mathbf{G}_{\mathrm{P}}^{\mathrm{T}}$ yields:

$$\ln(\det(\mathbf{H}_{\mathrm{P}})) = 2 \sum_{q=1}^{N_\kappa+1} \ln g_{\mathrm{P},qq}. \tag{50}$$

Finally, we note that the matrix $\mathbf{L}\mathbf{L}$ is of rank $N_{\mathrm{par}} - d = N_\kappa - N_{\mathrm{grp}}$, and hence $d = N_{\mathrm{grp}} + 1$.

Therefore, using equations (36) and (48)–(50) for the maximum penalized likelihood problem in Section 2.3, we have the following formulae for the relevant information criteria from Section 3.3:

$$\begin{aligned} \mathrm{GIC_P}(\lambda) = \; & \chi^2 + \sum_{ij} \ln\left(\sigma_{ij}^2\right) + 2N_{\mathrm{dat}} \sum_{q=1}^{N_\kappa+1} \sum_{q'=1}^{N_\kappa+1} \\ & \times \left([\mathbf{I}(\widehat{\boldsymbol{\alpha}}_{\mathrm{P}})]_{qq'} - \lambda^2 \, \Omega_q \, \Omega_{q'}\right) \, [\mathbf{H}_{\mathrm{P}}^{-1}]_{qq'} \end{aligned} \tag{51}$$

$$\begin{aligned} \mathrm{BIC_P}(\lambda) = \; & \chi^2 + \sum_{ij} \ln\left(\sigma_{ij}^2\right) + 2 \sum_{q=1}^{N_\kappa+1} \ln g_{\mathrm{P},qq} \\ & - (N_\kappa - N_{\mathrm{grp}}) \ln \lambda N_{\mathrm{dat}} - (N_{\mathrm{grp}} + 1) \ln 2\pi \\ & + \lambda N_{\mathrm{dat}} \sum_{q=1}^{N_\kappa} \widehat{\alpha}_{\mathrm{P},q} \, \Omega_q - \ln \Lambda_+. \end{aligned} \tag{52}$$

## 4 KERNEL DESIGN ALGORITHMS

Let us introduce the concept of a *kernel design*, which we define as a specific choice of DBFs (or, equivalently, kernel pixels) to be employed in the modelling of the convolution kernel. From a master set of $N$ DBFs, the model selection criteria will each select a single 'best' kernel design, which requires the evaluation of the criteria via the estimation of the model image parameters for each of the $2^N$ possible kernel designs.[5] This computational problem is formidable and currently infeasible for values of $N$ that are required for typical kernel models (e.g. a relatively small 9×9 kernel pixel grid yields $\sim$2.4 × $10^{24}$ potential kernel designs!). Furthermore, branch-and-bound algorithms (e.g. Furnival & Wilson 1974) for speeding up

[5] This number includes the kernel design with zero DBFs, i.e. a model image with the differential background as the only parameter.

**Table 1.** The number of DBFs in a circular kernel design for different ranges of the kernel radius $r_{\kappa}$. The ranges are defined by $r_{\kappa,\mathrm{lo}} \le r_{\kappa} < r_{\kappa,\mathrm{hi}}$. The table may be extended as appropriate for larger values of $r_{\kappa}$.

| $r_{\kappa,\mathrm{lo}}$ (pix) | $r_{\kappa,\mathrm{hi}}$ (pix) | $N_{\kappa}$ | $r_{\kappa,\mathrm{lo}}$ (pix) | $r_{\kappa,\mathrm{hi}}$ (pix) | $N_{\kappa}$ |
|---|---|---|---|---|---|
| 0.000 | 1.000 | 1 | 4.472 | 5.000 | 69 |
| 1.000 | 1.414 | 5 | 5.000 | 5.099 | 81 |
| 1.414 | 2.000 | 9 | 5.099 | 5.385 | 89 |
| 2.000 | 2.236 | 13 | 5.385 | 5.657 | 97 |
| 2.236 | 2.829 | 21 | 5.657 | 5.831 | 101 |
| 2.829 | 3.000 | 25 | 5.831 | 6.000 | 109 |
| 3.000 | 3.162 | 29 | 6.000 | 6.083 | 113 |
| 3.162 | 3.606 | 37 | 6.083 | 6.325 | 121 |
| 3.606 | 4.000 | 45 | 6.325 | 6.403 | 129 |
| 4.000 | 4.123 | 49 | 6.403 | 6.708 | 137 |
| 4.123 | 4.243 | 57 | 6.708 | 7.000 | 145 |
| 4.243 | 4.472 | 61 | – | – | – |

this exhaustive search are only applicable to some of our model selection criteria in Section 3.4.

It is well known that by not considering all of the possible combinations of predictor variables in a linear regression problem (e.g. by using stepwise regression for variable selection), the optimal set of predictors may be misidentified. However, in our case, we know from the nature/purpose of the kernel (and copious amounts of prior experience!) that the true kernel model has a peak signal at the kernel coordinates corresponding to the translational offset between the reference and target images (which is at the kernel origin when they are properly registered) and that this signal decays away from the peak. There may be other peaks (e.g. due to a telescope jump in the target image), but again these also have profiles that decay away from the peak(s). The best kernel designs are therefore generally limited to sets of DBFs in close proximity that form relatively compact and regular shapes. Based on these observations, we have devised two algorithms for automatic kernel design that compare a manageable number of sensible kernel models; the circular kernel design algorithm (Section 4.1) and the irregular kernel design algorithm (Section 4.2).

### 4.1 The circular kernel design algorithm

One very simple way to greatly reduce the number of candidate kernel designs that is in line with the expected kernel properties is to restrict the kernel designs to those that correspond to a circularly shaped pixel grid centred at the origin of the kernel pixel coordinates. We therefore define a circular kernel design of radius $r_{\kappa}$ as the set of DBFs corresponding to the kernel pixels whose centres lie at or within $r_{\kappa}$ pixels of the kernel origin, which is taken to be at the centre of the $(r, s) = (0, 0)$ kernel pixel. As $r_{\kappa}$ is increased, the circular kernel design includes progressively more DBFs leading to a set of nested kernel designs. In Table 1, we list the number of DBFs in a circular kernel design for a range of values of $r_{\kappa}$.

The circular kernel design algorithm (CKDA) works for a pair of images and an adopted model selection criterion. The algorithm sequentially evaluates a set of nested model images. It finishes when the current model image under consideration fails the selection criterion, and consequently the previously considered model image is selected. For the $\Delta\chi^2$ and $F$-tests, this means that the current model image does not provide a significantly better fit than the previous one. For the information criteria, this means that the current model image yields a larger value of the criterion than the previous one, where $\lambda$ has already been optimized individually for each model if appropriate. The algorithm proceeds as follows.

(i) Fit the model image with the differential background $B$ as the only parameter and calculate the desired model selection criterion.

(ii) Adopt a circular kernel design of radius $r_{\kappa} = 0.5$ pix, which defines a kernel model with a single DBF. Fit the model image consisting of the differential background and the kernel model, and calculate the desired model selection criterion. If the model image from (i) is selected, then finish.

(iii) Increment $r_{\kappa}$ until the kernel model includes a new (larger) set of DBFs. Fit the model image consisting of the differential background and the new kernel model, and calculate the desired model selection criterion. If the model image from the previous iteration is selected, then finish.

(iv) Repeat step (iii) until the algorithm terminates.

Note that the CKDA is intended to be applied to reference and target images that are registered to within a single pixel (but without requiring sub-pixel alignment necessitating image resampling).

Special care must be taken when using sigma-clipping during the fitting of the model images in the CKDA. Since each model image fit within the algorithm has the potential to clip different sets of target-image pixel values, the calculation of the model selection criterion may end up employing different sets of pixels at each step, which leads to undesirable jumps in its value that are unrelated to the properties of the fits. If sigma-clipping is required due to the presence of outlier pixel values, then, to avoid this problem, it is recommended to run the CKDA to conclusion without using sigma-clipping and to use the selected model image to identify and clip the outliers. The CKDA may then be re-run ignoring this same set of clipped pixel values at each step, and the whole process may be iterated more than once if necessary. This issue with sigma-clipping applies to all kernel design algorithms, and also whenever a fair comparison between algorithms is required (see Section 6).

In the early phases of testing the CKDA, we ran the algorithm past the finishing point to check that kernel designs with larger radii than the radius of the selected design do not yield smaller values of the information criterion, which, if this was the case, would indicate that the algorithm is terminating too early at a local minimum. We found that only in a relatively small proportion of the simulations (Section 5) a slightly smaller value of the information criterion is achieved for a kernel design with a larger radius than the selected design (usually 2–3 steps larger in Table 1), and that when this occurs, the values of the model performance metrics (Section 5.2) for the two designs are very similar with no systematic improvement for the kernel design with a larger radius. Given that running the CKDA to larger radii comes at considerable cost in terms of processing power, the termination criterion of the CKDA was fixed at the first minimum of the information criterion. The same conclusions were also found to apply to the irregular kernel design algorithm (Section 4.2).

### 4.2 The irregular kernel design algorithm

Another way to limit the number of candidate kernel designs is to ‘grow’ the kernel model as a connected set of DBFs from a single ‘seed’ DBF by including one new DBF at each iteration. We call this the irregular kernel design algorithm (IKDA), and it works for a pair of images and an adopted model selection criterion as follows.

(i) Fit the model image with the differential background $B$ as the only parameter and calculate the desired model selection criterion.

(ii) Define a master set of $N$ DBFs by taking an appropriately large grid of kernel pixels centred on the pixel at the kernel origin. For each DBF in the master set, fit the model image consisting of the differential background and a kernel model with the single DBF, and calculate the desired model selection criterion. If the model image from (i) is selected in all $N$ cases, then finish. Otherwise, accept the DBF from the master set that gives the best model image (according to the selection criterion) as the first DBF to be included in the kernel model (referred to as the seed DBF). Remove the seed DBF from the master set.

(iii) Find the subset of DBFs from the master set that are adjacent to at least one of the DBFs in the kernel model from the previous iteration. For each candidate DBF in this subset, fit the model image consisting of the differential background and a new kernel model with a set of DBFs that is the union of the set of DBFs in the kernel model from the previous iteration with the candidate DBF. If the model image from the previous iteration is selected in all cases, then finish. Otherwise, accept the candidate DBF that gives the best model image as the next DBF to be included in the kernel model. Remove the accepted candidate DBF from the master set.

(iv) Repeat step (iii) until the algorithm terminates.

Note that the IKDA may be applied to reference and target images that are not registered to within a single pixel since step (ii) is effectively a form of image registration. Again, special care must be taken with the application of sigma-clipping within the IKDA (see Section 4.1).

The IKDA may generate different sequences of DBFs during the growth of the kernel model for different model selection criteria. However, for the $\Delta\chi^2$ and $F$-statistics, the IKDA follows the same sequence of DBFs since both statistics are maximized at each iteration of the IKDA by minimizing $\chi^2_{\mathrm{B}}$. For similar reasons, the IKDA follows the same sequence of DBFs for the AIC$_{\mathrm{C}}$ and the BIC. In these cases, the different model selection criteria simply terminate the IKDA at different points in the sequence. Still, regardless of the actual model selection criterion used, we find that the IKDA always grows the kernel solution outwards from the selected seed DBF.

There are various alternative ways in which the kernel model may be grown within the IKDA. We have experimented with dropping the constraint that each new DBF must be adjacent to at least one DBF in the previous kernel model. However, this produced similar kernel solutions to those produced by the IKDA with the adjacency constraint but with an extra scattering of isolated DBFs arbitrarily far from the peak signal in the kernel. We also experimented with relaxing the definition of ‘adjacent’ to include more nearby kernel pixels, but the results from these versions of the IKDA are virtually indistinguishable in terms of the model performance metrics from the results for the IKDA described above (both for the simulated and real data). Hence we have not considered these variations on the IKDA any further.

Finally, we mention that the IKDA may be modified to generate multiple seed DBFs (possibly as part of step (ii) or by generating a new seed DBF after the algorithm terminates for the first time). This modification would be useful for adapting to situations similar to when the telescope has jumped during a target image exposure, and consequently the true kernel model consists of two or more disconnected peaks.

# 5 TESTING AUTOMATIC KERNEL DESIGN ALGORITHMS ON SIMULATED IMAGES

The main aim of this paper is to find out which combination of kernel design algorithm and model selection criterion consistently selects a kernel model that provides the best performance in terms of model error, fit quality, and photometric accuracy. The conclusions drawn from our investigation will likely depend on the properties of the reference and target images, and hence we must systematically map out the performance of each method accordingly. This task is most efficiently performed by generating and analysing simulated images. Furthermore, with respect to model error, the performance of each method may only be measured through the use of simulations where the true model image is known. Simulations also provide a setting in which the noise model is precisely known since it is used to generate the simulated data. Thus simulations allow for the degree of under- or overfitting to be assessed accurately. For these reasons, we have performed detailed DIA simulations for a wide range of image properties.

## 5.1 Generating simulated images

We employed a Monte Carlo method for our investigation. We adopted reasonable values for the CCD readout noise and gain of $\sigma_0 = 5$ ADU and $G = 1$ e$^-$/ADU, respectively. For each simulation, we generated both noiseless and noisy versions of a reference and target image pair, along with the noise maps used for generating the noisy images, via the following procedure.

(i) The size of the reference image was set to 141 × 141 pixels.

(ii) The sky background level for the reference image $S_{\mathrm{ref}}$ was drawn from a uniform distribution on the interval [16, 1000] ADU.

(iii) The log of the field star density, parametrized as the number of stars per 100 × 100 pixel image region, was drawn from a uniform distribution on the interval [0, 3], and this density was used to calculate the number of stars $N_{\mathrm{star}}$ to be generated in the reference image.

(iv) The pixel coordinates of each star in the reference image were drawn from a uniform distribution over the image area. Also, for each star, the value of $\mathcal{F}^{-3/2}$, where $\mathcal{F}$ is the star flux (ADU), was drawn from a uniform distribution on the interval $[10^{-9}, 10^{-9/2}]$. This flux distribution is appropriate when imaging to a fixed depth through a uniform space density of stars (e.g. a good approximation for certain volumes in our Galaxy). For the purposes of performing PSF photometry on the difference image, and without loss of generality, the pixel coordinates of the brightest star were modified by an integer pixel shift to lie within the central pixel of the reference image.

(v) The same normalized two-dimensional Gaussian profile of full-width at half-maximum (FWHM) $f_{\mathrm{ref}}$ pixels was adopted for the profile of each star in the reference image. The value of $f_{\mathrm{ref}}$ was drawn from a uniform distribution on the interval [1, 6] pix, adequately covering the under- to oversampled regimes.

(vi) A square image pixel array $R_{\mathrm{noiseless}}$ of size 141 × 141 pixels was created with all of the pixel values set to the sky background level $S_{\mathrm{ref}}$.

(vii) For each star in the reference image, the Gaussian profile was centred at the star pixel coordinates and sampled at 7 times the image resolution over the image area. The oversampled Gaussian profile was then binned (by averaging) to match the image resolution and renormalized to a sum of unity. Finally, it was scaled by the star flux and added to the image $R_{\mathrm{noiseless}}$.

(viii) An image of standard deviations $\sigma_{\mathrm{in,\,ref}}$ (i.e. a noise map) corresponding to $R_{\mathrm{noiseless}}$ was created via

$$\sigma_{\mathrm{in,ref},ij} = \sqrt{\sigma_0^2 + R_{\mathrm{noiseless},ij}/G} \tag{53}$$

which may be derived from equation (13) by setting the $F_{ij} = 1$. A 141 × 141 pixel image $W$ of values drawn from a Gaussian distribution with zero mean and unit standard deviation was also generated and used to construct a noisy reference image $R_{\mathrm{noisy}}$ via

$$R_{\mathrm{noisy},ij} = R_{\mathrm{noiseless},ij} + W_{ij}\,\sigma_{\mathrm{in,ref},ij}. \tag{54}$$

(ix) The size of the target image was set to 141 × 141 pixels.

(x) For simplicity, the sky background level for the target image $S_{\mathrm{tar}}$ was set to $S_{\mathrm{ref}}$, which is equivalent to assuming a differential background of zero.

(xi) A single sub-pixel shift in each of the $x$ and $y$ image coordinate directions was drawn from a uniform distribution on the interval $[-0.5, 0.5]$ pix and applied to the pixel coordinates of the stars in the reference image to generate the coordinates of the same stars in the target image. The fluxes of the stars in the target image were assumed to be the same as their fluxes in the reference image, which is equivalent to assuming non-variable stars and a photometric scale factor of unity.

(xii) The convolution kernel matching the PSF between the reference and target images was assumed to be a normalized two-dimensional Gaussian profile of FWHM $|f_{\mathrm{ker}}|$ pixels, where a non-negative or negative value of $f_{\mathrm{ker}}$ indicates that the kernel convolves the reference or target image PSF, respectively. The value of $f_{\mathrm{ker}}$ was drawn from a uniform distribution on the interval $[-1, 5]$ pix and the FWHM $f_{\mathrm{tar}}$ of the Gaussian profile for the stars in the target image was then calculated from

$$f_{\mathrm{tar}}^2 = \begin{cases} f_{\mathrm{ref}}^2 + f_{\mathrm{ker}}^2 & \text{for } f_{\mathrm{ker}} \geq 0 \\ f_{\mathrm{ref}}^2 - f_{\mathrm{ker}}^2 & \text{for } f_{\mathrm{ker}} < 0. \end{cases} \tag{55}$$

(xiii) A square image pixel array $I_{\mathrm{noiseless}}$ of size 141 × 141 pixels was created with all of the pixel values set to the sky background level $S_{\mathrm{tar}}$.

(xiv) The flux profiles of the stars in the target image were added to $I_{\mathrm{noiseless}}$ using the same method as that used in step (vii) for the reference image.

(xv) An image of standard deviations $\sigma_{\mathrm{in,\,tar}}$ corresponding to $I_{\mathrm{noiseless}}$ was created via

$$\sigma_{\mathrm{in,tar},ij} = \sqrt{\sigma_0^2 + I_{\mathrm{noiseless},ij}/G}. \tag{56}$$

A new 141 × 141 pixel image $W$ of values drawn from a Gaussian distribution with zero mean and unit standard deviation was also generated and used to construct a noisy target image $I_{\mathrm{noisy}}$ via

$$I_{\mathrm{noisy},ij} = I_{\mathrm{noiseless},ij} + W_{ij}\,\sigma_{\mathrm{in,tar},ij}. \tag{57}$$

(xvi) The images $I_{\mathrm{noiseless}}$, $I_{\mathrm{noisy}}$, and $\sigma_{\mathrm{in,\,tar}}$ were each trimmed to a size of 101 × 101 pixels. Hence the number of data values in each simulation is $N_{\mathrm{dat}} = 10201$.

(xvii) The signal-to-noise (S/N) ratio of the noisy target image $\mathrm{SN}_{\mathrm{tar}}$ was calculated as

$$\mathrm{SN}_{\mathrm{tar}} = \frac{\sum_{ij}(I_{\mathrm{noiseless},ij} - S_{\mathrm{tar}})}{\sqrt{\sum_{ij}\sigma_{\mathrm{in,tar},ij}^2}}. \tag{58}$$

The value of $\log(\mathrm{SN}_{\mathrm{tar}})$ is distributed approximately uniformly due to the way the field star density is generated in step (iii). It is important to note that it does not necessarily follow that a high S/N target image has a bright high-S/N star in the centre. The high S/N in the target image may be the consequence of the presence of a reasonable number of faint stars. In this case, the star at the centre of the image is of low S/N, even though it is the brightest star in the image.

In total, we generated the reference and target images for 548 392 simulations. We call this set of images 'Simulation set S1'.

The above method for generating reference and target images represents the case where the reference image has approximately the same S/N ratio as the target image. However, it is common practice in DIA to create a high-S/N ratio reference image by stacking images or integrating longer. We therefore also repeated the whole procedure of generating simulated images for reference images with 10 times less variance in each pixel value than the corresponding target images (or ∼3.16 times better S/N). This was achieved by scaling the $\sigma_{\mathrm{in,\,ref},\,ij}$ in step (viii) by $10^{-1/2} \sim 0.316$. This second set of reference and target images, 'Simulation set S10', was generated for a total of 529 571 simulations. Fig. 1 shows an example noisy reference and target image pair from one of these simulations.

### 5.2 Model performance metrics

We used the following metrics to measure the quality of each kernel and differential background solution.

(i) Model error. The mean squared error (MSE) measures how well the fitted model image $M$ matches the true model image $I_{\mathrm{noiseless}}$. It is defined by

$$\mathrm{MSE} = \frac{1}{N_{\mathrm{dat}}}\sum_{ij}(M_{ij} - I_{\mathrm{noiseless},ij})^2. \tag{59}$$

Kernel and differential background solutions with the smallest values of MSE exhibit the best performance in terms of model error.

We also consider the photometric scale factor $P$ and the differential background $B$ as supplementary measures of model error. For our simulations, the closer to unity the value of $P$ and the closer to zero the value of $B$, the better the performance of a kernel and differential background solution with respect to model error. Systematic errors in the photometric scale factor are especially important since a fractional error $\mathrm{E}_P$ in $P$ introduces a fractional error of $\mathrm{E}_P$ into the photometry (Bramich et al. 2015).

(ii) Fit quality. The bias and excess variance in the fitted model image may be measured by the following statistics

$$\mathrm{MFB} = \frac{1}{N_{\mathrm{dat}}}\sum_{ij}\frac{(I_{\mathrm{noisy},ij} - M_{ij})}{\sigma_{\mathrm{in,tar},ij}} \tag{60}$$

$$\mathrm{MFV} = \frac{1}{N_{\mathrm{dat}} - 1}\sum_{ij}\left(\frac{(I_{\mathrm{noisy},ij} - M_{ij})}{\sigma_{\mathrm{in,tar},ij}} - \mathrm{MFB}\right)^2. \tag{61}$$

MFB is the mean fit bias and MFV is the mean fit variance with units of sigma and sigma-squared, respectively. The closer to zero the value of MFB, and the closer to unity the value of MFV, the better the performance of a kernel and differential background solution with respect to fit quality.

(iii) Photometric accuracy. To assess the photometric accuracy, we perform PSF fitting on the difference image at the position of the brightest star in the reference image under the assumption that the reference image PSF is perfectly known. In detail, we generate a normalized two-dimensional Gaussian profile of FWHM $f_{\mathrm{ref}}$ pixels centred at the pixel coordinates of the brightest star in the reference image (guaranteed by construction to be within half a pixel of the

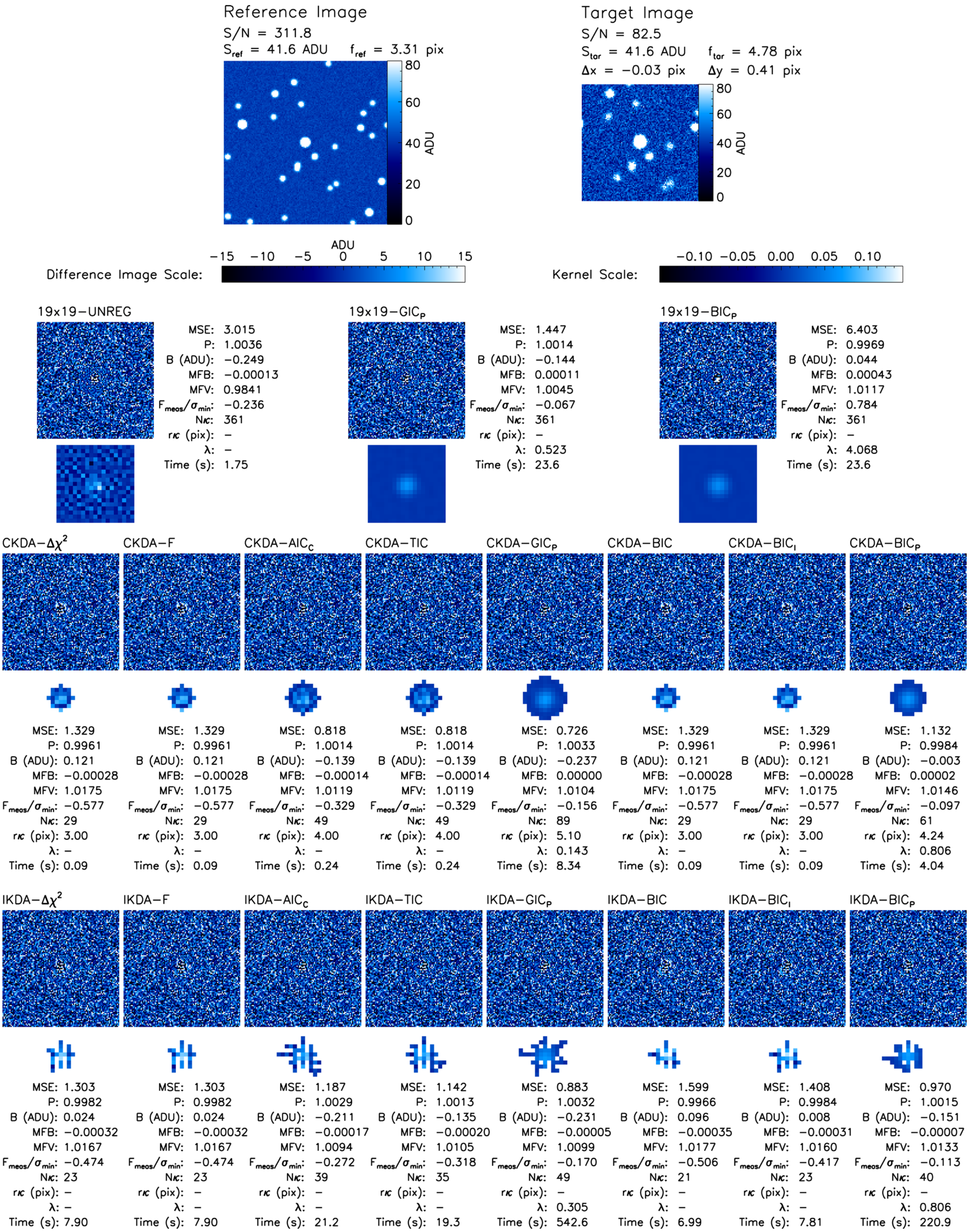


**Figure 1.** A reference and target image pair from simulation set S10 are shown at the top. The corresponding results for each of the 19 kernel solution methods are shown below. For each method, the difference image, kernel solution, and model performance metrics are displayed. The difference images and kernels are all displayed using the same linear scales of [−15, 15] ADU and [−0.14, 0.14], respectively. Processing times were measured for non-optimized code running on a desktop computer with an Intel Core i7-2600 CPU (3.40 GHz) and 16 Gb RAM.

image centre) and sampled at 7 times the image resolution. The oversampled Gaussian is then binned (by averaging) to match the image resolution, convolved with the kernel solution, trimmed in extent to a circularly shaped pixel grid of radius $\lceil 2f_{\rm tar} \rceil$ pixels around the star coordinates, and renormalized. This model PSF for the target image is then optimally scaled to the difference image at the position of the brightest star by simultaneously fitting a scaling factor $\mathcal{F}_{\rm diff}$ and an additive constant, and using the known pixel variances in the target image $\sigma^2_{\rm in,tar}$. The difference flux $\mathcal{F}_{\rm meas}$ of the brightest star on the photometric scale of the reference image is then computed using $\mathcal{F}_{\rm meas} = \mathcal{F}_{\rm diff}/P$.

The theoretical minimum variance $\sigma^2_{\rm min}$ in the difference flux $\mathcal{F}_{\rm meas}$ for PSF fitting with a scaling factor only is given by

$$\sigma^2_{\rm min} = \frac{1}{P^2_{\rm true}} \left( \sum_{ij} \frac{\mathcal{P}^2_{{\rm tar},ij}}{\sigma^2_{{\rm in,tar},ij}} \right)^{-1} , \qquad (62)$$

where $P_{\rm true}$ is the true photometric scale factor ($P_{\rm true} = 1$ in our simulations) and $\mathcal{P}_{\rm tar}$ is the true PSF for the brightest star in the target image (a normalized two-dimensional Gaussian profile of FWHM $f_{\rm tar}$ pixels in our simulations). Since all of the stars in the simulations are non-variable, the best kernel and differential background solutions should yield a distribution of values of $\mathcal{F}_{\rm meas}/\sigma_{\rm min}$ with zero mean and unit variance. Hence, for a set of $N_{\rm set}$ simulations indexed by $k$, appropriate measures for assessing the photometric accuracy are:

$$\mathrm{MPB} = \frac{1}{N_{\rm set}} \sum_k \frac{\mathcal{F}_{{\rm meas},k}}{\sigma_{{\rm min},k}} \qquad (63)$$

$$\mathrm{MPV} = \frac{1}{N_{\rm set} - 1} \sum_k \left( \frac{\mathcal{F}_{{\rm meas},k}}{\sigma_{{\rm min},k}} - \mathrm{MPB} \right)^2 . \qquad (64)$$

MPB is the mean photometric bias and MPV is the mean photometric variance with units of $\sigma_{\rm min}$ and $\sigma^2_{\rm min}$, respectively. We note that even though MPV is normalized by the theoretical minimum variance in the difference flux, it may still achieve values that are less than unity when the target image is overfit and/or when the model PSF for the target image differs from the true PSF.

### 5.3 Results

For each possible combination of kernel design algorithm and model selection criterion, we computed kernel and differential background solutions for all of the reference and target image pairs in both of the simulation sets S1 and S10. Furthermore, for comparison purposes, for each simulation we solved for a model image employing a square 19×19-pixel kernel design which corresponds to the unregularized kernel analysed in Be12. We also solved for the same 19×19-pixel kernel design with regularized DBFs where the optimal choice of λ was determined using either $\mathrm{GIC_P}(\lambda)$ or $\mathrm{BIC_P}(\lambda)$ (equations 51 and 52) which corresponds to the regularized kernel analysed in Be12. In all cases, we used three iterations for each solution, but without employing sigma-clipping since the simulated images do not suffer from outlier pixel values (see Section 2.1). The optimization of λ for the $\mathrm{GIC_P}$ and $\mathrm{BIC_P}$ model selection criteria was performed using a binary search algorithm in $\log(\lambda)$ for the range $-3 \le \log(\lambda) \le 3$ with a final resolution in λ of 15 per cent, while also considering the limit $\lambda = 0$. Finally, the corresponding model performance metrics from Section 5.2 were calculated for each solution.

Hereafter we use a string of the form <ALGORITHM>-<CRITERION> to refer to a specific combination of kernel design algorithm (CKDA or IKDA) and model selection criterion ($\Delta\chi^2$, $F$, $\mathrm{AIC_C}$, TIC, $\mathrm{GIC_P}$, BIC, $\mathrm{BIC_I}$ or $\mathrm{BIC_P}$). For the 19×19-pixel kernel design, we use 19×19-UNREG, 19×19-$\mathrm{GIC_P}$, and 19×19-$\mathrm{BIC_P}$. Each of these combinations constitutes a kernel solution method, and hence we have 19 methods to consider.

In Fig. 1, we show the difference images, kernel solutions, and model performance metrics for each of the 19 kernel solution methods applied to the reference and target image pair displayed at the top (taken from simulation set S10). The target image is of medium S/N, and the reference and target images are both oversampled (with $f_{\rm ker} > 2.35$ pix). Notice how the regularization in the 19×19-$\mathrm{GIC_P}$ and 19×19-$\mathrm{BIC_P}$ methods drastically reduces the noise in the kernel compared to the 19×19-UNREG method as demonstrated previously in Be12. Notice also how, as expected, the BIC-type criteria (BIC, $\mathrm{BIC_I}$, and $\mathrm{BIC_P}$) select kernel designs with fewer DBFs than the kernel designs selected by the AIC-type criteria ($\mathrm{AIC_C}$, TIC, and $\mathrm{GIC_P}$). Somewhat surprising is the 'spidery' form of the kernel solutions generated by the IKDA. A selection of model PSFs for this target image, used to perform PSF fitting on the difference images, are displayed in the top row of Fig. 2 alongside the true PSF for the brightest star. The residuals of these model PSFs from the true PSF (bottom row) demonstrate that the spidery form of the IKDA kernel solutions has no discernable detrimental effect, when compared to the other kernel solutions, on the convolution of the reference image PSF to obtain the target image PSF.

To provide an idea of what the functional forms of $\mathrm{GIC_P}(\lambda)$ and $\mathrm{BIC_P}(\lambda)$ look like, we plot these quantities as a function of λ for two example simulations in Fig. 3. Each plot shows the curves for the CKDA-$\mathrm{GIC_P}$, CKDA-$\mathrm{BIC_P}$, 19×19-$\mathrm{GIC_P}$ and 19×19-$\mathrm{BIC_P}$ methods. Clear minima exist indicating the optimal values of λ. All of the simulations yield similar functional forms for $\mathrm{GIC_P}(\lambda)$ and $\mathrm{BIC_P}(\lambda)$, and while the minima of the $\mathrm{GIC_P}(\lambda)$ curves may

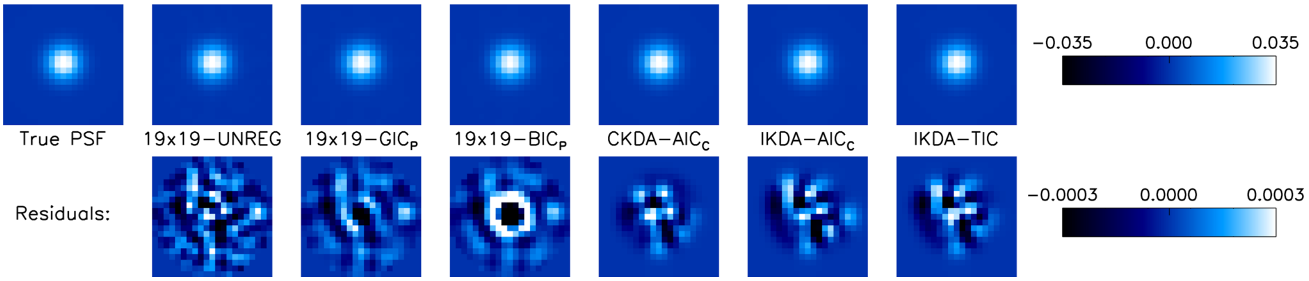
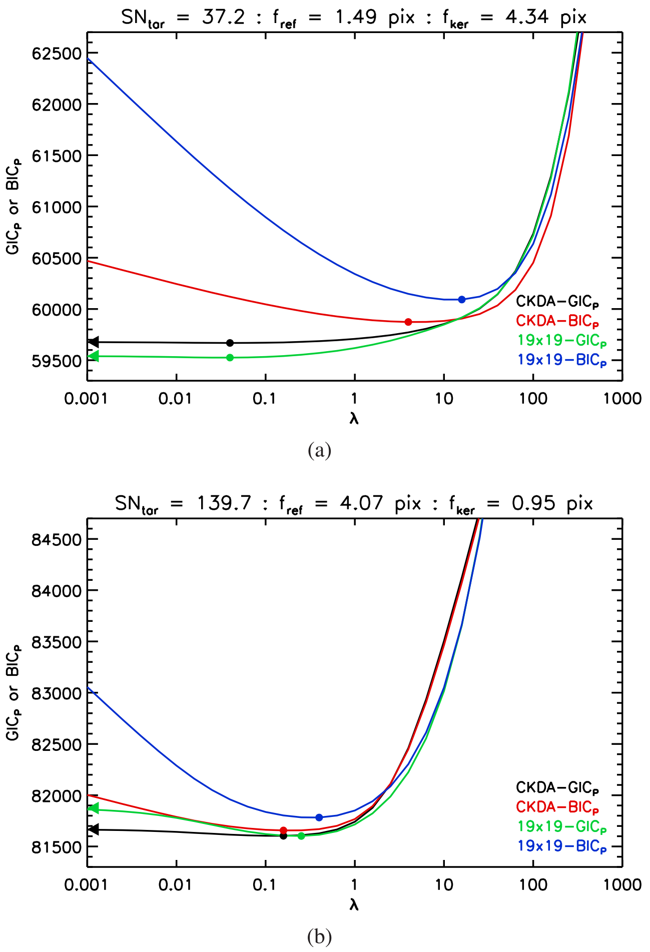


**Figure 2.** The true PSF for the brightest star in the target image from Fig. 1 is shown at the top left. The true PSF is a normalized two-dimensional Gaussian of FWHM 4.78 pix centred in the image stamp using the sub-pixel coordinates of the brightest star. A selection of six model PSFs for the target image are shown in the top row and labelled with the corresponding kernel solution methods. The residuals of these model PSFs from the true PSF are shown in the bottom row. Each row of plots uses the linear scale reproduced at the right-hand end of the row.



(a)

(b)

**Figure 3.** Examples from the simulations of the variation of the $\mathrm{GIC_P}$ and $\mathrm{BIC_P}$ criteria as a function of λ for the kernel solution methods CKDA-$\mathrm{GIC_P}$ (black), CKDA-$\mathrm{BIC_P}$ (red), 19x19-$\mathrm{GIC_P}$ (green), and 19×19-$\mathrm{BIC_P}$ (blue). For the CKDA-$\mathrm{GIC_P}$ and CKDA-$\mathrm{BIC_P}$ methods, the curves correspond to the selected kernel radius. For each curve, the minimum is marked with a filled circle. For the CKDA-$\mathrm{GIC_P}$ and 19×19-$\mathrm{GIC_P}$ methods, the value of the curve at $\lambda = 0$ is marked on the left-hand side of the plot with a triangle. Note that while $\mathrm{GIC_P}(\lambda)$ converges to TIC for $\lambda \to 0$, $\mathrm{BIC_P}(\lambda)$ diverges as $\lambda \to 0$ because of the divergence of the term involving $\ln \lambda$ in equation (52). Hence no triangles are plotted for the CKDA-$\mathrm{BIC_P}$ and 19×19-$\mathrm{BIC_P}$ methods.

sometimes lie at $\lambda = 0$, they very rarely lie at values of λ that are greater than 10 for $\mathrm{GIC_P}$, or that are greater than 100 for $\mathrm{BIC_P}$. Note that for the example shown in Fig. 3(b), the optimal value of λ for each method lies in the range 0.1–1.0 which matches with the recommendation for λ from Be12. However, for the other example shown in Fig. 3(a), the $\mathrm{GIC_P}$ and $\mathrm{BIC_P}$ criteria yield optimal values of λ that are <0.1 and >1.0, respectively.

In Fig. 4, for each kernel solution method we plot the median MSE, *P*, MFB, and MFV values, and the MPB and MPV measures, for a subset of our simulations corresponding to oversampled reference images ($f_\mathrm{ref} \geq 2.35$ pix) and kernels with $f_\mathrm{ker} \geq 2.35$ pix. The corresponding plots for *B* are not presented because the results are very similar to those for *P* since the photometric scale factor and differential background are correlated. We have further split the simulations into subsets based on target-image S/N (low: 8–40, medium: 40–200, high: 200–1000; three columns of plots) and reference image S/N (simulation sets S1 and S10; square or circular symbols). Similar style figures representing the results for different subsets of simulations chosen based on image sampling are presented in Appendix B (Figs B1–B4).

Within each subset of simulations, the distributions of the various model performance metrics are single-peaked bell-shapes with rapidly falling wings and they are not far-off being Gaussian in some cases. Skewness affects some of the distributions as do a few outlier points. However, for each simulation subset and model performance metric, the shape and width of the distributions are very similar between the kernel solution methods. The differences in the distributions lie in their central values. Consequently, we have used the median of the model performance metrics MSE, *P*, MFB, and MFV in Figs 4 and B1–B4 to compare the kernel solution methods since the median is a robust estimator of the central value. Given the Gaussian-like shape of the distributions of $\mathcal{F}_\mathrm{meas}/\sigma_\mathrm{min}$, our choice of measures MPB and MPV (equations 63 and 64) for assessing the photometric accuracy is justified.

The processing time to run the IKDA-$\mathrm{GIC_P}$ and IKDA-$\mathrm{BIC_P}$ methods is prohibitive (see the timings noted in Fig. 1). Hence we only ran these kernel solution methods on 25 410 and 25 320 reference and target image pairs from simulation sets S1 and S10, respectively. The results from these methods are plotted in Figs 4 and B1–B4, although they suffer from more noise than the other methods because they are derived from many fewer simulations. Consequently, we do not consider these two kernel solution methods any further.

### 5.4 Discussion

Unless otherwise stated, the discussion in this section refers to the results plotted in all of the Figs 4 and B1–B4, while Fig. 4 alone is sufficient to demonstrate the points raised.

In preparation for our discussion, it is worth considering how closely the candidate model images generated by our kernel design algorithms are able represent the true underlying model image. In each simulation, the Gaussian PSF profile in the reference (or target) image is convolved with a Gaussian kernel to obtain a Gaussian PSF profile in the target (or reference) image. In the limits of a noiseless reference image with infinitely fine image sampling, and for a kernel that convolves the reference image, a kernel of DBFs of infinite extent is sufficient to allow for a full representation of the true underlying model image (i.e. the noiseless target image). In practice, the reference image is noisy, the reference and target images are sampled at a finite scale with a spatial offset between them, the target image may be sharper than the reference image, and the kernel model employs a finite number of DBFs. It is clear therefore that none of the candidate model images will actually represent the true model image. However, for reference images with higher S/N and better sampling, and for kernel models employing more DBFs (without overfitting), the candidate model images will include models that are closer to the true model. Referring back to Section 3.2, it seems then that the model selection criteria derived considering the Kullback–Leibler divergence (i.e. the AIC-type criteria) should perform the best for DIA (especially in terms of model error), and that all of the criteria should perform better with improved reference image S/N and sampling.

Unsurprisingly then, the first major conclusion that can be drawn from the results of the simulations is that with very few exceptions it is vastly advantageous, as demonstrated by all of the model performance metrics, to use a reference image with a higher S/N than the target image regardless of the target-image S/N, the reference or target image FWHM, or the kernel solution method employed. Our discussion will therefore focus on the results for simulation set S10.

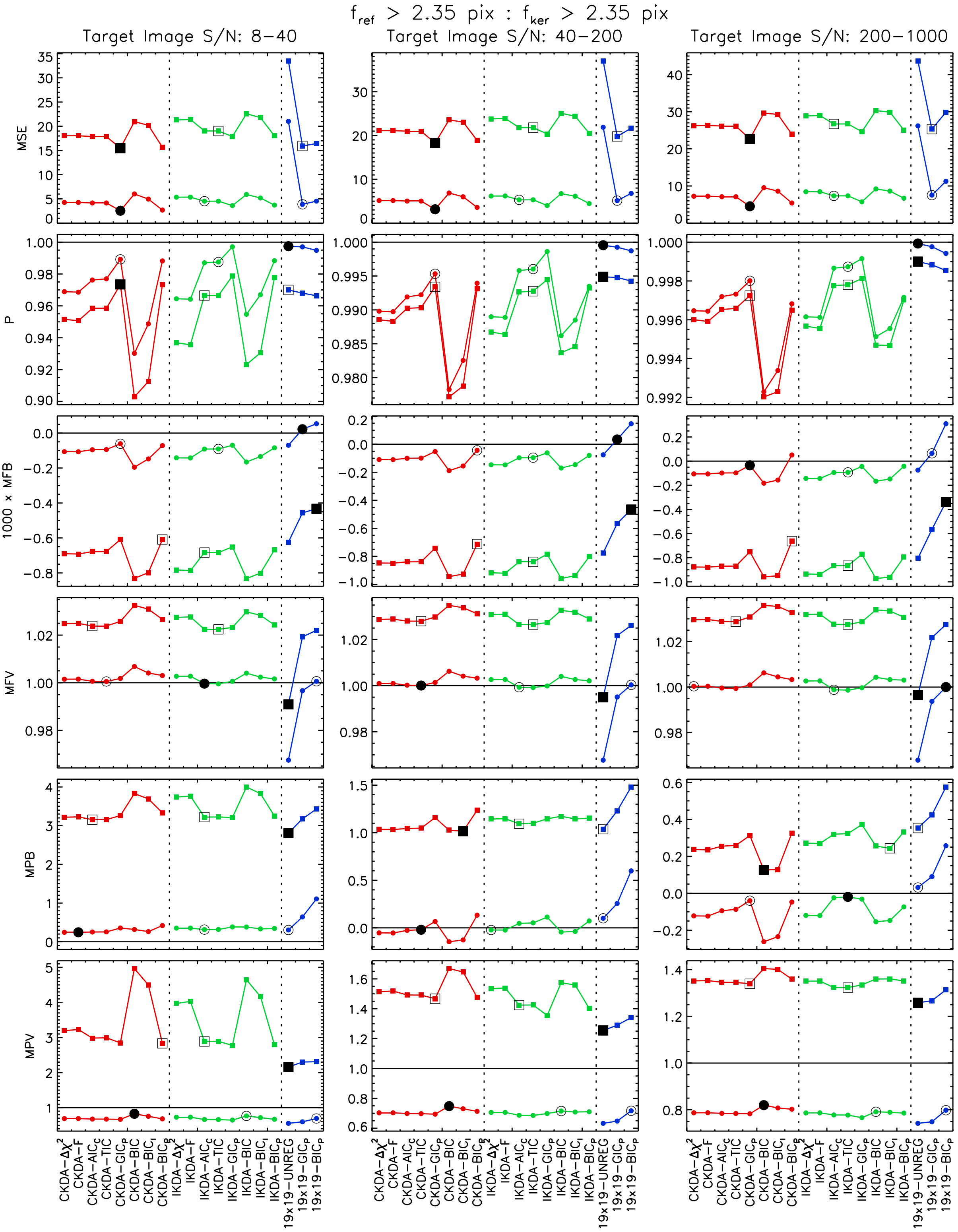


**Figure 4.** Plots of the median MSE, $P$, MFB, and MFV values (equations 59, 12, 60, and 61), and the MPB and MPV measures (equations 63 and 64), for each kernel solution method for $f_{\mathrm{ref}} \geq 2.35$ pix and $f_{\mathrm{ker}} \geq 2.35$ pix. The results in each plot have been calculated from ∼60 000 simulations for each of the simulation sets S1 and S10. Layout: the three columns of plots correspond to low (8–40), medium (40–200) and high (200–1000) S/N target images. Each row of plots corresponds to a different model performance metric. Individual plots: square and circular symbols represent the results for simulation sets S1 and S10, respectively. Red, green, and blue colours correspond to the kernel design algorithms CKDA, IKDA, and 19×19, respectively. For each algorithm, the kernel solution method with the best value of the relevant model performance metric is also plotted with an open black symbol. The method with the overall best metric value is plotted with a filled black symbol. The IKDA-GIC$_{\mathrm{P}}$ and IKDA-BIC$_{\mathrm{P}}$ methods are excluded when determining the best metric values since their results are noisier having been determined from many fewer simulations, and because they are too computationally intensive to be of practical use with currently available computing equipment.

Furthermore, the best estimates for the photometric scale factor are achieved for higher S/N target images, and that in general $P$ is underestimated. Since an accurate estimate of the photometric scale factor is crucial for performing accurate photometry (Bramich et al. 2015), our discussion will further focus on the results for target-image S/N ranges of 40–200 and 200–1000 where $P$ is estimated to better than 1 per cent for simulation set S10.

We observe that the smallest median MSE values for simulation set S10 are always achieved by a kernel solution method employing an AIC-type criterion. What differs between the S/N and FWHM regimes is which algorithm combined with an AIC-type criterion performs the best in terms of model error. This result validates our discussion at the beginning of this section about the fact that the set of candidate model images generated by our kernel design algorithms does not include the true underlying model.

We find that the median MFB values, which have units of sigma, are very small negative numbers with absolute values less than $\sim 3 \times 10^{-3} \sigma_{\rm min}$ (simulation set S10). Hence in terms of fit quality, we give more weight to the results for the MFV metric.

The results for the photometric accuracy in Fig. 4 reveal the surprising fact that the variance in the photometry of the brightest star for simulation set S10 is smaller than the theoretical minimum $\sigma^2_{\rm min}$. This can also be seen in Figs B2–B4. We have investigated how this might be possible. First, we checked the photometry of the faintest star in each simulation. We did this by modifying step (iv) of the image simulation procedure in Section 5.1 to shift the pixel coordinates of the faintest star (instead of the brightest) to lie within the central pixel of the reference image. We then generated 60 000 reference and target image pairs with the $\sigma_{{\rm in,\,ref,}\,ij}$ in step (viii) scaled by $10^{-1/2} \sim 0.316$ as was done for simulation set S10, and for each simulated image pair, we computed $\mathcal{F}_{\rm meas}/\sigma_{\rm min}$ for the faintest star for each kernel solution method. We found that the MPV measures for the faintest star are greater than unity with values in the range $\sim$1.1–1.6 $\sigma^2_{\rm min}$ for Figs 4 and B2–B4. Hence the variance in the photometry of the faintest stars does not achieve the theoretical minimum $\sigma^2_{\rm min}$.

We believe that these facts may be explained by considering that all of the kernel solution methods are overfitting the brightest star(s) and underfitting the faintest star(s) in most of our simulations. Careful inspection of the difference images in Fig. 1 reveals that the pixel noise in the area around the brightest star is suppressed,[6] and one can see that this effect is visible to various extents for all of the kernel solution methods including the methods employing regularization. Furthermore, the noise suppression around the brightest star is clearly visible in figs 2 and 3 of Be12. We experimented with using a noise model in our simulations with equal pixel variances calculated using the sky background level only. In this case, the target-image pixel values for the brightest star are given even more weight relative to the other pixels when solving for the kernel and differential background, and the MPV measures for the brightest star are found to be even smaller. However, if we increase the size of the images in our simulations, the MPV measures that are smaller than unity in Figs 4 and B2–B4 are found to increase to values that are closer to unity. The same effect may also be achieved by considering only those simulations with higher star densities.

From this we may conclude that the kernel solution methods which yield MPV measures that are closest to unity, regardless of whether they are greater than or less than unity, are those that achieve the best balance between under- or overfitting the target image for the brightest stars. Consequently, it is these methods that produce the most reliable photometry whenever the corresponding MPB measures are also closest to zero. Also, it is clear that in practice the image regions used to derive the kernel solutions should be as large as possible while satisfying the assumption of a spatially invariant kernel, and that ideally they should each contain at least a few bright objects. This helps to avoid the situation where a single bright object dominates the kernel solution in each region.

Based on the above general observations and discussion, we have attempted a detailed analysis of the results presented in Figs 4 and B1–B4. However, it has proven impossible to identify any single kernel solution method, or even an individual algorithm or criterion, that consistently performs the best for all of the model performance metrics. Even breaking the analysis down into each of the five image sampling regimes does not help much. Since we are unable to reach a clear conclusion from the way the results have been analysed and presented so far, further investigation is required.

Finally, we checked how well the kernel solutions recover the input sub-pixel shifts between the reference and target images. To do this, we computed the centroid of each kernel solution and compared the centroid coordinates to the appropriate sub-pixel shifts. Reassuringly, we find that the residuals are scattered around zero with decreasing scatter for higher S/N target images. Furthermore, we note that while all of the CKDA and IKDA methods perform equally well in recovering the shifts (e.g. $\sim$0.015 pix rms at $\mathrm{SN_{tar}} \sim 300$), the 19×19-UNREG, 19×19-GIC$_{\rm P}$, and 19×19-BIC$_{\rm P}$ methods all perform considerably worse in this respect (e.g. $\sim$0.043 pix rms at $\mathrm{SN_{tar}} \sim 300$).

### 5.5 Further investigation

One of the most important applications of DIA is for precision photometry. Therefore, we are highly motivated in developing a kernel solution method that provides the best possible photometry in the sense that the chosen method should deliver the smallest photometric bias while also striking the optimal balance between under- and overfitting. In photometric applications, DIA is typically used to obtain photometry for the objects in a set of time series images. The properties of these images, such as the PSF FWHM and S/N, will likely vary substantially during the course of the observations (e.g. due to the atmosphere). Therefore, a further aspect on which we may assess the kernel solution methods studied in this paper is on the uniformity of the photometric bias and variance as a function of FWHM and S/N.

In Fig. 5, for each kernel solution method and target-image S/N regime, we plot surfaces representing the MPB measure for simulation set S10 as a function of the reference image and kernel FWHM. A circular smoothing region of radius 0.33 pix, which encompasses the results from $\sim$2000 simulations, is used to calculate the MPB surface values. Blue and red colours in the plot panels indicate positive and negative mean photometric biases, respectively, while white indicates zero bias. The image sampling regimes corresponding to Figs 4 and B1–B4 are delimited in each plot by continuous black lines. These plots are complimentary to those in Figs 4 and B1–B4 in the sense that they reveal considerably more detail about the dependence of the MPB measure on image sampling, even though it is more difficult to assess the exact MPB values in each case.

Since the DIA photometry for a set of time series images is extracted using a single reference image, the uniformity of the MPB surfaces in Fig. 5 should be assessed via horizontal cuts (i.e. at fixed $f_{\rm ref}$), and by comparing the cuts between the plot columns

[6] The effect is more easily discernable on a digital display than on a printed copy.

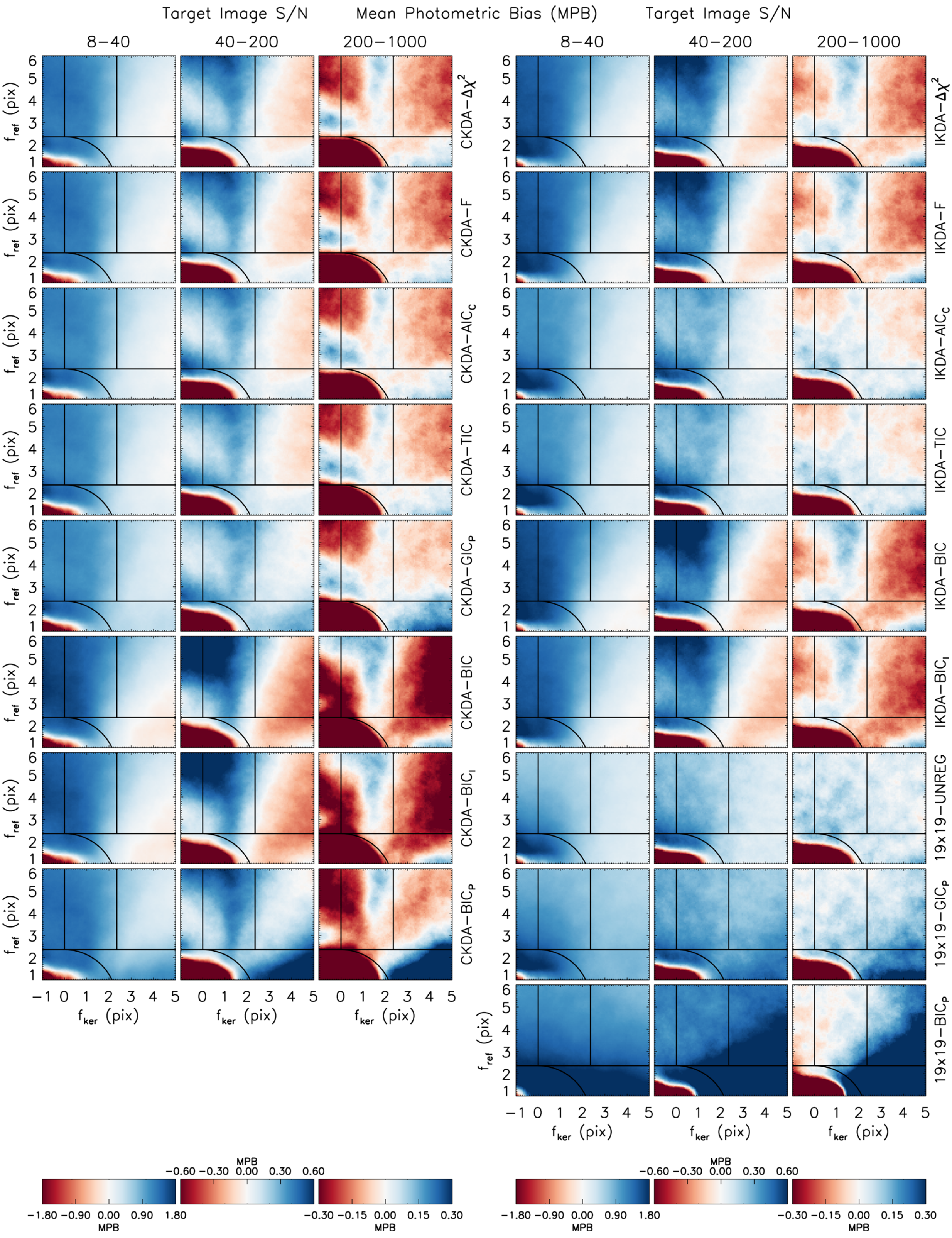


**Figure 5.** Plots of surfaces representing the MPB measure (equation 63) for simulation set S10 as a function of the reference image and kernel FWHM. Each plot corresponds to a specific kernel solution method (labelled on the right-hand side of each row of three plots) and target-image S/N regime (labelled at the top of each column of plots). The colour-intensity bar corresponding to each column of plots is reproduced at the bottom of the figure. In each plot, the surface values are calculated using a circular smoothing region of radius 0.33 pix and the image sampling regimes corresponding to Figs 4 and B1–B4 are delimited by continuous black lines. Specifically, the curved line corresponds to a critically sampled target image for an undersampled reference image (i.e. $f_{\rm tar} = 2.35$ and $f_{\rm ref} \leq 2.35$ pix).

(for S/N dependence). Immediately it is clear to the eye that the IKDA-AIC$_{\rm C}$, IKDA-TIC, 19×19-UNREG and 19×19-GIC$_{\rm P}$ methods produce by far the most uniform and least biased MPB surfaces in the S/N>40 regime with the IKDA-AIC$_{\rm C}$ and IKDA-TIC methods yielding nearly identical results. Specifically, in the oversampled reference image regime and for S/N>200, each of these four methods suffers from only a small photometric bias (MPB $\sim\pm\,0.1\,\sigma_{\rm min}$), with the two IKDA methods showing both positive and negative biases, and the two 19×19-pixel kernel designs showing just a positive bias. Between S/N regimes, the IKDA-AIC$_{\rm C}$, IKDA-TIC, and 19×19-UNREG methods are relatively uniform whereas the 19×19-GIC$_{\rm P}$ method exhibits more noticeable non-uniformity. In the undersampled reference image regime with oversampled target images, the IKDA-AIC$_{\rm C}$, IKDA-TIC, and 19×19-UNREG methods produce very similar uniform MPB surfaces with a slight positive bias of $\sim 0.1\sigma_{\rm min}$, while the 19×19-GIC$_{\rm P}$ method shows significant non-uniformity. In the undersampled reference and target image regime, the MPB surfaces for all 17 kernel solution methods have large gradients indicating that the time series photometry will suffer from large systematic variations whenever there are small variations in $f_{\rm ker}$ between the images (unsurprisingly). For S/N>200, the four kernel solution methods identified here provide uniform MPB surfaces down to $f_{\rm ref}\sim 2.1$ and $f_{\rm tar}\sim 2.1$ pix (slightly below critical sampling).

The surfaces representing the MPV measure for simulation set S10 as a function of the reference image and kernel FWHM, created in the same fashion as the MPB surfaces, are displayed in Fig. 6. They clearly show that all of the kernel solution methods yield an MPV that is smaller than the theoretical minimum $\sigma^2_{\rm min}$ (except in the undersampled reference and target image regime). In Section 5.4, we came to the conclusion that this is because the kernel solution methods are overfitting the brightest star(s). The 19×19-UNREG method is the worst performer in this respect, followed closely by 19×19-GIC$_{\rm P}$ and then 19×19-BIC$_{\rm P}$. Otherwise, the MPV surfaces for the CKDA and IKDA are all very similar with the IKDA providing more uniform photometric variance near the locus of critically sampled target images. The MPV surfaces for the IKDA-AIC$_{\rm C}$ and IKDA-TIC methods are virtually indistinguishable.

Hence, we may conclude that the IKDA-AIC$_{\rm C}$ and IKDA-TIC methods are equally the best kernel solution methods in terms of the photometry that they yield, with the 19×19-GIC$_{\rm P}$ method coming in as a close second best. The plots of the equivalent surfaces for the remaining model performance metrics for simulation set S10, reproduced in Appendix C, also support this conclusion. Briefly, Fig. C1 clearly demonstrates the very poor performance of the 19×19-UNREG method in terms of model error. The uniformity and accuracy of the estimated photometric scale factor as a function of FWHM and S/N is also important for obtaining time series photometry that is free from systematic errors, and the best methods in this respect are 19×19-UNREG and 19×19-GIC$_{\rm P}$, followed closely by IKDA-AIC$_{\rm C}$, IKDA-TIC, and 19×19-BIC$_{\rm P}$ (Fig. C2). In fact, the MPB surfaces (Fig. 5) show considerable correlation with the $P$ surfaces, which further highlights the importance of obtaining a precise and unbiased estimate of the photometric scale factor in order to obtain precise and unbiased photometry (Bramich et al. 2015). For the MFB surfaces (Fig. C3), which also correlate somewhat with the MPB surfaces, the most uniform and least biased methods are 19×19-UNREG and 19×19-GIC$_{\rm P}$, followed closely by IKDA-AIC$_{\rm C}$, IKDA-TIC, and 19×19-BIC$_{\rm P}$. The MFV surfaces in Fig. C4 further demonstrate the target-image overfitting by the 19×19-UNREG and 19×19-GIC$_{\rm P}$ methods.

Finally, we highlight the fact that all 19 of the kernel solution methods that we have tested show complicated (and different) functional dependences of the model performance metrics on PSF FWHM and S/N in each of the reference and target images. This has made it far more difficult to identify the best performing methods than was originally anticipated.

### 5.6 Optimal values of λ

Be12 recommend values of λ between 0.1 and 1 for regularized 19 × 19-pixel kernels although they caution that the optimal value will be PSF FWHM and S/N dependent. Four of the kernel solution methods that we have tested employ regularized DBFs (CKDA-GIC$_{\rm P}$, CKDA-BIC$_{\rm P}$, 19×19-GIC$_{\rm P,}$ and 19×19-BIC$_{\rm P}$) where the optimal value of lambda is selected using either the GIC$_{\rm P}$ or BIC$_{\rm P}$ criteria. In Fig. 7, for each of these four kernel solution methods, and for each target-image S/N regime, we plot surfaces representing the log of the median λ values for both simulation sets S1 (left-hand side) and S10 (right-hand side) as a function of the reference image and kernel FWHM. A circular smoothing region of radius 0.33 pix, which encompasses the results from ∼2000 simulations, is used to calculate the log (λ) surface values.

These surfaces clearly show that the optimal value of λ, when selected by either the GIC$_{\rm P}$ or BIC$_{\rm P}$ criteria, is highly correlated with the PSF FWHM and S/N in each of the reference and target images, and that it is further dependent on the kernel design algorithm employed (i.e. each plot panel in Fig. 7 shows a different non-uniform surface). The BIC$_{\rm P}$ criterion selects values of λ that are greater than those selected by the GIC$_{\rm P}$ criterion (i.e. the BIC$_{\rm P}$ criterion favours stronger kernel regularization), and the GIC$_{\rm P}$ or BIC$_{\rm P}$ criteria rarely select values of λ that are greater than 10 or 100, respectively. The general trends for λ are that higher S/N reference images require weaker kernel regularization, higher S/N target images require weaker regularization for $f_{\rm ker} < 2.35$ pix but stronger regularization for $f_{\rm ker} > 2.35$ pix, and the larger the values of $f_{\rm ref}$ and $f_{\rm ker}$, the stronger the required regularization. We conclude that the optimal regularization of the kernel for any particular kernel solution method is highly data set dependent and that it should be determined independently for each target image. We caution that the optimal value of λ, at least according to the GIC$_{\rm P}$ or BIC$_{\rm P}$ criteria, may lie in a very large range $0\leq\lambda\leq$ 100–1000.

### 5.7 IKDA: number of DBFs

In Section 5.5, we concluded that the IKDA-AIC$_{\rm C}$ and IKDA-TIC methods are equally the best kernel solution methods in terms of photometric accuracy. Therefore we are interested in the properties of the kernel designs that they generate, one of which is the number of DBFs $N_\kappa$ that are selected. In Fig. 8, for each of these two kernel solution methods, and for each target-image S/N regime, we plot surfaces representing the median $N_\kappa$ values for both simulation sets S1 (left-hand side) and S10 (right-hand side) as a function of the reference image and kernel FWHM. As usual, a circular smoothing region of radius 0.33 pix is used to calculate the $N_\kappa$ surface values.

These surfaces show a reasonably complicated dependence on PSF FWHM and S/N in each of the reference and target images, although they are very similar between the two kernel solution methods. The general trends for $N_\kappa$ are that higher S/N reference images require less DBFs, higher S/N target images show larger variations in $N_\kappa$ as a function of $f_{\rm ref}$ and $f_{\rm ker}$, and that all of the surfaces show a minimum trough of approximately the same shape at a similar position. Similar to λ, we find that the number of

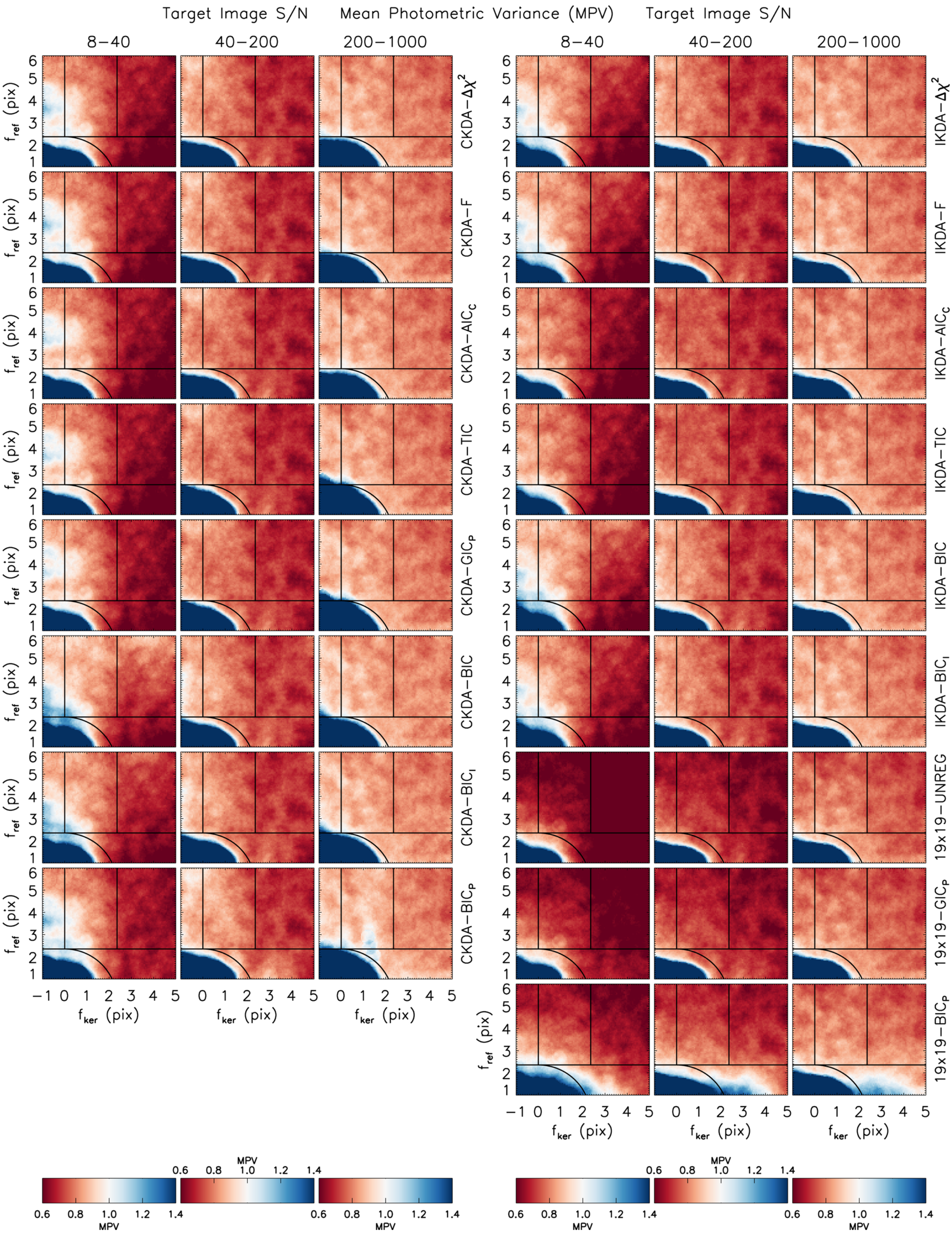


**Figure 6.** Plots of surfaces representing the MPV measure (equation 64) for simulation set S10 as a function of the reference image and kernel FWHM. The format of the figure is the same as in Fig. 5.

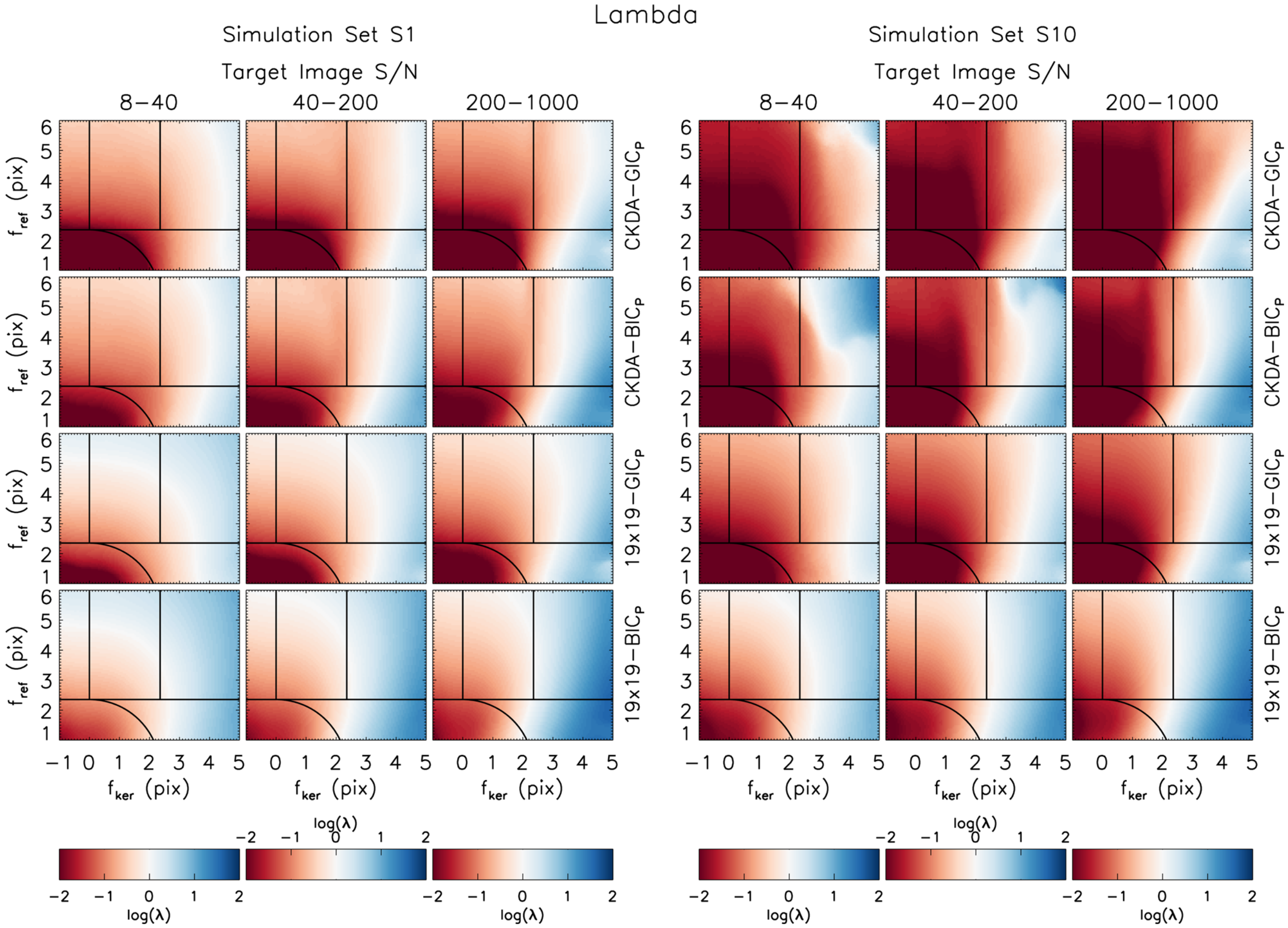


**Figure 7.** Plots of surfaces representing the log of the median λ values for simulation sets S1 (left-hand side) and S10 (right-hand side) as a function of the reference image and kernel FWHM. Each plot corresponds to a specific kernel solution method (labelled on the right-hand side of each row of three plots) and target-image S/N regime (labelled at the top of each column of plots). Otherwise, the format of the figure is the same as in Fig. 5.

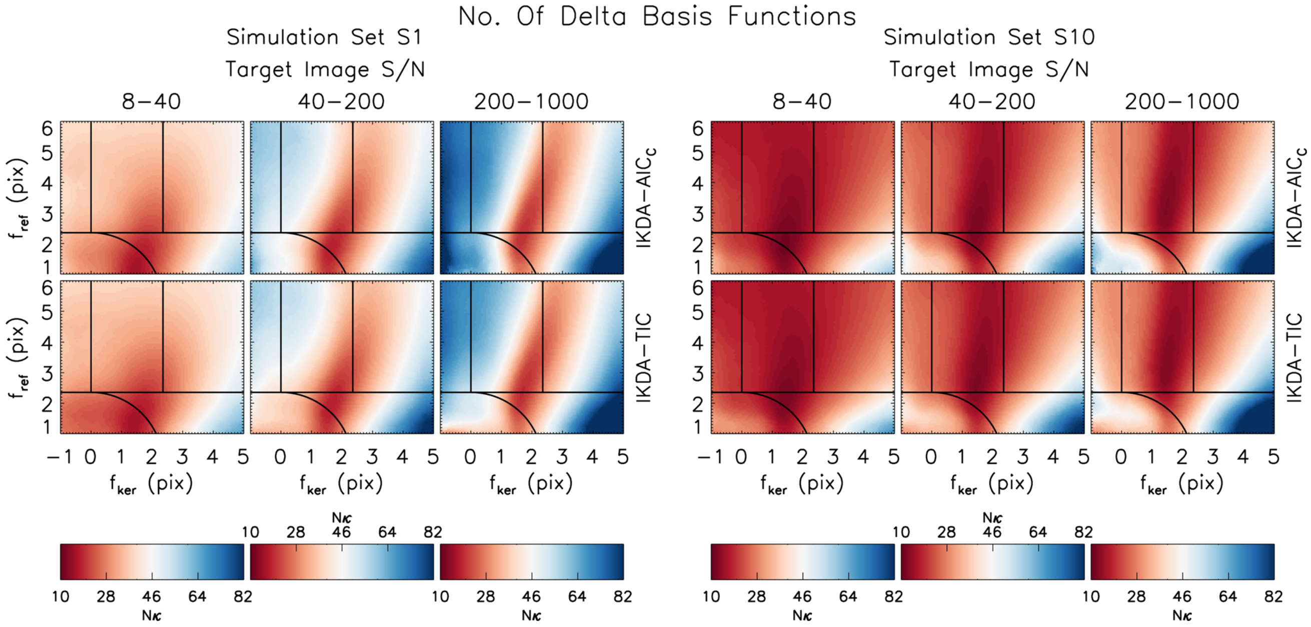


**Figure 8.** Plots of surfaces representing the median number of selected DBFs for simulation sets S1 (left-hand side) and S10 (right-hand side) as a function of the reference image and kernel FWHM. Each plot corresponds to a specific kernel solution method (labelled on the right-hand side of each row of three plots) and target-image S/N regime (labelled at the top of each column of plots). Otherwise, the format of the figure is the same as in Fig. 5.

selected DBFs is highly data set dependent and varies over a large range ($10 \lesssim N_\kappa \lesssim 150$).

## 6 APPLICATION TO REAL IMAGES

So far we have employed simulated image data to explore the performance of the proposed kernel solution methods for a wide range of reference and target image properties, and we have identified the best performing methods in terms of the photometry that they yield (IKDA-AIC$_\mathrm{C}$, IKDA-TIC, and 19×19-GIC$_\mathrm{P}$). However, the task remains to test the kernel solution methods on real image data to check the validity of the conclusions from the simulations. While it is not possible to cover the full range of reference and target image properties using real data in the same systematic and uniform way as it is possible to do with the simulations, we may certainly use real data to validate the results of the simulations for the small ranges of image properties that they cover. For this purpose, we will use two independent data sets.

### 6.1 Time-series observations of the open cluster NGC 7789

#### *6.1.1 Data and reductions*

The first data set comes from a transit survey of the open cluster NGC 7789 (Bramich et al. 2005). The data were observed using the Wide Field Camera on the 2.5-m Isaac Newton Telescope (INT) of the Observatorio del Roque de los Muchachos, La Palma, Canary Islands. We selected the data from chip 2 (2048 × 4096 pix; 0.33 arcsec pix$^{-1}$) for the 11 nights of observations taken between 2000 September 10 and 20. The exposure time was 300 s for each of the 691 selected images, and each image covers the same field in NGC 7789. The images had already been calibrated (i.e. bias subtraction and flat-fielding), and the readout noise and gain determined as $\sigma_0 = 3.1$ ADU and $G = 1.44$ e$^-$/ADU, in the work of Bramich et al. (2005).

We used the DanDIA[7] pipeline (Bramich et al. 2011) to create a high-S/N stacked reference image and an associated star list. First, stars were detected on, and matched between, the 13 best-seeing images (observed during an ∼2.8 h window on the 2000 September 10). Using the matching stars, a set of linear transformations were derived between the images, and each image was registered to the pixel grid of the first image using cubic O-MOMS resampling (Blu, Thévenaz & Unser 2001). The stacked reference image was then created by summing the registered images and dividing by 13. The PSF FWHM of the reference image was measured to be $f_\mathrm{ref}$ ∼3.44 pix.

Secondly, we measured the fluxes and positions of the stars in the reference image by extracting a spatially variable empirical PSF with polynomial degree 3 from the image and fitting this PSF to each detected object. Deblending of very close objects was attempted. From this analysis, we derived a list of 7604 stars with known fluxes and positions in the reference image.

#### *6.1.2 Applying the kernel solution methods*

We measured the PSF FWHM $f_\mathrm{tar}$ of each image and retained only those 587 images such that $-1 < f_\mathrm{ker} < 5$ (see equation 55) in order to match the range for $f_\mathrm{ker}$ employed in our simulations. We then selected 250 random stars from the reference image star list avoiding stars within 200 pixels of the image edges. The selected stars are approximately uniformly distributed across the image area and cover the range of brightest to faintest detected stars. For each star, we cut a 141 × 141 pixel region from the (parent) reference image, and a 101 × 101 pixel region from each of the 587 (parent) target images in the time series, such that the star in question lies at the centre of each region. This effectively registers each target image region with the relevant reference image region to the nearest integer pixel without performing image resampling. This process yielded 146 750 reference and target image pairs (along with associated bad pixel masks from the data reduction).

The S/N for each target image region was calculated as follows. The target-image pixel values were used in place of the $M_{ij}$ in equation (13) and the $\sigma_{ij}$ were calculated using the known readout noise, gain, and flat-field image from the data reduction. We then calculated SN$_\mathrm{tar}$ from equation (58) using the target-image pixel values in place of the $I_{\mathrm{noiseless},\, ij}$, the previously computed $\sigma_{ij}$ values in place of the $\sigma_{\mathrm{in,\,tar},\, ij}$, and the estimate of the local sky background level from the data reduction in place of $S_\mathrm{tar}$.

To flag any outlier pixel values in the target images, we first ran the 19×19-GIC$_\mathrm{P}$ method on each reference and target image pair without applying sigma-clipping. The model image fit and its noise model were used to clip pixel values with $|\varepsilon_{ij}| \geq 4$ and the corresponding pixels were included in the bad pixel mask for the target image. Then, for each kernel solution method, we computed kernel and differential background solutions for all of the reference and target image pairs, ignoring bad pixels and without using sigma-clipping. In all cases, we used three iterations for each solution. The optimization of $\lambda$ for the GIC$_\mathrm{P}$ and BIC$_\mathrm{P}$ model selection criteria was performed in the same way as for the simulations (see Section 5.3).

We are unable to assess the performance of each kernel solution method with regards to model error since the true model image is unknown for real data. Hence we do not calculate the MSE metric. However, the photometric scale factor for each solution may be compared on a relative scale. For each of the 587 parent images, we compute the median value of $P$ from the 19×19-GIC$_\mathrm{P}$ fits for the 250 corresponding target images, and, for comparison purposes only, we use these median $P$ values to normalize the values of $P$ estimated by each kernel solution method for each target image. Hence the median of the normalized $P$ values for each parent image for the 19×19-GIC$_\mathrm{P}$ method is always unity.

The remaining model performance metrics require a reliable noise model for their computation. For the simulations, this noise model was precisely known since it was used to generate the simulated data. However, for the real data, we may only estimate the noise in each pixel. For a fair comparison between the different kernel solution methods for each reference and target image pair, we employ a single noise model corresponding to the 19×19-GIC$_\mathrm{P}$ fit when calculating the model performance metrics. In other words, we use the pixel values from the 19×19-GIC$_\mathrm{P}$ model image fit in equation (13) to calculate the $\sigma_{ij}$ values which we then use in place of the $\sigma_{\mathrm{in,\,tar},\, ij}$ in equations (60) and (61) for the purpose of calculating the MFB and MFV metrics.

To assess the photometric accuracy of each kernel solution method, we again perform PSF fitting on the difference image at the position of the central star in the reference image region (which is one of the 250 stars selected randomly from the reference image star list). The method is the same as that used to perform PSF fitting in the simulations, except that we employ the empirical PSF at the measured star position on the reference image determined by shifting the empirical PSF model corresponding to the nearest pixel

[7] DanDIA is built from the DanIDL library of IDL routines available at http://www.danidl.co.uk.

by the appropriate sub-pixel shift using cubic O-MOMS resampling. The noise model used for the PSF fitting is the noise model corresponding to the model image fit for the kernel solution method under consideration. The computation of the MPB and MPV measures also requires the calculation of a reasonable and consistent normalization factor for the difference fluxes. Again we use an estimate of the theoretical minimum variance $\sigma^2_{\mathrm{min}}$ in the difference flux $\mathcal{F}_{\mathrm{meas}}$ on the photometric scale of the reference image. We calculate $\sigma^2_{\mathrm{min}}$ from equation (62) by setting $\mathcal{P}_{\mathrm{tar}}$ to the empirical PSF for the star convolved with the 19×19-GIC$_{\mathrm{P}}$ kernel solution and normalized to a sum of unity, and by again setting the $\sigma_{\mathrm{in,\,tar,}\,ij}$ to the $\sigma_{ij}$ values obtained using the 19×19-GIC$_{\mathrm{P}}$ model image fit. Finally, we take the appropriate normalization factor calculated earlier for the photometric scale factors (which also use the 19×19-GIC$_{\mathrm{P}}$ fits) as the best available estimate of the true photometric scale factor $P_{\mathrm{true}}$. The MPB and MPV measures are then calculated for a set of $N_{\mathrm{set}}$ flux measurements $\mathcal{F}_{\mathrm{meas},k}$ indexed by $k$ via equations (63) and (64).

### *6.1.3 Results and discussion*

In Fig. 9, for each kernel solution method we plot the median $P$, MFB, and MFV values, and the MPB and MPV measures, as a function of $f_{\mathrm{ker}}$ (determined from $f_{\mathrm{ref}}$ and $f_{\mathrm{tar}}$ using equation 55) for the results from the target images with SN$_{\mathrm{tar}} > 40$. The data are binned in $f_{\mathrm{ker}}$ with bins of size 0.2 pix. The results for the main methods of interest are plotted with coloured filled circles (light brown for IKDA-AIC$_{\mathrm{C}}$; green for IKDA-TIC; blue for 19×19-GIC$_{\mathrm{P}}$; red for 19×19-UNREG) such that the area of each circle is proportional to the number of data values used in the estimation of the central value. The largest circle corresponds to 9793 data values. The results for the remaining kernel solution methods are plotted with black dots. Note that the values of $P$ displayed in the top-left plot panel have been normalized using the results from the 19×19-GIC$_{\mathrm{P}}$ method as described in the previous section. We also produce a similar style plot for the median $\lambda$ values for the 19×19-GIC$_{\mathrm{P}}$ method in the top-right plot panel.

To facilitate a comparison with our simulations, we also plot curves in each panel for the IKDA-AIC$_{\mathrm{C}}$ (light brown), IKDA-TIC (green), 19×19-GIC$_{\mathrm{P}}$ (blue), and 19×19-UNREG (red) methods representing the results from simulation set S10 (high-S/N reference image) for the relevant model performance metric as a function of $f_{\mathrm{ker}}$. Each point on a curve is calculated as the median value (for $P$, $\lambda$, MFB, and MFV) of the results from ∼4000 simulations, or via equations (63) and (64) (for MPB and MPV), by using a smoothing radius of 0.33 pix in the ($f_{\mathrm{ker}}, f_{\mathrm{ref}}$) plane ($f_{\mathrm{ref}} = 3.44$ pix in this case) and considering only SN$_{\mathrm{tar}} > 40$. For an easier comparison to the results from the real data, the curves for the MFV and MPV metrics have been scaled by factors of 1.045 and 1.7, respectively.

The plots in Fig. 9 nicely demonstrate that the results for the real data follow the same patterns as those for the simulations. For instance, the results for the IKDA-AIC$_{\mathrm{C}}$ and IKDA-TIC methods are virtually indistinguishable, and the order of the methods from high to low values for each of the five model performance metrics are the same (e.g. MPB goes blue–red–brown/green for both the points and the curves). Furthermore, the trends in the results for the real data as a function of $f_{\mathrm{ker}}$ are similar to those seen in the simulation results (e.g. see $P$ and MPB) even if the actual values do not match up so closely (e.g. see $P$ for the two IKDA methods). Finally, our recommended kernel solution methods from Section 5.5 outperform the remaining methods (except 19×19-UNREG) with respect to each of the five model performance metrics. These are comforting results given the fact that the simulated data are generated based on approximations to the properties of real data (e.g. adopting circularly symmetric Gaussian PSF profiles in the simulations when real data exhibit PSFs that deviate from circular symmetry and Gaussian functions).

A few points about the plots in Fig. 9 are worth considering in more detail. We find that $P$ is equally underestimated for these real data by ∼1 per cent for the two IKDA methods relative to the 19×19-GIC$_{\mathrm{P}}$ and 19×19-UNREG methods, although we do not know which method provides the best estimate of the true value of $P$ for the real data. Also, this fact does not seems to have had a detrimental impact on the photometric bias for the two IKDA methods. The median MFB values for the real data do not follow the shape of the MFB curves derived from the simulations, but their absolute values are generally even smaller than those from the simulations ($<2 \times 10^{-4}\sigma_{\mathrm{min}}$). The scaling of the simulation results to match the results from the real data for the MFV metric is most likely necessary because the pixel uncertainties are somewhat underestimated for the real data due to unmodelled sources of error (e.g. flat-field errors, error in the gain, errors in the empirical PSF, etc.). However, for the MPV metric, the scale factor between the simulation results and those from the real data is much larger, and we suggest an alternative explanation for this below. Once the simulation results for the MFV and MPV metrics have been scaled, they match very satisfactorily with the results from the real data.

Focusing on the results for the real data with regards to photometric accuracy, we find that the gradients in the MPB measure as a function of $f_{\mathrm{ker}}$ for the IKDA-AIC$_{\mathrm{C}}$, IKDA-TIC, 19×19-GIC$_{\mathrm{P}}$, and 19×19-UNREG methods are very similar and cover a range of $\sim 0.40\sigma_{\mathrm{min}}$. However, the MPB values are closest to zero for the two IKDA methods indicating a smaller photometric bias. None of the remaining methods perform as well as these four methods in terms of MPB. In Fig. 10, we plot the MPV measure calculated for each star light curve (i.e. from 587 photometric measurements in each case) as a function of the ratio of the star flux to the total object flux within the target image region. The quantity on the $x$-axis ($f_{\mathrm{frac}}$) indicates by how much the flux from the star on which the photometry is performed dominates the total object flux, and hence the kernel solution, in the target image region. For $f_{\mathrm{frac}} \gtrsim 0.2$, it is clear that the four methods under consideration tend to overfit the real data in the same manner as we found for the simulations, with the 19×19-UNREG and 19×19-GIC$_{\mathrm{P}}$ methods doing the most overfitting. For $f_{\mathrm{frac}} \lesssim 0.2$, the MPV values scatter nicely around ∼1.2 for the IKDA-AIC$_{\mathrm{C}}$ and IKDA-TIC methods. This plot also explains the large scale factor for the MPV metric between the results for the simulations and those for the real data if one considers that ∼70 per cent of the simulations have $f_{\mathrm{frac}} > 0.1$ (where overfitting mainly occurs), compared to only ∼24 per cent of the reference and target image pairs for the real data. Referring back to the MPV plot panel in Fig. 9, if we ignore the 19×19-UNREG and 19×19-GIC$_{\mathrm{P}}$ methods because of their excessive overfitting, then we can see that the IKDA-AIC$_{\mathrm{C}}$ and IKDA-TIC methods consistently attain the best MPV values. Hence our conclusions from the simulations are fully validated by the application of the kernel solution methods to the INT image data; namely that the IKDA-AIC$_{\mathrm{C}}$ and IKDA-TIC methods are equally the best in terms of the photometry that they yield, and that the 19×19-GIC$_{\mathrm{P}}$ method is a close second best.

Finally, it is interesting to note that the values of $\lambda$ selected by the 19×19-GIC$_{\mathrm{P}}$ method for the real data are ∼10 times smaller than the values selected for the simulations, while the variation of $\lambda$ as a function of $f_{\mathrm{ker}}$ has the same form.

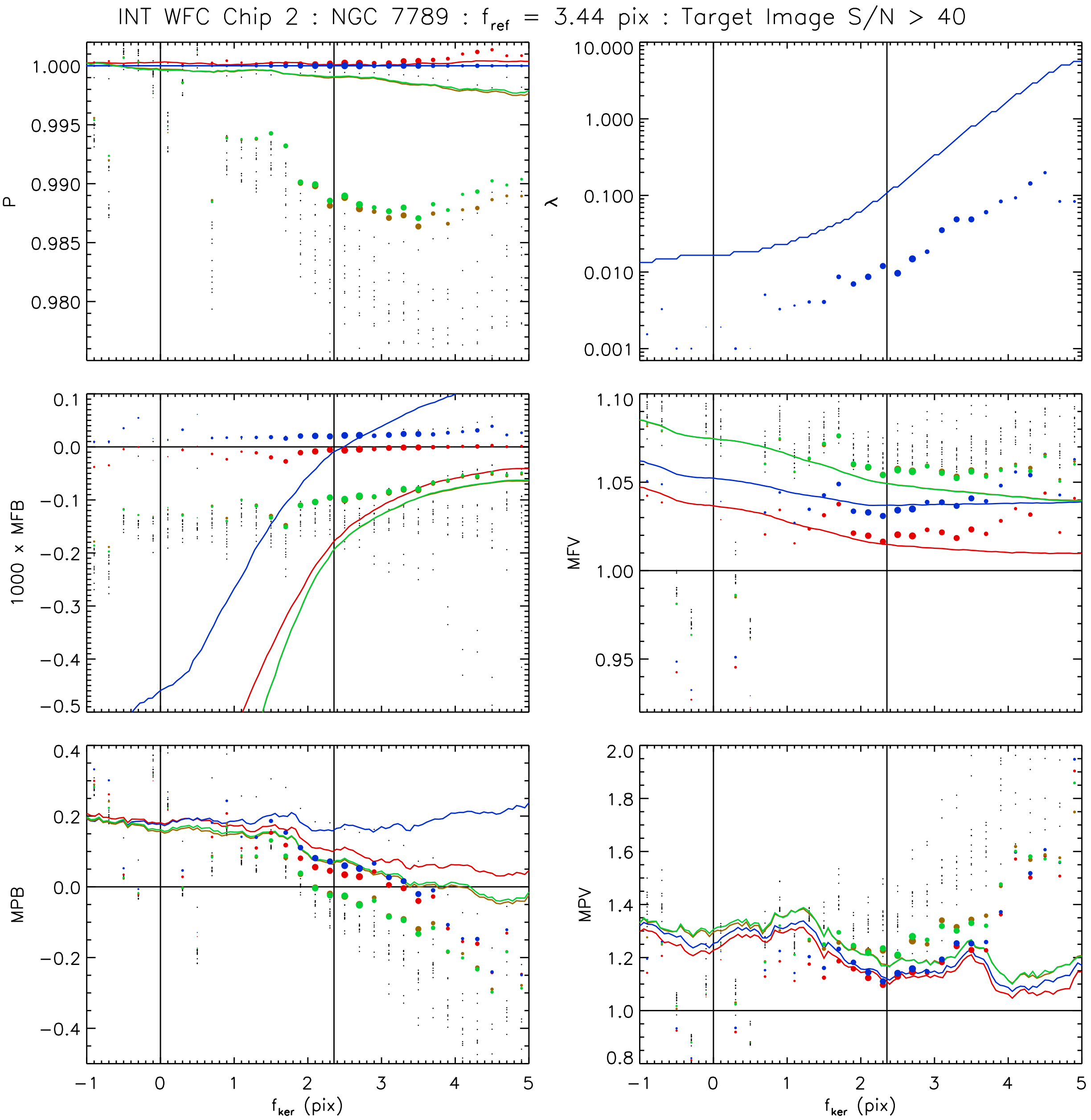


**Figure 9.** Plots of the median $P$, MFB, and MFV values (equations 12, 60, and 61), and the MPB and MPV measures (equations 63 and 64), for each kernel solution method as a function of $f_{\rm ker}$. The plots correspond to the results for the INT target images with $\mathrm{SN_{tar}} > 40$. The data are binned in $f_{\rm ker}$ with bins of size 0.2 pix. Coloured filled circles correspond to the IKDA-AIC$_\mathrm{C}$ (light brown), IKDA-TIC (green), 19×19-GIC$_\mathrm{P}$ (blue) and 19×19-UNREG (red) methods such that the area of each circle is proportional to the number of data values used in the estimation of the central value. The largest circle corresponds to 9793 data values. Black dots represent the results for the remaining kernel solution methods. The values of $P$ displayed in the top-left panel have been normalized using the results from the 19×19-GIC$_\mathrm{P}$ method as described in Section 6.1.2. The median λ values for the 19×19-GIC$_\mathrm{P}$ method are plotted in the top-right panel. The curves correspond to the results from simulation set S10 (see text in Section 6.1.3 for details).

### 6.2 Time-series observations of a Galactic field

#### 6.2.1 Data and reductions

The second data set comes from a commissioning run for the Qatar Exoplanet Survey (QES; Alsubai et al. 2013). The data were observed using camera 5 of the second QES observing station at the New Mexico Skies observatory, New Mexico, USA. We selected a block of 27 images (4096 × 4096 pix; 3.1 arcsec pix$^{-1}$) of the same field (RA∼14 h; Dec.∼0 deg) taken on the night of 2015 May 11 during an ∼2.2 h period. Each image has an exposure time of 30 s.

We used the `DanDIA` pipeline to calibrate the images (bias level subtraction and flat-fielding) and to measure the chip readout noise (∼7.85 ADU) and gain (∼1.65 e$^-$/ADU). We also used the pipeline to create a high-S/N stacked reference image from a block of ten 30 s images of the same field taken later during the night and to produce an associated star list with 84 069 stars (see Section 6.1.1 for the

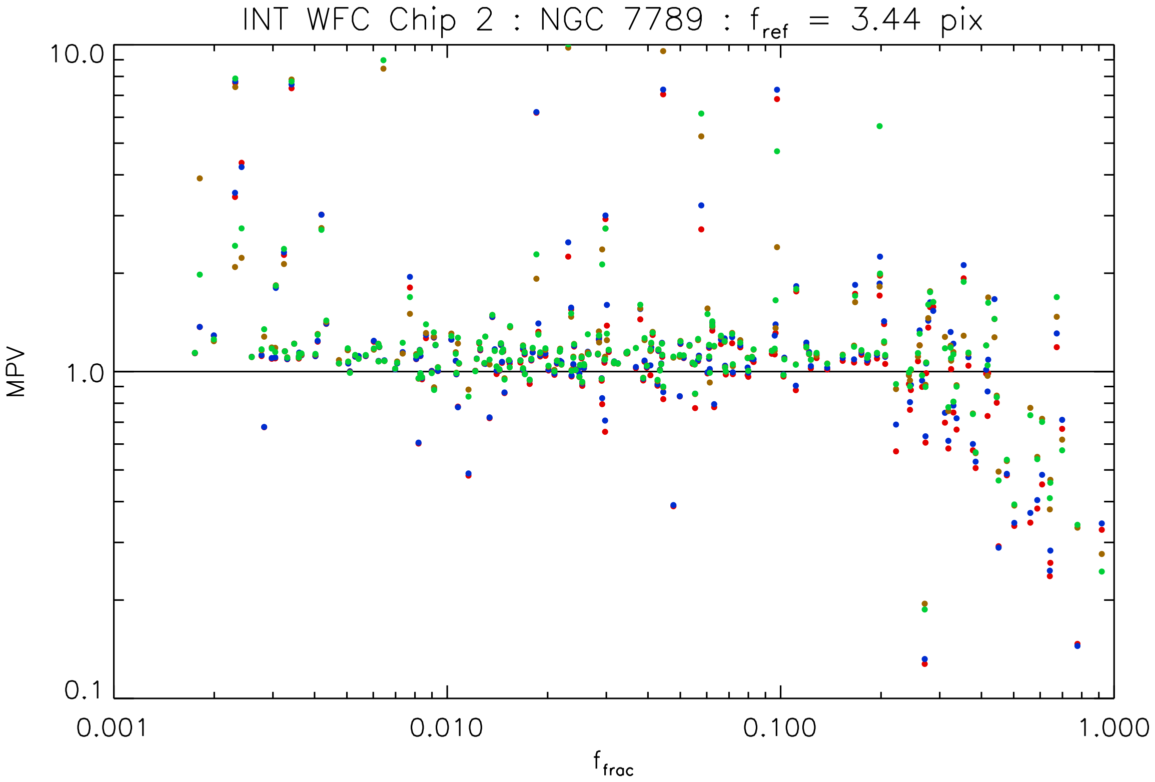


**Figure 10.** Plot of the MPV measure (equation 64) for each star light curve as a function of the ratio of the star flux to the total object flux within the target image region ($f_{\mathrm{frac}}$). Coloured filled circles correspond to the IKDA-AIC$_{\mathrm{C}}$ (light brown), IKDA-TIC (green), 19×19-GIC$_{\mathrm{P}}$ (blue) and 19×19-UNREG (red) methods. This plot corresponds to the results for the INT target images.

method). The PSF FWHM of the reference image was measured to be $f_{\mathrm{ref}} \sim 2.70$ pix.

We selected 1000 random stars from the reference image star list avoiding stars within 200 pixels of the image edges. Following the same steps as those used for the INT data, we created 1000 reference and target image pairs for each of the 27 parent target images in the time series, yielding 27 000 image pairs in total. The procedures described in Section 6.1.2 for the INT data were then applied to the QES data to compute the kernel and differential background solutions for each kernel solution method, and to calculate the model performance metrics.

### 6.2.2 *Results and discussion*

The results for the QES data are plotted in Fig. 11, which has been constructed in exactly the same way as Fig. 9 for the INT data. Note that the QES data are more limited in that they only cover the range $-0.4 < f_{\mathrm{ker}} < 1.8$ pix. In this case, the curves for the MFV and MPV metrics have been scaled by factors of 1.045 and 1.5, respectively.

The conclusions for the QES data are the same as those for the INT data. The few exceptions for these data are that $P$ is only underestimated by up to ∼0.2 per cent for the IKDA-AIC$_{\mathrm{C}}$ and IKDA-TIC methods relative to the 19×19-GIC$_{\mathrm{P}}$ and 19×19-UNREG methods, the absolute values of the MFB metric are larger, and there is more scatter in the MPB and MPV measures (probably because fewer data have been analysed). Therefore the results for the QES data add a further independent validation of the results from our simulations.

## 7 CONCLUSIONS AND RECOMMENDATIONS

The key achievement in this work is the elaboration of a framework for automatically constructing a kernel model (or, equivalently, a model image) for DIA where the user is only required to specify very few external parameters to control the kernel design (e.g. the maximum extent of the kernel). The framework requires the definition of a kernel solution method that consists of two components; namely, a kernel design algorithm to generate a set of candidate kernel models, and a model selection criterion to select the simplest kernel model from the candidate models that provides a sufficiently good fit to the target image (i.e. an implementation of the Principle of Parsimony). The framework also requires the definition of an appropriate detector noise model with associated parameters such as readout noise and gain. It is crucial that this noise model is accurate since the model selection criteria depend heavily on the pixel uncertainties provided by the noise model. We developed and tested 18 automatic kernel solution methods using comprehensive image simulations and real data, and we compared their performance to that of a fixed unregularized kernel design (i.e. the 19×19-UNREG method).

The main conclusion from the image simulations (Section 5) is that the IKDA-AIC$_{\mathrm{C}}$ and IKDA-TIC methods are equally the best kernel solution methods in terms of photometric accuracy. The 19×19-GIC$_{\mathrm{P}}$ method also performs very well and is a good second choice. This conclusion is also supported by considering the performance of these methods with regards to model error and fit quality. The 19×19-UNREG method gives excellent estimates of the photometric scale factor (Fig. C2) with what appear to be some of the

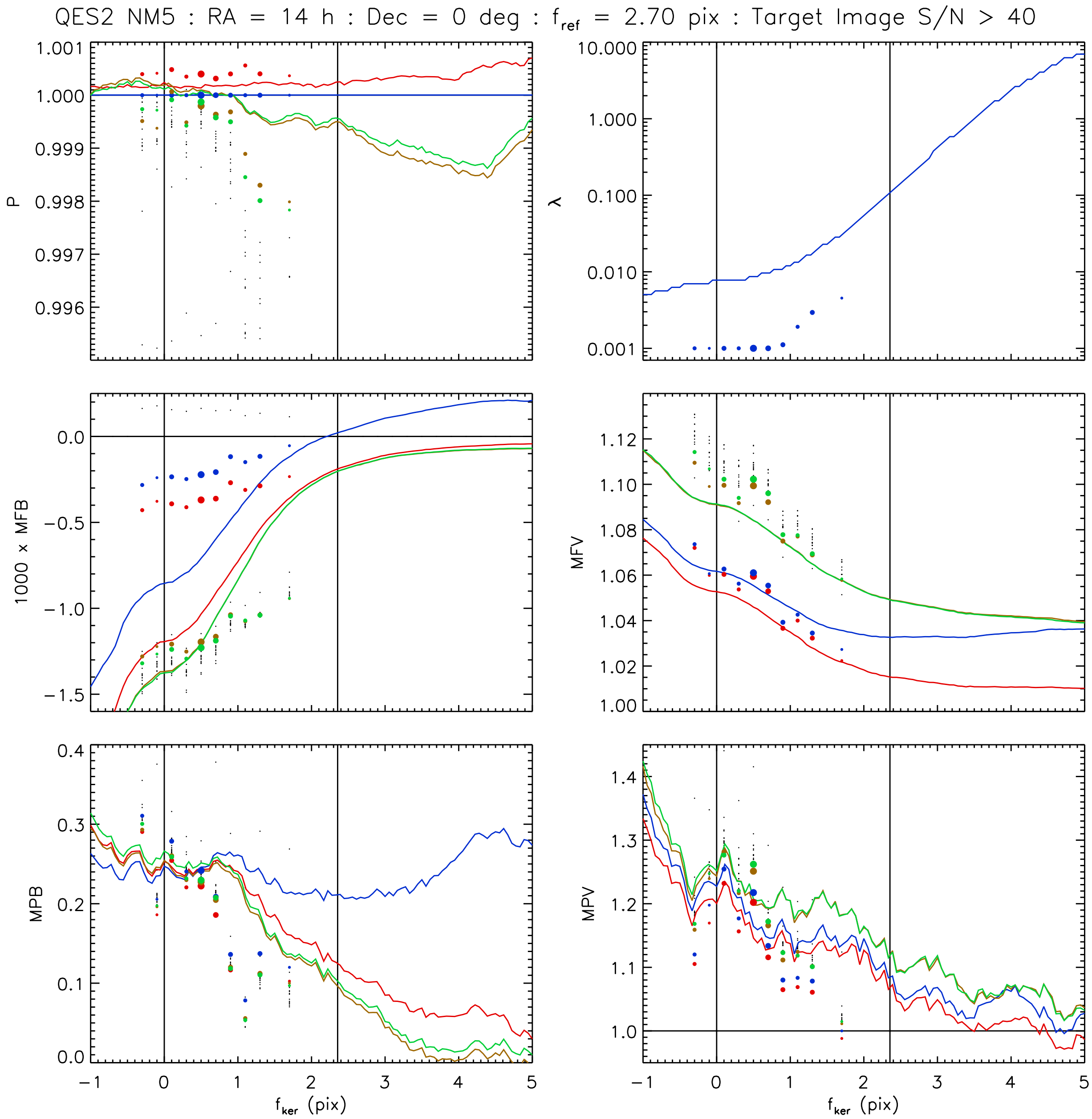


**Figure 11.** Same as Fig. 9 for the QES data. The largest circle corresponds to 5273 data values.

most uniform and least-biased model image fits (Fig. C3) and PSF photometry (Fig. 5). However, this is somewhat misleading since the model performance metrics that measure the fit variance (Fig. C4) and the photometric variance (Fig. 6) reveal that this method is the worst offender for overfitting the target image. Hence we concur with the findings from Be12 that the unregularized 19×19-pixel kernel design brings too many parameters to the model image. Moreover, we have shown that kernel regularization (via Tikhonov regularization) is not the only way, or even necessarily the best way, to control the overfitting. The IKDA-AIC$_C$ and IKDA-TIC methods achieve a better performance than the regularized kernel designs via a parsimonious choice of unregularized DBFs. Taking this further by combining the IKDA with regularization (i.e. the IKDA-GIC$_P$ and IKDA-BIC$_P$ methods) was unfortunately not possible as we found that the corresponding processing time was prohibitive. From the simulations, we also discovered that the AIC-type model selection criteria work better than the BIC-type criteria for DIA, which we explain by considering that the true model image is not included in the set of candidate model images generated by our kernel design algorithms.

We also analysed two independent sets of real image data covering different regions in the reference and target image PSF FWHM and S/N parameter space (Section 6). The results for the real data were found to follow the same patterns and trends as the results from the simulations. Most importantly, the IKDA-AIC$_C$, IKDA-TIC, and 19×19-GIC$_P$ methods were also found to be the best kernel solution methods in terms of photometric accuracy for the real data.

In practical terms, the $\mathrm{AIC_C}$ model selection criterion is much easier to implement (trivial in fact), and various orders of magnitude faster to calculate, than the TIC or $\mathrm{GIC_P}$ criteria. Since the IKDA-$\mathrm{AIC_C}$ and IKDA-TIC methods yield virtually the same results, it is clear that the IKDA-$\mathrm{AIC_C}$ method is the most desirable of the two for implementation. However, the IKDA can be somewhat slower to run than the optimization of $\mathrm{GIC_P}$ over $\lambda$ for the fixed 19×19-pixel kernel design. This is especially true when the IKDA attempts to grow a kernel model with many DBFs. Hence, if processing time is a concern, then the 19×19-$\mathrm{GIC_P}$ method may be more desirable than the IKDA-$\mathrm{AIC_C}$ method, even if the results are slightly less optimal. One caveat of the 19×19-$\mathrm{GIC_P}$ method is that a 19×19-pixel grid may not be large enough for an adequate kernel solution for some DIA problems, and its size may therefore needs to be increased as appropriate.

Our work constitutes the first fully systematic and comprehensive attempt to characterize the performance of DIA as a function of the reference and target image properties using simulated images and with validation on real data. We have learned some important facts from these experiments which may be translated into the following recommendations.

(i) It is vastly advantageous to use a reference image with higher S/N than that of the target image regardless of the image properties or the kernel solution method. We therefore recommend that the reference image is constructed either by exposing for longer than the target image(s), or by stacking a set of registered images to achieve a longer effective exposure time, while at the same time maintaining the requirement that the reference image has a PSF FWHM that is among the smallest PSF FWHMs of the target image(s). We realize that this particular advice is already followed for most DIA reductions. However, it is comforting to see that our comprehensive simulation work strongly supports this approach.

(ii) In general, the photometric scale factor between the reference and target image is underestimated. However, this effect is smaller for higher S/N target images. Given the importance of obtaining an accurate estimate of the photometric scale factor for accurate photometry, we recommend employing all of the pixels in the target image to solve for the kernel, since this maximizes the S/N of the data that are being fit.

(iii) In most of our simulations, we found that all of the kernel solution methods are overfitting the brightest star(s) and underfitting the faintest star(s) since the brightest star(s) dominate the kernel solution. The effect on the photometry of the brightest star in each target image is to yield variances that are impossibly, and therefore misleadingly, small. We used target images of size 101 × 101 pixels both for the simulations and the tests on the real data. We found that by increasing the size of the target images, this effect on the photometry is mitigated since more bright stars, and therefore more pixels from bright stars, are used to derive the kernel solution. Hence we again recommend employing all of the pixels in the target image to solve for the kernel.

These last two recommendations have important implications for some popular DIA software implementations that generate a spatially varying kernel solution for an image by interpolating a set of spatially invariant kernel solutions determined independently from small image regions called 'stamps' (e.g. ISIS – Alard 2000, HOTPANTS[8]). The stamps are chosen to be approximately uniformly distributed across the image area, centred on isolated bright stars, and only slightly larger than the objects they encompass (e.g. in Be12, the stamp size is ∼57 × 57 pix, with only ∼39 × 39 pix used for the kernel solution). We believe that it will be highly beneficial to modify the stamp selection strategy in these algorithms to match our recommendations. Specifically, image stamps should be defined to be as large as possible without seriously violating the assumption of a spatially invariant kernel model within the stamp, and they should be selected such that each stamp contains a minimum of at least a few bright objects (not necessarily stars, and there is no reason to avoid blended objects). Our tests on real data suggest the following useful rule-of-thumb: the ratio of the flux of the brightest star to the total flux of all objects in a stamp should be less than ∼0.1. It is not surprising that Be12 found that the unregularized 19×19-pixel kernel with 361 parameters was overfitting a target image stamp with just ∼1520 pixels, of which only a small proportion contain signal from the single object[9] (see their section 4). We are confident that if Be12 were to repeat their experiment for a set of image stamps selected following our recommendations, then the unregularized 19×19-pixel kernel would have been found to be overfitting the stamps to a much lesser extent than reported.

Be12 recommend values of $\lambda$ between 0.1 and 1 for regularized 19×19-pixel kernels while cautioning that the 'optimal value of $\lambda$ will be a function of the S/N in the template and science images,... and of the respective seeings in the input images,...'. We have characterized precisely how the optimal value of $\lambda$, as selected by the $\mathrm{GIC_P}$ and $\mathrm{BIC_P}$ criteria, varies as a function of the reference and target image properties for four kernel solution methods (Section 5.6). We find that the optimal value of $\lambda$ is highly correlated with the PSF FWHM and S/N in each of the reference and target images, and that it spans values from $\lambda = 0$ (i.e. no regularization) up to maximum values of the order of $\lambda = 10$ and 100 for $\mathrm{GIC_P}$ and $\mathrm{BIC_P}$, respectively. We conclude that the optimal regularization of the kernel model for any particular kernel solution method is highly data set dependent and that it should be determined *independently* for each target image.

Looking to the future, we can see much potential for the development and testing of new kernel design algorithms within our framework that may perform better than those presented in this work. In fact, we believe that there is still plenty of room for improvement in the kernel solution methods, especially with regards to achieving the best photometry. For example, the poor performance of the CKDA methods, including those employing kernel regularization, was a disappointment. It would be interesting to investigate whether adopting a radial dependence for the strength of the kernel regularization can improve the CKDA performance, since we expect the variations in the true kernel model to be smallest in the outer parts of the kernel. Most intriguingly, the 'spidery' form of the kernels that are generated by the IKDA methods, combined with the fact that they perform exceptionally well, implies that sparsity may be the key to the optimal use of DBFs in DIA. Finally, it remains to extend the methods presented here to the case of a spatially varying convolution kernel.

[8] http://www.astro.washington.edu/users/becker/v2.0/hotpants.html

[9] Be12 found that the normalized residuals in the difference image for their example have a standard deviation of ∼0.79 for the unregularized 19×19-pixel kernel. However, it should be noted that this is perfectly consistent with a reduced chi-squared of unity.

## ACKNOWLEDGEMENTS

I dedicate this work to the twin bright lights in my life – Phoebe and Chloe Bramich Muñiz.

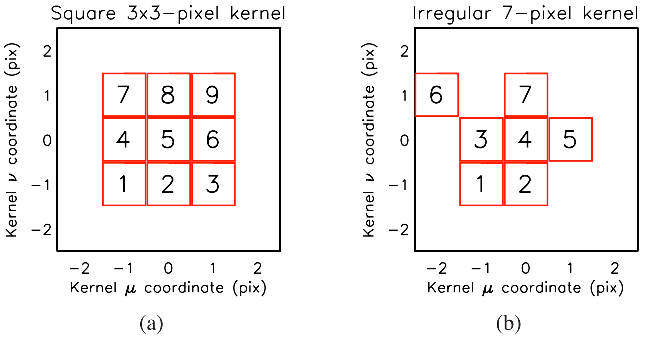
We thank the referee for investing the time to carefully study this paper and for providing useful feedback. This publication was made possible by NPRP grant # X-019-1-006 from the Qatar National Research Fund (a member of Qatar Foundation). The statements made herein are solely the responsibility of the authors. The HPC resources and services used in this work were provided by the IT Research Computing group in Texas A&M University at Qatar. IT Research Computing is funded by the Qatar Foundation for Education, Science and Community Development (http://www.qf.org.qa).

## REFERENCES

Akaike H., 1974, IEEE Trans. Autom. Control, 19, 716
Alard C., 2000, A&AS, 144, 363
Alard C., Lupton R. H., 1998, ApJ, 503, 325
Albrow M. D. et al., 2009, MNRAS, 397, 2099
Alsubia K. A. et al., 2013, Acta Astron., 63, 465
Becker A. C., Homrighausen D., Connolly A. J., Genovese C. R., Owen R., Bickerton S. J., Lupton R. H., 2012, MNRAS, 425, 1341 (Be12)
Blu T., Thévenaz P., Unser M., 2001, IEEE Trans. Image Process., 10, 1069
Bramich D. M., 2008, MNRAS, 386, L77
Bramich D. M. et al., 2005, MNRAS, 359, 1096
Bramich D. M., Figuera Jaimes R., Giridhar S., Arellano Ferro A., 2011, MNRAS, 413, 1275
Bramich D. M. et al., 2013, MNRAS, 428, 2275 (Br13)
Bramich D. M., Bachelet E., Alsubai K. A., Mislis D., Parley N., 2015, A&A, 577, A108
Furnival G., Wilson R., 1974, Technometrics, 16, 499
Golub G. H., Van Loan C. F., 1996, Matrix Computations, 3rd edn. Johns Hopkins Univ. Press, Baltimore, MD
Israel H., Hessman F. V., Schuh S., 2007, Astron. Nachr., 328, 16
Konishi S., Kitagawa G., 1996, Biometrika, 83, 875
Konishi S., Kitagawa G., 2008, Information Criteria and Statistical Modeling. Springer-Verlag, Berlin
Konishi S., Ando T., Imoto S., 2004, Biometrika, 91, 27
Kullback S., Leibler R. A., 1951, Ann. Math. Stat., 22, 79
Miller J. P., Pennypacker C. R., White G. L., 2008, PASP, 120, 449
Press W. H., Teukolsky S. A., Vetterling W. T., Flannery B. P., 2007, Numerical Recipes: The Art of Scientific Computing, 3rd edn. Cambridge Univ. Press, Cambridge
Schwarz G., 1978, Ann. Stat., 6, 461
Sugiura N., 1978, Commun. Stat. - Theory Methods, A7, 13
Takeuchi K., 1976, Math. Sci., 153, 12
Yuan F., Akerlof C. W., 2008, ApJ, 677, 808

## APPENDIX A

We provide two examples of the Laplacian matrix **L**. For the square 3 × 3-pixel kernel design shown in Fig. A1, where the value of $q$ is displayed inside each kernel pixel, we may use equation (20) to obtain the 10 × 10 matrix:


Square 3x3−pixel kernel
Kernel ν coordinate (pix)
2
1
0
−1
−2
7
8
9
4
5
6
1
2
3
−2
−1
0
1
2
Kernel μ coordinate (pix)
Irregular 7−pixel kernel
Kernel ν coordinate (pix)
6
7
3
4
5
1
2
Kernel μ coordinate (pix)

(a) (b)

**Figure A1.** Example configurations of sets of kernel DBFs. Individual DBFs are represented by red squares positioned at the kernel pixel coordinates where they take the value unity. Each red square displays the value of $q$ for the corresponding DBF.

$$\mathbf{L} = \begin{pmatrix} 2 & -1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 3 & -1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 2 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 3 & -1 & 0 & -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & -1 & 4 & -1 & 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 & -1 & 3 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 2 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & -1 & 3 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & -1 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

This matrix is of rank equal to 8, as expected since all of the DBFs are connected to each other. The elements of the last row and column correspond to the differential background parameter and are consequently all zero.

Kernels may of course be of any shape. For the 7-pixel kernel design shown in Fig. A1, equation (20) yields:

$$\mathbf{L} = \begin{pmatrix} 2 & -1 & -1 & 0 & 0 & 0 & 0 & 0 \\ -1 & 2 & 0 & -1 & 0 & 0 & 0 & 0 \\ -1 & 0 & 2 & -1 & 0 & 0 & 0 & 0 \\ 0 & -1 & -1 & 4 & -1 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

This matrix is of rank equal to 5, which is explained by the fact that there are two disconnected sets of connected DBFs in the kernel model.

## APPENDIX B

In this appendix, we present Figs B1–B4 where we plot the median MSE, $P$, MFB, and MFV values, and the MPB and MPV measures, for each kernel solution method for various subsets of our simulations chosen based on image sampling. These plots are referred to briefly in Sections 5.3 and 5.4.

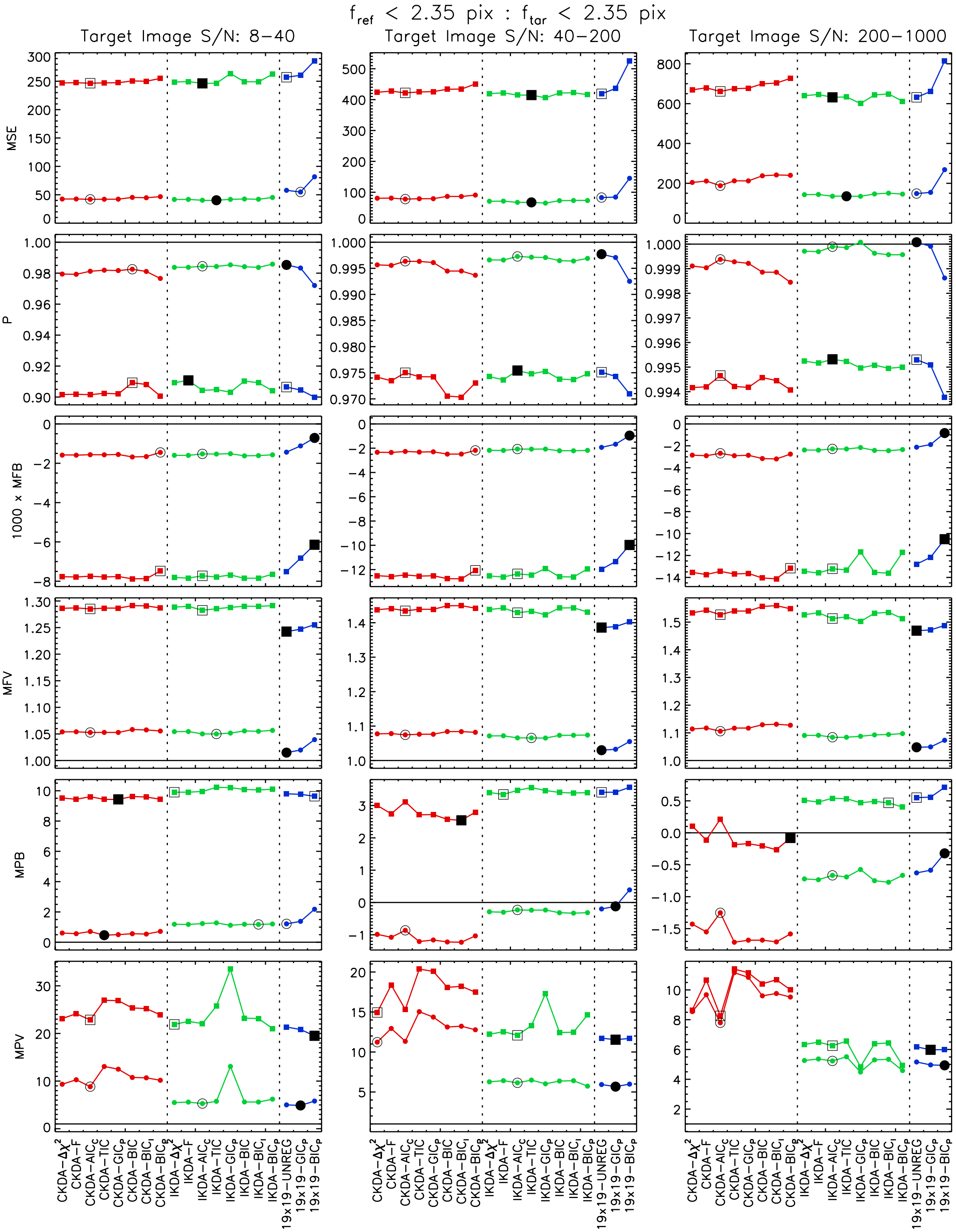


**Figure B1.** Plots of the median MSE, $P$, MFB, and MFV values (equations 59, 12, 60, and 61), and the MPB and MPV measures (equations 63 and 64), for each kernel solution method for $f_{\mathrm{ref}} \leq 2.35$ pix and $f_{\mathrm{tar}} \leq 2.35$ pix. The results in each plot have been calculated from ∼19 000 simulations for each of the simulation sets S1 and S10. The layout, symbols, and colours used are the same as in Fig. 4. The IKDA-GIC$_{\mathrm{P}}$ and IKDA-BIC$_{\mathrm{P}}$ methods are excluded when determining the best values of the relevant model performance metric.

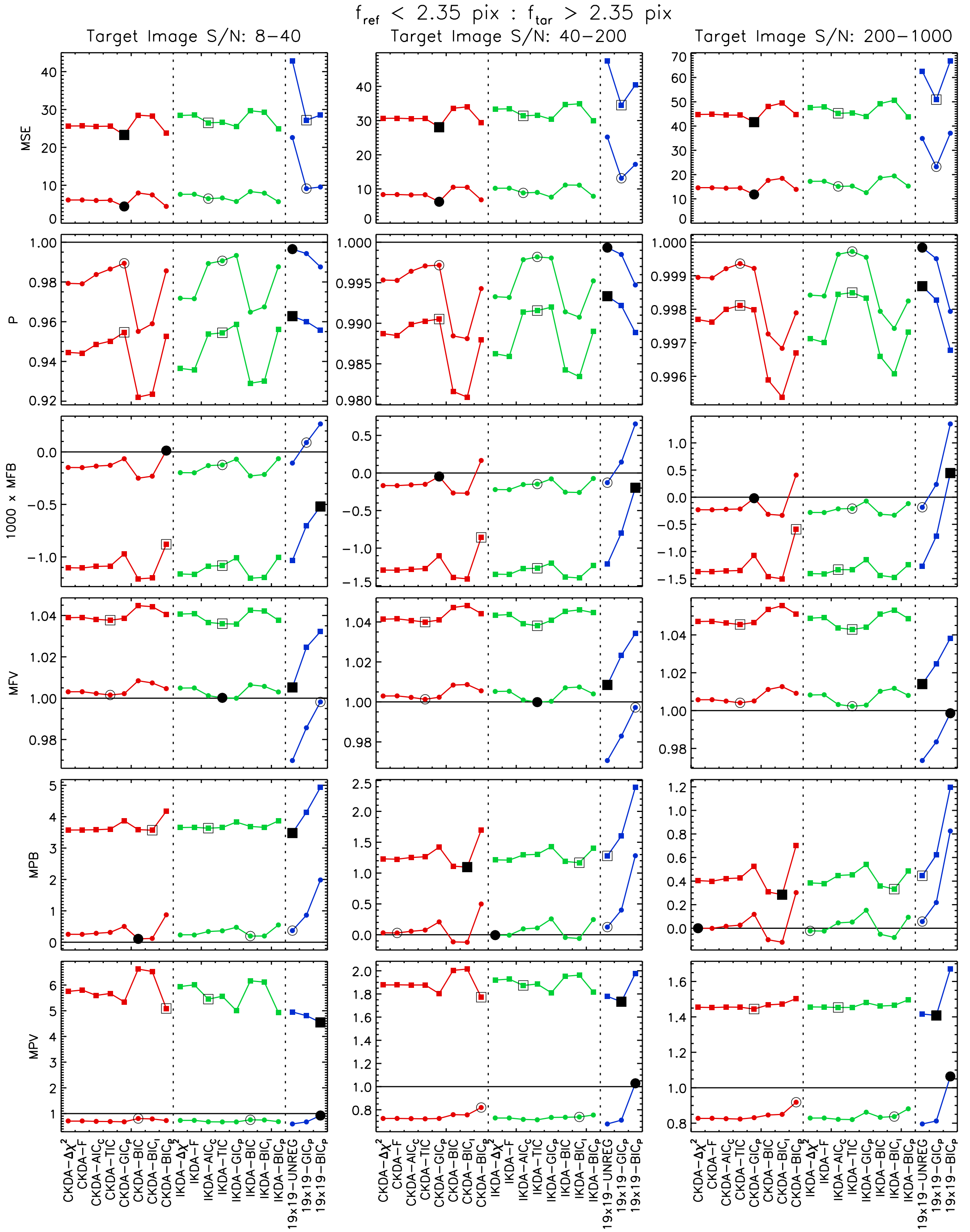


**Figure B2.** Plots of the median MSE, $P$, MFB, and MFV values (equations 59, 12, 60, and 61), and the MPB and MPV measures (equations 63 and 64), for each kernel solution method for $f_{\rm ref} \leq 2.35$ pix and $f_{\rm tar} \geq 2.35$ pix. The results in each plot have been calculated from ~29 000 simulations for each of the simulation sets S1 and S10. The layout, symbols, and colours used are the same as in Fig. 4. The IKDA-GIC$_{\rm P}$ and IKDA-BIC$_{\rm P}$ methods are excluded when determining the best values of the relevant model performance metric.

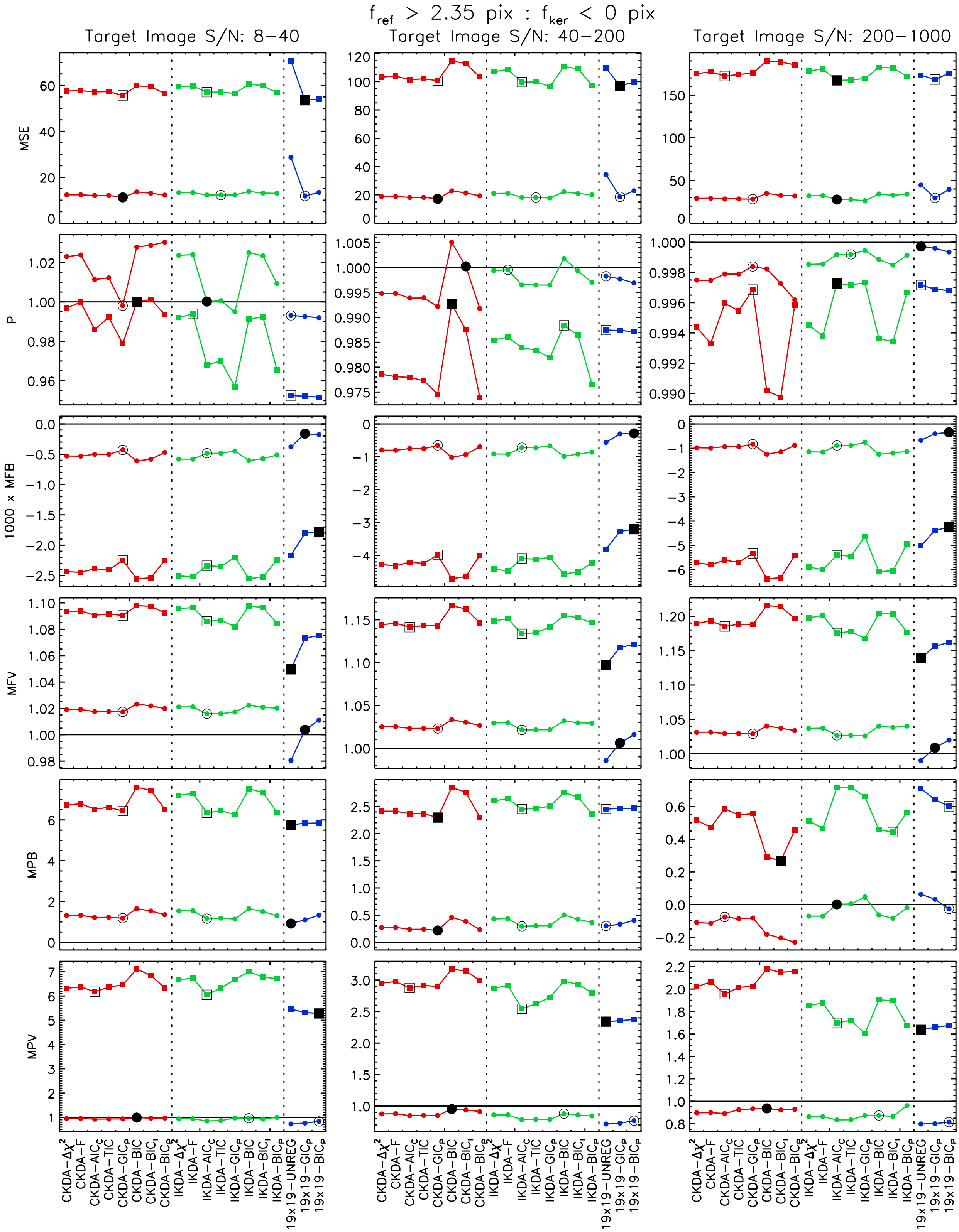


**Figure B3.** Plots of the median MSE, *P*, MFB, and MFV values (equations 59, 12, 60, and 61), and the MPB and MPV measures (equations 63 and 64), for each kernel solution method for $f_{\rm ref} \geq 2.35$ pix and $f_{\rm ker} \leq 0$ pix. The results in each plot have been calculated from ∼22 000 simulations for each of the simulation sets S1 and S10. The layout, symbols, and colours used are the same as in Fig. 4. The IKDA-GIC$_{\rm P}$ and IKDA-BIC$_{\rm P}$ methods are excluded when determining the best values of the relevant model performance metric.

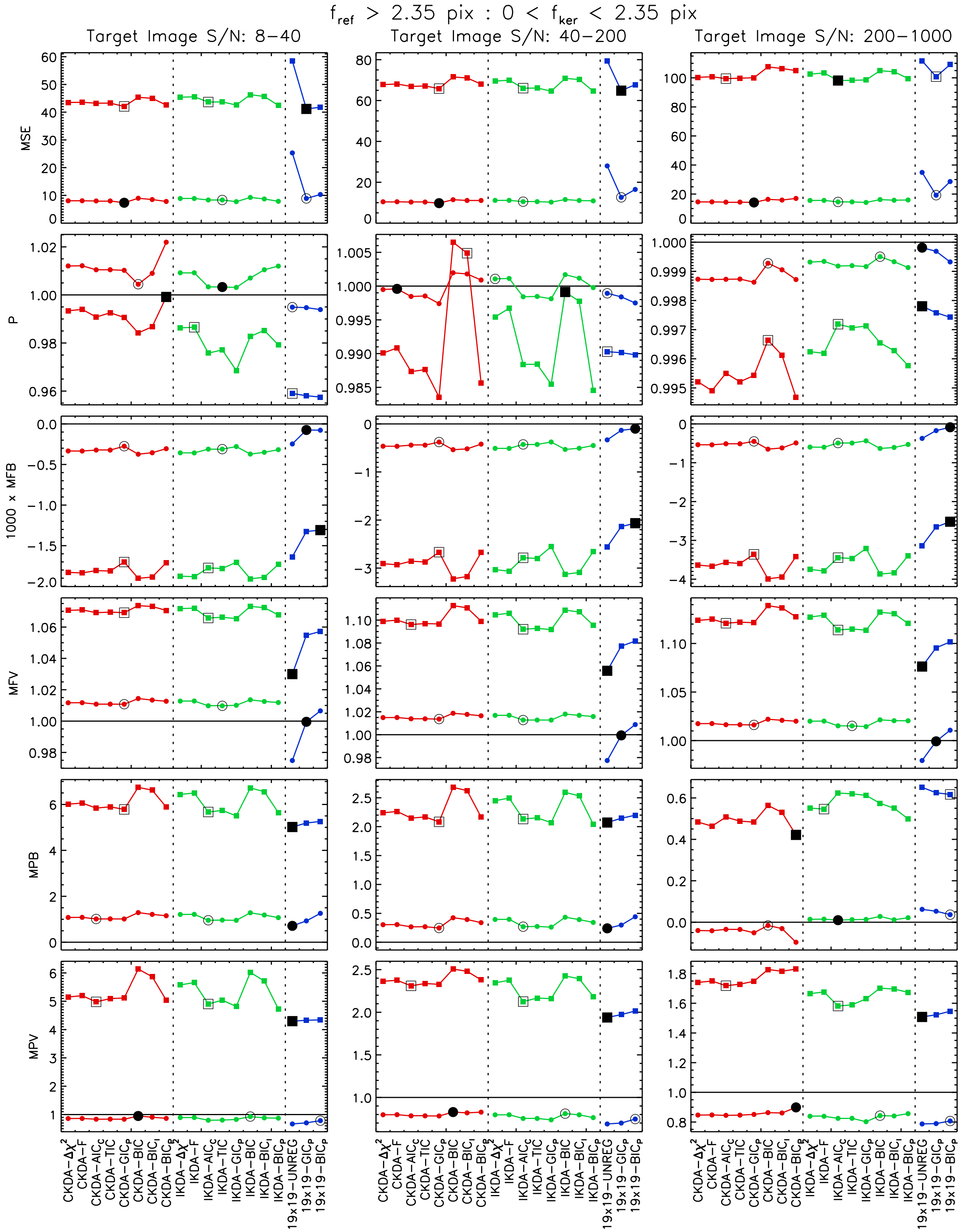


**Figure B4.** Plots of the median MSE, $P$, MFB, and MFV values (equations 59, 12, 60, and 61), and the MPB and MPV measures (equations 63 and 64), for each kernel solution method for $f_{\mathrm{ref}} \geq 2.35$ pix and $0 \leq f_{\mathrm{ker}} \leq 2.35$ pix. The results in each plot have been calculated from ∼53 000 simulations for each of the simulation sets S1 and S10. The layout, symbols, and colours used are the same as in Fig. 4. The IKDA-GIC$_{\mathrm{P}}$ and IKDA-BIC$_{\mathrm{P}}$ methods are excluded when determining the best values of the relevant model performance metric.

## APPENDIX C:

For completeness of this paper, in Figs C1–C4 we plot surfaces representing the median MSE, *P*, MFB, and MFV values for simulation set S10 as a function of the reference image and kernel FWHM. These plots are referred to briefly in Section 5.5.

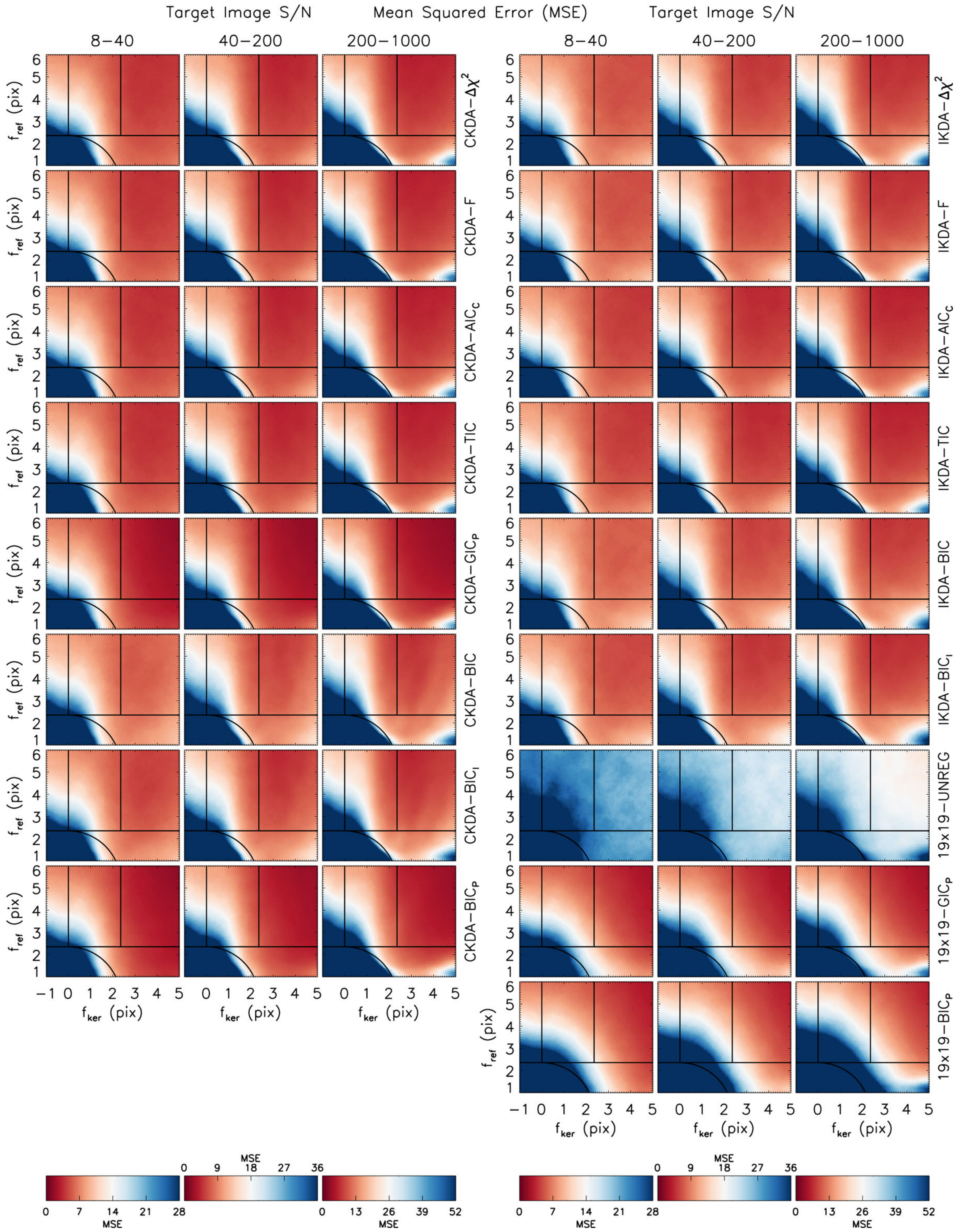


**Figure C1.** Plots of surfaces representing the median MSE values (equation 59) for simulation set S10 as a function of the reference image and kernel FWHM. The format of the figure is the same as in Fig. 5.

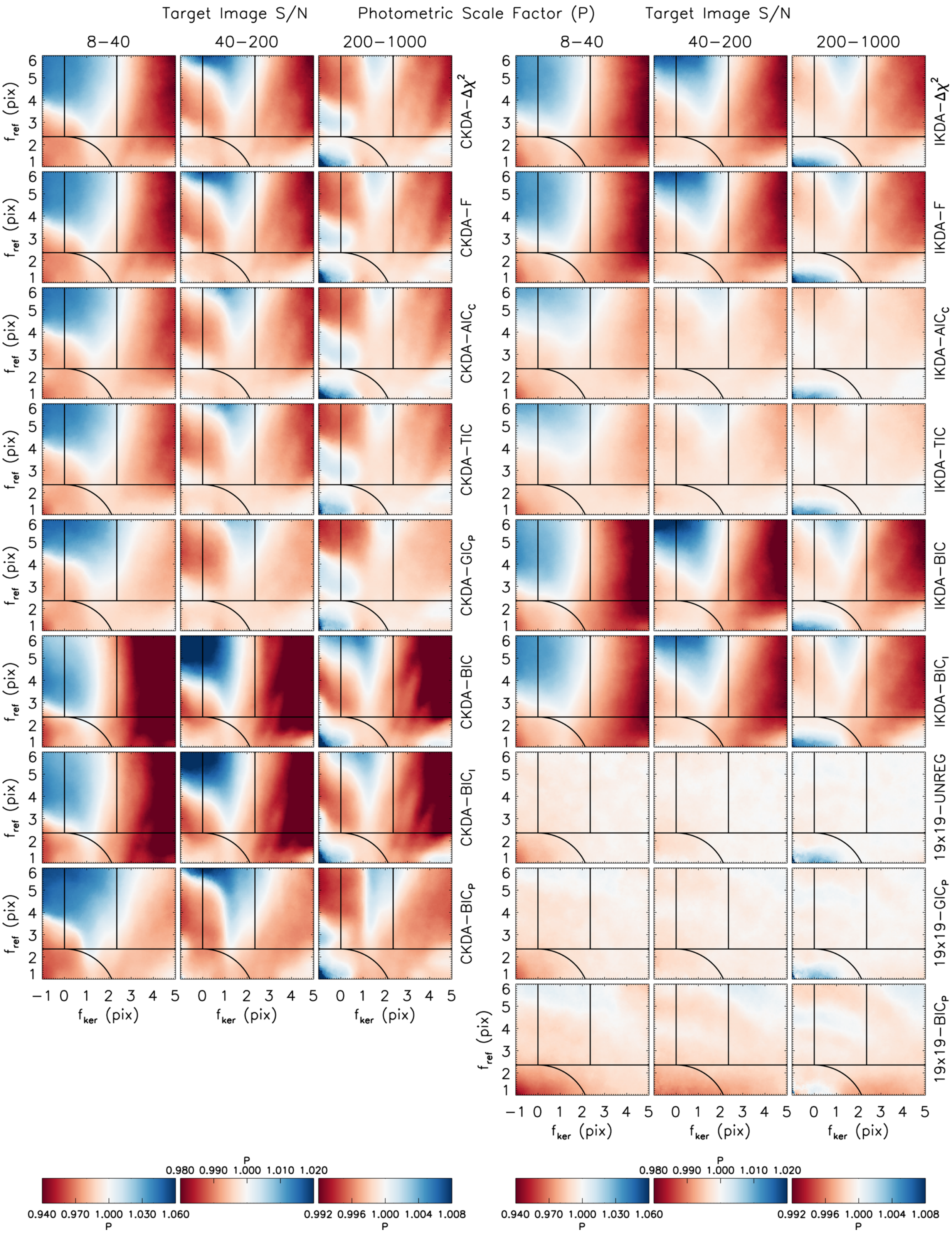


**Figure C2.** Plots of surfaces representing the median *P* values (equation 12) for simulation set S10 as a function of the reference image and kernel FWHM. The format of the figure is the same as in Fig. 5.

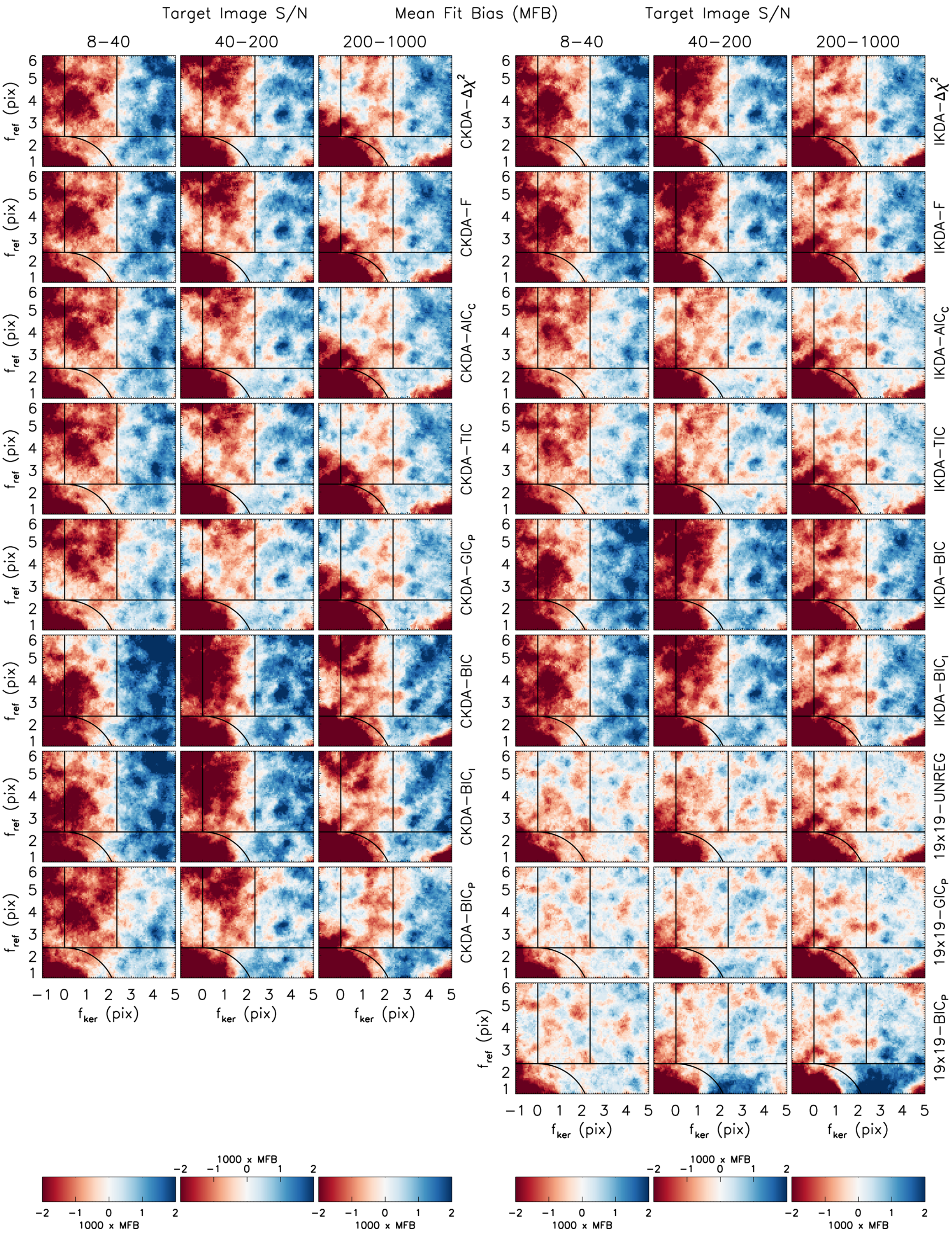


**Figure C3.** Plots of surfaces representing the median MFB values (equation 60) for simulation set S10 as a function of the reference image and kernel FWHM. The format of the figure is the same as in Fig. 5.

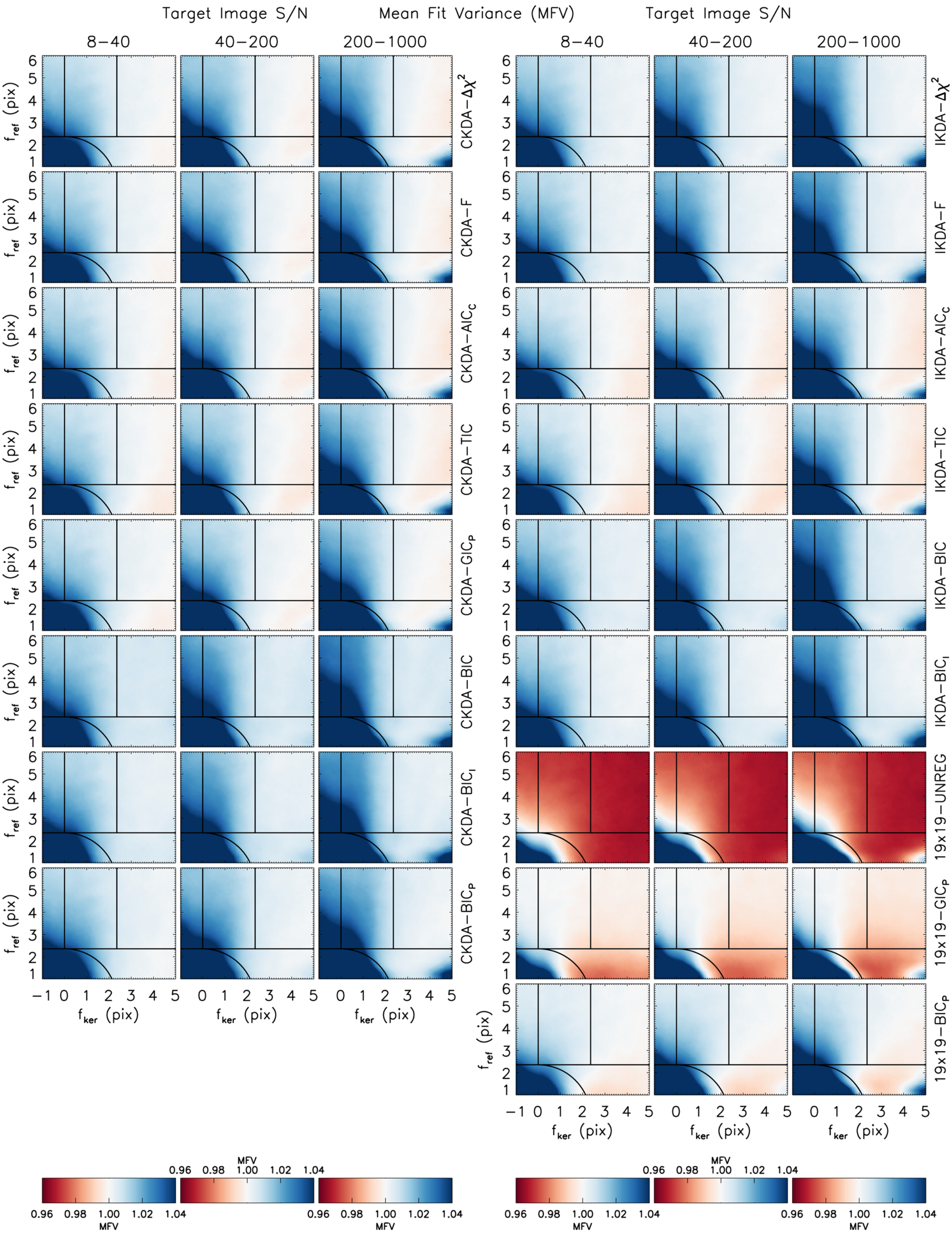


**Figure C4.** Plots of surfaces representing the median MFV values (equation 61) for simulation set S10 as a function of the reference image and kernel FWHM. The format of the figure is the same as in Fig. 5.

This paper has been typeset from a TEX/LATEX file prepared by the author.