Robust edge spread function construction methods to counter poor sample spacing uniformity in the slanted-edge method

F. van den Bergh,[1]

[1]CSIR, Meiring Naude road, Pretoria, South Africa
*fvdbergh@csir.co.za

Abstract: The slanted-edge method describes an algorithm for measuring the spatial frequency response (SFR) of digital imaging systems. The method can be applied to edges oriented at nearly any angle, but there are some angles that cause simplistic implementations of the algorithm to fail, or produce inaccurate measurements. These angle-dependent phenomena are demonstrated to stem from a lack of uniformity in supersample spacing in the edge spread function (ESF). Two well-known slanted-edge implementation variants are adapted to minimize edge orientation dependent errors. These robust slanted-edge implementations are demonstrated yield accurate measurements, regardless of edge orientation angle or moderate image noise.

1. Introduction

The slanted-edge method is an effective means of characterizing the spatial frequency response (SFR) of digital optical systems. It explicitly addresses the limited sampling resolution inherent to a digital system, allowing SFR measurements to be extended beyond the native Nyquist frequency implied by the discrete sample spacing of the digital sensor. The algorithm described by Reichenbach *et al.* in 1991 already included important refinements such as the necessary correction for employing a finite-difference numerical derivative in the line spread function (LSF) computation, and it was said that "the extended knife-edge technique can be applied to an edge at nearly any angle." [1]. Later descriptions of the algorithm, such as that contained in the ISO 12233 standard [2], restrict the allowed edge orientation to 5° relative to the sensor axes, presumably to minimize the computational complexity of the implementation.

Central to the idea of the slanted-edge algorithm is that the tilt in the knife-edge is responsible for the oversampling on which the method relies. The notion that different edge orientation angles result in different oversampling rates was mentioned by Reichenbach *et al.* [1, Section 3.2], but this does not fully characterize the impact of edge orientation on the quality of the measurement. In this paper a closer look is taken at the uniformity of the sample spacing as a function of edge orientation, identifying edge orientations that produce the worst uniformity. In addition, two variations on well-known edge spread function (ESF) construction methods are described in detail, including parameter choices that lead to slanted-edge implementations that are insensitive to edge orientation. These robust slanted-edge implementations are demonstrated to produce accurate results independent of edge orientation.

2. Background

An implementation of the slanted-edge method can typically be partitioned into three main steps [3, Section 2]: 1) the identification and modeling of the edge, 2) the construction of a regularly-spaced ESF using the image data and the edge model, and 3) the calculation of the spatial frequency response (SFR) as the Fourier Transform of the LSF, including the normalization of the SFR and the application of corrections . For brevity, this article will focus primarily on the second stage, the construction of the oversampled regularly-spaced ESF.
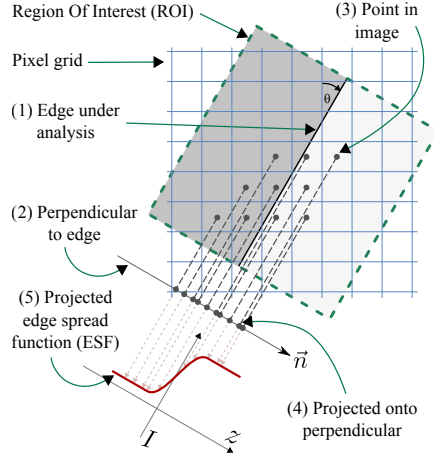
Fig. 1. The slanted-edge method

The slanted-edge algorithm operates on a subset of pixels surrounding the edge, referred to as the Region Of Interest (ROI), denoted by the symbol $\mathcal{R}$. In the implementation described by Kohm [4], the ROI is a rectangle that is aligned with the orientation of the edge, as illustrated in Figure 1. Note that the ROI is truncated at a specified distance of T pixels from the edge; the implications of the choice of T will be discussed in Section 4.1. For convenience, the pixels in $\mathcal{R}$ are enumerated sequentially using the index k, such that (x_k, y_k) represents the coordinates of the centre of pixel k, and the vector $\vec{p}_k \triangleq (x_k, y_k)$. The intensity of pixel k is denoted by I_k, or equivalently, $I(x_k, y_k)$.

Let the vector $\vec{c}$ denote a point falling on the edge, and let $\vec{n} = (-\sin\theta, \cos\theta)$ denote a unit vector normal to an edge at an angle of θ (Figure 1). The signed distance of each pixel $\vec{p}$ from the edge can be calculated by projecting the pixel's coordinates onto the vector $\vec{n}$, thus

$$d_k = d(\vec{p}_k) \triangleq (\vec{p}_k - \vec{c}) \cdot \vec{n} \tag{1}$$

An irregularly-spaced ESF can be obtained by forming a set of tuples combining the distance-from-edge and intensity of each pixel, to yield the set ESF_i:

$$\text{ESF}_i \triangleq \{(d_k, I_k)\} \quad \forall k | \vec{p}_k \in \mathcal{R}. \tag{2}$$

This step captures the essence of the slanted-edge method: the signed distance values d_k have a spacing that is much finer than the original pixel grid, thus allowing us to effectively sample the ESF with sub-pixel resolution. The ESF can be differentiated to yield the line spread function (LSF); from the projection slice theorem [5] we know that the Fourier Transform (FT) of the LSF will yield a radial transect through the system Optical Transfer Function (OTF) along the angle θ. The modulus of the OTF is called the modulation transfer function (MTF). The slanted-edge method can be used to measure the spatial frequency response (SFR) of the system; if the modulation of the test object is known, it can be used to derive the system MTF from the measured SFR. The irregular spacing of the samples in ESF_r precludes the direct application of the Fast Fourier Transform (FFT); typically we have to resample ESF_i to obtain an ESF with regular sample spacing, ESF_r.

3.   Details of constructing a regularly-sampled ESF

There are two main approaches to constructing a regularly-sampled ESF: binning, and interpolation. The ISO 12233 standard describes an implementation of the binning approach. Let the sequence
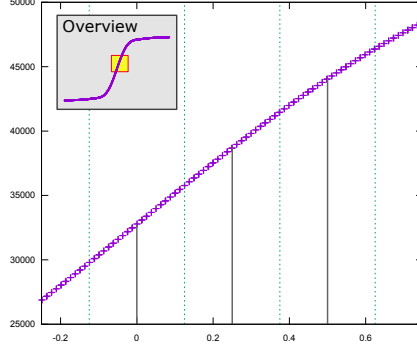
Fig. 2. An example of a uniform distribution of samples (the '+' symbols) from ESF$_i$ throughout each bin, resulting from an edge orientation angle of 5°.

x_j denote the desired positions at which the regular ESF is to be sampled. For 4× oversampling, the sequence x_j could be defined as

$$x_j = \frac{j}{4}, \quad \text{for } j \in \mathbb{Z}. \tag{3}$$

In practice, the values of j are restricted to the range that covers the ROI truncated at distance T pixels from the edge. The binning approach allocates each pixel k to a bin j such that

$$\frac{x_{j-1} + x_j}{2} \leq d_k < \frac{x_j + x_{j+1}}{2}. \tag{4}$$

To estimate the desired value of the regularly-sampled ESF at sampling position x_j some representative value must be constructed from all the I_k values of tuples with d_k values falling into bin j, as defined in Equation 4. The simplest solution is to compute the arithmetic mean of all the I_k values in bin j; this solution is effective provided that the d_k values falling in bin j are uniformly distributed throughout the bin, a condition which is easily met when the relative edge angle θ is constrained to be close to 5°. This condition is illustrated in Figure 2: the '+' symbols represent actual values of ESF$_i$, the thick vertical solid lines represent a 0.25-pixel regular sample spacing (the bin midpoints x_j), and the dotted vertical lines represent the boundaries of the bins. The inset in the top left corner illustrates the entire ESF, with the box representing approximately the section of the ESF shown in the main plot.

3.1. Critical angles

If a slanted-edge method implementation is intended for use outside of laboratory conditions, such as on-orbit validation of satellite sensors, then the 5° edge orientation requirement of the ISO 12233 method is impractical. The construction of the irregularly sampled ESF in Equations 1–2, as proposed by Kohm [4], is already suitable for constructing ESFs from edges with arbitrary orientation angles. Despite this, some edge orientation angles do not provide sufficient oversampling to allow the slanted-edge method to measure accurately, without aliasing, up to the Nyquist frequency (0.5 cycles/pixel) implied by the photosite pitch of the sensor. If θ is an integer multiple of 90°, then no oversampling is achieved, but there are other edge orientations that also produce sub-optimal oversampling.

Some insight can be gained by studying edges with slopes that are rational fractions, e.g., a slope of 1/2. The unit normal vector for an edge with a slope of p rows over q columns (where
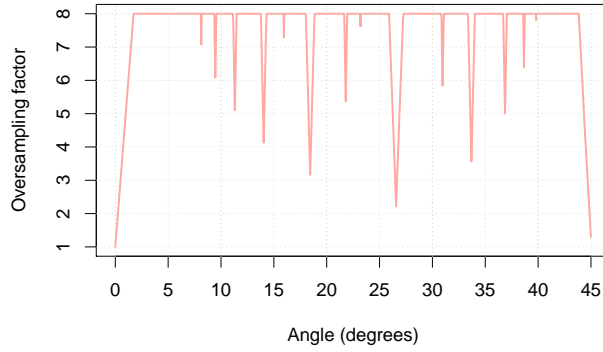
Fig. 3. The mean achievable oversampling factor, capped at 8×, as a function of edge orientation angle, measured on a simulated edge of 30 pixels in length.

the greatest common factor of p and q is equal to one) pixels is

$$\vec{n} = \frac{1}{\sqrt{p^2 + q^2}}(-p, q). \tag{5}$$

Substituting Equation 5 into Equation 1, and simplifying yields

$$d_k = \frac{(y_k q - x_k p)}{\sqrt{p^2 + q^2}} + C, \tag{6}$$

where $C = \vec{c} \cdot \vec{n}$. Since x_k, y_k, p and q are all integers, it follows that $d_k - C$ must be an integer multiple of $(p^2 + q^2)^{-1/2}$. The sub-pixel edge position (edge phase) of an edge with a slope of p/q will repeat with a period of q columns (assuming a nominally horizontal slanted-edge), producing at most q unique d_k values in the interval $[0, 1)$, thus limiting the maximum achievable oversampling factor. A subset of this phenomenon, limited to edges with a slope of $1/q$ for q between 0 (sic) and 9, was identified by Samei *et al.* [6]. A more comprehensive definition is that edges with slopes of the form p/q with $1 \le p \le q \le N$, and $\gcd(p, q) = 1$ can guarantee an oversampling factor of at most N.

Consider the example of an edge slope of 2/3 ($\approx 33.69°$), which implies the $d_k - C$ values are multiples of $13^{-1/2} \approx 0.2774$, thus either 3 or 4 of the d_k values will fall in any interval (along the edge normal) with a length of 1 pixel, resulting in 3× oversampling. What happens when we choose a nearby slope, such as 201/300? The edge with slope 2/3 repeats at an interval of 3 columns, at which point the edge phase (modulo 1) is exactly zero; the edge with slope 201/300 will have an edge phase (modulo 1) of 0.01 after advancing three columns. After 6 columns, the edge phase (modulo 1) will be 0.02, and so on. Given a sufficiently long edge, even such a small edge phase advance will allow the samples to spread throughout our 1-pixel-length interval (measured along the edge normal) with sufficient uniformity to support 8× oversampling at this specific slope.

A simulation can be used to demonstrate the mean achievable oversampling rate that can be obtained at any given edge orientation angle. The simulation ran over an ROI of 31×31 pixels, but the simulated edge was truncated to a length of 30 pixels. A total of 20000 uniformly-spaced edge orientation angles in the range $[0°, 45°]$ were simulated, calculating the value of d_k at each pixel coordinate using Equation 1, but keeping only the samples with $0 \le d_k < 1$. These samples were binned into 8 equal-width bins, keeping count of the number of non-empty bins. The sub-pixel edge position was varied over the range $[0, 1)$ pixels in 1000 steps, and the mean number of non-empty bins over all these edge positions was calculated. Figure 3 illustrates the
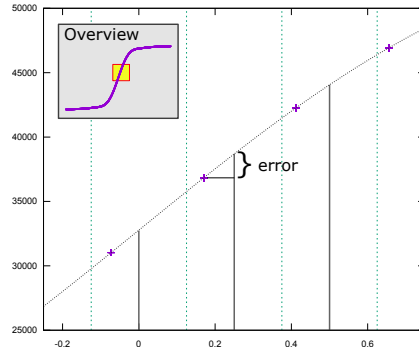
Fig. 4. An example of a very biased distribution of samples (the '+' symbols) from ESF_i within each bin, resulting from an edge orientation of 14.036°. Notice how all the samples fall exactly on top of each other, clumping to one side of the midpoint of each bin.

mean number of non-empty bins, which acts as a proxy for the mean achievable oversampling factor (capped at 8×), as a function of edge angle.

The significance of Figure 3 is that it provides a set of edge orientation intervals that should be avoided if it is likely that the system SFR is not bandlimited, and the ROI yields an edge length of 30 pixels. Regardless of whether the system SFR is bandlimited, it is prudent to avoid angles below 1.73° and above 43.84°. At longer edge lengths, the set of edge orientations that afford 8× oversampling increases slightly, compared to the Figure 3.

3.2. Poor sampling uniformity

Unfortunately the problems with simple binning are not limited to just empty bins; a more subtle but equally destructive phenomenon can manifest. Consider a 4× oversampling simple binning implementation (thus the bins are 0.25 pixels wide), encountering an edge with a slope of 1/4, resulting in a relative d_k spacing of ≈ 0.2425 pixels. This will not produce any empty bins, but almost all the bins will contain only one unique d_k value, regardless of the size of the ROI. In each bin the d_k value will typically not fall on the midpoint of the bin (our desired x_j sampling position), but rather it will be to one side of the midpoint. There will be no averaging of values on either side of the midpoint, as would be the case with a 5° edge orientation. This phenomenon is illustrated in Figure 4: notice how all the samples (the '+' symbols) in each bin are completely coincident (compare with Figure 2), resulting in a biased representation of the contents of the bin. In this example, the mean intensity of all the samples in the bin will underestimate the true value of the ESF at the midpoint of the bin, resulting in the error as indicated. This type of systematic distortion could lead to a narrowing (or widening) of the central peak of the LSF, with the resultant error in the estimated SFR.

This phenomenon is not limited to just the obvious rational edge slopes: For example, slopes that are close to 1/4, such as 1003/4000, or 999/4000 will produce a very small spread of d_k values within each bin, and simple binning will still fail to produce an unbiased estimate of the regularly-sampled ESF. Two reasonable questions arise: Is this phenomenon severe enough to impact the accuracy of a slanted-edge implementation, and does the phenomenon manifest at only a handful of discrete angles, or does it occur throughout the range [0°, 45°]?

Masaoka proposed a variant of the simple binning algorithm, discussed in more detail in Section 3.3, and performed an empirical study of the impact of edge orientation on this implementation [7]. Although the results of that study are inextricably entwined with the proposed algorithm implementation (as shown in Section 5), Masaoka's results do indicate that the phenomenon appears to affect the accuracy of the algorithm at angles throughout the range,
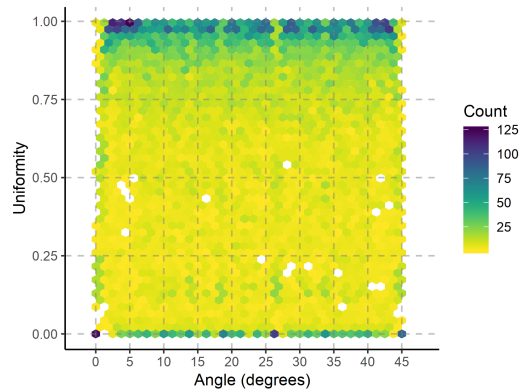
Fig. 5. The frequency of worst-case sample spacing uniformity scores as a function of edge angle. The y-axis represents the uniformity score, which is the probability of the samples being uniformly distributed throughout each 1/8-th pixel bin.

not just a small discrete subset.

In an attempt to quantify the prevalence of problematic angles without reference to a specific slanted-edge algorithm implementation, one can directly evaluate the uniformity of the distribution of the d_k values through simulation. The Cramér-von Mises test [8] measures the deviation of an empirical cumulative distribution function (CDF) from an expected analytical CDF. A simulation similar to the one described in Section 3.1 was performed, but this time over a 201×201 pixel ROI, while limiting the edge length to 200 pixels to ensure the same number of samples are evaluated at each edge orientation in the range $[0°, 45°]$. At each edge orientation, the sub-pixel position of the simulated edge was stepped through the range $[0, 1)$ in steps of $1/1000^{\text{th}}$ of a pixel. At each edge phase, the list of d_k values falling in the range $[0, 0.125)$ were evaluated (thus simulating $8\times$ oversampling) for uniformity using the Cramér-von Mises test, with the test statistic expressed as the probability of the samples being drawn from a uniform distribution.

To characterize the worst-case scenario over all edge phases, only the minimum uniformity probability at each edge orientation angle was retained. Since a real-world slanted-edge measurement will encounter arbitrary edge phase values, the decision to evaluated the uniformity of the worst-case scenario is justified by the desire to eventually derive a slanted-edge algorithm implementation that is guaranteed to perform predictably regardless of the uniformity of the d_k values in any particular bin. This implies that an edge orientation of $\approx 26.565°$, corresponding to a slope of 1/2 and a uniform d_k spacing of ≈ 0.44721, will receive a worst-case uniformity probability of exactly zero, since there are edge phase values that result in no d_k values falling in the range $[0, 0.125)$. Figure 5 illustrates the resulting 2D histogram of uniformity as a function of edge angle.

From the figure it is apparent that there are edges with a uniform distribution of d_k values throughout the approximate range $[1°, 44°]$, but that we also find edges with very poor uniformity throughout the same range, illustrated by the row of higher-frequency bins at zero uniformity. Approximately 8.71% of the edge orientations produce uniformity probabilities below 0.05; these are shown in Figure 6. Some of the known critical angles, such as $26.565°$ and $45°$, are clearly visible as a cluster of points with zero uniformity, but at this scale of visualization it is clear that there are very few "safe" angles, meaning angles that are not surrounded by nearby angles with very poor uniformity. Even the ISO 12233 recommended angle of 5° degrees is boxed in by the angles $4.763°$ and $5.192°$, both of which score poorly on the uniformity test.

It should be noted that the simulated edge length of 200 pixels was chosen because not only does it represent a reasonable edge length for high-quality slanted-edge measurements, but it also
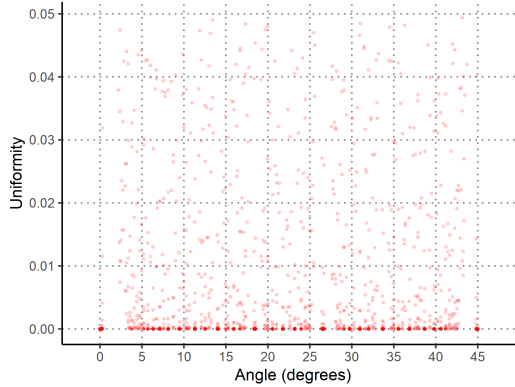
Fig. 6. Discrete points representing the uniformity score as a function of angle, focusing on angles with a uniformity score below 0.05, for a simulated edge length of 200 pixels.

results in a reasonable number of d_k samples falling in the bin $[0, 0.125)$, at a median value of 25 samples, thus conferring good statistical power to the uniformity test.

If the simulations are performed with a shorter simulated edge length, the apparent uniformity of the distribution of d_k values tends to decrease at all edge orientation angles, but the resulting smaller sample sizes reduces the statistical power of the uniformity test. The simulations results shown in Figure 5 thus represent the worst-case uniformity under good conditions (here taken to mean a sufficiently long edge), illustrating that poor sampling uniformity must be explicitly addressed by a robust slanted-edge implementation, such as those discussed in Sections 3.3–3.5.

3.3. Finer binning

Masaoka proposed a basic interpolation algorithm for estimating the value of a bin containing no samples at all (the 'empty bin' phenomenon described above): simply pick the value from the closest non-empty bin to the left [7]. If a bin is empty, and its immediate left and right neighbours are not empty (the left neighbour could have been interpolated already), then the estimate for the current bin is the arithmetic mean of the left and right neighbour bins. Masaoka further demonstrated that the performance of his slanted-edge implementation improved progressively as the oversampling factor was increased, up to 32× oversampling. Although this was not explicitly stated by Masaoka, the increase in the oversampling factor is in fact a strategy for dealing with the non-uniform distribution of samples within each non-empty bin, placing the x_j bin midpoints closer to the d_k values.

3.4. Kernel-based interpolation

The binning step of the basic slanted-edge implementation can be represented as a two-step process: the scene ESF is convolved with a box function, followed by point-sampling at the midpoint of the bin [9, Section IV.A]. The convolution of the scene ESF and the box function is only evaluated at the bin midpoint values x_j, and the scene ESF is approximated by the irregularly spaced discrete ESF_i, so the combined convolution and sampling can be expressed as a discrete sum

$$\text{ESF}_{r,j} = \frac{\sum_k \text{rect}(wd_k - wx_j)I_k}{\sum_k \text{rect}(wd_k - wx_j)} \tag{7}$$

where $\text{ESF}_{r,j}$ represents bin j of the regularly sampled ESF_r, and w represents the reciprocal bin width, e.g. $w = 8$ for 8× oversampling. The convolution with the box function imparts a negative bias onto the SFR obtained by the slanted-edge method, but this can be corrected by

dividing the SFR by the Fourier transform of the box function, i.e., sinc(f/w). Greer *et al.* go on to demonstrate that the full correction is a function of the edge orientation angle [9], but that detail will be omitted here for brevity.

This box function convolution gives all samples within the bin the same weight, as opposed to a more natural weighting that would assign a higher weight to samples I_k with d_k values closer to the midpoint of the bin. If the box function convolution is replaced by a convolution with some other compact function, such as a truncated Gaussian pulse, or perhaps a truncated Laplace distribution, then we can obtain two desirable properties: firstly, the samples I_k can be weighted according their distance d_k from the bin midpoints x_j, and secondly, the samples from nearby bins can seamlessly contribute to the current bin if the filter function is sufficiently wide relative to the bin size. The more generic form of Equation 7 thus becomes

$$\text{ESF}_{r,j} = \frac{\sum_k h(d_k - x_j)I_k}{\sum_k h(d_k - x_j)} \tag{8}$$

where h represents any suitable compact kernel function. Note that Equation 8 can be evaluated over only the subset of samples k that fall in the interval over which $h(d_k - x_j)$ is nonzero for improved efficiency. Towards the end of the slanted-edge algorithm, the estimated SFR is divided by the Fourier transform of the kernel h to remove any unwanted attenuation that may have been introduced by the convolution.

Three recent papers described implementations that employ a kernel-based interpolation method to construct the regularly-sampled ESF: Van den Bergh proposed a Laplacian distribution kernel [10], whereas Zhang *et al.* [11] and Duan *et al.* [12] used truncated Gaussian kernels. Interestingly enough, Duan *et al.* approached the problem from the perspective of applying the Non-Uniform FFT algorithm (NUFFT) [13] to ESF_i, but mathematically this is equivalent the kernel-based interpolation method described here.

3.5. Polynomial- or Spline-based interpolation

Fine detail that can be seen on close examination of analytical LSFs corresponding to the diffraction component (for small circular apertures) indicates that fitting a purely global function to the observed ESF_i data points is not advised. Instead, a locally adaptive method such as a local polynomial fit (e.g, locally estimated scatterplot smoothing, or LOESS) [14] or a smoothing spline [15] might be more appropriate. An early slanted-edge implementation of a local polynomial approach, involving a sliding 4th-order polynomial fit with Gaussian weighting, was proposed for use in the field of digital radiography by Samei *et al.* [6]. In a recent paper, Zhou *et al.* described an ESF construction method based on C-splines, which were further extended to include monotonicity constraints [16].

One of the attractive properties of local polynomial or spline interpolation is that a well-balanced fit can preserve the SFR contrast in the passband (e.g., between 0 and 1 cycles/pixel) without any additional bias, and without requiring any correction of the SFR like in the case of the kernel-based interpolation methods. This property can be used effectively to reduce the impact of image noise on the estimated SFR.

4. Proposed algorithms

While both local polynomial interpolation and kernel-based interpolation ESF construction methods are known from existing literature, no guidelines have been provided on how to fine-tune an implementation to offer good robustness in the face of edge orientation dependent sampling non-uniformity. The algorithms proposed here are therefore not new, but detailed descriptions are provided to reduce the amount of testing that future researchers will have to perform to obtain robust but relatively efficient implementations.
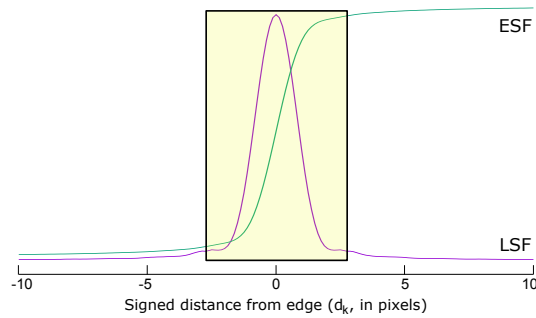
Fig. 7. An example of an ESF with its corresponding LSF (scaled for display purposes), with the box highlighting the central section of the ESF.

4.1. Apodization, truncation, and ESF tail noise filtering

Before delving into the details of the implementation of the ESF construction methods, it is helpful to consider some general properties of a typical ESF. The central section, as illustrated in Figure 7, carries the bulk of the information required to estimate the higher frequencies (say, above 0.1 cycles/pixel) of the SFR. In the ideal case of a noise-free image, it is desirable to apply the slanted-edge method to a wide ROI to minimize the truncation of the ESF. This follows from the observation that if the ESF is truncated at a distance T from the edge, then this is equivalent to multiplication of the ESF with a rectangular window of width $2T$, centred on the edge. This in turn implies that the SFR measured by the slanted-edge method is actually the convolution of the true system SFR and a sinc($2Tf$) function, which typically distorts the low frequencies of the estimated SFR, as illustrated by Viallefont-Robinet and Léger [17, Section 7]. Increasing T in turn decreases the width of the peak of the sinc($2Tf$) function involved in this convolution, which reduces the distortion of the estimated SFR.

On the other hand, if the image is not noise-free, then a wide ROI (large T) increases the impact of noise on the estimated SFR, to the extent that Burns and Williams recommended choosing a smaller T as a strategy for dealing with image noise [18]. Later, Williams and Burns [19] recommended an improved strategy: rather than truncating the ESF to control noise, a low-pass filter can be applied to the ESF tails (i.e., the regions outside of the box representing the central part of the ESF illustrated in Figure 7). To implement this strategy, a robust method is required to estimate the start of the ESF tails, expressed as a distance from the edge, d_T. A reasonable approach is to define d_T in terms of the knee-points of the ESF.

There are several methods that may be employed to estimate the location of the knee-points in curves, including the Kneedle algorithm [20], or locating the point of maximum curvature in the ESF [3, Section 3.6]. Another method, with very low computational complexity, is to identify the 10% and 90% rise points along the ESF, that is, the location at which the ESF reaches 10% and 90% of the overall contrast, with contrast being estimated as the difference between the asymptotes of the ESF tails. All of these methods require a relatively noise-free estimate of the ESF to obtain good results; experimentation has shown that building an 8× oversampled ESF_r using kernel-based interpolation (as described in Section 3.4), with the kernel

$$h(x) = \begin{cases} 1 - \frac{7}{4}|x|, & \text{if } |x| \le \frac{4}{7} \\ 0, & \text{otherwise} \end{cases} \tag{9}$$

produces acceptable results, even at an edge orientation of 45°. An additional pass with a 5-sample moving average filter over ESF_r is applied to improve performance at very low signal-to-noise ratios; the slight broadening of the ESF caused by this additional filter also helps to make the

estimate of d_T more conservative. Let d_{10} and d_{90} represent the signed distance-from-edge values at which ESF_r reaches 10% and 90% contrast. A conservative estimate of the start of the ESF tails, d_T, is obtained such that $d_T = 1.5 + \max(|d_{10}|, |d_{90}|)$. After d_T has been obtained, the temporary ESF_r can be discarded, to be replaced by a better estimate of ESF_r obtained using either local polynomial interpolation, or the kernel-based interpolation method described below in Section 4.2.

With the estimate of d_T it is possible to apply the ESF tail smoothing suggested by Williams and Burns [19]. In practice, a moving average filter with an effective width of 4 pixels (32 samples at 8× oversampling) has proved to be very effective at moderate noise levels (CNR of 30dB or better). The width of the filter can be tapered down near d_T to obtain even more conservative behaviour.

Near the end of a typical slanted-edge implementation an FFT is applied to the LSF to obtain the SFR. By its nature, the FFT treats the LSF as a signal that repeats periodically, with a period equal to the length of the array containing the LSF. This implies that ideally the LSF should be zero at both ends, or some spectral leakage will occur. It is not uncommon for a slanted-edge implementation to apply a windowing function to effectively apodize the LSF; one such example is the ISO 12233 standard [2]. If the ESF/LSF was truncated at a short distance from the edge, such as initially recommended Burns and Williams, it is much more likely that the LSF was not naturally near zero, especially for systems with wider PSFs (e.g., such as caused by a small lens aperture, or defocus), making the windowing step with the Hann window seem inevitable. If the ESF is truncated at a much greater distance from the edge, however, the LSF is much more likely to already be near zero, and any additional windowing with a Hann or similar window will just increase the distortion of the SFR at low frequencies. An alternative strategy, which is particularly effective when it is possible and/or appropriate to truncate the ESF only at a distance of 24 pixels or greater, is to use the ESF tail smoothing operation to achieve a natural taper of the LSF tails. This is accomplished by extending the ESF by replicating the last sample at each end of the ESF for an additional length of at least 4 pixels (equivalent to padding the LSF with zeroes over the same length) before applying the ESF tail filtering.

4.2. Robust kernel-based interpolation method

The kernel-based interpolation algorithm determines the start of the ESF tails, d_T, described in Section 4.1, and applies Equation 8 with the kernel

$$h(x) = \begin{cases} e^{-\alpha x^2}, & \text{if } |x| \le 2 \\ 0, & \text{otherwise} \end{cases} \tag{10}$$

to all bins where $|x_j| < d_T$. In the remaining bins, with $|x_j| \ge d_T$, a simple average over the interval $[x_j - 0.5, x_j + 0.5]$ is used to estimate $\text{ESF}_{r,j}$. An acceptable value for α is 26, obtained through optimization in Section 5.2. ESF tail smoothing as described in Section 4.1 is applied to effect apodization and to ESF_r to reduce the impact of noise.

After the FFT has been applied to the LSF, the SFR is corrected by dividing it by the FT of the kernel given in Equation 10; to incorporate the influence of the truncation, the correction is obtained numerically using an FFT. This implementation requires about 2.45 ms to produce an SFR over a 200×200 image on relatively modest modern hardware.

4.3. Robust local polynomial interpolation method

The local polynomial interpolation algorithm first determines the start of the ESF tails, denoted by the parameter d_T, described in Section 4.1. The strategy is to use 6th order orthogonal Chebyshev polynomials of the first kind to construct the central part of the ESF where $|x_j| < d_T$, dropping down to simple averaging over the window $[x_j - 0.5, x_j + 0.5]$ for the remaining bins

with $|x_j| \geq d_T$. The Chebyshev polynomials require the d_k values within the fitting window to be normalized to the range $[-1, 1]$, but this normalization is prudent to improve numerical accuracy regardless of which polynomial basis is used. To accommodate the worst-case poor sampling uniformity the local polynomial fit is performed over a sliding window with a width of 4.5 pixels; such a wide window must be paired with a fairly narrow weighting function. The recommended weighting function is

$$w(x) = \begin{cases} 1, & \text{if } |x| \leq \beta \\ e^{-\alpha|x|}, & \text{otherwise} \end{cases} \tag{11}$$

with $\beta = 0.125$ when $|x_j| <= 0.5d_T$ and $\beta = 0.250$ for $0.5d_t < |x_j| < d_T$. These proposed values of β, as well as the associated thresholds (e.g., $0.5d_T$) were found experimentally to produce acceptable results on a wide range of modelled systems. Individual samples from ESF_i are weighted in the polynomial regression using $w(d_k - x_j)$ when constructing bin j. Through the optimization process described in Section 5.2 the value $\alpha = 5.5$ is found to offer good performance in the passband $[0, 1]$ cycles per pixel.

The local polynomial coefficients can be obtained by solving for the design matrix using Cholesky factorization followed by forward substitution and back substitution, provided that a Tikhonov ridge regression parameter of approximately 5×10^{-7} is used when computing the solution using double-precision floating point numbers. This is significantly faster than the alternatives, such as singular value decomposition or QR factorization, and the choice of orthogonal polynomials and domain-normalization together with the Tikhonov regularization yields sufficient numerical precision.

After the initial ESF_r has been constructed using the polynomial fitting in the center, and simple averaging in the tails, the ESF tail filtering method described in Section 4.1 is applied to further reduce any residual noise in the tails. This implementation requires about 3.47 ms to produce an SFR over a 200×200 image on relatively modest modern hardware.

5. Results

5.1. Synthetic image set description

To evaluate the impact of poor sample uniformity owing to specific edge orientations, and the efficacy of the proposed algorithms at countering the poor sampling, a number of synthetic images were generated. The synthetic images simulate an ideal aberration-free lens, focused perfectly, attached to a sensor with a perfectly square 100% fill-factor photosite apertures. This system yields analytical MTFs of the form

$$\text{MTF}(f) = \frac{2}{\pi} |\text{sinc}(f \cos \theta)\text{sinc}(f \sin \theta)| \left(\arccos(f) - f\sqrt{1 - f^2} \right) \tag{12}$$

where $0 \leq f \leq 1$ represents normalized frequency, θ represents the edge orientation angle, and $\text{sinc}(x) = \sin(\pi x)/(\pi x)$. The normalized frequency f is obtained as $f = f_p N \lambda$, where f_p denotes a frequency in physical units (e.g., cycles per mm), N denotes the relative aperture f-number, and λ the wavelength of the simulated light. To render a simulated image, it is convenient to divide the product $f_p N \lambda$ by the sensor photosite pitch, allowing f_p, and thus f, to be expressed in units of cycles per pixel. The simulated images were generated to match a system with a photosite pitch of 5 μm, monochromatic light at a wavelength of 550 nm, and relative apertures of f/4, f/11 and f/16, using an importance sampling algorithm [21] with 40401 samples per pixel. The extinction frequencies at f/4, f/11 and f/16 are approximately 2.27, 0.83 and 0.57 cycles/pixel at small edge angles. Examples of the analytical system MTFs are shown in Figure 8.

The simulations were repeated at edge orientations from the set $\theta \in \{5, 7.125, 9.462, 11.310, 14.036, 18.435, 21.801, 26.565, 30.964, 33.690, 36.870, 38.660, 39.806, 40.601\}$ degrees. The
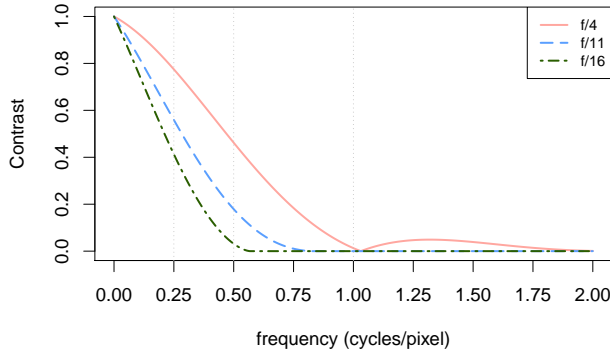
Fig. 8. Analytical MTF curves of the synthetic images, shown at an edge orientation of 14.036°.

5° edge orientation represents the reference angle for excellent sampling uniformity, but the other orientations were selected to correspond to the worst-case sampling uniformity observed in Section 3.2. At each edge orientation the sub-pixel edge phase was varied in steps of $1/37^{th}$ pixel through the range $[0, 36/37]$ in an attempt to elicit greater variance in the performance. The total set comprised 3 apertures × 14 angles × 37 edge phase values = 1554 images.

Signal-independent Gaussian noise was added to the noise-free versions of the simulated images to evaluate the performance of the algorithms under moderate noise conditions of a contrast-to-noise ratio of 35 dB. This signal-independent noise is not a realistic simulation of signal-dependent photon shot noise, but it should suffice to illustrate the impact of noise on the algorithms. Only one instance of noise was generated at each edge phase, so the total number of noisy images was 1554, just like the noise-free set.

In all the experiments the edge orientation angle was explicitly provided as an input to each implementation. This eliminates the possibility of an error in the edge orientation estimation algorithm (part of the edge modeling phase) as an additional source of error.

All the slanted-edge implementations used in the experiments truncated the ESF at a distance of 28 pixels from the edge, thus the analytical SFR of Equation 12 was convolved with a sinc$(56f)$ function before the RMSE was calculated. This eliminated any bias at low frequencies stemming from the ESF truncation, which proved to be an important consideration when optimizing the α parameters of the kernel-based and local polynomial interpolation implementations (Section 5.2 below).

The entire simulated image data set, including the expected analytical SFR at each edge orientation, is available for download from https://sourceforge.net/projects/mtfmapper/files/misc/critical_edge_simulation.zip/download. These simulated images have been prepared with great care, and can be demonstrated to accurately represent the expected system SFR in the range $[0, 2]$ cycles per pixel. Other researchers are encouraged to try this data set before attempting to create their own simulated imagery for testing slanted-edge implementations.

5.2. Optimization of parameters

Both the polynomial and kernel-based implementations accept a parameter α that controls the effective width of the support of their respective weighting kernels. A small α parameter should improve robustness against poor sampling uniformity, at the cost of distorting the SFR at high frequencies. The balance between improved robustness and distortion at high SFR frequencies was obtained through optimization.

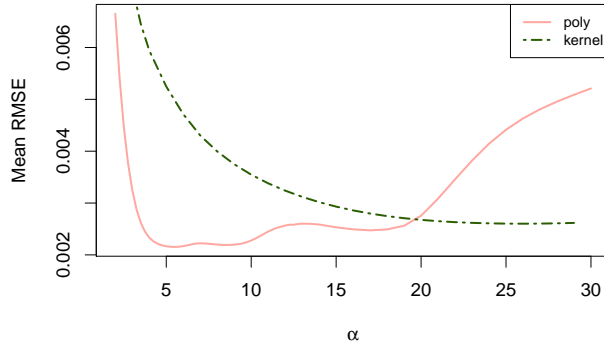A new set of 304 simulated images were generated specifically for the optimization process

Fig. 9. Mean RMSE values as a function of the parameter α.

using a slightly different sensor model, this time with a circular photosite aperture. The simulations covered both noise-free and noisy images (at a CNR of 35 dB), again using simulated lens apertures of f/4, f/11 and f/16, with the simulations running over the same set of angles listed in Section 5.1. The root mean square error (RMSE) between the measured SFR and the expected analytical SFR was computed for each simulated image over the frequency range [0, 1] cycles/pixel. The objective function for a coarse-to-fine optimization algorithm was defined as the mean of the RMSE values over the 304 images in the set. Figure 9 illustrates the value of this objective function over the range [2, 30] for both the polynomial and kernel-based interpolation implementations.

The global minimum RMSE of the polynomial-based implementation occurs at $\alpha \approx 5.5$, with some other local minima at $\alpha \approx 8.55$ and $\alpha \approx 17.0$. For the kernel-based interpolation implementation there is a single minimum at $\alpha \approx 26.0$, although this minimum is not clearly visible at the scale of the plot. It is important to note that these 'optimal' parameters are likely to change a little if a different set of simulated images were to be used, especially if the magnitude of the simulated noise is increased dramatically, however, the values of 5.5 and 26.0 for the polynomial- and kernel-based interpolation methods are reasonable starting points for further optimization.

5.3. Candidate algorithms

Three algorithms were selected for the comparisons presented in Section 5.4–5.5:

poly is an 8× oversampling implementation of the local polynomial interpolation method described in Section 4.3, with the weighting parameter α set to 5.5;

masaoka is a 32× oversampling implementation of Masaoka's method described in Section 3.3;

kernel is an 8× oversampling implementation of the kernel-based interpolation method described in Section 4.2, with the kernel parameter α set to 26.

All three these algorithms were furnished with the same ESF tail smoothing step to level the playing field when noisy images are evaluated.

5.4. Noise-free results

The goal of using noise-free simulated images is to identify any systematic bias in the estimated SFR curves, which would typically be present if the implementation was not fully corrected, or if there was some over-aggressive noise reduction present. The RMSE metric was chosen to express the deviation of the measured SFR from the expected analytical SFR; the RMSE was
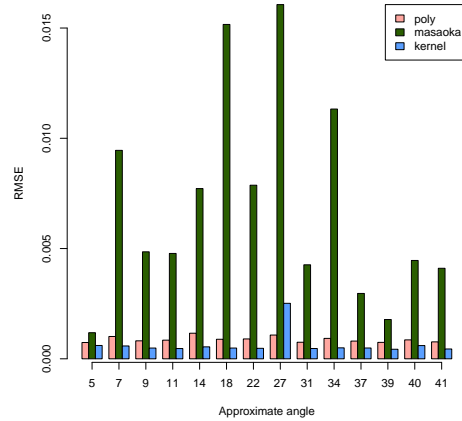
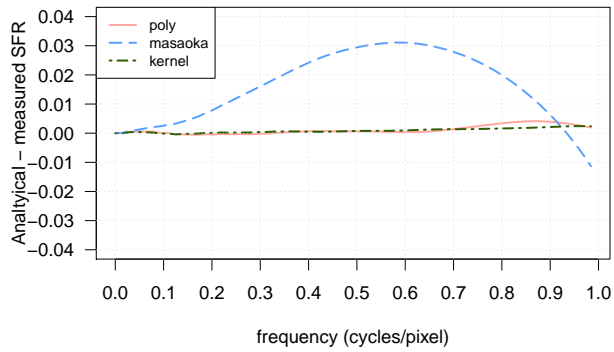Fig. 10. RMSE at various edge orientation angles, noise-free simulated images at f/11.

Fig. 11. SFR deviation from analytical, simulated f/4 image, sub-pixel edge phase 1/37 pixels, edge orientation 18.435°.

calculated only on frequencies from 0 through to 0.5 cycles/pixel to minimize the impact of aliasing that may arise at some edge angles in the f/4 and f/11 simulations.

Figure 10 illustrates the 95th percentile of RMSE over the 37 simulated edge phases, at each edge orientation, for the three candidate algorithms evaluated on the f/11 system model. The kernel-based interpolation implementation produced very consistent results, with almost no variation in performance with edge angle, except at 26.565°. Although Masaoka's method delivers competitive accuracy on the 5° edge orientation subset, the algorithm's RMSE values increase dramatically on the edge orientations with poor sampling uniformity.

Figure 11 illustrates the type of distortion experienced by Masaoka's method at some edge phase values; this distortion is not present at all 37 simulated edge phases, so this is more reflective of the worst-case behaviour. The flat difference curves produced by the polynomial and kernel-based methods demonstrate that there is very little bias in these implementations, even at a critical edge orientation angle of 18.435° where the sample uniformity is very poor, and the effective oversampling factor is only about 3.

The mean performance of the three implementations, evaluated over all edge orientations, is presented in Table 1. The standard deviation of Masaoka's method is at least an order of magnitude greater than the other two methods, demonstrating that those methods a more robust in the face of poor sample uniformity in ESF_i.

Table 1. Comparison of mean RMSE (± standard deviation) on noise-free images across all 14 angles

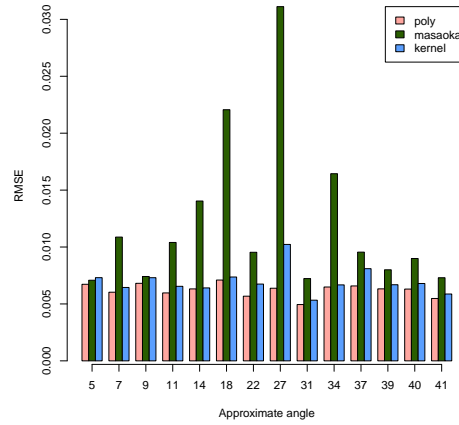| f/#	poly	masaoka	kernel
f/4	4.34e-04	6.10e-03	5.98e-04
	± 6.75e-05	± 2.36e-03	± 9.33e-05
f/11	7.55e-04	3.97e-03	5.90e-04
	± 8.33e-05	± 1.80e-03	± 3.36e-05
f/16	1.22e-03	3.40e-03	6.83e-04
	± 1.46e-04	± 1.17e-03	± 4.51e-05

Fig. 12. RMSE at various edge orientation angles, CNR 35 dB simulated images at f/4.

5.5. CNR = 35dB results

Table 2. Comparison of mean RMSE (± standard deviation) on CNR=35 dB images across all 14 angles

f/#	poly	masaoka	kernel
f/4	3.85e-03	7.43e-03	4.36e-03
	± 1.36e-03	± 2.81e-03	± 1.47e-03
f/11	4.07e-03	5.89e-03	4.49e-03
	± 1.36e-03	± 2.26e-03	± 1.49e-03
f/16	4.36e-03	5.51e-03	4.68e-03
	± 1.41e-03	± 1.91e-03	± 1.49e-03

Next, the three implementations were evaluated on the subset of the images that were perturbed with signal-independent additive Gaussian noise with a standard deviation magnitude that produces a contrast-to-noise ratio of 35 dB, representing moderately noisy images. The 95th percentile RMSE over the 37 edge phase simulations are plotted as a function of edge orientation in Figure 12, this time for the f/4 simulation. It is clear that the addition of the noise has levelled the playing field somewhat, with Masaoka's implementation now producing comparable RMSE scores on some of the edge orientations, but some edge orientations still produce errors that are three times greater than the error obtained at the 5° edge orientation. Another interesting
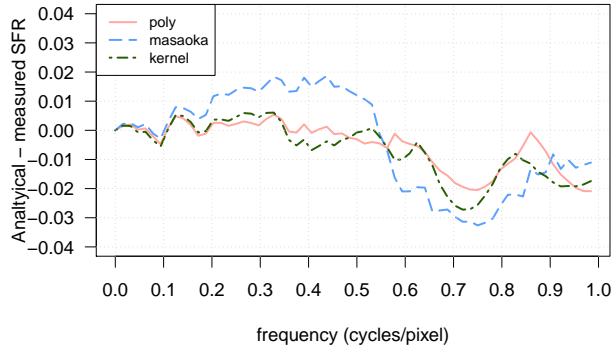
Fig. 13. SFR deviation from analytical, simulated f/16 image, sub-pixel edge phase 10/37 pixels, edge orientation $18.435°$, CNR=35 dB.

phenomenon is that the polynomial-based implementation consistently produces slightly smaller errors than the kernel-interpolation implementation, a reversal of what was observed on the noise-free images, indicating that the local polynomial fitting does indeed help to reduce the impact of noise.

Figure 13 shows a single example of the impact of noise on the SFR difference curve: all three methods exhibit local oscillations, but both the polynomial-based implementation and the kernel-interpolation method manage exhibit low bias below 0.5 cycles/pixel, only showing larger deviations (overestimation) at frequencies above 0.5 cycles/pixel. Masaoka's method still suffers from systematic bias owing to the poor sampling uniformity, even in the presence of noise.

A summary of the performance of the three methods is presented in Table 2. The difference in the RMSE standard deviations between the three methods is significantly reduced by the noise, but Masaoka's method is still trailing by between 35% and 100%, indicating that the edge orientations leading to poor sample uniformity remain a significant problem at these noise levels.

6. Discussion

The results presented in Section 5 demonstrate that it is possible to implement the slanted-edge method in such a way that it is largely insensitive to the poor sampling uniformity that results from certain edge orientations. In the case of the local polynomial fitting approach, the parameters of the robust implementation differ from those reported in the literature. For example, both Samei *et al.* [6] and Helder [3, Section 3.2] describe the use of a 4th order polynomial, with a window width of 1.7 and 2 pixels, respectively. During the development of the polynomial interpolation method described in Section 4.3 it was found that a window of 2 pixels was too narrow to obtain robust fitting at edge orientations of $26.565°$ or $45°$; the best compromise was obtained by using a 6th order polynomial, widening the window to 4.5 pixels, using ridge regression, and optimizing the α parameter.

The recent appearances of the kernel-based interpolation approach in the literature are slightly harder to compare because the information is sometimes missing or unclear. Duan *et al.* use a truncated Gaussian kernel with an apparent window length of 8 pixels, and possibly an α parameter of $3\pi \approx 9.42$ [12]. Zhang *et al.* provide no concrete values [11]. Van den Bergh proposed a Laplace distribution kernel with a scale parameter of 13; no window length is provided, but the publicly available source code reveals that the window length was quite narrow at 2.0 pixels [10]. The proposed value of $\alpha = 26$ for the implementation described in Section 4.2 produces a kernel that is narrower than that proposed by Duan *et al.*, but wider than the one proposed by Van den Bergh.

The ESF tail smoothing method described in Section 4.1 appears to be quite effective at

suppressing noise; detailed results were omitted for brevity, but traditional ESF truncation at 16 pixels from the edge, followed by a Hann window, produced RMSE values about twice that of the polynomial interpolation method with tail smoothing. As shown in Figure 13, image noise leads to an overestimate of contrast at frequencies above the extinction frequency (0.57 cycles/pixel), which is consistent with the theoretical results presented by Haefner and Burks [22], suggesting that there is still some scope for additional noise reduction, most likely stemming from noise in the central part of the ESF. It is difficult to add additional noise suppression to the central part of the ESF without distorting the SFR, for example, the Tikhonov regularization proposed by Xie *et al.* can be seen to introduce a negative bias in the measured SFR at lower frequencies [23, Figure 6]. A promising method combining the use of spline interpolation with an ESF monotonicity constraint was proposed by Zhou *et al.*, their results [16, Figure 4] demonstrate a peak SFR error of about 0.005 in the range $[0, 0.5]$ cycles/pixel, which is comparable to the robust polynomial method in Figure 13 above, especially when one takes into account that Zhou *et al.* appear to have simulated noise at a slightly higher CNR of 43.5 dB. Lastly, it is important to emphasize that the impact of angle-dependent sampling non-uniformity can be masked by image noise, without requiring any mitigation in the implementation; the relative performance of Masaoka's method on the noise-free images (Table 1) versus the noisy images (Table 2) compared to the other two implementations supports this observation.

The fact that simplistic implementations of the slanted-edge method lead to excessively poor performance at certain edge orientations cannot be emphasized enough. A recent paper by Xie *et al.* demonstrated how important it is to describe the slanted-edge implementation used in a particular study; their paper evaluated what appears to be a very poor (but completely unspecified) implementation, producing results that can at best be described as questionable [24, Figure 5].

The slanted-edge implementations described in Section 4.2 and 4.3 are robust, and can safely be used at almost any edge orientation in the range $[2, 43]°$. These implementations even work on edge orientations of $45°$, but aliasing will result if the system MTF is not bandlimited to below 0.71 cycles/pixel. This flexibility in acceptable edge angles allows full-field SFR measurements [25] to be approximated at minimal cost with a single test chart, as demonstrated in the MTF Mapper open source software package (https://sourceforge.net/projects/mtfmapper/).

7. Conclusion

Simulation results have been presented to illustrate that the uniformity of the distribution of samples within a single ESF bin in a typical slanted-edge algorithm implementation is strongly dependent on the relative edge orientation. A few critical edge orientations result in such poor uniformity that some ESF bins receive no samples at all. Local polynomial fitting and kernel-based interpolation are two well-known methods of reconstructing a regularly-sampled ESF; the results presented in this article demonstrate that with a suitable choice of parameters these algorithms can be used to obtain a very robust slanted-edge implementation that is largely insensitive to edge orientation. The efficacy of an ESF tail low-pass filtering step to reduce the impact of image noise was demonstrated, supporting the idea that this method could be used in the place of the traditional truncate-and-window approach.

References

1. S. E. Reichenbach, S. K. Park, and R. Narayanswamy, "Characterizing digital image acquisition devices," Opt. Eng. **30**, 170–178 (1991).
2. ISO 12233:2017(E), "Photography – Electronic still picture imaging – Resolution and spatial frequency responses," Standard, International Organization for Standardization, Geneva, CH (2017).
3. F. Viallefont-Robinet, D. Helder, R. Fraisse, A. Newbury, F. van den Bergh, D. Lee, and S. Saunier, "Comparison of MTF measurements using edge method: towards reference data set," Opt. express **26**, 33625–33648 (2018).
4. K. Kohm, "Modulation transfer function measurement method and results for the Orbview-3 high resolution imaging satellite," in *Proceedings of ISPRS,* (2004), pp. 12–23.
5. R. Bracewell, "Numerical transforms," Science **248**, 697–704 (1990).

6. E. Samei, M. J. Flynn, and D. A. Reimann, "A method for measuring the presampled MTF of digital radiographic systems using an edge test device," Med. physics **25**, 102–113 (1998).
7. K. Masaoka, "Accuracy and precision of edge-based modulation transfer function measurement for sampled imaging systems," IEEE Access **6**, 41079–41086 (2018).
8. H. Cramér, "On the composition of elementary errors: First paper: Mathematical deductions," Scand. Actuar. J. **1928**, 13–74 (1928).
9. P. B. Greer and T. Van Doorn, "Evaluation of an algorithm for the assessment of the MTF using an edge method," Med. Phys. **27**, 2048–2059 (2000).
10. F. van den Bergh, "Deferred slanted-edge analysis: a unified approach to spatial frequency response measurement on distorted images and color filter array subsets," JOSA A **35**, 442–451 (2018).
11. Z. Zhang, H. Feng, Z. Xu, Q. Li, and Y. Chen, "Distance-weighted modulation transfer function measurement method," in *Fifth Conference on Frontiers in Optical Imaging Technology and Applications,* vol. 10832 (International Society for Optics and Photonics, 2018), p. 108320N.
12. Y. Duan, S. Xu, S. Yuan, Y. Chen, H. Li, Z. Da, and L. Gao, "Modified slanted-edge method for camera modulation transfer function measurement using nonuniform fast fourier transform technique," Opt. Eng. **57**, 014103 (2018).
13. L. Greengard and J.-Y. Lee, "Accelerating the nonuniform fast Fourier transform," SIAM review **46**, 443–454 (2004).
14. W. S. Cleveland, "Robust locally weighted regression and smoothing scatterplots," J. Am. statistical association **74**, 829–836 (1979).
15. P. Craven and G. Wahba, "Smoothing noisy data with spline functions," Numer. mathematik **31**, 377–403 (1978).
16. Z. Zhou, Q. Zhu, H. Zhao, L. Zhang, W. Ma, and F. Gao, "Techniques to improve the accuracy of presampling MTF measurement in digital X-ray imaging based on constrained spline regression," IEEE Transactions on Biomed. Eng. **61**, 1339–1349 (2014).
17. F. Viallefont-Robinet and D. Léger, "Improvement of the edge method for on-orbit MTF measurement," Opt. express **18**, 3531–3545 (2010).
18. P. D. Burns and D. Williams, "Refined slanted-edge measurement for practical camera and scanner testing," in *IS and TS PICS Conference,* (Society for Imaging Science & Technology, 2002), pp. 191–195.
19. D. Williams and P. D. Burns, "Evolution of slanted edge gradient SFR measurement," in *Image Quality and System Performance XI,* vol. 9016 (International Society for Optics and Photonics, 2014), p. 901605.
20. V. Satopää, J. Albrecht, D. Irwin, and B. Raghavan, "Finding a "kneedle" in a haystack: Detecting knee points in system behavior," in *Distributed Computing Systems Workshops (ICDCSW), 2011 31st International Conference on,* (IEEE, 2011), pp. 166–171.
21. F. van den Bergh, "On the rendering of synthetic images with specific point spread functions," in *Twenty-Third Annual Symposium of the Pattern Recognition Association of South Africa,* (Pretoria, South Africa, 2012), p. 75.
22. D. P. Haefner and S. D. Burks, "Noise estimation of an MTF measurement," in *Infrared Imaging Systems: Design, Analysis, Modeling, and Testing XXIII,* vol. 8355 (International Society for Optics and Photonics, 2012), p. 835506.
23. X. Xie, H. Fan, A. Wang, N. Zou, and Y. Zhang, "Regularized slanted-edge method for measuring the modulation transfer function of imaging systems," Appl. optics **57**, 6552–6558 (2018).
24. X. Xie, H. Fan, H. Wang, Z. Wang, and N. Zou, "Error of the slanted edge method for measuring the modulation transfer function of imaging systems," Appl. optics **57**, B83–B91 (2018).
25. B. Dube, R. Cicala, A. Closz, and J. P. Rolland, "How good is your lens? Assessing performance with MTF full-field displays," Appl. optics **56**, 5661–5667 (2017).