

Compression of periodic images for faster phase retrieval

Amos Talmi¹ and Erez N. Ribak^{2,*}

¹Timi Technologies, Ltd., Ramat Hashofet 19238, Israel

²Department of Physics, Technion–Israel Institute of Technology, Haifa 32000, Israel

*Corresponding author: eribak@physics.technion.ac.il

Received 27 April 2010; revised 9 August 2010; accepted 20 August 2010;
posted 27 September 2010 (Doc. ID 127595); published 0 MONTH 0000

In fringe analysis, or in projected grids for shape-from-shade, deviations from periodicity are used for finding phase changes. In another example, a Hartmann–Shack sensor produces a deformed grid of spots on a camera. The gradients of the original wavefront are calculated from that image by centroiding the spots or by demodulating them. The computation time rises linearly with the number of pixels in the image. We introduce a method to reduce the size of the image without loss of accuracy prior to the calculation to reduce the total processing time. The compressed result is superior to an image measured with reduced resolution. Hence, higher accuracy and speed are obtained by oversampling the image and reducing it correctly prior to calculations. Compression or expansion coefficients are calculated through the requirement to maintain the integrity of the original phase data. © 2010 Optical Society of America

OCIS codes: 150.0150, 220.4840, 100.2650, 100.5070, 010.7350, 010.1080.

1. Analyzing Periodic Images

The image of a small source through a Hartmann–Shack (HS) sensor is a grid of light spots, separated by dark areas. Distortion of the phase of the incoming wavefront causes those spots to move from their original position [Fig. 1(a)]. The phase deviations are calculated from those movements [1,2]. The observed light spots are quantized by the pixels of the image. Smaller pixels (or a magnified image) produce more accurate positions of the spots (relative to the inter-spot distances). One prefers a very large image, with a large pitch (distance in pixels) between adjacent light spots. On the other hand, processing time is proportional to the total number of pixels in the image. Hence, one would prefer a smaller image with a shorter pitch for quicker processing. We are faced with the problem of wishing to have many pixels per period for accuracy and at the same time having fewer pixels for faster data processing. A similar con-

tradiction occurs in projected grid profilometry and in fringe analysis.

How to resolve this conflict? We propose to record the image using a large pitch between the light spots, then to reduce the image in a smart manner, and process the reduced image. The gain is enormous: using every second column in x and every second row in y would reduce the processing time by a factor of 4. If the original image has a pitch of 18, and we reduce it in each dimension by four, to get an effective pitch of $18/4 = 4.5$, we reduce the computation time by a factor of 16. Such a compression method should retain the higher accuracy in the spot positions. Simple binning of pixels does not conserve this accuracy. Instead, we suggest a compounded binning, where the intensity of the source pixel is divided among a few close destination pixels in the compressed image. The internal division among the destinations is chosen so as to conserve the HS phases. For example, let us assume that we wish to compress the one-dimensional pixel array $\{I_1, I_2, I_3, I_4, \dots\}$ by a factor of 2, into $\{I_{12}, I_{34}, I_{56}, \dots\}$. An ideal binning formula would allow one to reconstruct the original $\{I_1, I_2, I_3, \dots\}$ from the reduced $\{I_{12}, I_{34}, I_{56}, \dots\}$

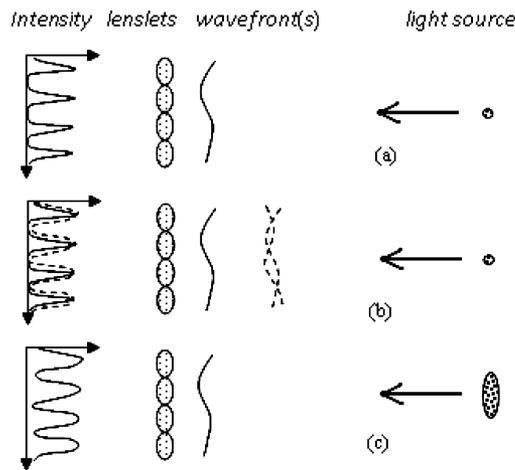


Fig. 1. (a) Wavefront slopes shift the focal spots of the lenslets. (b) An additional time-variable aberration widens the shifted focal spots during long integration—the fixed error can still be recovered. (c) Wider, even overlapping, spots can be formed from an extended object.

using an interpolation formula with few errors. The simplest binning rule is $I_{12} = I_1 + I_2$, $I_{34} = I_3 + I_4$, and a restoration formula of $I_3 = 1/2I_{34}$. This causes large restoration errors. A better procedure could be $I'_{34} = 1/4I_2 + 3/4I_3 + 3/4I_4 + 1/4I_5$ and a restoration formula of $I_3 = 3/4I'_{34} + 1/4I'_{12}$, which is more accurate (provided that the image varies slowly). In our case, we are interested in accurate positions of light spots. An ideal binning, for us, would have both $\{I_1, I_2, I_3, I_4, \dots\}$ and $\{I'_{12}, I'_{34}, I'_{56}, \dots\}$ yield identical center-of-mass (or centroid) results.

We assess the quality of the reduction by comparing the processed results of the input images versus the reduced ones. Our aim is to reduce the image in such a way that the calculated wavefront gradients will remain true. The wavefront is assumed to be sampled densely enough, such that its gradient varies smoothly between grid points. We exploit this fact to expand the gradient as a two-term Taylor series, with the error given by the third term. The reduction coefficients are tailored to keep the two terms exact.

Sometimes the intensity distribution at the lenslet focal plane is larger than the diffraction spot of a single lenslet. As an example, for calibration of permanent aberrations of a system, we have been taking a HS long exposure, averaging over many atmospheric (or other random phase) realizations [Fig. 1(b)]. Each integrated spot, by the central limit theorem, forms a wide Gaussian around the average wavefront tilt, and that average tilt is equal to the permanent local tilt. Thus the whole HS pattern can be considered a convolution between the array of permanent tilts and the wide time-averaged (atmospheric) point spread function. Thus it is possible to subtract from short-exposure wavefronts this permanent aberration, as well as to correct for non common path errors in adaptive optics. Recently, a similar method has been proposed also for phase diversity [3]. In another example, HS wavefront sensing can also be performed

for extended sources [Fig. 1(c)], whose images require a larger number of pixels per lenslet, even to the point of overlapping neighbors [4]. Thus there is a need to locate the shifts of a set of extended images, sometimes even with a constant background. Simple centroiding [1,2] fails, and one must resort to Fourier analysis [5–7] or to performing lenslet-by-lenslet matched filtering or correlation with the object to find its shift [8–10]. Such analysis is time (and memory) consuming for extended arrays with many pixels for each lenslet. In many other applications, one needs to locate the centerline of a winding fringe [1] or a projected binary grid on an object [11,12], for the purpose of wavefront sensing or shape-from-shade analysis. Here also, using fewer pixels per period would make the analysis faster.

The method we propose here is valid for Fourier or image HS phase analysis [5–7], as detailed in Section 2. It is also useful for cases where we wish to perform centroiding, directly or by correlation matching, and especially when the grid period is not a whole number of pixels. By reducing the period to an integer number of pixels, faster centroiding or correlation can be performed. (This noninteger case was solved by the Fourier methods by appropriate reference subtraction [5,7].)

Our technique is confined to images where, in Fourier space, the information content is band limited. That is, the desired phase varies slowly over the image. The HS technique measures an average displacement of the light beam per lenslet. Other phase-retrieval techniques give a single phase value per oscillation. Any finer resolution is lost.

2. Data Analysis

We explain briefly how the periodic images are analyzed for the HS wavefront sensor [5]. Similar methods are used for profiling surfaces with projected grids [11] or measuring aberrations using interference fringes [1]. While other analysis methods (such as centroiding) can be employed, harmonic methods simplify our explanation. We assume that the grid of lenslets is parallel to the x and y axes. $\mathbf{D} = (D_x, D_y)$ is the pitch (distance between spots in an undistorted image) in pixels, in the x and y directions. The grid base vectors, in the Fourier domain, are $\mathbf{k}_x = 2\pi/D_x$ and $\mathbf{k}_y = 2\pi/D_y$. We write the gradient components of the wavefront phase $\Phi(\mathbf{r})$ as $\Phi_x(\mathbf{r})$ and $\Phi_y(\mathbf{r})$. The two notations, $\mathbf{r} = (x, y)$, are used interchangeably. Thus the HS pattern intensity is $I(\mathbf{r})$ or $I(x, y)$.

Three procedures are commonly used to calculate the gradients $\Phi_x(\mathbf{r})$ and $\Phi_y(\mathbf{r})$: the phasor method, Fourier method, and centroids method. To calculate Φ_x in a grid $I(\mathbf{r})$ using the phasor method [6], we

- a. Multiply the intensity in each pixel by the complex phase in one direction, thereby creating a phasor

$$U_x(\mathbf{r}) = I(\mathbf{r}) \exp(ik_x \cdot \mathbf{r}). \quad (1)$$

b. Smooth by averaging: use a flat sliding-window $F_S(\mathbf{r})$, the window being a rectangle of size $D_x \times D_y$:

$$\begin{aligned} W_x(\mathbf{r}) &= F_S(\mathbf{r}) \otimes U_x(\mathbf{r}) \\ &= (D_x D_y)^{-1} \sum_{u=1}^{D_x} \sum_{\nu=1}^{D_y} U_x \left(u + x - \frac{1}{2} D_x, \nu + y - \frac{1}{2} D_y \right). \end{aligned} \quad (2)$$

Repeat this smoothing once or twice more. In general, get a phasor $W(\mathbf{r})$ by smoothing $U(\mathbf{r})$ with a kernel $F(\mathbf{r})$ or $W(\mathbf{r}) = F(\mathbf{r}) \otimes U(\mathbf{r})$.

c. Extract the phase of the phasor, $\Phi_x = \arg\{W(\mathbf{r})\}$. The amplitude $|W_x(\mathbf{r})|$ represents the smoothed intensity of the spots inside the pupil. It is assumed for the time being that all spots are equally bright and the amplitude is constant.

d. The gradient of the wavefront is proportional to that phase Φ_x , related through the pitch and the focal length. In other applications the surface relief is proportional to Φ_x .

e. Starting with multiplication with phase $\exp(ik_y \cdot \mathbf{r})$, repeat steps (a)–(c) to obtain Φ_y .

The double-pass flat window is equivalent to a triangular kernel, and a third pass smooths it even further into a bell shape similar to $1 + \cos(x)$. Other windows or convolution kernels were also proposed [9,10], to be used in a single pass.

As an intuitive explanation, assume that the image is composed of a single spot of light of amplitude A at (x_0, y_0) , plus some constant background C . Multiplying $I(\mathbf{r})$ by the phase $\exp(ix_0 2\pi/D)$ gives $U_x(x, y) = C \exp(ix_0 2\pi/D) + \delta(x - x_0) \delta(y - y_0) A \exp(ix_0 2\pi/D)$. Averaging over D , the constant term C oscillates and averages to zero, while the concentrated peak contributes $A \exp(ix_0 2\pi/D)$. Hence, the phase of the average is exactly $x_0 2\pi/D$ —proportional to the peak position. If the peak is two or three pixels wide, the resultant phase is the center of mass of the peak. If the image is almost periodic with period D , the next cell would have a similar peak at position $x_0 + D$ and contribute exactly the same phase; hence, one could average over a few neighboring cells and get the same result. Refer to [2] for a detailed analysis.

Performing the process by the Fourier method is essentially the same, with a slightly different smoothing kernel $f(x) = \sin(\pi x)/(x - x^3)$ [5,7]. The equivalent steps are a' and a'' replacing (a) and b' and b'' replacing (b):

a' . Calculate the (fast) Fourier transform of the image, $\tilde{I}(\mathbf{q}) = \mathcal{F}\{I(\mathbf{r})\}$.

a'' . Translate the transform by the reciprocal wave vector, $\tilde{I}_x(\mathbf{q}) = \tilde{I}(\mathbf{q} - k_x)$.

b' . Cut the high-frequency values by some low-pass filter $A(\mathbf{q})$, $\tilde{W}_x(\mathbf{q}) = A(|\mathbf{q}|)\tilde{I}_x(\mathbf{q})$.

b'' . Inverse transform the result into the spatial domain, $W_x(\mathbf{r}) = \mathcal{F}^{-1}\{\tilde{W}_x(\mathbf{q})\}$.

The resultant $W_x(\mathbf{r})$ is similar to the phasor of stage (b) above. Both methods are identical if $A(\mathbf{q})$ (step b') is a Fourier transform of the kernel $F(\mathbf{r})$. Note that the direct phasor method is quicker because it skips the transformations into the Fourier space and back.

The centroid method finds the positions of the spots by calculating their centers of mass. The gradient Φ is proportional to the displacement of the spot from the unperturbed position, divided by the focal length. Here the image is divided into cells around each unperturbed spot. The cell number (μ, ν) is of dimensions $D_x \times D_y$, centered on an unperturbed spot at $(\mu D_x, \nu D_y)$. The intensity of light in this cell is $I_{\mu, \nu}(\mathbf{r})$ where \mathbf{r} varies inside the cell. The centroid of each cell is the spot position, $\bar{\mathbf{r}}_{\mu, \nu} = \langle \mathbf{r} I_{\mu, \nu}(\mathbf{r}) \rangle / I_{\mu, \nu}(\mathbf{r})$. Using $F(\mathbf{r}) = 1$ for $-D/2 < \mathbf{r} < D/2$, $F(\mathbf{r}) = 0$; otherwise, the intensity in each cell may be written as $I_{\mu, \nu}(x, y) = I(x, y)F(\mu D_x - x, \nu D_y - y)$ and the centroid is

$$\begin{aligned} \bar{\mathbf{r}}_{\mu, \nu} &= \bar{\mathbf{r}}(\mu D_x, \nu D_y) \\ &= \frac{\sum_{x', y'} \mathbf{r}' F(\mu D_x - x', \nu D_y - y') I(x', y')}{\sum_{x', y'} F(\mu D_x - x', \nu D_y - y') I(x', y')} \\ x_{\mu, \nu} &= \lim_{w \rightarrow 0} \frac{1}{w} \arg\{F(\mathbf{r}) \otimes I(\mathbf{r}) e^{iwx}\}, \\ y_{\mu, \nu} &= \lim_{w \rightarrow 0} \frac{1}{w} \arg\{F(\mathbf{r}) \otimes I(\mathbf{r}) e^{iwy}\}. \end{aligned} \quad (3)$$

Hence, Eq. (3) is a special case of the phasor method where [step (c) above] $\Phi_x = \arg\{F(\mathbf{r}) \otimes U(\mathbf{r})\}$, with a boxcar kernel F , and w replacing $|\mathbf{k}|$ of Eq. (1). Note that (μ, ν) are integers, as well as the pixels x' and y' , while the cell boundaries are fractional; border pixels contribute partly to one cell, partly to the other. In general, the centroid method is very sensitive to slow variation in the average background illumination, and some threshold algorithm must be used. Also, the spots must lie within the unperturbed cells, namely, the displacements must be less than $D/2$.

Some may argue that the optimal smoothing $F(\mathbf{r})$ should be rotation invariant, others may choose decomposable $F(x, y) = F_1(x)F_2(y)$, and some others may proclaim that $F(\mathbf{r})$ should not be symmetric: $F(x, y) \neq F(y, x)$. For Φ_x it is $F(x, y)$, and for Φ_y it is $F(y, x)$. Our reduction scheme applies equally well to all such cases. We only require that it scales with the pitch, $F(D_x, D_y, x, y) = F(x/D_x, y/D_y)$. The examples cited below use $F(x/D_x, y/D_y) = F(x/D_x)F(y/D_y)$, because the equations are simpler in this case.

3. Image Reduction or Expansion

If the pitch, D , is an integer, the calculations are simpler. First, operations such as the moving average,

centroiding, or correlations are normally defined for integer-size windows. Fractional-size windows require interpolations. In practice, the design may aim at $D = 16$ and end with (measured) $D = 15.87$ pixels. In addition, it sometimes turns out that $D_x \neq D_y$. For instance, with resonant acousto-optic lenslets [4,13], the pitch is set by the cell dimensions. If we are going to change the image dimensions, why not aim at convenient integer pitch values $D = D_x = D_y$?

Resizing physical images is a trivial operation: use a different zoom setting of the camera lens. Once the image is digitized, resizing requires approximate interpolation formulas. Two well-known methods are widely used for reduction by an integer factor, say by R . The first one is binning, where each group of R pixels is represented by a single pixel whose value is the average of the original R values. The other one is sampling, where that value is one of the R values, usually the central. Those conventional binning methods treat every group separately. We use a different binning process, where the representing value for the group is influenced by values of pixels in nearby groups. We introduce a scheme that works with any resizing factor, integer or real, and produces much more accurate results for HS images.

We start in one dimension and define our requirements. Given a measured function, $I(t)$, $t = 1, 2, \dots, n$ create a new function $I'(g)$, $g = 1, 2, \dots, m$, magnified or demagnified M times (Fig. 2), such that the following properties are preserved:

- I. linear relation, $I'(g) = \sum_t K(g, t)I(t)$;
- II. total intensity of each individual source pixel, $\sum_t K(g, t) = 1$ for every g ; and
- III. position *and* amplitude of peaks in the input data,

$$\sum_{t=-\infty}^{\infty} I(t)e^{-(t-c)^2/2\sigma^2} \cong \sum_{g=-\infty}^{\infty} \frac{I'(g)}{M} e^{-(g-cM)^2/2\sigma^2 M^2}.$$

Property III states that the convolution of the original image with a Gaussian around c is the same as a convolution of the reduced image with a reduced Gaussian centered around a reduced position cM . If this requirement is fulfilled for Gaussian peaks, of any width σ and center c , it would be true for combinations thereof, namely any positive peaks, such as

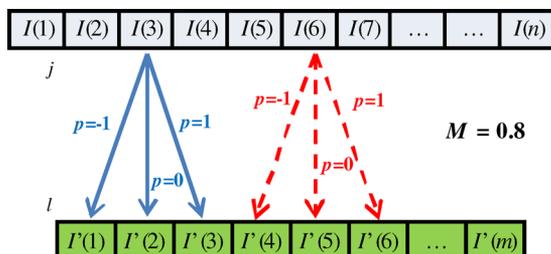


Fig. 2. (Color online) Each source pixel j contributes to three destination pixels, $l = [jM] - 1$, $[jM]$, and $[jM] + 1$. The weights of the contributions are set by conservation of location and intensity of the original pattern.

$F_S(r)$ of Section 2 step (b). If the conditions hold only for a wider Gaussian, $\sigma > \sigma_0$, one should not expect it to hold for general peaks narrower than $2\sigma_0$. To be perfect, the conditions should hold even for fine resolution steps σ . Because the digital image resolution is one pixel at best, this condition can be met only when $\sigma > 1$ and $M\sigma > 1$.

Note that property II could be derived from III, provided that III holds for any σ . But in our case, where III holds only for wider peaks, II is not redundant. The essential property of II is that the sum is a constant independent of g .

Fortunately, we are not interested in general resizing. For our purpose, it suffices that the phasors W and W' , smoothed from I and I' , are the same, namely $W(c) \cong W'(cM)$ for any position c . Even that is too stringent: only the *phase* φ of the phasor is of interest for the analysis of the HS images. Hence, we need only require minimization of

$$\delta\varphi(c) = \varphi(c) - \varphi'(cM) \equiv \arg\{W(c)\} - \arg\{W'(cM)\}. \quad (4)$$

To keep the reduction simple, we consider redirecting the intensity of each source pixel into three destination ones (one-to-two binning is insufficient). This produces both a simple and adequately accurate formula, while the cases of four and five destination pixels are discussed later. When we enlarge or reduce the array, the intensity of the source pixel t , $I(t)$, is distributed into three destination pixels at the $[tM] - 1$, $[tM]$, and $[tM] + 1$ positions, where $[x]$ is the rounded value (the closest integer) of x . These we designate as $[tM] + p$ where $p = -1, 0$, and 1 . Using the Kronecker $\delta_{i,j}$ (0 except when $i = j$) we get

$$\begin{aligned} I'(g) &= \sum_{p=-1}^1 \delta_{g, [tM] + p} \sum_{t=1}^n I(t)K(p, t) \\ &= \sum_{p=-1}^1 \sum_{t=1}^n I(t)K(p, s)\delta_{g-p, [tM]}, \end{aligned} \quad (5)$$

where we have introduced the fraction $s = s(t) = tM - [tM]$, which is important in setting the internal division of intensity among the three new pixels. Translational invariance require that the coefficients $K(p, s)$ should not depend on the pixel index t . $K(p, s)$ are normalized by condition II: $\sum_{p=-1}^1 K(p, s) = 1$. One could consider more complicated formulas, with each source contributing to four or more destinations. This differs from the conventional binning process, where a source pixel t contributes to a *single* destination at $[tM]$. Because $[tM]$ differs from tM , the conventional formula *cannot* conserve the center of mass. With three (or more) destinations, as we propose, the center, intensity, and gradient will be conserved.

For the special cases of reduction by an integer fraction, $M = 1/R$, where the reduction factor R is

an integer, the solution is simpler, as there are only R possible values of s . Enumerating them by d , $s = d/R$, we explicitly insert the Kronecker delta and get

$$I'(g) = \sum_{d=0}^{R-1} \sum_{p=-1}^1 \sum_{t=0}^{N/R} I(tR+d)K(p,d)\delta_{g,t+[dM]+p}, \quad (6)$$

and $\sum_p K(p,d) = 1$. There are exactly $3R$ coefficients K . This number of coefficients is much fewer than $3n$, required for general reduction (or expansion).

Now we wish to find the values of the coefficients K , demanding conservation of the phasors: the one calculated from the original image, $W(\mathbf{r}) = F(\mathbf{r}) \otimes U(\mathbf{r})$, should have the same phase as the one calculated from the resized image $W'(\mathbf{r}') = F'(\mathbf{r}') \otimes U'(\mathbf{r}')$ [see step (b) in Section 2]. For an arbitrary point c , those are

$$W(c) = \sum_{t=-\infty}^{\infty} I(t)e^{ikt}F\left(\frac{t-c}{D}\right),$$

$$W'(cM) = \sum_{g=-\infty}^{\infty} \frac{I'(g)}{M} e^{ikg/M} F\left(\frac{g-cM}{DM}\right),$$

where F is the smoothing kernel, and $k = 2\pi/D$. Minors or errors near the edges are ignored. Rewriting I' in terms of I [Eq. (5)] we get

$$W(c) = \sum_{t=-\infty}^{\infty} I(t)e^{ikt}F\left(\frac{t-c}{D}\right),$$

$$W'(cM) = \sum_{t,g=-\infty}^{\infty} \sum_{p=-1}^1 I(t)K(p,s)\delta_{g-p,[tM]} \times e^{ikg/M} F\left(\frac{g-cM}{DM}\right).$$

Using the definition of the fraction $s = s(t) = tM - [tM]$, we write $[tM]$ as $tM - s$, and the Kronecker delta differs from zero when $g = tM + p - s$, thus

$$W'(cM) = \sum_{t=-\infty}^{\infty} \sum_{p=-1}^1 I(t)F\left(\frac{tM+p-s-cM}{DM}\right) \times K(p,s)e^{ik(tM+p-s)/M}$$

$$= \sum_t I(t)e^{ikt} \sum_p F\left(\frac{t-c}{D} + \frac{p-s}{DM}\right) K(p,s) \times e^{+ik(p-s)/M}. \quad (7)$$

The formula for $W'(cM)$ is identical to the formula for $W(c)$, except the summation over p . The smoothing kernel is smooth itself and is expanded as a Taylor series:

$$F\left(\frac{t-c}{D} + \frac{p-s}{DM}\right) = F\left(\frac{t-c}{D}\right) + \frac{p-s}{DM} F'\left(\frac{t-c}{D}\right) + \frac{1}{2} \left(\frac{p-s}{DM}\right)^2 F''\left(\frac{t-c}{D} + \alpha\right), \quad (8)$$

where $0 < \alpha < (p-s)/(MD)$. Substituting into Eq. (7) and comparing the terms for each e^{ikh} we get from the zeroth derivative

$$I(t)e^{ikt}F\left(\frac{t-c}{D}\right) = I(t)e^{ikt}F\left(\frac{t-c}{D}\right) \sum_p K(p,s)e^{ik(p-s)/M}. \quad (9)$$

And from the first derivative

$$0 = I(t)e^{ikt}F'\left(\frac{t-c}{D}\right) \sum_p \frac{p-s}{DM} K(p,s)e^{ik(p-s)/M}.$$

The conditions are

$$1 = \sum_{p=-1}^1 K(p,s)e^{ik'(p-s)},$$

$$0 = \sum_{p=-1}^1 (p-s)K(p,s)e^{ik'(p-s)},$$

where $k' = k/M$. These are two complex conditions, solvable by two complex K coefficients, resulting in a complex reduced image. This valid mathematical option—a complex image—introduces physical complications we prefer to avoid. Writing $e^{i\rho} = \cos\rho + i\sin\rho$, $\rho = k'(p-s)$, we get four real equations for each s (fraction) value, which we wish to solve with only three real variables $K(p,s)$. Because we are interested in conserving the phase of W , while amplitudes are of secondary importance [if at all, see Eq. (4)], we choose the three real relations:

$$1 = \sum_{p=-1}^1 K(p,s) \cos[k'(p-s)], \quad 0$$

$$= \sum_{p=-1}^1 K(p,s) \sin[k'(p-s)], \quad 0$$

$$= \sum_{p=-1}^1 (p+s)K(p,s) \sin[k'(p-s)].$$

With the solution

$$K(p, s) = \frac{3p^2 - 2}{\sin[k'(p - s)]} \cdot \frac{1}{2\cotan(k's) + \cotan[k'(1 - s)] + \cotan[k'(-1 - s)]}. \quad (10)$$

Thus for each source pixel t in the original array I , we first calculate the fraction $s = tM - [tM]$. Next we calculate the three coefficients $K(-1, s)$, $K(0, s)$, and $K(1, s)$ using Eq. (10). Then we distribute $I(t)$ among the three target pixels $g = [tM] - 1$, $[tM]$, $[tM] + 1$ of the resized array I' as $I(t)K(-1)$, $I(t)K(0)$, and $I(t)K(1)$ (Fig. 2). The total $I'(g)$ is the sum of all such contributions [Eq. (6)]. K depends on k' , the wave vector in the new image $k' = k/M = 2\pi/(D, M)$.

We have chosen to conserve the real part of the intensity, rather than the intensity of the peak. This minute change, using $\cos[k'(p - s)]$ instead of 1, has a negligible effect. In general, the optimal relations would have both $1 = \sum_p K(p, s)$ and $1 = \sum_p K(p, s) \times \cos[k'(p - s)]$.

The centroid method [Eq. (3)] has the same two complex conditions, in the limit of w (or k) tending to zero [Eq. (3)]. Those complex conditions are fully satisfied by [in the limit $k \rightarrow 0$, $\exp(ikx)$ is 1]

$$\begin{aligned} K(-1, s) &= (|s| - s)/2; \\ K(0, s) &= 1 - |s|; \\ K(1, s) &= (|s| + s)/2. \end{aligned} \quad (11)$$

Note that one of the three constants is identically zero, and it is a linear interpolation two-term formula.

For five target pixels, we only provide the final values. The corresponding five-term formula is reached by using the following parameters (notice that the order of indices is inverted for clarity later):

$$\begin{aligned} a_1 &= \sin(k's)/\sin k'; \\ B_2 &= \frac{1}{2}[s \cos(k's) - a_1 \cos k']; \\ B_1 &= \frac{1}{2}a_1 - 2B_2 \cos k'; \quad A_2 = 2sB_2; \\ A_1 &= \frac{1}{2}a_1 - 4A_2 \cos k'; \\ A_0 &= \cos(k's) - 2A_1 \cos k' - 2A_2 \cos(2k'); \\ K(0, s) &= A_0; \quad K(\pm 1, s) = A_1 \pm B_1; \\ K(\pm 2, s) &= A_2 \pm B_2. \end{aligned} \quad (12)$$

4. Errors

The error $\delta\varphi$ [Eq. (4)] is the difference in the phase between the original and final arrays, $\arg\{W(c)\} - \arg\{W'(cM)\}$. It is caused by two different contributions. First, terms with the second derivative of the smoothing kernel F were ignored [Eq. (8)]. The smaller

the second derivative, the smaller is the relative error ε :

$$\varepsilon = \frac{1}{2} \left(\frac{p+s}{DM} \right)^2 F'' \left(\frac{t-c}{D} + \alpha \right) / F \left(\frac{t-c}{D} \right). \quad (13)$$

Second, the interpolation formula employs three terms, rather than five or more, and does not conserve the total number of photons, losing some accuracy. Weighted against the smaller error is the fact that interpolation formulas with fewer terms are more robust and less prone to artifacts (the higher the order, the more sensitive it is to noise).

This error estimation has two hidden assumptions—that the kernel F has finite second derivatives, and that F scales exactly with the expansion factor M . When this is not true, as is the case of a single-pass moving average (a boxcar integrator), one should use a more detailed analysis (if possible). Fortunately, this case is not of importance. Boxcar integration is used by the plain-vanilla centroid method. Modern versions [14] augment this with correlations, matching filters, or other operations. Such complicated centroid variations are best analyzed by simulations, as in [14].

The second case of interest is double-pass smoothing, used in the phasors method (Section 2). The convolution kernel for the double-pass moving average is $F(x) = D - |x|$. The second derivative {actually the second difference, $[f(x+1) + f(x-1) - 2f(x)]/2$ }, is zero except at $x = -D, 0, D$. Hence the error is

$$\begin{aligned} \delta\varphi(c) &= \arg\{W'(cM)\} - \arg\{W(c)\} \\ &= \arg \sum_{p,g,t} I(t)I(g)F \left(\frac{t-c}{D} \right) F \left(\frac{g-c+(p-s)/M}{D} \right) \\ &\quad \times K(p, s)e^{ik(g-t)+ik(p-s)/M} \\ &= \arg \sum_{p,g,t} I(t)I(g)[D - |t-c|][D \\ &\quad - |g-c-(p-s)/M|]K(p, s)e^{ik(g-t)+ik(p-s)/M}. \end{aligned}$$

Again, because $K(p, s)$ conserves linear terms, only terms with $|g-c-(p-s)/M| = 0, \pm D$ contribute any imaginary parts. An exact estimate is complicated; instead, we investigated the behavior by actual computations, or simulations.

We assume a practical situation: the pitch D is not an exact integer, and one cannot align a chosen spot to fall exactly in the center or edge of a pixel, but at some fraction of it. This means that the original image includes—right from the start—some sampling and quantization errors, in addition to noise-induced errors. The light distribution is sampled pixel by

pixel, introducing a quantization of space. The distribution of light inside the pixel area is never known. Once we shrink this image, the situation worsens: quantization of space is coarser and may cause larger errors, but by how much? To quantify that, we ran simulations with three-term and five-term formulas, as well as with different smoothing kernels. The results (Fig. 3) show that the second-derivative error is the dominant one, and formulas with three terms are good enough. For the tests, we simulated the problem by a series of maxima, each a Gaussian of width σ pixels ($\sigma = 1.5$, a light spot about four pixels wide), with the centers of the Gaussians located at $c(\kappa) = \kappa D + \kappa^2 d_1 + \kappa^3 d_2$, where $d_1 \approx 0.1D$ and $d_2 \approx 0.2D$. Thus the distances of the peaks are shifted by a third-order polynomial. This series was digitized at integer values $t = 1, 2, 3, \dots$ with $I(tj_0)$ being all the photons in the interval $t_0 - 0.5 < t < t_0 + 0.5$. We calculated the phasor function W by multiplying I with a rising phase (step b) and smoothing by a moving average passing two, three, or four times over the data [step (c)]. Next, we compressed I into I' and calculated W' from I' . The plot shows the phase differences $\delta\varphi = \arg(W) - \arg(W')$ [Eq. (4)]. Plotting the phase differences between original and compressed data, pixel by pixel, shows them to behave as alternating signs with zero expectation value. As expected, the error drops with the number of smoothing stages.

5. From One to Two Dimensions

In two dimensions, each source pixel $I(x, y)$, which would have projected into $I'(xM, yM)$, actually contributes to all neighboring pixels $\{[xM] + p_x, [yM] + p_y\}$ with $\mathbf{p} = p_x, p_y = -1, 0, 1$. Two phasors are to be conserved, $I(\mathbf{r}) \exp(i\mathbf{q}_1 \cdot \mathbf{r})$ and $I(\mathbf{r}) \exp(i\mathbf{q}_2 \cdot \mathbf{r})$, with the

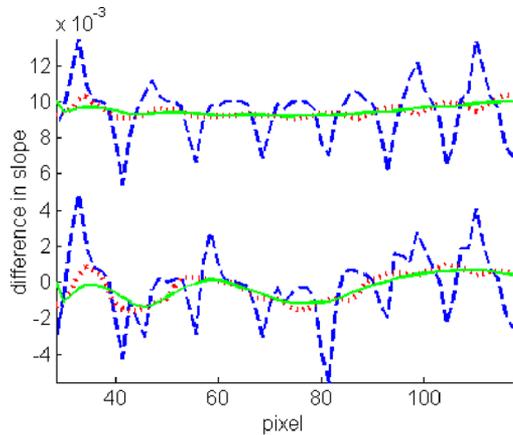


Fig. 3. (Color online) Phase errors: differences of phase between resized and original vectors, along the array of Hartmann spots. The parameters are as follows: pitch, $D = 13$ pixels; spot width, $\sigma = 1.5$ pixels; and magnification $M = 0.7$, with the phase gradient growing over the array from 0 to 0.1 rad. The number of smoothing passes is two (blue, broken curve), three (red, dotted curve), and four (green, full curve). Three-term fits (lower curves) are slightly less accurate than five-term fits (curves shifted up by 0.01). The difference is that slope is expressed in radians (the phase of the phasor) proportional to the gradient of the wavefront.

two reciprocal wave vectors \mathbf{q}_1 and \mathbf{q}_2 . If the directions are purely orthogonal, along x and y , and if the smoothing kernel F can be decomposed into $F(\mathbf{r}) = F(x)F(y)$, then the coefficients $K(p_x, s_x, p_y, s_y)$ can be decomposed into $K(p_x, s_x)K(p_y, s_y)$. The resizing can be carried out by first resizing only the x dimension, followed by a purely y resizing. Note that the two operations are independent, each with its own magnification factor M and wave vector \mathbf{q} . In the analysis of fringes and projected grids, which happen along one dimension, a different magnification is clearly preferred depending on data density. We did not consider here other directions, such as the diagonal case Φ_{x+y} and Φ_{x-y} . In these cases, different reduction coefficients should be used. For a general case of linear spot grids, even not orthogonal, the whole set of coefficients $K(p_x, s_x, p_y, s_y) \equiv K(\mathbf{p}, \mathbf{s})$ must be calculated by solving a two-dimensional set of equations:

$$1 = \sum K(\mathbf{p}, \mathbf{s}) e^{-i\mathbf{q}_1 \cdot (\mathbf{p} - \mathbf{s})};$$

$$1 = \sum K(\mathbf{p}, \mathbf{s}) e^{-i\mathbf{q}_2 \cdot (\mathbf{p} - \mathbf{s})};$$

$$0 = \sum (\mathbf{p} - \mathbf{s}) K(\mathbf{p}, \mathbf{s}) e^{-i\mathbf{q}_1 \cdot (\mathbf{p} - \mathbf{s})};$$

$$0 = \sum (\mathbf{p} - \mathbf{s}) K(\mathbf{p}, \mathbf{s}) e^{-i\mathbf{q}_2 \cdot (\mathbf{p} - \mathbf{s})},$$

where all summations are now over $\mathbf{p} = p_x, p_y = -1, 0, 1$. These are 12 real, scalar constraints, which in general are not independent. We prefer to retain the relations with the imaginary parts because they influence the phase (rather than the less important amplitude) of the phasors, together with some normalization:

$$0 = \sum K(\mathbf{p}, \mathbf{s}) \sin[\mathbf{q}_1 \cdot (\mathbf{p} - \mathbf{s})];$$

$$0 = \sum K(\mathbf{p}, \mathbf{s}) \sin[\mathbf{q}_2 \cdot (\mathbf{p} - \mathbf{s})];$$

$$0 = \sum K(\mathbf{p}, \mathbf{s}) (p_x - s_x) \sin[\mathbf{q}_1 \cdot (\mathbf{p} - \mathbf{s})];$$

$$0 = \sum K(\mathbf{p}, \mathbf{s}) (p_x - s_x) \sin[\mathbf{q}_2 \cdot (\mathbf{p} - \mathbf{s})];$$

$$0 = \sum K(\mathbf{p}, \mathbf{s}) (p_y - s_y) \sin[\mathbf{q}_1 \cdot (\mathbf{p} - \mathbf{s})];$$

$$0 = \sum K(\mathbf{p}, \mathbf{s}) (p_y - s_y) \sin[\mathbf{q}_2 \cdot (\mathbf{p} - \mathbf{s})];$$

$$1 = \sum K(\mathbf{p}, \mathbf{s}) \{ \cos[\mathbf{q}_1 \cdot (\mathbf{p} - \mathbf{s})] + \cos[\mathbf{q}_2 \cdot (\mathbf{p} - \mathbf{s})] \} / 2.$$

6. Examples

To test the resizing method, we produced tens of random wavefront realizations, each distorted according

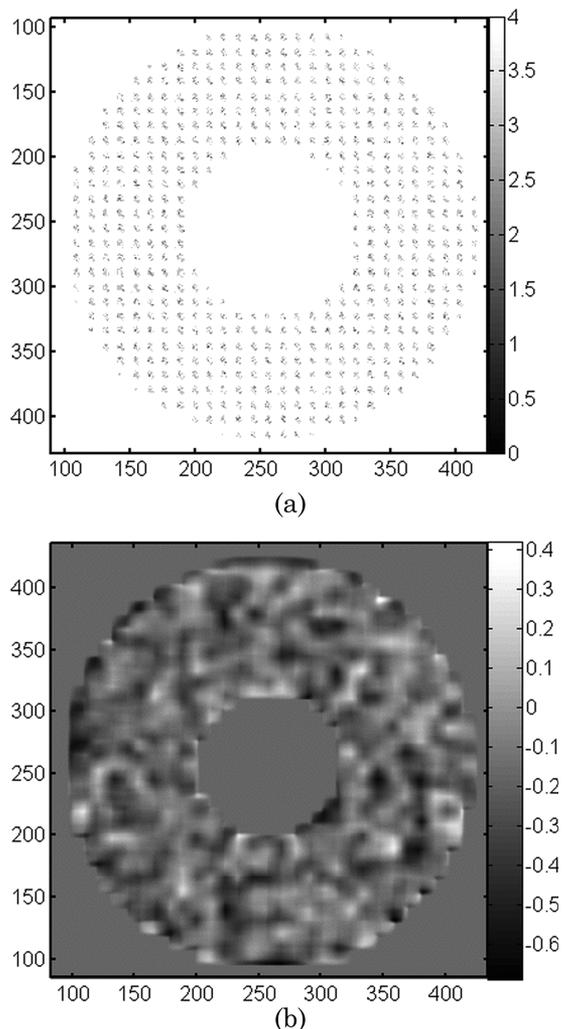


Fig. 4. Hartmann pattern, including Poisson noise, with (a) 0–6 photons per pixel (negative image), the source being an asterism, the reference a long exposure of the same asterism. The data were extended beyond the edge [6], and the (b) x phase slope was calculated. On the left and bottom are pixel numbers; on the right are (a) photons and (b) phase (radians).

to the Kolmogorov spectrum and various Fried r_0 values, and sampled by a centrally obscured telescope. These wavefronts were then split into different rectangular grids of lenslets, and each one was propagated to focus on the detector and squared to create the local intensity. These focal patterns were convolved with a simple asterism of a few nearby delta functions, each multiplied by a different factor. As a result, we obtained a set of blurry focal spots. We added Poisson noise to the intensity at different levels, from a few to thousands of photons per lenslet. We present here a faint source case under relatively strong turbulence [Fig. 4(a)].

We assumed that the wavefront sensor has its own internal aberrations. In order to calibrate the result, we also took many realizations of the atmosphere through this aberrating sensor using the same asterism. We added up these Hartmann patterns to create more aberrated focal spots, which, on average, follow

the wavefront sensor internal aberrations, as in Fig. 1.

For each wavefront to analyze, we resized the Hartmann pattern according to the procedure given above, for many reduction values. In the example shown, we show how the method also works for different compression values in x and y . Both the original and resized arrays were processed to find the wavefront slopes according to the demodulation steps (a)–(e) in Section 2. The resultant phasor was multiplied by the conjugate of the reference wavefront, processed in the same way, to remove the instrumental aberrations [Fig. 4(b)]. Because step (b) requires smoothing, we also tested the effect of the number of repeated smoothing iterations. Smoothing (or low-pass filtering in the Fourier domain) and image resizing both affect the final result, and thus they cannot be discussed separately. This is further detailed in the next section.

Thus each wavefront was processed four times, with two or four smoothing iterations, and with three or five interpolation coefficients. These were compared to the nonresized images, and the root mean square (RMS) of the phase difference between the original and compressed image was calculated. It never exceeded a few percent of the original wavefront slopes, and usually it was much lower (Figs. 5 and 6). Brighter sources produced even fewer errors, as did less turbulent atmospheres.

7. Discussion and Summary

We have proved that, with a smart binning formula, one may apply the resizing scheme on nearly periodic images and maintain their phase. Unlike normal binning, which is confined to integer reduction factors, with smart binning the magnification/reduction may be any real value. The magnification or compression scheme is independent of the smoothing kernel, but the binning-induced errors depend on it: the smoother the kernel, the smaller the errors. The same smart binning is valid for any convolution kernel as well as for centroiding and Fourier domain analysis.

The reduction process requires first a one-time calculation of the $3n$ coefficients for each dimension, then performing a matrix multiplication of the input array by these coefficients [Eq. (6)]. However, since only the nearest neighbors are being affected, this can be presented as a near-diagonal matrix, which is quicker to multiply by. For an array of size $n_x n_y$, the number of calculations is $6n_x n_y$ add/multiply operations, plus $3n_x + 3n_y$ calculations of the coefficients. Reduction by an integer number $R = 1/M$ in each of the two dimensions drops the number of calculations to $2R + 6n_x n_y$. This is a single calculation, and it has to be weighed against M^2 saving in each of the next processing stages. To further accelerate the analysis for fixed arrays, the precalculated coefficients can be implemented in hardware, as well as the Fourier analysis.

A good way of reducing the errors is by using third- or fourth-order sliding averages, instead of

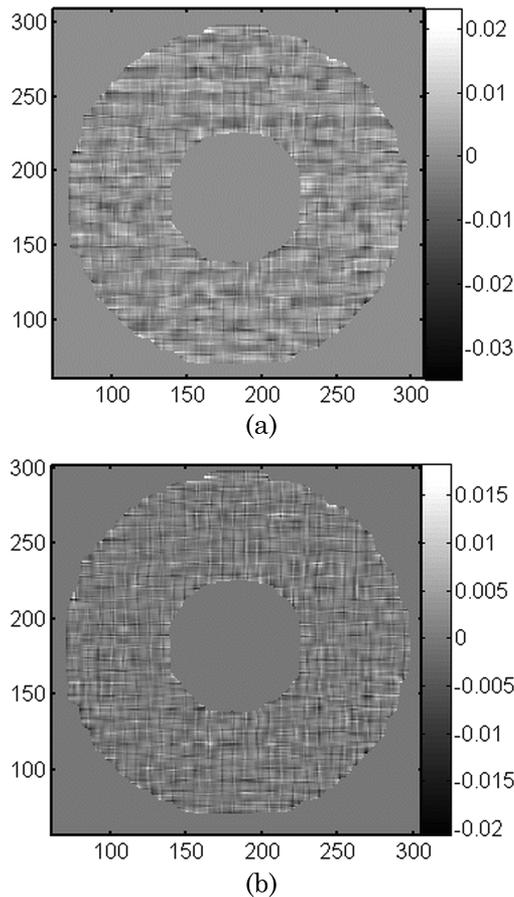


Fig. 5. Errors introduced by shrinking. The Hartmann array of Fig. 4 was shrunk by 0.71 in x and 0.61 in y (notice coordinates). The results are too similar to Fig. 4(b) for changes to be visible. Instead, the differences are drawn for (a), (b) a two-pass smoothing (varying gray scale), (a) three-term interpolation, and (b) five-term interpolation. The corresponding RMS of the differences in phase are 0.0024 and 0.0015 rad. Scale bar is expressed in radians.

the second-order one we have advocated before (Fig. 5). The cost is a small loss of spatial resolution and of time. The gain is a drop in binning-induced errors, as well as elimination of phase jumps and vortices due to defective lenslets and phasors with negligible intensity. A blocked lenslet transmits no light, causing phasors dominated by random noise; a half-blocked lenslet causes a nonsymmetric light spot and errors in spot position [15]. A third-order smoothing effectively replaces a blocked lenslet with a ghost of its neighbor lenslets. In general, it averages the lenslet intensity with ghosts of its neighbors. Fourth-order smoothing sums farther neighbors, too. In the corresponding Fourier analysis (steps a' , b' of Section 2) a low-pass filter plays a similar role. We used the window function $\cos^2(q_x)\cos^2(q_y)$.

There are a few pitfalls to note and avoid:

1. Applying the same smoothing kernel for both the reduced and the unreduced images, namely, the same function of the pitch D . This is natural if one smooths with a Gaussian or any other mathematical filter that is defined for real numbers. In the case of

the sliding sum [steps (a), (b) above], or centroiding, where operations are normally defined for an integer number of pixels, this may require interpolation. For instance, if the original image had been reduced by 2.27×2.46 , and if the reduced image is smoothed by summing over 4×4 pixels, this is equivalent to smoothing the original image by 9.04×9.84 pixels. In general, simple smoothing over some (integer) $P \times Q$ window in the compressed image corresponds to a computation-intensive smoothing over $MP \times MQ$ window in the original image.

2. The position of the average, for instance the average of pixels $j + 1$ through $j + 14$ should be attributed to $j + 7.5$ (which is then compressed into a pixel $[(j + 7.5)M]$).

3. Edge problems: edges need to be excluded or images extended before resizing. The same applies to smoothing operations. If there is no information near the edge (the HS pattern is smaller than the measured array), this is trivial: one extends the image by adding rows and columns of zeros around, then discards results influenced by those extra pixels. Otherwise, one should expand the HS image smartly, adding extra lines/columns whose values depend on the known boundary ones [6,7,16]. In general, reduction by a factor R ($R > 1$) requires extra T lines and columns on each side, where $T = \text{ceil}(1.5R)$ for the three-term formula. Three-pass smoothing adds $1.5D$ to T ; four-pass smoothing adds $2D$.

4. The case of zero amplitude. Measured HS images are usually composed of an aperture with spots of light, surrounded by a region with only background noise. Any calculation of spot shifts outside the aperture is doomed to failure: there are no lenslets there. Some of the lenslets may be partially occluded by the aperture and provide limited information [15]. Outside the aperture, the amplitude of the phasor W drops to zero and the phase is pure noise. This is in contrast with our assumption that the phasor has constant amplitude. We employed a mechanism to flag points with phasors smaller (in amplitude) than a fraction of the average amplitude. This was used in Fig. 4 to set the aperture borders as well. The same caution applies to cases of wavefront vortices or simply dirty lenslets, where the amplitude also drops to zero. The unreliable pixels were given a constant value, equal to the average of all others. In shape-from-shade problems (profiling by projected grids), the same problem occurs when changing facets, where the grid lines might seem to run in a different direction, helping also to find the facet edges.

5. Low signal cases. In stellar adaptive optics, where the number of photons is extremely limited, the division of photons into many pixels increases both shot noise and readout noise, and the benefits of large-format detectors are limited. This factor is less important now, with the recent introduction of very low noise detectors and with the need for position accuracy brought about by more pixels per spot. All large new telescopes (with diameter > 25 m) are planned to have more than eight pixels per spot.

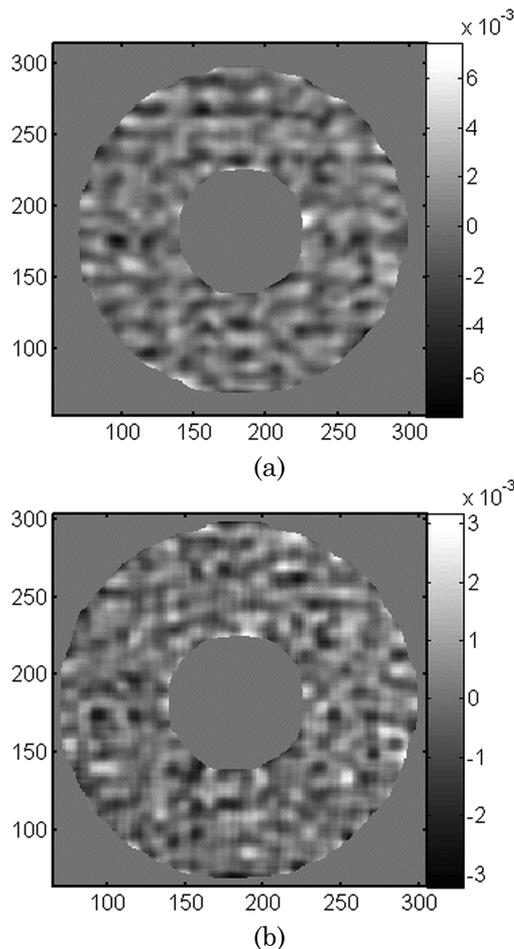


Fig. 6. Errors introduced in reduction from Fig. 4 by 0.71 in x and 0.61 in y , as in Fig. 5, but with further smoothing and thus lower errors. The difference between the original and the reduced slope is shown for (a), (b) a four-pass smoothing (varying gray scale), (a) three-term interpolation, and (b) five-term interpolation. The corresponding RMS differences are 0.0014 and 0.0012 radians.

6. Deviations from periodicity. We choose an optimal smart binning formula, in the sense that by using appropriate procedures, both the original image and the reduced one produce the same results. If the period is known in advance to be nonconstant, such as in closed fringes or a HS pattern of a large defocus, then the compression coefficients will not be global, but they might still be calculated locally.

Our method compresses grossly oversampled images, and it should be compared to other compressions methods applied to images with similar information content, namely, slowly changing phases. The oversampling comes about because $D \times D$ pixels (where D is the pitch) are used to record the x - y position and the intensity of a single peak. That is, as many as D^2 values are used where three should suffice. Ng and Ang [17] propose three figures of merit, to quantify compressions:

1. compression ratio, in pixels: (number of compressed pixels)/(number of original pixels);

2. compression ratio, in bits or bytes: (length of compressed file)/(length of original file); and
 3. root mean square error: $|Q(\text{compressed}) - Q(\text{original})|$, where Q is the desired quantity, such as phase versus position.

Using this terminology, our pixel compression ratio is exactly M^2 , where M is the magnification ratio. In bytes, our compression is $1/2M^2$, because the intensity in the new image must use 16, not 8, bits. The range of possible magnifications is limited by Nyquist: the pitch in the compressed image must be $D' > 2$, and practically $D' > 3$. Under such limitation, the compression ratio is $D^2/9$.

What about errors? We constructed our compression formula to produce zero errors in the ideal case. We tested our method with noisy data, and found it to produce results $Q(\text{compressed})$ very close to the original Q , with ΔQ much smaller than the expected noise in Q (Section 3 and Fig. 2). Figure 7 summarizes many simulations: the RMS of the difference (peaks in the reduced image minus their location before reduction). The figure demonstrates that very high levels of noise in the input have some influence over the produced differences. In general, five terms are better than three terms. But spreading the information over more than a single cell, while the phase varies largely from cell to cell, is not recommended: that is, using a five-term formula for a case of output pitch of three pixels (a full cell is 3×3 pixels), produces higher errors than a three-term formula. Thus, the five-term formula should only be used when $D' \geq 4$.

Alfalou and Brosseau [18] considered a much wider scope of compressions, both before and after

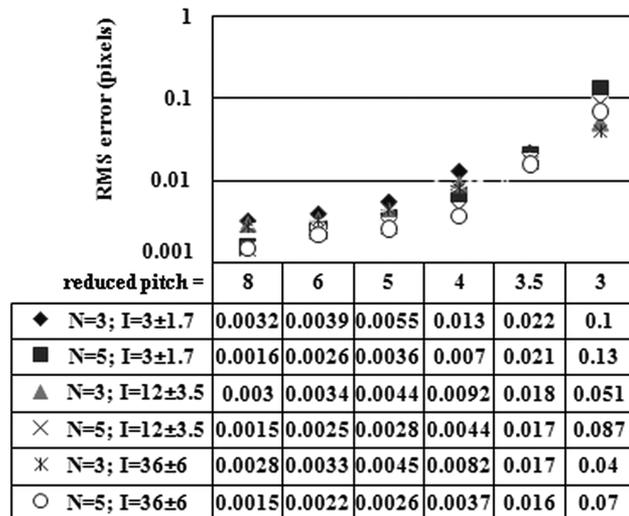


Fig. 7. Root mean square shift in the location of the peaks before and after reduction, in differently reduced images and different noise levels. An input image (as in Figs. 4–6) with a pitch of 16 pixels was reduced to various pitches as marked, both with $N = 3$ and $N = 5$ terms formulas. The indicated peak intensities I are realizations of $3 \pm \sqrt{3}$, $12 \pm \sqrt{12}$ and $36 \pm \sqrt{36}$ photons. Results were calculated using a four-pass smoothing, and are expressed in pixels of the reduced images.

processing, including schemes of reducing the images for storage and transmission, such as JPEG or JPEG2000 specially modified to the task at hand. In this respect, our work is very limited: after processing, the only relevant information is the position of the peak. That is, a compression ratio of D^2 . We dealt only with a preprocessing operation, so we did not improve, nor impair, this D^2 ratio. Perhaps a special version of wavelet compression, constructed to conserve both amplitude and position of peaks, could do better than our method. The gain, however, could not be larger than a factor of 2: at the very least, four pixels must define a peak in x and y , while our technique allows for nine pixels per peak.

A test version of the program is available at <http://physics.technion.ac.il/~eribak/reduce>.

This work was supported in part by the France–Israel Research Cooperation fund, the Israel Science Foundation (ISF), and by the Ministry of Industry Magnet program. E. R. wishes to thank C. Dainty and the Walton Fellowship of Science Foundation Ireland. The comments of anonymous reviewers helped improve this presentation.

References

1. D. Malacara, ed., *Optical Shop Testing* (Wiley, 1978).
2. R. K. Tyson, *Principles of Adaptive Optics*, 2nd ed. (Academic, 1998).
3. L. M. Mugnier, J.-F. Sauvage, T. Fusco, A. Cornia, and S. Dandy, "On-line long-exposure phase diversity: a powerful tool for sensing quasi-static aberrations of extreme adaptive optics imaging systems," *Opt. Express* **16**, 18406–18416 (2008).
4. E. N. Ribak, "Harnessing caustics for wave-front sensing," *Opt. Lett.* **26**, 1834–1836 (2001).
5. Y. Carmon and E. N. Ribak, "Phase retrieval by demodulation of a Hartmann–Shack sensor," *Opt. Commun.* **215**, 285–288 (2003).
6. A. Talmi and E. N. Ribak, "Direct demodulation of Hartmann–Shack patterns," *J. Opt. Soc. Am. A* **21**, 632–639 (2004).
7. C. Canovas and E. N. Ribak, "Comparison of Hartmann analysis methods," *Appl. Opt.* **46**, 1830–1835 (2007).
8. T. Berkefeld, D. Soltau, and O. von der Luhe, "Multi-conjugate adaptive optics at the Vacuum Tower Telescope, Tenerife," *Proc. SPIE* **4839**, 66–76 (2002).
9. L. Poyneer, "Scene-based Shack–Hartmann wave-front sensing: analysis and simulation," *Appl. Opt.* **42**, 5807–15 (2003).
10. S. Özder, Ö. Kocahan, E. Coskun, and H. Göktas, "Optical phase distribution evaluation by using an S-transform," *Opt. Lett.* **32**, 591–593 (2007).
11. D. M. Meadows, W. O. Johnson, and J. B. Allen, "Generation of surface contours by moiré patterns," *Appl. Opt.* **9**, 942–947 (1970).
12. G. S. Spagnolo, G. Guattaria, C. Sapia, D. Ambrosini, D. Paoletti, and G. Accardo, "Contouring of artwork surface by fringe projection and FFT analysis," *Opt. Lasers Eng.* **33**, 141–156 (2000).
13. A. Stup, E. M. Cimet, E. N. Ribak, and V. Albanis, "Acousto-optic wave front sensing and reconstruction," *Appl. Opt.* **48**, A1–A4 (2009).
14. S. Thomas, T. Fusco, A. Tokovinin, M. Nicolle, V. Michau, and G. Rousset, "Comparison of centroid computation algorithms in Shack–Hartmann sensor," *Mon. Not. R. Astron. Soc.* **371**, 323–336 (2006).
15. Y. Carmon and E. N. Ribak, "Centroid distortion of a wave-front with varying amplitude due to asymmetry in lens diffraction," *J. Opt. Soc. Am. A* **26**, 85–90 (2009).
16. A. Talmi and E. N. Ribak, "Wavefront reconstruction from its gradients," *J. Opt. Soc. Am. A* **23**, 288–97 (2006).
17. T. Ng and K. Ang, "Fourier-transform method of data compression and temporal pattern analysis," *Appl. Opt.* **44**, 7043–7049 (2005).
18. A. Alfalou and C. Brosseau, "Optical image compression and encryption methods," *Adv. Opt. Photon.* **1**, 589–636 (2009).