

Supplement for Computational Speckle Pattern Interferometry

Shengxi Wu¹ Sophia Yang² Dorian Chan¹ Matthew O'Toole¹
¹Carnegie Mellon University ²University of Toronto

8. Derivations for Motion

In the presence of motion during a camera's exposure, the intensity is given by the integral of the expression in Eq. (6), as follows:

$$\begin{aligned} l_{jk} &= \frac{1}{T} \left(\int_0^T \mathbf{v}_j \cdot \mathbf{u}_k(t) \right) dt \\ &= \mathbf{v}_j \cdot \left(\frac{1}{T} \int_0^T \mathbf{u}_k(t) \right) dt \\ &= \mathbf{v}_j \cdot \left[\int_0^T \frac{\cos(\phi_k(t))}{T} dt \quad \int_0^T \frac{\sin(\phi_k(t))}{T} dt \quad 1 \right]. \end{aligned} \quad (13)$$

where the phase $\phi_k(t)$ for frame k changes as a function of time. Therefore, when motion is present, our procedure simply solves for an average of unit-length phasors.

Linear motion. We can now analyze the specific effect of different types of motion. For example, let $\phi_k(t) = \phi_a + \frac{t}{T}(\phi_b - \phi_a)$, *i.e.*, an object moves at a fixed velocity from ϕ_a to ϕ_b during the camera's exposure period T . We first focus on the real component of the resulting phasor:

$$\begin{aligned} \text{Re}\{\vec{u}_k\} &= \int_0^T \frac{\cos(\phi_k(t))}{T} dt \\ &= \int_0^T \frac{\cos(\phi_a + \frac{t}{T}(\phi_b - \phi_a))}{T} dt \\ &= \frac{\sin(\phi_a + \frac{t}{T}(\phi_b - \phi_a))}{\phi_b - \phi_a} \Big|_{t=0}^T \\ &= \frac{\sin(\phi_b) - \sin(\phi_a)}{\phi_b - \phi_a} \\ &= \frac{2 \cos\left(\frac{\phi_b + \phi_a}{2}\right) \sin\left(\frac{\phi_b - \phi_a}{2}\right)}{\phi_b - \phi_a}. \end{aligned} \quad (14)$$

Similarly, imaginary component has the form:

$$\text{Im}\{\vec{u}_k\} = \frac{2 \sin\left(\frac{\phi_b + \phi_a}{2}\right) \sin\left(\frac{\phi_b - \phi_a}{2}\right)}{\phi_b - \phi_a}. \quad (15)$$

Combining both components produces the following:

$$\begin{aligned} \vec{u}_k &= \frac{2 \sin\left(\frac{\phi_b - \phi_a}{2}\right)}{\phi_b - \phi_a} \exp\left(i \frac{\phi_a + \phi_b}{2}\right) \\ &= \text{sinc}\left(\frac{\phi_a - \phi_b}{2}\right) \exp\left(i \frac{\phi_a + \phi_b}{2}\right). \end{aligned} \quad (16)$$

Note that this phasor also models any other motion where phasors are uniformly distributed between ϕ_a and ϕ_b , such as a sawtooth wave and a triangle wave. General non-oscillatory motions can also be approximated by this phasor, by assuming that any change in velocity is small during the camera's exposure.

When the function $\text{sinc}(x)$ becomes negative, the phasor angle rotates by π . As a result, there are two possible interpretations of the phasor angle, since $-ae^{i\theta} = ae^{i(\theta+\pi)}$. To avoid this ambiguity, one can constrain $|\phi_a - \phi_b| \leq 2\pi$.

It is also possible to recover velocity from phasor magnitudes, provided that we know the camera's exposure period and how to map phase to metric displacement.

Sinusoidal motion. Another common type of motion is sinusoidal oscillations between ϕ_a and ϕ_b , *i.e.*, $\phi_k(t) = \frac{\phi_a - \phi_b}{2} \sin(\omega t + \theta) + \frac{\phi_a + \phi_b}{2}$. Here, we assume a frequency ω such that an integer number of periods fits within the camera's exposure, *i.e.*, $\omega = \frac{2\pi n}{T}$ for some integer $n \in \mathbb{Z}$. As a result, the phase θ does not impact the measurements, and we can assume $\theta = 0$. We first focus on the real component of the resulting phasor:

$$\begin{aligned} \text{Re}\{\vec{u}_k\} &= \int_0^T \frac{\cos(\phi_k(t))}{T} dt \\ &= \int_0^T \frac{\cos\left(\frac{\phi_a - \phi_b}{2} \sin(\omega t) + \frac{\phi_a + \phi_b}{2}\right)}{T} dt \\ &= \int_0^T \frac{\cos\left(\frac{\phi_a - \phi_b}{2} \sin(\omega t)\right) \cos\left(\frac{\phi_a + \phi_b}{2}\right)}{T} dt \\ &\quad - \int_0^T \frac{\sin\left(\frac{\phi_a - \phi_b}{2} \sin(\omega t)\right) \sin\left(\frac{\phi_a + \phi_b}{2}\right)}{T} dt, \end{aligned} \quad (17)$$

where the second integral is zero because the positive and negative areas cancel out. Finally:

$$\begin{aligned} \text{Re}\{\vec{u}_k\} &= \int_0^T \frac{\cos\left(\frac{\phi_a - \phi_b}{2} \sin(\omega t)\right) \cos\left(\frac{\phi_a + \phi_b}{2}\right)}{T} dt \\ &= J_0\left(\frac{\phi_a - \phi_b}{2}\right) \cos\left(\frac{\phi_a + \phi_b}{2}\right), \end{aligned}$$

where $J_0(x)$ is the zero-order Bessel function of the first kind, which is defined by the Hansen-Bessel formula [6]:

$$J_0(x) = \frac{1}{\pi} \int_0^\pi \cos(x \sin(\tau)) d\tau. \quad (18)$$

This expression is also related to the derivations provided by Wang *et al.* [5] for sensing vibrations with ESPI. The imaginary component has a similar expression:

$$\text{Im}\{\vec{u}_k\} = J_0\left(\frac{\phi_a - \phi_b}{2}\right) \sin\left(\frac{\phi_a + \phi_b}{2}\right). \quad (19)$$

Combining the real and imaginary components produces the following phasor:

$$\vec{u}_k = J_0\left(\frac{\phi_a - \phi_b}{2}\right) \exp\left(i \frac{\phi_a + \phi_b}{2}\right). \quad (20)$$

Note that similar ambiguities occur when the Bessel function $J_0(x)$ turns negative. To avoid this, one can restrict motion such that $|\phi_a - \phi_b| \leq 4.8096$.

9. Sensitivity of In-Plane Arrangement

Recall that, for the in-plane arrangement, the sensitivity of the system can be defined by the magnitude of vector \mathbf{k} , where $\mathbf{k} = \hat{\mathbf{l}}_{s1} - \hat{\mathbf{l}}_{s2}$ is the difference of the two incident lighting directions at a particular scene point.

Conversion from phase to displacement is derived from the relationship between phase change and optical path difference, from Eq. (2):

$$\Delta\phi = \frac{2\pi}{\lambda} \mathbf{k} \cdot \Delta\mathbf{x}, \quad \|\mathbf{k}\| = \|\hat{\mathbf{l}}_{s1} - \hat{\mathbf{l}}_{s2}\| \approx 2 \sin\left(\frac{b}{2D}\right), \quad (21)$$

where b is the physical separation between the laser sources (baseline), and D is the object distance. By small-angle approximation, the component of displacement along the sensitivity direction \mathbf{k} can be expressed as

$$\Delta x \approx \frac{\lambda D}{2\pi b} \Delta\phi. \quad (22)$$

In Fig. 9 and Fig. 10, we use a baseline of 5.2 cm and 9.2 cm, with the target placed at a depth of 85 cm. Displacing the target by 1 radian in the lateral direction corresponds to a 1.43 μm shift for 5.2 cm, and 0.81 μm shift for 9.2 cm. These values are used when converting unwrapped phase to displacement.

10. Sensitivity of Out-of-Plane Arrangement

For out-of-plane arrangement, the magnitude of vector \mathbf{k} is now defined by $\mathbf{k} = \hat{\mathbf{l}}_{s1} + \hat{\mathbf{c}}$. Assuming the vectors point in the same direction, *i.e.*, $\hat{\mathbf{l}}_{s1} = \mathbf{k}$, this implies that 1 radian corresponds to a 43.6 nm shift—approximately 18 \times more sensitive than the large baseline in-plane arrangement. As a result, environmental noise can significantly impact measurements.

11. Displacement from Unknown Phase Shifts

We construct a matrix \mathbf{L} of size $J \times K$ with elements given by pixel intensities l_{jk} . This \mathbf{L} has rank 3, and our objective is to decompose this matrix as follows:

$$\mathbf{L} = \mathbf{V}\mathbf{U}^\top \quad (23)$$

where the j^{th} row of matrix $\mathbf{V} \in \mathbb{R}^{J \times 3}$ is the vector \mathbf{v}_j , and the k^{th} row of matrix $\mathbf{U} \in \mathbb{R}^{K \times 3}$ is the vector \mathbf{u}_k . The following procedure ensures that the rows of \mathbf{U} have the following form: $\mathbf{u}_k = [\cos(\phi_k) \quad \sin(\phi_k) \quad 1]$ for some phase ϕ_k . The steps of the algorithm outlined in Algorithm 1 are described as follows:

Step 1. First, since $u_{k3} = 1$, the row space of matrix \mathbf{L} contains the vector $\mathbf{1}$. We reduce the rank of the matrix to 2 by projecting each row into a subspace orthogonal to vector $\mathbf{1}$. Simply put, this amounts to subtracting each row's mean value from its elements.

We then compute the singular value decomposition (SVD) of this rank-2 matrix, and construct a matrix $\tilde{\mathbf{U}} \in \mathbb{R}^{K \times 2}$ composed of the two largest right-singular vectors. Since the columns of $\tilde{\mathbf{U}}$ are a linear combination of the columns of \mathbf{U} , the rows of $\tilde{\mathbf{U}}$ represent points (x_k, y_k) that lie on some 2D ellipse.

Step 2. The ellipse points (x_k, y_k) satisfy the following general quadratic equation of a conic section:

$$\begin{aligned} 0 &= \begin{bmatrix} x_k & y_k & 1 \end{bmatrix} \begin{bmatrix} c_1 & \frac{c_2}{2} & \frac{c_4}{2} \\ \frac{c_2}{2} & c_3 & \frac{c_5}{2} \\ \frac{c_4}{2} & \frac{c_5}{2} & c_6 \end{bmatrix} \begin{bmatrix} x_k \\ y_k \\ 1 \end{bmatrix} \\ &= \begin{bmatrix} x_k^2 & x_k y_k & y_k^2 & x_k & y_k & 1 \end{bmatrix} \mathbf{c}. \end{aligned} \quad (24)$$

We can therefore solve for the conic coefficients \mathbf{c} by solving a homogeneous least squares system using $K \geq 5$ points. This involves computing the smallest right-singular vector of the linear system.

Step 3. The spatial derivative of this polynomial is zero for the 2D point representing the center of the ellipse of

Algorithm 1 Phase Calibration

```

1: function CALIB(L)
  ▷ Step 1: Project rows of matrix  $\mathbf{L} \in \mathbb{R}^{J \times K}$  into sub-
  space orthogonal to vector  $\mathbf{1}$ .
2:    $\mathbf{P} = \mathbf{I} - \frac{1}{K}$ 
3:    $[-, -, \tilde{\mathbf{U}}] = \text{svd}(\mathbf{LP})$ 
4:    $\mathbf{x} = \tilde{\mathbf{U}}_{[:, 1]}, \mathbf{y} = \tilde{\mathbf{U}}_{[:, 2]}$ 
  ▷ Step 2: Fit general conic to points by solving a ho-
  mogeneous least squares system.
5:    $\mathbf{M} = \begin{bmatrix} \mathbf{x}^2 & \mathbf{xy} & \mathbf{y}^2 & \mathbf{x} & \mathbf{y} & 1 \end{bmatrix}$ 
6:    $[-, -, \mathbf{C}] = \text{svd}(\mathbf{M})$ 
7:    $\mathbf{c} = \mathbf{C}_{[:, 6]}$  ▷ Smallest singular vector
  ▷ Step 3: Compute center of ellipse.
8:    $\begin{bmatrix} x_0 \\ y_0 \end{bmatrix} = - \begin{bmatrix} 2c_1 & c_2 \\ c_2 & 2c_3 \end{bmatrix}^{-1} \begin{bmatrix} c_4 \\ c_5 \end{bmatrix}$ 
  ▷ Step 4: Compute quadric model for ellipse.
9:    $f = \begin{bmatrix} x_0^2 & x_0 y_0 & y_0^2 & x_0 & y_0 & 1 \end{bmatrix} \mathbf{c}$ 
10:   $\mathbf{A} = -\frac{1}{f} \begin{bmatrix} c_1 & \frac{c_2}{2} \\ \frac{c_2}{2} & c_3 \end{bmatrix}$ 
  ▷ Step 5: Transform points from ellipse to unit circle.
11:   $\mathbf{U}_{[:, 1:2]} = \begin{bmatrix} \mathbf{x} - x_0 & \mathbf{y} - y_0 \end{bmatrix} \mathbf{A}^{-\frac{1}{2}}$ 
12:   $\mathbf{U}_{[:, 3]} = \mathbf{1}$ 
  ▷ Step 6: Return factorization  $\mathbf{L} = \mathbf{V}\mathbf{U}^\top$ .
13:   $\mathbf{V} = \mathbf{L} \text{pinv}(\mathbf{U}^\top)$ 
14:  Return  $\mathbf{V}, \mathbf{U}$ 
15: end function
```

conic sections:

$$\begin{aligned} \mathbf{0} &= \nabla \left(\begin{bmatrix} x^2 & xy & y^2 & x & y & 1 \end{bmatrix} \mathbf{c} \right) \quad (25) \\ &= \begin{bmatrix} 2c_1 & c_2 \\ c_2 & 2c_3 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} c_4 \\ c_5 \end{bmatrix}. \end{aligned}$$

We can therefore compute the center (x_0, y_0) by solving this linear system.

Step 4. We can rewrite Eq. (24) with points translated by the center as follows:

$$1 = \begin{bmatrix} x_k - x_0 & y_k - x_0 \end{bmatrix} \mathbf{A} \begin{bmatrix} x_k - x_0 \\ y_k - x_0 \end{bmatrix} \quad (26)$$

$$\begin{aligned} \text{where } f &= \begin{bmatrix} x_0^2 & x_0 y_0 & y_0^2 & x_0 & y_0 & 1 \end{bmatrix} \mathbf{c} \\ \mathbf{A} &= -\frac{1}{f} \begin{bmatrix} c_1 & \frac{c_2}{2} \\ \frac{c_2}{2} & c_3 \end{bmatrix}. \end{aligned}$$

Step 5. Transforming points onto the unit circle requires finding any decomposition of the form $\mathbf{A} = \mathbf{B}\mathbf{B}^\top$, e.g.,

Algorithm 2 Horn-Schunck Regularization

```

1: function HS( $\mathbf{v}_j, \mathbf{u}_j, l_j, \alpha, T$ )
2:   Let  $\mathbf{K}$  be a local spatial averaging kernel.
3:   ▷ Iteratively minimize energy functional Eq. (12).
4:   for  $t = 0$  to  $T - 1$  do
5:     ▷ Compute locally smooth phasor.
6:      $\bar{\mathbf{u}}_j^{(t)} = \mathbf{K}(\mathbf{u}^{(t)})_j$ 
7:     ▷ Update phasor with interferometric constraint.
8:      $r_j^{(t)} = \mathbf{v}_j \cdot \bar{\mathbf{u}}_j^{(t)} - l_j$ 
9:      $\mathbf{u}_j^{(t+1)} = \bar{\mathbf{u}}_j^{(t)} - \frac{r_j^{(t)}}{\alpha + \|\mathbf{v}_j\|^2} \mathbf{v}_j$ 
10:   end for
11:   return  $\mathbf{u}_j^{(T)}$ 
12: end function
```

computing the matrix square root. We can then reconstruct the matrix \mathbf{U} as follows:

$$\begin{aligned} \mathbf{U}_{[:, 1:2]} &= \begin{bmatrix} \mathbf{x} - x_0 & \mathbf{y} - y_0 \end{bmatrix} \mathbf{A}^{-\frac{1}{2}} \quad (27) \\ \mathbf{U}_{[:, 3]} &= \mathbf{1}. \end{aligned}$$

This represents the transformation $\mathbf{U} = \tilde{\mathbf{U}}\mathbf{Q}^\top$ discussed in the main paper.

Step 6. We compute the corresponding matrix \mathbf{V} by multiplying the measurement matrix with the pseudoinverse \mathbf{U} .

This algorithm requires $J \geq 2$ because the measurement matrix must be at least of rank 2, and $K \geq 5$ to solve the general conic in step 2.

An important note is that this decomposition is not yet unique. This is because rotating and reflecting the unit circle produces a unit circle. To resolve these final ambiguities, assumptions must be imposed on the phase ϕ_k . In our work, we consider the case where the phase changes globally. We set the initial frame to be the one with zero displacement, which resolves rotational ambiguity. Knowing the direction of the phase change in the second frame resolves the reflection ambiguity (*i.e.*, do points rotate clockwise or counter-clockwise).

It is important to note that there are other factorization-based approaches proposed in the context of phase-shifting interferometry. Specifically, Escobar *et al.* [2] and Estrada *et al.* [3] describe iterative approaches that sequentially update matrices \mathbf{U} and \mathbf{V} until convergence. In contrast, our proposed approach uses a sequence of explicit steps for computing the two factors.

12. Per-pixel Displacement and Motion Recovery

In order to reconstruct spatially-varying phasors, we propose using the classic Horn-Schunck algorithm to solve the

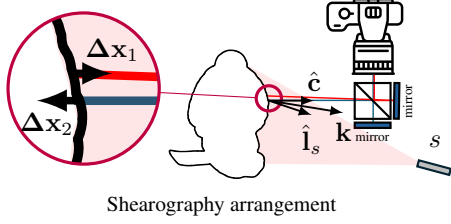


Figure 1. **Arrangement for shearography.** Shearography, also known as speckle pattern shearing interferometry, is supported by our CSPI framework. Shearography works by interfering two different spatial locations (illustrated as the red and blue rays), which is once again sensitive to the difference in their optical path lengths. This therefore makes the method sensitive to surface tilts, as illustrated by the arrows in the inset.

global energy functional (Eq. (12)). Under this model, displacement estimation is an optical-flow problem where the brightness constancy equation becomes the interferometric constraint $\mathbf{v}_j \cdot \mathbf{u}_j = l_j$. A weight α controls the strength of the smoothness regularizer (e.g., larger values of α result in a smoother solution).

Before running the algorithm, we produce an initial estimate of the phasors by solving a simple linear system per pixel; we construct this system using Eq. (6) for a small neighborhood of pixels and assuming the phasor is constant. Given this estimate of the phasors, the Horn-Schunck algorithm refines the solution, following the steps described in Algorithm 2.

In practice, we implement our algorithms such that they only solve for u_{j1} and u_{j2} , since $u_{j3} = 1$. This is achieved by rewriting $\mathbf{v}_j \cdot \mathbf{u}_j = l_j$ as $v_{j1}u_{j1} + v_{j2}u_{j2} = l_j - v_{j3}$.

13. Shearography

Shearography operates similarly to the out-of-plane arrangement, in that it uses a single coherent light source to illuminate the object surface. However, it interferes the reflected field with a laterally sheared copy of itself, achieved by slightly tilting a mirror in a setup reminiscent of a Michelson interferometer [7]; see Fig. 1. That is, the sensor receives two overlapping copies of the scene, with a spatial offset determined through the mirror’s tilt.

Under this configuration, suppose that the light incident on a particular pixel is the result of two fields ψ'_1 and ψ'_2 originating from two different points in a scene. The measured intensity can then be expressed as follows:

$$|\psi'_1 + \psi'_2|^2 = |\psi_1|^2 + |\psi_2|^2 + 2|\psi_1\psi_2^*| \cos \left(\angle\psi_1\psi_2^* - \frac{2\pi}{\lambda} \mathbf{k} \cdot (\Delta\mathbf{x}_1 - \Delta\mathbf{x}_2) \right). \quad (28)$$

Since both points are observed under approximately identical illumination and viewing geometry, the sensitivity vec-

tor for shearography is the same as out-of-plane sensitivity vector $\mathbf{k} = \hat{\mathbf{l}}^s + \hat{\mathbf{c}}$.

Notice in Eq. (28), the phase offset term corresponds to the difference in displacement between the two points rather than the displacement at a single point. For a sufficiently small separation, we can approximate this difference to be displacement gradient along the sensitivity direction.

The gradient sensitivity, similar to the out-of-plane case, is such that one radian of phase difference corresponds to 43.6 nm differential displacement between the sheared points in direction \mathbf{k} . This differential displacement approximates the local tilt (surface slope) in the direction of the spatial offset of the two points.

To validate our implementation, we conducted the experiment on acoustic recovery on this configuration, too. As shown in Fig. 3, the shearography configuration is also able to reconstruct the piano notes, producing a spectrogram that closely matches the ground-truth microphone recording. The recovered signal also preserves higher-order structure, clearly capturing up to fifth-order harmonics that are present in the original audio.

We were also able to recover the Chladni plate pattern using the shearography configuration in Fig. 2. In this case, the recovered phase encodes the local surface tilt on-axis with shear along the image y-axis. On the other hand, the in-plane configuration involved tilting the Chladni plate by approximately 45° to capture a portion of the out-of-plane motion in the in-plane sensitivity direction. Despite the difference in scene geometry and the physical quantity being measured (tilt and displacement), both measurements resemble a consistent nodal structure, revealing information about the vibration mode of the Chladni plate.

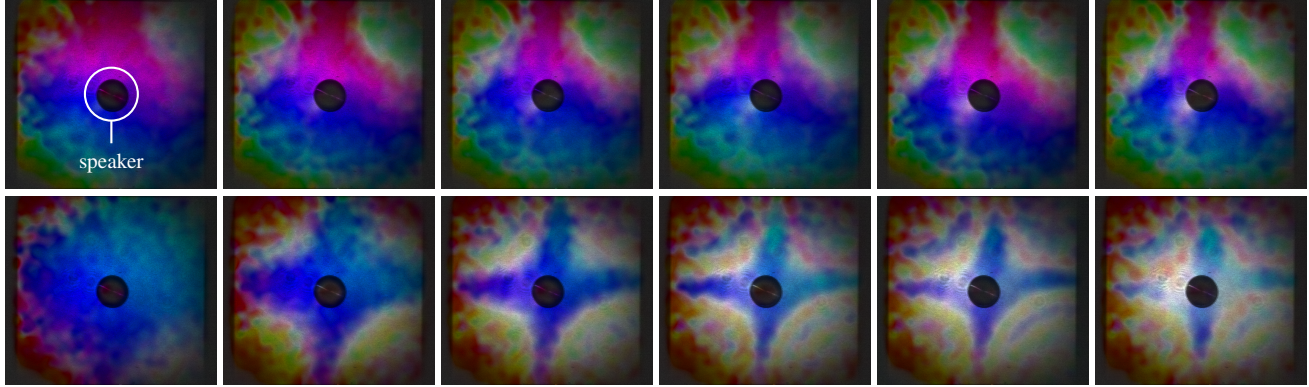


Figure 2. **Chladni patterns using Shearography.** Vibrational modes of a plate from a contact speaker vibrating at 134 Hz (**Row 1**) and 150 Hz (**Row 2**).

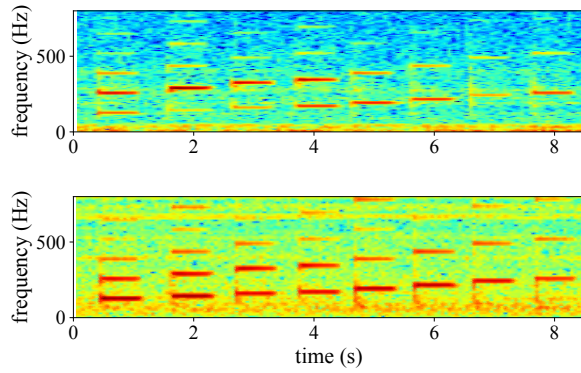


Figure 3. **Spectrogram of a chip bag using shearography.** The bag vibrates in response to a speaker playing piano notes from C3 to C4. (**Top**) The spectrogram recovered using shearography, which is sensitive to surface tilt. (**Bottom**) Microphone Spectrogram.

- [6] Eric W. Weisstein. Hansen-bessel formula. <https://mathworld.wolfram.com/Hansen-BesselFormula.html>, 2025. Accessed: 2025-11-15. ¹²
- [7] Sijin Wu, Xiaoyuan He, and Lianxiang Yang. Enlarging the angle of view in michelson-interferometer-based shearography by embedding a 4f system. *Applied Optics*, 50(21):3789–3794, 2011. ¹⁴

References

- [1] JN Butters and JA Leendertz. A double exposure technique for speckle pattern interferometry. *Journal of Physics E: Scientific Instruments*, 4(4):277, 1971. ¹⁶
- [2] Marco A Escobar, Julio C Estrada, and Javier Vargas. Phase-shifting vu factorization for interferometry. *Optics and Lasers in Engineering*, 124:105797, 2020. ¹³
- [3] Julio C Estrada, Victor H Flores, and Javier Vargas. A two steps phase-shifting demodulation method using the vu factorization. *Optics and Lasers in Engineering*, 147:106730, 2021. ¹³
- [4] Chih-Cheng Kao, Gym-Bin Yeh, Shu-Sheng Lee, Chih-Kung Lee, Ching-Sang Yang, and Kuang-Chong Wu. Phase-shifting algorithms for electronic speckle pattern interferometry. *Applied optics*, 41(1):46–54, 2002. ¹⁶
- [5] Wei-Chung Wang, Chi-Hung Hwang, and Shu-Yu Lin. Vibration measurement by the time-averaged electronic speckle pattern interferometry methods. *Applied optics*, 35(22):4502–4509, 1996. ¹²

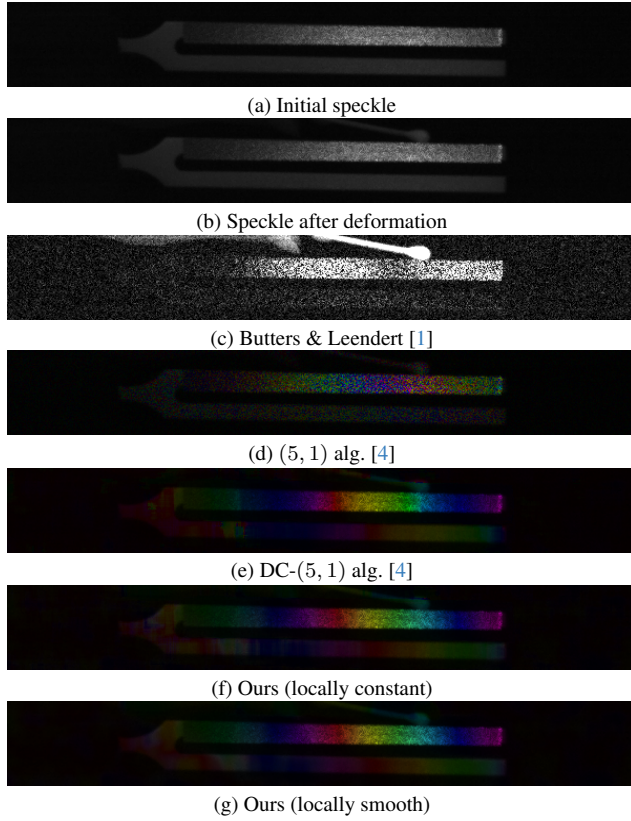


Figure 4. **Single-image methods for estimating displacement**, similar to the simulated results of Fig. 4. (a)-(b) Raw speckle images. (c) Difference between speckle images. Notice the dark fringe that appears underneath the Q-tip. (d)-(e) Estimates calculated according to Kao *et al.* [4]. Their proposed method requires 5 images with known phase, which we synthesized from our factorized representation. Note that this method only produces displacement (phase) and not motion. (f) Our method assuming locally constant phase in a 20×20 region, selected to be the same size window used in (e). (g) Refined phase estimate assuming smooth phasors.