Worcester Polytechnic Institute
Mechanical Engineering Department

Optical Metrology and NDT
ME-593L, B'2014

Introduction: Phase Unwrapping
03 December 2014
Fringe-locus function $\text{mod}[2\pi]$: actual distributions

4-frames:

$$\Omega(u, v) = \tan \left[ \frac{I_4(u, v) - I_2(u, v)}{I_1(u, v) - I_3(u, v)} \right]$$

(a)

8-frames:

$$\begin{align*}
N(u, v) &= (I_1 - I_3)(I_2' - I_4') - (I_2 - I_4)(I_1' - I_3') \\
D(u, v) &= (I_1 - I_3)(I_1' - I_3') + (I_2 - I_4)(I_2' - I_4')
\end{align*}$$

$$\Omega(u, v) = \tan \left[ -\frac{N(u, v)}{D(U, v)} \right]$$

(b)

1-frame, FFT:

$$\Omega(u, v) = \tan \left[ \frac{\text{Im}[I_c(u, v)]}{\text{Re}[I_c(u, v)]} \right]$$

(c)
Recovered optical phase

Fringe-locus function mod[2\pi]

Recovered wrapped phase

\[ N(\theta, v) \]  \[ D(\theta, v) \]  \[ \Omega(\theta, v) \]

Magnitude

\( v \)-direction, pixels
Fringe-locus function mod$[2\pi]$: actual distributions

4-frames:

8-frames:

1-frame, FFT:
Pre-processing: filtering (recommended but not necessary in phase unwrapping)

4-frames:

\[ \Omega(u, v) = \tan\left( \frac{I_4(u, v) - I_2(u, v)}{I_1(u, v) - I_3(u, v)} \right) \]

8-frames:

\[ \Omega(u, v) = \tan\left( -\frac{N(u, v)}{D(U, v)} \right) \]

\[ N(u, v) = (I_1 - I_3)(I_2 - I_4) - (I_2 - I_4)(I'_1 - I'_3) \]
\[ D(u, v) = (I_1 - I_3)(I'_1 - I'_3) + (I_2 - I_4)(I'_2 - I'_4) \]

1-frame, FFT:

\[ \Omega(u, v) = \tan\left( \frac{\text{Im}[I_c(u, v)]}{\text{Re}[I_c(u, v)]} \right) \]

Spatial filtering:
- Separate numerator (sine)
- Separate denominator (cosine)
- Filter numerator and denominator separately:
  - Low-pass filter
  - Median filter
  - Adaptive
  - Fourier
  - Other
- Phase re-evaluation
Pre-processing: filtering

Fourier filtering of sine/cosine images

Convolution in the frequency domain:
\[ Q_D^T(f_u, f_v) = Q_D(f_u, f_v) \cdot W_f(f_u, f_v) \]

Inverse Fourier transformation:
\[ Q'(m, n) = \mathcal{F}^{-1}\{Q_D^T(f_u, f_v)\} \]
Pre-processing: filtering

Fourier filtering of sine/cosine images

Filtered sine image:

Filtered cosine image:

Re-evaluated phase

v-direction, pixels
Phase unwrapping of $\text{phase mod}[2\pi]$

**Wrapped phase:** $\Omega = \Omega(u, v)$

**Unwrapped phase:** $U\{\Omega(u, v)\} = \Omega(u, v) + 2\pi N(u, v)$

$N(u, v) = \text{Round(fringe orders)}$

**Diagram:**

- Wrapped phase
- Unwrapped phase
- $\Omega(0, v)$
- $D(0, v)$
- $N(0, v)$

Magnitude vs. $v$-direction, pixels

$v$-direction, pixels
Row-by-row / Column-by-column algorithm

Evaluate finite difference $\Delta \ell(u,v)$:

$$\Delta \ell(u,v) = \begin{cases} 
\Omega(u,v) - \Omega(u-1,v), & \text{row}, \\
\Omega(u,v) - \Omega(u,v-1), & \text{column}
\end{cases}$$

Line unwrapping:

$$\Omega_c(u,v) = \begin{cases} 
\Omega(u,v) - C_d \cdot 2\pi & \text{if } \Delta \ell > \kappa \cdot 2\pi \\
\Omega(u,v) + C_d \cdot 2\pi & \text{if } \Delta \ell < -\kappa \cdot 2\pi
\end{cases}$$

$$C_d = \text{signed integer}$$

$$\kappa < 1, \text{ unwrapping threshold}$$
Row-by-row / Column-by-column algorithm

\[ \Omega(u_0, v_0) = \Omega_0 \]  (boundary condition)

<p>| | | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Row-Column algorithm

Unfiltered wrapped phase

Unwrapped phase

Phase discontinuities still present
Edge following algorithm

Part I:

Start

- Removal of low modulation pixels
- Spatial filtering?
- Filtering of data files
- Re-evaluation of \( \Omega(u,v) \) modulo \( 2\pi \)
- Choosing of the unwrapping start location \((u_0, v_0)\)

Valid pixel?

- yes
- no

Verify if the pixel is valid

- yes
- no

Phase discontinuity?

- yes
- no

Correct phase discontinuities

- yes
- no

Queue empty?

- yes
- no

Phase restoration

- yes
- no

Part II:

Obtain next pixel location from the FIFO queue

- yes
- no

Evaluate average phase value of unwrapped pixels contained in a 3x3 kernel surrounding the pixel in the queue. Average value is used to correct for phase discontinuities

- yes
- no

Correct phase discontinuities

- yes
- no

Phase discontinuity?

- yes
- no

Exit
Edge following algorithm

Part I

Star

Removal of low modulation pixels: invalid pixels

Spatial filtering?

yes

Filtering of data files

Re-evaluation of $\Omega(u,v)$ modulo $2\pi$

Choosing of the unwrapping start location $(u_0, v_0)$

no

Valid pixel?

Valid pixel?

no

Identify a new starting point among a 3x3 kernel around the initial point location

yes

yes

no

no

Valid pixel?

Terminate?

yes

Exit

yes

no

Add the $(u, v)$ locations of the 4-connected, non-processed, neighborhood pixels surrounding the processed pixel, to a FIFO queue.
Edge following algorithm

Part II

Correct phase discontinuities

Phase discontinuity?

yes

no

Obtain next pixel location from the FIFO queue

Queue empty?

yes

Phase restoration

no

Valid pixel?

yes

Evaluate average phase value of unwrapped pixels contained in a 3x3 kernel surrounding the pixel in the queue. Average value is used to correct for phase discontinuities

Add the \((u, v)\) locations of the 4-connected, non-processed, neighborhood pixels surrounding the processed pixel, to a FIFO queue

Correct phase discontinuities

yes

no

Valid pixel?

yes

no
Edge following algorithm: FIFO queue

(a) 4 connected pixels to processed pixel are added to the queue, (b) the next pixel to be processed is taken from the queue and after processing, the unprocessed pixels contained in the 4 connected pixels surrounding the processed pixel are added to the queue.
Example: interferogram with open-fringes

Wrapped phase → Unwrapped phase → Interpreted shape

Interpretation

Need:
Camera calibration & Sensitivity Vector
Example: interferogram with closed-fringes

Wrapped phase → Unwrapped phase

Interpretation

Need:
Camera calibration & Sensitivity Vector
Recommended Reading Assignment

Papers by
