

# Application of Phase Unwrapping to Digital Holographic Microscopy

---

**Sara Taylor**

Email : [sltaylor@calpoly.edu](mailto:sltaylor@calpoly.edu)

Dept. of Physics, California Polytechnic State University, San Luis Obispo

## **Abstract**

In this project, I demonstrate the use of phase unwrapping to recover the phase information in digital holographic microscopy. I show that the optical thickness of an object can be recovered but that the unwrapping is sensitive to noise.

## Introduction

When light passes through a transparent or semi-transparent object it undergoes a phase shift proportional to the thickness and refractive index of the object. Although this shift is generally not observable using a conventional microscope, such phase shifts can become visible using holographic microscopy.

### 1. Theory

#### 1.1 Microscopy

The setup for digital microscopy is almost the same as for transmission holography, only differing with the lens placed before the CCD, which refocuses the beam after being expanded. The setup is shown in Figure 1.1.1. By using a microscope objective lens to magnify the object, we were able to apply the same techniques to a buccal epithelial cell.

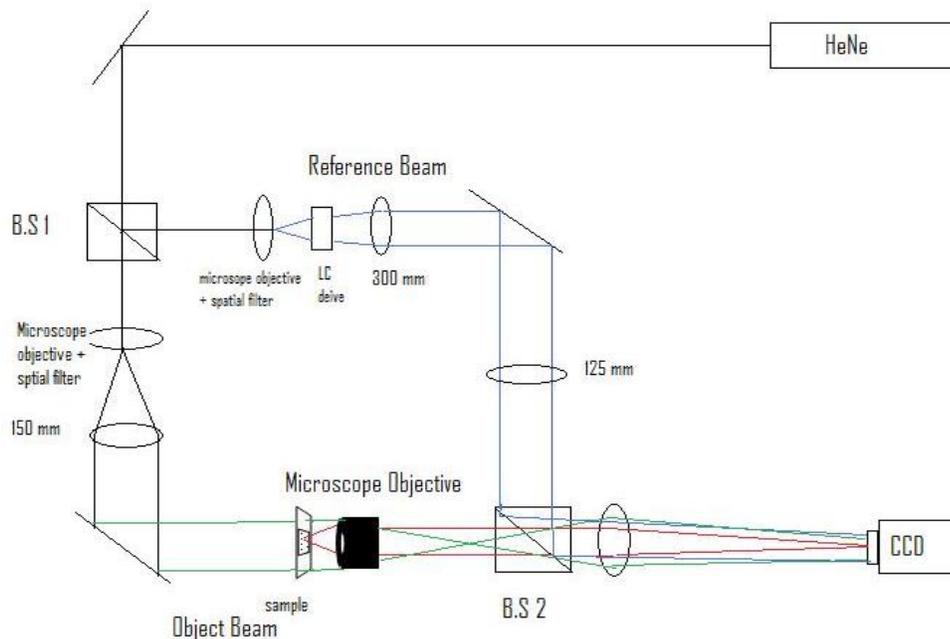


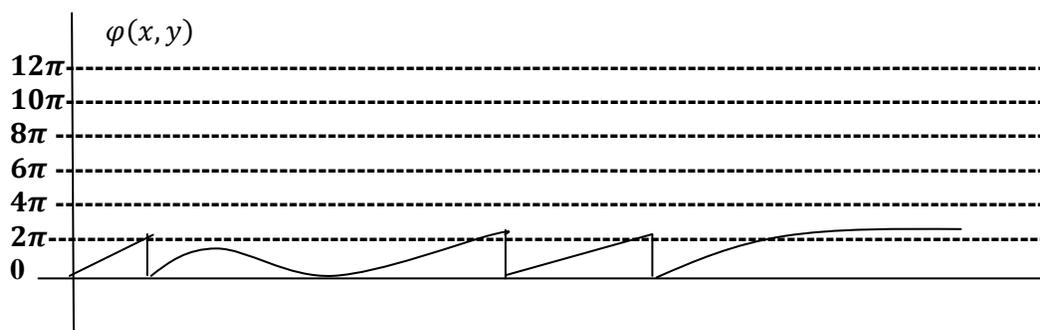
Figure 1.1.1 The slightly off axis microscopic interferometry setup. Drawing not to scale

#### 1.2 Principles of Phase Unwrapping

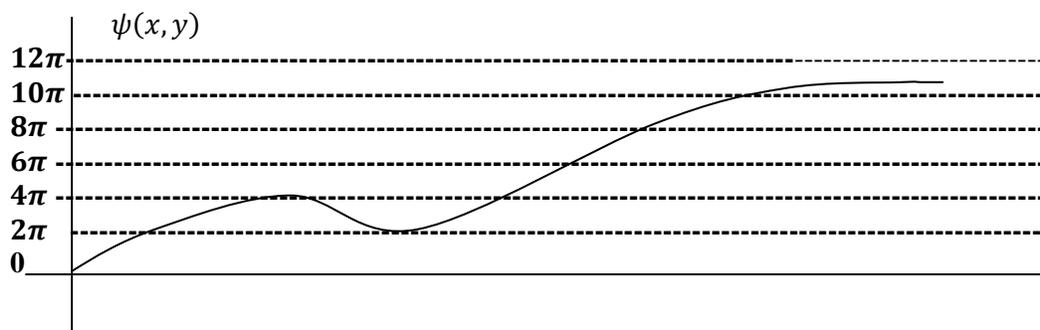
Coherent light is characterized by both amplitude and phase. Phase is a property related to the temporal or spatial wavelength and influences the signal only through those values that lie between  $\pm\pi$  radians. These values are referred to as the wrapped phase values because the absolute phase is wrapping into the interval  $(-\pi, \pi)$ . The wrapped phase,  $\varphi(x, y)$ , is then unwrapped to the true phase,  $\psi(x, y)$ , by the following equation

$$\psi(x, y) = \varphi(x, y) + 2\pi k(x, y) \quad (1.1)$$

Where  $k(x, y)$  is an integer function that forces  $-\pi < \varphi < \pi$ . Therefore,  $\psi(x, y)$  is a nonlinear function of  $\varphi(x, y)$ . In general,  $\psi(x, y)$  relates to a physical quantity such as a surface topography interferometry. The nonlinear  $\psi(x, y)$  that can be extracted from a signal does provide useful information but must be unwrapped for further use [2]. The relationship between the wrapped and unwrapped phase can be seen in Figure 1.2.1.



(a) Wrapped



(b) Unwrapped

Figure 1.2.1: Graph (a) demonstrates the wrapped function,  $\varphi(x, y)$ , and (b) demonstrates the unwrapped function,  $\psi(x, y)$ .

### Interference

When waves interfere with one another and are in phase with one another they add constructively and while out of phase add destructively. A single wave exhibits no interference properties unless split into two or more waves and recombined. This technique is used in various applications such as digital holography or microscopy.

### Principles of Digital Holography

Digital holography is based on the basic principles of classic holography only differing by using a digital device, such as a CCD array, to record the hologram, as opposed to holographic film. A typical setup consists of a beam of monochromatic light passed through an optical system which splits the beams in two.

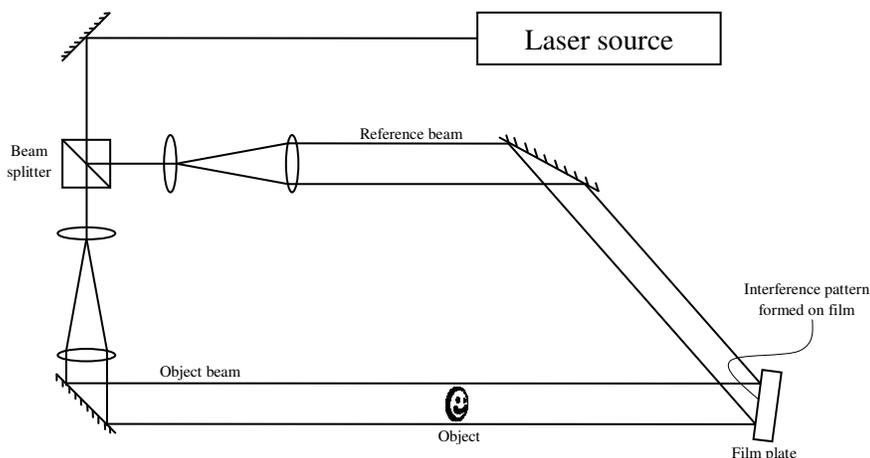


Figure 1.2.2: Schematic design of a typical setup for transmission holography

The two beams are referred to as the reference beam and the object beam. The object beam illuminates the object and the reference beams passes through unobstructed. The reference beam is then superimposed on the object beam. The recombining of the two beams results in an interference pattern, which is then recorded by the digital device, the CCD. A computer can analyze the hologram later. With no object present, a simple realization of an interference pattern can be seen in Figure 1.2.3

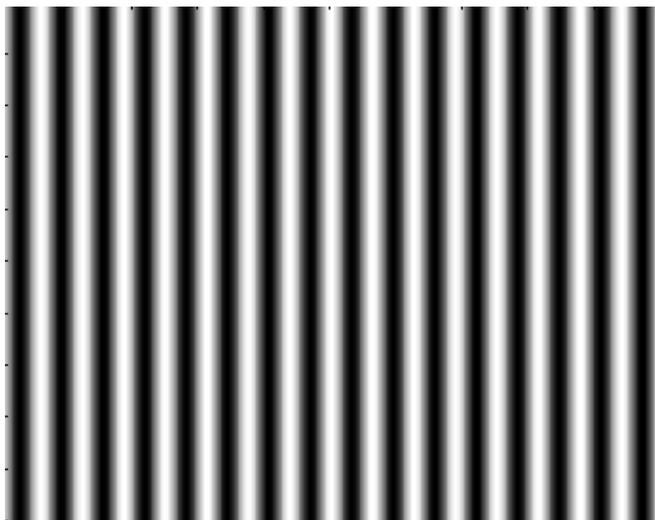
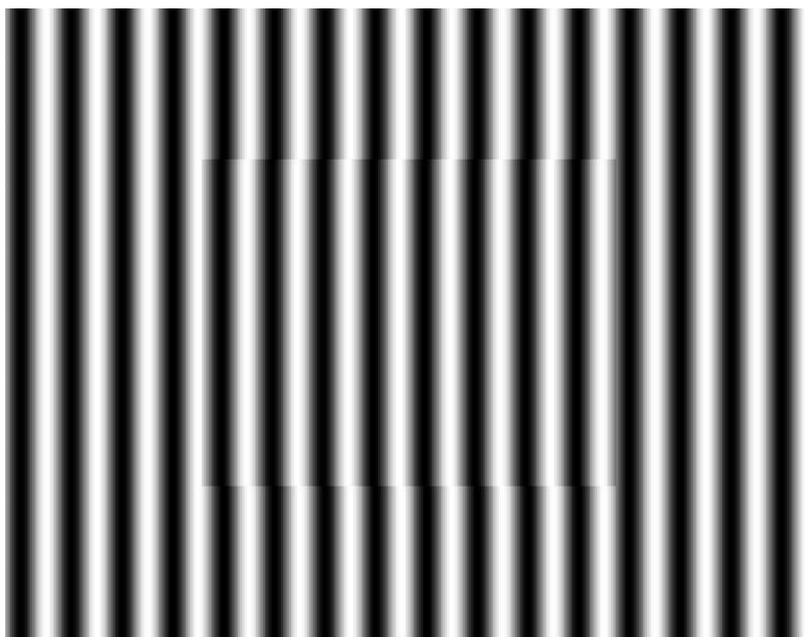


Figure 1.2.3 Simulated interference pattern for an unobstructed object beam

With no object present in the object beam, the interference pattern results in this case in straight vertical lines.

When an object is introduced in the path of the object beam the phase of the light shifts depending on the thickness of the object and its relative index of refraction. One important aspect of digital holography is the information carried in the interference pattern created when light was passed through the optical setup. The interference pattern contains information on the phase due to the object present, which can be unwrapped using a computer. When run through a computer program, the unwrapped phase creates an image of the object showing the phase “height” of the object. For example, if we image this rectangle of

phase “height” .4 in the object beam, the interference pattern would look something like that in figure 1.2.4.



*Figure 1.2.4 Simulated interference pattern with a square object in the in the object beam*

The code for the simulated image can be found in Appendix A.

### *1.3 Phase shifting Interferometry*

Phase shifting interferometry addresses several issues associated with the classical analysis of static interferograms. Classically, the center of a fringe is measured. The ability to determine the center of fringes creates errors, which limits the precision of classical analysis. In addition, any intensity variations across the interferogram will induce error. Phase shifting interferometry overcomes these errors by not relying on finding the fringe centers but rather by collecting intensity data from every point of the CCD. Phase shifting interferometry analyzes multiple interferograms with the reference optical surface shifted by a precise fraction of a wavelength between each exposure.

In order to create a phase shift in the interference pattern, a liquid crystal variable retarder was used. A liquid crystal variable retarder works by applying voltage to the reference beam causing a change in phase relative to the object beam. In order to determine the voltage needed to shift the phase by  $\frac{1}{2}\pi$ ,  $\frac{3}{4}\pi$ , and  $\frac{5}{4}\pi$ , a series of point were taken. A plot of the phase shift vs. voltage can be seen in Figure 1.3.1. This was obtained by putting a liquid crystal device in an interferometer and measuring the phase shift. This gave us a relationship between phase shift and voltage.

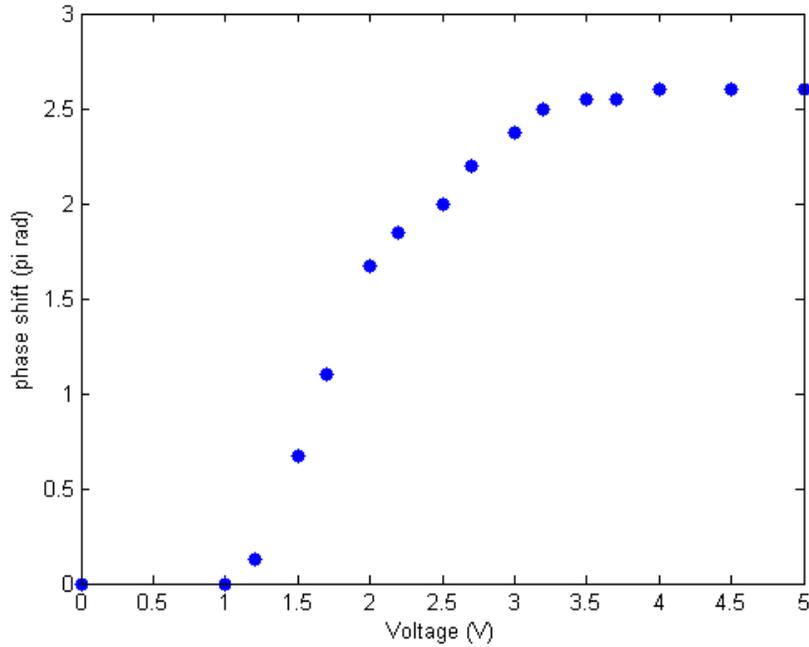


Figure 1.3.1 Relationship between phase shift and voltage using the liquid crystal device at  $\lambda=633\text{nm}$

The phase at each pixel can be determined by a pixel-by-pixel analysis using the intensity as a function of the phase. Since a minimum of three sets of recorded fringe patterns is needed to construct a wave front, the phase can be calculated from a phase shift of  $\frac{\pi}{2}$  per exposure. According to Creath, the intensity measurements are expressed as

$$\psi_i(x, y) = (1 + \cos(\phi(x, y) + \alpha_i)) \quad (1.2)$$

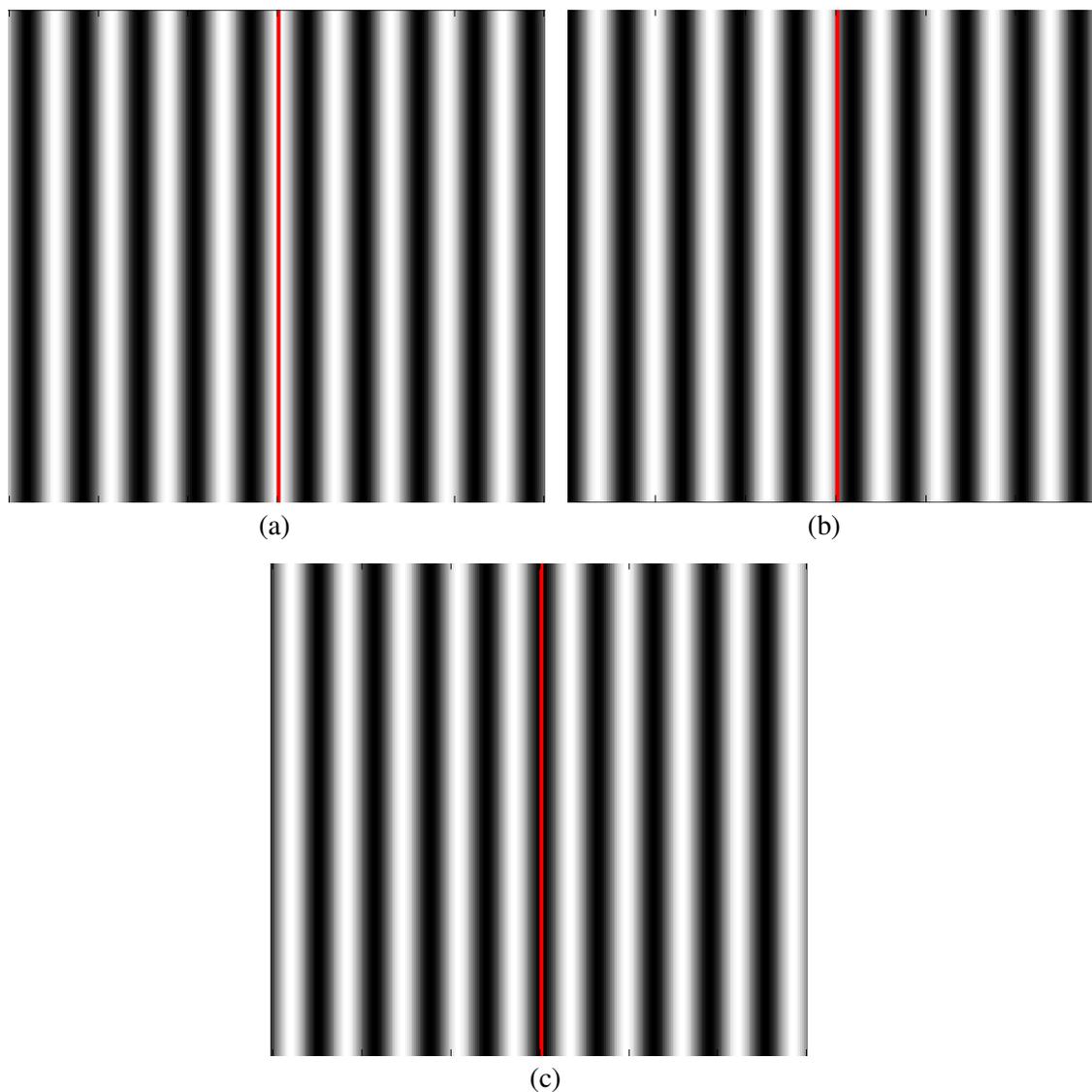
Where  $\alpha_i$  is the phase shift of each exposure and equivalent to  $\frac{1}{4}\pi$ ,  $\frac{3}{4}\pi$ , and  $\frac{5}{4}\pi$ . Plugging in the values for  $\alpha_i$ , the three intensity measurements may be expressed as

$$\psi_1(x, y) = (1 + \cos(\phi(x, y) + \frac{\pi}{4})) \quad (1.3)$$

$$\psi_2(x, y) = (1 + \cos(\phi(x, y) + \frac{3\pi}{4})) \quad (1.4)$$

$$\psi_3(x, y) = (1 + \cos(\phi(x, y) + \frac{5\pi}{4})) \quad (1.5)$$

The images of the intensity equations can be seen in figure 1.3.2.



*Figure 1.3.2 Each of the 3 phase shifts used to determine the phase at each point. From (a) to (c) the images have been shifted by  $\pi/2$ . The red line follows the shift of the interference pattern.*

From the images, we can see that each image shifts by  $\frac{\pi}{2}$  as indicated by equation of the intensity. From the presence of the red line, as we go from image (a) to (c), the interference pattern makes a full shift of  $\pi$ .

As discussed in Creath's analysis of phase shifting interferometry, when the phase is shifted over a  $\frac{\pi}{2}$  phase shift per frame, the phase at each point is then given by

$$\phi(x, y) = \tan^{-1} \frac{I_3(x, y) - I_2(x, y)}{I_1(x, y) - I_2(x, y)} \quad (1.6)$$

### 1.4 Simulation Results

The technique to determine the phase at each point used in the simulated image is the three-bucket technique of equation 1.5 as discussed by Creath [1]. We simulated a thin rectangular phase object which shifted the phase by .4 radians. The simulation is performed using a known phase function found in equation 1.3 through 1.5. We can obtain the wrapped phase  $\phi(x, y)$  from equation 1.6. Figure 1.4.1 plots the true, unwrapped phase, and wrapped phase of  $\phi(x, y)$  for one row in the image.

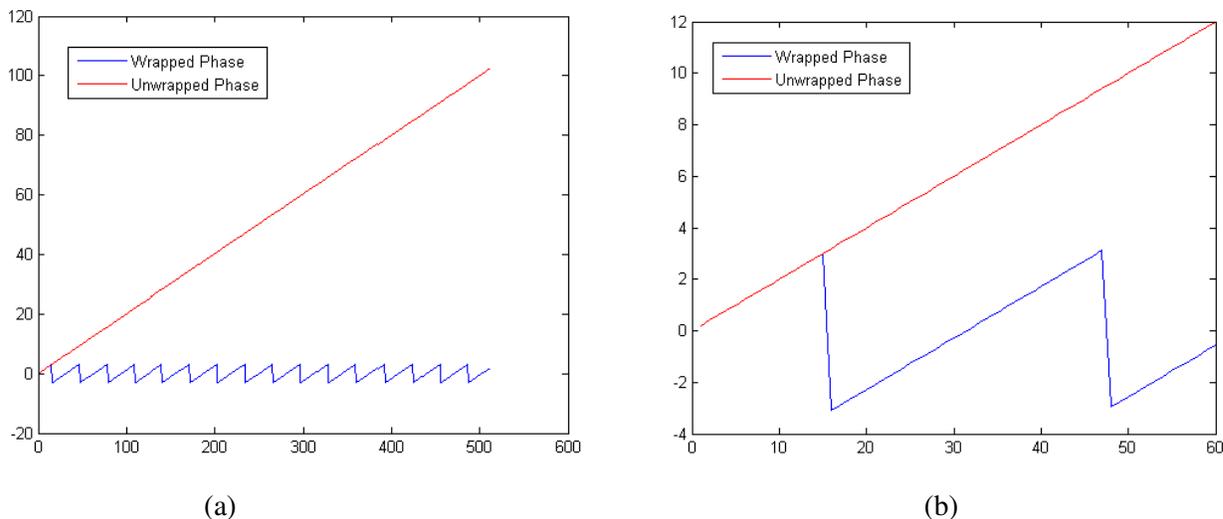
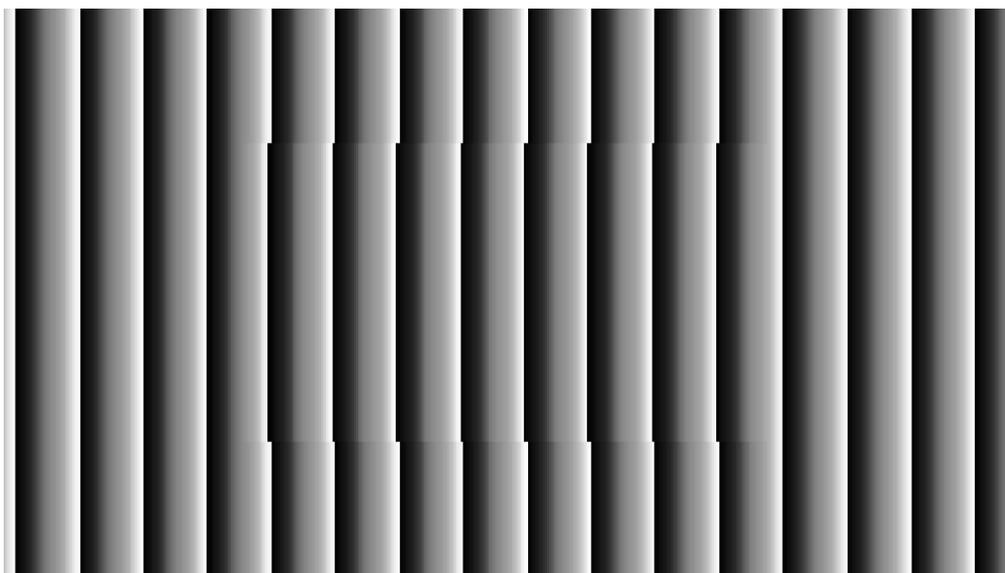


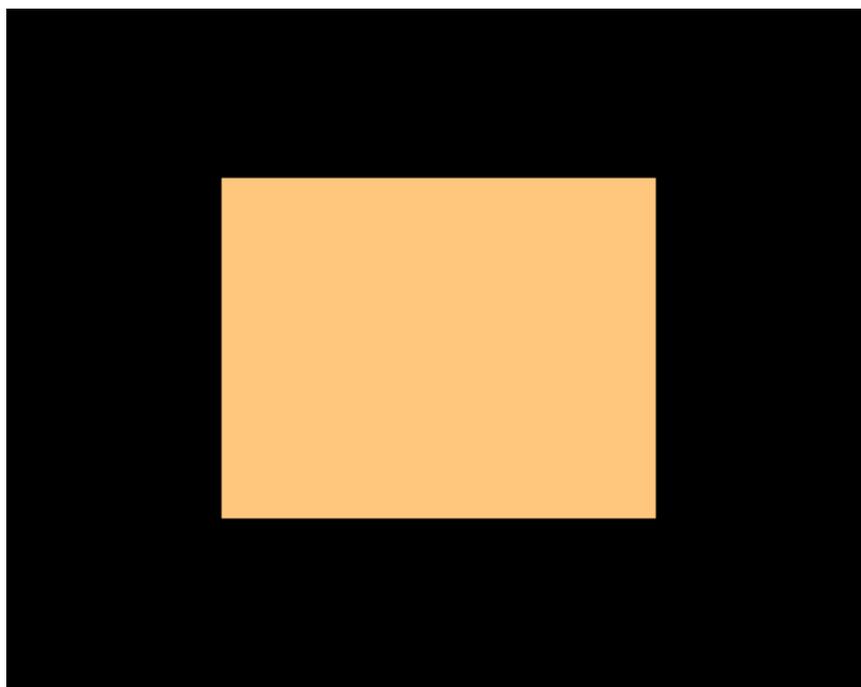
Figure 1.4.1 One row of the image has been plotted showing the wrapped and true phase. The right image shows the  $2\pi$  additions that are necessary in phase unwrapping.

It is seen that while the unwrapped phase is a linear function, the wrapped phase  $\phi(x, y)$  is plotted as a saw-toothed pattern in blue. Phase unwrapping is accomplished by detecting the wrapped phase jumps and adding an appropriate multiple of  $2\pi$  radians. Visually, from graph (b), it is easily determined that nothing be added to  $\phi(x, y)$  for  $0 \leq x \leq 15$  to give a correct estimate of the unwrapped phase within that same interval. Also, for the interval  $15 \leq x \leq 47$ , one multiple of  $2\pi$  must be added and  $2 \times 2\pi$  must be added in the interval  $47 \leq x \leq 78$ . We could determine the correct unwrapping from the graph by visually detecting the  $2\pi$  jumps in the wrapped phase and adding the appropriate  $2\pi$  multiple. In addition, we created a grayscale image of the phase at each point of the whole image shown in figure 1.4.2



*Figure 1.4.2 Resulting image of the phase map after equation 1.6 has been applied. Image contains the information necessary to complete phase unwrapping*

The phase information at each pixel can then be unwrapped, as demonstrated in Figure 1.4.1 , to create an image of the original object. After calculating the phase at each point, we then used the MATLAB program in appendix B to unwrap the phase. Figure 1.4.3 shows the final image created by unwrapping the whole image row by row at each pixel.



*Figure 1.4.3 The resulting image of the simulated square after the MATLAB phase unwrapping code has been applied.*

As we can see from the image, by unwrapping the phase at each pixel, we are able to create an image of the original object which in this case was a then rectangular phase object. In addition, not only can we create a two dimensional image of the object, the phase information also allows us to create a 3-D image,

revealing the optical thickness and size of the object shown in Figure 1.4.4. The simulated code for the square can be found in Appendix B.

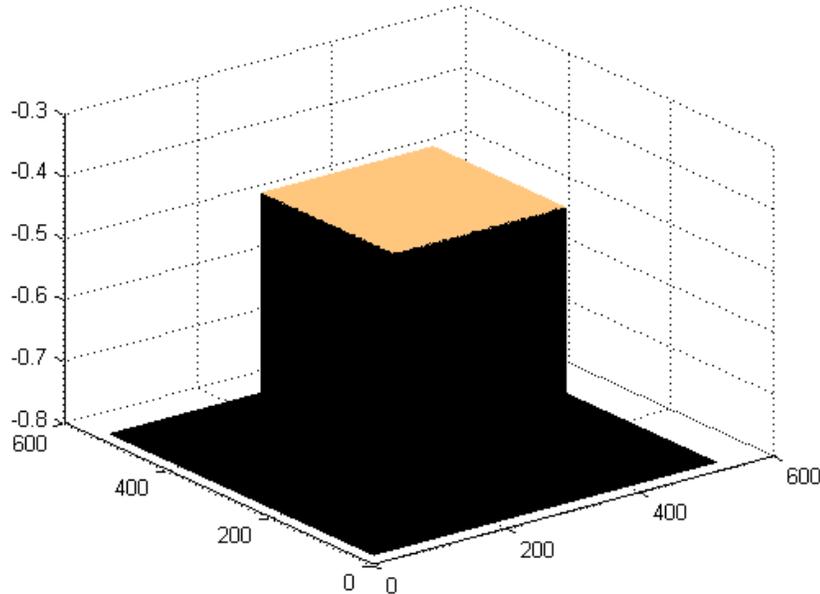


Figure 1.4.4: The resulting 3 dimensional image of the simulated square

## 2. Experiment

In order to determine the overall effective magnification, we used a USAF resolution chart and the setup in Figure 1.1, and placed it in the path of the object beam. By choosing element 1 in group 5 we were able to take a series of data points finding that the object size was equivalent to 526.8 pixels. As can be seen in Appendix D, the CCD pixel size is  $3.45\mu\text{m}$ . Therefore, the magnification was determined by

$$526.8 \text{ pixels} \times (3.45\mu\text{m}) = 1817.5\mu\text{m}$$

$$M = \frac{5(1817.5\mu\text{m})}{64\mu\text{m}} = 23.3$$

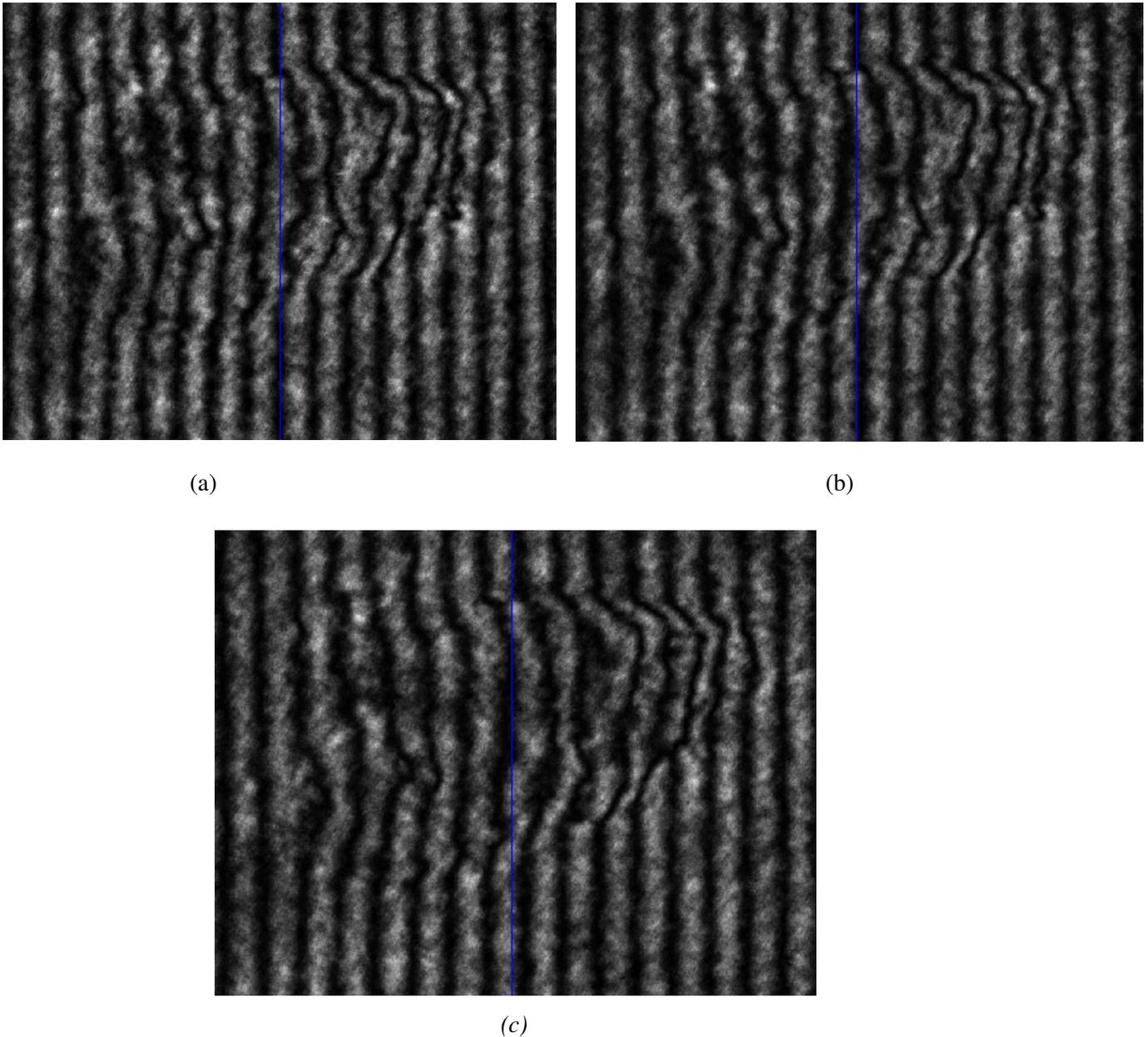
The overall magnification was determined to be 23.3. A sample was prepared using a human buccal cell. Using an image of the buccal cell and determining the number of pixels which the cell spans, the size was calculated to be  $99.2\mu\text{m}$ .

$$\frac{670 \text{ pixels} \times 3.45\mu\text{m}}{23.3} = 99.2\mu\text{m}$$

A human buccal epithelial cell varies in size from  $50\mu\text{m}$  to  $100\mu\text{m}$ , making our results in agreement with what we would expect.

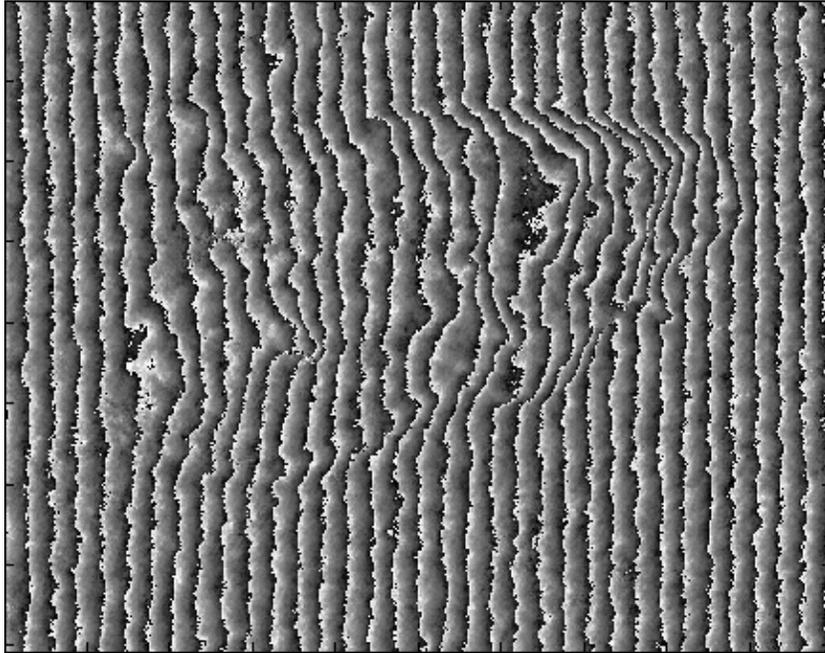
One application of the previous methods for phase unwrapping involves imaging biological cells. The first step we took to determine the phase at each pixel, was to introduce the reference beam to a liquid

crystal modulator as described before. We also shifted the beams by  $\frac{\pi}{2}$  the resulting images are shown in figure 2.1.1.



*Figure 2.1.1 Buccal epithelial cell with interference pattern overlay. From the blue line placed at the center of the image, we can see that phase has been shifted by  $\frac{\pi}{2}$  using the liquid crystal retarder for each consecutive image.*

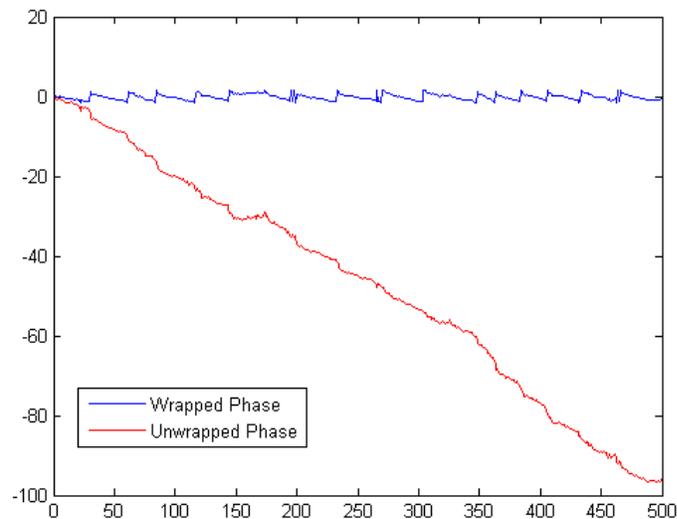
As we can see from the blue line, which is placed in the center of the image, the fringe pattern shifts for each consecutive image. This shift, allows for the phase to be extracted using equation 1.6. By applying equation 1.6, the phase at each pixel can be extracted, the result is shown in figure 2.1.2. The MATLAB code used to determine the phase at each point and generate the image can be found in Appendix C.



*Figure 2.1.2 Phase map of the buccal epithelial cell after the MATLAB code has been applied.*

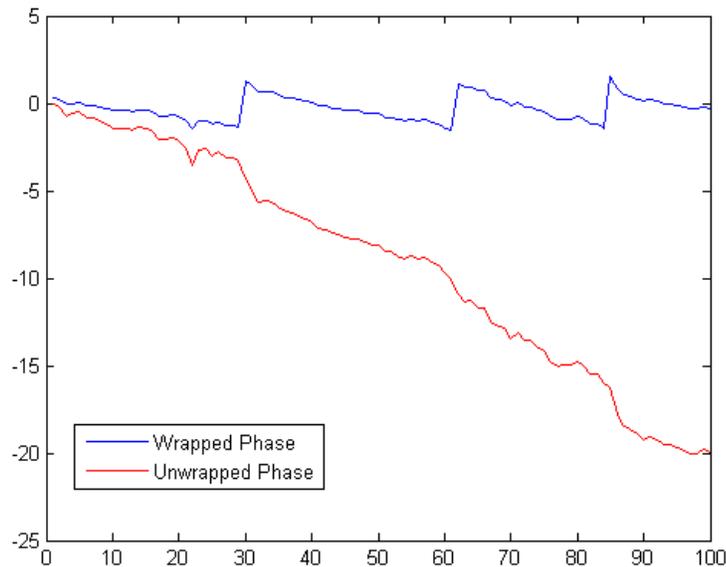
As we would expect, the phase map of the cell resembles that of the simulated square. It is known that the shift in phase is directly proportional to the optical thickness of an object. The regions where the phase is retarded the most will result in the highest peaks of the cell. From the phase map, we can clearly see where the buccal cell is thickest as well as the outline of the surface of the cell.

Like the simulated square, in order to unwrap the phase, the phase jumps need to be detected and the appropriate multiple of  $2\pi$  need to be added; this process results in the true phase. We can see from figure 2.1.3 the unwrapped phase also follows a linear trend, however, unlike the square it results in a negative slope.



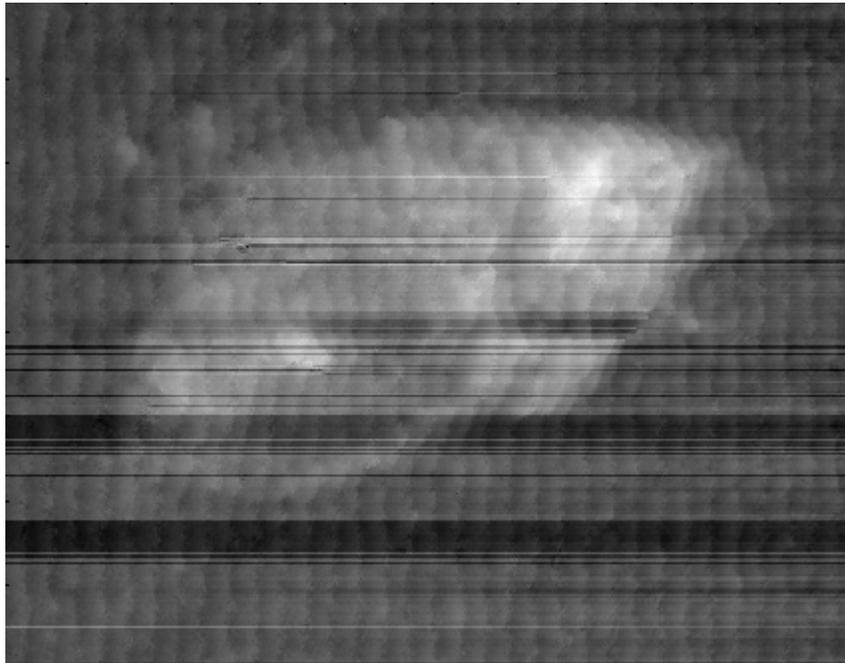
*Figure 2.1.3 Plotted wrapped and true phase of one row from the image of the buccal cell*

Also, in comparing the simulated wrapped and unwrapped phase plot, we can see that the unwrapping of the cell results in line which are not “smooth”. A close look at the plot, shows that each line appears to contain imperfections. These “imperfections” may be a result of noise or movement during the experiment. The optical equipment was extremely sensitive and any surrounding movements caused an unsteady beam of light.



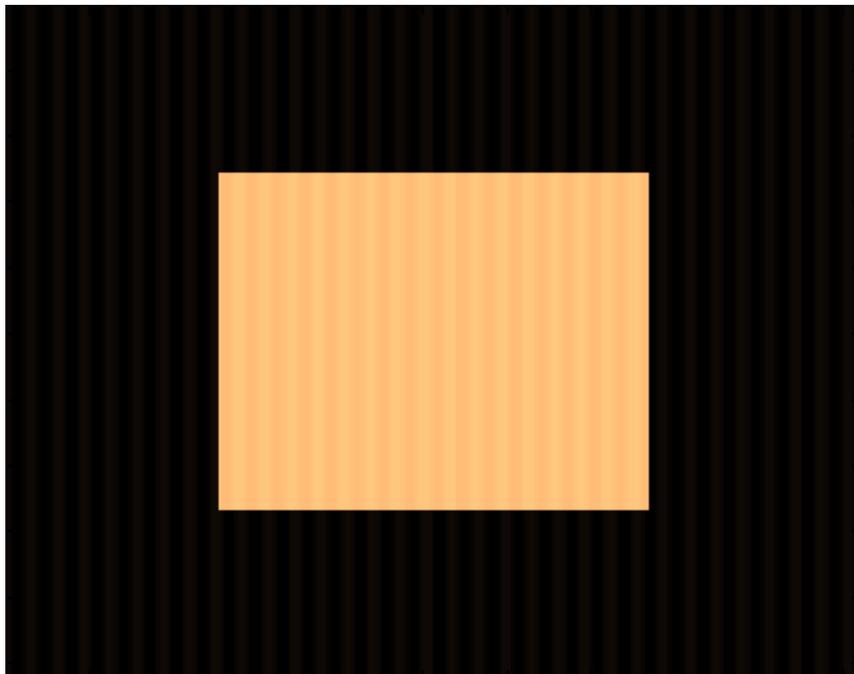
*Figure 2.1.4 A closer look at the plot of the wrapped and unwrapped phase for the buccal epithelial cell. We can see the linear trend of the unwrapped phase*

The imperfections which are evident when looking closely at the plotted wrapped and unwrapped phases of the image, also seemed to effect the overall quality of the final unwrapped image shown in Figure 2.1.4. The entire image was unwrapped using the built in MATLAB function ‘unwrap’. Like the code we wrote for the simulated image, the function operates by correcting the phase angles in a vector by adding multiples of  $\pm 2\pi$  when absolute jumps between consecutive elements are greater than or equal to  $\pi$  radians. In addition to using the unwrap function, we subtracted out the slope of the unwrapped phase line in order to create the image shown in figure 2.1.5.



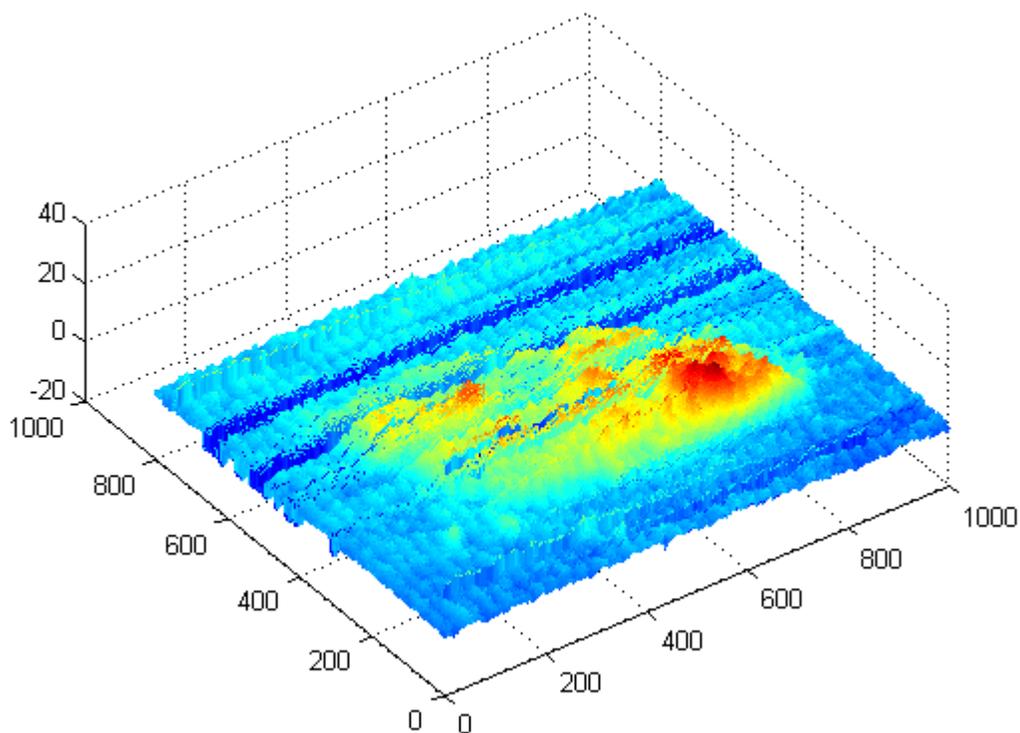
*Figure 2.1.5 Resulting image of the buccal epithelial cell after run through the phase unwrapping MATLAB code with evident striations.*

It is not know why the scalloping effect occurs throughout the picture occurred but is believed to have arisen from an inaccurate shift of  $\frac{\pi}{2}$  in the intensity equations. We can demonstrate this possibility using the simulated square image again. When the original equations (equations 1.3 to 1.5) have been shifted by  $\alpha_2 = \frac{3\pi}{4} + .03$  and  $\alpha_3 = \frac{5\pi}{4} + .01$  the image results in scalloping like that seen in the buccal cell image.



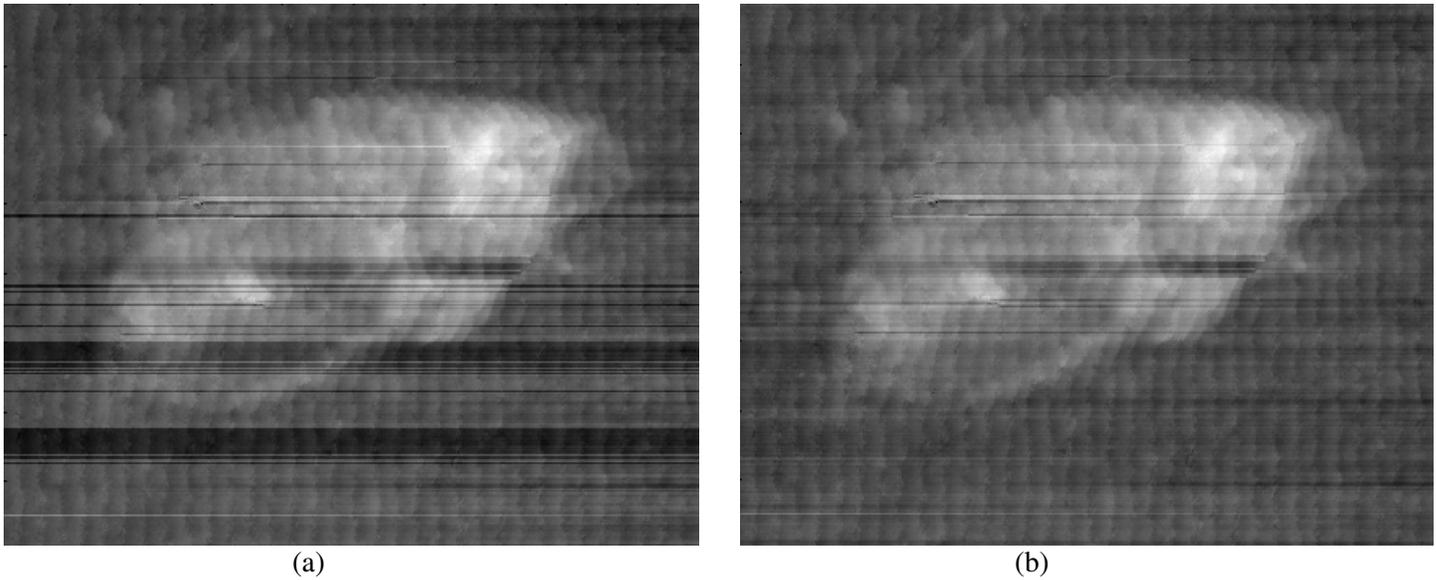
*Figure 2.1.6 When the phase shift is not equivalent to  $\frac{\pi}{2}$  a scalloping effect occurs throughout the image.*

We were also able to create a 3- dimensional image of the cell, shown in figure 2.1.7.



*Figure 2.1.7 3- Dimensional view the buccal epithelial cell*

From the 3-D image we can see that areas with the greatest phase retardance are in fact, the areas of greatest optical thickness. The scalloping, is also evident in the image but could be resolved through more accurate phase shifts. In addition the image appears to be “missing” areas, these areas are seen in figure 2.1.5 as the streaks or darker areas, present across each row. Although it is unknown why these have resulted, a close examination shows that the values appear to be lower in value. In order to correct for this, each row was shifted upward so that every initial point started at a value of zero. Although it did not completely correct the streaking, it did diminish the amount of streaking that was present. Figure 2.1.8 shows the original image with the correction made to each row.



*Figure 2.1.8 From the image on the right we can see by improving the MATLAB code, the streaks through the image have been improved. The image on the left is the original result.*

In comparing the two images side by side, we see that image (b) displays more detail of the cell and results in clearer image.

### **3. Conclusion**

We were able to use the setup for microscopic interferometry to record interference patterns of a buccal cell. By using the three-bucket technique as discussed in the work by Katherine Creath [1], we were able to extract the phase information. By extracting the phase information we were able to write a MATLAB code and unwrap the true phase. This allowed us to create a 2-Dimensional and 3-Dimensional image of the original object.

### **4. Possible Further Research**

#### *Effects of noise on phase unwrapping*

One difficulty we encountered when unwrapping the phase the imperfections which arose in the final image. By investigating the cause of the scalloping effect seen in the image as well as the streaks, we could smooth out any flaws. There are several difficulties that can arise when performing phase unwrapping, one of which the effect of noise. The combination of noise and marginal sampling work together could alone cause an unsuccessful attempt at phase unwrapping. One possible solution to the flaws could be investigating the effects of noise on phase unwrapping. We could illustrate this effect by generating a complex phasor of the same phase as the sinusoidal phase used in the simulated square. In addition, we could add Gaussian noise to produce a complex signal. By unwrapping a function of this form could reveal the impact has the overall result of phase unwrapping.

**Appendix A- simulated.m**

---

```
%This program creates the interference pattern of a thin rectangular object

arr=zeros(512,512); %holds the initial image
arr2=zeros(512,512); %holds the center part

ps1_arr=zeros(512,512); %first phase shift
ps2_arr=zeros(512,512); %2nd phase shift

for i=128:384;
for j=128:384;
    arr2(i,j)=.4;
end
end

imagesc(arr2)

for i=1:512;
    for j=1:512;
        arr(i,j)=1+cos(j/5+arr2(i,j)+pi/4);
    end
end

for i=1:512;
    for j=1:512;
        ps1_arr(i,j)=1+cos(j/5+arr2(i,j)+3*pi/4);
    end
end

for i=1:512;
    for j=1:512;
        ps2_arr(i,j)=1+cos(j/5+arr2(i,j)+5*pi/4);
    end
end

imagesc(arr)
```

## Appendix B- simulated. m

---

%This program computes the phase at each point of the thin rectangular object using the three bucket method, then unwraps each row on the image

```

top=ps2_arr-ps1_arr;
bottom=arr-ps1_arr;

phasesmap=atan2(top,bottom);
imagesc(phasesmap)

imagesc(phasesmap)
newmap=zeros(512,512);
for line=1:512
%process a single row
r=phasesmap(line, :);
p=zeros(1,512);
N=0
for i=1:511
    p(i)=r(i)+N*2*pi;
    if abs(r(i+1)-r(i))>pi
        N=N+1;
    end
end
end

sloped=(1:512)/5;
sloped(512)=0;
newmap(line,:)=p-sloped;
end
imagesc(newmap(1:512,1:511))
colormap(copper)
mesh(newmap(1:512,1:511))
colormap(copper)

```

## Appendix C- PhaseUnwrapping.m

---

% This program reads each cell image into the computer and computes the phase at each point using the three bucket technique. It then uses the unwrap function to unwrap each row of the image

```
% Finding Phase of Cheek cell
close all
clear all
phase1=imread('cheek1','png');
imagesc(phase1(906:1712, 528:1525))
colormap(gray)
line([499 499], [0 800])
phase1=double(phase1);
phase2=imread('cheek2','png');
phase2=double(phase2);
imagesc(phase2(906:1712, 528:1525))
line([499 499], [0 800])
colormap(gray)
phase3=imread('cheek3','png');
phase3=double(phase3);
imagesc(phase3(906:1712, 528:1525))
line([499 499], [0 800])
colormap(gray)

%compute the phase at each point
top=phase3-phase2;
bottom=phase1-phase2;
arctancheek=atan(.5*top./bottom);
imagesc(arctancheek(906:1712, 528:1525))
colormap(gray)
unwrap_image=zeros(807,998);
b=arctancheek(906:1712, 528:1525);
imagesc(b)
colormap(gray)
%%
```

## Appendix C continued- PhaseUnwrapping.m

---

```

% Unwrap the phase
for i=1:807;
    tilt=1:998;
    c=b(i,:);
c=c+pi/2;
c=2*c-pi;
cu=unwrap(c);
if cu(1,1) < 0
    cu(1,:)= cu(:,:)- cu(1,1)
end
if cu(1,1) >0
    cu(1,:)=cu(:,:)- cu(1,1)
end
endpoint=cu(998);
x=cu(1);
slope=(endpoint-x)/998;
cu=cu-slope*tilt;
figure(2)
plot((cu)); drawnow
uwrap_image(i,:)=(cu);
en

figure(5)
colormap(gray)
imagesc(uwrap_image)

figure(6)
mesh(uwrap_image)

plot(b(400,1:100))
hold on
plot(uwrap_image(400,1:100), 'r')
legend('Wrapped Phase', 'Unwrapped Phase')

```

## Appendix D- Hologram Capture Equipment Details

---

- **Capture Device**

<b>Basler piA2400-17gm</b> <a href="http://www.graftek.com/pdf/Brochures/basler/pilot_1.pdf">http://www.graftek.com/pdf/Brochures/basler/pilot_1.pdf</a>	
Sensor Size (H x V pixels)	2448 x 2050
Pixel Size ( $\mu\text{m}$ )	3.45 x 3.45
Max. Frame Rate (at full resolution)	17 fps
Data Output Type (Interface)	Gigabit Ethernet (GigE Vision compliant)

- **Capture Software**

- National Instruments Measurement & Automation Explorer

- **Central Processing Unit**

- Intel® Pentium® 4 CPU 3.80GHz (2 CPUs)

**References**

- [1] Creath, Katherine “Phase-Measurement Interferometry Techniques’ Progress in Optics XXVI pp 351- 389 (1988)
- [2] Ghiglia, Dennis and Pritt, Mark “ Two- Dimensional Phase Unwrapping- Theory, Algorithms, and Software”, John Wiley and Sons (1998)