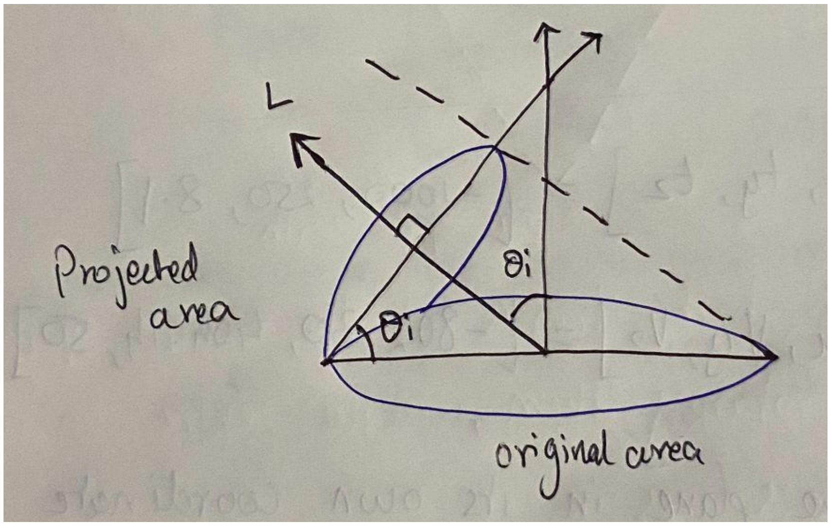
# Q1 (a)

We know that the surface radiance $L = \frac{\rho d}{\pi} . I . \cos\theta_i$



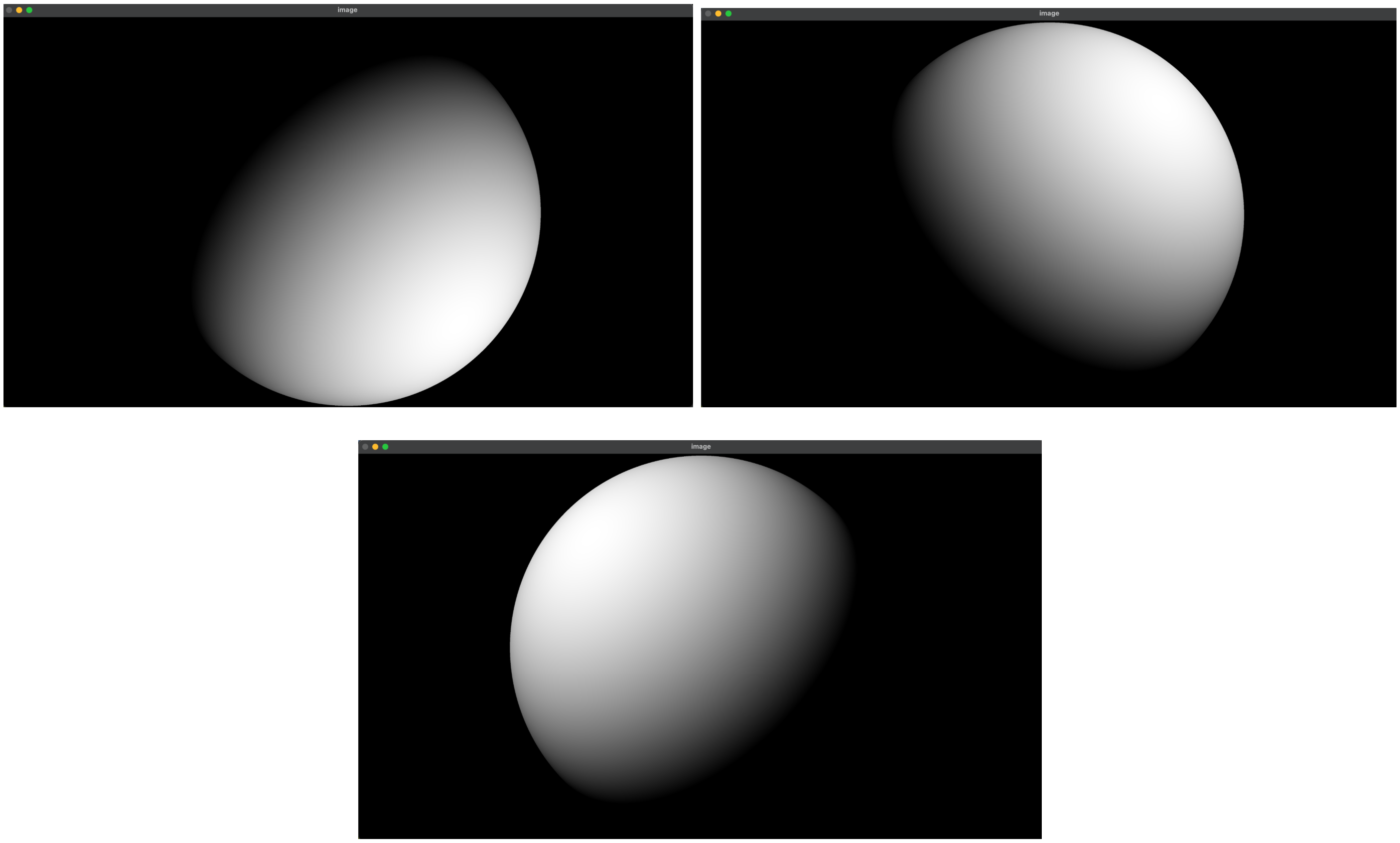

By geometry, we can say that the original area = projected area * $\cos\theta_i$

We know that the angle between L and the normal (n) is $\theta_i$.

Hence, $\cos\theta_i$ = n.L

# Q1 (b)

N-dot-l lighting can be rendered on a sphere as follows:




*Figures: Renderings of a sphere with 3 different lighting conditions*

Q1 (c)

```python
def loadData(path = "../data/"):
    """
    Question 1 (c)

    Load data from the path given. The images are stored as input_n.tif
    for n = {1...7}. The source lighting directions are stored in
    sources.mat.

    Paramters
    ---------
    path: str
        Path of the data directory

    Returns
    -------
    I : numpy.ndarray
        The 7 x P matrix of vectorized images

    L : numpy.ndarray
        The 3 x 7 matrix of lighting directions

    s: tuple
        Image shape

    """

    I = None
    L = None
    s = None
    images = []
    for i in range(0,7):
        image_path = path + "input_" + str(i+1) + ".tif"
        Im = cv2.imread(image_path, -1)
        Im_xyz = skimage.color.rgb2xyz(Im)
        Im_y = Im_xyz[:, :, 1].reshape(-1, 1)
        images.append(Im_y)

    I = np.asarray(images).reshape(len(images), -1)
    L = np.load(path + 'sources.npy').T
    sample_image = cv2.imread(image_path, -1)
    s = sample_image.shape[:2]
    return I, L, s
```

*Figure: Screenshot of the function* loadData

# Q1 (d)

The rank of the matrix I is supposed to be 3, since with 3 different light source directions, the intensity matrix I should have rank 3. Three directions are required since n is defined in 3 coordinates.

Upon performing SVD, we get matrix A with rank 7, with the values:

[72.40617702, 12.00738171, 8.42621836, 2.23003141, 1.51029184, 1.17968677, 0.84463311]

This doesn't agree with the 3-rank requirement, perhaps because the image is non-ideal. The image is capturing inter-reflected lights, and noise, hence leading to 7 values per pixel instead of 3

# Q1 (e)

We have the equation: $I = L^T B$

Therefore, we can say: $B = L^{T-1}. I$

Since L is not a square matrix, so it is non-invertible, so we can take a pseudo-inverse of the matrix:

$B = (L.\ L^T)^{-1} L\ .\ I$

## Q1 (f)

*Figure: (left) Normals Image, (right) Albedo image*

Certain dark areas of the image like the nose, ears and neck, are being represented as being brighter in the albedo image, since the model assumes directional lighting and doesnt account for shadows, violating the n-dot-l model.

# Q1 (g)

Getting depth from normals, we have

$V_1 = (x+1,\ y,\ z_{x+1,\ y}) - (x,\ y,\ z_{xy})$

$V_1 = (1,\ 0,\ z_{x+1,y} - z_{xy})$

$0 = N.\ V_1$

$0 = (n_x,\ n_y,\ n_z)\ .\ (1,\ 0,\ z_{x+1,y} - z_{xy})$

$0 = n_1 + n_3\ (\ z_{x+1,y} - z_{xy})$

Similarly,

$V_n = (1,\ 0,\ z_{x+1,y} - z_{xy})$

$0 = N.\ V_2$

$0 = n_2 + n_3\ (\ z_{x+1,y} - z_{xy})$

Therefore, taking parial derivatives, we get:

$$\frac{\partial f(x,y)}{\partial x} = -\frac{n1}{n3}$$

$$\frac{\partial f(x,y)}{\partial y} = -\frac{n2}{n3}$$

# Q1 (h)

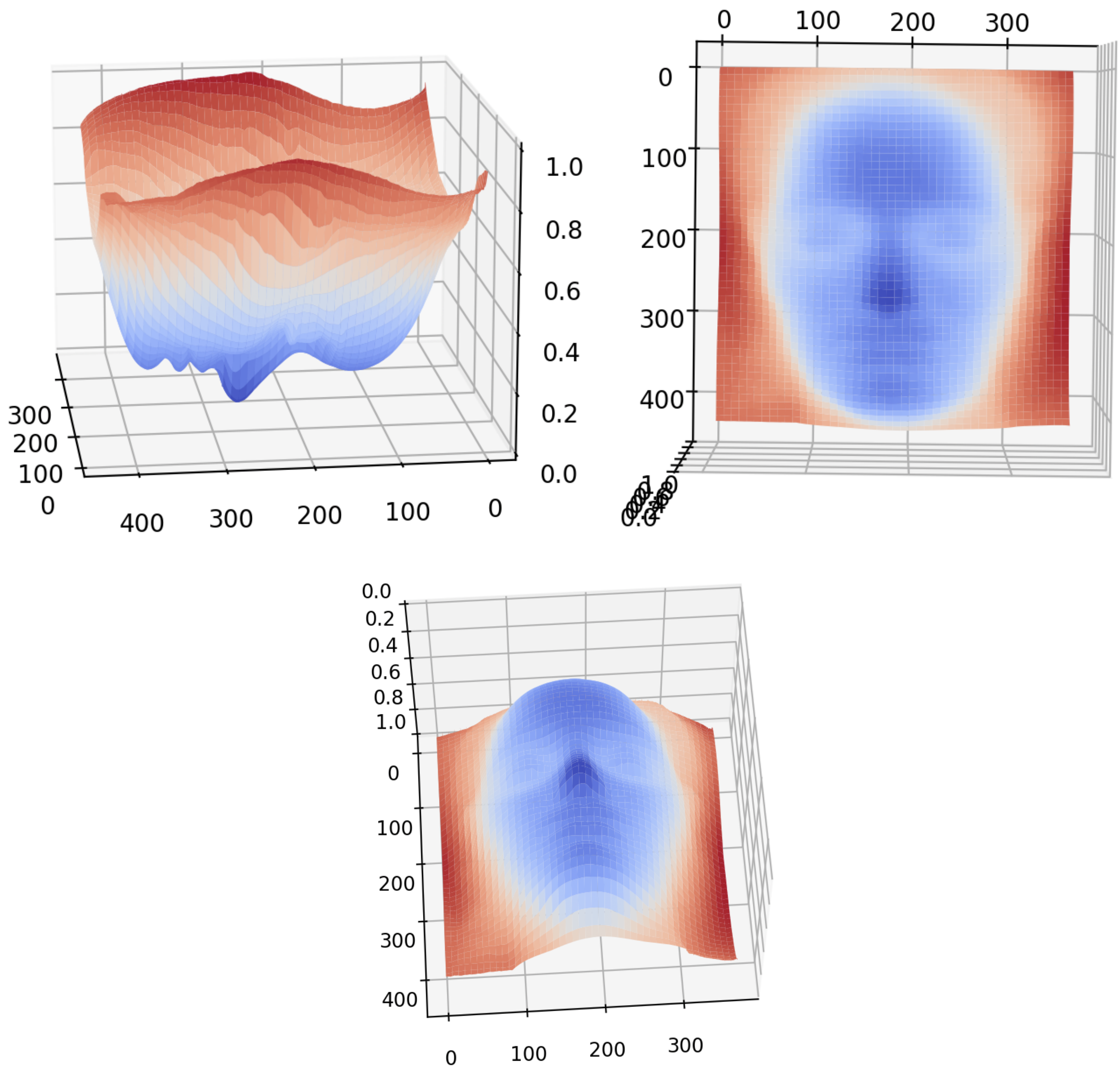
$$g_x = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

$$g_y = \begin{bmatrix} 4 & 4 & 4 & 4 \\ 4 & 4 & 4 & 4 \\ 4 & 4 & 4 & 4 \\ 4 & 4 & 4 & 4 \end{bmatrix}$$

The reconstructed g matrix using both the methods is the same.

The gradients $g_x$ and $g_y$ can be made non integrable in many ways. Firstly, when the gradient elements are negative, the addition of the gradient in the x and y directions becomes distinct, resulting in varied outcomes for 'g'. Secondly, discrepancies arise if the values of elements within the gradient matrix are unequal, leading to variations in 'g'. Lastly, noise presence can also render the gradients estimated in 'g' non-integrable, adding to the complexity of ensuring non-integrability in both $g_x$ and $g_y$.

Q1 (i)




*Figures: The surface plot after applying the Frankot Chellappa algorithm*

## Q2 (a)

The rank of the intensity matrix should be 3, but as seen in the previous question, the rank is 7.

We first start by taking the SVD of I: $I = U\Sigma V^T$. We get our new U and V by taking the first 3 columns of the current values. And we get our new $\Sigma$ by taking the upper left 3x3 values of it.

Using these new values, we get $L = U_{new}\Sigma_{new}$ and $B = \Sigma_{new} V^T$

Q2 (b)

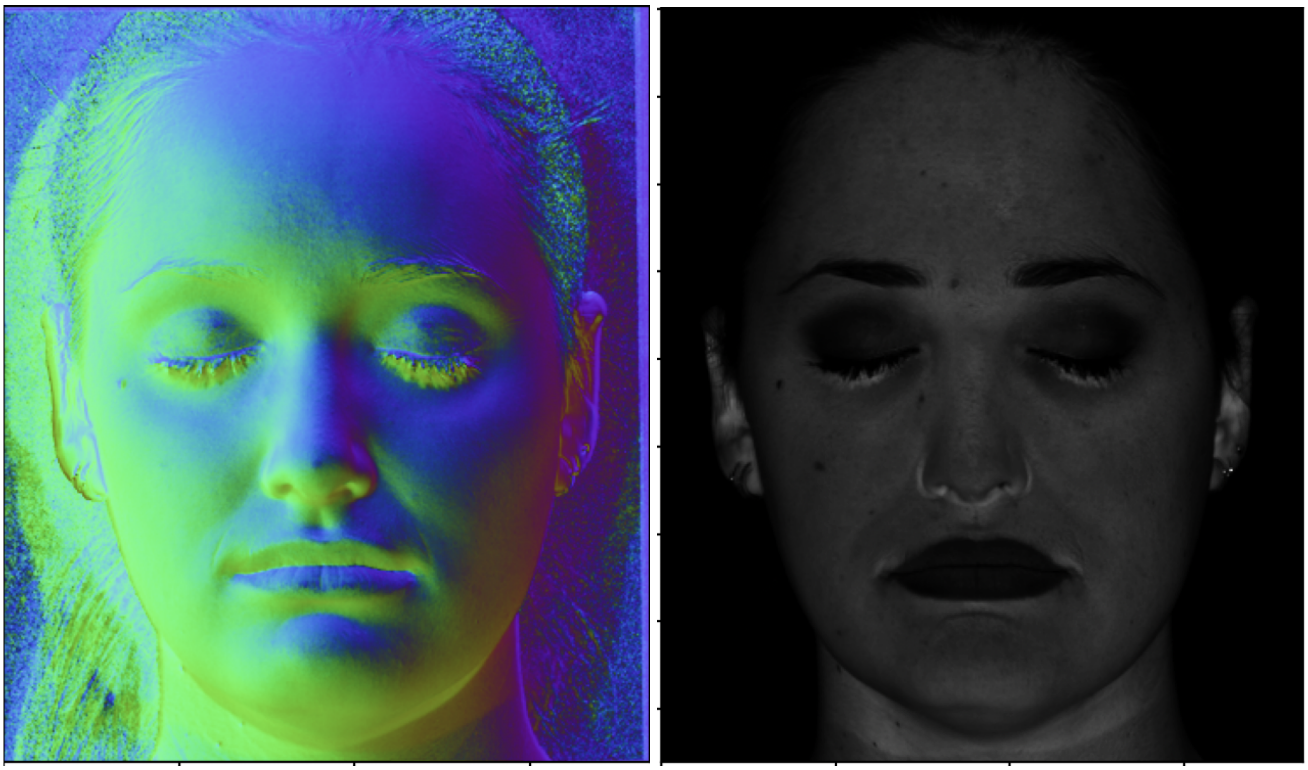


*Figure: (left) Normals Image, (right) Albedo image*

## Q2 (c)

Ground Truth Lighting L

*[[-0.1418  0.1215 -0.069   0.067  -0.1627  0.      0.1478]*

*[-0.1804 -0.2026 -0.0345 -0.0402  0.122   0.1194  0.1209]*

*[-0.9267 -0.9717 -0.838  -0.9772 -0.979  -0.9648 -0.9713]]*

L obtained by factorization:

*[[-0.33516968  0.25970881  0.61958053  0.59813427 -0.04764929 -0.21919104*
*0.16808572]*

*[-0.43508879 -0.6404168   0.33253561 -0.36780814 -0.38764433 -0.02590868*
*0.06133449]*

*[-0.27028587  0.13745369  0.14128625 -0.1016158   0.15061015 -0.04284624*
*-0.92371275]]*

These values are not similar

Normalizing L and B and multiplying B with a G matrix could change B without modifying rendering results.

## Q2 (d)

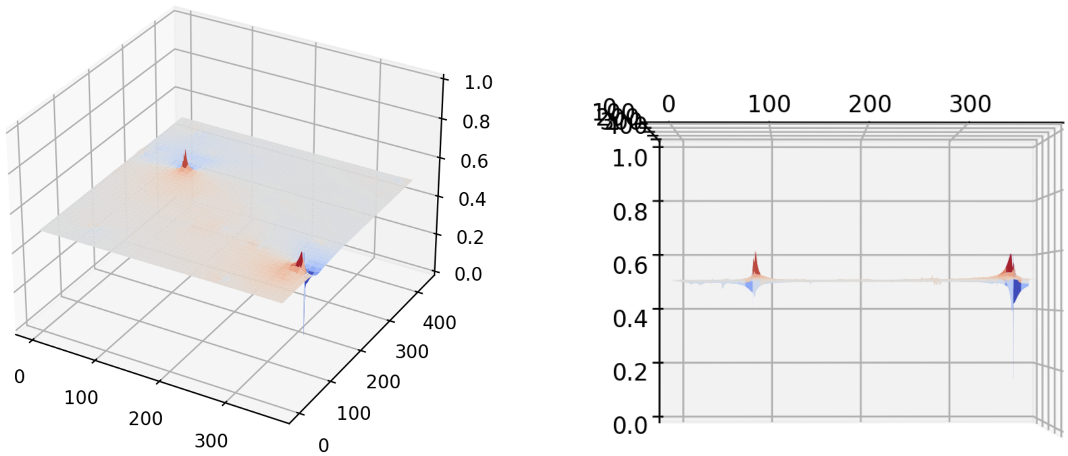



*Figure: Surface Plot visualization*

No, this does not look like a face.

Q2 (e)

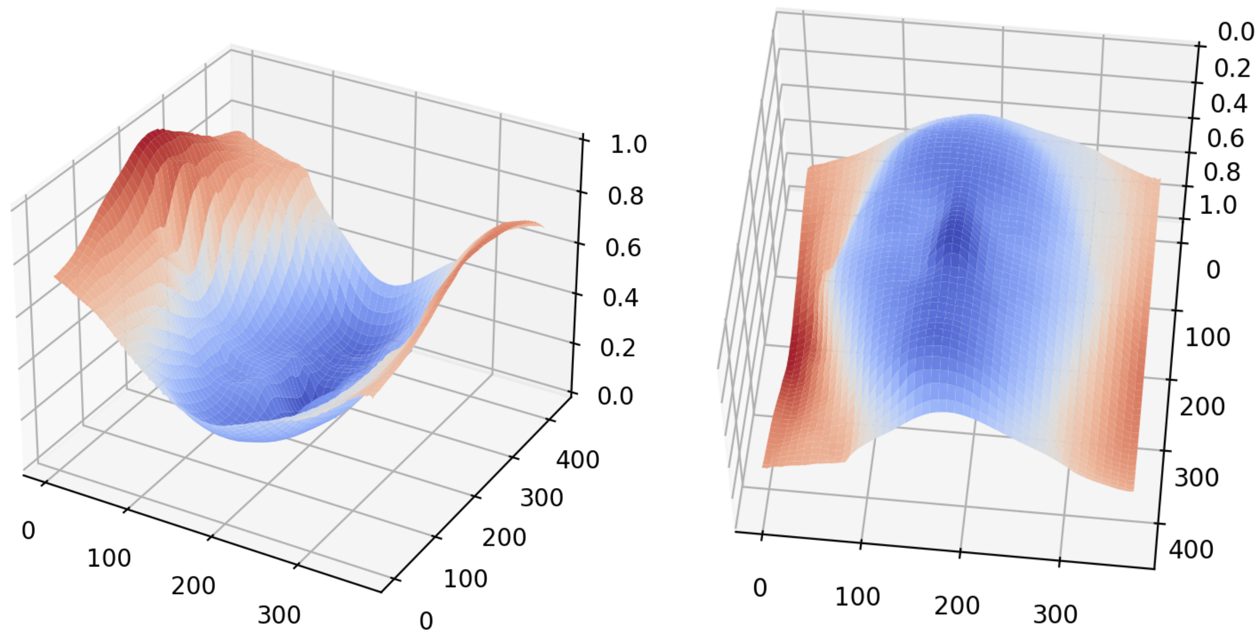
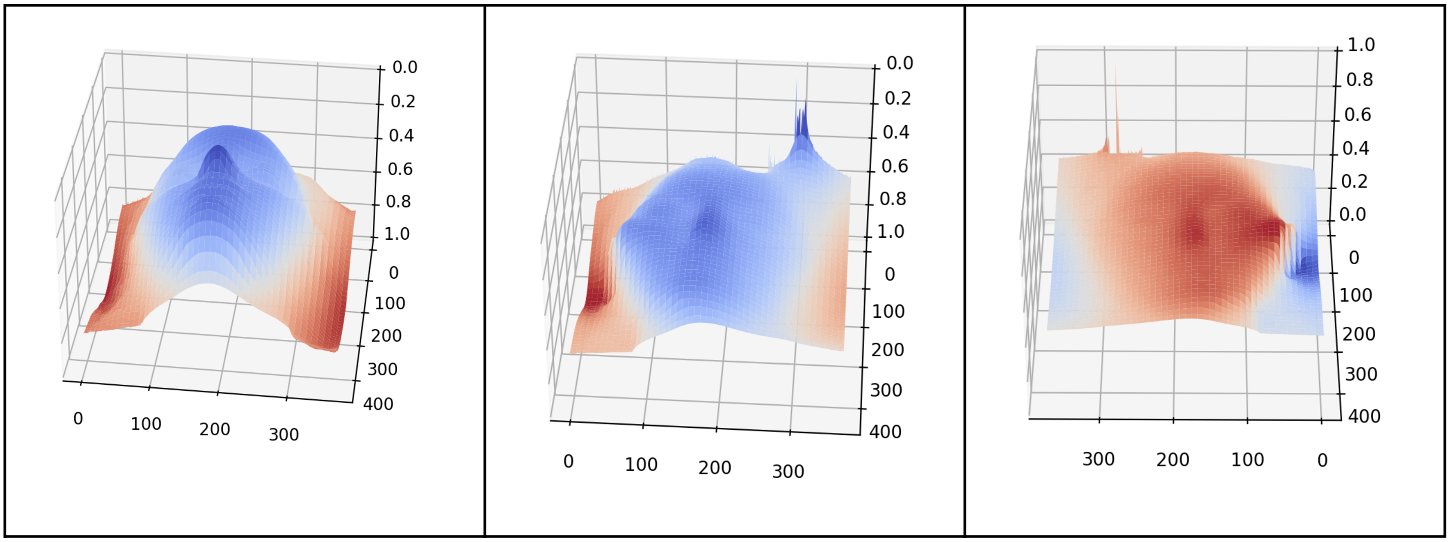



*Figure: Surface Plot visualization*

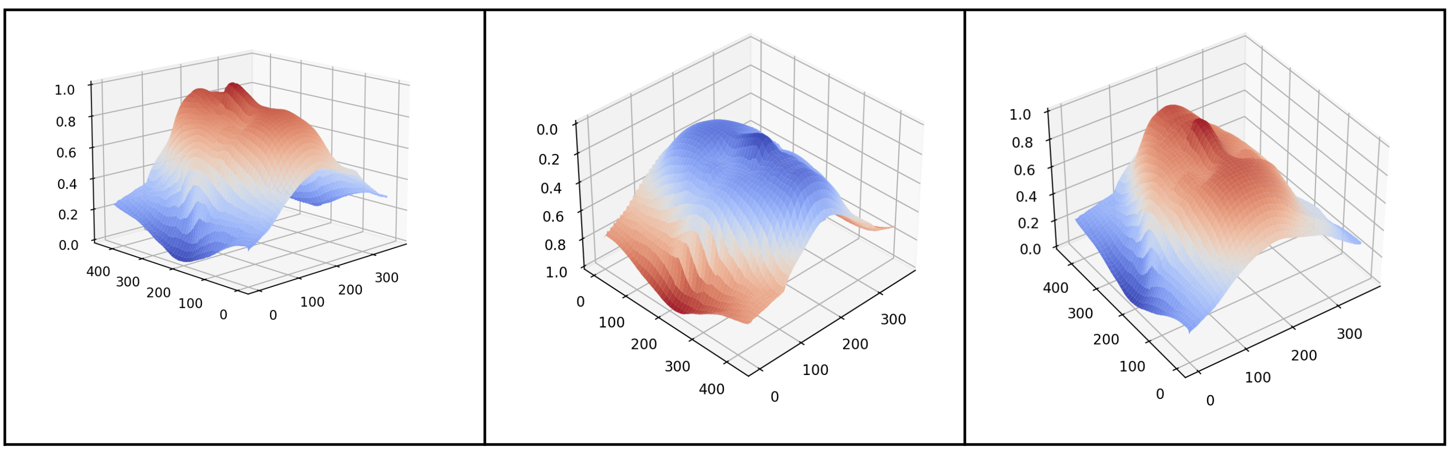
Enforcing intgeribility makes the results look much closer to the expected results.

# Q2 (f)



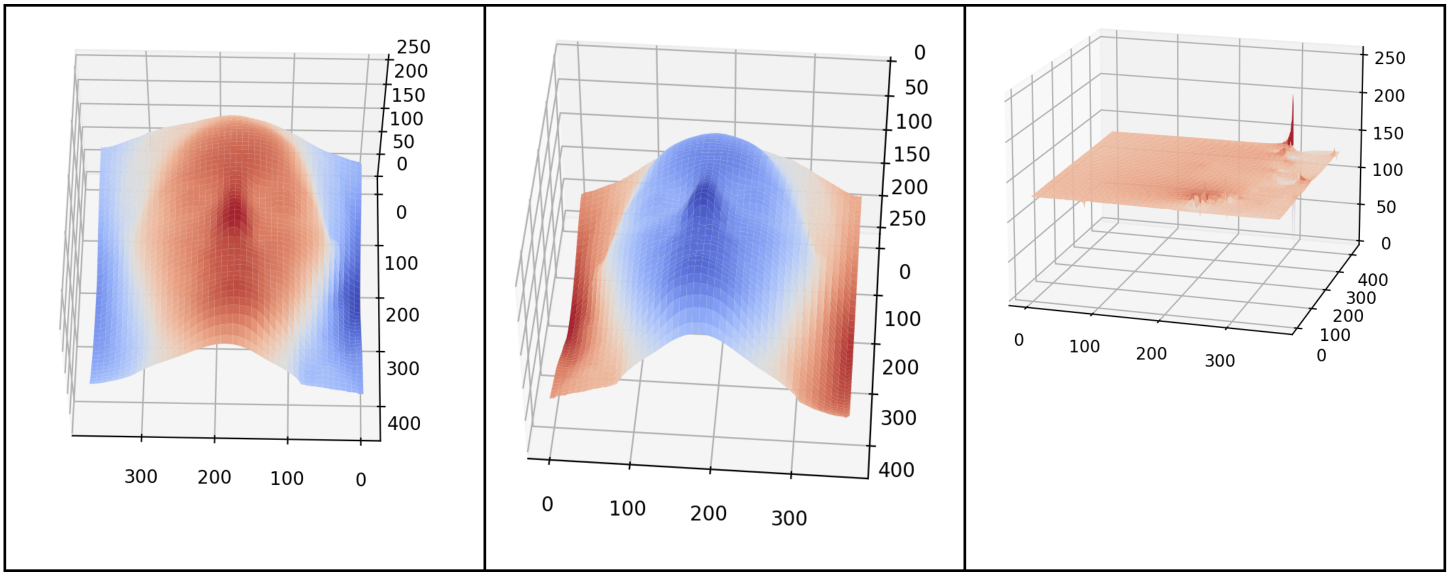

Figures: Surface plot with $\mu$ = (left) 5, (middle) 0.2, (right) -0.5 and $\lambda$ = 1 $\nu$ =1

As the value of miu reduces, the plot becomes flatter




Figures: Surface plot with $\nu$ = (left) 5, (middle) 0.2, (right) -0.5 and $\lambda$ = 1 $\mu$ =1

Changing the value of $\nu$ leads to one side of the face getting stretched




Figures: Surface plot with $\lambda$ = (left) -1, (middle) 2, (right) 5 and $\nu = 1$ $\mu = 1$

Changing the value of $\lambda$ changes the ability of the plot to capture depth

Bas-relief ambiguity is so named, since when the relief is low, there is ambiguity in the recovery of the surface

## Q2 (g)

With $\lambda$ set to a very large value and $\mu$ & $\nu$ set to zero, the estimated surface will be the flattest.

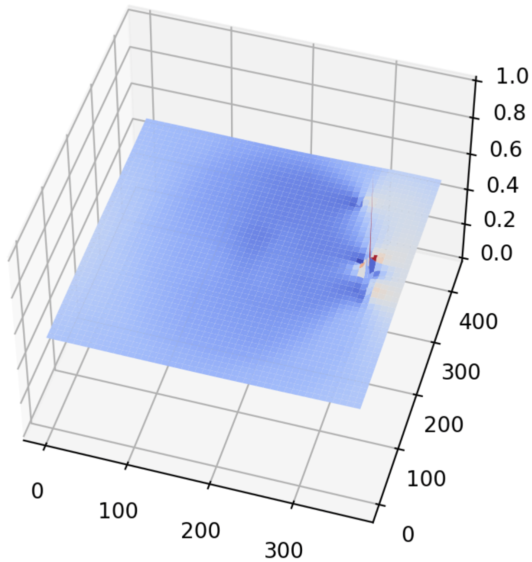



*Figure: Surface plot visualization*

## Q2 (h)

No, acquiring more pictures will not help solve the problem, since the current shadows will only increase, hence increasing the current ambiguity.