

```
import numpy as np
from matplotlib import pyplot as plt
from skimage import data, img_as_float, color, exposure
from skimage.restoration import unwrap_phase

phaseimg = np.load('/content/measured_phase_map_20MHz.npy')

unwrapped = np.unwrap(phaseimg, discontinuity=2*np.pi, axis=0)
plt.imshow(unwrapped)
plt.colorbar()
plt.savefig('depthestimate.png')
```

```
import numpy as np
import cv2 as cv
from matplotlib import pyplot as plt

groundTruth = np.load('/content/ground_truth_depth.npy')
plt.imshow(groundTruth)
plt.colorbar()
plt.savefig('depth.png')
```

```
import numpy as np
import cv2 as cv
from matplotlib import pyplot as plt

# Loading exposure images into a list
img_fn = ["/content/one_over_eight.png", "/content/one_over_four.png", "/c
img_list = [cv.imread(fn) for fn in img_fn]

# Exposure fusion using Mertens
merge_mertens = cv.createMergeMertens()
res_mertens = merge_mertens.process(img_list)

# Convert datatype to 8-bit and save
res_mertens_8bit = np.clip(res_mertens*255, 0, 255).astype('uint8')

cv.imwrite("fusion_1d.jpg", res_mertens_8bit)
```

True

```
import numpy as np
import cv2 as cv
from matplotlib import pyplot as plt
```

✓ 0s completed at 12:14 AM

● ✕

```
img_fn = ["/content/one_over_eight.png", "/content/one_over_four.png", "/content/one_over_two.png"]  
img_list = [cv.imread(fn) for fn in img_fn]  
exposure_times = np.array([100, 250, 250], dtype=np.float32)
```

```
# Merge exposures to HDR image  
merge_debevec = cv.createMergeDebevec()  
hdr_debevec = merge_debevec.process(img_list, times=exposure_times.copy())
```

```
# Tonemap HDR image  
tonemap1 = cv.createTonemap(gamma=2.2)  
res_debevec = tonemap1.process(hdr_debevec.copy())
```

```
# Convert datatype to 8-bit and save  
res_debevec_8bit = np.clip(res_debevec*255, 0, 255).astype('uint8')
```

```
cv.imwrite("ldr_1c.jpg", res_debevec_8bit)
```

True