```
import generator as gen
import cv2
import numpy as np
import matplotlib.pyplot as plt, matplotlib.cm as cm

## Provides standard methods to pre process the image

def normalize(im1, im2):
    norm1 = im1/im1.mean()
    norm2 = im2/im2.mean()
    return norm1, norm2

def blur_images(im1, im2, bluramount = 5):
    blur1 = cv2.GaussianBlur(im1, (bluramount, bluramount), 0)
    blur2 = cv2.GaussianBlur(im2, (bluramount, bluramount), 0)
    return blur1, blur2

def inside_mask(im1, im2, threshold = 1):
    mask1 = (im1 > threshold)
    mask2 = (im2 > threshold)
    return mask1, mask2, np.logical_and(mask1, mask2)

def baseline_process(preim, postim):
    preim, postim = normalize(preim, postim)
    preim, postim = blur_images(preim, postim)

    return preim, postim


if __name__ == "__main__":
    zern = [-0.2868445,
            0,
            0.08498557,
            -0.16551221,
            0,
            0.16330520,
            0.03002613,
            0,
            0.00034476,
            0,
            -0.0008443]
    plt.ion()
    plt.show()

    for defocus in [0.1e-3, 0.2e-3, 0.4e-3, 0.8e-3, 1.6e-3, 3.2e-3, 6.4e-3]:
        preim, postim = gen.generate_images(zern, defocus = defocus)
        preim, postim = normalize(preim, postim)
        #preim, postim = blur_images(preim, postim)

        diff = preim - postim
        plt.subplot(1,3,1)
        plt.title("Prefocal")
        plt.imshow(preim, cmap=cm.gray)
        plt.subplot(1,3,2)
        plt.title("Postfocal")
        plt.imshow(postim, cmap=cm.gray)
        plt.subplot(1,3,3)
        plt.title("Difference")
        plt.imshow(diff, cmap=cm.gray)
        plt.draw()
        plt.pause(0.001)
        plt.savefig("plots/" + 'OurZernikies' + '_{}um'.format(int(defocus*1e6
)) + ".png")
```