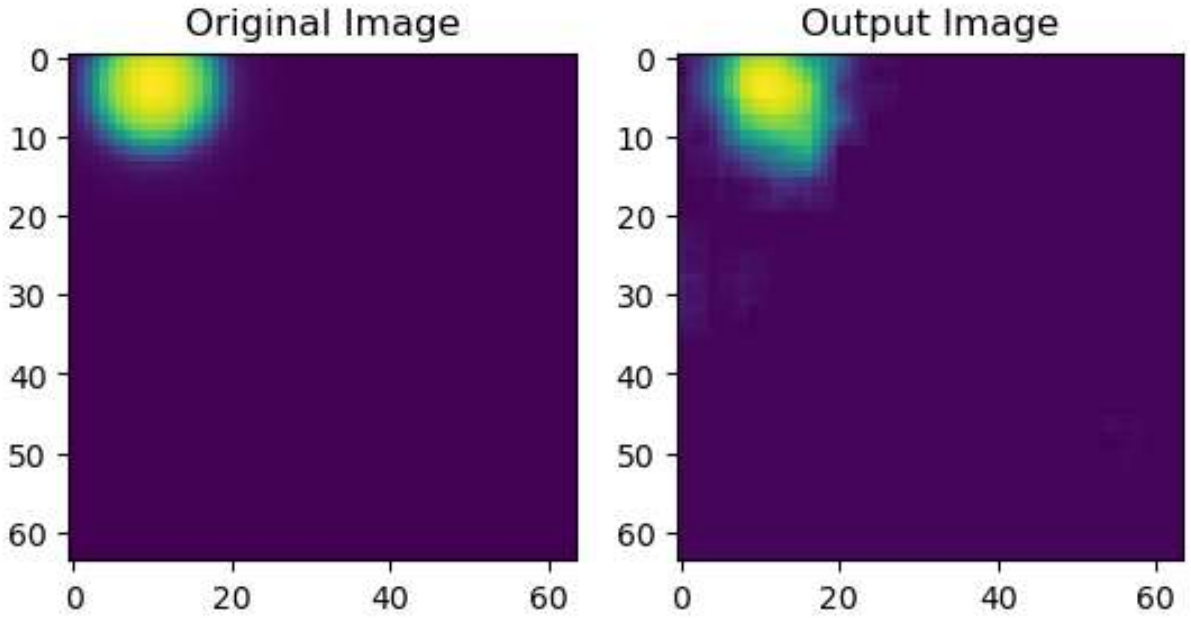
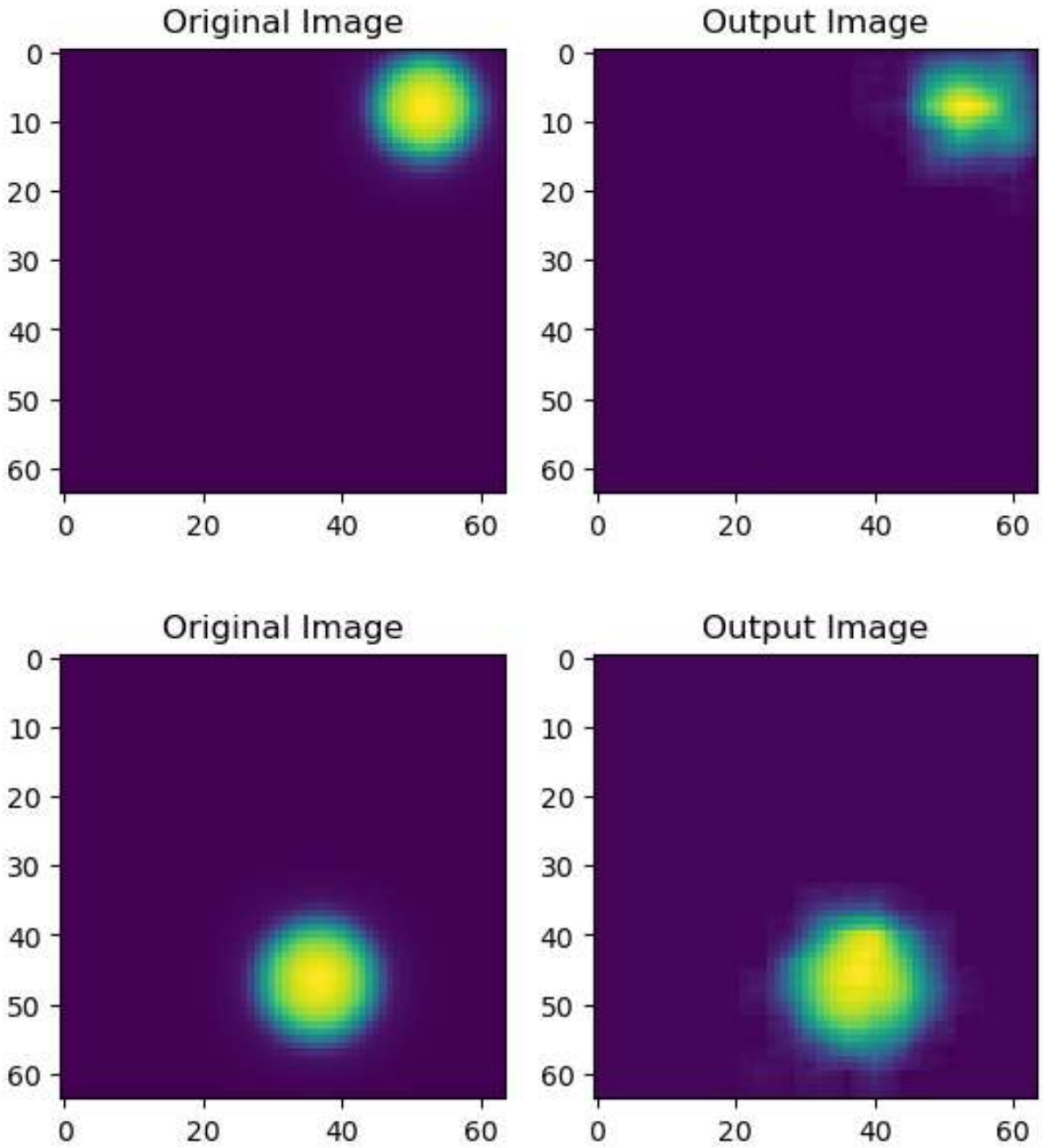
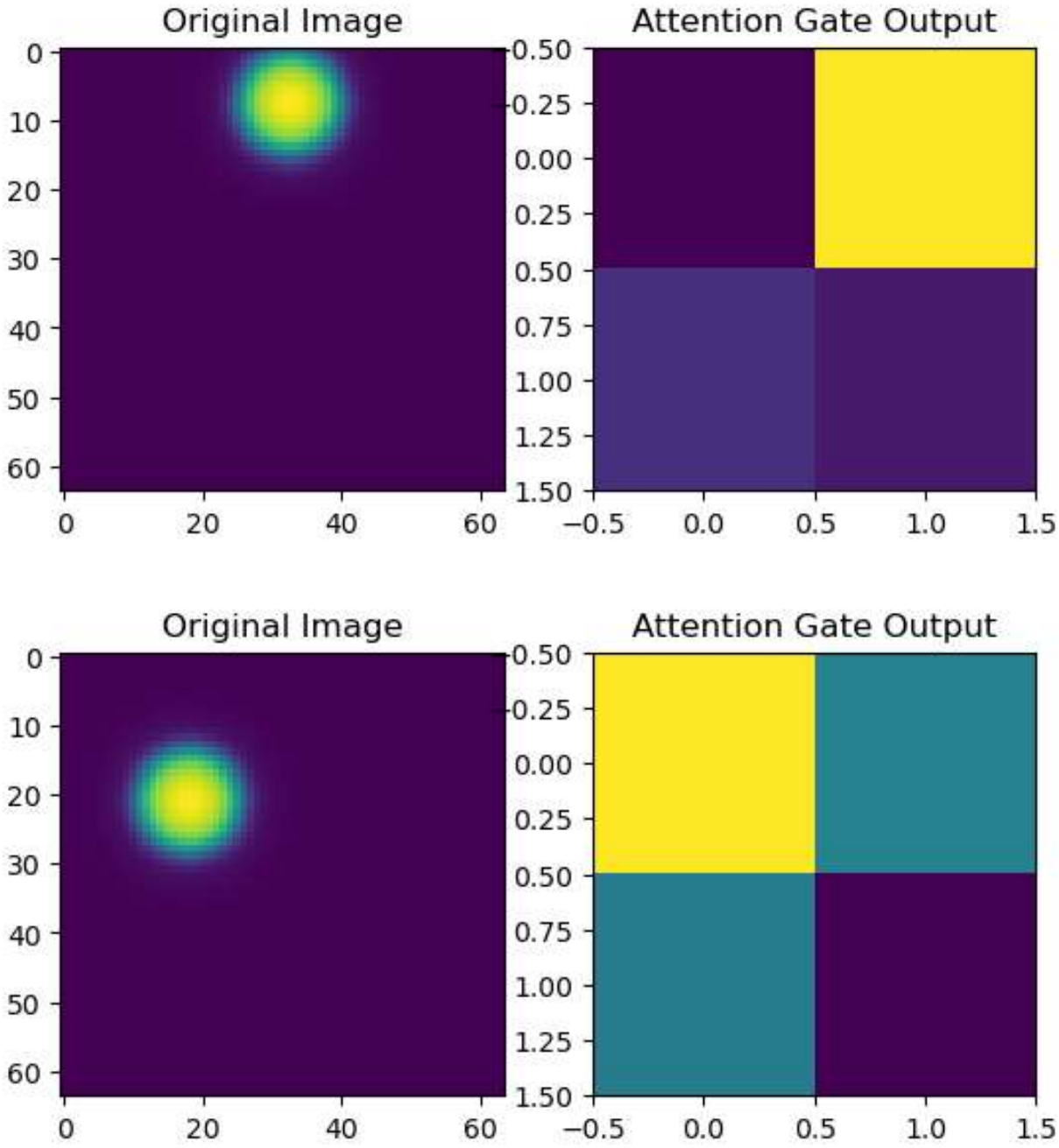
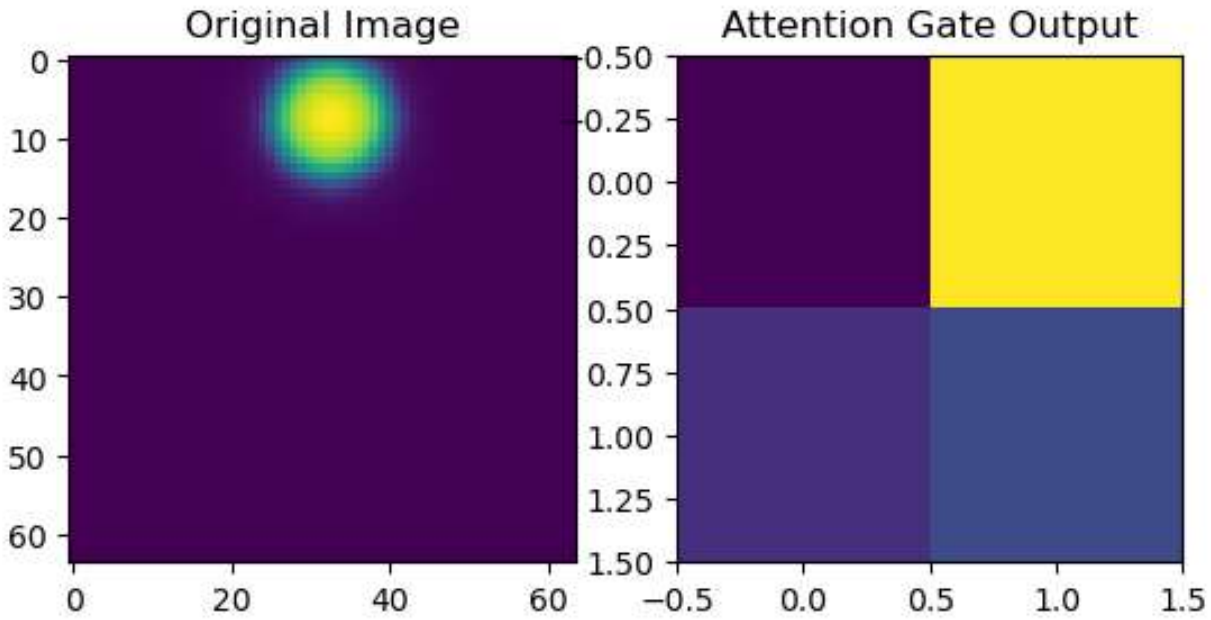
In [ ]:
```
for data in test_loader:
    img = data[torch.randint(low=0, high=31, size=(1,)).item(), :, :, :]
    output_img = gpu_model(img.permute(2, 0, 1).unsqueeze(0).to(device)).cpu()
    output_img = output_img.detach().squeeze(0).permute(1, 2, 0)
    figure = plt.figure()
    subplot1 = figure.add_subplot(1, 2, 1)
    subplot1.imshow(img)
    subplot1.set_title("Original Image")

    subplot2 = figure.add_subplot(1, 2, 2)
    subplot2.imshow(output_img)
    subplot2.set_title("Output Image")

    plt.show()
    break
```

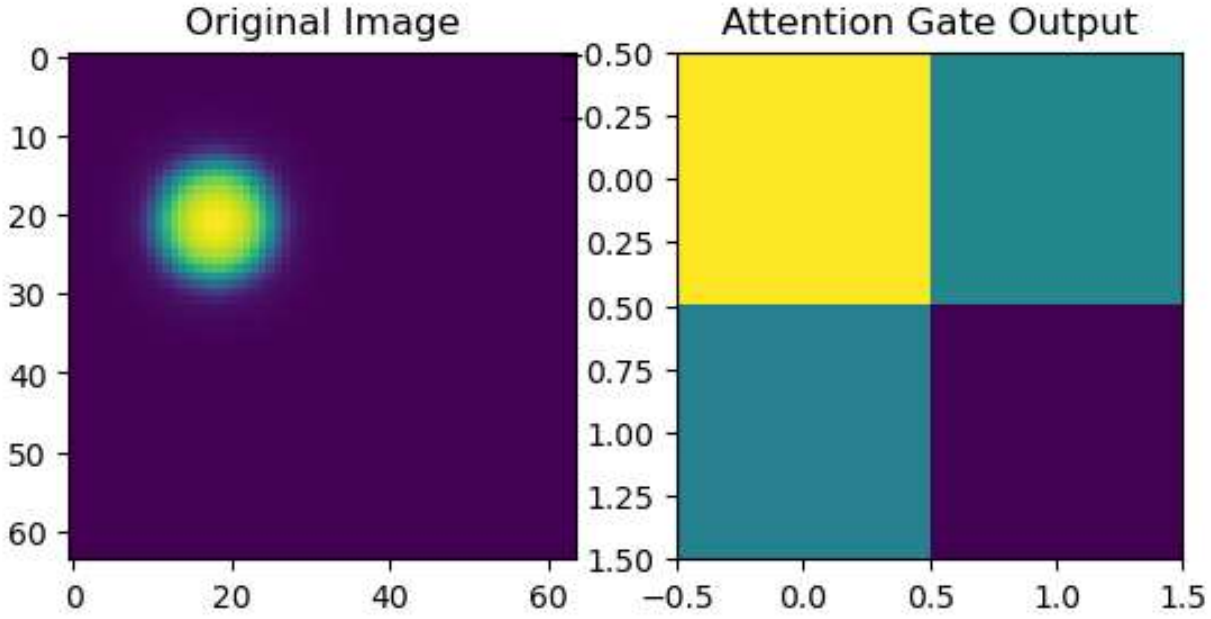


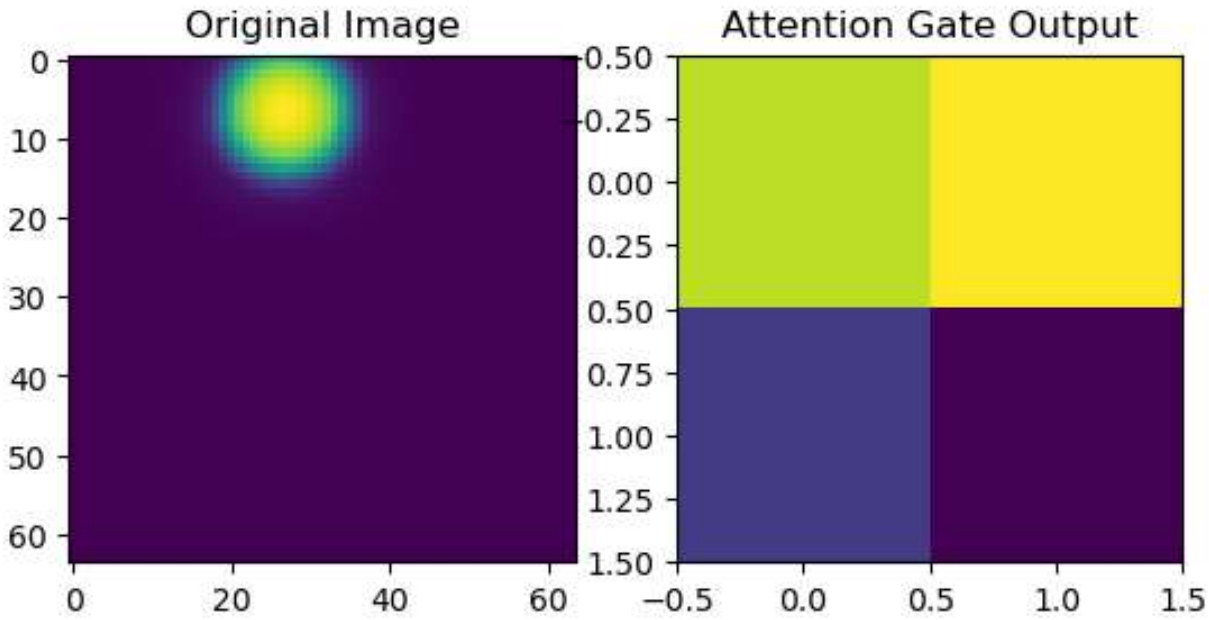

Original Image
Output Image
Original Image
Output Image
0
10
20
30
40
50
60
0
20
40
60

In [ ]:
```
for data in test_loader:
    img = data[torch.randint(low=0, high=31, size=(1,)).item(), :, :, :]

    attention_gate_output = (
        gpu_model.attention_gate_output(img.permute(2, 0, 1).unsqueeze(0).to(device))
        .detach()
        .squeeze(0)
        .cpu()
    )
    out_img_mean = attention_gate_output.norm(dim=1)[1:].reshape(2, 2)

    figure = plt.figure()
    subplot1 = figure.add_subplot(1, 2, 1)
    subplot1.imshow(img)
    subplot1.set_title("Original Image")

    subplot2 = figure.add_subplot(1, 2, 2)
    subplot2.imshow(out_img_mean)
    subplot2.set_title("Attention Gate Output")

    plt.show()
    min_val = attention_gate_output[:-1].min()
    max_val = attention_gate_output[:-1].max()

    plt.show()

    break
```


Original Image
Attention Gate Output
Original Image
Attention Gate Output


Original Image
Attention Gate Output
0
10
20
30
40
50
60
0
20
40
60
0.50
0.25
0.00
0.25
0.50
0.75
1.00
1.25
1.50
−0.5
0.0
0.5
1.0
1.5


Original Image
Attention Gate Output
0
10
20
30
40
50
60
0
20
40
60
0.50
0.25
0.00
0.25
0.50
0.75
1.00
1.25
1.50
−0.5
0.0
0.5
1.0
1.5


Original Image
Attention Gate Output
0
10
20
30
40
50
60
0
20
40
60
0.50
0.25
0.00
0.25
0.50
0.75
1.00
1.25
1.50
−0.5
0.0
0.5
1.0
1.5