

```
In 1 1 from google.colab import drive
2 import torch
3 drive.mount('/content/drive')
4 %cd /content/drive/MyDrive/DepthSense

Mounted at /content/drive
/content/drive/MyDrive/DepthSense

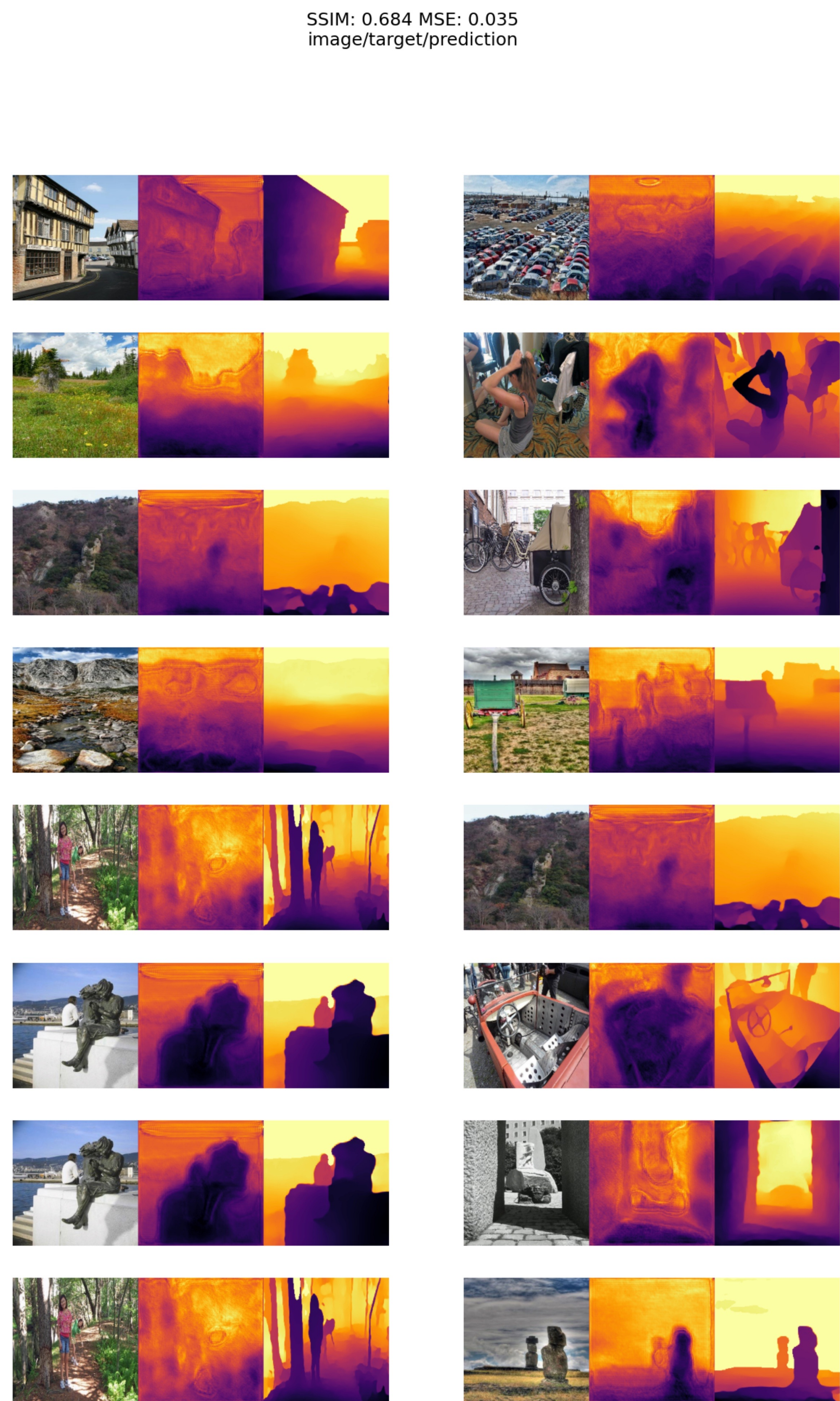
In 21 1 best_epoch

Out 21 4

In 23 1 from ResnetFF import ResnetFF
2 model = ResnetFF()
3 best_sd = torch.load('ResnetFF.pt')
4 model.load_state_dict(best_sd)
5

Out 23 <All keys matched successfully>

In 24 1 all_imgs, all_preds, all_targets = [], [], []
2 with torch.no_grad():
3     with autocast():
4         for img, mask in tqdm(test_dl, total=len(test_dl)):
5             img, mask = img.to(device), mask.to(device)
6             preds = model(img)
7             all_imgs.append(img)
8             all_preds.append(preds)
9             all_targets.append(mask)
10
11 test_metrics = metrics.clone()
12 test_metrics(
13     torch.vstack(all_preds),
14     torch.vstack(all_targets)
15 )
16 m = test_metrics.compute()
17 title = f"SSIM: {m['StructuralSimilarityIndexMeasure'].cpu().item():.3f} MSE: {m['MeanSquaredError'].cpu().item():.3f}"
18 plot_vals(
19     torch.vstack(all_imgs).cpu(),
20     torch.vstack(all_preds).cpu(),
21     torch.vstack(all_targets).cpu(),
22     n=16,
23     figsize=(10,15),
24     title=title
25 )
26
27
```

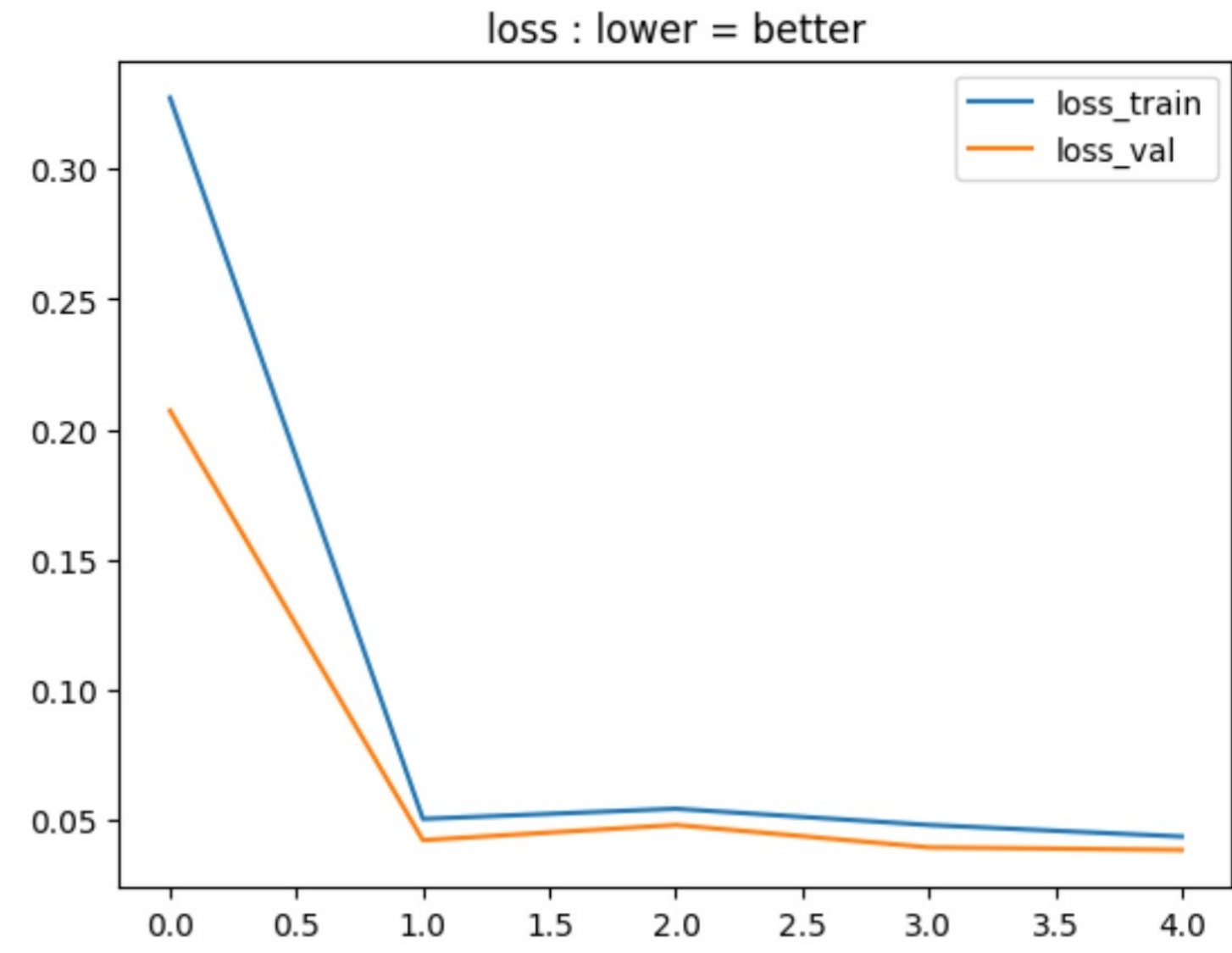


```
In 25 1 logs

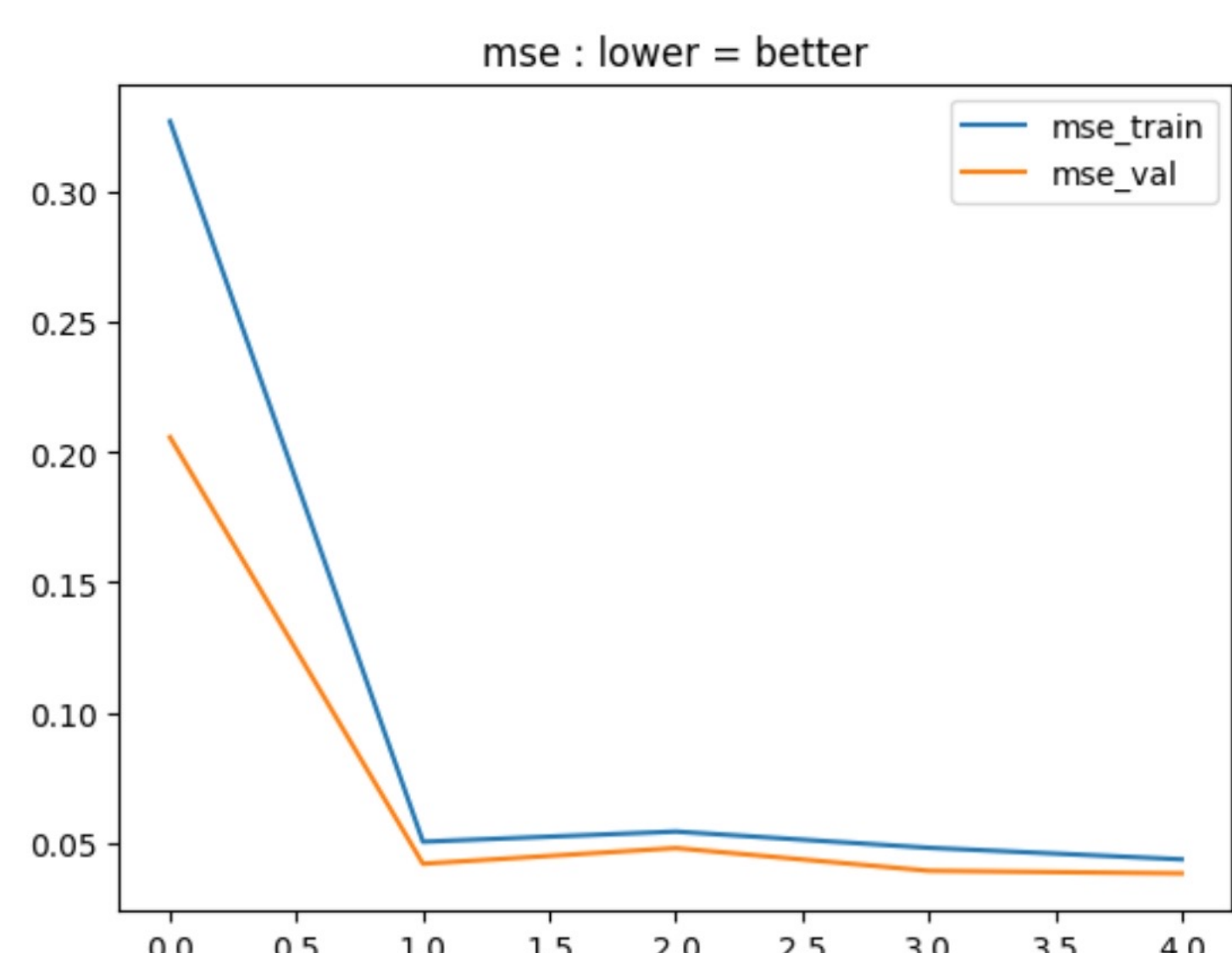
Out 25
```

	loss_train	loss_val	ssim_train	ssim_val	mse_train	mse_val
0	0.327318	0.2071	0.133579	0.425158	0.327318	0.20583
1	0.050247	0.042086	0.58262	0.608989	0.050247	0.041778
2	0.054153	0.047912	0.647623	0.622574	0.054153	0.047796
3	0.047921	0.039307	0.680513	0.654263	0.047921	0.039069
4	0.043474	0.038338	0.688893	0.655723	0.043474	0.038063

```
In 26 1 logs['loss_train'].plot()
2 logs['loss_val'].plot()
3 plt.title('loss : lower = better')
4 plt.legend()
5 plt.show()
```



```
In 27 1 logs['mse_train'].plot()
2 logs['mse_val'].plot()
3 plt.title('mse : lower = better')
4 plt.legend()
5 plt.show()
```



```
In 28 1 logs['ssim_train'].plot()
2 logs['ssim_val'].plot()
3 plt.title('ssim : higher = better')
4 plt.legend()
5 plt.show()
```

