IPR

Shiv Narayan

October 2024

GitHub Repository for the Project

1 Training Details

The model was trained on the GPU of the CC server provided by IIT Kanpur. The training process took approximately 26 seconds per epoch, with a total of 500 epochs. The total training time was recorded as 3 hours, 35 minutes, and 1 second. For the rest of the details refer to the GitHub repository.

2 Dataset Description

The dataset initially considered for this project was the NYUv2 depth dataset. It consists of around 280,000 RGB-D (color and depth) images captured from various indoor environments using the Microsoft Kinect sensor. However, due to the extensive size of the dataset and the limited computational resources, it was not feasible to train our models on the full dataset. Hence, I made use of a slimmed-down version of the dataset, which contains approximately 12,000 images.

3 Errors and Resolutions

During the implementation of the model, several errors were encountered and addressed as follows:

1. **Directory Errors:** Multiple directory errors occurred due to the absence of required folders in the file system. These issues were resolved by creating the necessary directories using the following commands:

```
mkdir -p data/nyu_v2/train/depths
mkdir -p data/nyu_v2/train/images
mkdir -p data/nyu_v2/val/depths
mkdir -p data/nyu_v2/val/images
mkdir -p models/den_gen2_v122orig
```

```
mkdir -p models/pretrained_resnet
mkdir -p models/temp_v3
```

2. **Missing Pre-trained Model:** The pre-trained ResNet model required for the training process was not readily available. I downloaded the ResNet model manually and adapted it according to the model architecture used in this project.

3. **Training Efficiency:** To improve the efficiency of the training process, early stopping was implemented to avoid overfitting. Additionally, the batch size was adjusted to better fit the available computational resources, which contributed to smoother training and faster convergence.

4 Results

The table below summarizes the comparison of the Root Mean Square Error (RMSE) between the original paper and my implementation:

Dataset Size	**RMSE (Original Paper)**	**RMSE (This Work)**
280,000 images	0.572	-
12,000 images	-	2.3812

Table 1: Comparison of RMSE between Original Paper and This Work

As shown, the original paper reported an RMSE of 0.572 with the full NYUv2 dataset, while my implementation using a reduced dataset of 12,000 images resulted in an RMSE of 2.3812. The performance drop is expected due to the smaller dataset size, but it is anticipated that training on the full dataset would yield results closer to the original paper's findings.

5 Error Screenshots

Below are two screenshots of the errors encountered during the implementation process:

	RMSE (lin)
Zoran *et al.* [43]	1.220
Li *et al.* [20]	0.821
Liu *et al.* [21]	0.824
Baig *et al.* [1]	0.802
Eigen *et al.* [6]	0.877
Wang *et al.* [38]	0.745
Roy *et al.* [28]	0.744
Eigen and Fergus [5]	0.641
Chakrabarti *et al.* [3]	0.620
Laina *et al.* [19]	0.597
Proposed	**0.572**

Figure 1: Original Error

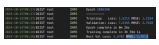

Figure 2: My Result Error