A comparative study of the impact of Loss functions on Monocular Depth Estimation

Himanshu Singhal
Department of Computer Science
Virginia Tech
himanshusinghal@vt.edu

Harish Ravi
Department of Computer Science
Virginia Tech
harishr@vt.edu

Abstract

Monocular Depth Estimation involves estimating the depth of each individual pixel of an input image. In this work, we perform this task using multiple Encoder-Decoder models. The input is an image consisting of three channels, and the output is the depth cue of every individual pixel in this image. We look to solve this problem with Encoder-Decoder models because of the rich representation capabilities of these classes of models, hence making an Encoder-Decoder model, the model of choice for this particular use case. We adopted multiple models to solve the problem, one by leveraging transfer learning to get a base encoder from Densenet169 and upsampling the encoded image representations to produce the depth cues. The second was to use a Vision transformer to predict the depth masks. The DPT transformer model was shown to perform quite well for the task of Monocular Depth Estimation. In this project, we study the impact of various techniques of model optimization like domain specialized loss functions on the performance of Encoder-Decoder models as well as how the various model architectures impact the performance. Test accuracy of around 41.167% was obtained by training the U-net architecture model while the transformer models gave an accuracy of about 88.63% on average using loss functions such as Mean Squared Error, Huber Loss, and Structured Similarity Index Measure.

1 Introduction

Depth estimation is a vital problem to address in many autonomous systems that will be the cornerstone of Artificial Intelligence of tomorrow like self- driving cars, augmented reality, and robotics. Self-driving cars and Robots need 3-dimensional information about the scene in order to make decisions and accurately move the actuators. Similarly, augmented reality requires depth information so as to calibrate the orientation and scale of the object. The task of Depth Estimation involves predicting the potential depth of each individual pixel of an RGB image. Depth cues are a vital process of human vision, and building a methodical cue perception is a potential machine learning problem that needs to be addressed. The earliest solutions for depth estimation were based on stereovision algorithms that utilized geometry to estimate the depth [1]. With the recent advancements in the field of Deep Learning, various neural network architectures are being applied to solve this problem [2]. This work also attempts to use different architectures to solve this problem and evaluate the performance of the models using different loss functions. We primarily look to investigate the effect of using loss functions in models that leverage transfer learning like Transformers and Densenet. Particularly, we would be building encoder-decoder models using Densenet[5] and DPT Depth Estimator [7], as these models have been shown to achieve state-of-the-art results in depth estimation. Also, the use of multiple loss terms can improve the inference capabilities of a model[9]. Based upon this hypothesis, this work aims to experiment with different loss functions and analyze the impact of each one of

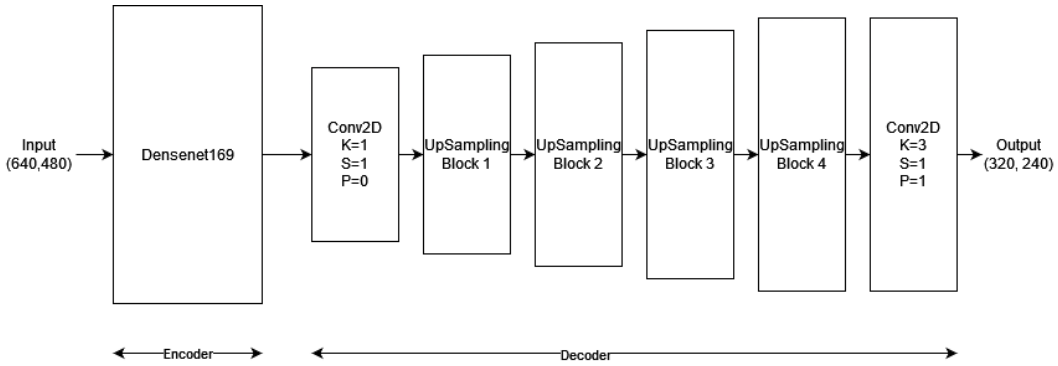

Figure 1: Unet model architecture

them on the model performance. By doing this, we hope to gain some insights into the key criterion for building better models for this problem domain.

2 Dataset

This study used the NYUv2[3] dataset for conducting the experiments. We use the 1449 labeled RGB images of size 640x480 and their corresponding depth frames. The entire dataset is split into training and testing subsets in the ratio of 80:20.

3 Methodology

This study aims to gauge the effectiveness of loss functions in governing the quality of the output of various neural network architectures. Firstly, we built a deep learning encoder-decoder model using Pytorch. We use Encoder-Decoder models for monocular depth estimation, as the prediction is the potential depth cue and Encoder-Decoder models have shown to be able to handle the requirements for the output representations. In this particular task, we would like to predict the depth cue of every individual pixel in the 640x480 input image.

For our first experiment, we built an Encoder-Decoder model with an architecture as described in Fig. 1. Alhashim et al. [3] have shown that the architecture described in Fig. 1 can perform well for the monocular Depth Estimation task, and thus we use this architecture to benchmark our results. Since it can be very computation-intensive to train both the encoder-decoder. We use the pretrained densenet-169 encoder[5], which is available in PyTorch for getting a representation of the image. The FC7 layer of the encoder provides a representation of the image of dimensions 1664x15x2. We construct a depth map of the input image by using the encoded representation of the image, by gradually upsampling the encoded representation, using residual connections from the Densenet Encoder. Huang et al. [5] have found that deep convolutional neural networks can get a huge performance boost for various image manipulation tasks, by appending the outputs of each encoder layer to the corresponding decoder layer input. This trick enables relatively short distances between the inputs and outputs, which they show can lead to better flow and gradients, feature reuse, and faster training. Considering the advantages provided by densenet architectures in addition to being a transfer learning technique, we utilized this architecture to iteratively upsample the image to a depth cue of dimensions 240x320. While upsampling, we concatenate the input from the corresponding encoder layer to the output obtained from forward propagation. This required our decoder model's upsampling block's specifications to mirror that of the densenet-169 encoder. The upsampling blocks are comprised of two convolution operations applied to the input and performing a concatenation of the convolutional outputs to increase the dimensions of the represented image. The model is trained for 50 epochs using an Adam optimizer with a learning rate of 0.00001. We used three loss functions namely- Mean Squared Error, Huber Loss, and SSIM (Structured Similarity Index measure)

For our second set of experiments, we utilized a Vision-induced transformer to generate depth masks. Ranftl et al. [7] have finetuned a Vision Induced Transformer for the task of depth mask prediction. This transformer was seen to operate in different pixel scales for depth estimation. For our task,

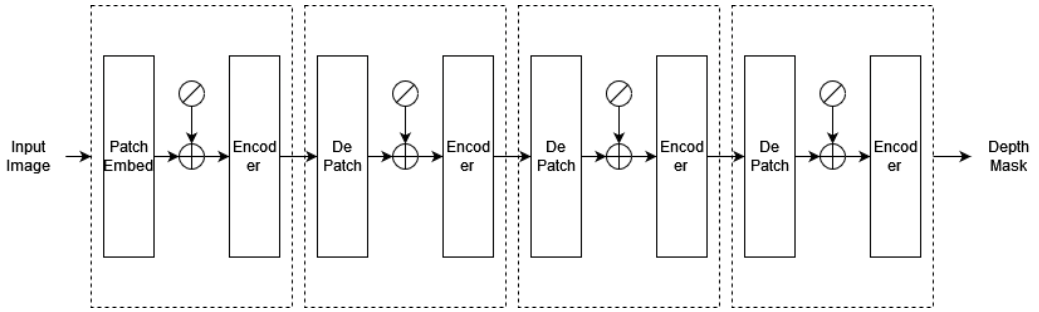

Figure 2: DPT architecture

however, we hope to fine-tune this model into our desired scale for the dataset at hand. This is again done, by utilizing the ViT as an encoder-decoder model, but instead of training a decoder from scratch, we utilized the decoder provided and interpolated the output of dimensions 384x384 to a profile of about 240x320. We further fine-tuned this model for 50 epochs using an Adam optimizer with a learning rate of 0.00001. We used three loss functions namely- Mean Squared Error, Huber Loss, and SSIM (Structured Similarity Index measure) to fine-tune the transformer performance.

For evaluating the model performance, we computed the number of exact pixel matches between the expected depth mask and the generated depth mask, and the average depth pixel match percentage gave a fairly good picture of the model's performance on the task of monocular depth estimation on the NYUv2[3] dataset.

4 Results and Discussion

We trained six unique models using the NYUv2 dataset, and the evaluation results are discussed in the Table [1]

Table 1: Table of accuracies

Model	MSE-Acc	Huber-Acc	SSIM-Acc
Densenet-169	39.50	42.57	41.43
DPT	89.34	90.03	86.52

The accuracy described in the table is described in equation 1, where a pixel-wise match of the normalized pixel output is done between the ground truth and the predicted depth cue by the model. We optimized both models using different loss functions and discerned average performance trends using the defined pixel-match accuracy. Firstly, we see that the transformers outpace the performance of Densenet Depth prediction model significantly. While the significance of loss functions was the primary focus of this project, this discrepancy in performance indicates that loss function is the primary criterion for improving the model performance in this domain, as it's clear that starting from optimal learning space is vital in transfer learning-based encoder-decoder models. Transformers have achieved state-of-the-art results in domains where self-attention and cross-attention can practically result in better knowledge representations. Thus the first inference we draw from our experimentations is that loss functions are not the primary criterion of optimization in monocular depth estimation.

While better transfer learning techniques are wise ways to vastly improve model performance, loss functions still play a major role in regularizing and fine-tuning the outputs obtained. We fine-tuned all our models using three loss functions namely Mean Squared Error, Huber Loss, and SSIM loss. On utilizing these loss functions, some interesting observations were made on the nature of the depth masks generated, which mirrored the design of the loss function to an extent.

The SSIM loss function as described in 4, minimizes the luminance, contrast, and structure discrepancies between the target and ground truth depth masks. We theorize that this loss function should generate depth masks with smooth contours retaining the major visual structural properties of the depth mask. Also, we theorize loss function would not provide the best pixel-to-pixel match accuracy since it concerns itself with generating a structural similar output. This proved to be largely true. Fig.

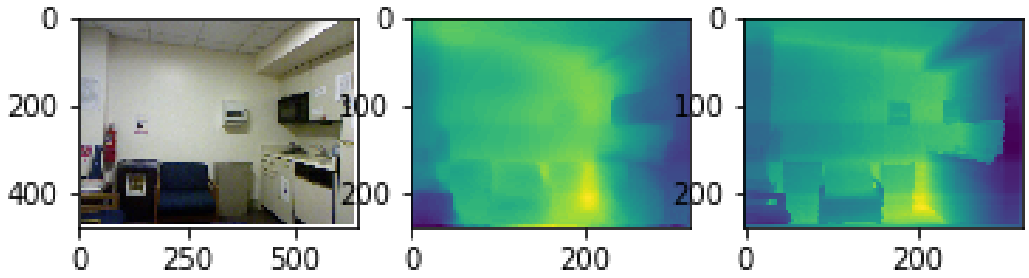

Figure 3: Input Image, b) Predicted Depth SSIM Transformer c) Actual Depth

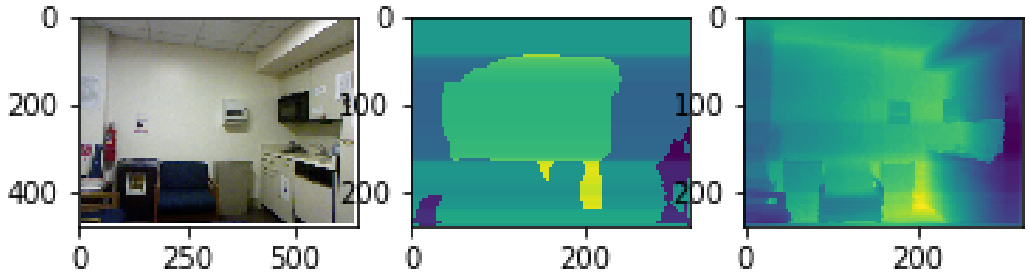

Figure 4: Input Image, b) Predicted Depth Huber Transformer c) Actual Depth U-net

3 visualizes the performance of a DPT fine-tuned with SSIM Loss on a test image, and it can be seen that the structural features of the input image are retained in the predicted depth mask.

However, on using Huber loss for the DPT transformer, the overall average pixel to pixel match performance was found to be highest, but on visualizing the results in Fig.4 we find the unregularized output to be visually less appealing than the regularized output from the SSIM-DPT. While the MSE loss function had promising results in fine-tuning the DPT performance, the depth masks themselves weren't any more notable than that of Huber. Thus, in essence, we had two sets of output patterns generated by utilizing various loss functions. A more accurate unregularized output from MSE and Huber, while the SSIM had a lesser percentage of exact pixel matches, but compensated for that by producing depth masks structurally similar to the input image.

We predominantly looked at the depth mask predictions of the transformers for evaluating the loss functions criterion, since the low quality of the Densenet-169 predictions made it hard to discern prediction patterns made evident by the use of loss functions. Fig. 5 shows the visualization obtained from Densenet. While the results could be promising if we fine-tuned the model further, we refrained from commenting on the nature of the depth masks generated at this point, given their poor performance for the task.

$$Accuracy(y, \hat{y}) = \frac{\sum_{i=1}^{D}(\lfloor y \rfloor = \lfloor \hat{y} \rfloor)}{D} \tag{1}$$

$$MSE(y, \hat{y}) = \sum_{i=1}^{D}(y_i - \hat{y}_i)^2 \tag{2}$$

$$Huber(y, \hat{y}) = \begin{cases} \frac{1}{2}(y - \hat{y})^2, & \text{for } |y - \hat{y}| \leq \delta, \\ \delta|y - \hat{y}| - \frac{1}{2}\delta^2 & \text{otherwise} \end{cases} \tag{3}$$

$$SSIM(y, \hat{y}) = f(l(y, \hat{y}), c(y, \hat{y}), s(y, \hat{y})) \tag{4}$$

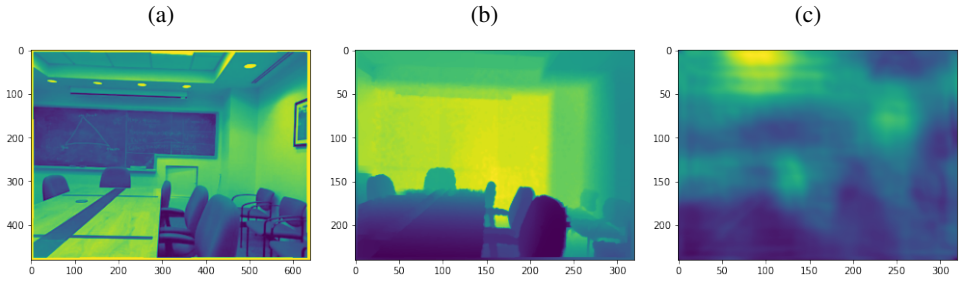

Figure 5: a) Input Image, b) Actual Depth, c) Predicted Depth

5 Conclusion and Future work

The mere generation of depth masks is not the ultimate objective of Monocular depth estimation, and it is not slated to be a supervised learning task, given its potential applications in areas like autonomous driving which rely predominantly rely on self-supervised learning techniques. Hence, we saw the potential applicability of transfer learning in solving problems on this task, with a major focus on Vision Induced Transformers and how various loss functions can result in the output masks having some predestined characteristics. While pixel match accuracy can be an optimal evaluation metric, unregularized depth outputs cannot be ideal. For instance, in the case of self-driving cars, having a better depth perception for humans or animals is more important than having a good depth perception of the sky. Our results from SSIM have been really promising, and we believe by carefully engineering loss functions, we can govern the generated output characteristics. For instance, we were able to smoothen the depth mask predictions with SSIM which can sanitize the predictions. Thus, in conclusion, our experiments reveal that utilizing handcrafted landscapes can regularize the output, but based on our results loss functions by themselves don't seem to be the vital piece of the puzzle for producing better results in Monocular Depth Estimation. We saw that Vision Induced Transformers beat Densenet-based models by a mile, and it has been shown that transformer-based models always have room to scale in performance with more data and bigger models. While loss functions have the potential to fine-tune performance, we failed to find convincing evidence for the same. However, we found that with proper engineering the loss functions can potentially enforce and regularize certain desired output characteristics. This, for instance, can be helpful in the real-life applications of monocular depth estimation, like self-driving cars where predicting the depth of shaped dynamic objects like other cars is way more important than predicting the depth of a static environment like that of the sky, which usually is shapeless.

Considering the implications of our observations, the future work of this particular work would be looking at regularizing the depth masks in the context of specific applications like self-driving cars by engineering relevant loss functions for desirable output characteristics in the purview of the application. Also, we believe that the pixel-match percentage as a validation metric is lacking in its ability to judge the goodness of output obtained, and hence a better metric could be devised, where the metric is more sensible than loss functions in standalone analysis while also encapsulating the model's performance on the task, and we believe this would be more sensible when done for the application, rather than for the problem domain.

6 Author Contributions

The author's contributions to the paper were as follows: Data Preprocessing: Himanshu Singhal; Data Visualizations: Harish Ravi; Model Building and Result analysis: Harish Ravi and Himanshu Singhal. Both the members contributed equally to the final report preparation and approving the final draft.

References

[1] D. Scharstein, R. Szeliski, and R. Zabih, “A taxonomy and evaluation of dense two-frame stereo correspondence algorithms,” in Proceedings IEEE Workshop on Stereo and Multi-Baseline Vision (SMBV 2001), pp. 131–140, 2001.

[2] D. Eigen, C. Puhrsch, and R. Fergus, “Depth map prediction from a single image using a multi-scale deep network,” in Advances in Neural Information Processing Systems (Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Q. Weinberger, eds.), vol. 27, Curran Associates, Inc., 2014.

[3] P. K. Nathan Silberman, Derek Hoiem and R. Fergus, “Indoor segmentation and support inference from rgbd images,” in ECCV, 2012.

[4] Alhashim, I., amp; Wonka, P. (2019, March 10). High quality monocular depth estimation via transfer learning. arXiv.org. Retrieved March 17, 2022, from https://arxiv.org/abs/1812.11941

[5] G. Huang, Z. Liu, L. Van Der Maaten and K. Q. Weinberger, "Densely Connected Convolutional Networks," 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 2261-2269, doi: 10.1109/CVPR.2017.243.

[6] R. Li et al., "An Effective Data Augmentation Strategy for CNN-Based Pest Localization and Recognition in the Field," in IEEE Access, vol. 7, pp. 160274-160283, 2019, doi: 10.1109/ACCESS.2019.2949852.

[7] Ranftl, R., Bochkovskiy, A., amp; Koltun, V. (2021, March 24). Vision Transformers for dense prediction. arXiv.org. Retrieved May 4, 2022, from https://arxiv.org/abs/2103.13413

[8] Zhou Wang, A. C. Bovik, H. R. Sheikh and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," in IEEE Transactions on Image Processing, vol. 13, no. 4, pp. 600-612, April 2004, doi: 10.1109/TIP.2003.819861.

[9] Jae-Han Lee and Chang-Su Kim. 2020. Multi-Loss Rebalancing Algorithm for Monocular Depth Estimation. In Proceedings of the European Conference on Computer Vision (ECCV)

[10] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., amp; Polosukhin, I. (2017, December 6). Attention is all you need. arXiv.org. Retrieved May 5, 2022, from https://arxiv.org/abs/1706.03762