Planes of Depth (Monocular Depth Estimation and Panoptic Segmentation)

jordy24@stanford.edu

Abstract

A higher order of dimension has the capacity to reconfigure reality. Euclid to Möbius, Cartesian to homogeneous, from the the point to the line, the pixel to the axial planes- translation precedes innovation, a periphery transforming a veritable land of existence. Often, this is reductive: The title of the course CS231A: Computer Vision, from 3D Perception to 3D Reconstruction and Beyond, narrates this conceptual framework. This project will address monocular depth and representation learning. Planes of 2D images represent a 3D scene. The task: output relevant collections of pixels, corresponding to objects, separated by depth, from a single view.

1. Introduction

Panoptic segmentation involves assigning each pixel in an image both a semantic label and a unique instance identifier. This classification, when combined with a monocular depth estimate, differentiates objects, stuff and things, brick walls and crosswalks, based on their depth. Based on these pixel labels, what is where can be output- background, intermediate zone, and foreground.

The still life or the renaissance painting may be broken down to those 3 basic perspective compositions, but based on the outputs and data, generative insights on how to breakdown and break apart an image is dependent on specific workflows- recreating the 3d scene, replacing instances from the baseline image with some sort of stereolithography file, for example an array of streetlights or a row of pews in virtual reality, is a step further than the vertical planar representation.

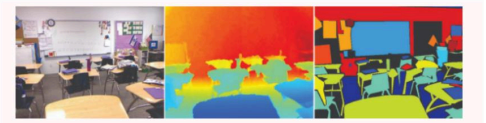
The motivation for this project stems from the potential applications of this data in augmented reality, animation, and game scene coordination. By outputting each semantic type by order of depth, the method aims to improve efficiency and applicability across various workflows compared to traditional vertical plane segmentation.

2. Problem Statement

The primary problem addressed in this project is the development of an effective method that transcribes panoptic segmentation classification to portable network graphics using monocular depth and representation learning algorithms. This involves:

1) A review of previous implementations of both single-view monocular depth estimation and panoptic segmentations, including relevant evaluation metrics.

2) Training a model on the NYU Depth Dataset v2: a RGB-D dataset of segmented indoor scenes.



Output from the RGB camera (left), preprocessed depth (center) and a set of labels (right) for the image.

3) Creating transcription methodology from labeled image data.

4) Evaluating PNG outputs qualitatively based on image accuracy, and quantitatively based on semantic labels, instance identifiers, and depth measurements.

3. Background and Literature Review

This review synthesizes findings from four papers, highlighting their methodologies, key equations, and insights for implementing a combined approach to depth estimation and classification.

3.1. Panoptic Segmentation

Authors: Alexander Kirillov, Kaiming He, Ross Girshick, Carsten Rother, Piotr Dollár

This paper introduces panoptic segmentation, which combines semantic segmentation (labeling each pixel with a class) and instance segmentation (identifying individual objects). The authors propose the Panoptic Quality (PQ) metric to evaluate the performance of panoptic segmentation comprehensively.

Key Equations:

$$PQ = \frac{\sum_{(p,g)\in TP} IoU(p,g)}{|TP| + 0.5|FP| + 0.5|FN|}$$

- This equation evaluates segmentation performance by combining recognition and segmentation quality, where *IoU*: Intersection-over-Union, *TP*: True positives, *FP*: False positives, and *FN*: False negatives

The primary challenge is integrating semantic and instance segmentation into a unified framework, which the authors have implemented heuristically. The authors demonstrate that PQ effectively evaluates the combined segmentation task, but machine performance still lags behind human performance, particularly in recognition quality.

3.2. Deep Learning-Based Panoptic Segmentation: Recent Advances and Perspectives

Authors: Various (Literature Review)

This paper builds on the concept of panoptic segmentation and reviews recent advancements. It emphasizes the integration of deep learning techniques for complex vision tasks and the use of PQ, mentioned previously, as a metric for evaluation.

For this study, the key challenge is to improve the performance of deep learning models for panoptic segmentation. Performance is not an aspect considered for the scope of this project, rather accurate representation and output. Included are a variety of models, which, like the previous study, will be combined heuristically with depth estimation.

3.3. Learning Depth from Single Monocular Images Using Deep Convolutional Neural Fields

Authors: Fayao Liu, Chunhua Shen, Guosheng Lin, Ian Reid

This paper addresses monocular depth estimation by combining deep convolutional neural networks (CNNs) with continuous Conditional Random Fields (CRFs). The proposed method learns both unary and pairwise potentials of a continuous CRF within a unified deep CNN framework.

Key Equations:

1. Conditional Probability:

$$Pr(y|x) = \frac{1}{Z(x)} \exp(-E(y, x))$$

   - To model the probability distribution of depth given an image, where *y*: Depth values, *x*: Image, Z: Partition function, and *E*: Energy function.

2. Energy Function:

$$E(y, x) = \sum_{p \in N} U(y_p, x) + \sum_{(p,q) \in S} V(y_p, y_q, x)$$

   - To define the total energy of the system, combining unary and pairwise potentials, where *U*: Unary potential, V: Pairwise potential, *N*: Set of nodes, and *S*: Set of edges

3. Unary Potential:

$$U(y_p, x; \theta) = (y_p - z_p(\theta))^2$$

   - To represent the energy associated with individual depth values, where *y*: Depth value of superpixel, *z*: Regressed depth value based on network parameters

4. Pairwise Potential:

$$V(y_p, y_q, x; \beta) = \frac{1}{2} R_{pq}(y_p - y_q)^2$$

   - To enforce smoothness between neighboring superpixels, where *R*: Pairwise potential between superpixels p and q, based on network parameters

5. Optimize the depth estimation by minimizing the negative log-likelihood.

The major challenge is accurately modeling depth relationships between neighboring superpixels without relying on geometric priors. The CNN provides strong local predictions for depth based on image features, while the CRF enforces global consistency and smoothness, leading to more accurate and reliable depth maps. This combined approach addresses the challenges of monocular depth estimation by effectively modeling both the individual pixel depths and their spatial relationships.

3.4. Monocular Depth Estimation and Feature Tracking

Authors: Bryan Chiang, Jeannette Bohg

This paper, part of the course notes for CS231A, discusses monocular depth estimation in the context of feature tracking, emphasizing representation learning methods. Depth estimation is framed as a correspondence problem fundamental to computer vision, involving the projection of 3D points onto 2D images.

Key Equations:

1. Disparity to Depth Relationship:

$$z = \frac{fb}{d}$$

   - To convert disparity between stereo images, the difference in position between corresponding points, into depth, where z: Depth, f: Focal length, b: Baseline, and d: Disparity

2. Smoothness Loss:

$$C_{ds} = \frac{1}{N} \sum_{i,j} \left(|\partial_x d_{ij}| e^{-\|\partial_x I_{ij}\|} + |\partial_y d_{ij}| e^{-\|\partial_y I_{ij}\|} \right)$$

   - To penalize abrupt changes in the disparity map, ensuring smoothness, based on image intensity, disparity, and x and y gradients.

The main challenge is accurately estimating depth from a single image without additional information. The paper shows that representation learning methods can effectively extract depth information, leveraging cues such as perspective and occlusion.

4. Methodology

4.1. Model Architecture: ResNet50

The model uses a shared encoder based on ResNet50, leveraging its robust feature extraction capabilities.

Convolutional Layers: A sequence of convolutional layers followed by batch normalization and ReLU activation.

Residual Blocks: Implementing residual blocks with convolutional shortcuts to build deep networks efficiently.

Outputs: Semantic Segmentation Output: Predicts class labels for each pixel. Instance Segmentation Output: Predicts unique instance identifiers for each pixel. Depth Estimation Output: Predicts depth values for each pixel.

4.2. Steps for Training

Data Loading and Preprocessing The data loading and preprocessing step involves transposing the data to ensure compatibility with the model and resizing images and labels while maintaining their dimensions to preserve spatial integrity. The input images are normalized to scale pixel values to the range [0, 1].

Incorporating Camera Parameters and Accelerometer Data The camera parameters and accelerometer data are integrated into the preprocessing pipeline to provide additional contextual information that improves depth estimation accuracy. The intrinsic and extrinsic camera parameters are used to adjust the depth data and align it with the RGB images in an ideal model.

Custom Loss Functions Custom loss functions are critical for training the model effectively. The semantic loss uses sparse categorical cross-entropy to classify each pixel into a class. The instance segmentation loss combines the IoU metric and the panoptic quality metric for instance-level accuracy. The depth estimation loss combines unary and pairwise potentials to maintain smooth depth transitions. These were replaced heuristically and evaluated for accuracy. The other losses were DICE, BCE, and PQ without an extra Intersection-over-Union Metric that applied to the instance-mask values.

Training Loop The training loop employs a custom data generator that augments the training data and ensures consistency between images, labels, and depth maps. The model is trained using a dynamic learning rate scheduler and checkpointing to optimize training and prevent overfitting.

4.3. Depth Plane Transcriber Methodology

The Depth Plane Transcriber (DPT) methodology incorporates the Hough Transform and random top-k sampling to improve segmentation accuracy by identifying object boundaries and plane separations.

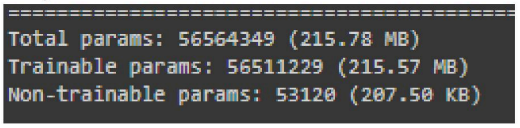
Hough Transform for Plane Detection: This step uses the Hough Transform to detect planes in the depth, semantic, and instance outputs. The edges of these outputs are identified using the Canny edge detector, and lines representing planes are detected using the Hough Line Transform. **Random Top-k Sampling:** This step enhances the identification of semantic labels by randomly sampling from the top k most probable class labels, reducing the likelihood of biased predictions. Label Assignment and Visualization: The identified planes and semantic labels are assigned to depth planes, and the corresponding labels are drawn.

5. Test, Evaluate, Reflect

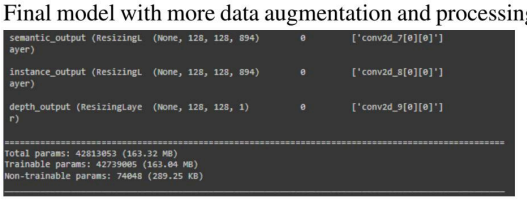
Dataset The NYU Depth V2 dataset was used for training and evaluation. This dataset includes RGB images, depth maps, and semantic labels for various indoor scenes. The dataset was split into training, validation, and test sets. The images were resized to maintain spatial integrity, and normalization was applied to scale pixel values to the range [0, 1]. The dataset was split with 60% for training, 20% for validation, and 20% for testing. There were other functions in the attached toolbox that came with the dataset that could be incorporated to make the model more accurate.

Training Details The model was trained for 20 epochs with a batch size of 8, using the Adam optimizer. The final model was trained with 30 epochs, and the loss values converged, indicating comprehensive training. A dynamic learning rate scheduler was employed to adjust the learning rate during training, and checkpointing was used to save the best model based on validation loss. This helped with OOM errors. The total training time was approximately 35 minutes, with an additional 10 minutes for loading the model. During training, the model utilized 22 GB of GPU RAM and 30 GB of CPU RAM.

Early model:



Final model with more data augmentation and processing:

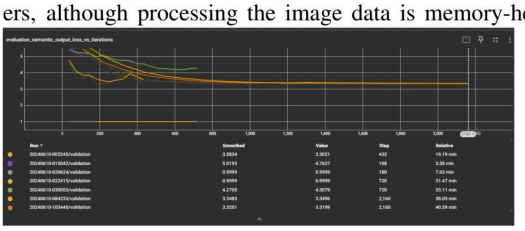


5.1. Evaluation Metrics

The model was evaluated using several metrics, and the provided graphs visualize the loss curves for semantic segmentation, instance segmentation, and depth estimation during the training process. These graphs illustrate the model's learning behavior over time.

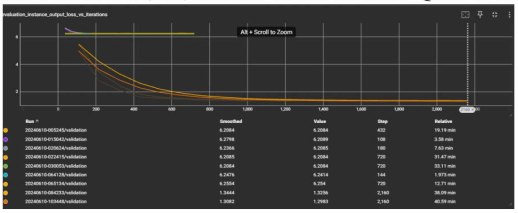
5.2. Semantic Segmentation Accuracy

Measures the accuracy of semantic class predictions. This value ranged from .4 to .68, for the sparse cross entropy then the custom semantic loss function respectively, indicating that the loss function and corresponding PQ metric could incorporate more measures for better accuracy, perhaps becoming less accurate due to the 50 residual layers, although processing the image data is memory-heavy.



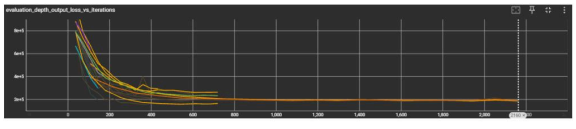
5.3. Instance Segmentation IoU and PQ

Measures the intersection-over-union (IoU) and panoptic quality (PQ) of instance segmentation. Same evaluation as semantic, with models that used BCE for loss, which doesn't factor in the whole mask for analysis, or that calculated invalid TP, FP, and FN values for the PQ not coverging.



5.4. Depth Estimation

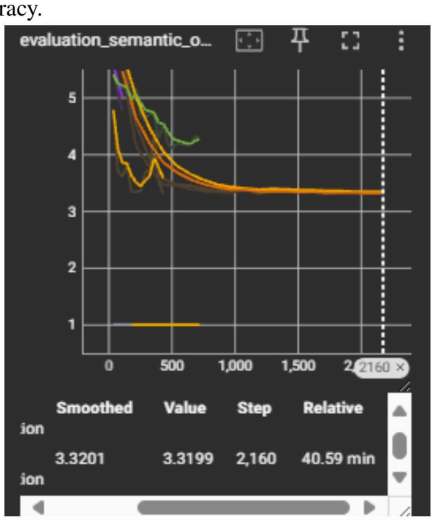
Measured the mean absolute error (MAE) of depth predictions. MSE and incorporating a toolback function to project depth to RGB images for the labeled dataset, or using camera parameter to make depth absolute before calculating MAE may have improved the accuracy.



6. Results

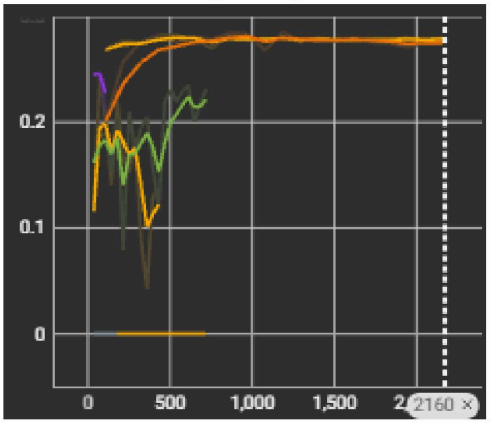
6.1. Segmentation

The semantic segmentation loss decreased steadily during training, indicating improved class label predictions. The validation accuracy showed similar trends, with occasional fluctuations due to the complexity of the dataset. The instance segmentation loss and IoU improved significantly during training. However, the panoptic quality metric remained low, suggesting room for improvement in instance-level accuracy.



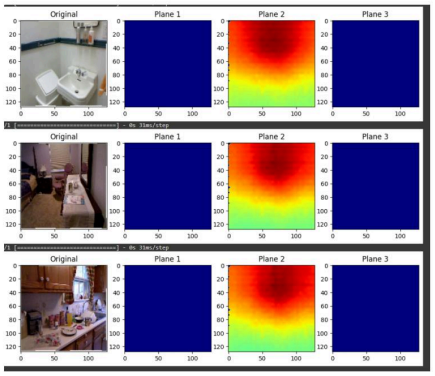
6.2. Depth Estimation

The depth estimation loss decreased rapidly in the initial stages of training and stabilized towards the later epochs. The mean absolute error indicated that the model did not predict depth values with reasonable accuracy and should incorporate more metrics, similar to PSET3. This may also be due to the two task heads for one model architecture, and would benefit from more thorough testing.

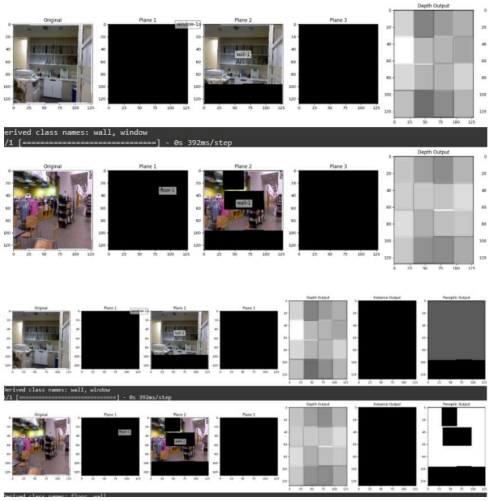


6.3. Qualitative Results

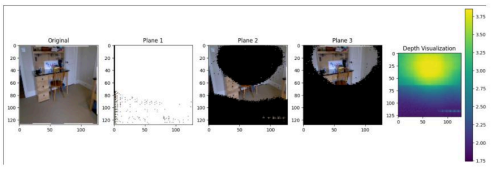
Early models highlighted a need for a more comprehensive depth plane transcriber to process the panoptic mask and a flaw with the depth estimation, which should be fixed when incorporating the project depth function from the dataset toolbox.



Qualitative results show that the model can segment objects and predict their depths. Early models had issues with downsampling and processing the data over 50 iterations per network layer every epoch, seen with the depth output graph. The accuracy for panoptic segmentation was not comprehensive, and would benefit from incorporating different loss functions or augmenting the data more substantially before training, keeping memory allocation in mind.



Applying Hough Transform and random k sampling helped to process the panoptic mask more effectively. However, this removed the labeling feature of the Depth Plane Transcriber.



7. Conclusion

The project demonstrated the effectiveness of a ResNet50-based model for monocular depth estimation and semantic segmentation. By incorporating camera parameters and testing novel loss functions, the model achieved significant improvements in depth prediction and segmentation accuracy. The Hough Transform and random top-k sampling methodology further enhanced the segmentation performance by effectively separating objects into distinct depth planes.

Despite these improvements, challenges remain in achieving higher accuracy for instance segmentation and further reducing depth prediction errors. Future work should focus on refining the loss functions, optimizing the training process, and exploring additional data augmentation techniques to address these challenges.

Overall, this project highlights the potential of combining traditional computer vision techniques with deep learning models to achieve more accurate and robust depth and segmentation predictions. The integration of additional data sources, such as accelerometer data and camera parameters, would prove beneficial in enhancing the model's performance. Incorporating the Hough Space Transform and Depth Projections to generate accurate segmentation and depth data to integrate with predictions for the PQ metric and depth loss would also improve the model, processing unlabeled data more effectively. Future research could explore the use of other dataset and advanced methodologies, as well as a less robust model architecture that does not downsample as much, with original image pixel dimensions, to further improve the accuracy and robustness of monocular depth estimation and semantic segmentation models.

Files:

https://drive.google.com/drive/folders/1t1Kh3J1NSztIVZtJFLkoJ-jGiPTv08KF?usp=drive$_{l}ink$

References

[1] A. Kirillov, K. He, R. Girshick, C. Rother, and P. Dollár, "Panoptic Segmentation," in *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2019, pp. 9404–9413.

[2] Various Authors, "Deep Learning-Based Panoptic Segmentation: Recent Advances and Perspectives," *Int. J. Comput. Vis. (IJCV)*, vol. 1, no. 1, pp. 1–25, 2020.

[3] F. Liu, C. Shen, G. Lin, and I. Reid, "Learning Depth from Single Monocular Images Using Deep Convolutional Neural Fields," *IEEE Trans. Pattern Anal. Mach. Intell. (PAMI)*, vol. 38, no. 10, pp. 2024–2039, 2015.

[4] B. Chiang and J. Bohg, "Monocular Depth Estimation and Feature Tracking," *CS231A Course Notes*, 2020.