Development and Optimization of a Monocular Depth Estimation Models

I. Leela Sai Abhiram
Contributors: Dr. Sandeep Paul

June 19, 2024

Abstract

This project focuses on developing and optimizing monocular depth estimation models. Initially, foundational concepts in computer vision and OpenCV were explored for one week. Pretrained models like MiDaS and DepthAnyThing were tested for monocular depth estimation. Following this, a custom dataset was created using a RealSense D455 camera, and a simple UNet model was developed and trained, showing promising but preliminary results.

Collaboration with Dr. Sandeep Paul provided access different models and DIODE and NYU Depth datasets. Due to computational constraints, focus shifted to the DIODE dataset (approx. 3GB). Various optimization strategies were explored, including integrating Inception blocks into multiple models (UNet, Attention-UNet, DWT-UNet, etc.) and implementing pruning techniques to improve efficiency. Evaluation metrics such as FLOPs, MACs, MAE, RMSE, and threshold measures were used to assess model performance, detailed in an Excel report comparing standard and Inception block versions.

1 Introduction

Monocular depth estimation is a crucial task in the field of computer vision, involving the prediction of depth information from a single RGB image. Traditional methods for depth estimation typically rely on expensive equipment such as LiDAR sensors or stereo cameras, which provide high accuracy but at a significant cost. However, many applications, such as augmented reality, robotics, and autonomous driving, do not require such precise measurements and can benefit from more cost-effective solutions.

Monocular depth estimation offers a viable alternative by using a single camera to estimate depth, making it accessible and practical for a broader range of applications. Recent advancements in deep learning have significantly improved the accuracy and reliability of monocular depth estimation models, making them a compelling choice for various real-world tasks. The primary objectives of this project are:

- To gain a comprehensive understanding of computer vision (CV) and monocular depth estimation techniques.
- To develop and train a monocular depth estimation model.
- To optimize the performance of different models using different methods.

2 Methodology

2.1 Approaches and Techniques

The project followed a systematic approach to develop and optimize the monocular depth estimation models. Here are the steps and techniques used:

- **Initial Learning Phase:** Spent one week learning the fundamentals of computer vision and familiarizing myself with OpenCV.
- **Exploring Pretrained Models:** Tested pretrained models such as MiDaS and DepthAnyThing for monocular depth estimation.
- **Custom Dataset Creation:** Created a custom dataset using the RealSense D455 camera.
- **Model Development:** Developed a simple UNet model and trained it using the custom dataset.
- **Collaboration and Data Acquisition:** Collaborated with Dr. Sandeep Paul, who provided access to his thesis on monocular depth estimation including different models and datasets like DIODE (Dense Indoor and Outdoor Depth) and NYU Depth datasets. Due to computational resource limitations, focused on the DIODE dataset (approximately 3GB).
- **Model Optimization:** Researched and integrated Inception blocks into various models: UNet, Attention-UNet, DWT-UNet, Attention-DWT-UNet, UNet++, DWT-UNet++, and DWT-Attention-UNet++. Investigated pruning techniques and implemented changes to nbfilter sizes.
- **Evaluation and Analysis:** Evaluated models using metrics such as FLOPs, MACs, MAE, RMSE, and various threshold measures. Compiled results into a comprehensive Excel sheet for both standard and Inception block versions of each model.

2.2 Tools and Technologies

- **Programming Languages:** Python
- **Development Environments:** Google Colab, Visual Studio Code (VS Code)

- **Libraries and Frameworks:** OpenCV, TensorFlow, Keras, PyTorch etc..
- **Hardware:** RealSense D455 camera , Laptop
- **Datasets:** DIODE, NYU Depth Dataset

2.3 Data Collection and Analysis

- **Custom Dataset:** Collected using RealSense D455 camera.
- **Training and Evaluation:** Models were trained on the DIODE dataset, and performance metrics were calculated to assess the efficiency of each model.

3 Observation and Results

The key findings and outcomes of the project are summarized below:

- **Model Performance:** The initial UNet model trained on the custom dataset showed promising results but required further tuning.
- **Inception Block Integration:** Integration of Inception blocks into various models improved their performance.
- **Pruning Techniques:** Successfully implemented changes to nbfilter sizes, enhancing model efficiency.
- **Evaluation Metrics:** Models were evaluated using metrics such as FLOPs, MACs, MAE, RMSE, and thresholds (1.25, 1.25^2, 1.25^3).
- **Comparative Analysis:** Compiled an Excel sheet comparing the performance of standard and Inception block versions of each model.
- **Model Efficiency:**The NESTNET(UNet++) INCEPTION model has the lowest MAE and RMSE values, indicating high accuracy. However, it requires significantly more computational resources, as reflected in its high FLOPs and MACs.
- **Pruning Effectiveness:**Pruned models, particularly NESTNET(UNet++) STANDARD, show a good balance between reduced computational cost and performance, with only a slight increase in error metrics compared to their non-pruned counterparts.
- **Inception vs. Standard:**Models with the INCEPTION module are having better accuracy but also higher computational demands compared to the STANDARD module models.

- **Best Overall Performance:**Considering both accuracy and computational efficiency, the NESTNET(UNet++) Pruned INCEPTION model stands out with relatively low error rates and moderate computational requirements.

4 Discussion

4.1 Comparison with Initial Objectives

The project met its objectives by successfully developing a basic monocular depth estimation model and optimizing some models. The integration of Inception blocks and pruning techniques demonstrated improvements in model performance.

4.2 Unexpected Outcomes and Challenges

- Faced so many compatibility issues with different libraries and many other errors in the initial days for work.
- Computational resource limitations hindered the training of models on the NYU Depth dataset.
- Integrating Inception blocks required careful tuning to balance model complexity and performance.

4.3 Contribution to the Field

The optimized models provide a cost-effective solution for depth estimation, making the technology more accessible. The comparative analysis offers valuable insights into the benefits of integrating Inception blocks and pruning the depth estimation models.

5 Conclusion

5.1 Main Takeaways

Monocular depth estimation is a viable alternative to expensive depth estimation methods. Optimizing models with Inception blocks and pruning techniques can significantly enhance performance.

5.2 Recommendations for Future Work

- Explore additional optimization techniques such as advanced pruning methods and quantization.
- Investigate the use of transfer learning to improve model performance with limited computational resources.

- Expand the evaluation to include real-world applications and scenarios.

6 References

- Creating a simple UNet model : Link
- Understanding Inception: Simplifying the Network Architecture. Available at: Medium Article on Inception Architecture

7 Appendices

7.1 Github Link for code :

https://github.com/leela4821u/Monocular-Depth-Estimation.git

7.2 Model Optimization

Researched and integrated Inception blocks into various models: UNet, Attention-UNet, DWT-UNet, Attention-DWT-UNet, UNet++, DWT-UNet++, and DWT-Attention-UNet++. Investigated pruning techniques and implemented changes to nbfilter sizes. The architecture of Inception blocks is illustrated below:

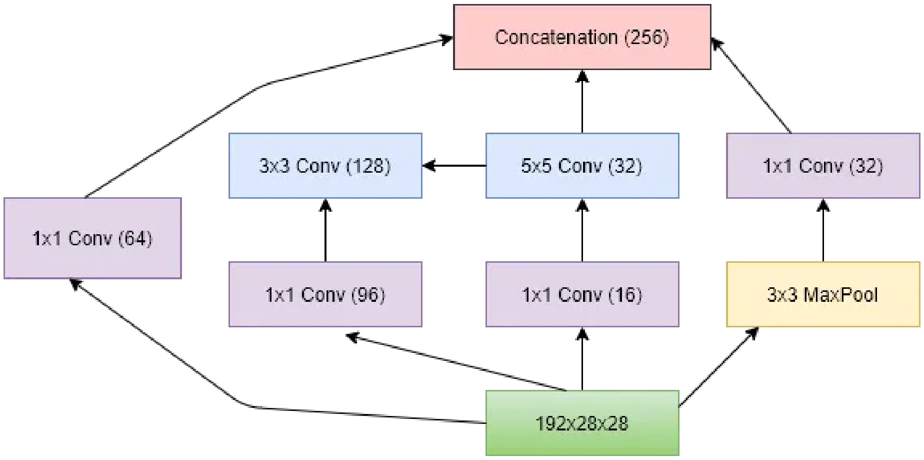

Figure 1: Inception Block Architecture