Thesis No: CSER-24-58

Depth Map Generation Using Monocular Depth Estimation of Deep Learning

By

Dabbrata Das

Roll: 1807109

Department of Computer Science and Engineering

Khulna University of Engineering & Technology

Khulna 9203, Bangladesh

February, 2024

Depth Map Generation Using Monocular Depth Estimation of Deep Learning

By

Dabbrata Das
Roll: 1807109

A thesis submitted in partial fulfillment of the requirements for the degree of
“Bachelor of Science in Computer Science and Engineering”

Supervisor:

Dr. Pintu Chandra Shill
Professor
Department of Computer Science and Engineering
Khulna University of Engineering & Technology

Signature

Department of Computer Science and Engineering
Khulna University of Engineering & Technology
Khulna 9203, Bangladesh
February, 2024

Acknowledgements

We would like to express my deepest gratitude to the Almighty Creator, the source of all knowledge and wisdom, whose divine guidance and blessings have illuminated our path throughout the journey of this thesis. We are immensely thankful for the strength and resilience that faith in the Creator has instilled in us during the challenging phases of this research. Your divine grace has been our constant source of inspiration and motivation, helping us persevere in the face of obstacles. We are also grateful for the unwavering support and encouragement of our family, friends, and mentors who have been instrumental in shaping this thesis. Their belief in us and their prayers have been invaluable.

Authors

Abstract

Monocular depth estimation is one of the most important tasks in computer vision and the estimation based on deep learning plays a significant role in various fields such as autonomous driving, navigation system with tracking, augmented reality, etc. There are some existing solutions of depth estimation which produced blurry, low-resolution and lower accurate depth map with higher cost. In this report, it presents a convolution neural network (CNN) for computing depth map with the help of transfer learning approaches where a single RGB image is given. Here U-Net architecture is followed which basically an encoder-decoder architecture. In encoder section some high performing pretrained models are applied which can extract more accurate features from the image along with augmentation and lead to more accurate result. A very simple decoder is used which decoded required depth map with lower number of parameters and generates more detailed depth map. Besides tuning the loss function increases more accuracy of the depth map. There are different types of datasets and models which are mostly used for the improvement among them NYU-DepthV2 dataset is used to estimate the depth map. Using the specific dataset with transfer learning approaches, the U-Net architecture will be able to generate a more detailed and more accurate depth map with better resolution. Finally, the improvements and the challenges which will arise in monocular depth estimation from a single RGB image will also be highlighted here.

Contents

List of Tables

List of Figures

CHAPTER I

Introduction

1.1 Introduction

Monocular depth estimation is one of the most advanced tasks in the field of computer vision. Depth estimation means estimate the depth of the objects or find the relative distance between two objects from an image. And depth map is an image which contains depth information of the image and is generated using depth estimation.There are different kinds of deep learning models like U-Net [1], AlexNet [2], ResNet [3], etc and functions such as evaluation functions, loss functions, etc. and datasets such as DIODE, NYU Depth V2, KITTI, etc. used as the basic tools for the implementation of the project.In this report, Convolution Neural Network is used for the task of depth estimation followed by U-Net architecture where U-Net architecture is basically an endcoder-decoder based architecture.Besides some high performing pretrained models are used with this architecture as encoder part which provide high quality depth map with better accuracy than normal encoder.Nyu Depth V2 data is used which is basically consisted with indoor RGB images with Ground Truth depth map provides more accurate result after training. Resnet50, Densenet169, VGG19 and Inception Resnet V2 are high performing pre-trained networks that are originally designed for image classification as the deep features encoder.

1.2 Background

Obtaining a depth map accurately is too much hard task which requires heavy costs and a lot of resources with times. Besides it is also possible to calculate the accurate depth map through sensors like LiDaR. But it is too much costly and time-consuming. So monocular depth estimation is highly necessary to meet up the problem where a single RGB image is needed as input and provide a depth map with good accuracy through well-trained depth estimation model.

1.3 Objectives

The main goal is to generate an accurate depth map with higher visual quality using monocular depth estimation approach. Besides, the other objectives are:

i. To learn about depth estimation based on deep learning.

ii. To learn about models and relatable datasets with supervised learning.

iii. To implement pre-processing steps on the dataset which consists both RGB image and ground truth depth map.

iv. To know about deep learning network and modify it with other high performing pre-trained network layers.

v. To implement U-Net encoder-decoder architecture via transfer learning.

vi. To increase the accuracy of output of the depth map with high visual quality using lower number of parameters.

vii. Tuning the loss function to improve the accuracy.

viii. Compare the predicted depth map with ground truth depth map using a customized loss function.

1.4 Scopes

There are some larger and high resolution datasets and the customization of the model which are used to generate depth map using depth estimation of deep learning.

1.4.1 Deep Learning Framework

There are some deep learning framework which are used to implement it.Basically these frameworks are TensorFlow, PyTorch, Keras etc. are the platform based on python language which are more efficient to implement the model than other languages.

1.4.2 Convolutional Neural Network(CNN)

Most monocular depth estimation models are built using CNNs due to their ability to learn hierarchical features from images.We can use various network architectures like U-Net [1], ResNet [3], MiDaS or custom-designed networks.

1.4.3 U-Net

It is actually a CNN based model based on encoder-decoder architecture as well as mostly used Network architecture to segment an object from a image.Using convolution layer and pooling layer,it downsamples the images to extract feature map until bottleneck layer and again upsampling the image by concatenating with previous featured image through skip connections generate U shape architecture. So the U-Net [1] is so much effective to make a efficient model. Various kinds of larger ,annotated and high resolution dataset and high performing pretrained models are used with this architecture for the implementation.

1.4.4 VGG19

VGG-19 [4] is a convolutional neural network trained on more than a million images from the ImageNet database. The network is 19 layers deep and can classify images into 1000 object categories. As a result, the network has learned rich feature representations for a wide range of images. This can be used as a pretrained model as an encoder along with U-Net architecture.

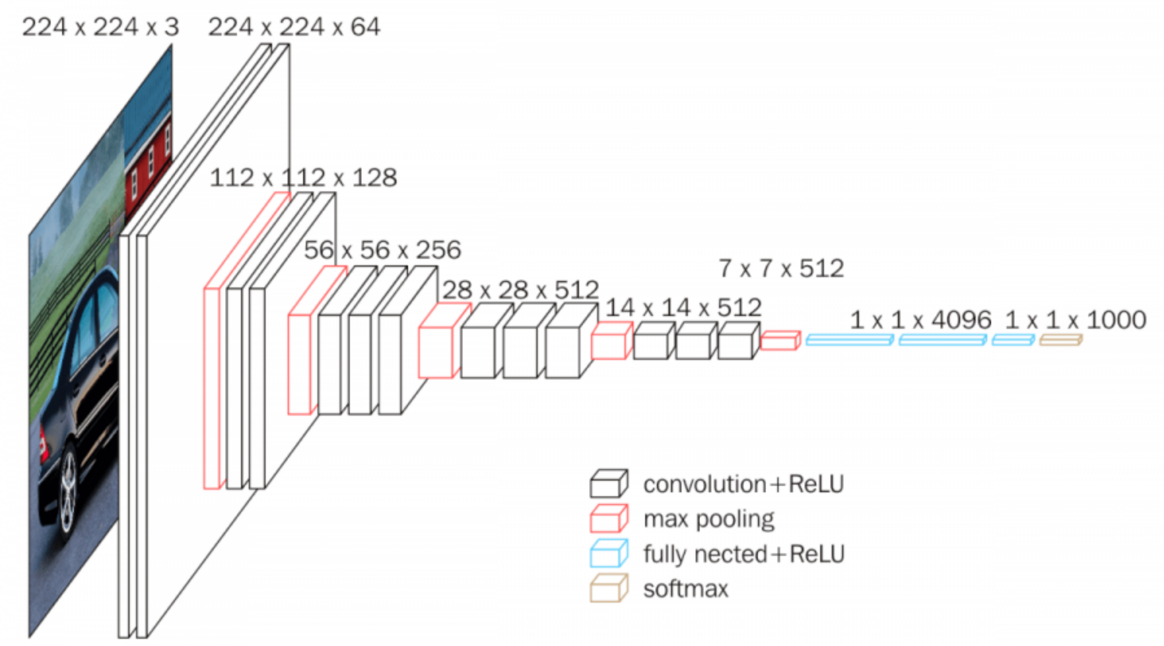

Figure 1.1: VGGNet architecture.

1.4.5 ResNet50

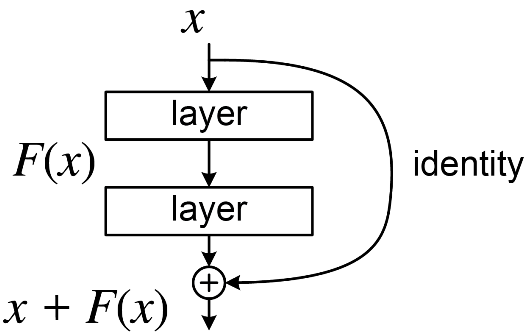

Figure 1.2: A building block of Residual Learning.

ResNet-50 is a convolutional neural network with 50 layers.It is trained on more than a million images from the ImageNet database and can be used as pre-trained model with encoder decoder architecture. Here, in each residual block the output is added with initial input through skip connection which is called the main principle of the residual network. So, it can address the vanishing gradient problem and minimize it.

1.4.6 DenseNet169

DenseNet-169 [5] is a convolutional neural network (CNN) architecture with 169 layers deep. The DenseNet architecture is known for its dense connectivity pattern, where each layer is connected to every other layer in a feedforward fashion. Here in dense block,it consists with multiple convolution layers and also no downsampling occured after each convolution. So the feature maps remain same size after each convolution which will be concatenated with next feature map. Here the concatenation is occured by using skip connections. Because each layer of the dense block is connected with other convolution layers by the skip connection. Besides in tansition layer,it consists with convolution and pooling layers and pooling reduces feature map sizes to reduce parameters to reach the bottleneck . This allows the network to learn both low-level and high-level features in a single pass and helps to combat the vanishing gradient problem. The dense connectivity and feature reuse in DenseNet create a more direct and efficient path for information and gradients to flow through the network during both forward and backward passes. This helps to alleviate the vanishing gradient problem.

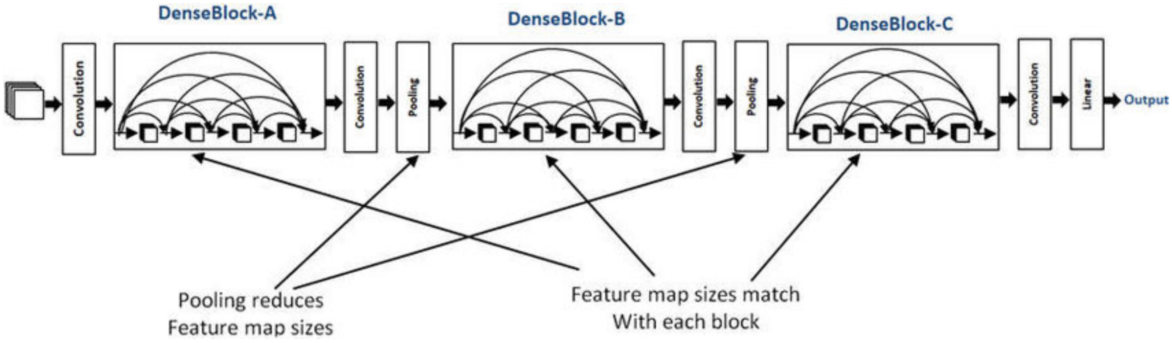

Figure 1.3: Densenet architecture with three dense blocks.

1.4.7 Inception Resnet V2

Inception-ResNet-v2 [6], a convolutional neural network, was trained with more than a million images from the ImageNet database. The network, which has 164 layers, can identify images of objects into 1000 different categories. As a result, the network has trained an enormous variety of feature representations that are rich in images. The input image has a resolution of 299 by 299 pixels, and the network produces a list of estimated class probabilities as its output. By incorporating residual connections, the Inception-ResNet-v2 convolutional neural architecture builds upon the Inception family of architectures.

In the architecture of Inception resnet V2, there are three types of inception blocks which are Inception Resnet Block A, Inception Resnet Block B, Inception Resnet Block C, two types of Reduction Block A and B, Stem Block and Average Pooling layer. Using dropout with keeping value 0.8 to reduce the overfitting issue. Here the Inception Blocks are occured multiple times to form a full layer architecture.

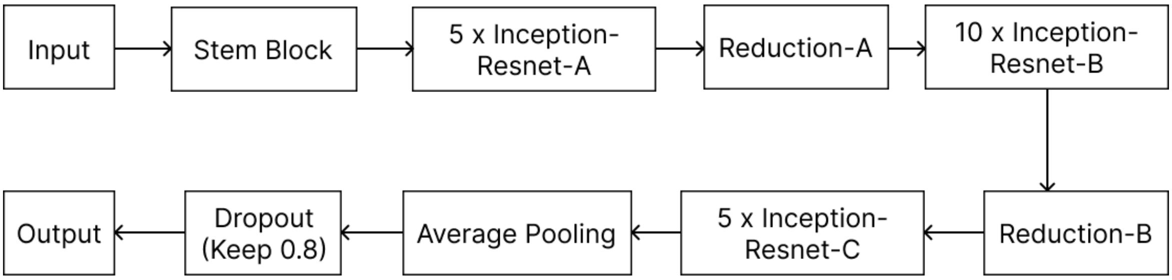

Figure 1.4: The basic architecture of Inception Resnet V2.

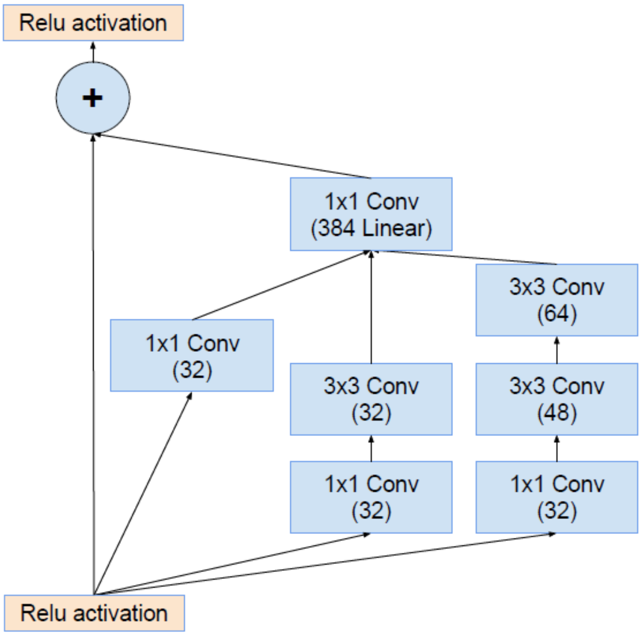

Figure 1.5: Block architectue of Inception Resnet A.

Inception Resnet A is one of the fundamental components of the Inception-ResNet v2 architecture. The goal of this block is to efficiently capture multi-scale properties in order to solve the vanishing gradient problem. The Initial-ResNet-A block usually consists of the following components:

Branches with Various Convolutional Kernels: Generally a block consists of many branches, each of which employs different kernel sizes (1x1, 3x3, 5x5) for convolutional layers. The network can thus record characteristics at different spatial scales.Here 1x1 Conv filter is used before one 3x3 Conv filter to limit the number of parameters and 5x5 filter is factorized into two 3x3 Conv to reduce the number of parameters.

Residual Connections: Here the links allow the gradient to move more quickly on both forward and backward trips across the network. They help mitigate the vanishing gradient problem, which makes it easier to train very deep networks.

Batch Normalization and Activation: To stabilize and activate the features, batch normalization and activation functions, typically ReLUs—are carried out after each convolutional process.

The general organizational structure of the Inception-ResNet A block can be seen of as an assembly of parallel branches with convolutional layers patterned after Inception and residual connections modeled after ResNet.

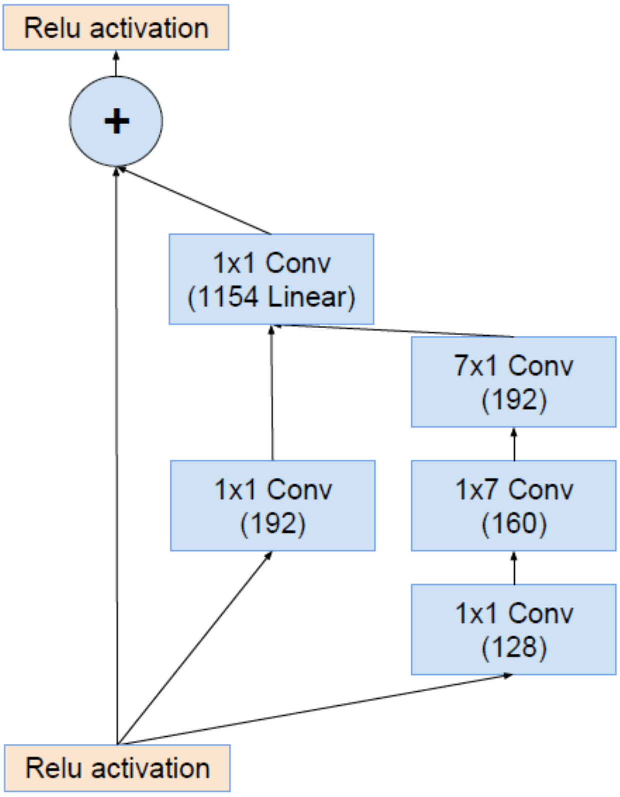

Figure 1.6: Block architectue of Inception Resnet B.

Another convolution block consist with 7x7 convolution filter where this filter is factorized into 1x7 and 7x1 two assymetric convolutions. This basically reduced the number of parameters.Here the 1x1 Conv is used before 7x7 Conv to limit the number of parameters as well.

Inception Resnet C is the another inception block in Inception Resnet V2 architecture where 3x3 convolution filter is used and before it 1x1 Conv is also used to reduce the parameters.Here, 3x3 convolution is also factorized into two asymmetric convolutions.The convolutions are occured parallelly and finally, all the outputs are concatenated. In this architecture, five times the Inception-Resnet-A block, ten times Inception-Resnet-B block and also five times of Inception-Resnet-C block are used with corresponding reduction blocks A and B. Reduction blocks are mostly used to reduce the spatial dimensions of feature maps prior to feeding them into higher layers. This leads to a reduction in computational complexity, allowing the network to focus on capturing more abstract and higher-level data. Additionally, the receptive field is increased and a larger context is captured by the model by reducing the spatial dimensions.

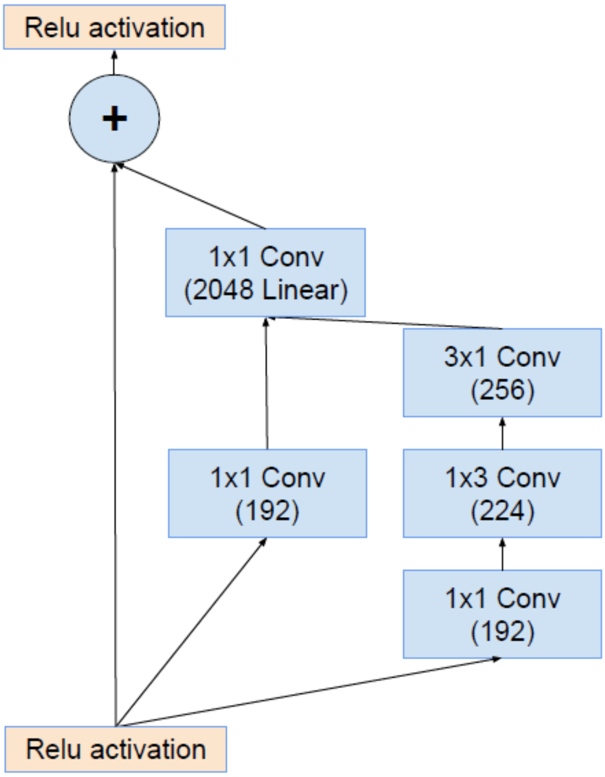

Figure 1.7: Block architectue of Inception Resnet C.

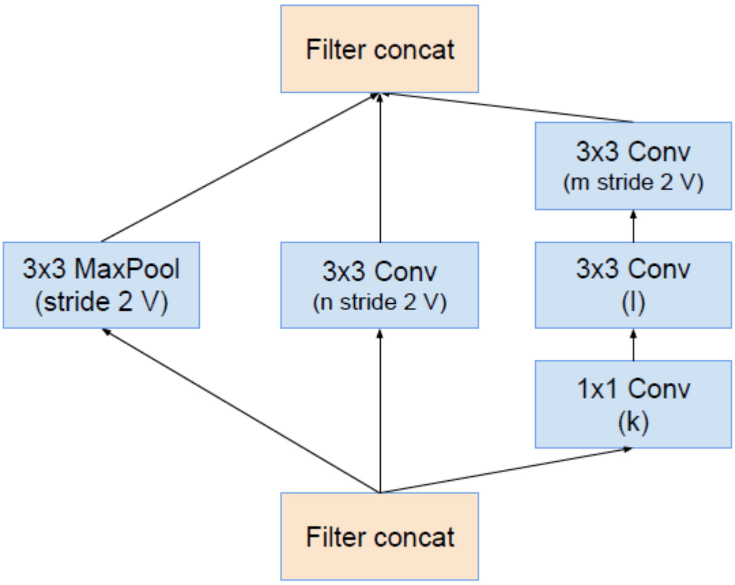

Figure 1.8: Block architectue of Reduction A.

In Inception-ResNet-V2 architecture, the term ”reduction block” refers to the part of the network where spatial dimensions are reduced and as a result the parameters are also reduced. This reduction is achieved by using convolutional layers with a stride greater than 1 or by using pooling operations. In IRv2 architecture there are two types of reduction blocks which are Reduction block A and B and they play a crucial role in reducing the spatial dimensions of the feature maps while preserving important information. Reducing spatial dimensions is beneficial for several reasons, including computational efficiency and the creation of a more compact feature representation. In Figure 1.8, it deontes the Reduction A block which contains convolutional layers with stride, pooling operation, dimensionality reduction and shortcut connection. Here the stride is 2 and k, l, m, n denotes the filter bank where the filter bank is the collection of multiple filters arranged systematically. The value of k, l, m, n are respectively 256, 256, 384 and 384 for the Inception-Resnet V2.

Again the Reduction-B block is specifically designed to reduce the spatial dimensions of feature maps while increasing the number of channels. Similar to other blocks in the Inception-ResNet-V2 network, Reduction-B block plays a great role in down-sampling the feature maps to capture high-level features as well as feature maps efficiently. The architecture of Reduction-B block is given below.

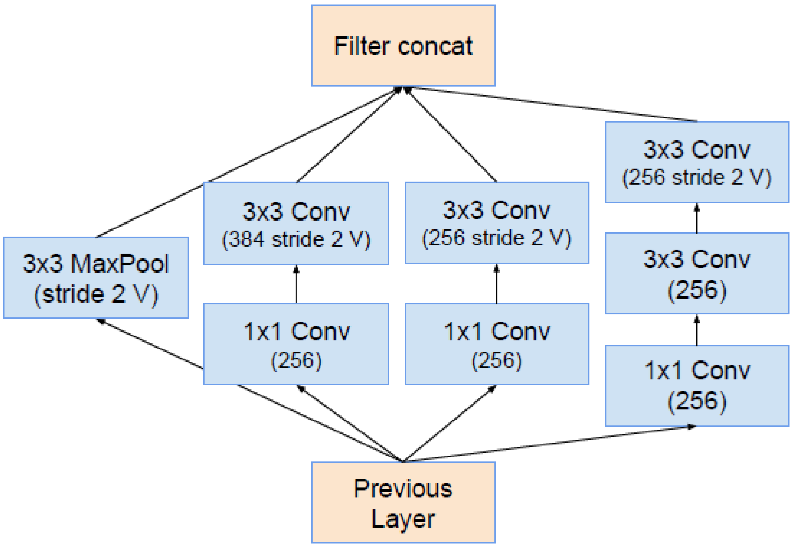

Figure 1.9: Block architectue of Reduction B.

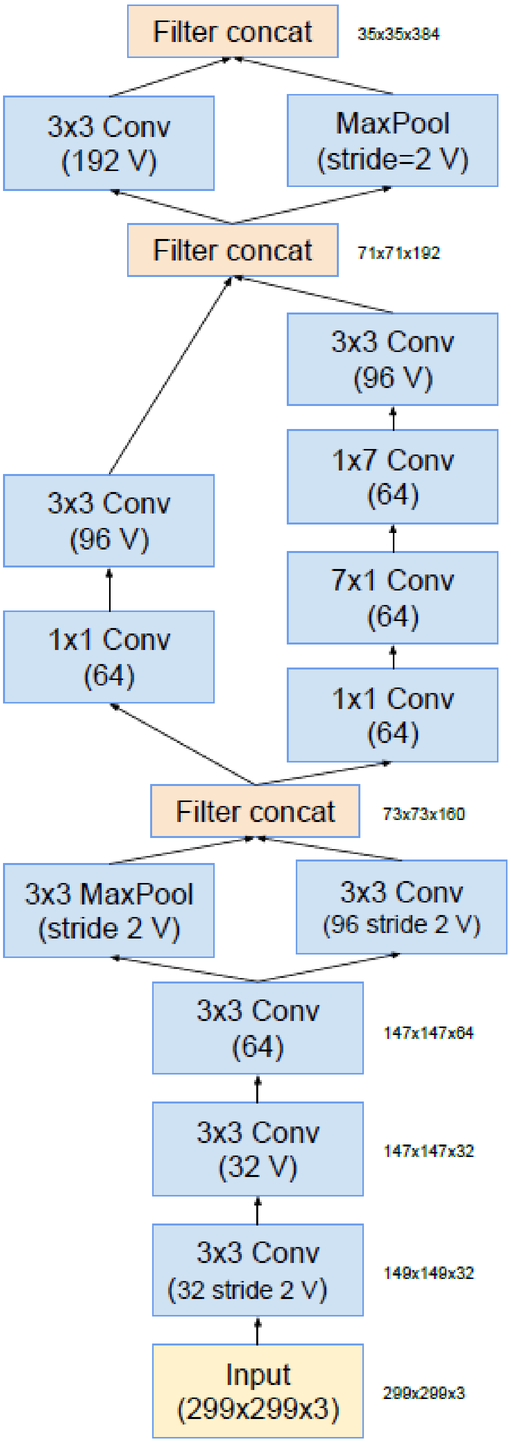

Figure 1.10: Block architecture of the stem.

The above figure denotes the stem block of Inception-Resnet V2 architecture where the block is used at the vary first time to extract the initial feature of the image.

1.4.8 Datasets

There are some labeled datasets which consist with RGB images with their corresponding ground truth depth maps that are used to train and evaluate the model.Some most used datasets for the purpose are NYU-Depth V2 [7], KITTI [8], DIODE [9], Make3D etc.In this model, NYU-Depth V2 [7] dataset is used which consists with labeling and has three types of data.Those are train, test and validation data.

Train data: The machine learning algorithm is developed using this kind of data. The input data that corresponds to an expected outcome is fed into the algorithm by the data scientist. The model continuously assesses the data to gain a deeper understanding of the data's behavior, after which it modifies itself to fulfill its intended function.

Validation data: During training, the model is supplied new data that it hasn't evaluated before. Using validation data, which provides the initial test against unknown data, data scientists can evaluate how well the model predicts based on the new data. While some data scientists do not use validation data, it can provide insightful information for refining hyperparameters, which impact the model's data evaluation process.

Test data: Once the model is built, testing data verifies that it can generate accurate predictions. The testing data shouldn't have labels if the training and validation data utilize labels to track the model's performance metrics. In order to confirm that the machine learning algorithm was effectively trained, test data provides the most comprehensive practical validation of an unknown dataset.

The structure of NYU-Depth V2 dataset:

- 1449 pairs of RGB and depth image alignment with densely labels.
- 464 scenes which are new and captured from three cities.
- 407,024 new unlabeled frames
- Every item has a lebel and an instance number assigned to it. The dataset consists of several components.
- Labeled: A segment of the video that exhibits a high level of label density across multiple classes. Preprocessing was also applied to this data to fill in any gaps in the

depth labels.

- Raw: The Kinect's unfiltered RGB, depth, and accelerometer data.
- Toolbox: Useful utilities to handle data and labels.

1.4.9 Others

There are some other tools which are used to implement the depth estimation model are likely to be loss functions with tuning, evaluation matrix, fine tuning of other pre-trained models etc.

1.5 Unfamiliarity of the Problem

There are some of the key areas where the problem can be particularly unfamiliar and challenging. Lack of accurate ground truth depth map, lack of efficient model architecture considering lower expensiveness and accuracy, ambiguity, data quantity and quality, model complexity etc are mostly common. There are some existing approaches to implement depth estimation model where increasing accuracy is the main factor.Various kinds of datasets ,models and other tools are used to implement it.If the image is too much noisy then proper depth map generation may be failed. So using more deeper network with different filtering approach in convolution layer may be effective to increase the accuracy and using lower number of parameters may reduce the model complexity. So using pre-trained model with annotated dataset, helps to generate the depth map of the complex objects with higher resolution as well.

1.6 Project Planning

A well-structured project plan increases the likelihood of project success, facilitates project management, and optimizes resource allocation. It is essential to update and adjust the plan frequently to ensure that it remains aligned with the project's evolving needs and circumstances. Therefore, the planning phase is crucial to the success of any project.

From Feb 21, 2023 to Feb 15, 2024 is the whole and estimated project completion time. Here from Feb 21, 2023, it takes 21 days to research and select the thesis topic. Then to learn about depth knowledge about the the topic it takes almost 28 days which is the literature review time. After around one month try to improve to existing work which are already done on the topic. Then from July 19,2023 experiment was started. After that within October the basic implementation of the experiment was done. After that some different strategies and methods are applied on the implementation for the improvement and compared them with other one. Then, the result analysis was started to fix the bugs and make the work more improved efficiently. Though there were some issues to implement it like low processing speed and takes more time to execute makes it more slower. After starting the experiment, writing a report was started on the topic and it was continued till January, 2024. Here report is written with specified formate which was given with required contents. Till now the progression of the work in provided in the chart.And finally, almost all the implementation is done with a good improvements of the existing model within Feb 15, 2024. Besides it is the plan to spent more time about result analysis to optimize the works. Here is the Gantt chart of the planning.

1.6.1 Gantt Chart

A Gantt chart is a visual representation of a project schedule that shows the timeline of various tasks or activities along with their dependencies. It provides a graphical view of the project plan, allowing project managers and team members to understand the schedule, track progress, and manage resources effectively. The Gantt Chart for project planning is shown in Figure 1.11.

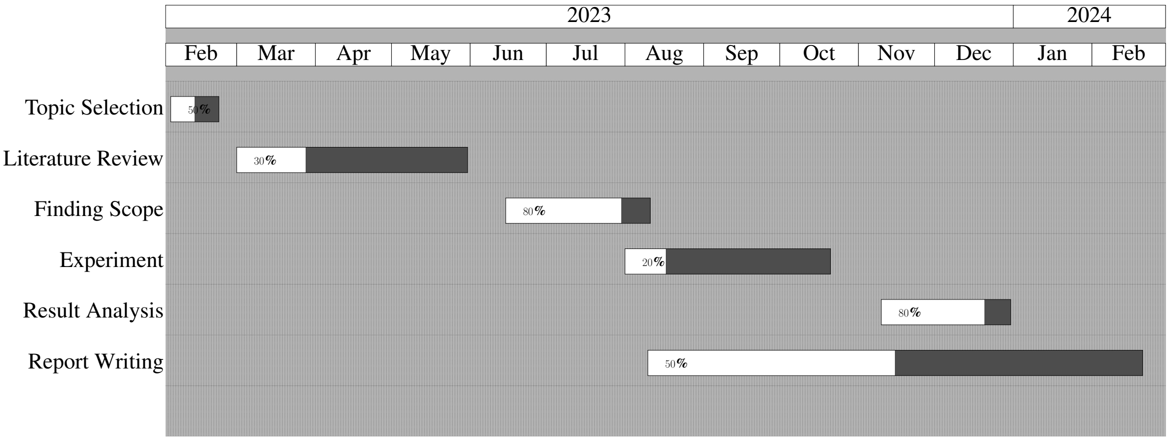

Figure 1.11: Gantt Chart for project planning.

1.7 Applications of the Work

i. Autonomus Driving

ii. Augmented Reality

iii. Navigation Systems and Tracking

iv. 3D Reconstruction

v. Medical Imaging and Robotics

vi. Automatic 2D-to-3D conversion in film

vii. Shadow mapping in 3D computer graphics

viii. Scene comprehension and simultaneous localization and mapping (SLAM)

1.8 Organization of the Thesis

The thesis is organized with all the steps to implement monocular depth estimation with other related topics. The report started with title page.There is an acknowledgement section. List of tables and figures are identified individually in this report. The report is divided into five chapters. In Chapter I, basic introduction with background, objectives, scopes and project planning with gantt chart are discussed. In Chapter II, there are some literature reviews included. The most and pupular literatures are reviewed here about monocular depth estimation and the literatures which are beneficial to understand the topic with different terms are analyzed properly. Here, the accuracy of different approaches and their effectiveness are compared. Besides, the basic functionalities which are needed to implement a model and their effectiveness are also highlighted in this section. After that some research gap solutions are identified in this chapter. In Chapter III, the methodology with necessary figures with equations are highlighted clearly. Here the working principle and the procedure about depth estimation models are analyzed. The model implementation associated with different functions and other models are analyzed here. After analyzing some models, the execution steps is focused here which can provide better improvements than before.

The implementation and the results are discussed in Chapter IV. Here the objectives which are achieved till now are provided. The results which are basically a predicted depth map are shown with different images serially by comparing with the corresponding depth map and RGB images from the test dataset. Here the comparison is occured among different pre-trained models by the percentage of mean accuracy and loss value helping with customized loss and accuracy functions. The depth map is shown using inferno-r cmap function for good visualization. In Chapter V by analyzing intellectual, ethical, safety and legal considerations the imapact of the thesis on societal, health and cultural issues are discussed here.

Moreover, there are some complex engineering problems and activities which are associated with current thesis are highlighted in this Chapter VI. Such as sensor modality where the choice of sensor has a big impact on the depth estimate. Different sensors, such as RGB-D cameras, LiDAR, ToF, stereo, and monocular cameras, have different features and challenges. And data collection where the quality and diversity of the dataset are significant factors. Putting together annotated data with accurate ground truth depth information can be a laborious task. Ensure that the collection include a range of environments, lighting conditions, and object types. And finally in Chapter VII, the overall summary of the thesis are described in a paragraph where different kinds of model architectures with their impacts, advantages and disadvantages and other essential functions are described. Besides the challenges and limitations which are already faced are highlighted and the future works of our project are also discussed in this chapter.

CHAPTER II

Literature Review

2.1 Introduction

This literature review aims to critically evaluate and summarize previous research, shedding light on important themes, approaches, and conclusions pertaining to monocular depth estimation. The objective in reviewing the literature is to find gaps, inconsistencies, and new patterns that will guide future research efforts. The literatures which are related to monocular depth estimation are reviewed here.

2.2 Literature Review

An improved model to present a convolutional neural network for computing a high-resolution depth map given a single RGB image with the help of transfer learning [10]. Here the depth map captured object boundaries more faithfully with fewer parameters and less training iterations.In this paper, in preprocessing steps the image is flipped horizontally if required which is called image augmentation. Besides tuning of the loss function by changing with weights make it more perfect. A straightforward encoder-decoder architecture with skip connections is used here where the encoder part is used as a pre-trained truncated DenseNet-169 [11] .In quantitavely the method may not be best but the quality of the produced depth map is much better. Here the accuracy on the test dataset (NYU-Depth V2) is 84.6% which is better than other previous results and using other pretrained model with fine tuning can make it more accurate.

Another proposal is a fully convolutional architecture, encompassing residual learning, to model the ambiguous mapping between monocular images and depth maps [12].It also focused on the improvement of the output resolution through efficient learning of feature map upsampling.The model is trained end-to-end and does not contains any post processing techniques like CRFs or other refinement techniques.The proposed architecture builds upon ResNet-50.Here NYU Depth v2 dataset is used where the size of the input image is 302x228. Here the error rate is 0.573 in RSE and 0.195 in RMSE(log) which are better than other approaches.

An approach [13] of using the tools that enable mixing multiple datasets during training, even if their annotations are incompatible.Basically, the experiments confirm that mixing data from complementary sources greatly improves monocular depth estimation.Mixing multiple datasets like NYU Depth v2, KITTI, TUM, DIODE etc. provides better accuracy than individual one.

A method [14] to match detailed depth boundaries without the need for superpixelation.A regress on the depth using a neural network with two components happened here. one that first estimates the global structure of the scene, then a second that refines it using local information. The network is trained using a loss that explicitly accounts for depth relations between pixel locations, in addition to pointwise error. The system achieves state-of-the art estimation rates on NYU Depth and KITTI, as well as improved qualitative outputs.Here the RMSE(linear) error is 7.156 and the RMSE(log) error is 0.270 as well.

A double estimation method [15] that improves the whole-image depth estimation and a patch selection method that adds local details to the final result.Here using modified MiDaS network ,the RMSE error is 0.1598.The method is demonstrated by merging estimations at different resolutions with changing context to generate high resolution depth map.

With a good benchmark values,one of the methods is GAN based monocular depth estimation [16] which works with three processes.Those are segmentation and depth estimation, adversarial loss calculations and cycle consistency loss calculations.NYU Depth v2 data set is here with the RMSE of 0.652 and RMSE(log) is 0.217 approximately.

In inception-Resnet V2 [6], training with residual connections accelerates the training of Inception networks significantly. It consists with 164 layers and worked on ImageNet dataset. In the architecture of the Inception-Resnet V2, there are three Inception-Resnet block and two reduction block. The inception blocks are used repeatedly at the time of training. In each block, there are 1x1, 3x3 and 5x5 types of convolution filters. Here the 1x1 Conv block is used before other filters to limit the number of parameters and asymmetric factorization on these filters plays a great role to reduce the number of parameters. Different sizes of these filters make it more versatile to extract the different sized features from an image. In this architecture, the input image size is 299x299 which can be RGB image.In this paper, different kinds of Inception architectures are analyzed. Such as, Inception V1 (GoogleNet), Inception V2, Inception V3, Inception V4, Inception-Resnet V1 and Inception-Resnet V2. Basically, the literature showed about the improvement of the inception architecture using Residual network.

A less complex Convolutional Neural Network (CNN) architecture is used to predict coarse depth, and surface normal guidance is then used to improve the coarse depth pictures. Compared to the current states, more comprehensive depth maps are generated with fewer network parameters and a more straightforward learning framework. Uisng less parameters makes it more faster for execution with considering detailed depth map. Furthermore, the framework's suitability for integration into a monocular simultaneous localization and mapping (SLAM) paradigm is confirmed by 3D point cloud maps that are rebuilt from depth prediction pictures. In this paper [17], the contribution is reflected in proposing an RGB-D surface normal network, which eectively captures the geometric relationships between RGB and depth images. Here the performance is evaluated on NYU-Depth V2 dataset and the RMSE error is 0.459 with 18 epochs. Besides by using Densenet 121 with 32.4 M parameters, the accuracy is 0.859 (85.9%) for the test set of the dataset.The subject of future study is pre-trained depth estimation networks which will be used to 3D vision techniques like SLAM, SFM, and AR (simultaneous localization and mapping).

2.3 Discussion of Research Gap Solution

To generate depth map using depth estimation, there are many more gaps and challanges which are needed to be solved. Analyze the complexity of the models is one of the major concerns. Another key points are edge handling and scale ambiguity which can be handled by evaluating depth maps. Besides there is a enough lack of accuracy of the generated depth map compared to ground truth depth map. So using fine tuning of the pre-trained model can put a great contribution to improve the accuracy. Identifying research gaps is a crucial aspect which has some potential solutions like multi-model fusion, some post-processing techniques, data synthesis, hybrid architectures, loss function configuration etc. Now the table proves that the problem idea is a new one and not acquired directly from existing sources.

Table 2.1: Table for proving the thesis idea unique.

Proposed Method	**Research Gap**	**Solution**
High Quality Monocular Depth Estimation via Transfer Learning [10] results high quality depth map with good resolution.	Fine tuning of the more efficient pre-trained model with less number of parameters can improve the depth map visualization	The model can be more flexible using lower number of parameters.So using U-Net architecture with Inception-Resnet V2 [6] as an encoder can be effective.
Another proposal is a fully convolutional architecture, encompassing residual learning, to model the ambiguous mapping between monocular images and depth maps [12].	The model is trained end-to-end and does not contains any post processing techniques like CRFs or other refinement techniques.Here, the error rate is 0.573 in RSE while using NYU Depth V2 dataset.	Using post processing technique is one of the vital tasks to make the depth estimation model more accurate.The proposed model can hopefully reduce this error.
Towards Robust Monocular Depth Estimation: Mixing Datasets for Zero-shot Cross-dataset Transfer	The model can not find the depth map of the printing or painting image it occured in the basis of reflections of the pixels.	Our model hopefully estimate the depth map for the printing image as well as the noisy image using preprocessing of the model.
A double estimation method [15] that improves the whole-image depth estimation.	Here using modified MiDaS network ,the RMSE error is 0.1598 and needs high resolution image to train the model.	Using high resolution image datasets can be huge and needed higher end pc to be executed.So using low resolution image with deeper network can increase accuracy.
With a good benchmark values,one of the methods is GAN based monocular depth estimation [16]	NYU Depth v2 data set is here with the RMSE of 0.652 and RMSE(log) is 0.217 approximately.	The accuracy can be improved with reducing the error rate by tuning loss functions.
A method [14] to match detailed depth boundaries without the need for superpixelation.	The method can not improve quantitative output that means there can be high errors.	Considering these, the improvement method is concerned about both qualitative and quantitative errors and increase the accuracy to find complex object’s depth map.

CHAPTER III

Methodology

3.1 Introduction

This section describes the encoder-decoder architecture that is used to estimate a depth map from a single RGB image. Next, the relationship between the complexity of the encoder and decoder and performance is examined. The next suggestion is a suitable loss function for the task at hand. Lastly, effective augmentation policies are given that greatly facilitate the training process.

3.2 Proposed Methodology

3.2.1 Problem Design and Analysis

Using U-Net architecture with normal encoder-decoder will not be able to generate better depth map. But the model can be fine tuned using pretrained model provides better accurate result. So neet to implement a model via transfer learning with proper optimization algorithm. So, to make a proper model, it is necessary to follow the organized way for implementing the model. At the below sections, all the steps are provided sequentially.

3.2.2 Data Collection and Preprocessing

This is the initial steps to collect dataset from online resources or real life resources. Then preprocess the dataset to resize every elements of it and make it convenient to implement with models.Besides, normalization and augmentation are more effective in pre-processing.

Selecting dataset is the main concern on which the execution or run time and the accuracy of the model depends on.Here NYU-Depth V2 [7] dataset is used which is basically an indoor dataset consisting with both RGB images and GT depth map where the data is divided into three sets: training, testing, and validation sets. In this dataset the image size is 640x480 with containing 120k training samples and 654 testing samples. But the model is trained on

65k subset. At the time of preprocessing, the images are resized into same sizes which are used as input. Then normalization is occured within [0 to 1]. After that, horizontal flipping (i.e. mirroring) of images are considered at a probability of 0.5. Ground truth depth map is used to find the accuracy by comparing with estimated depth map and also used to train the model as depth estimation model.

3.2.3 Model Architecture

Secondly , a CNN based simplified depth estimation model is selected which follows U-Net architecture. The model is implemented with the help of tensorflow python framework or library. The U-Net is an encoder-decoder network, with the RGB image as input and depth map as output.The encoder network consists of convolution and pooling layers to capture the depth features, and the decoder network includes deconvolution layers to regress the estimated pixel-level depth map, with the same size as the input.In encoding and decoding section, upsampling and downsampling are occured respectively.

In this thesis, U-Net architecure is used to implement the model, but at the encoder part of the architecture, it is substituted by some high performing pre-trained models. So, here the down-sampling is occured via truncated pre-trained model's layers and at the decoder section, the previous downsampled images are concatenated with upsamplemd images by skip connections. Therefore, as pretrained models, Resnet50, Densenet169, VGG19 and Inception-Resnet V2 [6] are used and among them using Inception-Resnet V2 provides a better result as an encoder. The Inception-Resnet V2 has three inception block with consisting of different sizes of convolution filters. And each block can be repeated multiple times in the architecture. Here, the model is trained using 15 epochs on the train set because more than 15 epochs causes overfit issue. In each epoch, the forwardpass is occured among the layers and resulted the predicted values. After that the predicted values are compared with actual values and loss values are found using appropriate loss functions. This is called finding gradient and then descent the loss, the backwardpass is occured and then previous weights are updated by new weights. So, in this model 'Adam' gradient-discent optimization algorithm is used. It is known as adaptive moment estimation. This is the method which can compute adaptive learning rate for each parameters.

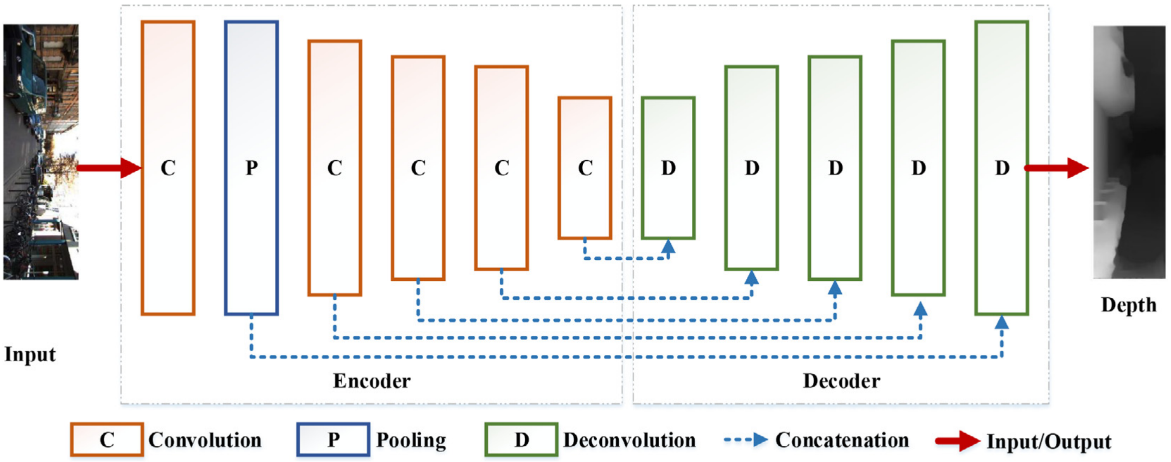

Figure 3.1: The CNN based general pipeline of deep learning for monocular depth estimation.

The above figure defines the fully connected convolutional network which is consisted with both convolutional and pooling layer in encoding poriton where through downscaling the image resolution is also reduced. In decoding section,additive skip-connections are implemented in the downscaling block where the deconvolution and unpooling are occured to recover the image with higher resolution.This is basically an U-Net architectural model which is used to segment to object to generate the depth map. As it is an encoder-decoder network, the encoder can part can be used as other pre-trained layers of the model which makes great contribution of the result. Using the network, the overall depth estimation model structure is given below.

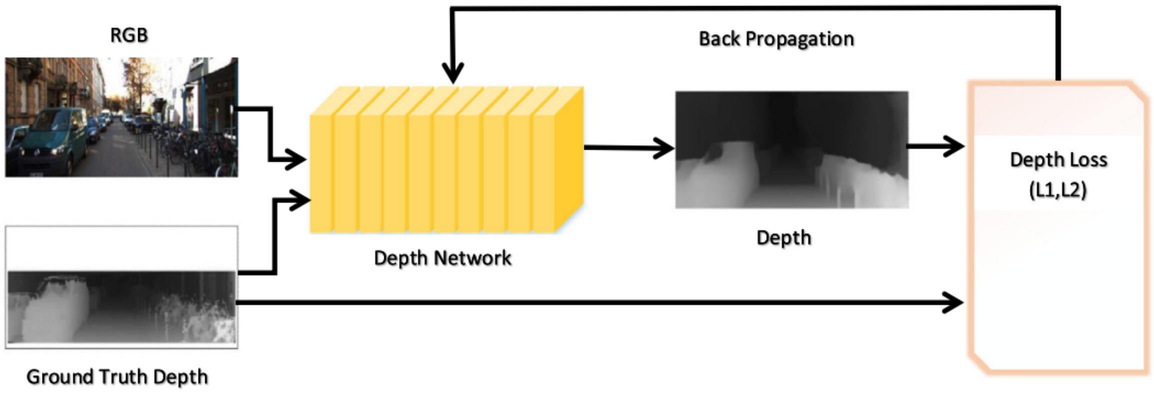

Figure 3.2: The general model of supervised learning for monocular depth estimation.

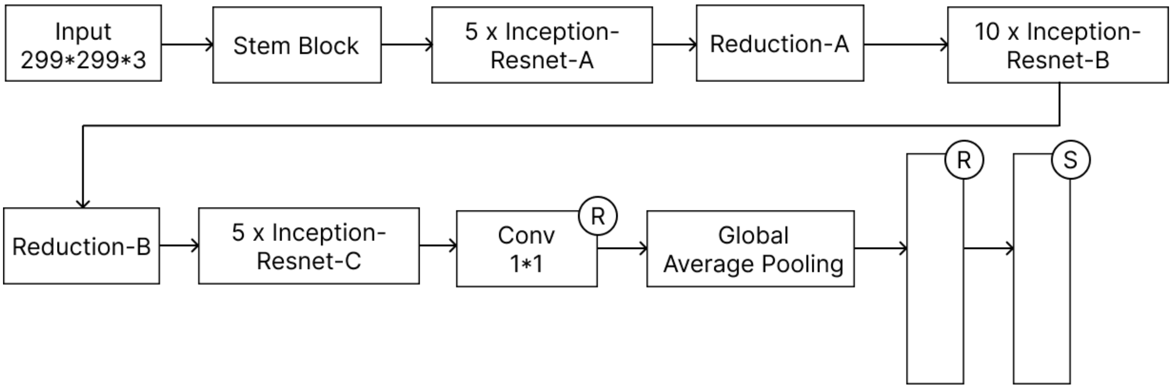

Figure 3.3: The flow diagram of Inception-Resnet V2.

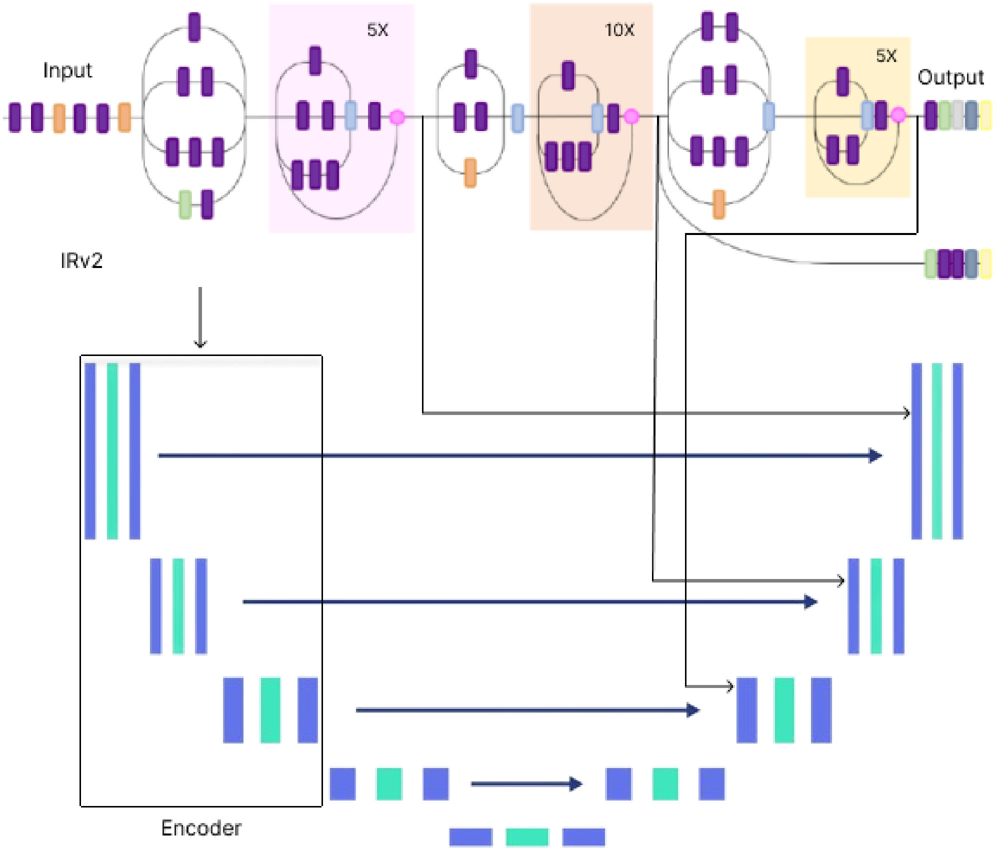

Figure 3.4: The detailed architecture of U-Net with Inception-Resnet V2.

In the final model, it is built on U-Net architecure with the help of high performing Inception-Resnet V2 pretrained model which help to generate more detailed and high quality depth map with better accuracy than other models. Here, the truncated layers of IRv2 are used for the purpose of downsampling and then the image is decoded with simplified way.

3.2.4 Building a Data Pipeline

The pipeline takes a dataframe containing the path for the RGB images, as well as the depth and depth mask files. It reads and resize the RGB images. It reads the depth and depth mask files, process them to generate the depth map image and resize it. It returns the RGB images and the depth map images for a batch.

3.2.5 Building the model

For building the model, there are two important functions need to implement. One is downsampling the block in encoder section and the other one is upsampling the block in the decoder section where skip-connections are implemented in the downsampling block. Here, the truncated layers of IRv2 are used for downsampling through convolution and pooling and then the downsampled image is concatenated at the time of upsampling through skip connection. Basically at bottleneck, a bridge establish the connection between encoder and decoder for upsampling and recovering the depth map.

3.2.6 Loss Function

The difference between the groundtruth depth map y and the predicted depth map $\hat{y}$ is taken into account by a typical loss function for depth regression issues. The training pace and total depth estimation performance can be greatly impacted by various loss function factors. A wide range of loss function modifications are used to optimize the neural network for depth estimation. In this report a customize loss funnction is used which helps to improve the accuracy of the model by tuning weight. For training the network, loss L is defined between y and $\hat{y}$ as the weighted sum of three loss functions:

$$L(y,\hat{y}) = \lambda L_{\text{depth}}(y,\hat{y}) + L_{\text{grad}}(y,\hat{y}) + L_{\text{SSIM}}(y,\hat{y}) \tag{3.1}$$

where L_{depth}, L_{grad}, and L_{SSIM} are depth, gradient, and SSIM loss terms, respectively.

$$L_{\text{depth}}(y, \hat{y}) = \frac{1}{n} \sum_{p}^{n} |y_p - \hat{y}_p| \tag{3.2}$$

$$L_{\text{grad}}(y, \hat{y}) = \frac{1}{n} \sum_{p}^{n} |g_x(y_p, \hat{y}_p)| + |g_y(y_p, \hat{y}_p)| \tag{3.3}$$

$$L_{\text{SSIM}}(y, \hat{y}) = \frac{1 - \text{SSIM}_{(}y, \hat{y})}{2} \tag{3.4}$$

Here, L_{depth} is the point-wise L1 loss defined on the depth values, L_{grad} is the L1 loss defined over the image gradient g of the depth image and L_{SSIM} uses the Structural Similarity (SSIM) term which is a commonly-used metric for image reconstruction tasks.

3.3 Conclusion

In conclusion, the proposed methodology represents a depth estimation model which can generate depth map with high detailing of the image. Here the methedology is described by analyzing all the steps which are needed to make the model. Besides the full model architecture is highlighted here with better visualization to make the better understanding of it. In this model, the customized loss function is described in more precisely where the functions can be tuned using several weights.

CHAPTER IV

Implementation, Results and Discussions

4.1 Introduction

In the implementation phase, the proposed methods are applied sequentially to implement the final model in order to get the desired depth map with better detailing of the objects. Here, the implementation of the model is described and after that, the predicted results are analyzed with depth map visualization and finally, compare the depth map with ground trupth depth map by considering different pre-trained models. Evaluation martrics are used in this section for the comparison through finding the loss values and accuracy values after training of the model.

4.2 Experimental Setup

The optimal setup proposed by the author is divided into two parts. One is hardware setup and the other one is software setup.

Hardware Setup:

- Camera: A monocular camera can capture RGB image.
- Sensor: At the time of capturing the image,it can measure the depth between two objects. Such as Lidar which can calculate the depth more precisely.
- Computer: GPU which can accelarate the training of the model and so NVIDIA GPUs are commonly used to train.
- Storage: The dataset which is larger, needs the memory to store it and besides, in the storage the trained model is saved.
- Power Supply: Ensuring of a stable power supply is necessary.

Software Setup:

- Operating System: Choose a operating system which is compatible with Linux, Windows and Mac OS.
- Deep Learning Frameworks: To implement the model, Keras is used where Keras is a high level API of tensorflow framework.
- Depth Estimation Model: U-Net architecture with IRv2 pre-trained model as an encoder.
- Libraries and Dependencies: Various kinds of libraries and dependencies are used which are already integrated in Keras. Such as OpenCV, Numpy, Matplotlib etc.
- Evaluation tools: For the evalution of the result, the error metrics are used for qualitative evaluation.
- Dataset: NYU-Depth V2 dataset, which is appropriate for depth estimation model is also used to train the model.

4.3 Evaluation Metrics

The error metrics which are defined for the evaluation of the model are defined by the following equation which is called quantitative evaluation.

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2} \tag{4.1}$$

$$\log(RMSE) = \left(\sqrt{\frac{1}{N}\sum_{i=1}^{N}(\log(y_i) - \log(\hat{y}_i))^2}\right) \tag{4.2}$$

$$\text{ARE} = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \tag{4.3}$$

$$\text{SRE} = \frac{1}{N}\sum_{i=1}^{N}\frac{(y_i - \hat{y}_i)^2}{y_i} \tag{4.4}$$

4.4 Dataset

This study uses the 640 x 480 resolution NYU Depth v2 dataset, which provides photos and depth maps for a variety of interior settings. A training subset of 65K samples is used with the dataset, which has 120K training samples and 654 testing samples. Ten meters is the top limit on the depth maps. The network produces predictions with a resolution of 320 x 240, which is half of the input resolution. The input photos are downscaled to 320 x 240 while the ground truth depths are maintained at their original resolution for training purposes. In the testing stage, the whole test image's depth map prediction is calculated and then upsampled by two times to match the resolution of the ground truth.

4.5 Implementation and Result

In this phase, after completing the initialization and preprocessing steps the built model is executed for the training on NYU-Depth V2 dataset where the image is already preprocessed. It takes 15 epochs to train the model which provides the best result rather than more than 15 epochs. For the epoch, gradient discent optimization algorithm is used named 'Adam'. Besides the loss function is customized here which can tune the loss values in order to find the best weights. Evaluation matrics are implemented to compare and evaluate the model with other models and determines it's accuracy.

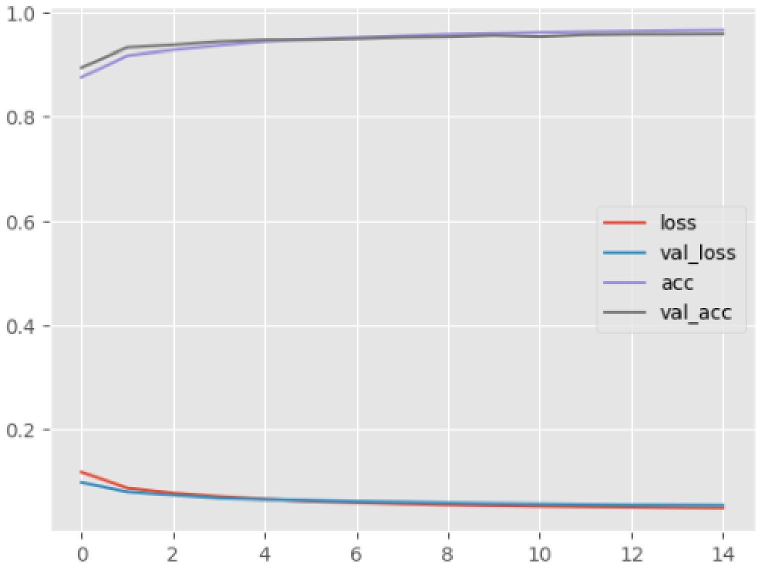

Figure 4.1: Loss and accuracy graph of training and validation set of the model.

The above graph determines the loss and accuracy of the train and validation set of the dataset. Here the above two curves denote the train and validation accuracy and the below two curves determine the train and validation loss. It is cleared that, initially there is a quite bit difference between train and validation loss as well as accuracy. But after some epochs, the training accuracy is getting closer to the validation accuracy. i.e the more closer training curve to validation curve, the more accurate the result is. So we got the highest accuracy is 0.8533 and the lowest loss is 0.1523 after fitting the model.

The depth maps which are generated using the trained models are given below. Here, the original image, their corresponding ground truth depth map and the predicted depth maps are provided in a row consecutively. Besides the detailed depth maps are visualized using inferno-r color map which is basically represented by RGB image and lower the intensity of the pixel means, higher the distance of the object from the camera.

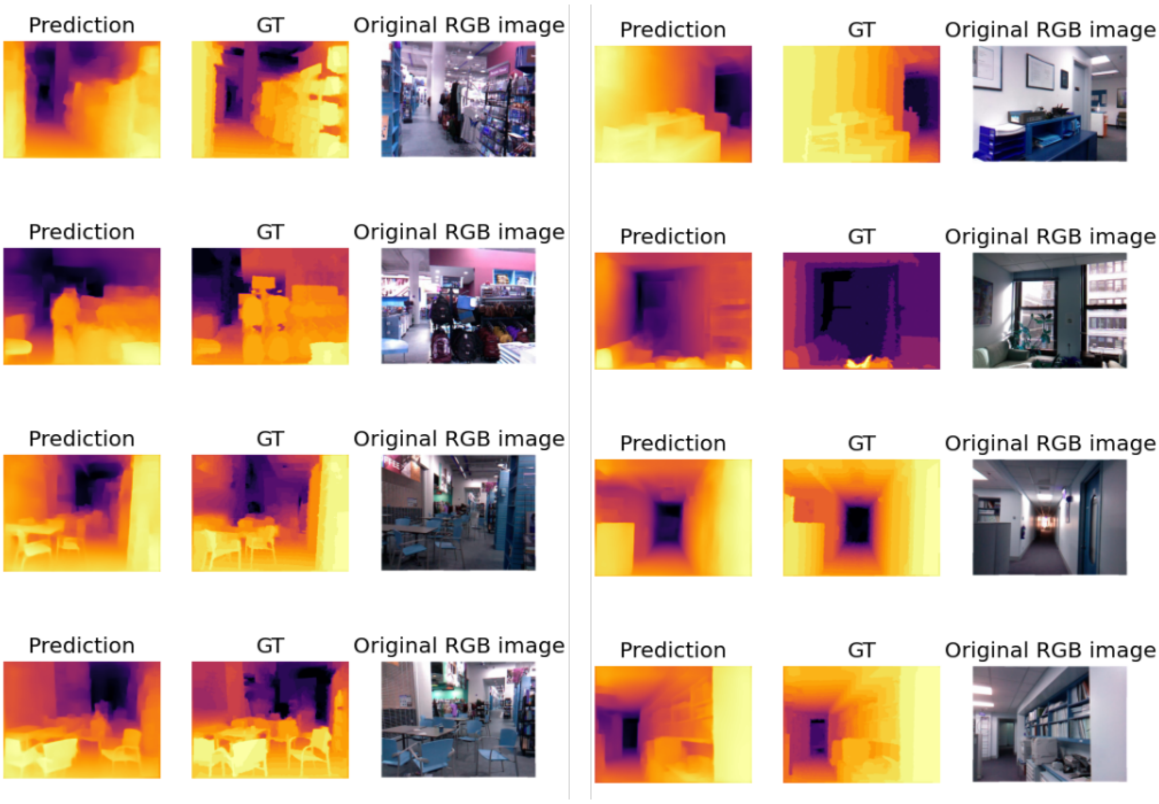

Figure 4.2: Generated depth maps with their corresponding ground truth depth maps of the RGB images.

Table 4.1: Evaluation metrics with different types of models.

Model Name	Parm.	Epochs	RMSE ↓	ARE ↓	CLFE ↓	Accuracy ↑
Enc-Unet	8.6 M	15	0.0063	0.0238	0.1921	0.7642
Resnet50	14.4 M	15	0.0069	**0.0149**	0.1675	0.8073
VGG19	22.4 M	15	0.0098	0.0559	0.1711	0.7979
Densenet169	12.2 M	15	**0.0057**	0.0406	0.1538	0.8489
IRv2	31.1 M	15	0.0066	0.0710	**0.1524**	**0.8533**

In this table, CLFE stands for Error by Customized Loss Function. RMSE is Root Mean Squared Error and ARE is called as Absolute Relative Error. Besides, the Inception Resnet V2 model is defined as IRv2. After analyzing the evaluation metrics, using IRv2 model can provide higher accuracy on test data of NYU-Depth V2 dataset. But in overall cosideration, Densene169 has almost closer accuracy to IRv2 but it has huge lower parameters and so densenet169 is more convenient in this case. Here, lower the value of RMSE, ARE, CLFE and higher the value of the accuracy denotes good result.

Quantitative results. Here the amount of RMSE error for IRv2 model is lowest, CLFE is also lower than others, but ARE error is lower in the resnet50 model. So by using overall considerations, IRv2 can be better model to improve the result. **Qualitative results.** Here the customized loss function is used to evaluate the quality of the model on NYU-Depth V2 dataset. Here, three types of loss functions are used to find the tuned values in field of depth estimation. One loss is point wise loss, another one is gradient loss and finally SSIM loss is calculated and the merging of these three losses build a well tuned function to define the quality. U-Net with normal encoder-decoder provides the accuracy of 0.7642 but using IRv2 as an encoder with U-Net improves the accuracy drastically which is 0.8533.

4.6 Objective Achieved

Almost all the objectives which are planned from started are achieved. The plan was using different types of high-performing pretrained model as an encoder can improve the accuray and so using IRv2 is able to improve the accuracy rather than using the Densenet169. Though there are still some issues about train time, but that is not the concern here where accuracy is the main focused.

4.7 Financial Analysis and Budget

Financial analysis and budget in monocular depth estimation is complex because it depends on various factors including the scope of the project, resources required, team size, infrastructure costs and more. But there are some requirements which are needed to implements and advertise the depth estimation model and those are equipment or software, dataset, publication fees etc. Here it was tried to minimize the financial cost by using online IDE like kaggle where it provides the facilities of GPU. Besides it provides a larger storage to save the model and store the dataset. Moreover, in this IDE it supports all kinds of python framework like tensorflow which makes it more easier to implement the model. So, these kinds of facilities reduce the financial cost drastically. Using table, it can be analyzed considering general factors and basic requirements. Besides the total budget allocation for maintaining the thesis is given through table.

Table 4.2: Table for Financial Budget Plan.

Budget Category	Estimated Cost
Software and Equipment	500-1000 BDT
Cost for Dataset	200-500 BDT
Cloud Computing Service	1000-1500 BDT
Infrastructure	800-1000 BDT
Miscellaneous	500-1000 BDT
Total Estimation of Budget	3000-5000 BDT

Table 4.3: Table for Financial Analysis.

Field	Analysis
Research and Development	Licensing fees for using existing deep learning frameworks or libraries
Infrastructure and Computing Resources	Cost of hardware and software for developing and testing the depth estimation models
Validation and Testing	Resources needed for testing the accuracy and costs associated with creating validation datasets
Project Management	Compensation for project managers and coordinators
Marketing and Communication	Budget for promoting the project's outcomes
Sustainability and Long-Term Costs	Cost for maintaining and updating the models and software

4.8 Conclusion

In conclusion, there are some basic implementation which must be needed to run the model properly. After implementing the model, the required output was found with a good accuracy but not the best. Here, the accuracy and the loss are measures through evaluation functions which are also discussed in this section. But to do all these things, financial analysis with the budget plan is required which provides the overall estimated cost to conduct the thesis. Though this is not the best way to improve the accuracy but, the idea leads to another better path for the same purposes. In summary, even if the current implementation shows encouraging results, more study and development are needed to address current issues and advance the field of depth estimation. Beyond theoretical frameworks, the model's potential use is demonstrated by its effective implementation in autonomous navigation.

CHAPTER V

Social Health, Environment, Safety, Ethical, Legal and Cultural Issues

5.1 Introduction

This chapter demonstrates the socio-economic impact and the ethical, safety, and legal considerations in the field of monocular depth estimation. It also describes the intellectual properties considerations and societal, health and cultural issues in this phase.

5.2 Intellectual Property Considerations

The legal and ethical considerations surrounding the use of datasets and models have received careful consideration throughout this thesis. A thorough approach to securing the necessary consents and authorizations from the relevant creators or owners of the datasets and models is combined with strict adherence to ethical norms. To ensure the highest regard for intellectual property rights and adherence to usage terms, a rigorous mechanism is in place. Additionally, a concentrated effort is made to provide credit and recognition to the original writers, both as a demonstration of intellectual integrity and as a way to foster a courteous and cooperative academic atmosphere. By upholding these principles, this thesis ensures the responsible and ethical use of data, honors the rights of content creators, and advances transparency in the realm of research practices.

5.3 Ethical Considerations

There are several ethical concerns with the use of depth estimation model. Concerns are raised by the advanced estimation technology's ability to measure distance with high resolution. Given that depth estimation model have the capacity to produce harmful effects—such as harmful robot design for wrong purposes. it is crucial that moral guidelines and standards be established in order to prevent any improper use of this technology. In doing the thesis, caution was taken to ensure that no privacy was violated by the data used to train the model. Furthermore, it is now crucial to establish moral standards for usage.

5.4 Safety Considerations

The thesis is established based on the model which needs to be trained on a large dataset, and originality of the dataset was a concern throughout the whole process. Before the training of the model, the data had been preprocessed and after training post processing refines the output but it had been made sure of the fact that, no original information carried in the data or image does not get altered in the process. This ensures the security of the depth estimation model. Because if the data in the object is altered, then there is a huge possibility of generating a significant error which can be more dangerous in the fields of it's applications. It only generates the depth map, keeping the original details of the image intact.

5.5 Legal Considerations

The thesis centers its legal concerns on three main issues: intellectual property, copyright, and responsibility. The dataset that was used for training and testing was, first of all, freely available to the public and contained no copyrighted content. It is essential to acknowledge that no private or personal data was included in the study without express consent from the relevant owners. Furthermore, there are possible hazards associated with developing any depth estimate methodology, especially when using it. As users are ultimately accountable for the pictures that the model processes, an application monitoring mechanism must be in place. This methodology guarantees an anticipatory response to any unanticipated difficulties or moral dilemmas that can emerge when applying the depth estimation model.

5.6 Impact of the Project of Societal, Health and Cultural Issues

The thesis on depth estimation has important ramifications for several social, health, and cultural concerns, which might lead to both opportunities and difficulties:

Social Impact: Depth estimation facilitates the making of apps for virtual reality (VR) and augmented reality (AR), advertising immersive communication experiences in a variety of social scenarios. Concerns about privacy violation are raised by the broad use of depth estimating systems. Strong privacy frameworks must be established in order to protect people's personal information and stop unlawful data gathering as these technologies become more widely used.

Health Impact: Medical imaging applications rely heavily on depth estimation algorithms to help with details diagnosis and treatment planning of a wide range of medical disorders. It helps with the accurate measuring of tumor forms, sizes and locations in oncology. With the use of technology, telemedicine solutions may be developed, giving medical personnel the ability to evaluate patients' illnesses from a distance and deliver prompt interventions, especially in underserved or rural places.

Cultural Impact: Depth estimation uses 3D scanning and modeling techniques to help preserve cultural heritage places and items. It makes it possible to create digital duplicates of historical sites and objects, ensuring their preservation and availability for the next generations. Through the creation of immersive experiences and virtual exhibits, it expands the creative possibilities in the entertainment and art sectors.

5.7 Impact of the Project on the Environment and Sustainability

The environmental and sustainable effects of a depth estimate project might take many different forms, with both positive and negative outcomes.

Positive Impacts:

- An essential component of robotics and autonomous systems is depth estimation. Improved depth sensing can result in autonomous navigation that is more energy-efficient, which makes lower total energy consumption in applications including environmental monitoring, agricultural, and warehouse automation.
- A more economical use of resources like water, fertilizer, and pesticides is made possible by precision farming, which is maintained by depth estimation in the field of agriculture. As a result, waste and environmental effects are reduced, which supports sustainable agriculture practices.
- The depth estimate abilities of the proposed thesis may be used as the basis of application to monitor the environment, supporting activities such as tracking species, mapping habitats, and evaluation of the effects of climate change.

Negative Impacts:

- The inadequate management of electronic trash (e-waste) may result from the manufacture of depth-sensing devices like cameras and sensors. To reduce the influence on the environment, it is essential to provide appropriate recycling and eliminating procedures.

- Large computational resources could be needed to develop reliable monocular depth estimation techniques. It is possible to reduce the possible environmental effect of algorithmic complexity by investigating optimization approaches and confirming energy-efficient implementations.

- The resources needed to manufacture depth sensing equipment are extracted from the environment, such as metals and rare earth minerals. Initiatives for recycling and sustainable sourcing can help in addressing these issues and difficulties.

- A major challenge is promoting sustainable applicatoins. Here, enhancing the technology's beneficial effects on the environment may be achieved by promoting the development and use of monocular depth estimation in fields like precision agriculture and environmental monitoring, which have obvious sustainability advantages.

5.8 Conclusion

In conclusion, the comprehensive evaluation of the project's social, health, environmental, safety, ethical, legal, and cultural components highlights the project's importance within the larger societal framework. The initiative emphasizes the necessity for a sensitive and responsible approach by acknowledging the interactions between technology and the several disciplines. By using these factors, the project may be made to achieve both technological and social goals, allowing innovation to be smoothly and properly incorporated into many areas of human existence. So, the thesis has a great impact on the wide range of fields, each with unique issues and outcomes.

CHAPTER VI

Addressing Complex Engineering Problems and Activities

6.1 Complex engineering problems associated with the current thesis

Table 6.1: Table for complex engineering problems associated with the current thesis.

Attribute		Address the Attributes of Complex Engineering Problems
Depth of knowledge required	P1	In the thesis, the proper reviews are occured on deep learning and monocular depth estimation. Besides, for choosing the appropriate model, deeper knowledge is necessary to know about the strengths and weakness of the model. The thesis includes training process, hyperparameters and loss function tuning, optimization alogrithms etc which helps to increase the knowledge more deeper. And also the thesis defines and justifies the evaluation metrics and their implementations to compare the model with others.
Range of conflicting requirements	P2	There is a major confliction occured between accuracy and computational efficiency. Because it is hard to maintain high accuracy and computational efficiency at the same time. Besides a conflict occurred between data quality and quantity whether to use high resolution data is hard with a large amount. So the adjustment is occured in the thesis with using proper ways to solve these.
Depth of analysis required	P3	The depth of analysis is reflected in the following key areas with their comprehensive analysis. Those areas are literature review, model selection and architecture, dataset analysis, training strategies, evaluation metrics, ethical implications etc.
Familiarity of issues	P4	The thesis identifies and discusses key challenges to monocular depth estimation, such as handling occlusions, addressing scale ambiguity, and dealing with limited training data. So, identifying challenges demonstrates a deep understanding of the unique issues associated with monocular depth estimation.
Extent of applicable codes	P5	Here the attributes are considered as code implementation, code documentation, code validation and testing, code integration etc. which can ensure the applicability of the thesis fields.
Extent of stakeholder involvement and conflicting requirements	P6	The thesis begins by identifying and categorizing relevant stakeholders associated with the idea.Besides gathering feedback from them is important to resolve the conflicts.
Interdependence	P7	Interdependencies occurred between the choices of hyperparameters and model performance. Good initialization of the hyperparameters make the model more appropriate. The justification is that, the thesis addresses the interdependence between the model's performance and its adaptability to diverse real-world scenarios.

6.2 Complex engineering activities associated with the current thesis

Table 6.2: Table for complex engineering activities associated with the current thesis.

Attribute		Address the Attributes of Complex Engineering Activities
Range of resources	A1	In this thesis, the computational resources are needed to train and implement deep learning models for monocular depth estimation. And the justification is, by considering computational resources, the thesis acknowledges the importance of scalability and efficiency in model development. Besides, the range of data resources needed for training deep learning models is addressed here and a variety of data resources enhances the model's ability which helps to apply in real-world applications. Hardware resources are another crucial things which should be addressed. The availability of GPUs or TPUs have a great impact on the execution of deep learning model which increases the speed and efficiency of training deep learning models. So by ensuring hardware resources, the thesis ensures that the proposed solution is adaptable to different computing infrastructures.
Level of interaction	A2	The thesis describes the iterative process of the model, highlights the level of interaction involved in adjusting parameters, architecture, or training strategies based on feedback and intermediate results. Here the process ensures that the model evolves and improves over time. User feedback is a good thing for the interaction and so by adding user feedback, the depth estimation model becomes more applicable and relevant. Collaborative development is another level of interaction where it can lead to diverse perspectives and expertise and enrich the research.
Innovation	A3	The thesis explore a innovation strategy which makes the model more prefarable to improve the accuracy. Using the Inception-Resnet-v2 pretrained model as an encoder with U-Net architecture to predict the depth map is a new thing that can provide a better result than other models.
Consequences for society and the environment	A4	Consequences lie in its application fields for society and the environment. Using the basic knowledge of the thesis, it may have a great impact on the field of autonomous driving, medical imaging, robotics, and other real-world applications.
Familiarity	A5	The thesis shows a deep understanding of core concepts related to monocular depth estimation, including stereo vision principles, depth perception etc. which are more familiar with previous ideas described in the literature review. Familiarity with core concepts is crucial for the better implementations with innovations. Moreover, familiarity with different models is an important term in implementation. Such as IRv2 pre-trained model added extra benefits by improving the result.

CHAPTER VII

Conclusions

7.1 Summary

The report introduces a convolutional neural network (CNN) for high-resolution monocular depth estimation from single RGB images, emphasizing the use of transfer learning. The network follows an encoder-decoder architecture called U-Net. In this architecture, initializing the encoder to extract features from pre-trained networks and employing augmentation, tuning of loss function and training strategies for improving accuracy. Here some pretrained networks are used like Resnet50, VGG19, Densenet169 and Inception-Resnet v2 (IRv2) and among them IRv2 provides the highest accuracy inspite of having high expensiveness. Before using a pre-trained model instead of an encoder, the U-Net (only) provides the accuracy 0.7620 but after using of IRv2 as an encoder, it produces the accuracy of 0.8533. Despite of having a simple decoder, the proposed method achieves detailed high-resolution depth maps. The network, with fewer parameters and training iterations, surpasses state-of-the-art performance on NYU-Depth v2 dataset, producing qualitatively superior results that faithfully capture object boundaries. Besides, the socio-economic impact and the ethical, safety, and legal considerations in the field of monocular depth estimation are demonstrated here and the complex engineering problems and activities are also addressed.

7.2 Limitations

The main goal is to generate a depth map with good accuracy which plays a great role in scene understanding. But there are some limitations which should be considered for the implementation:

i) Using larger dataset causes higher memory consumption and needed high-end pc to run the model.

ii) Using the truncated layers of the pre-trained model as an encoder, makes the model more complex which is computationally expensive.

iii) There is a higher chance to be delay issue because runtime of the model is extremely high which causes problems to train the model.
iv) Using a deeper network increases computational complexity as well.
v) If the images of the dataset are so noisy, it fails to generate a good depth map.
vi) The dimensions of the input image influence the number of parameters. So, using high-resolution images increases the number of parameters which can cause overfitting problems.

7.3 Recommendations and Future Works

There are some factors which should be improved to make the result more convenient. So, there will be further testing and validations using a range of datasets and benchmarking against state-of-the-art depth estimation methods to assess the resilience and generalization capabilities of the proposed model. Besides, fine-tuning the both trained and pre-trained model on specific target domains or datasets to enhance its performance in real-world applications. Merging multiple datasets to make it more versatile which can provide better result than before. Again optimization of the model's performance can be occured by hyper-parameter tuning, like tuning of learning rates, batch sizes, and regularization parameters. Integration of the model in the real-time application and so improve the model's ability to estimate depth in dynamic scenarios. Finally enhances the interpretability of the model, making it more transparent and understandable.

So, after considering the above recommendations, the future works are more likely to be:
i) Multi-model fusion to improve depth estimation accuracy and robustness.
ii) Fine-tuning the trained model to get a more convenient depth map.
iii) Continual learning approach where a model learns continuously from new data over time without forgetting the knowledge it has acquired from previous tasks.
iv) Using better optimization algorithm to make the model more flexible with less complexity and faster execution.
v) Keep the consistency of the accuracy of estimated depth map using a comparatively smaller dataset by training the model.
vi) Combining diverse datasets is a good strategy to enhance the model's robustness.

REFERENCES

[1] A. Basu, V. Buch, W. Vogels, and T. von Eicken, “U-net: A user-level network interface for parallel and distributed computing,” Oct. 1996. DOI: 10.1145/224056.224061.

[2] K. Prilianti, T. Brotosudarmo, S. Anam, and A. Suryanto, “Performance comparison of the convolutional neural network optimizer for photosynthetic pigments prediction on plant digital image,” vol. 2084, Mar. 2019. DOI: 10.1063/1.5094284.

[3] S. Targ, D. Almeida, and K. Lyman, “Resnet in resnet: Generalizing residual architectures,” Mar. 2016.

[4] K. Simonyan and A. Zisserman, *Very deep convolutional networks for large-scale image recognition*, 2015. arXiv: 1409.1556 [cs.CV].

[5] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, *Densely connected convolutional networks*, 2018. arXiv: 1608.06993 [cs.CV].

[6] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi, *Inception-v4, inception-resnet and the impact of residual connections on learning*, 2016. arXiv: 1602.07261 [cs.CV].

[7] I. Laina, C. Rupprecht, V. Belagiannis, F. Tombari, and N. Navab, *Deeper depth prediction with fully convolutional residual networks*, 2016. arXiv: 1606.00373 [cs.CV].

[8] H. Zhang, X. Chen, H. Lu, and J. Xiao, “Distributed and collaborative monocular simultaneous localization and mapping for multi-robot systems in large-scale environments,” *International Journal of Advanced Robotic Systems*, vol. 15, p. 172 988 141 878 017, May 2018. DOI: 10.1177/1729881418780178.

[9] R. Patwari and V. Ly, *Analysis computational complexity reduction of monocular and stereo depth estimation techniques*, Jun. 2022. DOI: 10.48550/arXiv.2206.09071.

[10] I. Alhashim and P. Wonka, “High quality monocular depth estimation via transfer learning,” Dec. 2018.

[11] K. Alafandy, H. Omara, M. LAZAAR, and M. Al Achhab, "Investment of classic deep cnns and svm for classifying remote sensing images," *Advances in Science Technology and Engineering Systems Journal*, vol. 05, pp. 652–659, Oct. 2020. DOI: 10.25046/aj050580.

[12] I. Laina, C. Rupprecht, V. Belagiannis, F. Tombari, and N. Navab, "Deeper depth prediction with fully convolutional residual networks," Oct. 2016. DOI: 10.1109/3DV.2016.32.

[13] R. Ranftl, K. Lasinger, D. Hafner, and V. Koltun, "Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PP, pp. 1–1, Aug. 2020. DOI: 10.1109/TPAMI.2020.3019967.

[14] D. Eigen, C. Puhrsch, and R. Fergus, "Depth map prediction from a single image using a multi-scale deep network," vol. 3, Jun. 2014.

[15] S. M. Hosseini Minagoleh, S. Dille, L. Mai, S. Paris, and Y. Aksoy, "Boosting monocular depth estimation models to high-resolution via content-adaptive multi-resolution merging," Jun. 2021, pp. 9680–9689. DOI: 10.1109/CVPR46437.2021.00956.

[16] D.-h. Kwak and S.-h. Lee, "A novel method for estimating monocular depth using cycle gan and segmentation," *Sensors*, vol. 20, p. 2567, Apr. 2020. DOI: 10.3390/s20092567.

[17] K. Huang, X. Qu, S. Chen, *et al.*, "Superb monocular depth estimation based on transfer learning and surface normal guidance," *Sensors*, vol. 20, p. 4856, Aug. 2020. DOI: 10.3390/s20174856.