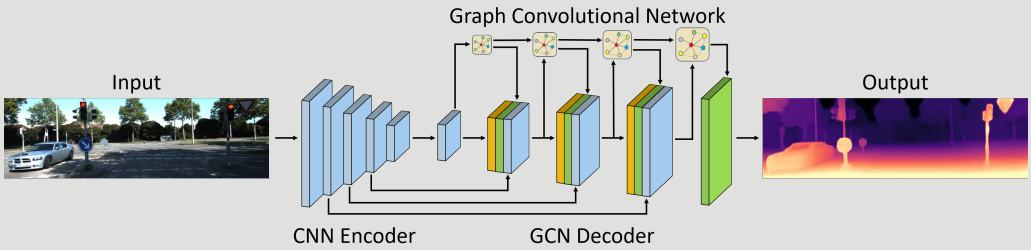
ENHANCING DISTANCE PREDICTION THROUGH MONOCULAR DEPTH ESTIMATION BASED ON GRAPH CONVOLUTIONAL NETWORKS

Armin Masoumian

UNIVERSITAT ROVIRA i VIRGILI

Enhancing Distance Prediction through Monocular Depth Estimation based on Graph Convolutional Networks

Armin Masoumian



DOCTORAL THESIS

2023

UNIVERSITAT ROVIRA I VIRGILI
ENHANCING DISTANCE PREDICTION THROUGH MONOCULAR DEPTH ESTIMATION BASED ON GRAPH CONVOLUTIONAL NETWORKS
Armin Masoumian

Enhancing Distance Prediction through Monocular Depth Estimation based on Graph Convolutional Networks

DOCTORAL THESIS

Author:

Armin Masoumian

Supervisors:

Prof. Domenec Puig

Dr. Hatem A. Rashwan

Department of Computer Engineering and Mathematics

UNIVERSITAT ROVIRA i VIRGILI

Tarragona

2023

UNIVERSITAT ROVIRA I VIRGILI
ENHANCING DISTANCE PREDICTION THROUGH MONOCULAR DEPTH ESTIMATION BASED ON GRAPH CONVOLUTIONAL NETWORKS
Armin Masoumian

UNIVERSITAT ROVIRA i VIRGILI

Departament d'Enginyeria Informàtica i Matemàtiques

Av. Països Catalans, 26
43007 Tarragona, Spain
Tel. +34 977 55 95 95
Fax. +34 977 55 95 97

We STATE that the present study, entitled “Enhancing Distance Prediction through Monocular Depth Estimation based on Graph Convolutional Networks ", presented by Armin Masoumian, for the award of the degree of Doctor, has been carried out under our supervision at the Departament d'Enginyeria Informàtica i Matemàtiques.

Tarragona, November 2023.

Doctoral Thesis Supervisors,

UNIVERSITAT ROVIRA i VIRGILI
PUIG VALLS DOMÈNEC SAVI - 39869760L
2023.11.01 18:04:58 +01'00'

Hatem Abdellatif Fatahallah Ibrahim Mahmoud - DNI Y0895796Y (TCAT)
Digitally signed by Hatem Abdellatif Fatahallah Ibrahim Mahmoud - DNI Y0895796Y (TCAT)
DN: C=ES, O=Universitat Rovira i Virgili, OID.2.5.4.97=VATES-Q9350003A, OU=Empleat públic de nivell mig, SN=Abdellatif Fatahallah Ibrahim Mahmoud - DNI Y0895796Y, G=Hatem, SERIALNUMBER=IDCES-Y0895796Y, CN=Hatem Abdellatif Fatahallah Ibrahim Mahmoud - DNI Y0895796Y (TCAT)
Reason: I am the author of this document
Location: your signing location here
Date: 2023.11.02 09:15:18+01'00'
Foxit Reader Version: 10.1.3

Prof. Dr. Domènec Savi Puig Valls

Dr. Hatem A. Rashwan

UNIVERSITAT ROVIRA I VIRGILI
ENHANCING DISTANCE PREDICTION THROUGH MONOCULAR DEPTH ESTIMATION BASED ON GRAPH CONVOLUTIONAL NETWORKS
Armin Masoumian

To be yourself in a world that is constantly trying to make you something else is the greatest accomplishment!

UNIVERSITAT ROVIRA I VIRGILI
ENHANCING DISTANCE PREDICTION THROUGH MONOCULAR DEPTH ESTIMATION BASED ON GRAPH CONVOLUTIONAL NETWORKS
Armin Masoumian

This thesis is dedicated to my family, for their unending love and support

UNIVERSITAT ROVIRA I VIRGILI
ENHANCING DISTANCE PREDICTION THROUGH MONOCULAR DEPTH ESTIMATION BASED ON GRAPH CONVOLUTIONAL NETWORKS
Armin Masoumian

Abstract

As robots and autonomous vehicles become more advanced, the need for accurate depth measurements is increasing. One way to achieve this is through depth estimation (DE), a crucial task in computer vision that can be accomplished through various techniques, including deep learning (DL). Self-supervised monocular depth estimation (MDE) is a cutting-edge technology that aims to estimate object depth in a scene using just one image without expensive stereoscopic or 3D cameras. Recent advancements in DL techniques have made this possible, with models using complex algorithms to extract features from the image and estimate object distances. Graph convolutional networks (GCN) have improved model accuracy by handling non-Euclidean data while combining multiple loss functions has helped to deal with lousy depth predictions and preserve object discontinuities. This promising technology has numerous applications in robotics engineering and autonomous vehicles.

Firstly, we present a comprehensive review of the latest advancements in MDE using deep learning techniques. We highlight critical points from various perspectives, including input data shapes, training methods, and evaluation indicators. In addition, we discuss the limitations of DL-based MDE models, including their accuracy, computational time, real-time inference, transferability, input image shape, domain adaptation, and generalization.

Secondly, we present a novel approach to MDE that utilizes GCN to estimate depth maps from monocular videos. Traditional convolutional neural networks (CNNs) struggle to handle non-Euclidean data and irregular image regions within a topological structure where GCNs excel. Our proposed self-supervised MDE model includes two parallel auto-encoder networks. One network uses ResNet-50 and multi-scale GCN to estimate the depth map, while the other uses ResNet-18 to calculate the ego-motion vector between consecutive frames. We use a combination of loss functions to handle bad depth prediction and preserve object discontinuities. Our method demonstrates

promising results, achieving a high prediction accuracy of 89% on the KITTI dataset and reducing the number of trainable parameters by 40% compared to state-of-the-art (SOTA) solutions.

Thirdly, estimating the distance between objects and the camera sensor using 2D images poses a challenging task. Therefore, we introduce a DL framework that utilizes two separate networks for depth estimation and object detection by using a single image. The proposed approach employs You Only Look Once (YOLOv5) to detect and localize objects within the scene and a deep autoencoder network to compute the estimated depth image. The presented framework was evaluated on real outdoor images, achieving an impressive accuracy rate of 96% with a root mean square error (RMSE) of 0.203 for the correct absolute distance.

Overall, our study demonstrates the effectiveness of our self-supervised MDE approach based on graph convolutional networks through both quantitative and qualitative comparisons with other SOTA methods. The results highlight the significant advantages of our proposed depth prediction technique.

Keywords: Deep learning, Monocular Depth Estimation, Autoencoder Network, Graph Convolutional Network, Self-supervision, Single Image Depth Estimation, Multi-task Learning, Unsupervised Learning.

Resum

A mesura que els robots i els vehicles autònoms es tornen més avançats, la necessitat de disposar de mesures de profunditat precises està augmentant. Una manera d'assolir això és mitjançant l'estimació de la profunditat (DE), una tasca crucial en la visió per computador que es pot aconseguir mitjançant diverses tècniques, incloent l'aprenentatge profund (DL). L'estimació de la profunditat monocular autosupervisada (MDE) és una tecnologia innovadora que pretén estimar la profunditat dels objectes en una escena utilitzant només una imatge sense necessitat de càmeres estereoscòpiques o 3D costoses. Els avenços recents en les tècniques de DL han fet això possible mitjançant models que utilitzen algoritmes complexos per extreure característiques de la imatge i estimar les distàncies dels objectes. Les xarxes convolucionals gràfiques (GCN) han millorat la precisió del model en gestionar dades no euclidianes, mentre que la combinació de múltiples funcions de pèrdua ha ajudat a lidiar amb prediccions de profunditat dolentes i a preservar les discontinuïtats dels objectes. Aquesta tecnologia prometedora té nombroses aplicacions en l'enginyeria de robots i vehicles autònoms.

En primer lloc, presentem una revisió exhaustiva dels darrers avanços en MDE utilitzant tècniques d'aprenentatge profund. Destaquem punts crítics des de diverses perspectives, incloent les formes de les dades d'entrada, els mètodes d'entrenament i els indicadors d'avaluació. A més, discutim les limitacions dels models de MDE basats en DL, incloent-ne la seva precisió, el temps computacional, la inferència en temps real, la transferència, la forma de la imatge d'entrada, l'adaptació de domini i la generalització.

En segon lloc, presentem una nova aproximació a MDE que utilitza GCN per estimar mapes de profunditat a partir de vídeos monoculars. Les xarxes de convolució tradicionals (CNN) tenen dificultats per gestionar dades no euclidianes i regions

d'imatge irregulars dins d'una estructura topològica, on les GCN destaquen. El nostre model auto-supervisat MDE proposat inclou dues xarxes autoencoder paral·leles. Una xarxa utilitza ResNet-50 i GCN de múltiples escales per estimar el mapa de profunditat, mentre que l'altra utilitza ResNet-18 per calcular el vector de moviment de la càmera entre fotogrames consecutius. Fem servir una combinació de funcions de pèrdua per gestionar una mala predicció de la profunditat i preservar les discontinuïtats dels objectes. El nostre mètode mostra resultats prometedors, aconseguint una alta precisió de predicció del 89% en la base de dades KITTI i reduint el nombre de paràmetres entrenables en un 40% en comparació amb les solucions de l'estat de l'art (SOTA).

En tercer lloc, estimar la distància entre els objectes i el sensor de càmera utilitzant imatges 2D suposa un repte. Per això, presentem un nou model de DL que utilitza dues xarxes separades per a l'estimació de la profunditat i la detecció d'objectes mitjançant una única imatge. L'aproximació proposada fa servir You Only Look Once (YOLOv5) per detectar i localitzar objectes dins de l'escena i una xarxa autoencoder profunda per calcular la imatge de profunditat estimada. El sistema presentat s'ha avaluat en imatges reals a l'exterior, aconseguint una excel·lent taxa de precisió del 96% amb un error mitjà quadràtic de 0,203 per a la distància absoluta correct.

En conjunt, el nostre estudi demostra l'eficàcia de la nostra aproximació d'auto-supervisió MDE basada en les xarxes convolucionals gràfiques mitjançant comparacions tant quantitatives com qualitatives amb altres mètodes SOTA. Els resultats destaquen les avantatges significatives de la nostra tècnica de predicció de la profunditat proposada.

Paraules clau: Aprenentatge profund, Estimació de la profunditat monocular, Xarxa autoencoder, Xarxa de convolució gràfica, Auto-supervisió, Estimació de la profunditat amb una sola imatge, Aprenentatge multi-tasca, Aprenentatge no supervisat.

Resumen

A medida que los robots y los vehículos autónomos se vuelven más avanzados, la necesidad de mediciones precisas de profundidad está aumentando. Una forma de lograr esto es a través de la estimación de profundidad (DE), una tarea crucial en la visión por computador que se puede llevar a cabo mediante diversas técnicas, incluido el aprendizaje profundo (DL). La estimación de profundidad monocular auto supervisada (MDE) es una tecnología de vanguardia que tiene como objetivo estimar la profundidad de los objetos en una escena utilizando solo una imagen sin necesidad de cámaras estereoscópicas o 3D costosas. Los avances recientes en las técnicas de DL han hecho posible esto, con modelos que utilizan algoritmos complejos para extraer características de la imagen y estimar las distancias de los objetos. Las redes convolucionales gráficas (GCN) han mejorado la precisión del modelo al manejar datos no euclidianos, mientras que la combinación de múltiples funciones de pérdida ha ayudado a lidiar con predicciones de profundidad deficientes y preservar las discontinuidades de los objetos. Esta tecnología prometedora tiene numerosas aplicaciones en la ingeniería de robótica y vehículos autónomos.

En primer lugar, presentamos una revisión exhaustiva de los últimos avances en MDE utilizando técnicas de aprendizaje profundo. Destacamos puntos críticos desde diversas perspectivas, incluidas las formas de los datos de entrada, los métodos de entrenamiento y los indicadores de evaluación. Además, discutimos las limitaciones de los modelos de MDE basados en DL, incluida su precisión, tiempo computacional, inferencia en tiempo real, transferibilidad, forma de la imagen de entrada, adaptación de dominio y generalización.

En segundo lugar, presentamos un enfoque novedoso para MDE que utiliza GCN para estimar mapas de profundidad a partir de videos monoculares. Las redes neuronales convolucionales tradicionales (CNN) tienen dificultades para manejar datos no euclidianos y regiones de imagen irregulares dentro de una estructura topológica en la

que las GCN sobresalen. Nuestro modelo MDE auto supervisado propuesto incluye dos redes autoencoder paralelas. Una red utiliza ResNet-50 y GCN de múltiples escalas para estimar el mapa de profundidad, mientras que la otra utiliza ResNet-18 para calcular el vector de movimiento de la cámara entre fotogramas consecutivos. Utilizamos una combinación de funciones de pérdida para manejar predicciones deficientes de profundidad y preservar las discontinuidades de los objetos. Nuestro método demuestra resultados prometedores, logrando una alta precisión de predicción del 89% en el conjunto de datos KITTI y reduciendo el número de parámetros entrenables en un 40% en comparación con las soluciones del estado del arte (SOTA).

En tercer lugar, estimar la distancia entre objetos y el sensor de la cámara utilizando imágenes 2D plantea reto. Para ello, presentamos un nuevo modelo de DL que utiliza dos redes separadas para la estimación de profundidad y la detección de objetos mediante el uso de una única imagen. El enfoque propuesto utiliza You Only Look Once (YOLOv5) para detectar y localizar objetos dentro de la escena y una red autoencoder profunda para calcular la imagen de profundidad estimada. El sistema presentado se evaluó en imágenes reales en exteriores, logrando una excelente tasa de precisión del 96% con un error cuadrático medio (RMSE) de 0.203 para la distancia absoluta correct.

En general, nuestro estudio demuestra la eficacia de nuestro enfoque de MDE auto supervisado basado en redes convolucionales gráficas mediante comparaciones cuantitativas y cualitativas con otros métodos SOTA. Los resultados destacan las ventajas significativas de nuestra técnica de predicción de profundidad propuesta.

Palabras clave: Aprendizaje profundo, Estimación de profundidad monocular, Red autoencoder, Red convolucional gráfica, Auto supervisión, Estimación de profundidad en una sola imagen, Aprendizaje multi tarea, Aprendizaje no supervisado.

Acknowledgements

I am extremely grateful to my supervisors, Prof. Dr. Domenec Puig Valls and Dr. Hatem A. Rashwan, for their invaluable guidance, constructive feedback, and constant encouragement throughout the completion of this thesis. Their extensive knowledge, experience, and innovative problem-solving skills were pivotal in shaping my work and helping me grow as a researcher. I am indebted to them for their patience and availability whenever I needed their support.

Furthermore, I would like to express my appreciation for the stimulating research environment fostered by Dr. Domenec Puig, which enabled me to interact with a group of exceptional graduate students and postdocs. In particular, I am grateful to Dr. Julian Cristiano and Dr. Mohammed Abdel Nasser for their insightful discussions and constructive criticisms that have significantly enhanced the quality of my research work.

During my six-month stay at the University of California, Riverside, USA, I was fortunate to have the opportunity to work in Dr. Salman Asif's team. I want to thank him for his generosity and support during my stay.

I also wish to acknowledge my family's unwavering love and support, without which I would not have been able to complete this challenging task. I am especially grateful to my parents and aunt for their constant encouragement and motivation.

Finally, I would like to acknowledge the financial support provided by the Secretaria d'Universitats i Recerca del Departament d'Empresa i Coneixement de la Generalitat de Catalunya (2020 FISDU 00405). that made this research possible. I am deeply grateful for their investment in my work.

UNIVERSITAT ROVIRA I VIRGILI
ENHANCING DISTANCE PREDICTION THROUGH MONOCULAR DEPTH ESTIMATION BASED ON GRAPH CONVOLUTIONAL NETWORKS
Armin Masoumian

Contents

Abstract ix

Resumen **xi**

Resum **xiii**

Acknowledgements **xv**

1 Introduction **1**

1.1 Motivation 2

1.2 Approach 5

1.3 Contributions and Publications 7

1.4 Thesis Organization 9

2 Background **11**

2.1 Introduction 12

2.2 Depth Estimation 13

2.2.1 Different Approaches of Depth Estimation 14

2.2.1.1 Traditional Methods 15

2.2.1.2 Machine Learning Methods 17

2.2.1.3 Deep Learning Methods 19

2.2.2 ML Vs. DL Depth Estimation 21

2.2.3 Different Types of Depth Estimation 22

2.2.3.1 Monocular Depth Estimation 22

2.2.3.2 Stereo Depth Estimation . . . 24
2.2.3.3 Structured Light Depth Estimation . . . 25
2.2.3.4 Time-of-Flight (TOF) Depth Estimation . . . 26
2.2.3.5 LiDAR Depth Estimation . . . 27
2.3 Graph Convolutional Networks . . . 28
2.4 Autoencoders . . . 30
2.5 Summary . . . 31

3 Monocular Depth Estimation Using Deep Learning: A Review 33
3.1 Introduction . . . 34
3.2 Depth Estimation . . . 38
3.3 Datasets . . . 40
3.3.1 KITTI . . . 41
3.3.2 NYU Depth-V2 . . . 42
3.3.3 Cityscapes . . . 43
3.3.4 Make3D . . . 43
3.3.5 DIODE . . . 44
3.3.6 Middlebury 2014 . . . 44
3.3.7 Driving Stereo . . . 45
3.4 Evaluation Metrics . . . 45
3.5 Input Data Shapes for MDE Applying DL . . . 46
3.5.1 Mono-Sequence . . . 47
3.5.2 Stereo Sequence . . . 48
3.5.3 Sequence to Sequence . . . 49
3.6 MDE Applying DL Training Manners . . . 50
3.6.1 Supervised Learning Approach . . . 51
3.6.2 Unsupervised Learning Approach . . . 53
3.6.3 Semi-Supervised Learning Approach . . . 56

3.7 Discussion . . . 58
3.7.1 Accuracy . . . 59
3.7.2 Computational Time . . . 60
3.7.3 Resolution Quality . . . 60
3.7.4 Real-Time Inference . . . 61
3.7.5 Transferability . . . 62
3.7.6 Input Data Shapes . . . 62
3.7.7 Future Study . . . 63
3.8 Conclusions . . . 64

4 GCNDepth: Self-supervised Monocular Depth Estimation Based on Graph Convolutional Network 67

4.1 Introduction . . . 68
4.2 Background and Related Work . . . 70
4.2.1 Supervised Depth Estimation . . . 71
4.2.2 Self-supervised Depth Estimation . . . 72
4.2.3 Graph Neural Network . . . 73
4.3 Method . . . 75
4.3.1 Problem Definition . . . 77
4.3.2 Graph Convolutional Network . . . 78
4.3.3 Self-supervised CNN-GCN Autoencoder . . . 79
4.3.3.1 DepthNet Encoder . . . 80
4.3.3.2 DepthNet Decoder . . . 80
4.3.3.3 PoseNet Estimator . . . 82
4.3.4 Overall Pipelines . . . 83
4.3.5 Geometry Models and Losses . . . 84
4.3.6 Implementation Details . . . 86
4.4 Experiments . . . 86

4.4.1 Depth Evaluation on the KITTI Dataset . . . 86
4.4.2 Ablation Study . . . 89
4.4.3 Depth Evaluation on the Make3D Dataset . . . 91
4.4.4 Limitations . . . 93
4.5 Conclusion . . . 94

5 Absolute Distance Prediction Based on Deep Learning Object Detection and Monocular Depth Estimation Models 97
5.1 Introduction . . . 98
5.2 Related Work . . . 100
5.2.1 Object Detection . . . 100
5.2.2 Absolute Distance Prediction . . . 101
5.2.3 Depth Prediction . . . 103
5.2.3.1 Supervised Depth Estimation . . . 103
5.2.3.2 Unsupervised Depth Estimation . . . 104
5.3 Methodology . . . 104
5.3.1 Absolute Distance Prediction . . . 108
5.4 Experiments . . . 110
5.4.1 Datasets . . . 111
5.4.2 Evaluation . . . 112
5.5 Conclusion . . . 115

6 Conclusion 117
6.1 Thesis Highlights . . . 118
6.2 Limitations . . . 119
6.3 Future Research Lines . . . 120

Bibliography 123

List of Figures

3.1 Evaluation trend of DE approaches divided into three sections: traditional methods, hand-crafting and ML methods, and DL methods. . . 35
3.2 (**Top**) Non-rectified left and right images, and (**down**) red–cyan anaglyph from stereo pair of rectified stereo images. 39
3.3 Main network structure for MDE (Masoumian et al., 2023). This network contains two sub-networks: DepthNet for predicting the depth map and PoseNet for estimating the camera pose. 41
3.4 Data input/output structure of mono-sequence models. Single image input and single image output. 47
3.5 Developed network by Shu et al. (Shu et al., 2020). 48
3.6 Data input/output structure of stereo sequence models. Stereo pairs of images as an input and single image output. 49
3.7 Developed network by Goldman et al. (Goldman, Hassner, and Avidan, 2019). 49
3.8 Data input/output structure of sequence-to-sequence models. Sequence of images as an input and sequence of images as an output. 50
3.9 Developed network by Kumar et al. (CS Kumar, Bhandarkar, and Prasad, 2018). 50
3.10 Developed network structure by Eigen et al. (**5**). 52
3.11 Developed geometric neural network by Qi et al. (Qi et al., 2018). . . 53
3.12 Developed network by Zhou et al. (Zhou et al., 2017). 54

3.13 Developed network by Masoumian et al. (Masoumian et al., 2023). . . 55
3.14 Components/inputs of the developed semi-supervised loss function by Kuznietsov et al. (Kuznietsov, Stuckler, and Leibe, 2017). 56
3.15 Components/inputs of developed semi-supervised loss function by Luo et al. (Luo et al., 2018). 57
3.16 Developed network by Cho et al. (Cho et al., 2019). 58

4.1 Overview of our MDE model based on GCN. 76
4.2 An illustration of the proposed GCN module containing two hidden layers. 79
4.3 Overview of DepthNet network architecture. 81
4.4 Schematic illustration of the whole framework. 83
4.5 Comparison of disparity results on KITTI dataset. (Col.1) original input images and the depth resulted with (Col.2) Monodepth2 (Godard et al., 2019), (Col.3) FeatDepth (Shu et al., 2020) and (Col.4) the proposed GCNDepth model. 90
4.6 Comparison of disparity results on Make3D dataset. (Col.1) original input images and the depth resulted with (Col.2) Monodepth2 (Godard et al., 2019), (Col.3) FeatDepth (Shu et al., 2020) and (Col.4) the proposed GCNDepth model. 93
4.7 Two examples of low-quality predicted depths. 94

5.1 An illustration of the overall framework. 105
5.2 Overview of DepthNet network architecture. 106
5.3 Overview of YOLOv5 network architecture. 108
5.4 Perform the relation between ABS and REV. 111
5.5 Visual process of the whole network. 113

List of Tables

3.1 Comprehensive to the related recent surveys in MDE in terms of six parameters; "TM": Training Manner, "ACC": Accuracy, "CT": Computational Time, "RQ": Resolution Quality, "RTI": Real-time Inference, "TRAN": Transferability, "IDS": Input Data Shapes. 37
3.2 Comprehensive information about the quantitative results of the SL, SSL, and UL algorithms investigated on the KITTI dataset. 42
3.3 Comprehensive information about the quantitative results of the DL algorithms investigated on the NYU-V2 dataset. 43
3.4 Comprehensive information about the quantitative results of the DL algorithms investigated on the Make3D dataset. 44
3.5 A summary of DE public datasets. 45
3.6 Comprehensive information about the applied procedures in the DL of MDE. 59
3.7 Comparison of complex and lightweight models based on the NYUDv2 dataset. 61

4.1 The network architecture of depth encoder. **K** is the number of block repetition, **S** the stride, **Chn** the number of output channel, **Input** corresponds to the input channel of each layer. 80

4.2 The network architecture of depth decoder. **K** is the kernel size, **S** the stride, **Chn** the number of output channels, **Input** corresponds to the input channel of each layer and ↑ represents upsampling by 2x. 82

4.3 The network architecture of pose decoder. **K** is the kernel size, **S** the stride, **Chn** the number of output channel and **Input** corresponds to the input channel of each layer. 83

4.4 Comparison of different methods on KITTI dataset. The best results are bolded in blue and the second-best results are bolded in red color. 88

4.5 Performance of our model on KITTI public benchmark. 89

4.6 Ablation results for different components. **SS** represents the single scale GCN and **MS** represent the multi-scale GCN. 91

4.7 Maked3D results. Type **D** represents depth supervision methods and type **M** represents self-supervised mono supervision. 92

5.1 Estimated distance vs. absolute distance. Note the objects are counted in the tested images from left to right. 113

List of Abbreviations

Abs Rel	**A**bsolute and **R**elative **E**rror
Adam	**A**daptive **M**oment **E**stimation **O**ptimization **M**ethod
AI	**A**rtificial **I**ntelligence
ANN	**A**rtificial **N**eural **N**etwork
BDE	**B**inocular **D**epth **E**stimation
CNN	**C**onvolutional **N**eural **N**etwork
DE	**D**epth **E**stimation
DI	**D**epth **I**nformation
DL	**D**eep **L**earning
FCN	**F**ully **C**onvolutional **N**etwork
GCN	**G**raph **C**onvolutional **N**etwork
GTD	**G**round **T**ruth **D**epth
LiDAR	**Li**ght **D**etection **A**nd **R**anging
MDE	**M**onocular **D**epth **E**stimation
ML	**M**achine **L**earning
MLF	**M**ulti **L**oss **F**unction
MSE	**M**ean **S**quared **E**rror
Rel	**R**elative **E**rror
ReLU	**Re**ctified **L**inear **U**nit
RMSE	**R**oot **M**ean **S**quare **E**rror
RNN	**R**ecurrent **N**eural **N**etwork

SOTA	**S**tate **-O**f-**T**he-**A**rt
SSIM	**S**tructural **Sim**ilarity
SL	**S**upervised **L**earning
SSL	**S**emi **S**upervised **L**earning
SVM	**S**upport **V**ector **M**achine
TOF	**T**ime **O**f **F**light
UL	**U**nsupervised **L**earning
VO	**V**isual **O**dometry

Chapter 1

Introduction

In the first chapter, the motivation, approach, and contributions of this thesis are presented. It outlines the key objectives of this thesis, emphasizing the importance of advancing DE through DL techniques. Additionally, notable publications resulting from this thesis are highlighted, serving as evidence of its scientific quality. The chapter concludes by providing an overview of the thesis organization, setting the stage for the subsequent chapters.

1.1 Motivation

Depth estimation has become a fundamental problem in computer vision, with its applications expanding across various domains such as robotics, autonomous driving, virtual reality, and more. MDE aims to predict a depth map from a single RGB image, providing crucial 3D information about the scene and objects captured in the image (Saxena et al., 2023).

Several techniques have been developed for depth estimation in computer vision, such as Stereo Matching, Structured Light, Time-of-Flight (ToF), LiDAR, Monocular cameras, and Depth from Focus/Defocus (Hu, Xu, and Yang, 2014; Maximov, Galim, and Leal-Taixé, 2020; Daneshmand et al., 2018). It's important to note that the accuracy of these techniques can vary depending on factors such as lighting conditions (Sun, Zheng, and Shum, 2003), scene complexity (Saudabayev and Varol, 2015), and the availability of appropriate calibration and post-processing techniques (Zhang, Wang, and Chan, 2015). The choice of one of these techniques depends on the specific application requirements and constraints.

Regarding cost and computational resources, among all techniques mentioned above, monocular depth estimation (MDE) can be less expensive compared to hardware-intensive techniques like LiDAR or structured light, as it mainly relies on a single camera (Eigen, Puhrsch, and Fergus, 2014). However, the cost also includes the computational resources required for training and running deep neural networks, which can vary based on the complexity and scale of the network architecture (He et al., 2016). MDE also is inherently challenging due to losing depth information (DI) during the projection from three-dimensional space to the two-dimensional image plane (Masoumian et al., 2022).

Deep learning (DL) techniques have emerged as a powerful tool for addressing the complexities of MDE and have demonstrated remarkable progress in recent years. The advantages of using deep learning for MDE are numerous and contribute to the

improved performance and state-of-the-art (SOTA) results in this field (Zhang et al., 2023).

Firstly, DL models enable end-to-end learning, allowing the network to learn directly from input RGB images and predict corresponding depth maps. This eliminates the need for manual feature extraction and explicit algorithm design, making the process more automated and efficient. By training the model on large-scale datasets, DL models can effectively learn complex representations, capturing low-level visual features, such as edges and corners, and high-level semantic features, such as object boundaries and textures (Kaur et al., 2023). These learned representations aid in accurately estimating depth information, enhancing the overall performance of the system.

DL models also excel at handling the inherent ambiguity in MDE. Through their ability to learn from vast amounts of data, DL models can capture statistical regularities and exploit contextual cues to infer depth information, even in challenging scenarios such as occlusions or textureless regions (Zhou et al., 2023). DL models can effectively resolve depth ambiguities and produce accurate depth maps by leveraging global and local image contexts.

Another advantage of DL models is their ability to learn scale-invariant representations, which is crucial for depth estimation (DE) tasks. DL models can generalize well to different scene scales and handle objects at varying distances from the camera (Li et al., 2023). This adaptability allows DL-based MDE systems to estimate depth consistently across diverse scenes, improving the robustness and reliability of the results.

Furthermore, DL models exhibit a remarkable generalization capability. Once trained on diverse datasets, these models can capture and exploit common patterns and structures, enabling them to estimate depth accurately in novel scenes. This generalization capability makes DL-based MDE systems versatile and applicable in real-world scenarios (Chen et al., 2023).

The ongoing progress in the field of DL also contributes to its advantages in MDE. Researchers continuously explore innovative approaches, such as convolutional neural networks (CNNs) (Cong and Zhou, 2023), recurrent neural networks (RNNs) (Salehinejad et al., 2017), and attention mechanisms (Niu, Zhong, and Yu, 2021), to enhance the performance of MDE. These advancements in network architectures, loss functions, and training techniques result in incremental improvements in DE accuracy and push the boundaries of what is achievable in the field.

However, MDE models require large-scale training datasets, typically consisting of stereo or RGB-D images with corresponding depth maps (Liu et al., 2015; Guo et al., 2018). Acquiring such datasets can be challenging and time-consuming. While synthetic data generation techniques can assist in this regard, obtaining real-world ground truth depth maps is still complex (Pillai, Ambruş, and Gaidon, 2019). Self-supervised DL models offer a potential solution to the problem of acquiring labeled training data for depth estimation (Zhou et al., 2017). By leveraging unlabeled data and formulating proxy tasks, these models can learn to estimate depth without requiring extensive manual annotations (Godard, Mac Aodha, and Brostow, 2017). In addition, self-supervised learning aims to capture generic depth cues, which can lead to better generalization and robustness (Kuznietsov, Stuckler, and Leibe, 2017). The models learn to extract meaningful features applicable across different scenes and imaging conditions (Mahjourian, Wicke, and Angelova, 2018). This can be advantageous when deploying the model in real-world scenarios where the training and testing conditions may differ. In turn, self-supervised learning still has limitations. The performance of self-supervised depth estimation models may not match the accuracy achieved by supervised approaches trained with large-scale labeled data. The models can struggle in complex or challenging scenes where direct supervision is beneficial, such as in textureless or occluded regions (Yuan et al., 2016).

Indeed, ongoing research and scope exist for further advancements in self-supervised

MDE techniques. Despite the progress made in recent years, there are still several challenges and areas for improvement (Saikia et al., 2019). For instance, enhancing the accuracy of MDE remains an active research area (Zhou et al., 2017). Techniques that can better handle occlusions and textureless regions and improve depth estimation in large-scale scenes would be beneficial—exploring novel network architectures and loss functions (Masoumian et al., 2023). In addition, real-time MDE is crucial for applications like robotics, augmented reality, and autonomous vehicles. Research efforts can focus on developing efficient network architectures, optimization techniques, and hardware acceleration methods to enable real-time depth estimation on resource-constrained platforms (Ranftl, Bochkovskiy, and Koltun, 2021). By addressing such research challenges, further advancements can be made in MDE techniques based on self-learning, leading to more accurate, robust, and practical solutions for depth estimation using a single camera (Zou, Luo, and Huang, 2018).

1.2 Approach

In this doctoral thesis, our extensive research focuses on investigating novel DL architectures and loss functions to enhance the accuracy and robustness of MDE in real-world scenarios. Our primary objective is to address the existing challenges and limitations in MDE and contribute to advancing computer vision and its applications.

To achieve our objectives, we explore the utilization of two key components: autoencoders and Graph Convolutional Networks (GCNs). As unsupervised learning models, autoencoders allow us to extract meaningful features and reduce noise by reconstructing input from compressed latent representations. By incorporating autoencoders into the MDE framework, we aim to improve the quality of reconstructions.

In addition, we leverage the power of GCNs, which excel at capturing relational dependencies within data by operating on graph structures. By considering the inherent geometric relationships present in multi-view data, we employ GCNs to model

interdependencies among views, resulting in more accurate and reliable MDE.

Furthermore, in graph connections, we recognize the importance of causality in capturing the flow of information within a graph. Causality plays a crucial role in accurately modeling the underlying structure of data and capturing relational dependencies. In the case of GCNs, it ensures that information propagation follows the underlying cause-and-effect relationships.

To incorporate causality into graph connections, we consider various techniques. This includes using directed graphs, where edges have specific directions associated with them, allowing the GCNs to capture the causality in the graph. Additionally, temporal information or sequential ordering can be employed to establish causal relationships, with temporal graphs capturing the evolution of a system over time.

Through our extensive research and evaluation of publicly available datasets, we compare our proposed approaches against existing methods widely used in MDE. We assess the performance using quantitative and qualitative metrics, demonstrating significant improvements in the reconstructed scenes' accuracy, precision, recall, and visual inspection.

Once the depth map is obtained, the depth values can be converted to metric units (e.g., meters) using appropriate calibration and camera parameters. This allows for estimating the physical distance between the camera and the objects in the scene "Absolute distance". Estimating the absolute distance of objects aids in scene understanding. Knowing the distance of objects makes it possible to analyze their spatial relationships, identify occlusions, and infer scene geometry for tasks such as scene reconstruction and semantic understanding. By integrating the developed MDE and object detection techniques, i.e., YOLO-v5 (Jocher et al., 2022), we will estimate the absolute distance of the objects in a scene.

Overall, this doctoral thesis contributes to the ongoing efforts to develop reliable and efficient solutions for MDE and absolute distance estimation. By incorporating

novel DL architectures and loss functions and considering causality in graph connections, our research aims to advance the field of computer vision. The potential applications of our work extend to domains such as robotics, scene understanding, and 3D reconstruction. Furthermore, our investigations into these areas provide valuable insights into DL and pave the way for future advancements in related fields.

1.3 Contributions and Publications

This thesis focuses on estimating depth maps from monocular images, considering both the object present in a scene and the complete scene. The thesis is divided into two research lines.

The first research line involves an extensive investigation of MDE, including exploring datasets and evaluation metrics. Existing techniques and methodologies in the field are thoroughly analyzed and examined.

In the second research line, a novel model for unsupervised MDE is developed. This model combines GCNs with an autoencoder architecture. The GCN component captures relational dependencies and inherent geometric relationships within the multi-view data. At the same time, the autoencoder reconstructs the input from compressed latent representations to learn efficient data representations and reduce noise.

To further enhance the DE, the proposed model is integrated with object detection techniques. By combining the unsupervised MDE model with object detection, the absolute distance of objects in the scene is computed. This integration improves the accuracy and reliability of DE by considering both the scene context and the objects within it.

This research provides a comprehensive understanding of MDE and the evaluation of existing methodologies. The proposed model, combining GCNs and autoencoders, offers a novel approach to unsupervised MDE. By incorporating object detection, the estimation process is enhanced, enabling the computation of absolute distances for

objects. These contributions have the potential to advance the field of MDE, with implications in domains such as robotics, augmented reality, and 3D reconstruction.

The results of this research line have been published in the following papers:

- ***Armin Masoumian***, Hatem A. Rashwan, Julián Cristiano, Salman M Asif and Domenec Puig, "**Monocular depth estimation using deep learning: A review**", Sensors 22.14 (2022): 5353

- ***Armin Masoumian***, Hatem A Rashwan, Saddam Abdulwahab, Julián Cristiano, Salman M Asif and Domenec Puig, "**GCNDepth: Self-supervised monocular depth estimation based on graph convolutional network**", Neurocomputing 517 (2023): 81-92

- ***Armin Masoumian***, David GF Marei, Saddam Abdulwahab, Julián Cristiano, Domenec Puig and Hatem A. Rashwan, "**Absolute Distance Prediction Based on Deep Learning Object Detection and Monocular Depth Estimation Models**", CCIA Conference. 2021

- Saddam Abdulwahab, Hatem A. Rashwan, Najwa Sharaf, ***Armin Masoumian*** and Domenec Puig:, "**Promising Depth Map Prediction Method from a Single Image Based on Conditional Generative Adversarial Network**", CCIA 2021

- Saddam Abdulwahab, Hatem A Rashwan, Miguel Angel Garcia, ***Armin Masoumian*** and Domenec Puig, "**Monocular depth map estimation based on a multi-scale deep architecture and curvilinear saliency feature boosting**", Neural Computing and Applications 34.19 (2022): 16423-16440

1.4 Thesis Organization

The thesis contains five chapters. Below, we briefly describe the work done in each chapter:

- Chapter 1: Introduction

  In this chapter, we explore DE within the context of MDE systems. The chapter begins by elucidating the underlying motivation behind the thesis and delving into the primary contributions to enhancing MDE systems.

- Chapter 2: Background

  This chapter comprehensively overviews DE and MDE and their diverse types. Furthermore, we delve into the fundamental concepts of Graph Convolutional Networks (GCNs) and autoencoders, emphasizing their importance in tackling DE tasks, especially for the self-supervised MDE models.

- Chapter 3: Monocular depth estimation using deep learning: A review

  In this chapter, we describe the background of various aspects of DE from monocular images. Also, review the methods and algorithms used. It also introduces the datasets of MDE and evaluation metrics used in the thesis.

- Chapter 4: GCNDepth: Self-supervised monocular depth estimation based on graph convolutional network

  This chapter introduces our innovative MDE approach based on a graph convolutional network (GCN). We present the architecture and methodology of our proposed model, highlighting its unique features and advantages. Furthermore, we conduct a comprehensive comparative analysis, evaluating the performance of our GCN-based model against SOTA techniques in terms of accuracy, robustness, and computational efficiency.

- Chapter 5: Absolute Distance Prediction Based on Deep Learning Object Detection and Monocular Depth Estimation Models

  In this chapter, we introduce a novel approach for computing the absolute distance of objects in a scene, leveraging our newly developed MDE method. Our approach combines a GCN-based MDE model with YOLOv5 object detection, enabling accurate estimation of object distances. We provide a detailed description of the architecture and methodology of our approach, emphasizing its efficacy in accurately estimating the distances of objects in the scene.

- Chapter 6: Conclusion

  In this chapter, We provide a comprehensive summary of the main concluding remarks, highlighting the contributions and implications of the research. Additionally, we outline potential lines of future research, shedding light on the unexplored areas and opportunities for further advancements in the field.

Chapter 2

Background

This chapter provides an overview of DE by discussing the background, various DE methods, and the relevance of autoencoders. It explores the advantages and disadvantages of different DE methods, emphasizing the importance of datasets and evaluation metrics for assessing performance. Additionally, it highlights the role of autoencoders in reconstructing depth maps and their potential for unsupervised deep learning. This chapter establishes a foundation for the subsequent chapters' discussions on DE.

2.1 Introduction

Significant advancements in computational photography have paved the way for innovative capabilities in image processing (Petrou and Petrou, 2010). The field of computer vision has witnessed numerous studies, including DE, which is a well-established task involving the prediction of depth from one or more two-dimensional (2D) images (Lee et al., 2020). DE aims to estimate the depth of individual pixels in an image using pre-trained models. Within machine perception, it is essential to recognize scene shapes from images and understand the independence of image appearance. DE holds immense potential for diverse applications, such as robotics, robot-assisted surgery, computer graphics, and computational photography (Poggi et al., 2020; Masoumian et al., 2020a).

To accomplish the DE task, an RGB image and its corresponding depth image are required. The depth image typically contains information regarding the distance between objects in the image and the camera viewpoint (Yoo et al., 2020). Researchers worldwide have extensively investigated computer-based approaches for DE, resulting in a vibrant research field. Traditional methods have heavily relied on stereo vision techniques for depth determination. However, recent advancements in DL models have demonstrated remarkable efficacy in DE across various applications.

This chapter provides a comprehensive exploration of DE techniques, encompassing various types of DE, their advantages and disadvantages, and a comparison between traditional DE, ML-based DE, and DL-based DE which plays a pivotal role in computer vision, allowing machines to perceive the spatial structure of a scene based on two-dimensional images or videos. Accurate depth estimation enables machines to discern object distances, comprehend the three-dimensional layout of a scene, and enhance their perception and decision-making capabilities.

The following sections delve into the intricacies of DE, beginning with an examination of different types of DE techniques. These include stereo-based DE, which leverages disparity between stereo image pairs to estimate depth, and structured light-based DE, which employs projected patterns to infer depth information. Other techniques, such as depth from focus and depth from motion, are also explored, each with its own set of advantages and disadvantages.

Furthermore, this chapter will delve into two specific approaches used for DE: GCNs and autoencoders (Bank, Koenigstein, and Giryes, 2020). GCNs have garnered significant attention due to their capability to capture pixel relationships and dependencies, thereby improving DE accuracy. The principles of GCNs will be explored, along with their application in DE tasks. Additionally, the utilization of autoencoders, a type of neural network architecture, for DE will be investigated. Autoencoders have demonstrated promising results in learning efficient representations of DI from input images. The working mechanisms of autoencoders will be examined, and their advantages and limitations in the context of DE will be discussed. By exploring these specific approaches, readers will gain insights into the SOTA techniques that have contributed to advancements in depth estimation.

2.2 Depth Estimation

DE is a crucial task in computer vision that involves predicting the distance or depth of objects in a scene from a given input image or video. It is an essential problem that enables machines to understand the three-dimensional structure of the environment and interact with it more effectively. DE has many applications, including robotics, augmented and virtual reality, autonomous driving, and more (Eigen, Puhrsch, and Fergus, 2014).

One of the main challenges in DE is the need for a comprehensive understanding of the complex interactions between lighting, shading, and perspective in a scene.

This problem is further complicated because most cameras are designed to capture two-dimensional images, which lack DI (Liu et al., 2015).

Recent advances in DL and computer vision have led to significant progress in MDE, where depth is predicted from a single 2D image. SOTA methods use deep neural networks to learn complex mappings between image features and depth values, achieving impressive results on various datasets and benchmarks (Eigen, Puhrsch, and Fergus, 2014).

The importance of DE lies in its ability to provide machines with a richer understanding of the environment, allowing them to navigate and interact with it more effectively. For example, DE is crucial for autonomous vehicles to perceive their surroundings and make decisions based on the distance and position of objects. It also plays a critical role in virtual and augmented reality applications, enabling users to interact with virtual objects and environments as if they were real.

In summary, DE is a fundamental problem in computer vision with many practical applications, and it continues to be an active area of research. The ongoing development of new algorithms and techniques for DE is expected to improve the performance of existing applications further and enable new ones.

2.2.1 Different Approaches of Depth Estimation

DE is an essential task in computer vision as it enables machines to understand the three-dimensional structure of their surroundings. Researchers have developed three primary approaches for DE, including traditional methods, machine learning (ML) methods, and deep learning (DL) methods (Masoumian et al., 2022).

Traditional methods rely on handcrafted features and mathematical models to estimate depth from images. These methods have been used for many years and are computationally efficient. However, they can be limited in complex environments, often requiring strong assumptions about the scene's structure.

ML methods involve training models on large datasets to learn patterns and relationships between image features and depth values. These methods have shown promising results in recent years and can handle more complex scenes than traditional methods. However, they still require some hand-engineered features, their performance heavily depends on the quality and quantity of the training data.

DL methods use neural networks to learn complex mappings between image features and depth values. These methods can automatically learn high-level representations of the scene and have achieved SOTA results in various DE tasks. However, they require large amounts of annotated training data and extensive computational resources, making them more computationally expensive than traditional and ML methods.

In summary, each of these approaches has its own strengths and limitations, and researchers continue to explore new techniques to improve the accuracy and efficiency of DE in computer vision. The choice of approach will depend on the specific task and available resources.

2.2.1.1 Traditional Methods

Traditional methods of DE usually involve handcrafted features and heuristics, rather than DL techniques. Some examples of traditional methods for DE include:

1. Structure from motion: Structure from motion is a traditional method that involves estimating the 3D structure of a scene by analyzing the motion of objects in a sequence of 2D images. By tracking the motion of objects over time, structure from motion can estimate the scene's depth.

2. Stereo matching: Stereo matching is a classic technique for DE that involves comparing the disparities between corresponding pixels in a pair of stereo images. By triangulating the disparities, stereo matching can estimate the depth of objects in the scene.

3. Photometric stereo: Photometric stereo is a method that estimates the surface normals of objects in a scene by analyzing the lighting and shading variations in multiple images of the same scene captured under different lighting conditions. By integrating the surface normals, photometric stereo can estimate the depth of the scene.

4. Shape from Shading: Shape from shading is a technique that estimates the 3D shape of an object from a single image by analyzing the variations in lighting and shading. By assuming a known lighting model and reflectance properties, shape from shading can estimate the object's depth.

While traditional methods have been helpful in specific applications, they typically require manual tuning of parameters and may not generalize well to complex real-world scenes. In recent years, DL techniques have shown remarkable progress in DE, largely replacing traditional methods with SOTA approaches (Haque et al., 2016). Some of the advantages and disadvantages of traditional DE include:

Advantages:

- Computational efficiency: Traditional DE methods are typically computationally efficient, requiring relatively little computational power compared to ML and DL techniques. This makes them useful in applications with limited computational resources (e.g., real-time video processing on mobile devices).

- Robustness to lighting conditions: Traditional DE methods, such as stereo DE, are generally robust to changes in lighting conditions, making them useful in outdoor environments and other situations where lighting is variable (Khoshelham and Elberink, 2012).

- Absolute DI: Traditional DE techniques can provide absolute DI, which can be useful in applications where an accurate and consistent scale of depth is essential.

Disadvantages:

- Limited accuracy: Traditional DE methods often rely on assumptions about the scene, such as the availability of known feature points, and can be limited in their accuracy as a result (Hosni et al., 2012).
- Limited adaptability: Traditional DE methods are often designed for specific scenarios and may not generalize well to new or more complex situations.
- Requires manual tuning of parameters: Traditional DE methods often rely on manually tuning parameters and assumptions, which can be time-consuming and require expert knowledge.
- Limited applicability: Traditional DE methods may not be applicable in scenarios with limited availability of information, such as in occluded scenes or scenes with homogeneous textures.

Traditional DE can be useful in specific scenarios where computational efficiency or robustness to lighting conditions is essential. Still, its accuracy and adaptability limitations should be considered when selecting a DE technique for a specific application.

2.2.1.2 Machine Learning Methods

ML methods for DE involve training a model to predict depth from input images using supervised learning (SL) techniques. The model is trained on a large dataset of images with corresponding depth maps, and the goal is to learn a mapping between image features and depth values.

Some of the most commonly used ML methods for DE include:

1. Random forests: An ML algorithm that uses an ensemble of decision trees to predict depth from input images. Random forests have been shown to achieve high accuracy in DE tasks.

2. Support vector machines (SVMs): An SL algorithm that separates data into different classes based on their features. SVMs have been used for DE tasks, with promising results.

3. Conditional random fields (CRFs): A probabilistic graphical model that can be used for image segmentation and DE tasks. CRFs have been shown to achieve SOTA results in DE, especially when combined with DL techniques.

Overall, ML methods for DE have been shown to achieve high levels of accuracy and can be trained on large datasets of labeled data. However, they may require significant computational resources and expertise to train and deploy, and may not be as adaptable to new or complex environments as DL methods (Godard, Mac Aodha, and Brostow, 2017). Some of the main advantages and disadvantages of ML DE are:

Advantages:

- High accuracy: ML DE techniques can achieve high levels of accuracy, surpassing traditional DE methods in many cases. This is due to their ability to learn complex features and patterns from large amounts of data (Eigen, Puhrsch, and Fergus, 2014).
- Adaptability: ML DE methods can be trained on a wide range of environments and scenarios, making them adaptable to different applications and situations (Liu et al., 2015).
- Handle complex scenes: ML DE techniques can handle complex scenes, such as occluded scenes or scenes with homogeneous textures, where traditional DE methods may fail.
- Learn from data: ML DE techniques can learn from data, which means they do not rely on handcrafted features or heuristics and can automatically adapt to different environments and scenarios (Laina et al., 2016).

Disadvantages:

- Requires large amounts of data: ML DE techniques require large amounts of labeled data to achieve high accuracy, which can be challenging to obtain in certain scenarios.

- Requires high computational resources: ML DE techniques can be computationally expensive, requiring high computational resources and specialized hardware to train and run models (Yin et al., 2019).

- Sensitive to data biases: ML DE models can be sensitive to data biases, which may not generalize well to new or different scenarios.

- Limited interpretability: ML DE models can be difficult to interpret, making it challenging to understand how they arrived at their predictions.

Overall, ML DE can provide high levels of accuracy and adaptability. Still, its limitations in terms of data requirements, computational resources, and interpretability should be taken into consideration when selecting a DE technique for a specific application (Chen et al., 2017).

2.2.1.3 Deep Learning Methods

DL methods for DE involve training deep neural networks to predict the depth of a scene from a given input image or video. These methods use Convolutional neural networks (CNNs) to learn complex mappings between image features and depth values.

There are several types of DL methods for DE, including:

1. Single-image DE: These methods use a single image as input to predict the depth of a scene. They often rely on encoder-decoder architectures with skip connections to handle multi-scale features and refine the output depth map.

2. Stereo DE: These methods use two or more images of a scene captured from different viewpoints to estimate depth. They often use a Siamese network architecture

to extract features from both images and compute the disparity map, which is then converted to depth.

3. Monocular depth and pose estimation: These methods simultaneously predict the depth and camera pose from a single image, which can be useful for applications such as augmented reality and robotics. They often use a multi-task learning approach with a shared encoder and separate decoders for depth and pose estimation.

DL methods for DE have shown significant improvements over traditional and ML methods in terms of accuracy and generalization to new environments. However, they also require large amounts of labeled data and high computational resources for training and inference (Ming et al., 2021). Some of the main advantages and disadvantages of DL DE are:

Advantages:

- High accuracy: DL DE techniques can achieve high levels of accuracy, often surpassing traditional and ML DE methods.
- Learn from large amounts of data: DL DE techniques can learn from large amounts of data, which means they do not rely on handcrafted features or heuristics, and can automatically adapt to different environments and scenarios.
- Handle complex scenes: DL DE techniques can handle complex scenes, such as occluded scenes or scenes with homogeneous textures, where traditional and ML DE methods may fail.
- Provide real-time performance: DL DE techniques can provide real-time performance, which can be useful in applications where speed is important.

Disadvantages:

- Requires even larger amounts of data: DL DE techniques require even larger amounts of labeled data than ML DE techniques to achieve high accuracy, which can be challenging to obtain in certain scenarios.

- Requires high computational resources: DL DE techniques can be computationally expensive, requiring high computational resources and specialized hardware to train and run models.
- Sensitive to data biases: DL DE models can be sensitive to data biases, which means they may not generalize well to new or different scenarios.
- Limited interpretability: DL DE models can be difficult to interpret, making it challenging to understand how they arrived at their predictions.

Overall, DL DE can provide high levels of accuracy, adaptability, and real-time performance, but its limitations in terms of data requirements, computational resources, and interpretability should be taken into consideration when selecting a DE technique for a specific application.

2.2.2 ML Vs. DL Depth Estimation

ML and DL are both techniques for training models to estimate depth from input images. However, there are some differences between ML and DL DE techniques:

Model architecture: ML DE techniques typically use simpler model architectures, such as decision trees, random forests, or support vector machines, while DL DE techniques use deep neural networks with multiple layers.

Feature engineering: ML DE techniques require handcrafted features or heuristics to be extracted from input images, which are then used to train models. In contrast, DL DE techniques can learn features automatically from raw input images, which means they do not require manual feature engineering.

Data requirements: ML DE techniques require labeled training data to be used for model training, but the amount of data needed is typically smaller than for DL DE techniques. DL DE techniques require large amounts of labeled data to be used for model training, often in the millions of images.

Computational requirements: DL DE techniques are computationally more demanding than ML DE techniques due to the larger and more complex model architectures used, which can require specialized hardware for efficient training and inference.

Accuracy: DL DE techniques typically achieve higher levels of accuracy than ML DE techniques due to their ability to learn more complex and hierarchical features from large amounts of data.

Overall, while both ML and DL DE techniques are effective at estimating depth from images, DL techniques are generally more powerful due to their ability to learn complex features automatically from large amounts of data. However, they also require more computational resources and larger amounts of labeled data, which can be a limitation in some applications (Masoumian et al., 2022).

2.2.3 Different Types of Depth Estimation

There are several types of DE techniques used in computer vision, including:

2.2.3.1 Monocular Depth Estimation

MDE refers to the task of predicting the depth of a scene from a single 2D image. This is a challenging problem since DI is lost in the projection from 3D to 2D. MDE typically involves using deep neural networks to learn complex mappings between image features and depth values (Masoumian et al., 2022). Here are some of the main advantages and disadvantages:

Advantages:

- Single image input: MDE can approximate depth from a single image, which makes it more convenient than other DE techniques that require multiple images or specialized equipment such as LiDAR or multiple cameras.

- Cost-effective: Since MDE can be performed using a single camera, it is a more cost-effective option compared to other DE techniques that require expensive equipment.
- Versatile: MDE can be used in a wide range of applications, such as autonomous vehicles, robotics, and augmented reality, where accurate depth perception is important.
- Flexible: MDE can be applied to different environments and scenes, making it a versatile approach.

Disadvantages:

- Limited accuracy: MDE is not as accurate as other DE techniques, such as stereo DE or LiDAR, because it relies on projecting a 3D scene onto a 2D image.
- Lack of scale: MDE provides only relative DI, which means that it cannot provide an absolute scale for depth. This is in contrast to LiDAR, which provides DI with an absolute scale.
- Data requirements: MDE relies on deep neural networks, which require large amounts of training data to achieve high accuracy. This can be a challenge when training data is limited.
- Lighting conditions: MDE can be sensitive to changes in lighting conditions, which can lead to errors in DE.

In summary, MDE is a useful technique for estimating depth from a single image, but its accuracy and limitations should be taken into account when choosing a DE method for a specific application.

2.2.3.2 Stereo Depth Estimation

Stereo DE involves estimating depth from a pair of images captured by two cameras placed at different positions. By comparing the differences between the two images, stereo DE can determine the distance to objects in the scene (Masoumian et al., 2022). Here are some of the main advantages and disadvantages:

Advantages:

- High accuracy: Stereo DE techniques can achieve high levels of accuracy, often surpassing MDE techniques.
- Robustness: Stereo DE is robust to many of the limitations of MDE, such as occlusions, textureless regions, and lighting changes.
- Handle large-scale scenes: Stereo DE can handle large-scale scenes, making it suitable for applications such as autonomous driving or aerial mapping.
- Provide real-time performance: Stereo DE can provide real-time performance, which can be useful in applications where speed is important.

Disadvantages:

- Requires stereo images: Stereo DE requires the acquisition of stereo images, which can be more challenging than obtaining a single image.
- Requires calibration: Stereo DE requires the camera parameters to be accurately calibrated, which can be time-consuming and challenging in practice.
- Limited field of view: Stereo DE is limited by the field of view of the cameras, which means it may not be able to estimate depth for objects outside the field of view.
- Suffer from errors due to misalignments: Stereo DE can suffer from errors due to misalignments between the stereo images, which can occur due to camera motion or changes in the scene.

In summary, the stereo DE can yield accurate and reliable results in real-time. However, its requirement for significant data and calibration should be considered when deciding on a suitable DE technique for a particular application.

2.2.3.3 Structured Light Depth Estimation

Structured light DE is a technique that uses a projector to project a pattern onto a scene, which is then captured by a camera. By analyzing the distortions in the projected pattern, structured light DE can determine the distance to objects in the scene (Wang et al., 2012). Here are some of the main advantages and disadvantages:

Advantages:

- High accuracy: Structured light DE techniques can achieve high levels of accuracy, often surpassing MDE techniques.
- High speed: Structured light DE can provide fast depth estimates, making it suitable for real-time applications.
- Handle large-scale scenes: Structured light DE can handle large-scale scenes, making it suitable for applications such as 3D scanning and robotics.
- Robustness: Structured light DE is robust to many of the limitations of MDE, such as occlusions, textureless regions, and lighting changes.

Disadvantages:

- Limited range: Structured light DE is limited by the range of the projector and camera, which can be a limitation in some applications.
- Limited lighting conditions: Structured light DE requires a controlled lighting environment, which can be challenging in some situations.
- Require calibration: Structured light DE may require calibration of the projector and camera, which can be time-consuming and challenging in practice.

- Limited field of view: Structured light DE is limited by the field of view of the projector and camera, which means it may not be able to estimate depth for objects outside the field of view.

Overall, structured light DE can provide high levels of accuracy, speed, and robustness, but its limitations in terms of range, lighting conditions, and calibration should be taken into consideration when selecting a DE technique for a specific application.

2.2.3.4 Time-of-Flight (TOF) Depth Estimation

TOF DE uses a special type of camera that emits a short burst of light and measures the time it takes for the light to bounce back from objects in the scene. By measuring the TOF of the light, TOF DE can determine the distance to objects in the scene (Jiménez et al., 2014). Here are some of the main advantages and disadvantages:

Advantages:

- High speed: TOF DE can provide fast depth estimates, making it suitable for real-time applications.
- High accuracy: TOF DE techniques can achieve high levels of accuracy, often surpassing traditional stereo vision techniques.
- Handle large-scale scenes: TOF DE can handle large-scale scenes, making it suitable for applications such as 3D scanning and robotics.
- Robustness: TOF DE is robust to many of the limitations of MDE, such as occlusions, textureless regions, and lighting changes.

Disadvantages:

- Limited range: TOF DE is limited by the range of the light source and sensor, which can be a limitation in some applications.

- Limited lighting conditions: TOF DE requires a controlled lighting environment, which can be challenging in some situations.
- Limited resolution: TOF DE may have a lower resolution compared to stereo vision or structured light DE techniques.
- Suffer from interference: TOF DE can suffer from interference from other light sources or reflective surfaces.

Overall, TOF DE can provide high levels of speed, accuracy, and robustness, but its limitations in terms of range, lighting conditions, and resolution should be taken into consideration when selecting a DE technique for a specific application.

2.2.3.5 LiDAR Depth Estimation

LiDAR DE is similar to TOF DE, but it uses laser beams instead of light. By emitting laser beams and measuring the time it takes for the beams to bounce back from objects in the scene, LiDAR DE can determine the distance to objects in the scene (Yan et al., 2018). Here are some of the main advantages and disadvantages:

Advantages:

- High accuracy: LiDAR DE techniques can achieve very high levels of accuracy, often surpassing other DE techniques.
- Handle large-scale scenes: LiDAR can handle large-scale scenes, making it suitable for applications such as 3D mapping, surveying, and autonomous driving.
- Provide 3D information: LiDAR can provide 3D information about the scene, making it useful for applications that require detailed spatial information.
- Work in any lighting condition: LiDAR can work in any lighting condition, making it robust to changes in ambient lighting.

Disadvantages:

- Expensive: LiDAR sensors are often expensive compared to other DE techniques.
- Limited range: The range of LiDAR is limited compared to other DE techniques, which can be a limitation in some applications.
- Limited resolution: LiDAR may have a lower resolution than other DE techniques.
- Limited field of view: The field of view of LiDAR can be limited compared to other DE techniques, which can limit its use in some applications.

Overall, LiDAR DE can provide high levels of accuracy and robustness, but its limitations in terms of cost, range, resolution, and field of view should be considered when selecting a DE technique for a specific application.

2.3 Graph Convolutional Networks

GCNs have emerged as a powerful technique in computer vision tasks, offering distinct advantages over traditional CNNs in various applications, including DE. GCNs leverage the inherent graph structure present in images to capture spatial relationships and dependencies between pixels, enabling more accurate depth estimation.

Unlike CNNs which operate on regular grids and treat each pixel independently, GCNs explicitly model the underlying graph structure of an image. In the context of DE, the graph can be constructed based on pixel connectivity or image superpixels. By considering the local and global relationships between pixels, GCNs enable more informed depth predictions.

One of the key advantages of GCNs over CNNs in DE is their ability to capture long-range dependencies and global contextual information. CNNs typically have limited receptive fields, which restricts their ability to capture global relationships between distant pixels. In contrast, GCNs can efficiently propagate information across the graph structure, allowing pixels to communicate with their neighbors regardless

of their spatial distance. This capability enables GCNs to leverage global contextual cues, leading to more accurate DE, especially in complex scenes or regions with occlusions.

Another advantage of GCNs lies in their ability to handle irregular or non-uniform data. Traditional CNNs assume regular grids as input, making them less suitable for tasks where the data is inherently graph-structured. In DE, where the spatial relationships between pixels are crucial, GCNs provide a natural framework for incorporating such dependencies. By explicitly modeling the graph structure, GCNs can effectively capture the interplay between pixels and exploit the contextual information encoded in the graph.

Furthermore, GCNs facilitate the incorporation of prior knowledge or external information into the DE process. By encoding additional features or cues into the graph structure, such as semantic information or edge weights, GCNs can enhance the DE accuracy by incorporating higher-level contextual cues. This ability to leverage external information allows GCNs to make more informed and reliable depth predictions, particularly in challenging scenarios.

In summary, GCNs offer distinct advantages over traditional CNNs in DE. They can capture long-range dependencies, leverage global contextual information, handle irregular data, and incorporate prior knowledge. By explicitly modeling the graph structure of images, GCNs enable more accurate and robust DE, especially in complex scenes or regions with occlusions. The utilization of GCNs in DE tasks holds great promise for advancing the field of computer vision and enhancing the performance of depth estimation algorithms in real-world applications.

2.4 Autoencoders

Autoencoder networks, a prevalent DL algorithm within computer vision, have emerged as a powerful tool for a range of tasks, including DE. Comprising two essential components, namely an encoder and a decoder, autoencoders excel at compressing input data into a lower-dimensional latent space and then reconstructing it back to its original form. Unlike other SL methods, autoencoders can also be utilized for self-supervised learning, where the network learns from unlabeled data without relying on explicit ground truth annotations.

In the context of DE, autoencoders offer several advantages. The encoder-decoder architecture allows the network to learn a compact and informative representation of the data, enabling accurate predictions of object depths in images. By leveraging the intrinsic structure and patterns within the data, autoencoders can capture important visual cues that contribute to depth perception. Additionally, the flexibility of autoencoders allows them to adapt to various input data types, making them well-suited for DE tasks across different domains and imaging conditions.

Training an autoencoder for DE typically involves optimizing a loss function that measures the discrepancy between the reconstructed output and the original input. By minimizing this reconstruction loss, the network learns to extract meaningful DI from the input images. The self-supervised nature of autoencoders makes them particularly valuable in scenarios where ground truth depth annotations may be scarce or challenging to obtain.

Beyond DE, autoencoders have demonstrated their efficacy in a wide range of vision-related problems. They have been successfully applied to tasks such as image reconstruction, image registration, image segmentation, and human health posture analysis. The inherent capability of autoencoders to learn data representations directly from the input, without relying on predefined filters or explicit supervision, contributes to their versatility and effectiveness in various applications.

In this thesis, autoencoder networks play a pivotal role in our proposed models for DE. Leveraging the benefits of self-supervised learning, we utilize autoencoders to estimate the depth of a target domain from monocular images in a self-supervised manner. By leveraging the inherent structure within the data, our models aim to capture intricate DI and achieve accurate DE without relying on explicit supervision. The utilization of autoencoders as a fundamental component empowers our models to learn rich representations and leverage the latent information for robust depth prediction.

Overall, autoencoders offer a powerful framework for DE, enabling the learning of meaningful depth representations from unlabeled data. Through their self-supervised nature, they contribute to the advancement of DE techniques and open doors to novel applications in computer vision.

2.5 Summary

This chapter has covered various types of DE methods, encompassing traditional, ML-based, and DL-based approaches. The advantages and disadvantages of these methods have been discussed. Furthermore, the concepts of GCNs and autoencoders, which play a significant role in DL-based DE, have been explored. A thorough understanding of these techniques empowers researchers and practitioners to make informed decisions tailored to their applications.

In the next chapter, a comprehensive review of the current advancements in MDE utilizing DL techniques will be presented. The chapter's objective is to provide a comprehensive understanding of the field by offering an overview of the SOTA works on MDE. Key aspects such as input data shapes, training methodologies (including supervised, semi-supervised, and unsupervised approaches), and the use of various datasets and evaluation metrics will be emphasized. Furthermore, the chapter will address the limitations of DL-based MDE models, such as accuracy, computational

requirements, real-time inference, transferability, input image shapes, domain adaptation, and generalization. This discussion will highlight potential avenues for future research, setting the groundwork for further advancements in DL-based MDE.

Chapter 3

Monocular Depth Estimation Using Deep Learning: A Review

The aim of this chapter is to provide a comprehensive review of the current advancements in MDE utilizing DL techniques. To achieve this goal, we present an overview of the SOTA works on MDE, emphasizing key aspects such as input data shapes, training methodologies, including supervised, semi-supervised, and unsupervised approaches, and the use of various datasets and evaluation metrics. Furthermore, this chapter discusses the limitations of DL-based MDE models, such as their accuracy, computational requirements, real-time inference, transferability, input image shapes and domain adaptation, and generalization, thereby highlighting potential avenues for future research.

3.1 Introduction

Indisputable breakthroughs in the field of computational photography have helped the emergence of novel functionalities in the imaging process (Sun et al., 2016; Lam, 2015). Many works have been carried out so far in the field of computer vision (Rashwan et al., 2016). DE is a traditional computer vision task that predicts depth from one or more two-dimensional (2D) images. DE estimates each pixel's depth in an image using offline-trained models. In machine perception, recognition of some functional factors such as the shape of a scene from an image and image independence from its appearance seems to be fundamental (Godard, Mac Aodha, and Brostow, 2017; Liu et al., 2015; Eigen, Puhrsch, and Fergus, 2014). DE has great potential for use in disparate applications, including grasping in robotics, robot-assisted surgery, computer graphics, and computational photography (Cocias, Grigorescu, and Moldoveanu, 2012; Kalia, Navab, and Salcudean, 2019; Suo, Ji, and Dai, 2012; Lukac, 2017; Masoumian et al., 2020b; Ming et al., 2021). Figure 3.1 schematically illustrates the evaluation trend of DE.

The DE task needs an RGB image and a depth image as output. The depth image often consists of data about the distance of the object in the image from the camera viewpoint (Zhou et al., 2017). The computer-based DE approach has been under evaluation by various investigators worldwide, and the DE problem has been an exciting field of research. Most successful computer-based methods are employed by determining depth by applying stereo vision. With the progress of recent DL models, DE based on DL models has been able to demonstrate its remarkable efficiency in many applications (Khan, Salahuddin, and Javidnia, 2020; Tosi et al., 2019; Ramamonjisoa and Lepetit, 2019). DE can be functionally classified into three divisions, including MDE, binocular depth estimation (BDE), or multi-view depth estimation (MVDE).

MDE is an identified significant challenge in computer vision, in which no reliable

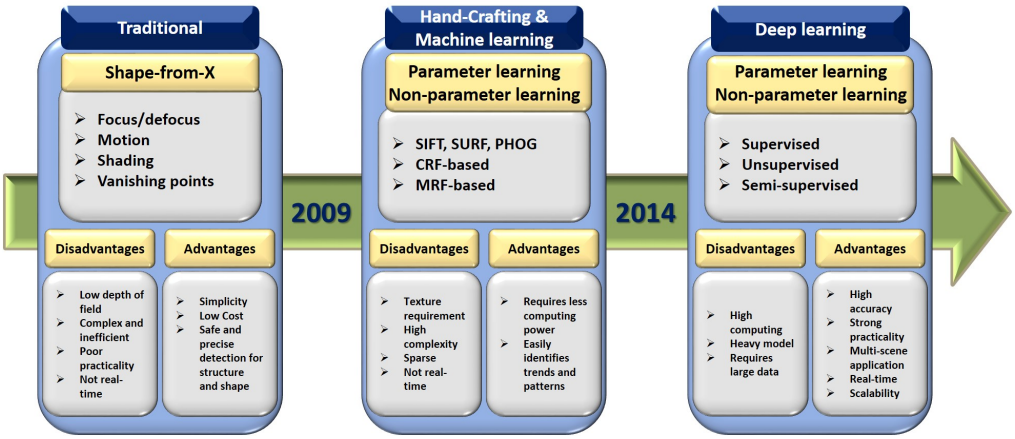


Figure 3.1: Evaluation trend of DE approaches divided into three sections: traditional methods, hand-crafting and ML methods, and DL methods.

cues exist to perceive depth from a single image. For instance, stereo correspondences are easily lost from MDE images (Schonberger and Frahm, 2016). Thus, the classical DE methods profoundly depend on multi-view geometry such as stereo images (Javidnia and Corcoran, 2017; Scharstein and Szeliski, 2002). These approaches need alignment procedures, which are of great importance for stereo- or multi-camera depth measurement systems (Heikkila and Silvén, 1997; Zhang, 2000). Consequently, using visual cues and disparate camera parameters, BDE and MVDE methods help to obtain DI. The majority of BDE or MVDE techniques can accurately estimate DI; however, many practical/operational challenges, such as calculation time and memory requirements for different applications, should be considered (Khan, Salahuddin, and Javidnia, 2020; Javidnia and Corcoran, 2016). The application of monocular images seems to be an excellent idea to capture DI to solve the memory requirement problem. The recent progression in using CNN and recurrent neural networks (RNN) yields a considerable improvement in the performance of MDE procedures (Kuznietsov, Stuckler, and Leibe, 2017; Bazrafkan et al., 2018; Fu et al., 2018).

Scientists worldwide have conducted various medical-based investigations to study the difference in depth perception with MDE or BDE systems. Despite the efforts to

use BDE or MVDE systems to estimate depths up to hundreds of meters, the majority of results imply that the most efficient distance for a BDE system is restricted to almost 10 m (Allison, Gillam, and Vecellio, 2009; Palmisano et al., 2010; Glennerster, Rogers, and Bradshaw, 1996). A small baseline of stereo pairs is the main reason behind the small depth range. Beyond this amount, human vision follows a monocular situation (Glennerster, Rogers, and Bradshaw, 1996). According to this information, it is obvious that the MDE systems can make better depth predictions than a human. Some problems, including the requirement for a great amount of training data and domain adaptation issues, exist and must be solved appropriately (Süvari, 2021).

In addition, research shows that industrial companies are looking at reducing costs and increasing the performance of their AI-based systems. Therefore, this chapter discusses the main advantages of MDE compared to stereo-based DE due to the low cost of grabbing sensors. In addition, it compares the MDE models from different aspects such as input data shapes and training manner. It discusses the advantages and disadvantages of each model to make it easier for the companies to better understand the differences between these models and select the suitable model for their system.

This chapter aims to review the highlighted studies on the recent advancements in the functional application of deep-learning-based MDE. Thus, many DE works from different aspects, including data input types (mono-sequence (Zhou et al., 2017; Mahjourian, Wicke, and Angelova, 2018; Masoumian et al., 2023), stereo sequence (Kuznietsov, Stuckler, and Leibe, 2017; Godard, Mac Aodha, and Brostow, 2017) and sequence-to-sequence (CS Kumar, Bhandarkar, and Prasad, 2018; Mancini et al., 2017)) and the training manner (supervised learning (SL) (Qi et al., 2018; Ummenhofer et al., 2017), unsupervised learning (UL) (Zhan et al., 2018; Garg et al., 2016; Zhou et al., 2017), and semi-supervised learning (SSL) (Kuznietsov, Stuckler, and Leibe, 2017; Luo et al., 2018; Xie, Girshick, and Farhadi, 2016) approaches) combined with the application of different datasets and evaluation indicators have been studied. Eventually, key points and future outlooks such as the accuracy, computational

time, resolution quality, real-time inference, transferability, and input data shapes are discussed to open new horizons for future research.

This survey includes over 150 papers, most of them recent, on a wide variety of applications of DL in MDE. To identify relevant contributions, PubMed was queried for papers containing ("Depth Estimation" OR "Relative Distance Prediction") in the title or abstract. ArXiv searched for papers mentioning one of a set of terms related to computer vision. Additionally, conference proceedings for CVPR and ICCV were searched based on the titles of papers. We checked references in all selected papers and consulted colleagues. The papers without reported results are excluded. When overlapping work had been reported in multiple publications, only the publication(s) deemed most important were included.

Several surveys concerning MDE have been published in recent years, as summarized in Table 3.1. In this survey, we are concerned with six parameters that are used to assess any MDE method; "TM": training manner, "ACC": accuracy, "CT": computational Time, "RQ": resolution quality, "RTI": real-time inference, "TRAN": transferability, "IDS": input data shapes. In Table 3.1, we also compare our review to the recent surveys in terms of the six parameters to show that all of these surveys do not focus on all of these parameters.

Table 3.1: Comprehensive to the related recent surveys in MDE in terms of six parameters; "TM": Training Manner, "ACC": Accuracy, "CT": Computational Time, "RQ": Resolution Quality, "RTI": Real-time Inference, "TRAN": Transferability, "IDS": Input Data Shapes.

Title	Year	TM	ACC	CT	RQ	RTI	TRAN	IDS
Deep Learning-Based Monocular Depth Estimation Methods (Khan, Salahuddin, and Javidnia, 2020)	2020	✓	✓	✓				
Monocular depth estimation based on deep learning (Zhao et al., 2020)	2020	✓	✓			✓	✓	
Deep Learning for Monocular Depth Estimation (Ming et al., 2021)	2021	✓			✓			
Towards Real-Time Monocular Depth Estimation for Robotics (Dong et al., 2021)	2021	✓	✓	✓		✓		
Outdoor Monocular Depth Estimation (Vyas et al., 2022)	2022	✓				✓		
Ours (Masoumian et al., 2022)	**2022**	✓	✓	✓	✓	✓	✓	✓

This chapter is organized in the following way: Section 3.2 describes the background of DE. The DE task's main datasets and evaluation metrics are reviewed in Sections 3.3 and Section 3.4, respectively. MDE based on DL models and a comparison of three main data input shapes and training manner approaches are described in Sections 3.5 and 3.6. Section 3.7 presents the discussion, and Section 3.8 concludes this review.

3.2 Depth Estimation

Objects' depth in a scene possesses the remarkable ability of estimation/calculation by applying passive and active approaches. In the active approaches (i.e., applications of LiDAR sensors and RGB-D cameras), the DI is achieved quickly (Trouvé et al., 2013; Rodrigues et al., 2020). RGB-D camera is a specific type of depth-sensing device that combines an RGB image and its corresponding depth image (Ulrich et al., 2020). RGB-D cameras can be used in various devices such as smartphones and unmanned aerial systems due to their low cost and power consumption (Kim et al., 2020). RGB-D cameras have limited depth range and they suffer from specular reflections and absorbing objects. Therefore, many depth completion approaches have been proposed to mitigate the gap between sparse and dense depth maps (Dong et al., 2021).

In passive techniques, DI is often achieved using two principal methodologies: depth from stereo images and monocular images. The main purpose of both techniques is to assist in building the spatial structure of the environment, which presents a 3D view of the scene. After achieving DI, the situation of the viewer would be recognized relative to the surrounding objects. Stereo vision is a widely applied depth calculation procedure in the computer vision area. Stereo vision is known as a computer-based passive approach in which stereo images are applied to extract DI (Boykov, Veksler, and Zabih, 1998; Meng et al., 2021; Sanz, Mezcua, and Pena, 2012). To compute disparity, pixel matching must be implemented among the pixels of both images. It

is worth noting that a good correspondence (pixels) matching needs the rectification of both images. Rectification is defined as the transformation process of images to match the epipolar lines of the original images horizontally (Loop and Zhang, 1999; Fusiello, Trucco, and Verri, 1997). Figure 3.2 demonstrates the images before and after the rectification process. The matching process of the pixel in an image with its similar pixel in another image along an epipolar line occurs using a matching cost function. By matching the pixels of both images, the calculation of depth applying the distance between two cameras and the pixel distance between matched pixels will be possible (Kat, Jevnisek, and Avidan, 2018; Zhong and Quan, 2021). Reflective and highly transparent zones accompanied by smooth areas are the major challenges for stereo-matching algorithms. Owing to perspective alteration, an image's edge details can disappear in the second image. If the algorithm does not have sufficient capability to match the edge points on another image, it can create an erroneous depth value and noise in the predicted depth map at those points (Zhou, Meng, and Cheng, 2020; Alagoz, 2008).

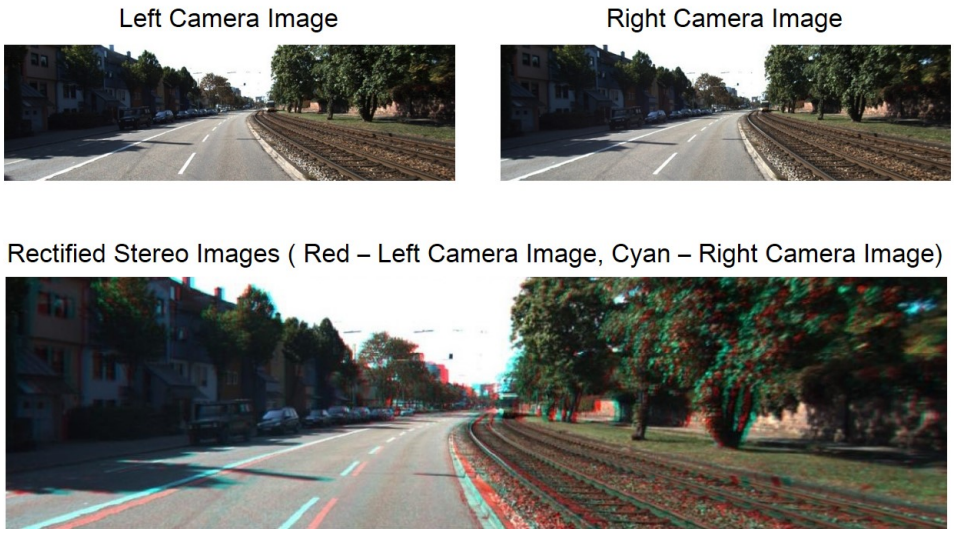


Figure 3.2: (**Top**) Non-rectified left and right images, and (**down**) red–cyan anaglyph from stereo pair of rectified stereo images.

Sometimes, the application of algorithms for calculating depth may create different challenges. For instance, the matching cost function utilized in the algorithm can generate false-positive signals, which result in the creation of depth maps with low accuracy. Thus, the use of post-processing approaches (i.e., median filter, bilateral filter, and interpolation) is of great importance in stereo vision applications to delete noise and refine depth maps (Luo, Schwing, and Urtasun, 2016; Aboali, Abd Manap, Yusof, et al., 2018; Hyun et al., 2020; Silva Vieira et al., 2018).

On the contrary, MDE does not require rectified images since MDE models work with a sequence of images extracted from a single camera. This simplicity and easy access are some of the main advantages of MDE compared to stereo models, which require additional complicated pieces of equipment. Because of that, in recent years, demand for MDE increased significantly. Most MDE methods focus on estimating distances between scene objects and the camera from one viewpoint. It is essential for regressing depth in 3D space in MDE methods since there is a lack of reliable stereoscopic visual relationship in which images adopt a 2D form to reflect the 3D space (Ming et al., 2021). Therefore, MDE models try to recover the depth maps of images, which reflect the 3D structure of the scene. Most of the MDE models have the main architecture, which contains two main parts: depth and pose networks. The depth network predicts the depth maps. In turn, the pose network works as an ego-motion estimation (i.e., rotation and translation of the camera) between two successive images. The estimated depth (i.e., disparity) maps with the ego-motion parameters used to reconstruct an image should be compared to the target image. Figure 3.3 represents the schematic illustration of this method.

3.3 Datasets

There are various types of datasets for depth prediction based on different viewpoints. This section highlights the most popular public datasets of DL models for MDE.

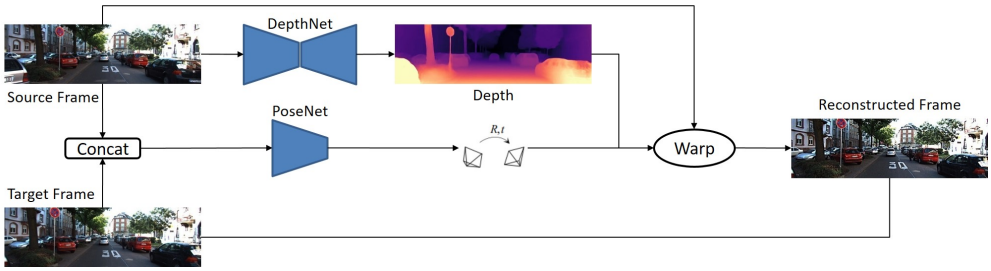


Figure 3.3: Main network structure for MDE (Masoumian et al., 2023). This network contains two sub-networks: DepthNet for predicting the depth map and PoseNet for estimating the camera pose.

3.3.1 KITTI

The KITTI dataset (Geiger, Lenz, and Urtasun, 2012) is considered the most commonly applied dataset in computer vision, such as optical flow, visual odometry (VO), and semantic segmentation (Geiger, Lenz, and Urtasun, 2012; Mayer et al., 2016; Zhao et al., 2021; Wang et al., 2018b; Abdulwahab et al., 2022). This dataset is also the most prevalent criterion in the unsupervised/semi-supervised MDE. In this dataset, 56 scenes are divided into two main compartments: 28 scenes for training and the rest for testing (Eigen, Puhrsch, and Fergus, 2014). Due to the incredible capability of the KITTI dataset to create the pose ground truth for 11 odometry sequences, it is extensively applied to assess deep-learning-based VO algorithms (Xue et al., 2019; Clark et al., 2017). This dataset contains 39,810 images for training, 4424 for validation, and 697 for testing. The resolution of the images is 1024×320 pixels. The MDE results of the UL, SL, and SSL procedures investigated on the KITTI dataset are presented in Table 3.2.

Table 3.2: Comprehensive information about the quantitative results of the SL, SSL, and UL algorithms investigated on the KITTI dataset.

		Lower Better				Higher Better		
Method	**Training Pattern**	**Abs-Rel**	**Sq-Rel**	**RMSE**	**RMSE-Log**	**δ<1.25**	**δ<1.25^2**	**δ<1.25^3**
Bhat (Bhat, Alhashim, and Wonka, 2021)	SL	0.058	0.190	2.360	0.088	0.964	0.995	0.999
Wang (Wang, Pizer, and Frahm, 2019)	SL	0.088	0.245	1.949	0.127	0.915	0.9984	0.996
Patil (Patil et al., 2020)	SL	0.102	0.655	4.148	0.172	0.884	0.966	0.987
BTS (Lee et al., 2019)	SL	0.059	0.241	2.756	0.096	0.956	0.993	0.998
DepthNet (CS Kumar, Bhandarkar, and Prasad, 2018)	SL	0.137	1.019	5.187	0.218	0.809	0.928	0.971
Kuznietsov (Kuznietsov, Proesmans, and Van Gool, 2021)	SL	0.122	0.763	4.815	0.194	0.845	0.957	0.987
Monodepth (Godard, Mac Aodha, and Brostow, 2017)	SSL	0.148	1.344	5.927	0.247	0.803	0.922	0.964
SemiSup (Kuznietsov, Stuckler, and Leibe, 2017)	SSL	0.113	0.741	4.621	0.189	0.803	0.960	0.986
GMS (Ramirez et al., 2018)	SSL	0.143	2.161	6.526	0.222	0.850	0.939	0.972
GAN (Aleotti et al., 2018)	SSL	0.119	1.239	5.998	0.212	0.849	0.940	0.976
DepthGAN (Pilzer et al., 2018)	SSL	0.152	1.388	6.016	0.247	0.789	0.918	0.965
MonoRes (Tosi et al., 2019)	SSL	0.111	0.867	4.714	0.199	0.864	0.954	0.979
Hints (Watson et al., 2019)	SSL	0.112	0857	4.807	0.203	0.862	0.952	0.978
SfMLearner (Zhou et al., 2017)	UL	0.208	1.768	6.958	0.283	0.678	0.885	0.957
Vid2Depth (Mahjourian, Wicke, and Angelova, 2018)	UL	0.163	1.240	6.220	0.250	0.762	0.916	0.968
GeoNet (Yin and Shi, 2018)	UL	0.155	1.296	5.857	0.233	0.793	0.931	0.973
Struct2Depth (Casser et al., 2019)	UL	0.141	1.036	5.291	0.215	0.816	0.945	0.979
CC (Ranjan et al., 2019)	UL	0.140	1.070	5.326	0.217	0.826	0.941	0.975
LearnK (Gordon et al., 2019)	UL	0.128	0.959	5.232	0.212	0.845	0.947	0.976
DualNet (Zhou et al., 2019)	UL	0.121	0.837	4.945	0.197	0.853	0.955	0.982
Monodepth2 (Godard et al., 2019)	UL	0.115	0.882	4.701	0.190	0.879	0.961	0.982
FeatDepth (Shu et al., 2020)	UL	0.104	0.729	4.481	0.179	0.893	0.965	0.984
GCNDepth (Masoumian et al., 2023)	UL	0.104	0.720	4.494	0.181	0.888	0.965	0.984

3.3.2 NYU Depth-V2

The NYU Depth (Silberman et al., 2012) is a vital dataset, which includes 464 indoor scenes that concentrate on indoor environments. Compared to the KITTI dataset, which collects ground truth with LiDAR, this dataset accepts monocular video sequences of scenes and an RGB-D camera's ground truth of depth. The NYU Depth is the main training dataset in the supervised MDE. The indoor scenes are divided into 249 and 215 sections for training and testing. Due to disparate variable frame rates, there is no one-to-one communication between depth maps and RGB images. Intending to arrange the depth and the RGB images, each depth map is related to the nearest RGB image. In addition, due to the discretion of the projection, all pixels do not possess an associated depth value. Therefore, those pixels that do not have depth values are masked within the experiments (Fu et al., 2018; Silberman et al., 2012). The resolution of the RGB images in sequences is 640×480 pixels. The MDE results

of the investigation on the NYU-V2 dataset are presented in Table 3.3.

Table 3.3: Comprehensive information about the quantitative results of the DL algorithms investigated on the NYU-V2 dataset.

		Lower Better				Higher Better		
Method	**Training Pattern**	**Abs-Rel**	**Sq-Rel**	**RMSE**	**RMSE-Log**	δ<1.25	δ<1.25^2	δ<1.25^3
DeepV2D (Teed and Deng, 2018)	SL	0.061	0.094	0.403	0.026	0.956	0.989	0.996
VNL (Yin et al., 2019)	SL	0.113	0.034	0.364	0.054	0.815	0.990	0.993
Fast-MVSNet (Yu and Gao, 2020)	SL	0.551	0.980	3.241	0.243	0.816	0.915	0.939
DORN (Fu et al., 2018)	SL	0.138	0.051	0.509	0.653	0.825	0.964	0.992
BTS (Lee et al., 2019)	SL	0.110	0.066	0.392	0.142	0.885	0.978	0.994
GASDA (Zhao et al., 2019)	SSL	1.356	1.156	0.963	1.223	0.765	0.897	0.968
DnD (Jung et al., 2021)	SSL	0.213	0.320	2.360	0.084	0.761	0.889	0.932
DenseDepth (Alhashim and Wonka, 2018)	SSL	0.093	0.589	4.170	0.171	0.886	0.965	0.986
SharpNet (Ramamonjisoa and Lepetit, 2019)	UL	0.139	0.047	0.495	0.157	0.888	0.979	0.995
MonoRes (Tosi et al., 2019)	UL	1.356	1.156	0.694	1.125	0.825	0.965	0.967
DepthComple (Ma, Cavalheiro, and Karaman, 2019)	UL	0.842	0.760	5.880	0.233	0.863	0.921	0.972
Packnet-SfM (Guizilini et al., 2020)	UL	2.343	1.158	0.887	1.234	0.821	0.945	0.968
Monodepth2 (Godard et al., 2019)	UL	2.344	1.365	0.734	1.134	0.826	0.958	0.979

3.3.3 Cityscapes

This dataset prominently concentrates on semantic segmentation tasks. In this dataset, 5000 fine-annotation images and 20,000 coarse-annotation images exist (Wang et al., 2018b; Cordts et al., 2016). Cityscapes dataset includes a series of stereo video sequences, which has only the potential of application for the training process of disparate unsupervised DE procedures (Yin and Shi, 2018). The efficiency of depth networks can be significantly improved by pretraining the networks on the Cityscapes (Godard, Mac Aodha, and Brostow, 2017; Zhou et al., 2017; Bian et al., 2019). The training data of this dataset include 22,973 stereo image pairs with a resolution of 1024×2048.

3.3.4 Make3D

These data include both monocular RGB and depth images but do not possess stereo images that are different from the datasets mentioned above (Saxena, Sun, and Ng, 2008b; Saxena, Sun, and Ng, 2008a). Due to the non-existence of monocular sequences

in the Make3D dataset, SSL and UL procedures do not apply it as the training set, while SL techniques often adopt it for training. The fact of the matter is that the Make3D dataset is extensively used as a testing set of unsupervised algorithms to assess the production capability of networks on disparate datasets (Godard, Mac Aodha, and Brostow, 2017). The RGB image resolution is 2272×1704, and the depth map resolution is 55×305 pixels. The MDE results of the investigation on the Make3D dataset are presented in Table 3.4.

Table 3.4: Comprehensive information about the quantitative results of the DL algorithms investigated on the Make3D dataset.

Method	Training Pattern	Abs_Rel	Sq_Rel	RMSE	$\log_{10}$
Karsch (Karsch, Liu, and Kang, 2014)	SL	0.428	5.079	8.389	0.149
Liu (Liu, Salzmann, and He, 2014)	SL	0.475	6.562	10.05	0.165
Laina (Laina et al., 2016)	SL	0.204	1.840	5.683	0.084
SfMLearner (Zhou et al., 2017)	UL	0.383	5.321	10.47	0.478
DDVO (Wang et al., 2018a)	UL	0.387	4.720	8.090	0.204
Monodepth2 (Godard et al., 2019)	UL	0.322	3.589	7.417	0.201
Jia (Jia et al., 2021)	UL	0.289	2.423	6.701	0.348
GCNDepth (Masoumian et al., 2023)	UL	0.424	3.075	6.757	0.107

3.3.5 DIODE

DIODE (Vasiljevic et al., 2019) is the Dense Indoor/Outdoor Depth dataset for MDE comprising diverse indoor and outdoor scenes acquired with the same hardware setup. This dataset consists of 8574 indoor and 16,884 outdoor samples from 20 scans each for training and 325 indoor and 446 outdoor samples with each set from 10 different scans for validation with the resolution of 768×1024. The indoor and outdoor ranges for the dataset are 50 m and 300 m, respectively.

3.3.6 Middlebury 2014

Middlebury (Scharstein et al., 2014) is a dense indoor scene dataset that contains 33 images of 6-megapixel high resolution. Images are captured via two stereos DSLR cameras and two point-and-shoot cameras. Disparity ranges are between 200 and

800 pixels at a resolution of 6 megapixels. The image resolution of this dataset is 2872×1984.

3.3.7 Driving Stereo

The driving stereo (Yang et al., 2019) is one of the new large-scale stereo driving datasets that contains 182k images. The disparity images are captured via LiDAR, the same as the KITTI dataset. They mainly focus on two new metrics, a distance-aware metric and a semantic-aware metric, for evaluating stereo matching on MDE. The image resolution of this dataset is 1762×800. Table 3.5 represents the summary of dataset features for DE.

Table 3.5: A summary of DE public datasets.

Dataset	Sensors	Annotation	Type	Scenario	Images	Resolution	Year
KITTI (Geiger, Lenz, and Urtasun, 2012)	LiDAR	Sparse	Real	Driving	44K	1024×320	2013
NYU-V2 (Couprie et al., 2013)	Kinect V1	Dense	Real	Indoor	1449	640×480	2012
Cityscapes (Cordts et al., 2016)	Stereo Camera	Disparity	Real	Driving	5K	1024×2048	2016
Make3D (Saxena, Sun, and Ng, 2008b)	Laser Scanner	Dense	Real	Outdoor	534	2272×1704	2008
DIODE (Vasiljevic et al., 2019)	Laser Scanner	Dense	Real	In/Outdoor	25.5K	768×1024	2019
Middlebury 2014 (Scharstein et al., 2014)	DSLR Camera	Dense	Real	Indoor	33	2872×1984	2014
Driving Stereo (Yang et al., 2019)	LiDAR	Sparse	Real	Driving	182K	1762×800	2019

Although many valuable datasets and benchmarks exist for assessing monocular and stereo DE methods, there are still some limitations in the available datasets. For instance, all these datasets include images captured only during the day or night, yet there are no datasets to have both together, and the same applies to indoor or outdoor images. In addition, no dataset concerns different challenges related to the change in weather conditions (e.g., fog, sunny, snow, etc.).

3.4 Evaluation Metrics

To assess the efficiency of the DE models, an accepted evaluation procedure was recommended by Eigen et al. (Eigen, Puhrsch, and Fergus, 2014), which possesses five

evaluation metrics, including absolute relative difference (Abs-Rel), square relative error (Sq-Rel), root mean square error (RMSE), RMSE-log, and accuracy, with a threshold (δt). They are formulated using the following equations (Eigen, Puhrsch, and Fergus, 2014):

$$Abs - Rel = \frac{1}{|D|} \sum_{pred \in D} |gt - pred|/gt \tag{3.1}$$

$$Sq - Rel = \frac{1}{|D|} \sum_{pred \in D} ||gt - pred||^2/gt \tag{3.2}$$

$$RMSE = \sqrt{\frac{1}{|D|} \sum_{pred \in D} ||gt - pred||^2} \tag{3.3}$$

$$RMSE - Log = \sqrt{\frac{1}{|D|} \sum_{pred \in D} ||\log(gt) - \log(pred)||^2} \tag{3.4}$$

$$\delta t = \frac{1}{|D|} |\{pred \in D \, | max(\frac{gt}{pred}, \frac{pred}{gt}) < 1.25^t\}| \times 100\% \tag{3.5}$$

In these equations, the pred and gt denote predicted depth and ground truth, respectively. D represents the set of all predicted depth values for a single image, $|\,.\,|$ returns the number of the elements in each input set, and δt represents the threshold.

3.5 Input Data Shapes for MDE Applying DL

This section mainly introduces common types of data input for MDE. The input data shapes in MDE networks can be divided into three main categories: mono-sequence, stereo-sequence, and sequence-to-sequence input data. Based on the architecture of the networks, the input data shapes will be different.

3.5.1 Mono-Sequence

Monocular sequence input is mainly used for training the UL models. Figure 3.4 shows the basic structure of mono-sequence models, which have a single input image and a single output image. UL networks consist of a depth network for predicting depth maps and a pose network for camera pose estimation. The camera pose estimation works similarly to image transformation estimation, which helps to improve the results of MDE. These two sub-networks are connected in parallel, and the whole model is obliged to reconstruct the image. In mono-sequence, mostly the geometric constraints are built on adjacent frames. Lately, researchers have used VO (Nistér, Naroditsky, and Bergen, 2004) to predict the camera motion for learning the scene depth. Zhou et al. (Zhou et al., 2017) were the pioneers of the mono-sequence input type, and they proposed a network to predict camera motion and depth maps with photometric consistency loss and reconstruction loss.

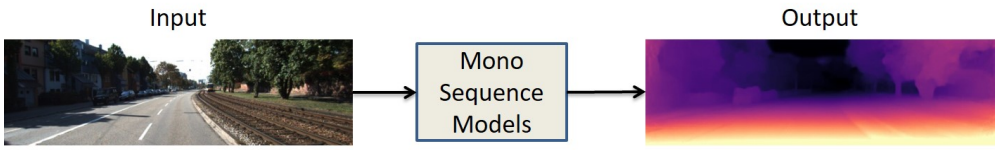


Figure 3.4: Data input/output structure of mono-sequence models. Single image input and single image output.

Furthermore, Mahjourian et al. (Mahjourian, Wicke, and Angelova, 2018) introduced a network with 3D geometric constraints and enforced consistency of the estimated 3D point clouds and ego-motion across consecutive frames. Recently, Masoumian et al. (Masoumian et al., 2023) designed two jointly connected sub-networks for depth prediction and ego-motion. They used CNN-GCN encoder-decoder architecture for their networks with three losses: reconstruction loss, photometric loss, and smooth loss. In addition, Shu et al. (Shu et al., 2020) proposed a similar method with two jointly connected depth and pose predictions that were slightly different. They

also added a feature extractor encoder to their model to improve the quality of their predicted depth maps. Their proposed architecture is shown in Figure 3.5.

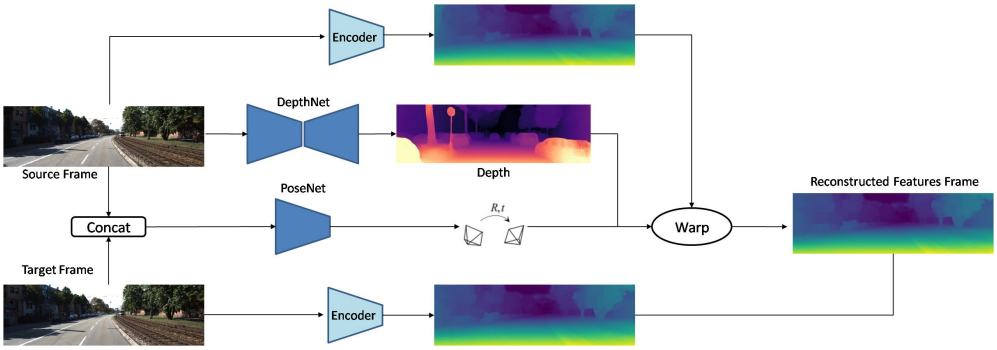


Figure 3.5: Developed network by Shu et al. (Shu et al., 2020).

3.5.2 Stereo Sequence

The projection and mapping relationship between the left and right pairwise images is mainly constrained by stereo matching. In order to build geometric constraints, a stereo-image dataset is required. These types of inputs are commonly used in UL and SL networks. Figure 3.6 represents the basic structure of stereo sequence models which have left and right images as input and a single output. Similar to the monocular sequence input data shape, the stereo sequence works with image reconstruction with slight differences. An image will be reconstructed based on warping between the depth map and the right image. For instance, Kuznietsov et al. (Kuznietsov, Stuckler, and Leibe, 2017) proposed an SSL model for MDE with sparse data, and they built a stereo alignment as a geometric constraint.

Furthermore, Godard et al. (Godard, Mac Aodha, and Brostow, 2017) designed a UL network with left-right consistency constraints. They used CNN-based encoder-decoder architecture for their model with the reconstruction loss, left–right disparity consistency, and disparity smoothness loss. Recently, Goldman et al. (Goldman, Hassner, and Avidan, 2019) proposed a Siamese network architecture with weight sharing,

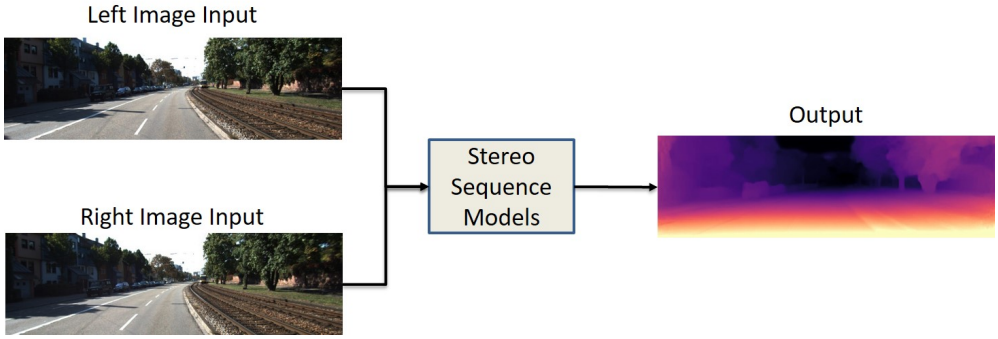


Figure 3.6: Data input/output structure of stereo sequence models. Stereo pairs of images as an input and single image output.

which consists of two twin networks, each learning to predict a disparity map from a single image. Their network is composed of an encoder-decoder pair with skip connections, which is shown in Figure 3.7.

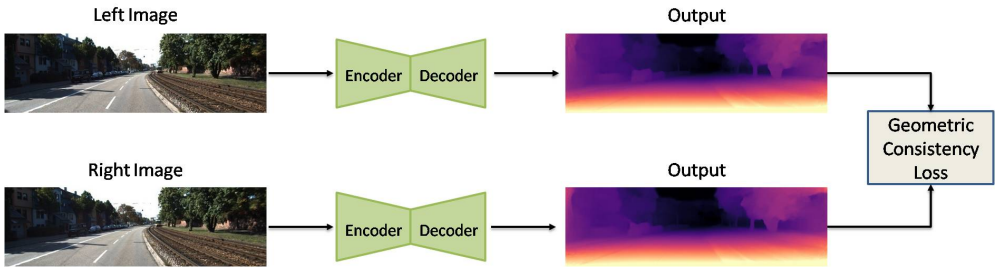


Figure 3.7: Developed network by Goldman et al. (Goldman, Hassner, and Avidan, 2019).

3.5.3 Sequence to Sequence

Sequence-to-sequence data input is necessary for RNN models (Makarov et al., 2022). These models have memory capability, which helps the system learn a group of features in sequence images. Figure 3.8 represents the basic structure of sequence-to-sequence models, which have a sequence of images as input and a sequence of depth maps

as output. Most RNN methods use long short-term memory (LSTM) to learn the long-term dependencies with a three-gate structure (Makarov et al., 2022).

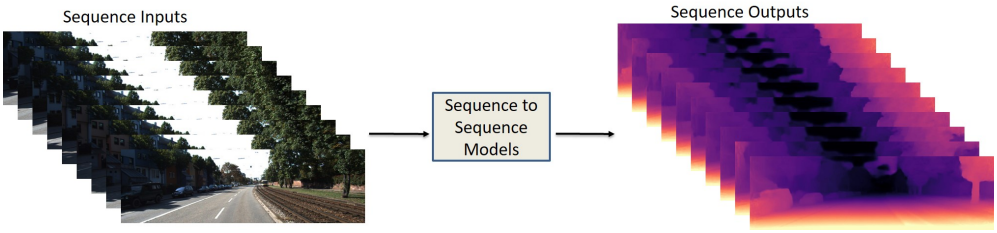


Figure 3.8: Data input/output structure of sequence-to-sequence models. Sequence of images as an input and sequence of images as an output.

However, RNN and CNN networks will be combined to extract spatial-temporal features. The sequence-to-sequence data primarily will be trained on SL models. Kumar et al. (CS Kumar, Bhandarkar, and Prasad, 2018) proposed an MDE model with ConvLSTM layers for learning the smooth temporal variation. Their model consists of encoder-decoder architecture, which is shown in Figure 3.9.

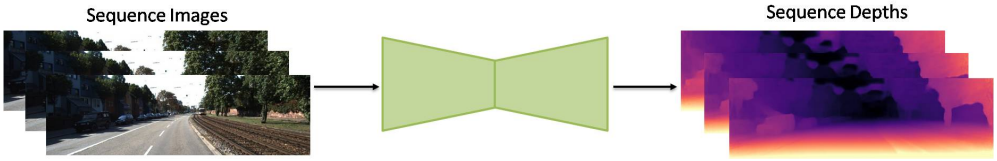


Figure 3.9: Developed network by Kumar et al. (CS Kumar, Bhandarkar, and Prasad, 2018).

Furthermore, Mancini et al. (Mancini et al., 2017) improved LSTM layers to obtain the best outcome of the predicted depth maps by feeding the input images sequentially to the system.

3.6 MDE Applying DL Training Manners

Although DE from multiple images possesses a lengthy background in the computer vision area, the DI extraction process from single images is considered a novel concept

in DL. The advancements have initiated comprehensive investigations of the DI concept in DL techniques. The most critical challenge towards the application of DL is the absence of datasets that fit the problem (Bugby et al., 2021; Praveen, 2020; Mandelbaum, Kamberova, and Mintz, 1998). This challenge may also be of great importance for the MDE network. Data applied in training may be collected by LiDAR sensors, RGB-D cameras, or stereo vision cameras. Despite the expensive data collection process, disparate learning strategies have been developed to decrease dependency on the dataset used for training. The learning process in MDE networks can be divided into three parts, including SL, UL, and SSL (Godard, Mac Aodha, and Brostow, 2017; Kuznietsov, Stuckler, and Leibe, 2017; Qi et al., 2018; Garg et al., 2016; Poggi et al., 2018).

3.6.1 Supervised Learning Approach

The SL approach for DE needs pixel-wise ground truth DI (Cunningham, Cord, and Delany, 2008). The SL procedure applies ground truth depth (GTD) to train a neural network as a regression model (Liu et al., 2019; Godard et al., 2019; Abdulwahab et al., 2021). Eigen et al. (Eigen, Puhrsch, and Fergus, 2014) were pioneers in investigating DI to train a model using DL. They explained that their developed CNN-based network consists of two deep network stacks. Figure 3.10 presents a schematic illustration of the network structure proposed in (Eigen, Puhrsch, and Fergus, 2014). As shown in Figure 3.10, the preparation of the input image occurred for both stacks. Additionally, the preparation of the output depth map of the first stack takes place to refine the depth map. The main responsibility of the second stack is to arrange obtained coarse depth predictions with the objects in the scene (Eigen, Puhrsch, and Fergus, 2014).

After Eigen's investigation, different procedures were implemented to increase the precision of the estimated depth map (EDP). For example, Li et al. (Li et al., 2015) developed a DL network applying conditional random fields (CRFs). They utilized

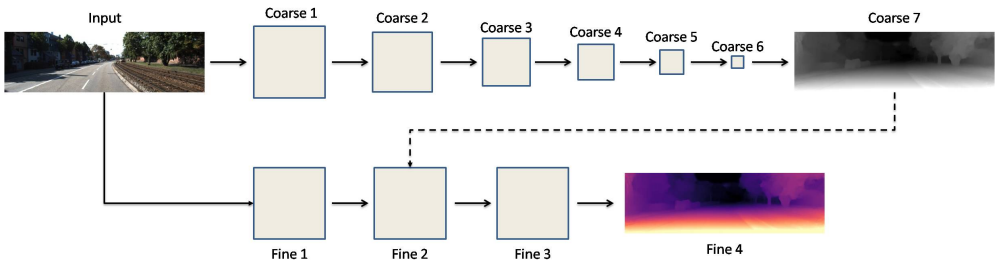


Figure 3.10: Developed network structure by Eigen et al. (**5**).

a two-stage network for depth map estimation and refinement. In the first stage, a super-pixel technique on the input image is applied, and image patches are extracted around these super-pixels. In the second stage, CRFs are applied to refine the depth map by changing the super-pixel depth map to the pixel level. In order to extract an appropriate depth map, some approaches use geometric relationships. For example, Qi et al. (Qi et al., 2018) utilized two networks to estimate the depth map and surface normal from single images. Figure 3.11 depicts the developed network in (Qi et al., 2018). These two networks enable the conversion of depth-to-normal and normal-to-depth and collaboratively increase the accuracy of the depth map and surface normal. Although their neural network can increase the accuracy of depth maps, for training, they require ground truth, including surface normal, which is hard to obtain. Ummenhofer et al. worked on developing a network to estimate depth maps using the structure from motion (SfM) technique. They corroborated that basic encoder-decoder architecture does not have sufficient capacity to process two input images simultaneously. Therefore, they developed a computer-based neural architecture that can extract optical flow, ego-motion, and a depth map from an image pair (Ummenhofer et al., 2017).

The dataset's quality is an introductory section in SL systems, similar to methodology. Dos Santos et al. (Santos Rosa, Guizilini, and Grassi, 2019) paid enough attention to this challenge. They developed an approach to creating denser GTD maps from

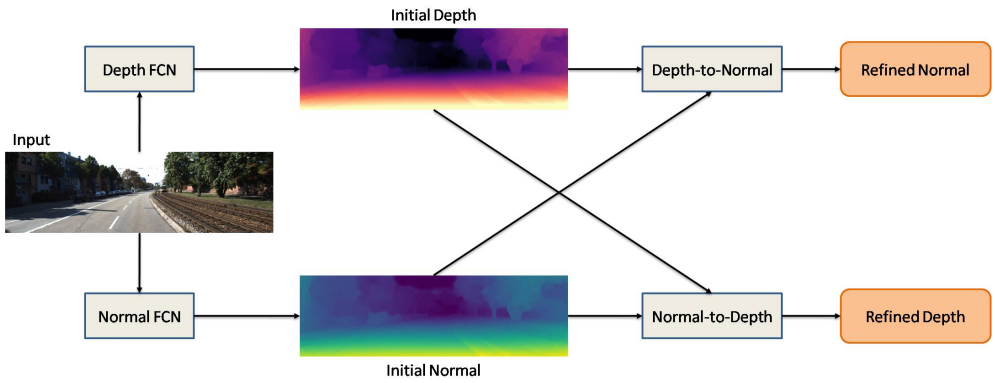


Figure 3.11: Developed geometric neural network by Qi et al. (Qi et al., 2018).

sparse LiDAR measurements by enhancing the valid depth pixels in depth images. They compared the obtained results of their trained model with both sparse GTD maps and denser GTD maps. They understood that the application of denser ground truth results in yields increasing performance compared to sparse GTD maps. Ranftl et al. (Ranftl et al., 2019) developed an outstanding learning strategy that can involve various datasets to improve the efficiency of the MDE network. To prepare their dataset for three-dimensional movies, they applied stereo matching to conclude the depth of frames of these movies. Disparate unclear problems, including changing resolution and negative/positive disparity values, emerged during the creation of this dataset. With the assistance of their developed procedures for incorporating multiple datasets, they achieved high precision with their model MDE problem. Recently, Sheng et al. (Sheng et al., 2022) proposed a lightweight SL model with local-global optimization. They used an autoencoder network to predict the depth and used a local-global optimization scheme to realize the global range of scene depth.

3.6.2 Unsupervised Learning Approach

The increment of layers and trainable parameters in deep neural networks significantly increases the requirement for the train data, resulting in difficulty in achieving GTD

maps. For this reason, UL approaches become an appropriate choice because unlabeled data is relatively easier to find (Geng et al., 2020; Zhan et al., 2018; Lu and Lu, 2019). Garg et al. (Garg et al., 2016) were the pioneers of developing a promising procedure to learn depth in an unsupervised fashion to remove the requirement of GTD maps. Up until now, developed UL approaches have applied stereo images, and thus, supervision and train loss depend intensely on image reconstruction. In order to train a depth prediction network, consecutive frames from a video may have great potential for application as supervision. Camera transformation estimation (pose estimation) between successive frames is the major challenge of this procedure, which results in extra complexity for the network. As illustrated in Figure 3.12, Zhou et al. (Zhou et al., 2017) developed computer-based architecture to estimate depth map and camera pose simultaneously. As input, three successive frames are fed to the network. Pose CNN and Depth CNN estimate relative camera poses and a depth map from the first image.

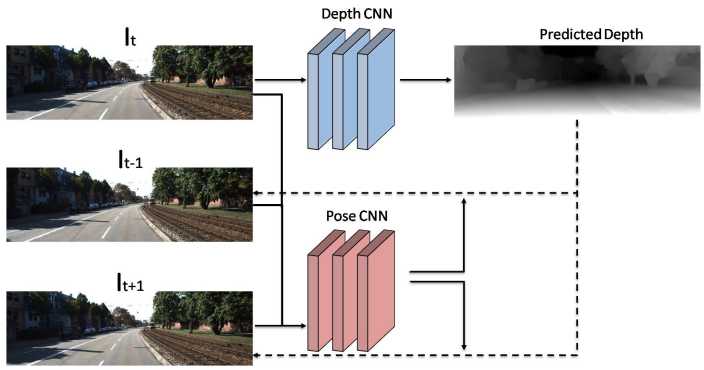


Figure 3.12: Developed network by Zhou et al. (Zhou et al., 2017).

In order to obtain greater accuracy in DE, some approaches have existed that possess the great potential of application to merge multiple self-supervision procedures into one. For instance, Godard et al. (Godard et al., 2019) applied MDE and estimated relative camera poses to build other stereoviews and contiguous frames in the video

sequence. They added a pose network to their model to predict relative camera pose in adjacent frames. One of the crucial challenges to using self-supervised approaches via video is occluded pixels. They applied minimum loss compared to the classical average loss to obtain non-occluded pixels, which is known as a significant improvement (Godard, Mac Aodha, and Brostow, 2017). The improvement in the precision of UL approaches has motivated other investigators to modify knowledge distillation methods for the MDE problem. Pilzer et al. developed a system to adapt an unsupervised MDE network to the teacher-student learning framework by applying stereo image pairs to train a teacher network. Despite the promising performance of their student network, it was not as accurate as their teacher network (Pilzer et al., 2019). Masoumian et al. (Masoumian et al., 2023) developed a multi-scale MDE based on a GCN. Their network consists of two parallel autoencoder networks: DepthNet and PoseNet. The DepthNet is an autoencoder composed of two parts: encoder and decoder; the CNN encoder extracts the feature from the input image, and a multi-scale GCN decoder estimates the depth map, as illustrated in Figure 3.13. PoseNet is used to estimate the ego-motion vector (i.e., 3D pose) between two consecutive frames. The estimated 3D pose and depth map is used to construct a target image.

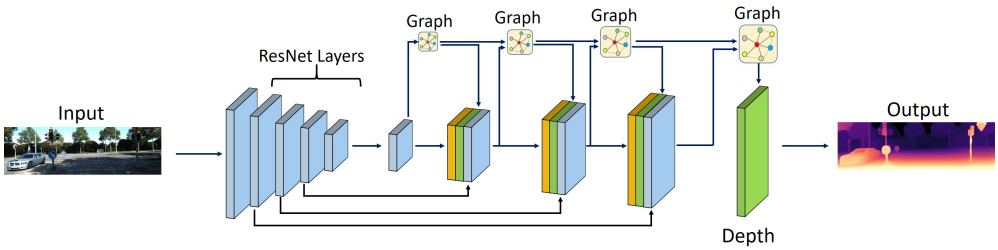


Figure 3.13: Developed network by Masoumian et al. (Masoumian et al., 2023).

3.6.3 Semi-Supervised Learning Approach

Compared to SL and UL approaches, few investigations have been conducted to study the performance of SSL methods for MDE. Apart from SL and UL approaches, Kuznietsov et al. (Kuznietsov, Stuckler, and Leibe, 2017) developed an SSL method by simultaneously applying supervised/unsupervised loss terms during training. Figure 3.14 demonstrates the components/inputs of the developed semi-supervised loss function in (Kuznietsov, Stuckler, and Leibe, 2017).

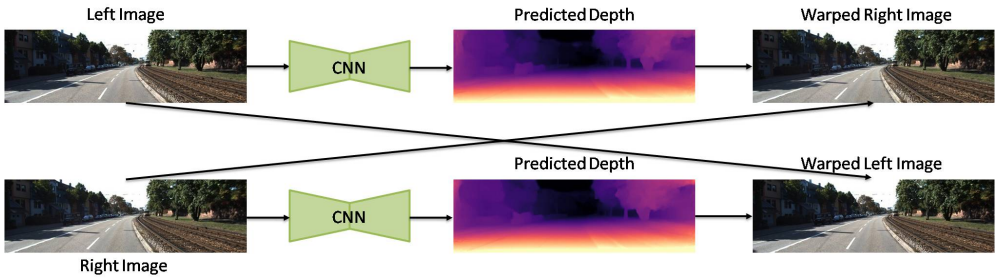


Figure 3.14: Components/inputs of the developed semi-supervised loss function by Kuznietsov et al. (Kuznietsov, Stuckler, and Leibe, 2017).

In their approach, the estimated disparity maps (i.e., inverse depth maps) were used to rebuild left and right images via warping. Computation of unsupervised loss term took place by rebuilding the target images. Simultaneously, the calculation of the supervised loss term occurred by the estimated depth, and GTD maps (Kuznietsov, Stuckler, and Leibe, 2017). Luo et al. (Luo et al., 2018) classified the MDE problem into two subdivisions and investigated them separately. Based on their procedure, the network requirement for labeled GTD data decreased. Additionally, they corroborated that the application of geometric limitations during inference may significantly increase efficiency and performance. Their proposed architecture is shown in Figure 3.15. Their developed architecture consists of two sub-networks, including a view synthesis network (VSN) and a stereo matching network (SMN). Their proposed VSN synthesizes the right image of the stereo pair via the left image. In SMN, the

simultaneous application of left and synthesized right images occurs in an encoder-decoder architecture pipeline to achieve a disparity map. In SMN, GTD maps are used to calculate the loss for estimated depth maps.

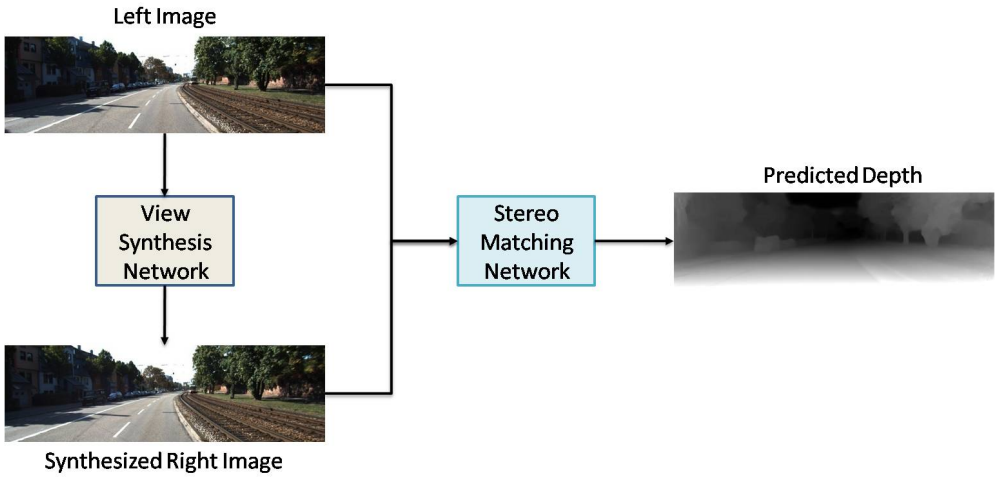


Figure 3.15: Components/inputs of developed semi-supervised loss function by Luo et al. (Luo et al., 2018).

Cho et al. (Cho et al., 2019) developed a novel teacher-student learning strategy to train an MDE network in an SSL approach. Their proposed procedure is demonstrated in Figure 3.16. They first introduced a stereo-matching network with GT-labeled data and permitted the teacher network to estimate depth from stereo pairs of an extensive unlabeled dataset. Then, they applied the aforementioned estimated depth maps/unlabeled dataset to train an optimized student network for MDE (Cho et al., 2019). They also investigated the trade-off between the precision and the density of pseudo-labeled depth maps. The density increases as the pixels in the depth map increase. They concluded the increment of the pseudo-labeled depth maps' precision by enhancing the density. Additionally, they reported that their MDE network achieved the greatest accuracy when the density of pseudo-labeled depth maps was almost 80% (Cho et al., 2019).

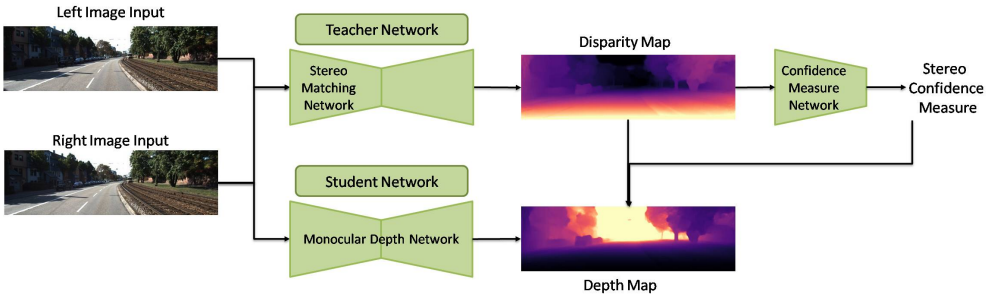


Figure 3.16: Developed network by Cho et al. (Cho et al., 2019).

3.7 Discussion

Due to the ability of humans to use theoretical-based information about the world, estimating depth maps from a single image may be easy for them (Zhao et al., 2020). Relying on the aforementioned fact, former investigations obtain MDE via mixing some old data, such as the communication between some geometric structures (Khan, Salahuddin, and Javidnia, 2020; Fu et al., 2018; Hoiem, Efros, and Hebert, 2005; Masoumian et al., 2021). Due to the acceptable efficacy of image processing, CNN has illustrated a powerful capability to precisely predict dense depth maps from single images (Eigen, Puhrsch, and Fergus, 2014; Dijk and Croon, 2019). In recent years, numerous researchers have studied different types of cues of depth networks required for MDE according to four corroborated procedures, including MonoDepth, SfMLearner, Semidepth, and GCNDepth (Godard, Mac Aodha, and Brostow, 2017; Zhou et al., 2017; Kuznietsov, Stuckler, and Leibe, 2017; Masoumian et al., 2023). Deep neural networks are identified as black boxes. In this black box, the supervised signals are applied to accelerate the learning process of some structural information for depth inference. The lack of sufficient datasets with ground truth due to their high economic cost can be considered one of the most critical DL problems. Table 3.6 aims to represent comprehensive information about the existing procedures based on their training data, supervised signals, and contributions.

Table 3.6: Comprehensive information about the applied procedures in the DL of MDE.

Ref	Training Set	SL	SSL	UL	Major Contribution
Mousavian *et al.* (Mousavian, Pirsiavash, and Košecká, 2016)	RGB + Depth	✓			Multi-task (Semantic + Depth)
Jung *et al.* (Jung et al., 2017)	RGB + Depth	✓			Adversarial Learning, global-to-local
Mayer *et al.* (Mayer et al., 2016)	RGB + Depth	✓			Multi-task (Optical flow + Depth)
Laina *et al.* (Laina et al., 2016)	RGB + Depth	✓			Residual learning, BerHu loss
Kendall *et al.* (Kendall et al., 2017)	Stereo sequences + Depth	✓			End-to-end learning
Fu *et al.* (Fu et al., 2018)	RGB + Depth	✓			Ordinal regression
Facil *et al.* (Facil et al., 2019)	RGB + Depth	✓			Multi-scale convolution
Wofk *et al.* (Wofk et al., 2019)	RGB + Depth	✓			Lightweight network
Garg *et al.* (Garg et al., 2016)	Stereo sequences		✓		Image reconstruction, CNN
Chen *et al.* (Chen et al., 2016)	RGB + Relative depth annotations		✓		The wild scene dataset
Godard *et al.* (Godard, Mac Aodha, and Brostow, 2017)	Stereo sequences		✓		Left-right consistency
Kuznietsov *et al.* (Kuznietsov, Stuckler, and Leibe, 2017)	Stereo sequences + LiDAR		✓		Direct image alignment
Ramirez *et al.* (Ramirez et al., 2018)	Stereo sequences + Semantic label		✓		Semantic prediction
Pilzer *et al.* (Pilzer et al., 2018)	Stereo sequences		✓		Cycled generative network
Aleotti *et al.* (Aleotti et al., 2018)	Stereo sequences		✓		Generative adversial network
He *et al.* (He et al., 2018)	Stereo sequences + LiDAR		✓		sparse optimization
Fei *et al.* (Fei, Wong, and Soatto, 2019)	Stereo sequences + IMU + Semantic label		✓		Physical information
Li *et al.* (Li et al., 2018)	Stereo sequences		✓		Absolute scale recovery
Zhao *et al.* (Zhao et al., 2019)	Stereo sequences + Synthesized Depth		✓		Domain adaptation
Wu *et al.* (Wu et al., 2019)	Mono sequences+LiDAR		✓		Attention mechanism
Zhou *et al.* (Zhou et al., 2017)	Mono sequences			✓	ego-motion framework
Wang *et al.* (Wang et al., 2019)	Stereo sequences			✓	Multi-task (Optical flow + Depth)
Zhan *et al.* (Zhan et al., 2018)	Stereo sequences			✓	Deep feature reconstruction
Chen *et al.* (Chen, Schmid, and Sminchisescu, 2019)	Mono sequences			✓	Connecting flow, depth, and camera
Gordon *et al.* (Gordon et al., 2019)	Mono sequences			✓	Camera intrinsic prediction
Li *et al.* (Li et al., 2019)	Mono sequences			✓	Sequential adversarial learning
Almalioglu *et al.* (Almalioglu et al., 2019)	Mono sequences			✓	Generative adversial network
Godard *et al.* (Godard et al., 2019)	Mono sequences			✓	Left-right consistency
Shu *et al.* (Shu et al., 2020)	Mono sequences			✓	Feature metric
Masoumian *et al.* (Masoumian et al., 2023)	Mono sequences			✓	Graph multi layer

3.7.1 Accuracy

To achieve high accuracy, several factors are involved. The first factor is using the supervised or unsupervised model. Our evaluation proves that SL methods achieved higher accuracy than UL and SSL methods due to labeling the original ground truth. However, collecting a large dataset of monocular videos with accurate depth maps is a challenging task. Therefore, we can consider that unsupervised methods perform better than supervised methods if we neglect the slight difference in precision against the time for labeling data. Another factor is the frameworks of the developed networks. For instance, developing a DL model, such as graph convolution (Masoumian et al., 2023), 3D convolution (Godard et al., 2019), and 3D geometry constraint (Shu et al., 2020) outperforms other DL methods for DE. The last factor can be the loss of functions. There is some lack of information from monocular videos, such as scale

inconsistency and scale ambiguity. One of the solutions for that is using semantic information and smooth loss to learn the scales. However, increasing the loss of functions will create more complicated networks and cause more computational time.

3.7.2 Computational Time

Computational times depend on the number of parameters of the whole network. The complex networks can predict high-quality and accurate depths, but this will cause them to not be considered in real-time applications due to the increased consumption power requirement. One of the best ways to reduce the computational time is to use pre-trained models such as ResNet (He et al., 2016) or DenseNet (Huang et al., 2017) for feature extractions, and the model can focus only on the decoder part of the network. Table 3.7 represents the comparison of complex and lightweight models developed so far for MDE based on the NYUDv2 dataset. As shown in Table 3.7, there is a kind of trade-off between the accuracy and the complexity of the models. The complex models (Bhat, Alhashim, and Wonka, 2021) (e.g., (Sheng et al., 2022) with 77 million parameters (params) and 186 G floating-point operations per second (FLOPs)) require higher computational time and with a large number of trained parameters; however, they give a more accurate DE. On the contrary, lightweight models (e.g., (Sheng et al., 2022) with 1.7 million params and 1.5 G FLOPs) require low computational time with a low number of trained parameters. Still, the accuracy is lower than in complex models. In addition, the resolution of the resulting depth images is an essential key for increasing or decreasing the computational resources for the developed MDE models.

3.7.3 Resolution Quality

Computing a high-resolution DE is one of the main challenging tasks for researchers. Most of the current DE methods are suffering from this, and their results are not

Table 3.7: Comparison of complex and lightweight models based on the NYUDv2 dataset.

Group	Method	Resolution	FLOPs	Params	REL	RMS	RMSE-Log	δ<1.25	δ<1.25^2	δ<1.25^3
Complex	Hu et al. (Hu et al., 2019)	228 × 304	107G	67M	0.130	0.505	0.057	0.831	0.965	0.991
	Chen et al. (Chen, Chen, and Zha, 2019)	228 × 304	150G	258M	0.111	0.420	0.048	0.878	0.976	0.993
	Yin et al. (Yin et al., 2019)	384 × 384	184G	90M	0.105	0.406	0.046	0.881	0.976	0.993
	Lee et al. (Lee et al., 2019)	416 × 544	132G	66M	0.113	0.407	0.049	0.871	0.977	0.995
	Bhat et al. (Bhat, Alhashim, and Wonka, 2021)	426 × 560	186G	77M	0.103	0.364	0.044	0.902	0.983	0.997
Light Weight	Wofk et al. Wofk et al., 2019	224 × 224	0.75G	3.9M	0.162	0.591	-	0.778	0.942	0.987
	Nekrasov et al. (Nekrasov et al., 2019)	480 × 640	6.49G	2.99M	0.149	0.565	-	0.790	0.955	0.990
	Yin et al. (Yin et al., 2019)	338 × 338	15.6G	2.7M	0.135	-	0.060	0.813	0.958	0.991
	Hu et al. (Hu et al., 2021)	228 × 304	14G	1.7M	0.138	0.499	0.059	0.818	0.960	0.990
	Sheng et al. (Sheng et al., 2022)	228 × 304	1.5G	8.2M	0.135	0.488	0.057	0.831	0.966	0.991

satisfied in reality. Based on the discussed training manners, it is evident that SL models (Eigen, Puhrsch, and Fergus, 2014; Qi et al., 2018; Ummenhofer et al., 2017) achieved higher quality resolution of depth maps than other models, such as UL (Zhan et al., 2018; Garg et al., 2016; Zhou et al., 2017).SSL (Kuznietsov, Stuckler, and Leibe, 2017; Luo et al., 2018; Xie, Girshick, and Farhadi, 2016), because training the models with original ground truth helps the model to learn more accurately with higher quality resolution. However, one of the solutions for improving the resolution quality is to use super-resolution color images for training. However, this requires creating a new dataset which is expensive and time-consuming. In addition, the processing of high-resolution images/videos needs high computational resources that increase the cost, and obtaining high-resolution depth maps and computational resources is a trade-off.

3.7.4 Real-Time Inference

For using the MDE methods in industrial applications, it is very important that the model can perform in real-time. There is a negative correlation between real-time performance and the complexity of the network, as shown in Table 3.7. Therefore, for better performance in real-time applications, lightweight MDE networks are required. However, researchers need to consider that lightweight networks sometimes reduce the accuracy and resolution of the predicted depth maps.

3.7.5 Transferability

Some networks are limited to working on the exact scenarios or environments, making them useless for other types of datasets. The transferability will make them more useful for different scenarios, cameras, and datasets. Training and testing the methods on different datasets, using domain adoption technology and 3D geometry, will improve the transferability of the models, and that will cause them to become more valuable in real life.

3.7.6 Input Data Shapes

As discussed earlier in Section 5, there are three types of input data: mono-sequence (Zhou et al., 2017; Mahjourian, Wicke, and Angelova, 2018; Masoumian et al., 2023), the stereo sequence (Kuznietsov, Stuckler, and Leibe, 2017; Godard, Mac Aodha, and Brostow, 2017), and sequence-to-sequence (CS Kumar, Bhandarkar, and Prasad, 2018; Mancini et al., 2017). The mono-sequence input shapes models receive a single image as an input and provide a single output. These types are most commonly used in UL models. On the contrary, stereo-based models receive left and right pairwise images as inputs (i.e., one pair of images is used as a target image for UL) and provide a single output as depth maps. These input shapes are mainly used for UL and SL models. The last type, sequence-to-sequence, is necessary for RNN models. These types receive a series of images as input and provide a sequence of depth maps as output. Due to the simplicity of the resources for mono-sequence and sequence-to-sequence models, which require a single camera compared to the stereo models, which require at least a pair of cameras, it is more economical to use mono-sequence or sequence-to-sequence models. On the other hand, sequence-to-sequence models require higher computational resources to train the model than mono-sequence models, since they need to process a sequence of images. Therefore, the most suitable models regarding low cost and computational resources are mono-sequence models.

3.7.7 Future Study

The current DL methods (Zhou, Greenwood, and Taylor, 2021; Masoumian et al., 2023; Godard et al., 2019) have achieved the best performance so far. However, there is still no unit network that can predict a depth with high accuracy and resolution using low computational resources and without needing the actual ground truth. Therefore, future studies can create lightweight networks working on limited-memory devices without reducing the quality and resolution of predicted depth. In addition, the developed models should achieve higher accuracy under UL models to remove the original ground truth from training and create a self-adaption network for 3D reconstruction. Currently, the main challenges of MDE are that most MDE approaches depend on high-resolution images and large-size DL models with a high number of trained parameters that help predict depth maps with high accuracy. However, these models cannot be worked in real-time applications because they require high computational time and resources. On the contrary, lightweight networks are more useful for real-time applications and can be executed on devices with limited resources. However, reducing the networks' complexity will significantly degrade the results' quality and accuracy. Therefore, there is still a gap and limitation in this area to be discovered and solved.

Accurate real-depth annotations are difficult to acquire, needing special and expensive devices such as a LiDAR sensor. Self-supervised DE methods try to overcome this problem by processing video or stereo sequences, which may not always be available. Therefore, for DE, the researchers need to cope with the issue of domain adaption that will help train an MDE model using a fully annotated source dataset and a non-annotated target dataset. Additionally, although the MDE networks can be trained on an alternative dataset to overcome the dataset scale problem, the trained models cannot generalize to the target domain due to the domain discrepancy. For instance, there is no general MDE network that can still correctly predict the depth maps from

day and night or indoor and outdoor images. In addition, most advanced MDE methods fail to predict accurate depth maps with adverse weather conditions (fog, sunny, snow, etc.). Therefore, the future study requires a complete dataset to include day and night or indoor and outdoor images with different weather conditions.

3.8 Conclusions

DL techniques possess great potential to predict depth from monocular images. Implementation of depth prediction from monocular images is possible using an efficacious DL network structure and a dataset appropriate for the technique applied in learning. This chapter presented a comprehensive overview of the contribution of this growing area of science in deep-learning-based MDE. Therefore, an extensive review of SOTA studies in MDE has been conducted, encompassing various aspects such as data input types, training methodologies, and the application of SL, UL, and SSL approaches. The review also includes an examination of different datasets utilized in MDE research and the evaluation metrics employed to assess the performance of DE models. By comprehensively analyzing these aspects, a comprehensive understanding of the advancements and progress in MDE can be achieved. Finally, we highlight valuable opinions related to accuracy, computational time, resolution quality, real-time inference, transferability, and input data shapes, opening new horizons for future research. This chapter demonstrates that the networks could train for various representation problems. In future perspectives, the architecture of DL models has to be improved to enhance the precision and reliability of the proposed networks and decline their inference time. Additionally, MDE networks have brilliant potential to be used in autonomous vehicles if high reliability is obtained. In addition, they must have the capability to output real-time depth maps.

In the next chapter, a new method for MDE based on GCN, which represents the SOTA approach in the field, will be introduced. This chapter presents a detailed

explanation of the proposed method, including the architecture, training methodology, and evaluation metrics employed. The performance of the GCN-based MDE model will be extensively evaluated on various datasets to demonstrate its superiority over existing techniques. Emphasizing the key advantages of the GCN-based approach, such as improved accuracy and enhanced handling of non-Euclidean data, this chapter explores its potential applications in robotics and autonomous vehicles. Overall, the proposed method utilizes GCN to handle the convolution of non-Euclidean data, allowing for irregular image regions within a topological structure. The model consists of two parallel autoencoder networks, one of which depends on ResNet-50 and multi-scale GCN to estimate the depth map, while the other estimates the ego-motion vector between two consecutive frames. A combination of loss functions is used to cope with bad depth prediction and preserve object discontinuities. The proposed method significantly improves both the quantitative and qualitative performance, with a high prediction accuracy of 89% on the KITTI dataset (Geiger, Lenz, and Urtasun, 2012) and a 40% reduction in the number of trainable parameters compared to the SOTA solutions. The trained model was also evaluated on a new dataset with low-resolution images, and the source code is publicly available at https://github.com/ArminMasoumian/GCNDepth.git

UNIVERSITAT ROVIRA I VIRGILI
ENHANCING DISTANCE PREDICTION THROUGH MONOCULAR DEPTH ESTIMATION BASED ON GRAPH CONVOLUTIONAL NETWORKS
Armin Masoumian

Chapter 4

GCNDepth: Self-supervised Monocular Depth Estimation Based on Graph Convolutional Network

The aim of this chapter is to provide a new solution that uses GCN for self-supervised MDE. The proposed model consists of two parallel auto-encoder networks, with the first utilizing ResNet-50 and multi-scale GCN to estimate the depth map, while the second estimates the ego-motion vector using ResNet-18. A combination of loss functions is used to cope with bad depth prediction and preserve object discontinuities. The proposed method achieves comparable and promising results, with a high prediction accuracy of 89% *on the KITTI dataset and a* 40% *reduction in trainable parameters compared to SOTA solutions. The trained model is also tested on the Make3D dataset, demonstrating its ability to perform well on a new dataset with low-resolution images.*

4.1 Introduction

In the Artificial Intelligence (AI) field, DL networks have accomplished high performance in various DE and ego-motion prediction tasks, and nowadays, it is steeply expanding. The importance of DE, as a pull factor for the entry of modern technologies into self-driving vehicles (Badue et al., 2021; Daily et al., 2017), object distance prediction (Masoumian et al., 2021), helping impaired/blind people, is targeting the improvement of the quality and productivity in the day-to-day life of humankind. The DE based on DL can be utilized in simultaneous localization and mapping (SLAM), navigation, object detection, and semantic segmentation (Zhao et al., 2020).

The stereo vision system is one of the common techniques for DE (Laga et al., 2020; Ming et al., 2021; Gan et al., 2021). However, in order to save costs and computational resources, many methods have been presented to perform DE based on a monocular camera. The MDE methods can be divided into two categories in terms of the learning approach: SL methods (Eigen, Puhrsch, and Fergus, 2014; Mayer et al., 2016) and UL methods (Garg et al., 2016; Xie, Girshick, and Farhadi, 2016). The primitive works focused on studying the extent of depth prediction with deep supervised networks. Nevertheless, gathering extensive and accurate datasets and GTD for training supervised models is a difficult task (Eigen, Puhrsch, and Fergus, 2014), especially for developing a ground truth with high resolution and quality. Besides, costly components such as 2D/3D LiDAR sensors are needed to capture depth maps. Thus, many works, such as (Noraky and Sze, 2019), used time-to-flight cameras to reduce the power for depth sensing to reduce the cost.

Therefore, many DE works were proposed based on UL to avoid collecting real depth maps. Most UL methods are used to estimate both depth and camera ego-motion (Godard, Mac Aodha, and Brostow, 2017) to reconstruct the target frame. The main idea is to receive a sequence of frames as an input and to minimize the error between a warped frame and the target one. The warped frame is obtained from an

adjacent one, predicted depth, and relative camera motion of the target frame (Godard et al., 2019).

Most existing DL MDE networks use CNN to extract the feature information and construct the depth images (Abdulwahab et al., 2020; Abdulwahab et al., 2021). However, CNN is limited, since it does not consider the characteristics of the geometric DI and object location, as well as contextual features in the scene. Besides, there is recently a need to extend deep neural models from Euclidean domains achieved by CNNs to non-Euclidean domains (Bronstein et al., 2017). Thus, the research community has started to observe the importance of DL networks based on graphs (Kipf and Welling, 2016). The effectiveness of the graph convolution network (GCN) has been proved in processing graph data on the tasks of classification (Liang, Deng, and Zeng, 2020) and segmentation (Zhang et al., 2019). For self-supervised MDE, there are few works based on GCN, such as depth prediction (Fu, Liang, and Wang, 2019). The DL model proposed in (Fu, Liang, and Wang, 2019) consists of two stages. The first stage is to use an autoencoder network based on CNNs to estimate the coarse depth and extract latent features of the input image. The second stage is a reconstruction network based on GCNs to refine the predicted depth map. Nonetheless, using two consequent networks leads to an increase in the complexity of the proposed model. Thus, in this work, we propose a novel one-stage architectural DL network based on GCN, the so-called GCNDepth, that can help advance MDE.

In general, the two main contributions are summarized as follows:

- We introduce an innovative autoencoder, denoted as CNN-GCN, designed for MDE. In this architecture, the encoder network leverages ResNet (He et al., 2016) as a foundational backbone to extract pivotal features from the input frames. A decoder network then utilizes the structure of the GCN through the whole decoding process to improve the accuracy of depth maps by learning the nodes' (i.e., pixels) representation via constructing the depth maps via iteratively

propagating neighbor's information until reaching a stable point.

- To widely exploit the diverse spatial correlations between the pixels at multiple scales and to refine the final estimated depth image by preserving the global information that crosses the coarser feature maps and detailed local information passed from lower layer feature maps, we propose a multi-level depth predictor based on GCN in each layer of the decoder network. The updated graph is fed to the next GCN layer in the decoder network.

In addition, for training the proposed model, we utilize a combination of different loss functions, related to photometric (Zhao et al., 2016), reprojection (Godard et al., 2019), and smoothness (Shu et al., 2020) to improve the quality of predicted depth maps. The reprojection loss is used to cope with the object's occlusion, and the photometric loss is proposed for feature reconstruction to reduce the losses between the target and reconstructed images. In turn, smoothness loss is used to preserve the edges and boundaries of the objects and reduce the effect of texture regions on the estimated depth.

This chapter is organized as follows, Section 4.2 reviews the background and related works on MDE, and a detailed explanation of the proposed model is described in Section 4.3. The validation of our system through experimental results is given in Section 4.4 and Section 4.5 represents the conclusion of this research.

4.2 Background and Related Work

MDE can be widely categorized into supervised and self-supervised DL. In this section, we present a brief review of both supervised and self-supervised based methods. Additionally, we will present DE based on GCN networks.

4.2.1 Supervised Depth Estimation

Single image DE is an intrinsically ill-posed dilemma: a single input image can project multiple feasible depth maps. SL methods proved that fitting the relation between color images and their corresponding depth maps by learning ground truth can solve the problem of MDE. Diversified approaches have been explored for solving this problem. Fu et al. (Fu et al., 2018) proposed a multi-layer deconvolution network for obtaining high-resolution depth maps. However, this will require a high computation system for training and create a complicated network. Alhashim et al. (Alhashim and Wonka, 2018) proposed a DL method via transfer learning to reduce the computation time and network complexity. Their technique contains a standard CNN autoencoder (i.e., encoder-decoder) architecture to estimate high-quality depth maps. However, their depth maps have low pixel resolution, leading to missing DI in many regions in complex scenes. All fully supervised training approaches require RGB images and the corresponding depth maps as ground truth. However, finding and collecting the original ground truths for supervised training is one of the main limitations. Chen et al. (Chen et al., 2021) proposed an attention-based context aggregation network (ACAN) to tackle the continuous context information capturing problems. Their network improved in detecting the sharp boundaries in the resulting depth maps, but still, they need an original ground truth for labeling and training their model. Real ground truth can be delicately collected from LiDAR sensors or be rendered from simulation engines (Mayer et al., 2016). However, the LiDAR sensors limit allocating to new vision sensors and rendering real scenes (Shu et al., 2020). In general, creating or collecting datasets with accurate depth maps for SL models is still challenging. In addition, most SL models are excellent for specific-purpose environments involved in the datasets, but they can not be easily generalized to different environments. Thus, in this work, we will depend on an unsupervised DL model to estimate the depth maps to cope with the problem of collecting GTD maps and discovering hidden and

interesting patterns in unlabeled images.

4.2.2 Self-supervised Depth Estimation

As an alternative to the absence of ground truth, self-supervised models can be trained by comparing a target image to a reconstructed image as a supervisory signal. Image reconstruction can be achieved either by stereo training or monocular training.

Stereo training uses synchronized stereo pairs of images and predicts the disparity pixel between the pairs (Xie, Girshick, and Farhadi, 2016). There are various DE approaches based on stereo pairs. For instance, Garg et al. (Garg et al., 2016) introduce a predicting continuous disparity feature matching framework, which does not require a pre-training stage or annotated ground-truth depths. Their methods consist of a deep CNN autoencoder with inverse warping. However, the generated view synthesis is only a proxy for depth and may not always yield high-quality predicted depth. In order to improve the constraints of the depth prediction, DL networks need to predict the camera pose between the two consecutive frames (or left-right pairs) during training. Consequently, Babu et al. (Babu et al., 2018) used an autoencoder network and a temporal photometric warp error for 6 DoF camera pose estimation (ego-motion) and depth maps. Most of the approaches mentioned above depended on standard loss function, such as the L1/L2 norm that yields blurred depth maps. Thus, many trials used robust loss functions to preserve the edges in the predicted depth images. For instance, Zhang et al. (Zhang et al., 2020) proposed a hybrid geometric-refined loss function to explore a more accurate geometric relationship between the input color image and the predicted depth map and preserve depth boundaries and fine structures in depth maps. These approaches rely on geometry to estimate depths from triangulation which often ignores monocular cues (e.g. linear perspective, texture gradients, familiar sizes) (Masoumian et al., 2022).

In turn, MDE approaches, such as using enforced edge consistency (Yang et al.,

2018a), multi-layer feature fusion CNN (Lei et al., 2021), and adding a depth normalization layer as smoothness term (Godard, Mac Aodha, and Brostow, 2017), have achieved high performance compared to the stereo pair training. For instance, Wang et al. (Wang et al., 2021) proposed a multi-task network based on differentiable direct visual odometry, which is fused with an appearance-matching loss to predict depth maps. However, this multi-task model reduces the quality of the predicted depths. Some self-supervised training also makes assumptions about material properties and appearance, such as the brightness constancy of object surfaces between frames (He et al., 2021). For instance, Liu et al. (Liu et al., 2021) used domain separation to relieve the illumination variation between day and night images. However, their model needs to learn an illumination-invariant feature space. Most of the models mentioned above tackled the problem self-supervised by learning the depth map based on the photometric error and adopting differentiable interpolation (Zhou et al., 2017; Yin and Shi, 2018; Wang et al., 2018a; Mahjourian, Wicke, and Angelova, 2018; Godard et al., 2019; Gordon et al., 2019) as loss functions. However, these methods often fail to represent the depth boundaries of objects. This problem happens because of an inefficient decoding scheme that causes blurring artifacts at the depth boundary. Consequently, in order to preserve the objects' boundaries and the small details in the predicted depths, new reconstruction strategies are required to build non-Euclidean DI. It is worth mentioning that networks such as Graph Neural Networks (GNN) can be used to adapt existing MDE approaches to directly process and build non-Euclidean structured depth maps.

4.2.3 Graph Neural Network

Most self-supervised MDE methods such as (Godard et al., 2019; Shu et al., 2020) depend on CNN-based autoencoder networks to extract visual features from whole scene images and estimate the depth of images. However, in most cases, CNN-based

networks yield blurred edges and boundaries of the objects. Here, GNN can help capture the dependencies among objects, and GNN can also extract object-based location features from the scene. CNN and GNN models are similar in weight sharing; the main difference is their data structure. Regarding work on data with underlying non-regular structures (irregular on non-Euclidean structured data), the GNN performs better than CNN because the model learns the features by inspecting neighboring nodes. The basic idea behind most GNN architectures is graph convolution networks (GCN). The GCN models can learn feature representation even before training because of the adjacency matrix, which helps the model understand adjacent nodes' characteristics (Singh and Sharma, 2012). There are two main types of GCNs: Spectral Convolution and Spatial Convolution. The spectral convolution networks use Eigen decomposition of the Laplacian Matrix of the graph. In contrast, spatial convolution networks use the local neighborhood of nodes and understand the properties of a node based on its k local neighbor, which helps to reduce the computing time significantly without reducing the performance. For the first time, for an SSL node classification task on graphs, GCN was proposed by Kipf et al. (Kipf and Welling, 2016) for a learning method for the target node to propagate the neighboring information through CNNs (ConvGNNs). Their primary purpose of using GCN was to reduce the number of lost features during the feature extraction, help the model learn small details, and predict better results. Their model employed a propagation rule based on graphs' first-order approximation of spectral convolutions. However, this method requires high computational resource consumption depending on the input data size. Another example of a graph-based model based on CNNs was proposed in (Estrach et al., 2014) by arranging the adjacent nodes' information with convolution based on spectral graph theory. However, this causes the loss of many nodes of the image when 3D objects are mapped in 2D planes.

There are not many works using GCN in MDE. Fu et al. (Fu, Liang, and Wang, 2019) created a topological depth graph from a coarse depth map based on spatial

graph theory, and they used this graph as a depth clue in their model to avoid depth node losses. However, this technique generates a topological depth graph from a coarse depth map obtained from a pre-trained DE model; thus, their approach consists of two consequent networks. That increases the complexity of the model. In this work, we propose a self-supervised CNN-GCN auto-encoder for MDE to solve the above-mentioned problems. The reason for using GCN as a decoder network is to improve the detection of sharp boundaries, reduce the background noise, and compute precise depth maps with full object details compared to the self-supervised SOTA models. Based on (Fu, Liang, and Wang, 2019) and the pixel similarity connection, the graph-based decoder will also sharply detect the edges in depth maps and preserve the small details. In our proposed model, we will use one stage network consisting of an encoder to extract the features of the input image-based CNNs and then a decoder based on GCNs (Kipf and Welling, 2016) to predict multi-scale depth maps. In addition, our approach will use a combination of different warping errors proposed in the SOTA, such as the reconstruction error presented in (Zhao et al., 2016) to minimize the errors in the reconstructed image, the photometric reprojection error proposed in (Godard et al., 2019) to optimize the values which provide matching pixel intensities between the target and reconstructed images. Finally, a combination of discriminative and curvature errors (Shu et al., 2020) highlights the geometric characteristics of the objects and textured regions in the scene. We used the spectral GCN model proposed (Kipf and Welling, 2016) as a baseline, but we changed it to be as a spatial GCN in order to reduce the computing time without affecting the quality of our results.

4.3 Method

In this section, we first describe the architecture of the proposed model introducing our GCN model and the whole structure of our self-supervised model, including DE (DepthNet) and pose estimation, i.e., ego-motion (PoseNet) networks. Our method

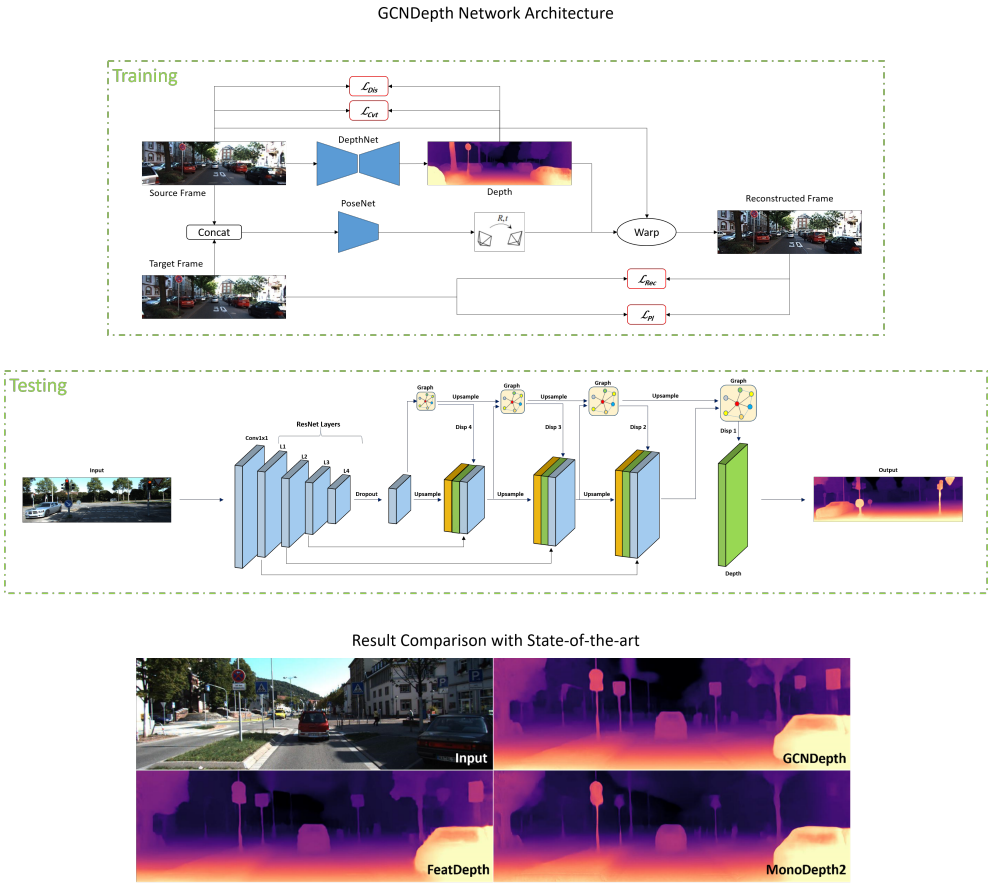


Figure 4.1: Overview of our MDE model based on GCN.

will estimate the depth images and the ego motion to increase the constraints of depth prediction. For MDE, the relationship between object location and visual and contextual features in the scene is significant to preserve the objects' boundaries. Besides, we present the loss functions used for training the model. Fig. 4.1 illustrates the architecture of our GCN model and the result comparison with the SOTA models.

4.3.1 Problem Definition

GCNDepth is a multi-task DL-based system that consists of two parallel networks, DepthNet and PoseNet. If $I \in \mathbb{A}$ represents a monocular RGB image, the problem of generating its corresponding depth image, $D \in \mathbb{B}$, can be formally defined as a function $\Psi_D : \mathbb{A} \longrightarrow \mathbb{B}$ that maps elements from domain $\mathbb{A}$ to elements in its corresponding domain $\mathbb{B}$, as follows:

$$D = \Psi_D(I_s), \tag{4.1}$$

where the proposed model, DepthNet, approximates the prediction of a depth map, D, as a function, Ψ_D, which is fed by a source RGB frame, I_s as an input with pixels p.

Similarly, the problem of estimating the viewpoint between two consequent RGB images can be formally defined as a function $\Psi_E : \mathbb{A} \longrightarrow \mathbb{R}^3$, which is fed by two consequent frames, I_s and I_s as an input and predicts an ego-motion vector, as follows: $E_{I_s \longrightarrow I_t} = [r^T, t^T]$, where $r = [\Delta\theta, \Delta\phi, \Delta\psi]^T$ is is a rotation vector, and $t = [\Delta x, \Delta y, \Delta z]^T$ is a translation vector. The mapping process can be approximated as follows:

$$E_{I_s \longrightarrow I_t} = \Psi_E(I_s, I_t). \tag{4.2}$$

Both the depth and ego-motion vector along with the I_s source frame are used for reconstructing an image, I_{rec} that has to be close to the target image, I_t. Thus, our model, GCNDepth, aims at approximating the total process for estimating depth and pose with the I_{rec} in a final function Ψ that accepts two inputs I_s and I_t, as follows:

$$\Psi(I_s, I_t) = (D, E_{I_s \longrightarrow I_t}, I_{rec}). \tag{4.3}$$

4.3.2 Graph Convolutional Network

One of the main problems with CNN-based networks is that they cannot compute the data of non-Euclidean domains, and they extract appearance features rather than object-location features. The use of DL models based on CNN on complex 3D scenes, such as depth maps estimation, can yield a significant loss in the details of the objects in the scene or even break the topological structure of the scene (Bronstein et al., 2017). Thus, GCN networks that introduce topological structure and node features can increase the feature representation of hidden layers. That helps the model learn how to map the DI from low-dimensional features. Besides, they can represent the topological structure of the scene by describing the relations between objects allowed. Generally, the graph convolution is defined as follows:

$$Z = \sigma(AXW), \tag{4.4}$$

where $\sigma(.)$ defines a non-linear activation function, $\mathbf{A} \in R^{N \times N}$ is an adjacency matrix (i.e., binary matrix), with N being the number of nodes in the given graph that measures the relationship between the nodes in the graph. $\mathbf{X} \in R^{N \times C}$ represents the input N nodes into the graph, and the feature vector dimensionality C, which in our case is high-level (latent) features extracted from the CNN-based encoder. In turn, $\mathbf{W} \in R^{H \times F}$ is the trainable weights, where H is the number of the nodes in the hidden layer and F is the dimensions of the resulting vector. Note that $H = C$ is in the first layer.

To avoid the adjacency matrix changing the scale of the feature vector, we added an identity matrix I to obtain the self-loop as follows:

$$\hat{A} = A + I. \tag{4.5}$$

In our case and regarding the non-linear activation function, for the first layer of

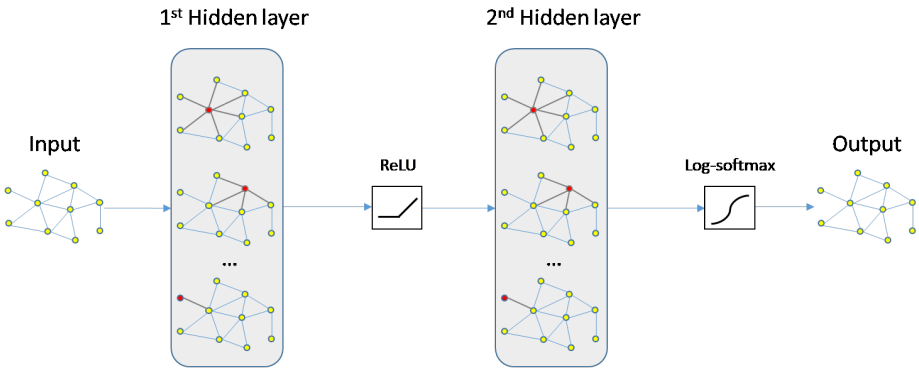


Figure 4.2: An illustration of the proposed GCN module containing two hidden layers.

GCN, the ReLU activation is used to reduce the dependency of the parameters and avoid over-fitting. For the second layer of GCN, Log-Softmax is used to normalize the output of the graph. Figure. 4.2 illustrates the architecture of our GCN model. We randomly initialized the first adjacency matrix of the first graph in the decoder network with the exact size of the nodes in the first layer of the depth decoder. We fine-tuned a parameter, P, which represents the probability for edge creation and the percentage similarity of each node (i.e., vertices) or pixel with their neighbor nodes in the graph. Using $P = 0.7$ with the first random adjacency matrix yields the best-estimated depth maps.

In the end, in order to boost and increase the quality of predicted depth maps, a multi-scale GCN-based is used in the decoder network. This technique combines the feature information of each scale with a depth graph topology.

4.3.3 Self-supervised CNN-GCN Autoencoder

To predict the depth map of a single image, the self-supervised training DE network of our model, DepthNet, is an autoencoder network. The autoencoder network consists of two successive sub-networks: the first one is an encoder that maps the input into high-level feature representation and a decoder that maps the feature representation to

a reconstruction of the depth. In this work, we proposed to use a CNN-based encoder and a GCN-based decoder.

4.3.3.1 DepthNet Encoder

For the encoder network, the input is an image represented as grid-like data, which is regular, and its pixels have the same amount of neighbors. CNNs can exploit the local connectivity and global structure of image data by extracting meaningful local features shared within the input images used during the training stage. Therefore, in our case, CNNs are suitable for extracting global-based visual features from the whole scene shown in the input image. Our encoder network consists of 5 deep layers. The last four layers are standard ResNet-50 (He et al., 2016) blocks. The first layer before the ResNet blocks is a fast convolutional layer, Conv1x1, which consists of a convolution + batch normalization + max-pooling operation. Table 4.1 represents the network details of the encoder network.

Table 4.1: The network architecture of depth encoder. **K** is the number of block repetition, **S** the stride, **Chn** the number of output channel, **Input** corresponds to the input channel of each layer.

Layer	K	S	Chn	Input	Activation
Conv1x1	1	1	64	Img (1024×320×3)	ReLU
ResNet-50 L1	3	1	256	Conv1x1 (512×160×64)	ReLU
ResNet-50 L2	4	1	512	ResNet L1 (256×80×256)	-
ResNet-50 L3	6	1	1024	ResNet L2 (128×40×512)	-
ResNet-50 L4	3	1	2048	ResNet L3 (64×20×1024	SoftMax

4.3.3.2 DepthNet Decoder

Regarding the depth decoder and for large-scale DE, we aim to use a geometric DL network that can help extract object-based location features and keep the relationships between nodes in the resulting depth maps by generating a topological depth graph

in multi-scale. Therefore, we used multi-scale GCN as shown in Figure. 4.3. The adjacency matrix of the initial graph is built based on the number of nodes of the features generated by the last layer of the encoder network.

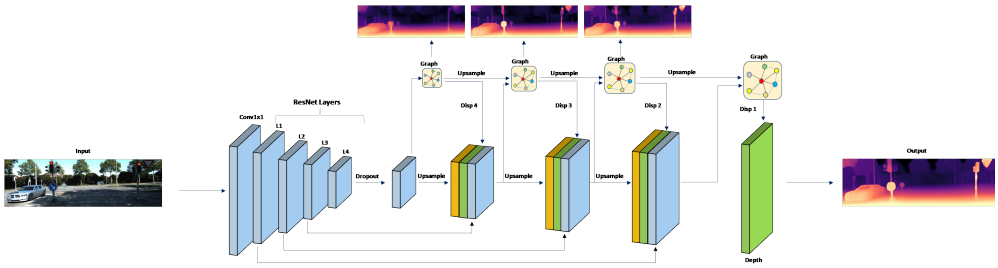


Figure 4.3: Overview of DepthNet network architecture.

Our approach is to use four levels of GCN in constructing the depth images. The main components of the decoder network are 'upconvolution' layers, consisting of unpooling (up-sampling the feature maps, as opposed to pooling) and a transpose convolution that performs an inverse convolution operation. To accurately estimate the depth images, we apply the 'upconvolution' to feature maps and concatenate it with corresponding feature maps from the corresponding layers of the encoder network and an up-sampled coarser depth prediction using the GCN of the previous layer. This approach helps the proposed model preserve the high-level information passed from coarser feature maps and the fine local information provided in lower-layer feature maps. Each step increases the resolution twice. This process is repeated four times, providing a predicted depth map, which is half of the input image. This loop cycle is called multi-scale because, in each layer of our decoder network, the GCN is updated and up-sampled, and is sent to the next layer. The parameters of each layer used in our depth decoder are described in Table 4.2.

Table 4.2: The network architecture of depth decoder. **K** is the kernel size, **S** the stride, **Chn** the number of output channels, **Input** corresponds to the input channel of each layer and ↑ represents upsampling by 2x.

Layer	K	S	Chn	Input	Activation
iL4	3	1	512	L4	Leaky-ReLU
GC4-1	3	1	1	Adj4, iL4	ReLU
GC4-2	3	1	1	Adj4, GC4-1	Log-SoftMax
Disp4	3	1	1	GC4-2	Sigmoid
iL3	3	1	256	L3	Leaky-ReLU
Adj3	3	1	1	↑Adj4	-
Disp4	3	1	1	↑Disp4	-
GC3-1	3	1	1	Adj3, iL3, Disp4	ReLU
GC3-2	3	1	1	Adj3, GC3-1	Log-SoftMax
Disp3	3	1	1	GC3-2	Sigmoid
iL2	3	1	128	L2	Leaky-ReLU
Adj2	3	1	1	↑Adj3	-
Disp3	3	1	1	↑Disp3	-
GC2-1	3	1	1	Adj2, iL2, Disp3	ReLU
GC2-2	3	1	1	Adj2, GC2-1	Log-SoftMax
Disp2	3	1	1	GC2-2	Sigmoid
iL1	3	1	64	L1	Leaky-ReLU
Adj1	3	1	1	↑Adj2	-
Disp2	3	1	1	↑Disp2	-
GC1-1	3	1	1	Adj1, iL1, Disp2	ReLU
GC1-2	3	1	1	Adj1, GC1-1	Log-SoftMax
Disp1	3	1	1	GC1-2	Sigmoid

4.3.3.3 PoseNet Estimator

The pose estimation network is a regression network with encoder and decoder parts. The pose encoder receives a concatenated pair of images, I_s and I_t. Our encoder network consists of 5 deep layers; the first layer is a fast convolutional layer consisting of a 1×1 convolution fed by a concatenation of a pair of images, I_s and I_p, followed by batch normalization and max-pooling. The last four layers are standard ResNet-18 blocks (He et al., 2016), which is similar to our depth encoder with fewer hidden layers. The output of the last layer (i.e., ResNet-18-L4) from the pose encoder is a 512-feature map. In turn, our pose decoder contains four convolution layers. The input of the pose decoder is the output of ResNet-18-L4. Besides, the pose decoder has a convolutional weight in the first layer similar to that proposed in (Godard et al., 2019). The decoder layer parameters are shown in Table 4.3.

Table 4.3: The network architecture of pose decoder. **K** is the kernel size, **S** the stride, **Chn** the number of output channel and **Input** corresponds to the input channel of each layer.

Layer	K	S	Chn	Input	Activation
Out1	1	1	256	ResNet-18 L4	ReLU
Out2	3	1	256	Out1	ReLU
Out3	3	1	256	Out2	ReLU
Out4	1	1	6	Out3	-

4.3.4 Overall Pipelines

The proposed method consists of two main networks. The first network, called DepthNet explained in the previous subsection. The source image is an input of the DepthNet, and the output is the depth map. The second network is PoseNet, a pose predictor to estimate the ego-motion vector of the source and the target images (in our case, a consecutive image). The output of PoseNet is the relative pose between the source and target images. Afterward, a warping process, as proposed in (Godard et al., 2019), is applied to find the corresponding pixels in the adjacent frames through the estimated depth map of the source frame and the camera ego-motion vector, and then synthesize the target frame. These two main networks provide geometry information to provide point-to-point correspondences of the reconstructed image. The whole architecture of our model is illustrated in Figure. 4.4.

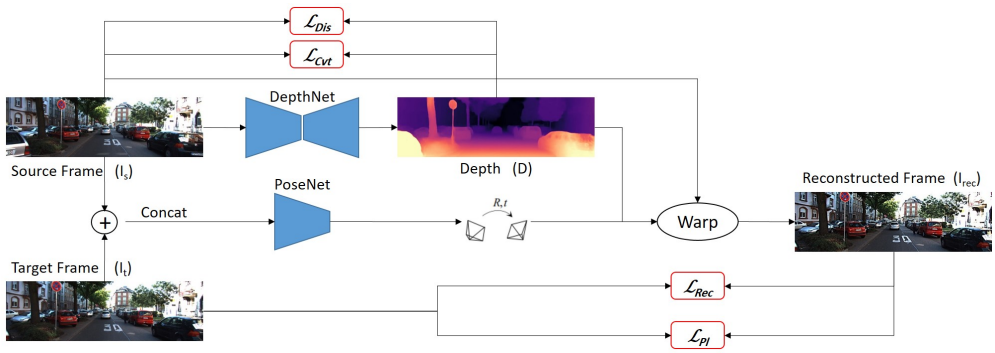


Figure 4.4: Schematic illustration of the whole framework.

4.3.5 Geometry Models and Losses

In monocular video datasets, based on the source frame I_s and the target frame I_t, the reconstructed image I_{rec} can be reconstructed using the resulting depth and the 3D pose. The total loss for the whole network contains three main losses, which penalizes the losses between reconstructed and target images on one side and the resulting depth and the source image on the other side.

The first loss function called the reconstruction loss L_{Rec}, is a common context loss function for an autoencoder network used for constraining the quality of the learned features to reconstruct the target image, as proposed in (Godard et al., 2019). Thus we used the mean square error between the source and the reconstructed images, as:

$$L_{Rec} = \sum_{p} \sqrt{(I_{rec}(p) - I_t(p))^2}. \tag{4.6}$$

Regarding achieving better performance and coping with occlusions between frames in a monocular video, the reconstruction loss L_{Rec} is combined with the reprojection loss, L_{Pl}, which combines the L1-norm and SSIM losses as defined in (Godard, Mac Aodha, and Brostow, 2017).

$$L_{Pl} = 0.15 \sum_{p} |I_{rec}(p) - I_t(p)| + 0.85 \sum_{p} \frac{1 - SSIM(I_{rec}, I_t)}{2} \tag{4.7}$$

In addition, if we consider that the image intensity function obeys the Lambertian shading function, the network should extract gradient-based features corresponding to the object's shapes in the input color image. Handle the depth discontinuity is usually problematic due to occlusion, over-smoothing, and textured regions, the resulting depth map requires a loss function to preserve the edges and boundaries of the objects and degrade the texture effects. Thus, the first and the second derivative of depth images can highlight geometric characteristics of the objects and homogeneous regions in the image (Rashwan et al., 2018). Consequently, to ensure that the learned features

of the input image yield edge-preserving depth maps, a discriminative loss function, L_{Dis}, can be defined to give significant weight to the low-texture regions.

$$L_{Dis} = \sum_{p} e^{-\lambda \nabla^1 I_s(p)} |\nabla^1 D(p)|, \tag{4.8}$$

where, D represents the predicted depth maps at each pixel p, ∇^1 represents the first order derivative at each pixel p, and λ, a weight factor, is empirically set by 0.5 in this work that yielded the highest accuracy.

In addition, the second-order behavior of the surface in a scene is compatible with the curvature measurements of the depth surface relative to the normal at one of its points near this point. Thus, a curvature loss L_{Cvt} can be defined based on the second-order derivative of gradients as proposed in (Shu et al., 2020). L_{Cvt} also keeps the geometric characteristics of the objects and gives a low weight for textured regions:

$$L_{Cvt} = \sum_{p} e^{-\lambda \nabla^2 I_s(p)} |\nabla^2 D(p)|. \tag{4.9}$$

The combination of discriminative and curvature losses is used as a smoothness loss function which can be defined as:

$$L_{Smooth} = \alpha L_{Dis} + \beta L_{Cvt}. \tag{4.10}$$

The α and β are set to $1e-3$ via cross validation as proposed in (Shu et al., 2020).

The final loss can be used for the optimization process of the whole network, and a penalty for a lousy depth prediction is defined as:

$$L_{Final} = L_{Pl} + L_{Rec} + L_{Smooth}. \tag{4.11}$$

4.3.6 Implementation Details

We implemented our method by using the PyTorch framework (Paszke et al., 2017), and the proposed model was trained for 20 epochs with a batch size of 10 with one GTX 1080-TI GPU. The Adam optimizer (Kingma and Ba, 2014) has been utilized with an initial learning rate of 0.0001 and reduced by half after 75% of the total iterations. The pre-trained ResNet-18 and ResNet-50 layers are used for the PoseNet and DepthNet encoders, respectively (Tan and Le, 2019).

4.4 Experiments

In this section, we demonstrate the evaluation performance of our proposed model. To evaluate our approach, we carry out comprehensive experiments on public benchmark datasets such as the KITTI dataset (Geiger, Lenz, and Urtasun, 2012) and Make3D dataset (Saxena, Sun, and Ng, 2008b).

4.4.1 Depth Evaluation on the KITTI Dataset

The KITTI dataset is a vision dataset for depth and pose estimation. The dataset contains 200 videos of street scenes in the daylight captured by RGB cameras and depth maps captured by the Velodyne laser scanner. The synchronized single images from a monocular camera were used and Eigen split (Eigen and Fergus, 2015) with 39810 images for training, 4424 for validation, and 697 images for testing. The image pre-processing method proposed in (Zhou et al., 2017) has been used for removing static frames. The resolution of the images is 1024×320 pixels.

Regarding the evaluation, we used the standard metrics of depth evaluation, such as Absolute and Relative Error (Abs-Rel), Squared Relative Error (Sq-Rel), Root Mean Squared Error (RMSE), and Root Mean Squared Log Error (RMSE-Log). Besides,

we used δt to calculate the accuracy of the estimated depth with different thresholds as proposed in (Geiger, Lenz, and Urtasun, 2012).

The same original input image size is used for evaluation and depth is capped at 80 meters based on the information from the KITTI dataset. Both the input size and output size of images are 1024×320 pixels.

The median scaling introduced by (Shu et al., 2020) is used for predicted depths to match the ground-truth scale. The median scaling is multiplying predicted depth maps by a computed scale factor to match the median with the ground truth. A different scaling factor is calculated for each test image individually.

The proposed framework is compared with the SOTA of self-supervision-based MDE (Godard et al., 2019; Shu et al., 2020; Yang et al., 2018a; Zhou et al., 2017; Yang et al., 2018b; Mahjourian, Wicke, and Angelova, 2018; Yin and Shi, 2018; Zou, Luo, and Huang, 2018; Wang et al., 2018a; Luo et al., 2019; Casser et al., 2019; Meng et al., 2019; Ranjan et al., 2019; Gordon et al., 2019). Where (Zhou et al., 2017; Yang et al., 2018b; Mahjourian, Wicke, and Angelova, 2018; Casser et al., 2019) used DispNet (Mayer et al., 2016) as a backbone for the encoder network. DispNet is a network that uses a standard CNN to build the encoder and decoder network to find the disparity between two successive or stereo images. In turn (Yang et al., 2018a; Gordon et al., 2019; Godard et al., 2019) exploited ResNet-18 and (Wang et al., 2018a; Luo et al., 2019) used VGG as a backbone. In addition, ResNet-50 was used in (Shu et al., 2020; Yin and Shi, 2018; Zou, Luo, and Huang, 2018; Meng et al., 2019; Guizilini et al., 2020; Kim, Kim, and Kim, 2020) and GCNDepth (our proposed model). The performances of our model compared with the SOTA solutions are summarized in Table 4.4. All tested models shown in Table 4.4 are trained with UL and monocular images with a resolution of 1024×320. As shown in Table 4.4, the GCNDepth method achieved the highest performance in terms of Abs-Rel, Sq-Rel, second and third accuracy of (δ_2,δ_3) evaluation metrics. In addition, the proposed method also achieved second-best results in RMSE, RMSE-Log, and first accuracy of (δ_1) with a slight difference

of 0.003 with RMSE-log, and 0.5% with δ_1 compared to the highest results achieved by (Shu et al., 2020). In general, the model of Featdepth (Shu et al., 2020) and our model, GCNDepth, provided comparable results and they outperformed the other tested methods.

Table 4.4: Comparison of different methods on KITTI dataset. The best results are bolded in blue and the second-best results are bolded in red color.

		Lower Better				Higher Better		
Method	**Backbone**	**Abs-Rel**	**Sq-Rel**	**RMSE**	**RMSE-Log**	**δ<1.25**	**δ<1.25^2**	**δ<1.25^3**
SfMLearner (Zhou et al., 2017)	DispNet	0.208	1.768	6.958	0.283	0.678	0.885	0.957
DNC (Yang et al., 2018b)	DispNet	0.182	1.481	6.501	0.283	0.725	0.906	0.963
Vid2Depth (Mahjourian, Wicke, and Angelova, 2018)	DispNet	0.163	1.240	6.220	0.250	0.762	0.916	0.968
LEGO (Yang et al., 2018a)	ResNet-18	0.162	1.352	6.276	0.252	0.783	0.921	0.969
GeoNet (Yin and Shi, 2018)	ResNet-50	0.155	1.296	5.857	0.233	0.793	0.931	0.973
DF-Net (Zou, Luo, and Huang, 2018)	ResNet-50	0.150	1.124	5.507	0.223	0.806	0.933	0.973
DDVO (Wang et al., 2018a)	VGG	0.151	1.257	5.583	0.228	0.810	0.936	0.974
EPC++ (Luo et al., 2019)	VGG	0.141	1.029	5.350	0.228	0.816	0.941	0.976
Struct2Depth (Casser et al., 2019)	DispNet	0.141	1.036	5.291	0.215	0.816	0.945	0.979
SIGNet (Meng et al., 2019)	ResNet-50	0.133	0.905	5.181	0.208	0.825	0.947	0.981
CC (Ranjan et al., 2019)	DispNet	0.140	1.070	5.326	0.217	0.826	0.941	0.975
LearnK (Gordon et al., 2019)	ResNet-18	0.128	0.959	5.232	0.212	0.845	0.947	0.976
PackNet (Guizilini et al., 2020)	ResNet-50	**0.107**	0.802	4.538	0.186	**0.889**	**0.962**	0.981
DualNet (Zhou et al., 2019)	HRNet	0.121	0.837	4.945	0.197	0.853	0.955	**0.982**
SimVODIS (Kim, Kim, and Kim, 2020)	ResNet-50	0.123	0.797	4.727	0.193	0.854	0.960	0.984
Monodepth2 (Godard et al., 2019)	ResNet-18	0.115	0.882	4.701	0.190	0.879	0.961	**0.982**
FeatDepth (Shu et al., 2020)	ResNet-50	0.104	**0.729**	4.481	0.179	0.893	0.965	0.984
GCNDepth (Masoumian et al., 2023)	ResNet-50	0.104	0.720	**4.494**	**0.181**	**0.888**	0.965	0.984

Although the Featdepth model achieved similar results to our model, the GCNDepth model yields a 40% reduction in the number of trainable parameters compared to the Featdepth model. Where the GCNDepth model has trainable parameters of $48,220,954$, in turn, the Featdepth model has $79,681,406$. Since the Featdepth model has an extra deep feature network for feature representation learning to cope with the geometry problem of self-supervision DE. The comparable results show that the use of GCN in reconstructing the depth images can improve the photometric error that appeared in the self-supervision problem without using the feature network as proposed in (Shu et al., 2020).

In addition, our model achieved high performance on the KITTI benchmark evaluation in the SILog and iRMSE metrics and achieved comparable results in the Sq-Rel and Abs-Rel metrics compared to other SOTA of self-supervised methods as shown in Table 4.5. The results shown in Table 4.5 supported that the use of GCN in estimating depth maps from a monocular video can yield depth maps outperforming or matching the SOTA on the KITTI dataset.

Table 4.5: Performance of our model on KITTI public benchmark.

Method	SILog	Sq-Rel	Abs-Rel	iRMSE
GCNDepth (Masoumian et al., 2023)	**15.54**	4.26	12.75	**15.99**
packnSFMHR (Guizilini et al., 2020)	15.80	4.75	**12.28**	17.96
MultiDepth (Liebel and Körner, 2019)	16.05	**3.89**	13.82	18.21
LSIM (Goldman, Hassner, and Avidan, 2019)	17.92	6.88	14.04	17.62

Qualitatively, the comparison of predicted depth results of the proposed model can be seen in Figure 4.5. The first row of Figure 4.5 represents a clear DE of far and small objects with our GCNDepth model compared to the two methods (Shu et al., 2020; Godard et al., 2019). In the second row of Figure 4.5, our method estimates the depth between the consecutive cars and correctly detects the boundaries of the two cars. In the third and fourth rows, our method properly preserves the discontinuities of the objects without any distortion that occurred with the two other methods. In the last row of Figure 4.5, our model is able to detect the human body in its full shape showing the depth of the key points of body parts, such as the head, neck, shoulder, etc. However, the other models proposed in (Shu et al., 2020; Godard et al., 2019), could not be able to detect the head of the human and there are no homogeneous depth values for other body parts. The qualitative results support that GCNDepth can extract precise depth maps and recover the depth of objects with higher precision compared to the baselines (Shu et al., 2020; Godard et al., 2019). The depth maps generated by GCNDepth maintain the boundaries and details of objects that can be clearly realized. In contrast, depth maps resulting from baselines have crumbled boundaries and the objects can not be recognized. The preservation of the objects' discontinuities can help in building more accurate semantic maps and visual-inertial odometry for autonomous vehicles.

4.4.2 Ablation Study

To get a better understanding of the performance of the proposed method, in Table 4.6, we showed an ablation study by changing different components of the proposed

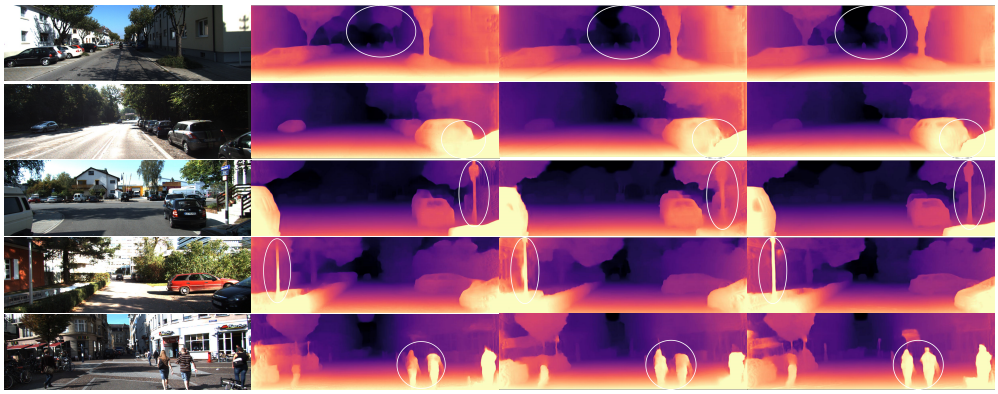


Figure 4.5: Comparison of disparity results on KITTI dataset. (Col.1) original input images and the depth resulted with (Col.2) Monodepth2 (Godard et al., 2019), (Col.3) FeatDepth (Shu et al., 2020) and (Col.4) the proposed GCNDepth model.

model, GCNDepth, as follows:

- Baseline, which is similar to our model with a CNN-based decoder instead of GCN with different losses.
- Single-scale GCN, (SS), where a single scale was added on the first layer of the depth decoder.
- Multi-scale GCN layers (MS) with different losses.
- GCN network with different pre-trained backbones (i.e., ResNet-18 and ResNet-50).

As shown in Table 4.6, we tested our baseline model with three different loss combinations (reconstruction loss (L_{Rec}), photometric loss (L_{Pl}), and smoothness loss (L_{Smooth})), GCN model with three different loss combinations, single scale GCN and multi-scale GCN, and different pre-trained backbones of ResNet-18 and ResNet-50. Adding the photometric loss leads to improving the Asb-Rel with 0.13 and it yields a significant improvement of 8% in the accuracy δ compared to the baseline. Furthermore, adding smoothness loss, besides improving the quality of visual depths, improves

the Asb-Rel by 0.04. The single-scale GCN results were not good compared to the baseline, however, the multi-scale GCN with the three loss functions and using a pre-trained model of ResNet-50 achieved higher results compared to the other variations of the proposed models. Using the ResNet-50 instead of ResNet-18, improved the results and accuracy slightly.

Table 4.6: Ablation results for different components. **SS** represents the single scale GCN and **MS** represent the multi-scale GCN.

Methods and Losses	**Asb-Rel**	**Sq-Rel**	**RMSE**	**RMSE-Log**	δ**<1.25**	δ**<1.25^2**	δ**<1.25^3**
Baseline-Res18 (L_{Rec})	0.132	1.052	5.649	0.237	0.791	0.922	0.959
Baseline-Res18 (L_{Rec}+L_{Pl})	0.119	0.880	4.689	0.204	0.877	0.961	0.981
Baseline-Res18 (L_{Rec}+L_{Pl}+L_{Smooth})	0.115	0.882	4.701	0.190	0.879	0.961	0.982
Ours-MS-Res50 (L_{Rec})	0.111	0.867	5.109	0.198	0.853	0.959	0.980
Ours-MS-Res50 (L_{Rec}+L_{Pl})	0.107	0.748	4.635	0.199	0.881	0.960	0.981
Ours-SS-Res50 (L_{Rec}+L_{Pl}+L_{Smooth})	0.135	0.991	5.148	0.213	0.814	0.939	0.977
Ours-MS-Res18 (L_{Rec}+L_{Pl}+L_{Smooth})	0.105	0.739	4.585	0.191	0.883	0.961	0.982
Ours-MS-Res50 (L_{Rec}+L_{Pl}+L_{Smooth})	**0.104**	**0.720**	**4.494**	**0.181**	**0.888**	**0.965**	**0.984**

Regarding the structure of GCN, we changed the activation function of the GCN layer after the second hidden layer by ReLU or Log-softmax. Besides, we changed the P value for the initialization of the random graph. The experiments showed that multi-scale GCN with a P value of 0.7 (70 percent of similarity) achieved accurate quantitative results than using other values of P, such as $P = 0.1$, 0.3, 0.5, and 0.9. Regarding the activation functions, the proposed final model with multi-scale GCN has achieved the highest score with Log-softmax as an activation function.

4.4.3 Depth Evaluation on the Make3D Dataset

Additionally, we tested the performance of the GCNDepth model on the Make3D dataset using our trained model based on the KITTI dataset. In other words, we used the Make3D dataset purely for validation and testing. The Make3D dataset contains 400 RGB images for training and 134 images for a test set. The results in Table 4.7 show that we outperformed the SOTA of self-supervised methods (Godard et al., 2019; Wang et al., 2018a; Zhou et al., 2017) evaluated on the Make3D dataset in terms of Sq-Rel, RMSE, and RMSE-log metrics of 3.075, 6.757 and 0.107, respectively

without fine-tuning the GCNDepth model with the training set of Make3D. In turn, the Monodepth2 model (Godard et al., 2019) yielded the best Abs-Rel error among the four self-supervised approaches with a value of 0.322. While GCNDepth provided the second-best Abs-Rel error of 0.424. Besides, the GCNDepth model yielded the second-best results after the supervised-based model proposed in (Laina et al., 2016), which provided the best results with differences of 0.22, 1.235, 1.075, and 0.023 of the four metrics: Abs-Rel, Sq-Rel, RMSE, and RMSE-log, respectively. These can be considered promising results compared to the supervised-based approaches.

Table 4.7: Maked3D results. Type **D** represents depth supervision methods and type **M** represents self-supervised mono supervision.

Method	Type	Abs_Rel	Sq_Rel	RMSE	$\log_{10}$
Karsch (Karsch, Liu, and Kang, 2014)	D	0.428	5.079	8.389	0.149
Liu(Liu, Salzmann, and He, 2014)	D	0.475	6.562	10.05	0.165
Laina (Laina et al., 2016)	D	**0.204**	**1.840**	**5.683**	**0.084**
Zhou (Zhou et al., 2017)	M	0.383	5.321	10.47	0.478
DDVO (Wang et al., 2018a)	M	0.387	4.720	8.090	0.204
Monodepth2 (Godard et al., 2019)	M	**0.322**	3.589	7.417	0.201
GCNDepth (Masoumian et al., 2023)	M	0.424	**3.075**	**6.757**	**0.107**

Qualitative results with the Make3D dataset are shown in Figure. 4.6. GCNDepth can estimate depth values even in low texture regions and with different illumination, changes compared to the two other self-supervision models (Shu et al., 2020; Godard et al., 2019). For instance, in the first row of Figure. 4.6, compared to the two other models, the depth map resulting from our model showed that the column of the light in the input image is more visible and with homogeneous depth values and closer to the camera than the other objects (e.g., trees). In turn, the second row of Figure. 4.6 shows that the green view in the image faded into the background in the depth maps from the baselines, but with our model, the green view in the depth image can be clearly recognized, and with boundaries distinguished from the background. In contrast to the other methods in the last row of Figure 4.6, the house can be easily identified in the depth map resulting from GCNDepth. In the graph network, the relationships between nodes are of importance and constitute the path of information transmission in

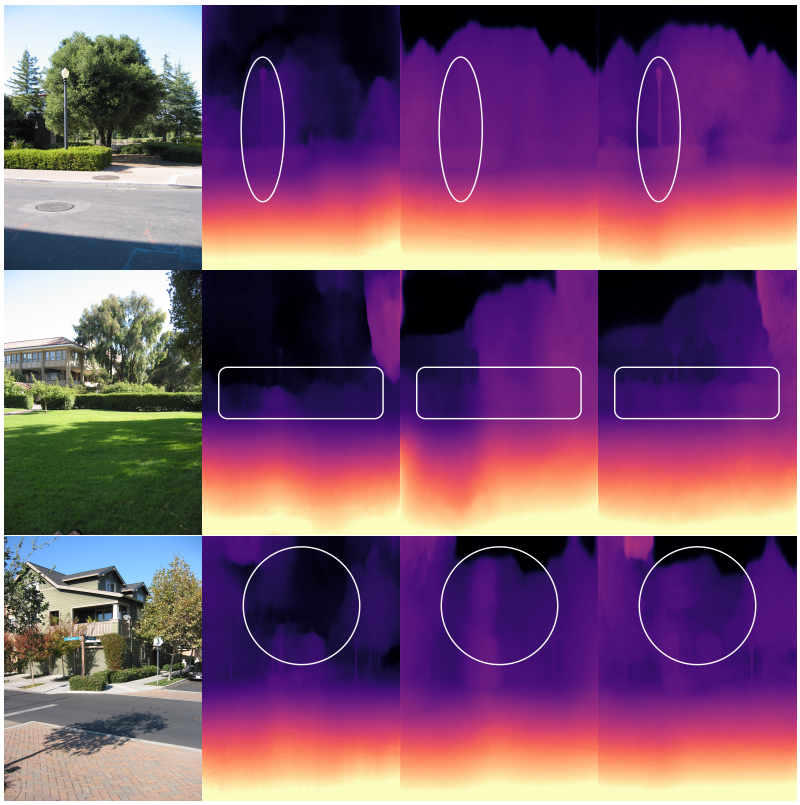


Figure 4.6: Comparison of disparity results on Make3D dataset. (Col.1) original input images and the depth resulted with (Col.2) Monodepth2 (Godard et al., 2019), (Col.3) FeatDepth (Shu et al., 2020) and (Col.4) the proposed GCNDepth model.

GCN. Thus, we believe that the features extracted from GCNs maintain the weights of different objects in the scenes and these features help deal with reconstructing depth maps preserving the discontinuities of the objects. This can possibly improve the performance of reconstructing geometric information for more accurate depth map prediction.

4.4.4 Limitations

Despite achieving comparable and promising results with the GCNDepth model, the model still has some limitations. Firstly, GCN is inefficient in updating the nodes'

hidden states iteratively for a fixed number of the feature vector dimensions. However, we can get a stable representation of the node and its neighborhood by designing a multi-layer GCN as we proposed in this work. Secondly, we must create a random graph with a connection edge probability between each pixel and neighbors for the initial graph. The randomization may increase the training time. Lastly, increasing the number of layers in GCN increases the training time and complexity of the model. In addition, Figure. 4.7 shows that the image's shadow badly affects the estimated depth which cannot show the small details of the objects and accurate boundaries between the objects. The reason is that the shadows cause the region to become textureless in the image, and the similarities between pixels are high. We aim to develop a model that can cope with the illumination distribution in the images.

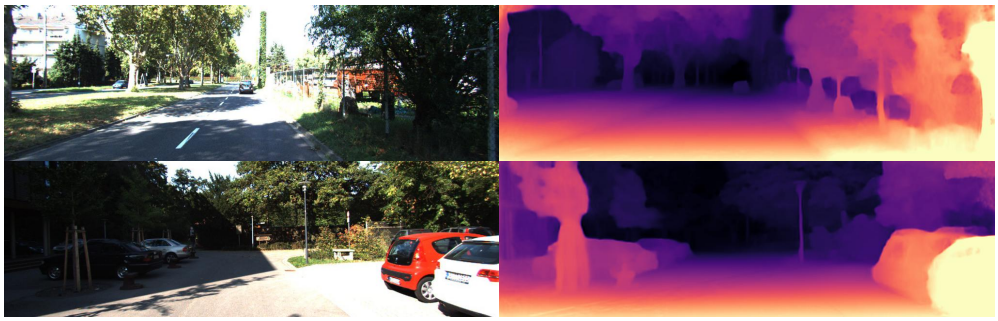


Figure 4.7: Two examples of low-quality predicted depths.

4.5 Conclusion

This chapter presents a self-supervised DL model for MDE based on a multi-scale GCN. The proposed model consists of two networks: 1) depth prediction and 2) pose estimation. The use of GCN in the decoder of the DE auto-encoder can map the DI from low-dimensional features. It can represent the topological structure of the scene by describing the relations between the scene pixels. Besides, to improve the DE, a combination of different loss functions is used I) absolute mean error between the target

image and the reconstruction image, II) perceptional loss to minimize the photometric reprojection error, and III) a combination between discriminative and curvature losses to highlight geometric characteristics of the objects and textured regions in the image. The proposed method achieved a comparable DE from a monocular video single image to the existing KITTI and Make3D datasets. The generated depth maps with GCNDepth depict object edges and boundaries, helpful for semantic maps and visual odometry. The ongoing work is to improve the network that can predict depth maps for night-time images. In turn, future work aims at developing a complete model for pose, depth, and motion estimation from monocular videos.

In the next chapter, a new DL model developed for computing absolute distance will be presented. This model incorporates two parallel networks, one for object detection and the other for MDE. The chapter will provide a detailed explanation of the architecture and methodology used in the development of this innovative model. The performance and accuracy of the model will be evaluated, showcasing its effectiveness in simultaneously performing object detection and MDE tasks. By combining these two crucial aspects, the model offers a comprehensive solution for understanding scenes and computing accurate depth information.

UNIVERSITAT ROVIRA I VIRGILI
ENHANCING DISTANCE PREDICTION THROUGH MONOCULAR DEPTH ESTIMATION BASED ON GRAPH CONVOLUTIONAL NETWORKS
Armin Masoumian

Chapter 5

Absolute Distance Prediction Based on Deep Learning Object Detection and Monocular Depth Estimation Models

The aim of this chapter is to introduce a DL framework for estimating depth images and object detection using a single 2D image. The proposed framework comprises two deep networks: You Only Look Once (YOLOv5) for object detection and a self-supervised deep autoencoder for relative distance estimation via depth images. The object detection network is trained with SL, while the DE network is self-supervised. The framework is evaluated on real images of outdoor scenes and achieves a promising accuracy of 96% *with an RMSE of* 0.203 *of the correct absolute distance.*

5.1 Introduction

For enabling autonomous driving and navigation, one of the main challenges is to achieve reliable and accurate obstacle detection. Many works have been proposed to cope with the problem of obstacle detection (Szikora and Madarász, 2017). Object detection and distance prediction are effectively used in a variety of different fields such as industrial robots (Andhare and Rawat, 2016), research robots (Masoumian et al., 2020b; Nomani et al., 2022), self-driving cars (Agarwal, Chiang, and Sharma, 2019) etc. Regarding object detection, to successfully navigate the environment, it must have knowledge about the objects in its immediate vicinity. Among many sensors available for object detection we are primarily interested in a camera-based vision for indoor/outdoor navigation. Thus, object recognition refers to a collection of related tasks for identifying objects in digital photographs. With the progress of DL networks (e.g., CNN), many accurate methods for object recognition have been developed. For instance, region-based CNN, or R-CNNs (Ren et al., 2015), are a family of techniques for addressing object localization and recognition tasks, designed for model performance. In turn, You Only Look Once, or YOLO (Redmon et al., 2016), is a second family of techniques for object recognition designed for speed and real-time use. Region-based detectors include two stages. Firstly, the model suggests a set of regions of interest (ROIs) by a regional proposal network. Since the potential bounding box candidates can be infinite, the proposed regions are sparse. Secondly, the region candidates are then processed by a classifier. In turn, the one-stage family skips the region proposal stage and directly runs the detection over a dense sampling of possible locations. This yields that the one-stage detectors are faster and simpler, but might potentially reduce the performance a bit. Since YOLO has the advantage of being much faster than other networks in the one-stage family. Besides it achieved comparable results to the SOTA and still maintains accuracy. The predictions depend on the global context of the input image. Consequently, our proposed framework will

be based on the YOLO architecture as a baseline. Regarding distance prediction, it is important to estimate depth maps from the input images. For DE, most computer-vision systems depend on stereo vision by following several time-consuming stages, such as unipolar geometry, rectification, and matching. Alternatively, when stereo vision is not useful or applicable, LiDAR cameras can be used for many applications for mobile robots. However, LiDARs are very costly, and most depth cameras have serious limitations in real environments, such as the synchronization of the optical and imaging elements (Olanrewaju and Popoola, 2017). With the DL spread, many works have been proposed for MDE which is the task of estimating scene depth using a single image. The appearance of objects significantly changes with their pose. Estimating a depth map from a 2D image is an important step in order to determine the 3D pose of the objects present in a scene. MDE based on DL methods can be performed by supervised (Abdulwahab et al., 2020) or unsupervised (Masoumian et al., 2023) learning techniques. Supervised methods perform better accuracy, however, the depth maps of images are needed for training which is difficult to get in real scenarios. On the other hand, unsupervised methods do not require original depth maps, thus, the performance is degraded a bit. Thus, in this chapter, we propose a new framework to predict the absolute distance of each object in 2D images from the camera, based on estimating depth images using self-supervised DL and supervised DL object detection. The contributions of this chapter are:

- Developing a deep object detection model based on two-stage YOLOv5 architecture. A lightweight model is used that can be easily deployed on embedded systems and devices with limited memory and CPU.

- Developing an unsupervised depth and pose estimation DL model based on an autoencoder network.

- Integrating the two models in a framework for absolute distance estimation of obstacles. Integrating the two models will not affect the overall efficiency of the

proposed models, because the two models are structurally independent, and the whole framework is executed by multiple processes, meaning that each model has a separate process responsible for it.

This chapter is organized as follows, Section 5.2 reviews the background and related works on MDE, and a detailed explanation of the proposed model is described in Section 5.3. The validation of our system through experimental results is given in Section 5.4 and Section 5.5 represents the conclusion of this research.

5.2 Related Work

This section aims to provide a comprehensive overview of the SOTA techniques in DE, object detection, and absolute distance prediction systems, offering concise insights into the latest advancements in these fields.

5.2.1 Object Detection

Object detection is a computer vision technique that allows the designed model to locate and identify an object in an image or video by drawing a bounding box around each one of them. It is one of the most challenging issues in the field of computer vision as the object detection model is trained to identify objects within a dataset and it cannot identify an object that is not labeled during the training this is considered one of the limitations. However, trained object detection models can always be retrained again to obtain new knowledge about new objects. Object detection techniques are used in applications like self-driving cars, video surveillance, or crowd counting. There are some popular object detection algorithms like YOLO (Redmon et al., 2016), R-CNN (Ren et al., 2015), and MobileNet (Howard et al., 2017). In this chapter, the YOLO Algorithm has been chosen for object detection. It is considered the SOTA right now and it produced the needed result in the testing phase. The YOLO object

detection model has had several different versions through the years. The YOLOv1 paper was published in 2015 and the subsequent versions were published the next year until it reached YOLOv5 in 2020 it is considered the SOTA due to its good performance and efficiency and constantly being improved.

5.2.2 Absolute Distance Prediction

Computing the absolute distance of objects in a scene from a camera is a crucial task in computer vision. It involves determining the precise distance between the camera and each object in the scene. This information is valuable for a wide range of applications, such as autonomous navigation, augmented reality, robotics, and 3D scene reconstruction.

There are several approaches to computing the absolute distance of objects from a camera:

Stereo Vision: Stereo vision utilizes a pair of cameras with a known baseline separation to capture two slightly different views of the scene. By comparing the disparities (horizontal shifts) between corresponding pixels in the left and right images, it is possible to triangulate and compute the depth using principles of geometry and stereo correspondence (Zaarane et al., 2020).

Time-of-Flight (ToF): ToF cameras emit a short pulse of light or infrared signal and measure the time it takes for the signal to bounce back from objects in the scene. By knowing the speed of light, the time-of-flight is converted into distance information, providing depth measurements for each pixel.

Structured Light: Structured light techniques project a known pattern, such as a grid or a set of coded patterns, onto the scene. By analyzing the deformation of the projected pattern on the objects, DI can be extracted. This is commonly used in depth sensors like Microsoft Kinect (Rahim, Maqbool, and Rana, 2021).

LiDAR: LiDAR systems use lasers to emit pulses of light and measure the time it takes for the reflected light to return. By scanning the scene with laser beams and analyzing the return signals, 3D point clouds are generated, providing accurate distance measurements (Zehao, Cheng, and Guodong, 2022).

Monocular Depth Estimation: MDE refers to estimating depth from a single camera image. This is a challenging task as it requires leveraging visual cues such as perspective, texture, shading, and object size to infer depth. ML techniques, including CNNs and DL models, have been applied to learn depth from monocular images (Masoumian et al., 2021).

MDE models, despite being a challenging task, offers several advantages that make it valuable in various scenarios:

Cost and Accessibility: MDE relies on a single camera, which is a common component in many devices such as smartphones, surveillance systems, and autonomous vehicles. Compared to other methods like stereo vision or LiDAR, MDE requires minimal additional hardware, making it more cost-effective and accessible.

Flexibility: MDE provides flexibility in terms of camera placement and mobility. It enables DE from a wide range of viewpoints, allowing for versatile applications where fixed stereo camera setups or LiDAR scanning may be impractical.

Rapid Deployment: MDE algorithms can be quickly deployed on existing camera systems without the need for extensive calibration or specialized equipment setup. This enables faster integration into real-world applications and reduces deployment time and effort.

Wide Range of Applications: MDE has proven to be effective in a wide range of applications, including robotics, autonomous navigation, augmented reality, virtual reality, and scene understanding. By leveraging deep learning models and computer vision techniques, MDE algorithms can provide valuable depth information to enhance these applications.

Potential for Real-Time Processing: MDE algorithms can be designed to operate in real-time, allowing for dynamic environments and applications that require fast depth updates. This capability is particularly useful in tasks such as obstacle avoidance, object tracking, and real-time visual feedback.

Therefore, we have decided to use MDE for our absolute distance prediction due to its cost-effectiveness, versatility, rapid deployment, wide applicability, and real-time processing.

5.2.3 Depth Prediction

Depth and ego-motion estimation are critical tasks in computer vision, which involve understanding the 3D structure and motion of the scene from 2D images or videos. These tasks are fundamental for a wide range of applications such as robotics, autonomous driving, virtual and augmented reality, and more. To estimate depth and ego-motion, two main approaches are commonly used: supervised DL and unsupervised DL models.

5.2.3.1 Supervised Depth Estimation

Predicting a depth from a single image is an innately difficult task as the same image can project multiple conceivable depths. To prevent this, predicted depth needed to have some relationship with a color image. There are various approaches such as end-to-end (Laina et al., 2016), sense sampling of non-parametric (Karsch, Liu, and Kang, 2014), optical flow (Ilg et al., 2017), transfer learning (Alhashim and Wonka, 2018), and combining local predictions (Saxena, Sun, and Ng, 2008b), have been done. In supervised methods, the original depth maps of images will be used to train alongside color images. This will help the system to learn better and therefore the results of supervised methods usually have better performance than unsupervised methods.

However in reality it is difficult to construct depth maps in real-time and to do that, the use of a stereo camera or 3D LiDARs is necessary.

5.2.3.2 Unsupervised Depth Estimation

In contrast, unsupervised methods do not require annotated data and instead, learn from the structure and patterns in the input data. These methods often use techniques such as photometric loss, geometric consistency, or self-supervised learning to estimate depth or ego motion from single or multiple views. Although unsupervised methods are less accurate than supervised methods, they are more flexible and applicable to a wider range of scenarios, including unstructured or dynamic environments where annotated data is scarce or unreliable.

To avoid the aforementioned problems, unsupervised methods have been used for training the systems, only original images and pre-trained models such as DenseNet (Huang et al., 2017), ResNet (He et al., 2016), and ImageNet (Deng et al., 2009) are needed. Regarding unsupervised methods, various approaches for DE have been proposed, such as generative adversarial networks (Pilzer et al., 2018), temporal information (Babu et al., 2018), and separate pose networks (Zhou et al., 2017).

5.3 Methodology

In this work, we propose a framework for object detection and distance estimation using two parallel deep networks, as illustrated in Figure 5.1. The first network, YOLOv5, is used for object detection and classification, while the second network, DepthNet, is used for DE. The predicted depth map is obtained from DepthNet, which takes the input image as its input and outputs a corresponding depth map. This depth map contains depth values for every pixel in the input image.

The objects detected by YOLOv5 are localized and classified. The localization of each object is defined by a bounding box, which is a rectangular area surrounding

the object. These bounding boxes are then overlaid on the estimated depth map to determine the distance of each object. In particular, we determine the median estimated distance for each object by considering all the pixels within its bounding box. This methodology yields an approximation of the object's distance from the camera.

The proposed framework integrates the outputs of the two networks to provide accurate distance estimation for each object in the input image. The framework is designed to be computationally efficient and can be easily deployed on embedded systems and devices with limited memory and CPU resources.

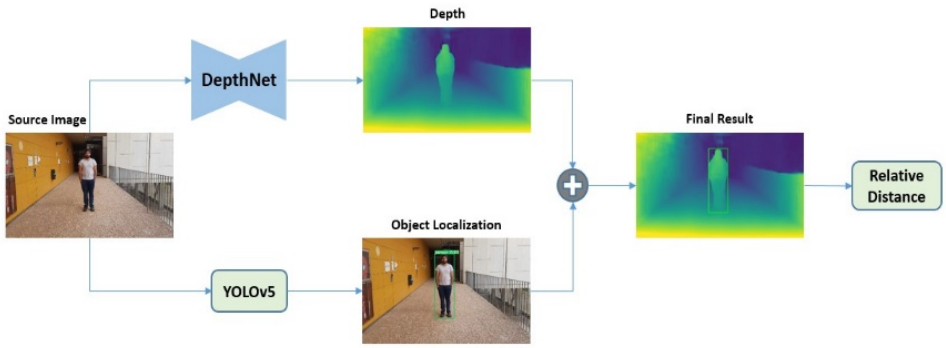


Figure 5.1: An illustration of the overall framework.

DepthNet is a deep neural network that plays a key role in our approach to depth and object detection. It consists of two networks, one for estimating depth images and the other for estimating the image pose. Both networks are based on autoencoder architecture, which is composed of two serial networks: encoders and decoders.

To extract features and represent input images, we utilized pre-trained weights from the ResNet network in the encoder of both depth and pose networks. Specifically, we used ResNet 50 as a backbone network for our depth prediction, and ResNet 18 for pose estimation. Before entering the first layer of the ResNet network, we used a block called Conv1, which consists of a convolutional layer, batch normalization layer, and

a max pooling operation. This was followed by four blocks of the ResNet network.

In the decoder, each layer consists of a deconvolutional layer and upsampling. Figure 5.2 provides an overview of the architecture. The last layer of the decoder produces the estimated depth map. Additionally, we incorporated our initial work based on GCN, which is one of the most powerful neural network architectures. GCN can properly find the similarity of pixels and make a graph connection between them. The proposed GCN model learns features by inspecting neighboring nodes. We used the GCN network in the decoder network to construct accurate depth images at multiple scales, as shown in Figure 5.2.

Our approach to depth and object detection leverages the strengths of each network to produce accurate and reliable results. The depth network estimates the distance of each pixel in the image from the camera, while the object detection network identifies and localizes objects within the image. The integration of both networks provides a more comprehensive understanding of the scene, enabling the calculation of the relevant distance of an object by the median estimated distance of all pixels inside the defined bounding box.

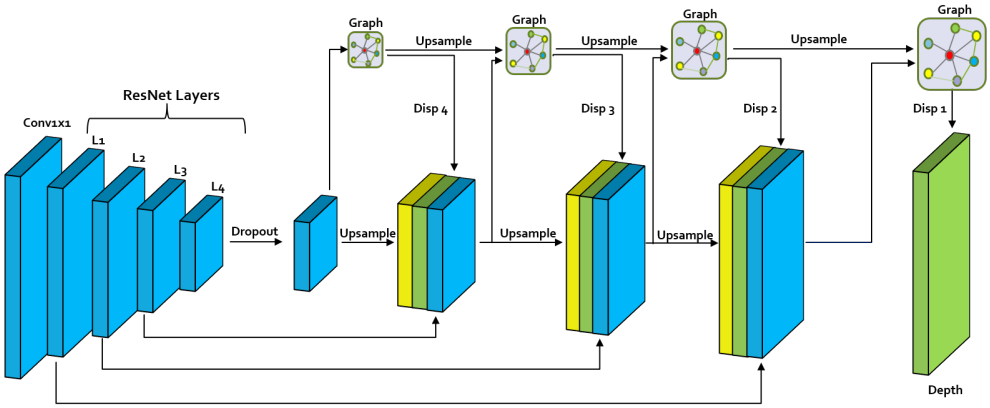


Figure 5.2: Overview of DepthNet network architecture.

Object detection is a critical task in computer vision that is vital for various applications such as self-driving cars, robotics, and surveillance systems. YOLOv5 is a

SOTA object detection model that can accurately and efficiently detect objects.

The YOLOv5 architecture has three main components: the Backbone, Neck, and Head. The Backbone creates features from input images, and YOLOv5 uses EfficientNet as its backbone to efficiently learn the complex features of input images. The Neck fuses feature from different scales, and YOLOv5 uses an improved PANet called Bi-FPN as its neck. Bi-FPN introduces learnable weights, allowing the network to learn the importance of different input features. It repeatedly applies top-down and bottom-up multi-scale feature fusion, making it easy and fast to fuse multi-scale features. The Head predicts the bounding boxes around objects and their classes.

To ensure the model's ability to learn complex features of input images, YOLOv5 leverages the SOTA network EfficientNet (Tan and Le, 2019) as its backbone. Additionally, YOLOv5 incorporates Bi-FPN, an improved PANet (Liu et al., 2018), as its neck, allowing for fast and easy multi-scale feature fusion. Bi-FPN introduces learnable weights, enabling the network to learn the importance of different input features. It repeatedly applies top-down and bottom-up multi-scale feature fusion to produce accurate results.

Furthermore, YOLOv5 integrates a compound scaling method that uniformly scales the resolution, depth, and width for all backbone, feature networks, and box/class prediction networks simultaneously. This ensures maximum accuracy and efficiency, even with limited computing resources.

To detect objects, YOLOv5 feeds the created features through a prediction system that predicts the bounding boxes and classes of objects. YOLOv5 can predict objects in real-time and has high accuracy compared to other object detection models. Figure 5.3 illustrates the YOLOv5 architecture used to detect objects in our work.

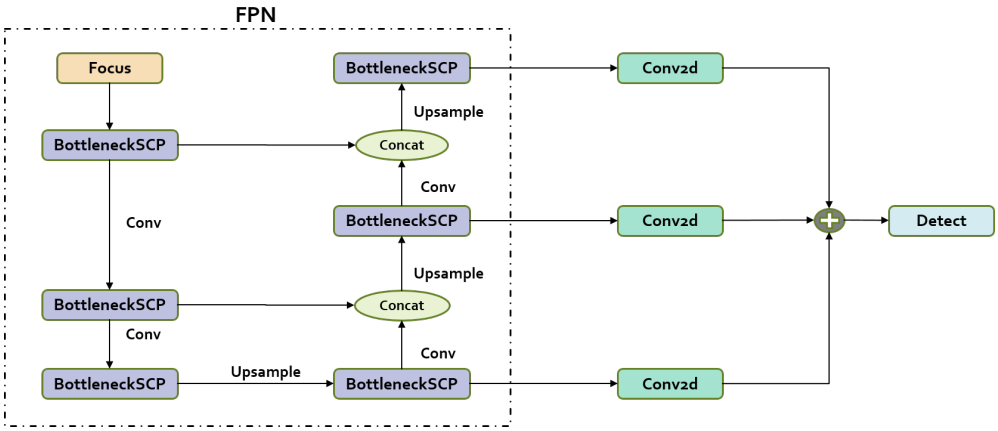


Figure 5.3: Overview of YOLOv5 network architecture.

5.3.1 Absolute Distance Prediction

After training our depth prediction model with the KITTI dataset (Geiger, Lenz, and Urtasun, 2012), we subsequently conducted testing on our proprietary dataset, as elaborated in section 5.4. The aim of the test was to utilize our DepthNet model to estimate the depth of each image within the testing dataset. Additionally, we utilized the YOLOv5 model to detect any objects present in the image and determine the location of their bounding boxes. By using the coordinates of these bounding boxes, we were able to accurately localize the corresponding predicted depth image.

It's important to highlight that the DepthNet model calculates disparity maps, which depict the relative motion between pixels in the input and target images, where the target image can be a derived image. Subsequently, we convert these disparity maps into depth maps, as detailed in (Uhrig et al., 2017), to gain a more comprehensive insight into the spatial relationships among objects and their distances from the camera.

In the DepthNet network, we set the minimum and maximum depth as 0 to 100 meters. This allowed us to effectively capture the DI of the objects in the scene, which is crucial for a variety of applications such as self-driving cars and robotics. Overall, the

combination of our DepthNet and YOLOv5 models enabled us to accurately estimate the depth of objects in real-world scenes, which has significant implications for the field of computer vision.

Following the depth image prediction and object detection process using YOLOv5, the subsequent task involves estimating the relative distance of the detected objects. To achieve this, we begin by calculating the median distance value for all the pixels contained within the bounding box of each object. This computed value serves as the relative distance of the object (REV). In various applications, including autonomous driving, understanding the distance of objects is paramount for ensuring safe decision-making. The REV value offers a dependable and precise estimation of the detected objects' distances, enhancing the quality of decision-making in these contexts.

To obtain the absolute distance (ABS) of objects in images, the real distance of those objects is required. Therefore, it is necessary to establish a relationship between the absolute distance and the relative distance obtained through the median value of the estimated distances of all pixels inside the bounding box of an object in a depth image, which we refer to as the relative distance of an object (REV).

Traditionally, the ABS estimation of an object has been dependent on various factors such as the object's shape and size, the image size, and the focal length of the sensor. However, this approach can be limiting in terms of applicability to different unknown objects. Hence, in this work, we aim to develop a calibration method that does not depend on this type of information and can work effectively for a wide range of unknown objects.

As a result, we can calculate the ABS of objects using the Taha and Jizat technique (Taha and Jizat, 2012), which involves a mathematical quadratic function. This method allows us to estimate the distance accurately without relying on the type and shape of objects, as well as the image size and focal length of the sensor. The quadratic function takes into account the relative distance of the object (REV) and several calibration parameters. By fine-tuning these parameters based on our private

dataset, we can ensure that our method works for different unknown objects. Overall, this approach provides a reliable and robust way to estimate the absolute distance of objects in images.

the ABS distance can be calculated based on a mathematical quadratic function:

$$Y = (c_0 + c_1 X + c_2 X^2) \times h \tag{5.1}$$

The coefficients c_0, c_1, and c_2 can be determined by using the least squares method, while h represents the height of the camera and X denotes the distance of the object from the starting point of the camera's field of view. To establish this relationship, a curve-fitting approach and least-squares optimization are employed to find the best estimate of the four unknown coefficients. In order to obtain the most accurate results, the solution is optimized to fit a set of data points, which in this case consists of 100 images with various objects and distances. This resulted in the quadratic function:

$$Y = 0.0036X^2 - 0.5373X + 21.714 \tag{5.2}$$

Figure 5.4 provides a visual representation of the relationship between ABS and REV distances.

5.4 Experiments

We implemented a Pytorch-based code for DE. The training was carried out for 20 epochs, with a batch size of 10, a learning rate of 0.0001, and the Adam optimizer. The training process lasted for 5 days, utilizing a single GTX 1080 TI GPU. To detect objects within the images, we employed YOLOv5, using the Pytorch library with 80 different classes. In terms of pre-trained checkpoints, we opted for YOLOv5s (light version) due to its lower computational cost.

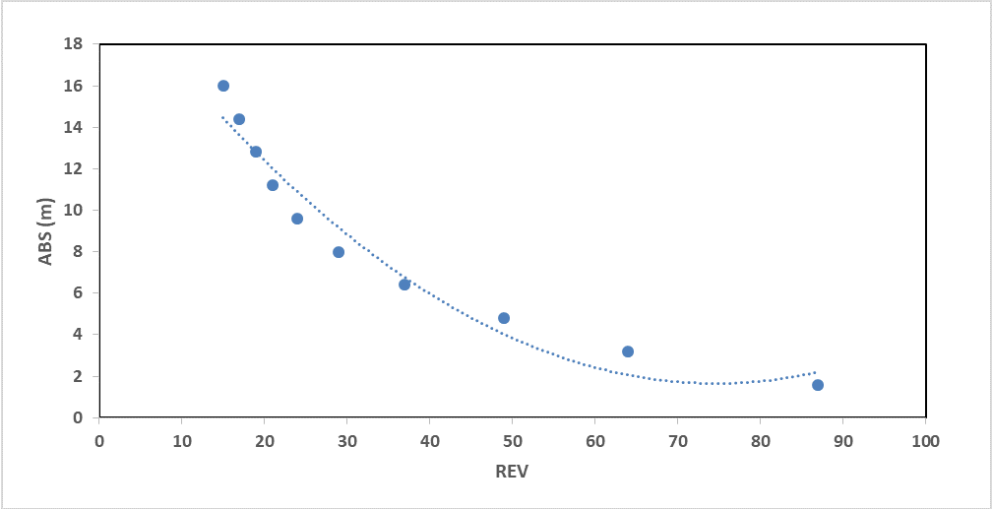


Figure 5.4: Perform the relation between ABS and REV.

5.4.1 Datasets

The KITTI dataset is widely recognized as one of the most prominent datasets in the field of computer vision for both depth and pose estimation tasks. It comprises 200 videos of real-world street scenes captured using RGB cameras, accompanied by the corresponding depth maps generated by a Velodyne laser scanner. For our study, we utilized the Eigen split (Eigen and Fergus, 2015) with 39810 images for training, 4424 for validation, and 697 for testing, and employed the (Masoumian et al., 2023) preprocessing method to eliminate static frames. The input images had a resolution of 1024 x 320.

On the other hand, the Coco dataset (Lin et al., 2014) is a comprehensive collection of images widely used for object detection, segmentation, and captioning. It includes 80 different object classes, comprising 2.5 million labeled instances in 328k images. We used the original split dataset, which contains 165482 images for training, 81208 for validation, and 81434 for testing. The image sizes were set to 640 x 480.

To calculate the absolute distance of objects, we prepared a private dataset consisting of 100 images with a resolution of 1350 x 777. A monocular RGB camera was mounted on a static stand, and the absolute distance of each object was manually measured from the camera. We collected the dataset by simulating various static obstacles at different distances from the camera test stand to ensure robustness. Having access to this information allowed us to obtain the absolute distance of each object, which was crucial for calculating the relative distance and evaluating our method's results.

5.4.2 Evaluation

During the testing phase, we evaluated the performance of our proposed method on all 100 images, which contained various objects such as persons, cars, and chairs, among others. Notably, our method is capable of detecting 80 classes of objects, as utilized in the COCO dataset.

We utilized two standard evaluation metrics to assess the performance of our framework: Accuracy and Root Mean Square Error (RMSE). Accuracy is a measure of how often our estimation is correct, based on a given threshold. We used the threshold accuracy measure from (Liu, Shen, and Lin, 2015), which is defined as the expected absolute distance error value of a given object in a scene that is lower than a threshold T (in this work, T is set to 0.2 m). The RMSE is another measure used to quantify the overall error in our predictions.

Figure 5.5 shows the qualitative results of our proposed framework, which involves DE using DepthNet and object detection using YOLOv5. Figure 5.5 includes examples from our own private dataset, which contains 100 images with different objects such as people, cars, and chairs, among others. For each example, Figure 5.5 displays the original image, the estimated depth image, the object localization results using YOLOv5, and the relevant depth estimated by DepthNet. These results demonstrate

the effectiveness of our approach in accurately estimating depth and localizing objects in complex scenes.

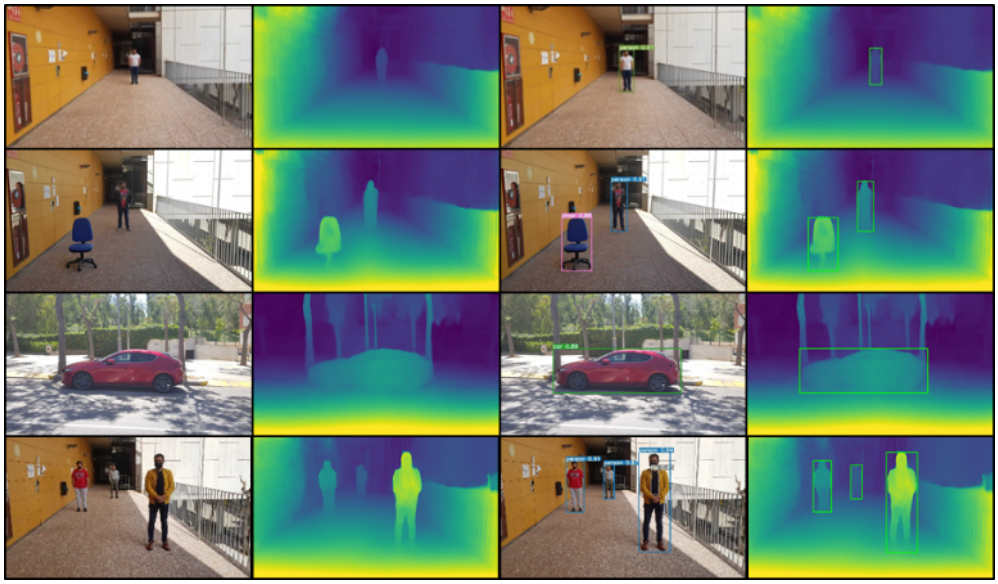


Figure 5.5: Visual process of the whole network.

Furthermore, Table 5.1 provides a measurement of the absolute distance versus the predicted distance. This table shows that our proposed method achieves satisfactory absolute distance estimation, despite the fact that our own private dataset did not contain object boxes from the captured scenes. This is an important finding, as it suggests that object detection using YOLOv5 is reliable even when it is used in its original form without fine-tuning with images from our private dataset.

Table 5.1: Estimated distance vs. absolute distance. Note the objects are counted in the tested images from left to right.

Figure 4.5	Object	Absolute Distance (m)	Predicted Distance (m)	Error (m)
Row 1	Person	11.20	10.91	0.29
Row 2	Chair	3.50	3.45	0.05
	Person	8.00	8.09	0.09
Row 3	Car	10.10	9.83	0.27
Row 4	Person 1	8.00	8.13	0.13
	Person 2	12.00	11.69	0.31
	Person 3	4.00	3.88	0.12

To assess the performance of our proposed framework, we used two standard evaluation measures: Accuracy and RMSE. The accuracy measure estimates errors under a given threshold, indicating how often our estimation is correct. In our work, we set the threshold T to 0.2 m, which is in line with the approach used in (Liu, Shen, and Lin, 2015). Our results show that our proposed framework achieves high accuracy, demonstrating its effectiveness in estimating depth and localizing objects in complex scenes.

Overall, the qualitative and quantitative results presented in Figure 5.5 and Table 5.1, respectively, demonstrate the effectiveness of our proposed framework for DE and object detection in complex scenes.

Table 5.1 presents the performance evaluation of the proposed framework in terms of absolute distance estimation. It can be observed that the farther away an object is, the greater the error in the predicted distance. Despite this limitation, the proposed framework achieved a high accuracy rate of 96% and an average RMSE of 0.203 (m), which demonstrates its effectiveness in estimating the absolute distance of objects. Importantly, it is noteworthy that our private dataset was not used in the training of DepthNet and YOLOv5, yet they still performed satisfactorily on the dataset.

In contrast to the DispNet method (Haseeb et al., 2018), which only detected objects on railways, our proposed framework was able to recognize different objects in various scenes captured by our private dataset. As expected, the YOLO network was able to easily detect big objects such as cars, even from a large distance. However, for small objects like chairs or people, YOLO was occasionally unable to detect them from a distance, which led to a slight degradation in the overall performance of the framework. Therefore, in future work, we plan to update YOLO to improve its performance with tiny objects.

5.5 Conclusion

In this chapter, we have presented a novel approach for estimating the absolute distances of objects in real-world scenes. Our proposed framework consists of two parallel networks: one that predicts the depth values of images using an unsupervised autoencoder network based on a 2D monocular camera, and the other that detects objects and extracts their localization boxes in the scene using YOLOv5. By calibrating our framework with real images, we were able to compute the absolute distance of an object from its relative distance.

Our results show that the proposed framework achieved an accuracy of 96% and an average RMSE of 0.203 (m), indicating that our approach is reliable in estimating absolute distances even when the private dataset was not part of the training dataset for DepthNet and YOLOv5. Our method outperforms the DispNet method, which only detected objects on railways, by recognizing different objects in the scene recorded by our private dataset.

In our future research endeavors, we intend to enhance the precision of our distance estimation approach through the creation of a trainable network capable of adapting the framework to diverse objects and shapes. Furthermore, we have plans to optimize the YOLOv5 network for the detection of smaller objects like chairs and individuals, which may occasionally elude detection at extended distances. Additionally, our aspirations involve the development of an intelligent assistant system designed to assist individuals with visual impairments, leveraging the foundations of our proposed framework.

Our proposed approach has the potential to significantly improve the quality of life of visually impaired people by providing them with a reliable and accurate distance estimation of objects in their environment. It can also be used in various applications, including autonomous vehicles, robotics, and surveillance systems, to estimate the absolute distances of objects accurately and reliably.

In the next chapter, the conclusion of the thesis, the highlights, limitations, and future works of the research will be discussed. This final chapter will provide an overview of the key findings and contributions of the thesis, including the novel approaches proposed for MDE using DL techniques and the significant advancements achieved. Additionally, the limitations encountered throughout the research process will be addressed, and potential avenues for future research will be outlined. This chapter will serve as a comprehensive summary, bringing together the main outcomes of the thesis and setting the stage for further developments in DL-based MDE.

Chapter 6

Conclusion

This final chapter presents this dissertation's most important contributions and main conclusions, emphasizing their significance. Likewise, the chapter also includes approaches for future work.

6.1 Thesis Highlights

As technology advances, accurate depth measurements become increasingly essential in many fields. One area that benefits greatly from this is robotics engineering, where robots must navigate their environments safely and effectively. Autonomous vehicles also need accurate depth measurements to navigate roads and avoid collisions. Self-supervised MDE is a cutting-edge technology that aims to estimate object depth in a scene using just one image without expensive stereoscopic or 3D cameras. The advancements in DL techniques have made this possible, with models using complex algorithms to extract features from the image and estimate object distances.

The review chapter provides a comprehensive review of the current advancements in DL-based MDE. It covers key aspects such as input data shapes, training methodologies, and evaluation metrics. The chapter discusses limitations in accuracy, computational requirements, real-time inference, transferability, input image shapes, domain adaptation, and generalization. Potential avenues for future research are highlighted to address these limitations.

To overcome these limitations, this thesis presents two novel approaches for MDE using DL techniques. The first approach proposes a self-supervised MDE model that utilizes GCN to estimate depth maps from monocular videos. GCNs improve the model's accuracy by handling non-Euclidean data, and a combination of loss functions is used to preserve object discontinuities and manage bad depth prediction. The proposed model achieved a high prediction accuracy of 89% on the KITTI dataset and reduced the number of trainable parameters by 40% compared to SOTA solutions. This demonstrates the proposed self-supervised MDE approach's effectiveness based on GCN through quantitative and qualitative comparisons with other SOTA methods.

The second approach proposes a DL framework that utilizes two separate networks for DE and object detection by using a single image. The proposed approach employs YOLOv5 to detect and localize objects within the scene and a deep autoencoder

network to compute the estimated depth image. The presented framework achieved an impressive accuracy rate of 96% with an RMSE of 0.203 for the correct absolute distance, demonstrating the effectiveness of the proposed multitask learning approach for MDE and object detection.

Overall, this study demonstrates the significant advantages of DL-based approaches for MDE, including the ability to handle non-Euclidean data, irregular image regions within a topological structure, and preserve object discontinuities. The proposed approaches in this thesis provide promising results and can contribute to developing more accurate and efficient computer vision systems for various applications. Nonetheless, our work is a significant step forward in the field of MDE using DL, and it has the potential to impact multiple fields that require accurate depth measurements significantly.

6.2 Limitations

While our proposed method for MDE based on GCN has shown promising results, several limitations should be considered.

Firstly, the model's accuracy heavily relies on the quality and diversity of the training data. Without a diverse and representative dataset, the model may struggle to generalize to new scenes and conditions, leading to poor performance.

Secondly, the computational requirements for training and inference can be significant, particularly for models with larger architectures and datasets. This can limit the practicality of the technology for real-time applications or resource-limited devices.

Thirdly, the proposed model is currently limited to estimating the depth of static scenes and cannot handle dynamic scenes or moving objects. This is a significant limitation for applications such as robotics or autonomous vehicles, which often involve dynamic environments.

Fourthly, while our proposed model has achieved good accuracy on the KITTI dataset, it may not generalize well to other datasets or environments, particularly those with different characteristics or properties.

Lastly, the proposed model is limited to estimating depth from monocular images and cannot take advantage of the additional information provided by stereo or multiple cameras. This can limit the accuracy and range of applications for the technology.

In conclusion, while our proposed method has shown promising results, these limitations should be considered when applying the technology in practical scenarios. Addressing these limitations through further research and development will be crucial for improving the accuracy, applicability, and efficiency of MDE based on GCN.

6.3 Future Research Lines

Based on the limitations identified in this study, several avenues for future research can be pursued to improve the accuracy and applicability of MDE using GCN. In addition to the options mentioned earlier, there are several more directions to explore, particularly in the development of lightweight models for real-time applications. Firstly, exploring the potential of using other types of GCN could improve the accuracy of the DE model. Further research could also focus on combining different types of GCN to leverage their respective strengths and improve performance. Secondly, while UL has proven effective in MDE, incorporating additional supervision from other sources, such as stereo images or LiDAR data, could further improve the models' accuracy. Thirdly, adapting the models to handle different environmental conditions and scenes, such as indoor environments or night-time conditions, could significantly expand the applicability of the technology. Fourthly, developing methods to improve the efficiency and speed of the models to allow for real-time inference on resource-limited devices, such as embedded systems or mobile devices, would be beneficial for practical applications.

This could involve exploring techniques like model compression, architecture optimization, hardware acceleration, and knowledge transfer to create lightweight GCN models. Fifthly, investigating multi-modal fusion techniques for integrating information from multiple sources, such as RGB images, depth maps, or sensor data, can enhance the accuracy of MDE models while considering computational and memory efficiency. Sixthly, enabling online learning or incremental updates in GCN-based models can ensure their relevance and accuracy over time, which is crucial for real-time applications. Seventhly, collecting large-scale datasets designed explicitly for lightweight MDE models and carefully annotating them can facilitate the development and evaluation of such models. In conclusion, the promising results of our proposed method for MDE based on GCN highlight the potential of this technology in various fields, such as robotics and autonomous vehicles. By addressing the mentioned avenues for future research, including the development of lightweight GCN models for real-time applications, significant advancements can be made in terms of accuracy, applicability, and efficiency. This, in turn, can have far-reaching implications in various industries, enabling the deployment of real-time MDE on resource-limited devices and fostering progress in domains such as robotics, augmented reality, and mobile applications.

UNIVERSITAT ROVIRA I VIRGILI
ENHANCING DISTANCE PREDICTION THROUGH MONOCULAR DEPTH ESTIMATION BASED ON GRAPH CONVOLUTIONAL NETWORKS
Armin Masoumian

Bibliography

Abdulwahab, Saddam, Hatem A Rashwan, Miguel Angel Garcia, Mohammed Jabreel, Sylvie Chambon, and Domenec Puig (2020). "Adversarial Learning for Depth and Viewpoint Estimation from a Single Image". In: *IEEE Transactions on Circuits and Systems for Video Technology*.

Abdulwahab, Saddam, Hatem A Rashwan, Miguel Angel Garcia, Armin Masoumian, and Domenec Puig (2022). "Monocular depth map estimation based on a multi-scale deep architecture and curvilinear saliency feature boosting". In: *Neural Computing and Applications* 34.19, pp. 16423–16440.

Abdulwahab, Saddam, Hatem A Rashwan, Armin Masoumian, Najwa Sharaf, and Domenec Puig (2021). "Promising Depth Map Prediction Method from a Single Image Based on Conditional Generative Adversarial Network". In: *Artificial Intelligence Research and Development: Proceedings of the 23rd International Conference of the Catalan Association for Artificial Intelligence*, p. 392.

Aboali, Maged, Nurulfajar Abd Manap, Zulkalnain Mohd Yusof, et al. (2018). "A Multistage Hybrid Median Filter Design of Stereo Matching Algorithms on Image Processing". In: *Journal of Telecommunication, Electronic and Computer Engineering (JTEC)* 10.4, pp. 133–141.

Agarwal, Nakul, Cheng-Wei Chiang, and Abhishek Sharma (2019). "A study on computer vision techniques for self-driving cars". In: *Frontier Computing: Theory, Technologies and Applications (FC 2018) 7*. Springer, pp. 629–634.

Alagoz, B Baykant (2008). "Obtaining depth maps from color images by region based stereo matching algorithms". In: *arXiv preprint arXiv:0812.1340*.

Aleotti, Filippo, Fabio Tosi, Matteo Poggi, and Stefano Mattoccia (2018). "Generative adversarial networks for unsupervised monocular depth prediction". In: *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, pp. 0–0.

Alhashim, Ibraheem and Peter Wonka (2018). "High quality monocular depth estimation via transfer learning". In: *arXiv preprint arXiv:1812.11941*.

Allison, Robert S, Barbara J Gillam, and Elia Vecellio (2009). "Binocular depth discrimination and estimation beyond interaction space". In: *Journal of Vision* 9.1, pp. 10–10.

Almalioglu, Yasin, Muhamad Risqi U Saputra, Pedro PB de Gusmao, Andrew Markham, and Niki Trigoni (2019). "Ganvo: Unsupervised deep monocular visual odometry and depth estimation with generative adversarial networks". In: *2019 International conference on robotics and automation (ICRA)*. IEEE, pp. 5474–5480.

Andhare, Pratiksha and Sayali Rawat (2016). "Pick and place industrial robot controller with computer vision". In: *2016 International Conference on Computing Communication Control and automation (ICCUBEA)*. IEEE, pp. 1–4.

Babu, V Madhu, Kaushik Das, Anima Majumdar, and Swagat Kumar (2018). "Undemon: Unsupervised deep network for depth and ego-motion estimation". In: *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, pp. 1082–1088.

Badue, Claudine, Rânik Guidolini, Raphael Vivacqua Carneiro, Pedro Azevedo, Vinicius B Cardoso, Avelino Forechi, Luan Jesus, Rodrigo Berriel, Thiago M Paixao, Filipe Mutz, et al. (2021). "Self-driving cars: A survey". In: *Expert Systems with Applications* 165, p. 113816.

Bank, Dor, Noam Koenigstein, and Raja Giryes (2020). "Autoencoders". In: *arXiv preprint arXiv:2003.05991*.

Bazrafkan, Shabab, Hossein Javidnia, Joseph Lemley, and Peter Corcoran (2018). "Semiparallel deep neural network hybrid architecture: first application on depth from monocular camera". In: *Journal of Electronic Imaging* 27.4, p. 043041.

Bhat, Shariq Farooq, Ibraheem Alhashim, and Peter Wonka (2021). "Adabins: Depth estimation using adaptive bins". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4009–4018.

Bian, Jiawang, Zhichao Li, Naiyan Wang, Huangying Zhan, Chunhua Shen, Ming-Ming Cheng, and Ian Reid (2019). "Unsupervised scale-consistent depth and ego-motion learning from monocular video". In: *Advances in neural information processing systems* 32, pp. 35–45.

Boykov, Yu, Olga Veksler, and Ramin Zabih (1998). "A variable window approach to early vision". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 20.12, pp. 1283–1294.

Bronstein, Michael M, Joan Bruna, Yann LeCun, Arthur Szlam, and Pierre Vandergheynst (2017). "Geometric deep learning: going beyond euclidean data". In: *IEEE Signal Processing Magazine* 34.4, pp. 18–42.

Bugby, SL, JE Lees, WK McKnight, and NS Dawood (2021). "Stereoscopic portable hybrid gamma imaging for source depth estimation". In: *Physics in Medicine & Biology* 66.4, p. 045031.

Casser, Vincent, Soeren Pirk, Reza Mahjourian, and Anelia Angelova (2019). "Depth prediction without the sensors: Leveraging structure for unsupervised learning from monocular videos". In: *Proceedings of the AAAI conference on artificial intelligence*. Vol. 33. 01, pp. 8001–8008.

Chen, Liang-Chieh, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille (2017). "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs". In: *IEEE transactions on pattern analysis and machine intelligence* 40.4, pp. 834–848.

Chen, Weifeng, Zhao Fu, Dawei Yang, and Jia Deng (2016). "Single-image depth perception in the wild". In: *Advances in neural information processing systems* 29, pp. 730–738.

Chen, Xiaotian, Xuejin Chen, and Zheng-Jun Zha (2019). "Structure-aware residual pyramid network for monocular depth estimation". In: *arXiv preprint arXiv: 1907.06023*.

Chen, Xingyu, Ruonan Zhang, Ji Jiang, Yan Wang, Ge Li, and Thomas H Li (2023). "Self-supervised monocular depth estimation: Solving the edge-fattening problem". In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 5776–5786.

Chen, Yuhua, Cordelia Schmid, and Cristian Sminchisescu (2019). "Self-supervised learning with geometric constraints in monocular video: Connecting flow, depth, and camera". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7063–7072.

Chen, Yuru, Haitao Zhao, Zhengwei Hu, and Jingchao Peng (2021). "Attention-based context aggregation network for monocular depth estimation". In: *International Journal of Machine Learning and Cybernetics* 12.6, pp. 1583–1596.

Cho, Jaehoon, Dongbo Min, Youngjung Kim, and Kwanghoon Sohn (2019). "A large RGB-D dataset for semi-supervised monocular depth estimation". In: *arXiv preprint arXiv:1904.10230*.

Clark, Ronald, Sen Wang, Hongkai Wen, Andrew Markham, and Niki Trigoni (2017). "Vinet: Visual-inertial odometry as a sequence-to-sequence learning problem". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 31. 1.

Cociaş, Tiberiu T, Sorin M Grigorescu, and Florin Moldoveanu (2012). "Multiple-superquadrics based object surface estimation for grasping in service robotics". In: *2012 13th International Conference on Optimization of Electrical and Electronic Equipment (OPTIM)*. IEEE, pp. 1471–1477.

Trouvé, Pauline, Frédéric Champagnat, Guy Le Besnerais, Jacques Sabater, Thierry Avignon, and Jérôme Idier (2013). "Passive depth estimation using chromatic aberration and a depth from defocus approach". In: *Applied optics* 52.29, pp. 7152–7164.

Cong, Shuang and Yang Zhou (2023). "A review of convolutional neural network architectures and their optimizations". In: *Artificial Intelligence Review* 56.3, pp. 1905–1969.

Cordts, Marius, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele (2016). "The cityscapes dataset for semantic urban scene understanding". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3213–3223.

Couprie, Camille, Clément Farabet, Laurent Najman, and Yann LeCun (2013). "Indoor semantic segmentation using depth information". In: *arXiv preprint arXiv: 1301.3572*.

CS Kumar, Arun, Suchendra M Bhandarkar, and Mukta Prasad (2018). "Depthnet: A recurrent neural network architecture for monocular depth prediction". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 283–291.

Cunningham, Pádraig, Matthieu Cord, and Sarah Jane Delany (2008). "Supervised learning". In: *Machine learning techniques for multimedia*. Springer, pp. 21–49.

Daily, Mike, Swarup Medasani, Reinhold Behringer, and Mohan Trivedi (2017). "Self-driving cars". In: *Computer* 50.12, pp. 18–23.

Daneshmand, Morteza, Ahmed Helmi, Egils Avots, Fatemeh Noroozi, Fatih Alisinanoglu, Hasan Sait Arslan, Jelena Gorbova, Rain Eric Haamer, Cagri Ozcinar, and Gholamreza Anbarjafari (2018). "3d scanning: A comprehensive survey". In: *arXiv preprint arXiv:1801.08863*.

Deng, Jia, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei (2009). "Imagenet: A large-scale hierarchical image database". In: *2009 IEEE conference on computer vision and pattern recognition*. Ieee, pp. 248–255.

Dijk, Tom van and Guido de Croon (2019). "How do neural networks see depth in single images?" In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2183–2191.

Dong, Xingshuai, Matthew A Garratt, Sreenatha G Anavatti, and Hussein A Abbass (2021). "Towards real-time monocular depth estimation for robotics: A survey". In: *arXiv preprint arXiv:2111.08600*.

Eigen, David and Rob Fergus (2015). "Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture". In: *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2650–2658.

Eigen, David, Christian Puhrsch, and Rob Fergus (2014). "Depth map prediction from a single image using a multi-scale deep network". In: *Advances in neural information processing systems* 27.

Estrach, Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann LeCun (2014). "Spectral networks and deep locally connected networks on graphs". In: *2nd international conference on learning representations, ICLR*. Vol. 2014.

Facil, Jose M, Benjamin Ummenhofer, Huizhong Zhou, Luis Montesano, Thomas Brox, and Javier Civera (2019). "CAM-Convs: Camera-aware multi-scale convolutions for single-view depth". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11826–11835.

Fei, Xiaohan, Alex Wong, and Stefano Soatto (2019). "Geo-supervised visual depth prediction". In: *IEEE Robotics and Automation Letters* 4.2, pp. 1661–1668.

Fu, Huan, Mingming Gong, Chaohui Wang, Kayhan Batmanghelich, and Dacheng Tao (2018). "Deep ordinal regression network for monocular depth estimation". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2002–2011.

Fu, Junwei, Jun Liang, and Ziyang Wang (2019). "Monocular depth estimation based on multi-scale graph convolution networks". In: *IEEE Access* 8, pp. 997–1009.

Fusiello, Andrea, Emanuele Trucco, and Alessandro Verri (1997). "Rectification with unconstrained stereo geometry." In: *BMVC*, pp. 400–409.

Gan, Wanshui, Pak Kin Wong, Guokuan Yu, Rongchen Zhao, and Chi Man Vong (2021). "Light-weight network for real-time adaptive stereo depth estimation". In: *Neurocomputing* 441, pp. 118–127.

Garg, Ravi, Vijay Kumar Bg, Gustavo Carneiro, and Ian Reid (2016). "Unsupervised cnn for single view depth estimation: Geometry to the rescue". In: *European conference on computer vision*. Springer, pp. 740–756.

Geiger, Andreas, Philip Lenz, and Raquel Urtasun (2012). "Are we ready for autonomous driving? the kitti vision benchmark suite". In: *2012 IEEE conference on computer vision and pattern recognition*. IEEE, pp. 3354–3361.

Geng, Mingyang, Suning Shang, Bo Ding, Huaimin Wang, and Pengfei Zhang (2020). "Unsupervised learning-based depth estimation-aided visual slam approach". In: *Circuits, Systems, and Signal Processing* 39.2, pp. 543–570.

Glennerster, Andrew, Brian J Rogers, and Mark F Bradshaw (1996). "Stereoscopic depth constancy depends on the subject's task". In: *Vision research* 36.21, pp. 3441–3456.

Godard, Clément, Oisin Mac Aodha, and Gabriel J Brostow (2017). "Unsupervised monocular depth estimation with left-right consistency". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 270–279.

Godard, Clément, Oisin Mac Aodha, Michael Firman, and Gabriel J Brostow (2019). "Digging into self-supervised monocular depth estimation". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3828–3838.

Goldman, Matan, Tal Hassner, and Shai Avidan (2019). "Learn stereo, infer mono: Siamese networks for self-supervised, monocular, depth estimation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 0–0.

Gordon, Ariel, Hanhan Li, Rico Jonschkowski, and Anelia Angelova (2019). "Depth from videos in the wild: Unsupervised monocular depth learning from unknown cameras". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 8977–8986.

Guizilini, Vitor, Rares Ambrus, Sudeep Pillai, Allan Raventos, and Adrien Gaidon (2020). "3d packing for self-supervised monocular depth estimation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2485–2494.

Guo, Xiaoyang, Hongsheng Li, Shuai Yi, Jimmy Ren, and Xiaogang Wang (2018). "Learning monocular depth by distilling cross-domain stereo networks". In: *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 484–500.

Haque, Albert, Boya Peng, Zelun Luo, Alexandre Alahi, Serena Yeung, and Li Fei-Fei (2016). "Towards viewpoint invariant 3d human pose estimation". In: *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*. Springer, pp. 160–177.

Haseeb, Muhammad Abdul, Jianyu Guan, Danijela Ristic-Durrant, and Axel Gräser (2018). "DisNet: a novel method for distance estimation from monocular camera". In: *10th Planning, Perception and Navigation for Intelligent Vehicles (PPNIV18), IROS*.

He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun (2016). "Deep residual learning for image recognition". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778.

He, Lei, Jiwen Lu, Guanghui Wang, Shiyu Song, and Jie Zhou (2021). "SOSD-Net: Joint semantic object segmentation and depth estimation from monocular images". In: *Neurocomputing* 440, pp. 251–263.

He, Li, Chuangbin Chen, Tao Zhang, Haife Zhu, and Shaohua Wan (2018). "Wearable depth camera: Monocular depth estimation via sparse optimization under weak supervision". In: *IEEE Access* 6, pp. 41337–41345.

Heikkila, Janne and Olli Silvén (1997). "A four-step camera calibration procedure with implicit image correction". In: *Proceedings of IEEE computer society conference on computer vision and pattern recognition*. IEEE, pp. 1106–1112.

Hoiem, Derek, Alexei A Efros, and Martial Hebert (2005). "Automatic photo pop-up". In: *ACM SIGGRAPH 2005 Papers*, pp. 577–584.

Hosni, Asmaa, Christoph Rhemann, Michael Bleyer, Carsten Rother, and Margrit Gelautz (2012). "Fast cost-volume filtering for visual correspondence and beyond". In: *IEEE transactions on pattern analysis and machine intelligence* 35.2, pp. 504–511.

Howard, Andrew G, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam (2017). "Mobilenets: Efficient convolutional neural networks for mobile vision applications". In: *arXiv preprint arXiv:1704.04861*.

Hu, Junjie, Chenyou Fan, Hualie Jiang, Xiyue Guo, Yuan Gao, Xiangyong Lu, and Tin Lun Lam (2021). "Boosting Light-Weight Depth Estimation Via Knowledge Distillation". In: *arXiv preprint arXiv:2105.06143*.

Hu, Junjie, Mete Ozay, Yan Zhang, and Takayuki Okatani (2019). "Revisiting single image depth estimation: Toward higher resolution maps with accurate object boundaries". In: *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, pp. 1043–1051.

Hu, Zhe, Li Xu, and Ming-Hsuan Yang (2014). "Joint depth estimation and camera shake removal from single blurry image". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2893–2900.

Huang, Gao, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger (2017). "Densely connected convolutional networks". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4700–4708.

Hyun, Jongkil, Younghyeon Kim, Junghwan Kim, and Byungin Moon (2020). "Hardware-friendly architecture for a pseudo 2D weighted median filter based on sparse-window approach". In: *Multimedia Tools and Applications*, pp. 1–16.

Ilg, Eddy, Nikolaus Mayer, Tonmoy Saikia, Margret Keuper, Alexey Dosovitskiy, and Thomas Brox (2017). "Flownet 2.0: Evolution of optical flow estimation with deep networks". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2462–2470.

Javidnia, Hossein and Peter Corcoran (2016). "A depth map post-processing approach based on adaptive random walk with restart". In: *IEEE Access* 4, pp. 5509–5519.

Javidnia, Hossein and Peter Corcoran (2017). "Accurate depth map estimation from small motions". In: *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pp. 2453–2461.

Jia, Shaocheng, Xin Pei, Wei Yao, and SC Wong (2021). "Self-supervised Depth Estimation Leveraging Global Perception and Geometric Smoothness Using On-board Videos". In: *arXiv preprint arXiv:2106.03505*.

Jiménez, David, Daniel Pizarro, Manuel Mazo, and Sira Palazuelos (2014). "Modeling and correction of multipath interference in time of flight cameras". In: *Image and Vision Computing* 32.1, pp. 1–13.

Jocher, Glenn, Ayush Chaurasia, Alex Stoken, Jirka Borovec, Yonghye Kwon, Kalen Michael, Jiacong Fang, Zeng Yifu, Colin Wong, Diego Montes, et al. (2022). "Ultralytics/yolov5: v7. 0-YOLOv5 SotA realtime instance segmentation". In: *Zenodo*.

Jung, Dongki, Jaehoon Choi, Yonghan Lee, Deokhwa Kim, Changick Kim, Dinesh Manocha, and Donghwan Lee (2021). "DnD: Dense Depth Estimation in Crowded Dynamic Indoor Scenes". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 12797–12807.

Jung, Hyungjoo, Youngjung Kim, Dongbo Min, Changjae Oh, and Kwanghoon Sohn (2017). "Depth prediction from a single image with conditional adversarial networks". In: *2017 IEEE International Conference on Image Processing (ICIP)*. IEEE, pp. 1717–1721.

Kalia, M., N. Navab, and T. Salcudean (2019). "A Real-Time Interactive Augmented Reality Depth Estimation Technique for Surgical Robotics". In: *2019 International Conference on Robotics and Automation (ICRA)*, pp. 8291–8297.

Karsch, Kevin, Ce Liu, and Sing Bing Kang (2014). "Depth transfer: Depth extraction from video using non-parametric sampling". In: *IEEE transactions on pattern analysis and machine intelligence* 36.11, pp. 2144–2158.

Kat, Rotal, Roy Jevnisek, and Shai Avidan (2018). "Matching pixels using co-occurrence statistics". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1751–1759.

Kaur, Harpreet, Nam Pham, Sergey Fomel, Zhicheng Geng, Luke Decker, Ben Gremillion, Michael Jervis, Ray Abma, and Shuang Gao (2023). "A deep learning framework for seismic facies classification". In: *Interpretation* 11.1, T107–T116.

Kendall, Alex, Hayk Martirosyan, Saumitro Dasgupta, Peter Henry, Ryan Kennedy, Abraham Bachrach, and Adam Bry (2017). "End-to-end learning of geometry and context for deep stereo regression". In: *Proceedings of the IEEE International Conference on Computer Vision*, pp. 66–75.

Khan, Faisal, Saqib Salahuddin, and Hossein Javidnia (2020). "Deep learning-based monocular depth estimation methods—A state-of-the-art review". In: *Sensors* 20.8, p. 2272.

Khoshelham, Kourosh and Sander Oude Elberink (2012). "Accuracy and resolution of kinect depth data for indoor mapping applications". In: *sensors* 12.2, pp. 1437–1454.

Kim, Hyun Myung, Min Seok Kim, Gil Ju Lee, Hyuk Jae Jang, and Young Min Song (2020). "Miniaturized 3D depth sensing-based smartphone light field camera". In: *Sensors* 20.7, 2129.

Kim, Ue-Hwan, Se-Ho Kim, and Jong-Hwan Kim (2020). "Simvodis: Simultaneous visual odometry, object detection, and instance segmentation". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44.1, pp. 428–441.

Kingma, Diederik P and Jimmy Ba (2014). "Adam: A method for stochastic optimization". In: *arXiv preprint arXiv:1412.6980*.

Kipf, Thomas N and Max Welling (2016). "Semi-supervised classification with graph convolutional networks". In: *arXiv preprint arXiv:1609.02907*.

Kuznietsov, Yevhen, Marc Proesmans, and Luc Van Gool (2021). "Comoda: Continuous monocular depth adaptation using past experiences". In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 2907–2917.

Kuznietsov, Yevhen, Jorg Stuckler, and Bastian Leibe (2017). "Semi-supervised deep learning for monocular depth map prediction". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6647–6655.

Laga, Hamid, Laurent Valentin Jospin, Farid Boussaid, and Mohammed Bennamoun (2020). "A survey on deep learning techniques for stereo-based depth estimation". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Laina, Iro, Christian Rupprecht, Vasileios Belagiannis, Federico Tombari, and Nassir Navab (2016). "Deeper depth prediction with fully convolutional residual networks". In: *2016 Fourth international conference on 3D vision (3DV)*. IEEE, pp. 239–248.

Lam, Edmund Y (2015). "Computational photography with plenoptic camera and light field capture: tutorial". In: *JOSA A* 32.11, pp. 2021–2032.

Lee, Jin Han, Myung-Kyu Han, Dong Wook Ko, and Il Hong Suh (2019). "From big to small: Multi-scale local planar guidance for monocular depth estimation". In: *arXiv preprint arXiv:1907.10326*.

Lee, Sihaeng, Janghyeon Lee, Doyeon Kim, and Junmo Kim (2020). "Deep architecture with cross guidance between single image and sparse lidar data for depth completion". In: *IEEE Access* 8, pp. 79801–79810.

Lei, Zeyu, Yan Wang, Zijian Li, and Junyao Yang (2021). "Attention based multilayer feature fusion convolutional neural network for unsupervised monocular depth estimation". In: *Neurocomputing* 423, pp. 343–352.

Li, Bo, Chunhua Shen, Yuchao Dai, Anton Van Den Hengel, and Mingyi He (2015). "Depth and surface normal estimation from monocular images using regression on deep features and hierarchical crfs". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1119–1127.

Li, Rui, Danna Xue, Shaolin Su, Xiantuo He, Qing Mao, Yu Zhu, Jinqiu Sun, and Yanning Zhang (2023). "Learning depth via leveraging semantics: Self-supervised monocular depth estimation with both implicit and explicit semantic guidance". In: *Pattern Recognition*, p. 109297.

Li, Ruihao, Sen Wang, Zhiqiang Long, and Dongbing Gu (2018). "Undeepvo: Monocular visual odometry through unsupervised deep learning". In: *2018 IEEE international conference on robotics and automation (ICRA)*. IEEE, pp. 7286–7291.

Li, Shunkai, Fei Xue, Xin Wang, Zike Yan, and Hongbin Zha (2019). "Sequential adversarial learning for self-supervised deep visual odometry". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2851–2860.

Liang, Jiali, Yufan Deng, and Dan Zeng (2020). "A deep neural network combined CNN and GCN for remote sensing scene classification". In: *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 13, pp. 4325–4338.

Liebel, Lukas and Marco Körner (2019). "Multidepth: Single-image depth estimation via multi-task regression and classification". In: *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*. IEEE, pp. 1440–1447.

Lin, Tsung-Yi, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick (2014). "Microsoft coco: Common objects in context". In: *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*. Springer, pp. 740–755.

Liu, Fayao, Chunhua Shen, and Guosheng Lin (2015). "Deep convolutional neural fields for depth estimation from a single image". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5162–5170.

Liu, Fayao, Chunhua Shen, Guosheng Lin, and Ian Reid (2015). "Learning depth from single monocular images using deep convolutional neural fields". In: *IEEE transactions on pattern analysis and machine intelligence* 38.10, pp. 2024–2039.

Liu, Lina, Xibin Song, Mengmeng Wang, Yong Liu, and Liangjun Zhang (2021). "Self-supervised Monocular Depth Estimation for All Day Images using Domain Separation". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 12737–12746.

Liu, Miaomiao, Mathieu Salzmann, and Xuming He (2014). "Discrete-continuous depth estimation from a single image". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 716–723.

Liu, Shu, Lu Qi, Haifang Qin, Jianping Shi, and Jiaya Jia (2018). "Path aggregation network for instance segmentation". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 8759–8768.

Liu, Xingtong, Ayushi Sinha, Masaru Ishii, Gregory D Hager, Austin Reiter, Russell H Taylor, and Mathias Unberath (2019). "Dense depth estimation in monocular endoscopy with self-supervised learning methods". In: *IEEE transactions on medical imaging* 39.5, pp. 1438–1447.

Loop, Charles and Zhengyou Zhang (1999). "Computing rectifying homographies for stereo vision". In: *Proceedings. 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No PR00149)*. Vol. 1. IEEE, pp. 125–131.

Lu, Yawen and Guoyu Lu (2019). "Deep unsupervised learning for simultaneous visual odometry and depth estimation". In: *2019 IEEE International Conference on Image Processing (ICIP)*. IEEE, pp. 2571–2575.

Lukac, Rastislav (2017). *Computational photography: methods and applications*. CRC press.

Luo, Chenxu, Zhenheng Yang, Peng Wang, Yang Wang, Wei Xu, Ram Nevatia, and Alan Yuille (2019). "Every pixel counts++: Joint learning of geometry and motion with 3d holistic understanding". In: *IEEE transactions on pattern analysis and machine intelligence* 42.10, pp. 2624–2641.

Luo, Wenjie, Alexander G Schwing, and Raquel Urtasun (2016). "Efficient deep learning for stereo matching". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5695–5703.

Luo, Yue, Jimmy Ren, Mude Lin, Jiahao Pang, Wenxiu Sun, Hongsheng Li, and Liang Lin (2018). "Single view stereo matching". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 155–163.

Ma, Fangchang, Guilherme Venturelli Cavalheiro, and Sertac Karaman (2019). "Self-supervised sparse-to-dense: Self-supervised depth completion from lidar and monocular camera". In: *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, pp. 3288–3295.

Mahjourian, Reza, Martin Wicke, and Anelia Angelova (2018). "Unsupervised learning of depth and ego-motion from monocular video using 3d geometric constraints". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5667–5675.

Makarov, Ilya, Maria Bakhanova, Sergey Nikolenko, and Olga Gerasimova (2022). "Self-supervised recurrent depth estimation with attention mechanisms". In: *PeerJ Computer Science* 8, e865.

Mancini, Michele, Gabriele Costante, Paolo Valigi, Thomas A Ciarfuglia, Jeffrey Delmerico, and Davide Scaramuzza (2017). "Toward domain independence for learning-based monocular depth estimation". In: *IEEE Robotics and Automation Letters* 2.3, pp. 1778–1785.

Mandelbaum, Robert, G Kamberova, and Max Mintz (1998). "Stereo depth estimation: a confidence interval approach". In: *Sixth International Conference on Computer Vision (IEEE Cat. No. 98CH36271)*. IEEE, pp. 503–509.

Masoumian, Armin, Pezhman Kazemi, Mohammad Chehreghani Montazer, Hatem A Rashwan, and Domenec Puig Valls (2020a). "Designing and analyzing the PID and fuzzy control system for an inverted pendulum". In: *2020 6th International Conference on Mechatronics and Robotics Engineering (ICMRE)*. IEEE, pp. 199–203.

Masoumian, Armin, Pezhman Kazemi, Mohammad Chehreghani Montazer, Hatem A Rashwan, and Domenec Puig Valls (2020b). "Using The Feedback of Dynamic Active-Pixel Vision Sensor (Davis) to Prevent Slip in Real Time". In: *2020 6th International Conference on Mechatronics and Robotics Engineering (ICMRE)*. IEEE, pp. 63–67.

Masoumian, Armin, DG Marei, Saddam Abdulwahab, Julian Cristiano, Domenec Puig, and Hatem A Rashwan (2021). "Absolute distance prediction based on deep learning object detection and monocular depth estimation models". In: *Artificial*

Intelligence Research and Development: Proceedings of the 23rd International Conference of the Catalan Association for Artificial Intelligence. Vol. 339. IOS Press, p. 325.

Masoumian, Armin, Hatem A Rashwan, Saddam Abdulwahab, Julián Cristiano, M Salman Asif, and Domenec Puig (2023). "GCNDepth: Self-supervised monocular depth estimation based on graph convolutional network". In: *Neurocomputing* 517, pp. 81–92.

Masoumian, Armin, Hatem A Rashwan, Julián Cristiano, M Salman Asif, and Domenec Puig (2022). "Monocular depth estimation using deep learning: A review". In: *Sensors* 22.14, p. 5353.

Maximov, Maxim, Kevin Galim, and Laura Leal-Taixé (2020). "Focus on defocus: bridging the synthetic to real domain gap for depth estimation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1071–1080.

Mayer, Nikolaus, Eddy Ilg, Philip Hausser, Philipp Fischer, Daniel Cremers, Alexey Dosovitskiy, and Thomas Brox (2016). "A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4040–4048.

Meng, Yue, Yongxi Lu, Aman Raj, Samuel Sunarjo, Rui Guo, Tara Javidi, Gaurav Bansal, and Dinesh Bharadia (2019). "Signet: Semantic instance aided unsupervised 3d geometry perception". In: *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition*, pp. 9810–9820.

Meng, Zelin, Xiangbo Kong, Lin Meng, and Hiroyuki Tomiyama (2021). "Stereo Vision-Based Depth Estimation". In: *Advances in Artificial Intelligence and Data Engineering*. Springer, pp. 1209–1216.

Ming, Yue, Xuyang Meng, Chunxiao Fan, and Hui Yu (2021). "Deep learning for monocular depth estimation: A review". In: *Neurocomputing* 438, pp. 14–33.

Mousavian, Arsalan, Hamed Pirsiavash, and Jana Košecká (2016). "Joint semantic segmentation and depth estimation with deep convolutional networks". In: *2016 Fourth International Conference on 3D Vision (3DV)*. IEEE, pp. 611–619.

Nekrasov, Vladimir, Thanuja Dharmasiri, Andrew Spek, Tom Drummond, Chunhua Shen, and Ian Reid (2019). "Real-time joint semantic segmentation and depth estimation using asymmetric annotations". In: *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, pp. 7101–7107.

Nistér, David, Oleg Naroditsky, and James Bergen (2004). "Visual odometry". In: *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.* Vol. 1. Ieee, pp. I–I.

Niu, Zhaoyang, Guoqiang Zhong, and Hui Yu (2021). "A review on the attention mechanism of deep learning". In: *Neurocomputing* 452, pp. 48–62.

Nomani, Ashkan, Yasaman Ansari, Mohammad Hossein Nasirpour, Armin Masoumian, Ehsan Sadeghi Pour, and Amin Valizadeh (2022). "PSOWNNs-CNN: A Computational Radiology for Breast Cancer Diagnosis Improvement Based on Image Processing Using Machine Learning Methods". In: *Computational Intelligence and Neuroscience* 2022.

Noraky, James and Vivienne Sze (2019). "Low power depth estimation of rigid objects for time-of-flight imaging". In: *IEEE Transactions on Circuits and Systems for Video Technology* 30.6, pp. 1524–1534.

Olanrewaju, Hammed G and Wasiu O Popoola (2017). "Effect of synchronization error on optical spatial modulation". In: *IEEE Transactions on Communications* 65.12, pp. 5362–5374.

Palmisano, Stephen, Barbara Gillam, Donovan G Govan, Robert S Allison, and Julie M Harris (2010). "Stereoscopic perception of real depths at large distances". In: *Journal of vision* 10.6, pp. 19–19.

Paszke, Adam, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer (2017). "Automatic differentiation in pytorch". In.

Patil, Vaishakh, Wouter Van Gansbeke, Dengxin Dai, and Luc Van Gool (2020). "Don't forget the past: Recurrent depth estimation from monocular video". In: *IEEE Robotics and Automation Letters* 5.4, pp. 6813–6820.

Petrou, Maria MP and Costas Petrou (2010). *Image processing: the fundamentals*. John Wiley & Sons.

Pillai, Sudeep, Rareş Ambruş, and Adrien Gaidon (2019). "Superdepth: Self-supervised, super-resolved monocular depth estimation". In: *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, pp. 9250–9256.

Pilzer, Andrea, Stephane Lathuiliere, Nicu Sebe, and Elisa Ricci (2019). "Refine and distill: Exploiting cycle-inconsistency and knowledge distillation for unsupervised monocular depth estimation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9768–9777.

Pilzer, Andrea, Dan Xu, Mihai Puscas, Elisa Ricci, and Nicu Sebe (2018). "Unsupervised adversarial depth estimation using cycled generative networks". In: *2018 International Conference on 3D Vision (3DV)*. IEEE, pp. 587–595.

Poggi, Matteo, Filippo Aleotti, Fabio Tosi, and Stefano Mattoccia (2018). "Towards real-time unsupervised monocular depth estimation on cpu". In: *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, pp. 5848–5854.

Poggi, Matteo, Filippo Aleotti, Fabio Tosi, and Stefano Mattoccia (2020). "On the uncertainty of self-supervised monocular depth estimation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3227–3237.

Praveen, Satyarth (2020). "Efficient depth estimation using sparse stereo-vision with other perception techniques". In: *Coding Theory*, p. 111.

Qi, Xiaojuan, Renjie Liao, Zhengzhe Liu, Raquel Urtasun, and Jiaya Jia (2018). "Geonet: Geometric neural network for joint depth and surface normal estimation". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 283–291.

Rahim, Adina, Ayesha Maqbool, and Tauseef Rana (2021). "Monitoring social distancing under various low light conditions with deep learning and a single motionless time of flight camera". In: *Plos one* 16.2, e0247440.

Ramamonjisoa, Michael and Vincent Lepetit (2019). "Sharpnet: Fast and accurate recovery of occluding contours in monocular depth estimation". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pp. 0–0.

Ramirez, Pierluigi Zama, Matteo Poggi, Fabio Tosi, Stefano Mattoccia, and Luigi Di Stefano (2018). "Geometry meets semantics for semi-supervised monocular depth estimation". In: *Asian Conference on Computer Vision*. Springer, pp. 298–313.

Ranftl, René, Alexey Bochkovskiy, and Vladlen Koltun (2021). "Vision transformers for dense prediction". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 12179–12188.

Ranftl, René, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun (2019). "Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer". In: *arXiv preprint arXiv:1907.01341*.

Ranjan, Anurag, Varun Jampani, Lukas Balles, Kihwan Kim, Deqing Sun, Jonas Wulff, and Michael J Black (2019). "Competitive collaboration: Joint unsupervised learning of depth, camera motion, optical flow and motion segmentation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12240–12249.

Rashwan, Hatem A, Sylvie Chambon, Pierre Gurdjos, Géraldine Morin, and Vincent Charvillat (2018). "Using Curvilinear Features in Focus for Registering a Single Image to a 3D Object". In: *arXiv preprint arXiv:1802.09384*.

Rashwan, Hatem A, Agusti Solanas, Domènec Puig, and Antoni Martínez-Ballesté (2016). "Understanding trust in privacy-aware video surveillance systems". In: *International Journal of Information Security* 15.3, 225–234.

Redmon, Joseph, Santosh Divvala, Ross Girshick, and Ali Farhadi (2016). "You only look once: Unified, real-time object detection". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 779–788.

Ren, Shaoqing, Kaiming He, Ross Girshick, and Jian Sun (2015). "Faster r-cnn: Towards real-time object detection with region proposal networks". In: *Advances in neural information processing systems* 28.

Rodrigues, Rômulo T, Pedro Miraldo, Dimos V Dimarogonas, and A Pedro Aguiar (2020). "Active depth estimation: Stability analysis and its applications". In: *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, pp. 2002 –2008.

Saikia, Tonmoy, Yassine Marrakchi, Arber Zela, Frank Hutter, and Thomas Brox (2019). "Autodispnet: Improving disparity estimation with automl". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1812–1823.

Salehinejad, Hojjat, Sharan Sankar, Joseph Barfett, Errol Colak, and Shahrokh Valaee (2017). "Recent advances in recurrent neural networks". In: *arXiv preprint arXiv: 1801.01078*.

Santos Rosa, Nícolas dos, Vitor Guizilini, and Valdir Grassi (2019). "Sparse-to-continuous: Enhancing monocular depth estimation using occupancy maps". In: *2019 19th International Conference on Advanced Robotics (ICAR)*. IEEE, pp. 793–800.

Sanz, Pablo Revuelta, Belén Ruiz Mezcua, and José M Sánchez Pena (2012). *Depth estimation-an introduction*. IntechOpen.

Saudabayev, Artur and Huseyin Atakan Varol (2015). "Sensors for robotic hands: A survey of state of the art". In: *IEEE Access* 3, pp. 1765–1782.

Saxena, Ashutosh, Min Sun, and Andrew Y Ng (2008a). "Make3D: Depth Perception from a Single Still Image." In: *AAAI*. Vol. 3, pp. 1571–1576.

Saxena, Ashutosh, Min Sun, and Andrew Y Ng (2008b). "Make3d: Learning 3d scene structure from a single still image". In: *IEEE transactions on pattern analysis and machine intelligence* 31.5, pp. 824–840.

Saxena, Saurabh, Abhishek Kar, Mohammad Norouzi, and David J Fleet (2023). "Monocular depth estimation using diffusion models". In: *arXiv preprint arXiv: 2302.14816*.

Scharstein, Daniel, Heiko Hirschmüller, York Kitajima, Greg Krathwohl, Nera Nešić, Xi Wang, and Porter Westling (2014). "High-resolution stereo datasets with subpixel-accurate ground truth". In: *German conference on pattern recognition*. Springer, pp. 31–42.

Scharstein, Daniel and Richard Szeliski (2002). "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms". In: *International journal of computer vision* 47.1, pp. 7–42.

Schonberger, Johannes L and Jan-Michael Frahm (2016). "Structure-from-motion revisited". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4104–4113.

Sheng, Fei, Feng Xue, Yicong Chang, Wenteng Liang, and Anlong Ming (2022). "Monocular Depth Distribution Alignment with Low Computation". In: *arXiv preprint arXiv:2203.04538*.

Shu, Chang, Kun Yu, Zhixiang Duan, and Kuiyuan Yang (2020). "Feature-metric loss for self-supervised learning of depth and egomotion". In: *European Conference on Computer Vision*. Springer, pp. 572–588.

Silberman, Nathan, Derek Hoiem, Pushmeet Kohli, and Rob Fergus (2012). "Indoor segmentation and support inference from rgbd images". In: *European conference on computer vision*. Springer, pp. 746–760.

Silva Vieira, Gabriel da, Fabrizzio Alphonsus AMN Soares, Gustavo T Laureano, Rafael T Parreira, Júlio C Ferreira, and Rogério Salvini (2018). "Disparity Map Adjustment: a Post-Processing Technique". In: *2018 IEEE Symposium on Computers and Communications (ISCC)*. IEEE, pp. 00580–00585.

Singh, Harmanjit and Richa Sharma (2012). "Role of adjacency matrix & adjacency list in graph theory". In: *International Journal of Computers & Technology* 3.1, pp. 179–183.

Sun, Jian, Nan-Ning Zheng, and Heung-Yeung Shum (2003). "Stereo matching using belief propagation". In: *IEEE Transactions on pattern analysis and machine intelligence* 25.7, pp. 787–800.

Sun, Xing, Zhimin Xu, Nan Meng, Edmund Y Lam, and Hayden K-H So (2016). "Data-driven light field depth estimation using deep convolutional neural networks". In: *2016 International Joint Conference on Neural Networks (IJCNN)*. IEEE, pp. 367–374.

Suo, JinLi, XiangYang Ji, and QiongHai Dai (2012). "An overview of computational photography". In: *Science China Information Sciences* 55.6, pp. 1229–1248.

Süvari, Cemal Barışkan (2021). "Semi-supervised iterative teacher-student learning for monocular depth estimation". MA thesis. Middle East Technical University.

Szikora, Péter and Nikolett Madarász (2017). "Self-driving cars—The human side". In: *2017 IEEE 14th international scientific conference on informatics*. IEEE, pp. 383–387.

Taha, Zahari and Jessnor Arif Mat Jizat (2012). "A comparison of two approaches for collision avoidance of an automated guided vehicle using monocular vision". In: *Applied Mechanics and Materials*. Vol. 145. Trans Tech Publ, pp. 547–551.

Tan, Mingxing and Quoc Le (2019). "Efficientnet: Rethinking model scaling for convolutional neural networks". In: *International conference on machine learning*. PMLR, pp. 6105–6114.

Teed, Zachary and Jia Deng (2018). "Deepv2d: Video to depth with differentiable structure from motion". In: *arXiv preprint arXiv:1812.04605*.

Tosi, Fabio, Filippo Aleotti, Matteo Poggi, and Stefano Mattoccia (2019). "Learning monocular depth estimation infusing traditional stereo knowledge". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9799–9809.

Uhrig, Jonas, Nick Schneider, Lukas Schneider, Uwe Franke, Thomas Brox, and Andreas Geiger (2017). "Sparsity invariant cnns". In: *2017 international conference on 3D Vision (3DV)*. IEEE, pp. 11–20.

Ulrich, Luca, Enrico Vezzetti, Sandro Moos, and Federica Marcolin (2020). "Analysis of RGB-D camera technologies for supporting different facial usage scenarios". In: *Multimedia Tools and Applications* 79.39, 29375–29398.

Ummenhofer, Benjamin, Huizhong Zhou, Jonas Uhrig, Nikolaus Mayer, Eddy Ilg, Alexey Dosovitskiy, and Thomas Brox (2017). "Demon: Depth and motion network for learning monocular stereo". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5038–5047.

Vasiljevic, Igor, Nick Kolkin, Shanyi Zhang, Ruotian Luo, Haochen Wang, Falcon Z Dai, Andrea F Daniele, Mohammadreza Mostajabi, Steven Basart, Matthew R Walter, et al. (2019). "Diode: A dense indoor and outdoor depth dataset". In: *arXiv preprint arXiv:1908.00463*.

Vyas, Pulkit, Chirag Saxena, Anwesh Badapanda, and Anurag Goswami (2022). "Outdoor Monocular Depth Estimation: A Research Review". In: *arXiv preprint arXiv: 2205.01399*.

Wang, Chaoyang, José Miguel Buenaposada, Rui Zhu, and Simon Lucey (2018a). "Learning depth from monocular videos using direct methods". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2022–2030.

Wang, Haixia, Yehao Sun, QM Jonathan Wu, Xiao Lu, Xiuling Wang, and Zhiguo Zhang (2021). "Self-supervised monocular depth estimation with direct methods". In: *Neurocomputing* 421, pp. 340–348.

Wang, Jianfeng, Cha Zhang, Wenwu Zhu, Zhengyou Zhang, Zixiang Xiong, and Philip A Chou (2012). "3D scene reconstruction by multiple structured-light based commodity depth cameras". In: *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, pp. 5429–5432.

Wang, Panqu, Pengfei Chen, Ye Yuan, Ding Liu, Zehua Huang, Xiaodi Hou, and Garrison Cottrell (2018b). "Understanding convolution for semantic segmentation". In: *2018 IEEE winter conference on applications of computer vision (WACV)*. IEEE, pp. 1451–1460.

Wang, Rui, Stephen M Pizer, and Jan-Michael Frahm (2019). "Recurrent neural network for (un-) supervised learning of monocular video visual odometry and depth". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5555–5564.

Wang, Yang, Peng Wang, Zhenheng Yang, Chenxu Luo, Yi Yang, and Wei Xu (2019). "Unos: Unified unsupervised optical-flow and stereo-depth estimation by watching videos". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8071–8081.

Watson, Jamie, Michael Firman, Gabriel J Brostow, and Daniyar Turmukhambetov (2019). "Self-supervised monocular depth hints". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2162–2171.

Wofk, Diana, Fangchang Ma, Tien-Ju Yang, Sertac Karaman, and Vivienne Sze (2019). "Fastdepth: Fast monocular depth estimation on embedded systems". In: *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, pp. 6101–6108.

Wu, Zhenyao, Xinyi Wu, Xiaoping Zhang, Song Wang, and Lili Ju (2019). "Spatial correspondence with generative adversarial network: Learning depth from monocular

videos". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7494–7504.

Xie, Junyuan, Ross Girshick, and Ali Farhadi (2016). "Deep3d: Fully automatic 2d-to-3d video conversion with deep convolutional neural networks". In: *European conference on computer vision*. Springer, pp. 842–857.

Xue, Fei, Xin Wang, Shunkai Li, Qiuyuan Wang, Junqiu Wang, and Hongbin Zha (2019). "Beyond tracking: Selecting memory and refining poses for deep visual odometry". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8575–8583.

Yan, Zhi, Li Sun, Tom Duckctr, and Nicola Bellotto (2018). "Multisensor online transfer learning for 3d lidar-based human detection with a mobile robot". In: *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, pp. 7635–7640.

Yang, Guorun, Xiao Song, Chaoqin Huang, Zhidong Deng, Jianping Shi, and Bolei Zhou (2019). "Drivingstereo: A large-scale dataset for stereo matching in autonomous driving scenarios". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 899–908.

Yang, Zhenheng, Peng Wang, Yang Wang, Wei Xu, and Ram Nevatia (2018a). "Lego: Learning edge with geometry all at once by watching videos". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 225–234.

Yang, Zhenheng, Peng Wang, Wei Xu, Liang Zhao, and Ramakant Nevatia (2018b). "Unsupervised learning of geometry from videos with edge-aware depth-normal consistency". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 32. 1.

Yin, Wei, Yifan Liu, Chunhua Shen, and Youliang Yan (2019). "Enforcing geometric constraints of virtual normal for depth prediction". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 5684–5693.

Yin, Zhichao and Jianping Shi (2018). "Geonet: Unsupervised learning of dense depth, optical flow and camera pose". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1983–1992.

Yoo, Jin Hyeok, Yecheol Kim, Jisong Kim, and Jun Won Choi (2020). "3d-cvf: Generating joint camera and lidar features using cross-view spatial feature fusion for 3d object detection". In: *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVII 16*. Springer, pp. 720–736.

Yu, Zehao and Shenghua Gao (2020). "Fast-mvsnet: Sparse-to-dense multi-view stereo with learned propagation and gauss-newton refinement". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1949–1958.

Yuan, Yule, Wenbin Zou, Yong Zhao, Xinan Wang, Xuefeng Hu, and Nikos Komodakis (2016). "A robust and efficient approach to license plate detection". In: *IEEE Transactions on Image Processing* 26.3, pp. 1102–1114.

Zaarane, Abdelmoghit, Ibtissam Slimani, Wahban Al Okaishi, Issam Atouf, and Abdellatif Hamdoun (2020). "Distance measurement system for autonomous vehicles using stereo camera". In: *Array* 5, p. 100016.

Zehao, Yu, Lu Cheng, and Liu Guodong (2022). "FMCW LiDAR with an FM nonlinear kernel function for dynamic-distance measurement". In: *Optics Express* 30.11, pp. 19582–19596.

Zhan, Huangying, Ravi Garg, Chamara Saroj Weerasekera, Kejie Li, Harsh Agarwal, and Ian Reid (2018). "Unsupervised learning of monocular depth estimation and visual odometry with deep feature reconstruction". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 340–349.

Zhang, Li, Xiangtai Li, Anurag Arnab, Kuiyuan Yang, Yunhai Tong, and Philip HS Torr (2019). "Dual graph convolutional network for semantic segmentation". In: *arXiv preprint arXiv:1909.06121*.

Zhang, Mingliang, Xinchen Ye, Xin Fan, and Wei Zhong (2020). "Unsupervised depth estimation from monocular videos with hybrid geometric-refined loss and contextual attention". In: *Neurocomputing* 379, pp. 250–261.

Zhang, Ning, Francesco Nex, George Vosselman, and Norman Kerle (2023). "Lite-mono: A lightweight cnn and transformer architecture for self-supervised monocular depth estimation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18537–18546.

Zhang, Shuai, Chong Wang, and Shing-Chow Chan (2015). "A new high resolution depth map estimation system using stereo vision and kinect depth sensing". In: *Journal of Signal Processing Systems* 79, pp. 19–31.

Zhang, Zhengyou (2000). "A flexible new technique for camera calibration". In: *IEEE Transactions on pattern analysis and machine intelligence* 22.11, pp. 1330–1334.

Zhao, Chaoqiang, Qiyu Sun, Chongzhen Zhang, Yang Tang, and Feng Qian (2020). "Monocular depth estimation based on deep learning: An overview". In: *Science China Technological Sciences*, pp. 1–16.

Zhao, Chaoqiang, Yang Tang, Qiyu Sun, and Athanasios V Vasilakos (2021). "Deep direct visual odometry". In: *IEEE Transactions on Intelligent Transportation Systems*.

Zhao, Hang, Orazio Gallo, Iuri Frosio, and Jan Kautz (2016). "Loss functions for image restoration with neural networks". In: *IEEE Transactions on computational imaging* 3.1, pp. 47–57.

Zhao, Shanshan, Huan Fu, Mingming Gong, and Dacheng Tao (2019). "Geometry-aware symmetric domain adaptation for monocular depth estimation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9788–9798.

Zhong, Fuqiang and Chenggen Quan (2021). "Stereo-rectification and homography-transform-based stereo matching methods for stereo digital image correlation". In: *Measurement* 173, p. 108635.

Zhou, Hang, David Greenwood, and Sarah Taylor (2021). "Self-Supervised Monocular Depth Estimation with Internal Feature Fusion". In: *arXiv preprint arXiv: 2110.09482*.

Zhou, Junsheng, Yuwang Wang, Kaihuai Qin, and Wenjun Zeng (2019). "Unsupervised high-resolution depth learning from videos with dual networks". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 6872–6881.

Zhou, Kun, Xiangxi Meng, and Bo Cheng (2020). "Review of stereo matching algorithms based on deep learning". In: *Computational intelligence and neuroscience* 2020.

Zhou, Tinghui, Matthew Brown, Noah Snavely, and David G Lowe (2017). "Unsupervised learning of depth and ego-motion from video". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1851–1858.

Zhou, Xinchi, Dongzhan Zhou, Di Hu, Hang Zhou, and Wanli Ouyang (2023). "Exploiting Visual Context Semantics for Sound Source Localization". In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 5199–5208.

Zou, Yuliang, Zelun Luo, and Jia-Bin Huang (2018). "Df-net: Unsupervised joint learning of depth and flow using cross-task consistency". In: *Proceedings of the European conference on computer vision (ECCV)*, pp. 36–53.

UNIVERSITAT ROVIRA I VIRGILI
ENHANCING DISTANCE PREDICTION THROUGH MONOCULAR DEPTH ESTIMATION BASED ON GRAPH CONVOLUTIONAL NETWORKS
Armin Masoumian

UNIVERSITAT ROVIRA i VIRGILI