Proceedings of the Seventh International Conference on
Railway Technology: Research, Development and Maintenance
Edited by: J. Pombo
Civil-Comp Conferences, Volume 15, Paper 17.5
Civil-Comp Press, Edinburgh, United Kingdom, 2026
ISSN: 2753-3239, doi: 10.4203/ccc.15.17.5

A Novel Approach to Railway Wheel Defect Detection Using Custom Deep Learning Models for Monocular Depth Estimation

M. Z. Shaikh[1,2,3], B. Abro[2,3], E. N. Baro[4], S. Jatoi[2,3], E. B. Blazquez-Parra[5], B. S. Chowdhry[2,3] and A. Kolmykova[6]

[1]Mechanical Engineering and Energy Efficiency, School of Industrial Engineering, University of Malaga, Spain
[2]National Center for Robotics, Automation and Artificial Intelligence, Mehran University of Engineering and Technology (MUET), Jamshoro, Pakistan
[3]NCRA-CMS Lab, Mehran University of Engineering and Technology (MUET), Jamshoro, Pakistan
[4]Departamento de Ingenier´ıa de Comunicaciones, Campus de Teatinos, Universidad de Malaga, Spain
[5]Department of Graphical Engineering, Design and Projects, Universidad de Malaga, Spain
[5]Zhytomyr Polytechnic State University, Ukraine

Abstract

The monocular depth estimation in the field of computer vision represents significant challenges that are crucial in many applications such as autonomous driving and augmented reality. Recent development in deep learning addresses the challenges of depth prediction requiring both fine-grained details and structural integrity. This research introduces novel techniques that integrate the state-of-the-art convolutional neural networks with other advanced architectures including custom U-Net, ResNet50 and Vision Transformer to produce depth maps based on images. The evaluation of experiments showed that custom UNet outperformed other models with high accuracy and generalization than ResNet50 and ViT. It achieved significantly small RMSE of 0.1267 and loss of 0.0160 compared to other models that overfit the results with higher RMSE and loss scores.

Keywords: railway, depth estimation, computer vision, convolutional neural networks, U-Net, ResNet50, transformer

1. Introduction

Depth estimation (DE) is a traditional problem in the field of computer vision (CV) as it changes two-dimensional images to three-dimensional space information. Autonomous driving, robotics, object recognition and augmented reality heavily relies on efficient DE. Depth maps (DM) provide fundamental spatial relationship information about objects in the scene that allow machines to interpret the world for the purpose of navigation in real world. Traditional approaches of DE imply the usage of hardware devices like stereo cameras and LiDAR [1]. These techniques of DE involve high financial cost and complex calibration that require specialized hardware setup expert [2].

Monocular depth estimation (MDE) predicts depth based on a single RGB image, which makes it reliable and cost-effective. The conventional methods of mono-depth estimation relied heavily on stereo matching image-based feature detection and hand coded algorithms [3]. The MDE methods address many challenges of DE from one view as the identical 2D image projection may be produced by different 3D scenes. The generalization capability of manual feature extraction techniques is low in terms of application in different scenes and environments due to limitations in handling occlusions and variation in light [4], [5].

The advancement of deep learning (DL), particularly in Convolutional Neural Networks (CNNs), has led to significant changes in end-to-end learning approaches [6], [7]. DL techniques help in automatic detection of hierarchical features that eliminate the manual selection of features [8]. DL used in MDE has been a great resource for precise detection under various environments. Although there is progress in the DL-based MDE, problems like vanishing and exploding gradients and being unable to efficiently learn multi-scale contextual information [9], [10].

This research presents a novel approach for MDE that uses custom DL techniques in identifying defects in railway wheels. The contributions of the research article are given below:

1. A railway wheel test rig was indigenously designed and developed for real railway wheel inspection. It is used to capture the DM of moving railway wheels to collect real world data for model training.
2. The data collection was performed using an Onion Tau depth camera of the rotating wheels. The camera stores DM and amplitude and real data that is the foundation of DL model training. The data is then transformed from raw data to convenient PNG images of amplitude depth and real values for model training.
3. The experimentation of various DL architectures including custom U-Net, ResNet in encoder-decoder structure and ViT in estimating depth of our dataset. Experiments conducted with the custom U-Net model showed better performance as compared to other architecture.

This structure of the paper is as follows: Section 2 reviews the recent research and advancement in MDE. Section 3 consists of a detailed explanation of our proposed

approach elaborating the architecture and workflow of our custom U-Net model. Section 4 presents the experimentation and comparison of our results with the state-of-the-art techniques. Section 5 shows the results with recommendation and future directions.

2. Methods

2.1 Creation of Wheelset Test Rig

The Automated Wheelset test rig has been indigenously developed and emphasized the achievement of precision in testing railway wheelsets in real-time as shown in Figure 1. The initial stage in the development process was to create a structural frame that ensures the assembly of the key components of the system. High-strength steel was required in the base that support the assembly to ensure that they could be tested with heavy loads. The base frame was welded to give it structural integrity and minimize the vibration interference with the data accuracy. The integration of the wheel-actuator coupler and AC servo actuators managed the rotational movement of the wheelsets during the testing. The precise positioning of the actuators ensured the existence of smooth and controlled motion of wheelsets and least friction as possible to increase durability.

Forms of v-shaped support bars were thoroughly mounted to accommodate various types of wheelsets and ease the process of loading and offloading to be achievable. The mounting brackets were placed in strategic positions that made the wheelset fixed during testing. To achieve stability of the rig in dynamic tests, iron blocks were used to maintain the weight stability of the rig. The addition of the motor assembly provided the ability to have controlled rotation of the wheelsets. The motors had speed controls and feedback devices to enable the motor systems to achieve the precise rotational speed that replicates real world conditions. The major step of sensor positioning entailed the incorporation of the sensors into the rig to monitor the wheel profile, wear patterns, and internal defects. The data collected was crucial in creating the rig to detect small defects in the condition of wheels.

All components were assembled and calibration procedures were performed on them. The sensors were tested and adjusted through different tests, that provided quality data on the tests. Fine-tuning of the actuators made the system have efficient and smooth movement. The testing phase involved the use of several wheelsets in the rig to ensure sensor precision in the wear and tear measurements as well as testing the actuators to ensure the accuracy of operation. The control system was adjusted as per the real-world environments to detect wear and tear during testing as the control system was meant to handle various testing conditions.

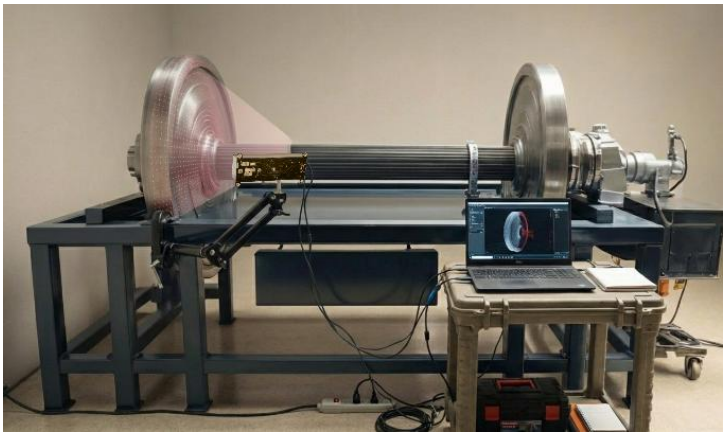

Figure 1: Wheelset Test Rig

2.2 Data Collection with Onion Tau Camera

The data collection was implemented using the Onion Tau LiDAR Camera to capture real-time depth information in 3D of the wheelset test rig. The Onion Tau Camera is a LiDAR based camera that integrates both technologies to form a small sized device that produces depth data and greyscale images of the surrounding world as shown in Figure 2. It is a regular webcam but with the additional feature of forming 3D depth maps using flight LiDAR technology. This device has a resolution of 160x60 pixels and depth ranging between 0.1 meters to 4.5 meters hence it is ideal in collecting detailed depth data of the wheelsets in the test facility. The camera was set up in the test rig in a strategic position of the wheelset to collect the depth data of the wheels.
The data collected by using Onion Tau Camera was in the form of frame which included the depth data as well as the grey scale images. The processing was followed by conversion of the. frame files to image formats that could be analyzed. Depth images were used as a visualization of the camera distance to the points in the wheelset, amplitude images showed the intensity of LiDAR sign reflections and real images were used to collect the traditional greyscale view of the wheelset. The images were converted into .png format so they can be analyzed and models would be easily trained on them. The preprocessing phase converted depth data into a visual representation that could be used by DL model for its training to detect wear and tear prediction on wheelsets.

Figure 2: the Onion Tau LiDAR Camera

The methodology of data collection ensured collecting of a large DE data in various environments of wheel rotation. It resulted in a useful dataset that was useful in constructing and training DL models to detect defects in wheelsets.

2.3 Models

- **UNET**

The custom U-Net model is a sophisticated encoder-decoder architecture that can extract both detailed features and global features at the same time and predict depth with high accuracy. The encoder is based on number of 2D convolutions with increasingly larger filter sizes (64, 128, 256, and 512) and includes residual connections and dilated convolutions. The residual connections help networks in learning residual mappings that address the vanishing gradients problem in deep networks and dilated convolutions provide a wider receptive field without compromising the spatial resolution. This combination of features allows simultaneous detection of local and global patterns and keeps spatial details. After every convolution operation, max pooling decrease spatial dimensions gradually assists the network to acquire high-level features on various levels.

Transposed convolutions allow the decoder to upsample the feature maps and to get the spatial information again. When the skip connections between the encoder and decoder are merged with the upsampled features using the feature fusion technique rather than concatenation, the performance of the decoder is improved. The decoder can synthesize high-level and detailed properties of the encoder that provides precision in reconstruction of depth maps. The decoder has attention mechanisms that guide the model to give more importance to focus on part of images to achieve better DE. The decoder dropout prevents overfitting, whereas the batch normalization ensures the stability of learning as the feature map normalization takes place across all the layers.

The last layer offers a single-channel depth map which is spatially comparable to the input image based on the predicted depths. The output layer activation is implemented with either sigmoid or tanh depending on the depth value range expected. To optimize the performance of the network, a hybrid loss function, which combines the L2 loss with the structural similarity loss (SSIM), is used to enhance pixel-level accuracy and structural accuracy of depth map predictions. The model uses data augmentation such as random rotation and flipping to improve the generalization process in varied environmental conditions. The architecture also provides accurate estimation of depth of complex tasks like in the case of the inspection of the railway wheelsets by combining high-resolution feature extractions and complex upsampling techniques as shown in Figure 3.

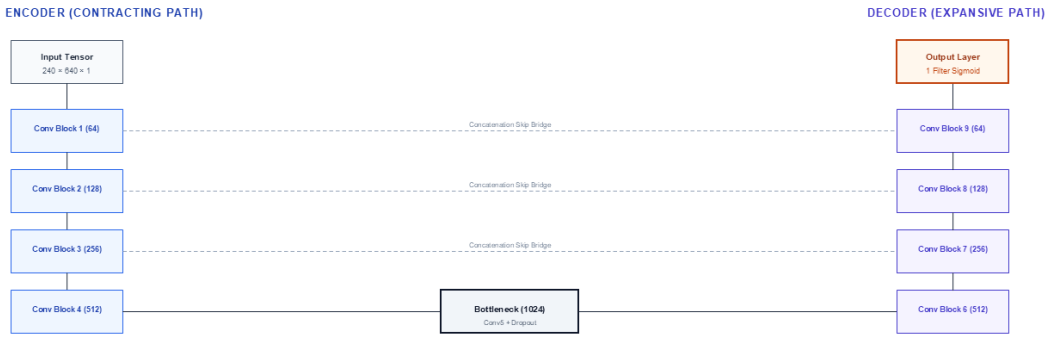

Figure 3: UNET ARCHITECTURE

- **RESNET50**

The proposed ResNet-50-DE custom architecture is based on sequence of convolutional layers that are to extract high-level features at gradually increasing levels with low computational efficiency as shown in Figure 4. This network has an input layer, which takes 240x640 RGB images and goes through the initial convolutional layers, including conv1_pad and conv1_conv that perform padding and convolution of the input images to extract initial features. The batch normalization and ReLU activation are implemented to improve the representation of features and provide non-linearity. The convolutional block is followed by max-pooling with padding to downsample the spatial dimension after which the features are provided on to additional residual blocks.

The residual block of the custom ResNet-50 model is used to enable the training of complex features with the network to maintain its stability throughout the training. Every residual block contains several convolutional layers that are proceeded by batch normalization and ReLU activation. The blocks rely on skip connections to ensure that the residual mappings can be learned by the network to handle the vanishing gradient problem in the backpropagation. The output of every residual block is obtained by adding the output of the convolutions with activation function. This is replicated multiple times and with each repetition downsizing the spatial dimensions and upsizing the depth of the features occur.

The last blocks of the custom ResNet-50 feature additional convolutional and batch normalization blocks with the purpose of narrowing down the feature maps and training them to be used in depth prediction. The components of the final block of the residual are used to produce the final depth map. This architecture is full of deep residual learning to effectively learn the hierarchical structure of the image, gradually

enhancing the representation at any level, and would be useful in scenarios such as DE where high-level spatial relationships are important.

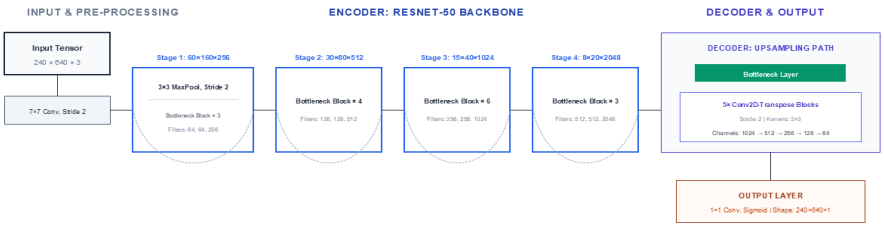

Figure 4: RESNET50 Architecture

- **VIT**

Custom Vision Transformer (ViT) model is an image classification model that utilizes the strength of transformer and CNNs as shown in Figure 5. It starts with the encoder which rescales and normalizes the input images to allow the model to learn effectively, and then ZeroPadding2D to adapt the spatial dimensions. The convolutional blocks contain Conv2D and DepthwiseConv2D layers, that capture local features, and Residual connections are used to preserve gradient to resolve the vanishing gradient problem. Addition of max-pooling also downsizes the spatial dimensions, and the model focuses on the high-level representations of the image but loss the important details at various scales.

Conv2DTranspose layers are employed in the decoder upsampling the feature maps, that refine the spatial resolution of the images. Encoders use skip connections to keep fine-grained features while attention mechanism to ensure the model gives attention to specific areas that are most important to enhance the accuracy of the model. Dropout layers help to avoid overfitting whereas batch normalization helps to stabilize learning, which ensures smooth convergence and solid performance in a wide variety of training conditions. All these methods combined to ensure that the model is effective in images reconstruction and capturing both local and global features. The output layer is a multi-class classification using a softmax activation function. To enhance the model generalization, it is trained using categorical cross-entropy loss and data augmentation methods such as random rotations and flipping. This architecture integrates transformer-based attention, convolutional feature extraction, and up sampling. It is good model for complex image classification problems with high performance and accurate predictions even with heterogenous visual datasets.

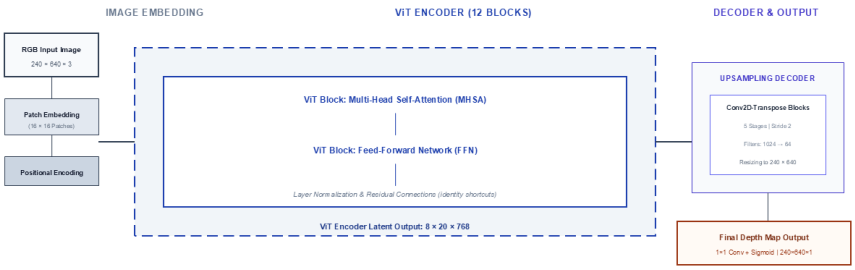

Figure 5: Vit Transformer

2.4 Evaluation Metrics

Evaluation metrics are essential for evaluating the performance of ML and DL models, particularly for regression and classification tasks. These metrics help in understanding the accuracy, error, and efficiency of the model in creating predictions. Some commonly used evaluation metrics with their formulas are given below:

- **Loss**

  The loss function is used to represent the difference between the predicted and actual values is shown in Equation (1). For classification, it could use cross-entropy loss, and for regression, it could mean squared error.

$$loss = \frac{1}{N}\sum_{i=1}^{N}\mathcal{L}\,(y_i, \hat{y}_i)$$

Where N is the number of data points, $\boldsymbol{y_i}$ is the true value, and $\hat{\boldsymbol{y}}_{\boldsymbol{i}}$ is the predicted value.

- **Mean Absolute Error (MAE)**

MAE measures the average magnitude of the errors in a set of predictions, without considering their direction which is given in Equation (2).

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i|$$

- **Mean Squared Error (MSE)**

MSE measures the average squared difference between the true values and the predicted values. It gives higher weight to larger errors which is given in Equation (3).

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2$$

- **Root Mean Squared Error (RMSE)**

RMSE is the square root of the MSE, providing error values in the same units as the original data. It is sensitive to outliers which is given in Equation (4).

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

These metrics show the performance of models, that helps in model selection and tuning for optimal accuracy and error reduction.

3. Results

3.1 System Specifications

The models were trained using a computer system that consists of an Intel i7 9th generation processor with 16GB of RAM that allows processing big datasets and model parameters. It has also NVIDIA GeForce RTX 2060 Super graphics card and 8GB VRAM, effective in providing acceleration of graphics in DL model training. It includes a 256GB SSD to support rapid read/write functions and 1 TB hard disk to provide more storage. These resources helped to train and test the models successfully.

3.2 Hyperparameters

These hyperparameters were used for all the models as shown in Table 1:

Hyperparameter	**Value**
Learning Rate	1×10^{-4}
Optimizer	Adam
Loss Function	Mean Squared Error (MSE)
Metrics	Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE)
Batch Size	16

Table 1: Hyperparameters

The hyperparameters were standard with all models' implementation as their performance had to be compared and to optimise the training process to predict depth. The learning rate was set to 1×10^{-4} to maintain the training stability whereas the Adam optimizer was selected due to its ability to manage sparse gradients efficiently. The Evaluation measures (MAE, MSE and RMSE) have been used to evaluate the

performance of the respective models. Training of the models was implemented with batch size of 16, a balance was kept between the training time and the performance of the model.

3.3 Results and Comparative Analysis

The three models were compared using multiple evaluation metrics, including loss, MAE, MSE, and RMSE, with all models trained and validated on the same dataset to ensure a fair assessment of efficiency as shown in Table 2. The UNet model achieved a training loss of 0.0134 and a validation loss of 0.0160, with a validation MAE of 0.0889, MSE of 0.0160, and RMSE of 0.1267. These results indicate that UNet maintained low error during both training and validation, with a minimal gap between the two, suggesting strong generalization capability. The relatively low RMSE further reflects its ability to produce accurate depth predictions, which is particularly important in fine-grained depth estimation tasks. In contrast, ResNet50 achieved a lower training loss of 0.0051 but a significantly higher validation loss of 0.1516, along with a validation MAE of 0.3273, MSE of 0.1516, and RMSE of 0.3894. Although ResNet50 learned the training data effectively, the large discrepancy between training and validation metrics indicates overfitting and weaker generalization performance. Vision Transformer (ViT) exhibited an even greater gap between training and validation results, with a training loss of 0.0067 and a validation loss of 0.2616, and validation MAE, MSE, and RMSE values of 0.4230, 0.2616, and 0.5114, respectively. The substantially higher validation errors suggest that ViT struggled most with generalization, likely due to limitations in capturing fine spatial relationships essential for accurate depth estimation. Overall, UNet demonstrated the best balance between accuracy and generalization, likely benefiting from its encoder–decoder architecture with skip connections, whereas ResNet50 and ViT would require further hyperparameter tuning or architectural modifications to improve validation performa

Model	**Training Loss**	**Validation Loss**	**Validation MAE**	**Validation MSE**	**Validation RMSE**
UNet	0.0134	0.0160	0.0889	0.0160	0.1267
ResNet50	0.0051	0.1516	0.3273	0.1516	0.3894
ViT	0.0067	0.2616	0.4230	0.2616	0.5114

Table 2: Comparative Analysis

4. Conclusions and Contributions

In this study, we compared the performance of three DL architectures including UNet, ResNet50, and Vision Transformer (ViT) for DE. The results demonstrated that UNet outperformed both ResNet50 and ViT, achieving the lowest testing errors and better generalization. ResNet50 achieved low training error but overfitted, while ViT struggled with the highest validation error. These findings suggest that UNet's encoder-decoder architecture with skip connections is particularly good for DE.

However, future work could focus on fine-tuning ResNet50 and ViT models, potentially incorporating hybrid architectures, to improve their performance for DE.

Acknowledgements

This research is supported by Departamento de Ingeniería de Comunicaciones, Universidad de Malaga, Spain, National Center for Robotics, Automation and Artificial Intelligence (NCRAAI MUET), Higher Education Commission Pakistan, Sindh Higher Education Commission Pakistan, NCRA-CMS Lab of Mehran University of Engineering and Technology, Jamshoro, and by the doctoral program of Mechanical Engineering and Energy Efficiency, School of Industrial Engineering, University of Malaga, Spain. We would also like to thank Pakistan Railways, Kotri Railway Station, and Carriage and wagon workshop Hyderabad near American quarters, Pakistan railways for their assistance and support. Furthermore, we would like to acknowledge the International Electronic Machines Corporation of USA, EU Funded Erasmus Plus Capacity Building in Higher Education ACTIVE Climate Action Project and Capacity building and Exchange towards attaining Technological Research and modernizing Academic Learning- CENTRAL Project, ID: 598914, Programme: EPLUS - Erasmus+ Capacity Building in Higher Education, Grant Agreement Number: EAC/A05/2017- Project, Reference: 598914-EPP-1-2018-1-DKEPPKA2- CBHE-J. We would also acknowledge Institute of Oceanic Engineering Research of the University of Málaga for their support.

References

[1] M. Poggi, F. Tosi, K. Batsos, P. Mordohai, and S. Mattoccia, 'On the Synergies between Machine Learning and Binocular Stereo for Depth Estimation from Images: A Survey', *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 9, pp. 5314–5334, Sep. 2022, doi: 10.1109/TPAMI.2021.3070917.

[2] K. Guo, S. Song, S. Chang, T. U. Kim, S. Han, and I. Kim, 'Robust Full-Fov Depth Estimation in Tele-Wide Camera System', *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, vol. 2020-May, pp. 1993–1997, May 2020, doi: 10.1109/ICASSP40776.2020.9053724.

[3] C. Zhao, Q. Sun, C. Zhang, Y. Tang, and F. Qian, 'Monocular Depth Estimation Based On Deep Learning: An Overview', *Sci. China Technol. Sci.*, vol. 63, no. 9, pp. 1612–1627, Jul. 2020, doi: 10.1007/s11431-020-1582-8.

[4] A. Masoumian, D. G. F. Marei, S. Abdulwahab, J. Cristiano, D. Puig, and H. A. Rashwan, 'Absolute Distance Prediction Based on Deep Learning Object Detection and Monocular Depth Estimation Models', *Frontiers in Artificial Intelligence and Applications*, vol. 339, pp. 325–334, Oct. 2021, doi: 10.3233/FAIA210151.

[5] F. Tosi, F. Aleotti, M. Poggi, and S. Mattoccia, ‘Learning monocular depth estimation infusing traditional stereo knowledge’. [Online]. Available: https://github.com/

[6] F. Liu, C. Shen, and G. Lin, ‘Deep convolutional neural fields for depth estimation from a single image’, *Computer Vision and Pattern Recognition*, vol. 07-12-June-2015, pp. 5162–5170, Oct. 2014, doi: 10.1109/CVPR.2015.7299152.

[7] F. Liu, C. Shen, G. Lin, and I. Reid, ‘Learning Depth from Single Monocular Images Using Deep Convolutional Neural Fields’, *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 10, pp. 2024–2039, Oct. 2016, doi: 10.1109/TPAMI.2015.2505283.

[8] X. Xu, Y. Li, G. Wu, and J. Luo, ‘Multi-modal deep feature learning for RGB-D object detection’, *Pattern Recognit.*, vol. 72, pp. 300–313, Dec. 2017, doi: 10.1016/j.patcog.2017.07.026.

[9] H. Zhan, R. Garg, C. S. Weerasekera, K. Li, H. Agarwal, and I. M. Reid, ‘Unsupervised Learning of Monocular Depth Estimation and Visual Odometry with Deep Feature Reconstruction’, *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 340–349, Mar. 2018, doi: 10.1109/CVPR.2018.00043.

[10] J. Liu, Q. Li, R. Cao, W. Tang, and G. Qiu, ‘MiniNet: An extremely lightweight convolutional neural network for real-time unsupervised monocular depth estimation’, *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 166, pp. 255–267, Aug. 2020, doi: 10.1016/j.isprsjprs.2020.06.004.