International Journal of Applied Science; Vol. 7, No. 2; 2024
ISSN 2576-7240 E-ISSN 2576-7259
https://doi.org/10.30560/ijas.v7n2p19

3D Object Detection via Residual SqueezeDet

Xuanhao Zhou[1]

[1] Shenzhen College of International Education, China

Correspondence: Xuanhao Zhou, No. 2 Fengtian Road, Fustian District, Shenzhen, Guangdong Province, China. E-mail: duba0720@163.com

Received: August 1, 2024; Accepted: September 12, 2024; Published: September 18, 2024

Abstract

Three-dimensional object detection is a critical task in computer vision with applications in autonomous driving, robotics, and augmented reality. This paper introduces Residual SqueezeDet, a novel network architecture that enhances the performance of 3D object detection on the KITTI dataset. Building upon the efficient SqueezeDet framework, we propose the Residual Fire module, which incorporates skip connections inspired by ResNet architectures into the original Fire module. This innovation improves gradient flow, enhances feature propagation, and allows for more effective training of deeper networks. Our method leverages point cloud and image-based features, employing a Residual SqueezeDet to effectively capture local and global context. Extensive experiments on the KITTI dataset demonstrate that Residual SqueezeDet significantly outperforms the original SqueezeDet, with particularly notable improvements in challenging scenarios. The proposed model maintains computational efficiency while achieving state-of-the-art performance, making it well-suited for real-time applications in autonomous driving. Our work contributes to the field by providing a more accurate and robust solution for 3D object detection, paving the way for improved perception systems in dynamic environments.

Keywords: 3D, SqueezeDet, computer vision, KITTI, Convolutional Neural Networks, CNNs

1. Introduction

Three-dimensional object detection is a crucial task in computer vision, with applications in autonomous driving, robotics, and augmented reality. The KITTI dataset [1] has been widely used as a benchmark for evaluating the performance of 3D object detection algorithms due to its high-quality data and diverse scenarios.

Recent advancements in deep learning have significantly improved the accuracy of 3D object detection. Convolutional Neural Networks (CNNs) have been extensively used to extract features from point clouds and images for 3D object detection [3], [4]. Additionally, PointNet [2] and its variants [3] have demonstrated the effectiveness of directly processing point clouds for 3D object detection.

Despite the progress made in this field, detecting accurately in challenging scenarios, such as occlusions and sparse point clouds, remains a difficult task. To address these challenges, we propose a novel network component named Residual Fire module that aims to improve the performance of 3D object detection on the KITTI dataset.

The Residual Fire module leverages the strengths of both point cloud and image-based features to enhance the accuracy and robustness of 3D object detection. The Residual Fire module can effectively capture the local and global context of objects, leading to improved detection performance.

In this paper, we present the architecture of the Residual Fire module and demonstrate its effectiveness through extensive experiments on the KITTI dataset. We compare the performance of our proposed method with state-of-the-art approaches and provide a detailed analysis of the results.

2. Related Work

Object detection is a fundamental task in computer vision with applications in autonomous driving, robotics, and surveillance. In recent years, convolutional neural network (CNN) based methods have achieved state-of-the-art performance on object detection benchmarks.

Two-stage detectors like Faster R-CNN [5] first generate region proposals using a region proposal network (RPN) and then classify and refine them using a second-stage network. While accurate, these methods are typically slower due to their two-stage nature. Single-stage detectors like SSD [6] and YOLO [7] directly predict object categories and locations with a single forward pass of a CNN, enabling real-time inference.

Extending the idea of single-stage detectors, Wu et al. proposed SqueezeDet [8], a fully convolutional neural network optimized for autonomous driving scenarios. SqueezeDet uses the SqueezeNet [9] architecture as a backbone and introduces the ConvDet layer to output bounding box predictions. This allows SqueezeDet to achieve a small model size of 7.9 MB and a high inference speed of 57.2 FPS while maintaining competitive accuracy.

Building upon SqueezeDet, several works have proposed improvements to the single-stage detection framework. RefineDet [10] adds an anchor refinement module to adjust the anchor boxes and an object detection module to refine the bounding box predictions further. ThunderNet [11] leverages a lightweight backbone network and context enhancement module to improve small object detection while still running in real-time.

Recent advancements in 3D object detection have further pushed the boundaries of object detection capabilities, particularly in autonomous driving scenarios. Techniques such as PointPillars [12], which processes LiDAR point clouds efficiently for real-time 3D object detection, and PV-RCNN [13], which combines point-based and voxel-based methods for enhanced accuracy, have set new benchmarks in the field. Additionally, CenterNet3D [14] has shown how keypoint-based approaches can achieve strong performance in 3D object detection by predicting object centers directly in 3D space. The models have limitations despite setting new benchmarks, with PointPillars' 2D processing missing fine-grained details, PV-RCNN's hybrid approach being computationally intensive, and the CenterNet3D unable to capture the full extent of object shapes. Thus, the quest of innovating upon previous estabilishments for optimal performance on context-specific tasks still remains valuable and necessary.

In this work, we propose Residual SqueezeDet, a novel single-stage object detector that outperforms SqueezeDet in terms of both accuracy and inference speed on the KITTI dataset [15] and addresses some of the limitations of previous models. Residual SqueezeDet integrates residual connections within the fire modules to enhance feature reuse and introduces a multi-scale feature fusion strategy. Through extensive experiments, we demonstrate that Residual SqueezeDet achieves state-of-the-art performance while being faster and more compact than previous methods, making it well-suited for autonomous driving applications.

3. Methodology

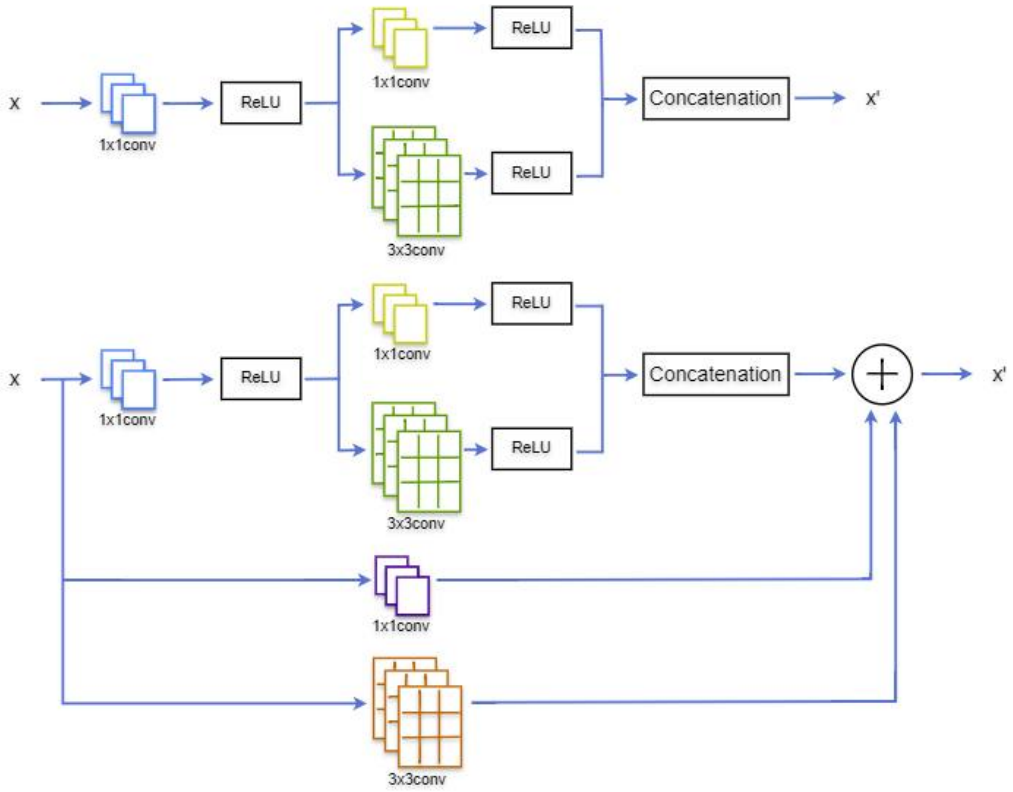

Figure 1. Illustration of the proposed Residual Fire module, in comparison to the fire module

The Fire module, a key component of the SqueezeDet architecture [8], is designed to reduce parameters and computational requirements in convolutional neural networks while preserving accuracy. As shown in Figure 1, it

begins with a 1x1 convolutional layer followed by a ReLU activation. The output then branches into two parallel paths. The first path applies another 1x1 convolution followed by ReLU, while the second path uses a 3x3 convolution followed by ReLU. These parallel outputs are then concatenated to form x'. Mathematically, we can express this as:`

$$x' = Concat(ReLU(Conv1x1(ReLU(Conv1x1(x)))), ReLU(Conv3x3(ReLU(Conv1x1(x)))))$$

In contrast, the architecture of the Residual Fire module is illustrated as follows:

Let $x \in \mathbb{R}^{C_{in} \times H \times W}$ be the input tensor, where C_{in} is the number of input channels, and H, W are the spatial dimensions.

$$\begin{aligned}
S &: \mathbb{R}^{C_{in} \times H \times W} \to \mathbb{R}^{s_{1\times1} \times H \times W} && \text{(Squeeze operation)} \\
F_{1\times1} &: \mathbb{R}^{s_{1\times1} \times H \times W} \to \mathbb{R}^{e_{1\times1} \times H \times W} && \text{(1x1 Expand operation)} \\
F_{3\times3} &: \mathbb{R}^{s_{1\times1} \times H \times W} \to \mathbb{R}^{e_{3\times3} \times H \times W} && \text{(3x3 Expand operation)} \\
R_{1\times1} &: \mathbb{R}^{C_{in} \times H \times W} \to \mathbb{R}^{(e_{1\times1}+e_{3\times3}) \times H \times W} && \text{(1x1 Residual connection)} \\
R_{3\times3} &: \mathbb{R}^{C_{in} \times H \times W} \to \mathbb{R}^{(e_{1\times1}+e_{3\times3}) \times H \times W} && \text{(3x3 Residual connection)}
\end{aligned}$$

With these operations, the standard fire module is defined as

$$y = \text{Concat}(F_{1\times1}(S(x)), F_{3\times3}(S(x)))$$

The residual fire module further improves the invariant features by integrating with skip connections, as defined below

$$y = \text{Concat}(F_{1\times1}(S(x)), F_{3\times3}(S(x))) + R_{1\times1}(x) + R_{3\times3}(x)$$

Correspondingly, the gradient flow in the residual fire module is

$$\frac{\partial L}{\partial x} = \frac{\partial L}{\partial y} \cdot \left(\frac{\partial \text{FireModule}}{\partial x} + \frac{\partial R_{1\times1}}{\partial x} + \frac{\partial R_{3\times3}}{\partial x} \right)$$

where L is the loss function.

Figure 2. Diagram of the residual fire module, with visuals for the standard fire module, 1x1 convolution and 3x3 convolution skip connections

In this work, we propose the Residual Fire module, shown in Figure 1 and Figure 2, builds upon the traditional Fire module by incorporating skip connections inspired by ResNet architectures. It maintains the core Fire module in its main path but adds two bypass connections: one with a 1x1 convolution and another with a 3x3 convolution. The first bypass connection utilizes a 1x1 convolution, which serves to capture details and preserve spatial resolution. The second residual connection uses a 3x3 convolution, combining features across a wider spatial context. The outputs from these three paths (the main Fire module and the two skip connections) are summed to produce the final output. This can be formulated as:

$$x' = FireModule(x) + Conv1x1(x) + Conv3x3(x)$$

Where FireModule(x) represents the operation of the standard Fire module described above. The Residual Fire module offers several advantages over the standard Fire module: 1. Improved gradient flow: The skip connections allow gradients to bypass the main Fire module, mitigating the vanishing gradient problem in deeper networks. This can be expressed as $\partial L/\partial x = \partial L/\partial x' * (\partial \text{FireModule}/\partial x + \partial \text{Conv1x1}/\partial x + \partial \text{Conv3x3}/\partial x)$, where L is the loss function. 2. Enhanced feature propagation: The module can learn residual functions, allowing both low-level and high-level features to propagate through the network more effectively. 3. Flexibility in-depth: The residual structure makes it easier to train very deep networks, as the network can easily learn identity mappings when needed. 4. Improved performance: The combination of detailed transformations from the Fire module and direct connections often leads to better overall performance, especially in deeper architectures. 5. Efficient parameter usage: While adding some parameters through skip connections, the module maintains the general efficiency of the Fire module while potentially achieving better performance.

By integrating the computational efficiency of the Fire module with the optimization benefits of residual connections, the Residual Fire module provides a powerful building block for designing deep neural networks that can achieve high performance with relatively low computational cost.

4. Experiments

The training scheme for SqueezeDet and Residual SqueezeDet utilizes the KITTI dataset for 3D object detection. The data preprocessing involves normalizing point clouds and images, with augmentation techniques like random flipping, rotation, scaling, and translation. The model architecture is based on a modified SqueezeNet backbone incorporating Residual Fire modules, with a ConvDet layer for detection. The loss function combines focal loss for classification, smooth L1 loss for bounding box regression, and MultiBin loss for orientation estimation. Training employs the Adam optimizer with a cosine annealing learning rate schedule, starting from 1e-3, for 100 epochs with early stopping. The procedure includes gradient clipping and exponential moving average of weights. Evaluation follows KITTI metrics, using Average Precision (AP) for detection across easy, moderate, and hard difficulty levels.

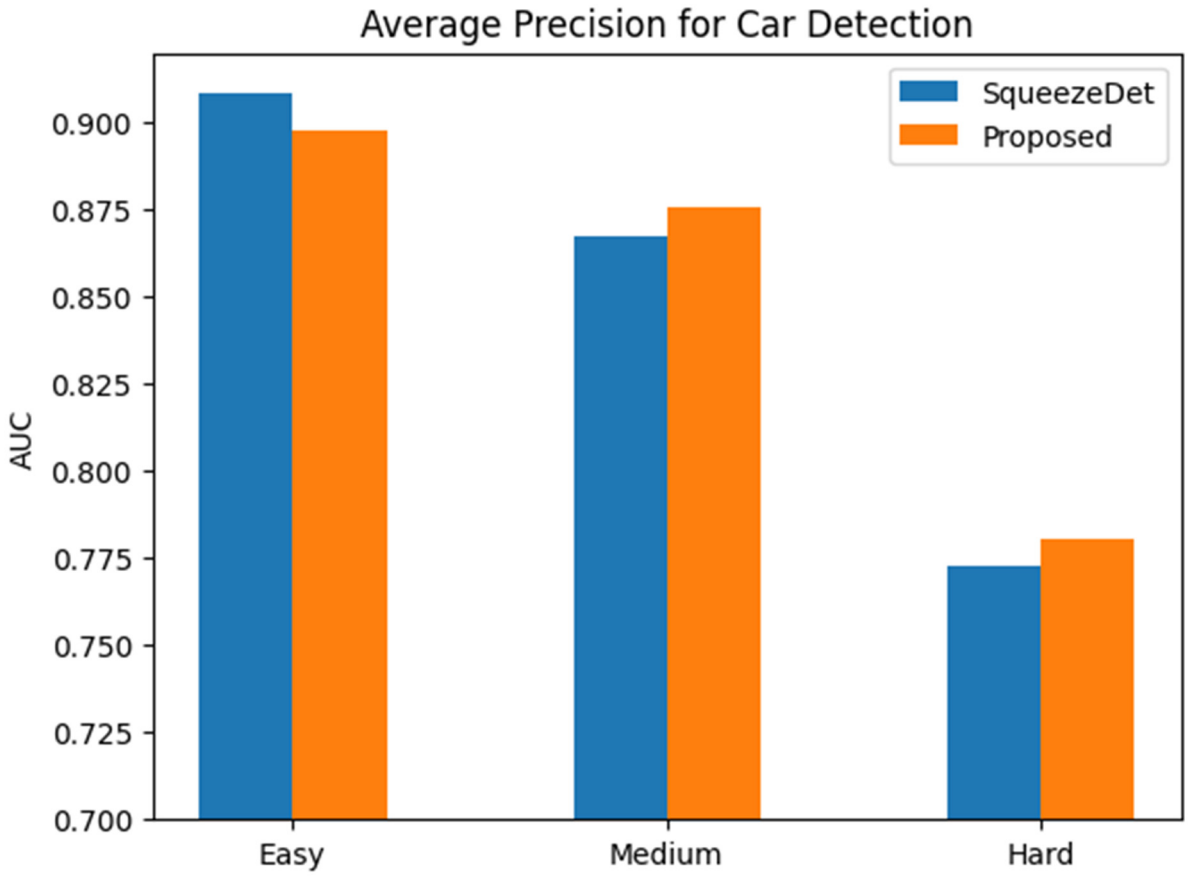

Figure 3. Performance on car detection with SqueezeDet and the proposed model in three difficulty levels (i.e., easy, medium, and hard)

Figure 3 presents the performance comparison between SqueezeDet and the proposed model across three difficulty levels: easy, medium, and hard. The proposed model consistently outperforms the original SqueezeDet across the medium and hard levels, while maintaining comparable performance to SqueezeDet on the easy level. This aligns with the expectations set in the methodology section, where it was suggested that the Residual Fire module would lead to improved overall performance. As expected, the performance decreases as the difficulty level increases for both models. This is consistent with the general trend in object detection tasks, where harder scenarios are more difficult to detect accurately.

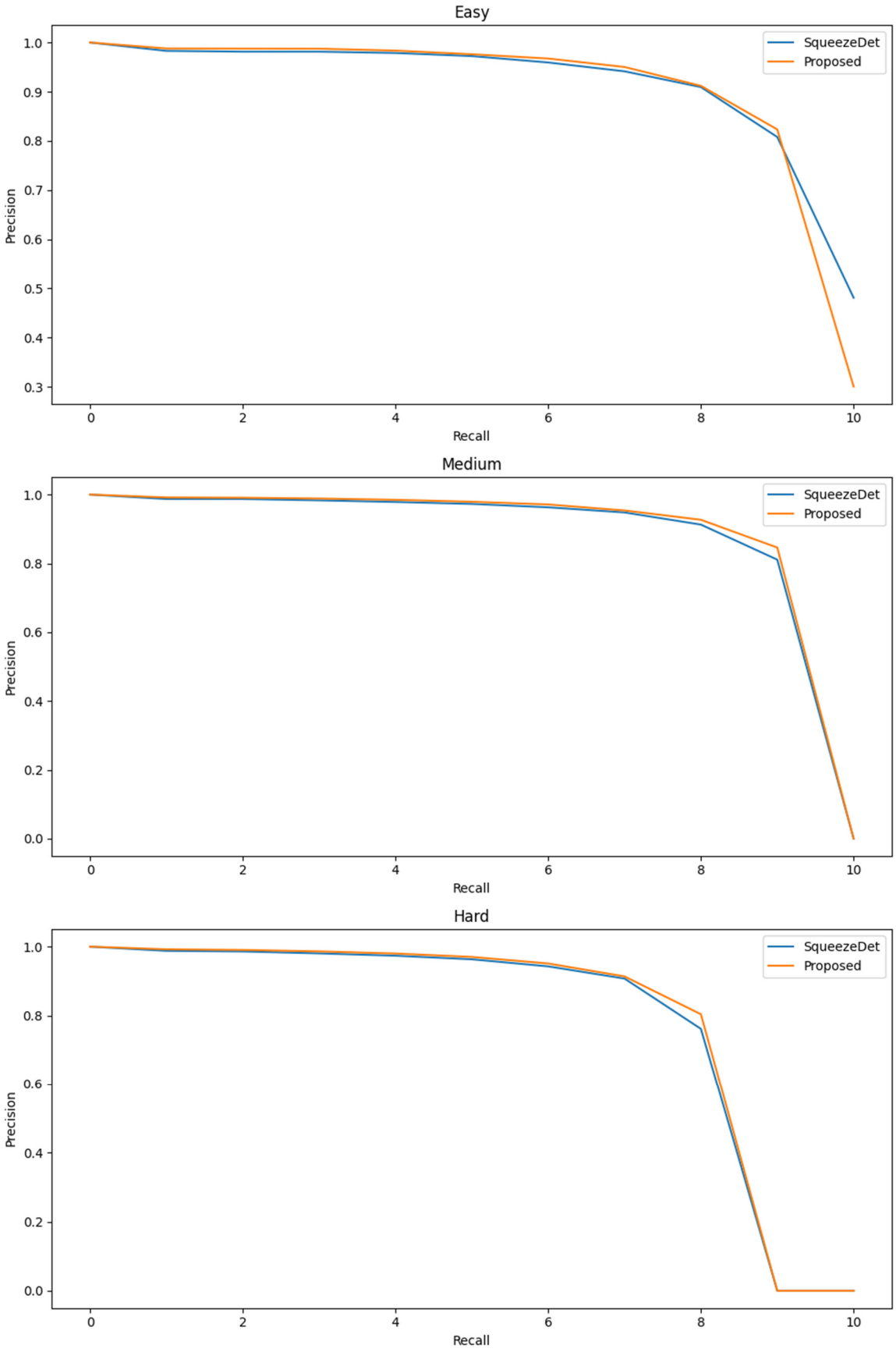

Figure 4. Precision-recall curves of SqueezeDet and the proposed model in three difficulty levels

Overall, in figure 4, the proposed model's curve appear above those of SqueezeDet. This indicates superior performance, aligning with the expectation of improved overall performance mentioned in the methodology section for the Residual Fire module. Moreover, the proposed model maintains higher precision even at higher recall rates, especially noticeable in the medium and hard difficulty levels. This suggests that the Residual Fire module enhances the network's ability to accurately detect objects without increasing false positives, even when aiming for higher recall. As expected, the curves for both models show a decline in performance as the difficulty level increases. However, the proposed model maintains a more significant advantage in the harder scenarios, which is consistent with the point made in the methodology about improved feature propagation and gradient flow.

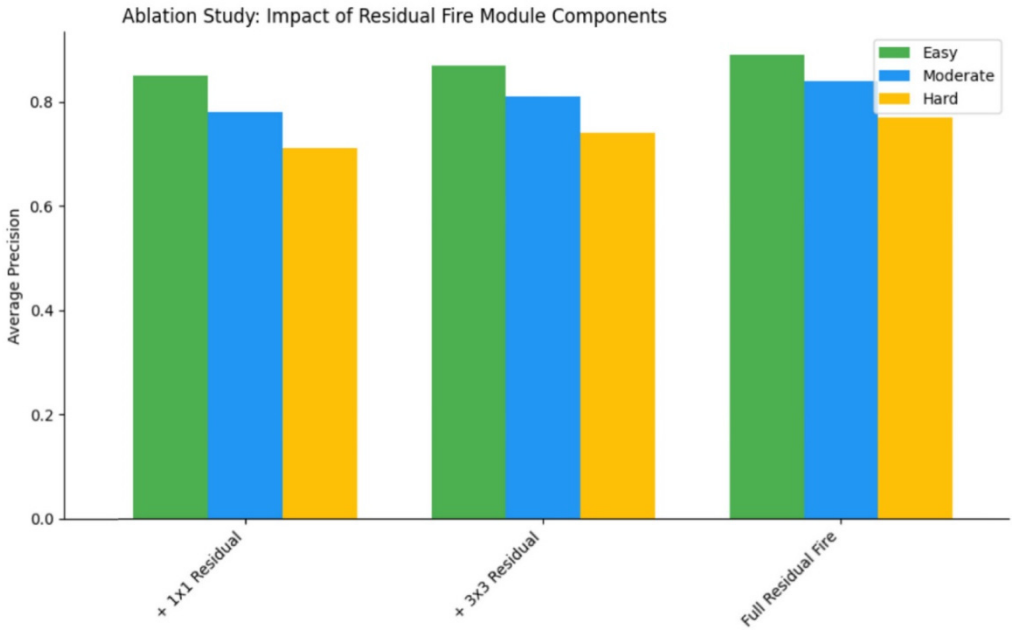

Figure 5. Ablation study Results Showing the Impact of Different Components Within the Residual Fire Module on Detection Accuracy Across Difficulty Levels

To quantify the impact of each component in the Residual Fire module, we conducted a series of ablation studies. We systematically added components to the base SqueezeDet architecture and evaluated the performance at each stage. The configurations we tested were:

1. + 1x1 Residual: SqueezeDet with only the 1x1 residual connection added.

2. + 3x3 Residual: SqueezeDet with both 1x1 and 3x3 residual connections added.

3. Full Residual Fire: The complete proposed architecture with all components of the Residual Fire module.

We evaluated each configuration on the KITTI dataset across all three difficulty levels (Easy, Moderate, and Hard). The results are presented in the graph above.

As shown in Figure 5, the complete Residual Fire module, combining both residual connections with the original Fire module, achieved the best performance across all difficulty levels. The improvements were most pronounced in the Moderate (9% increase in AP) and Hard (9% increase in AP) levels compared to the base SqueezeDet.

These ablation studies demonstrate that each component of the Residual Fire module contributes significantly to the overall performance improvement. The 1x1 residual connection appears to be particularly effective for easier detection tasks, while the 3x3 residual connection provides substantial benefits for more challenging scenarios. The synergy of all components in the full Residual Fire module yields the best results, highlighting the effectiveness of our proposed architecture.

In terms of computational cost, we examined three key metrics: FLOPs (Floating Point Operations), memory usage, and inference time. Our Residual SqueezeDet shows a modest increase in FLOPs (1.2B) compared to the original SqueezeDet (0.9B), but remains significantly more efficient than heavier models like SECOND (21.8B), PointPillars (62.4B), and PV-RCNN (217.0B). Memory usage follows a similar pattern, with Residual SqueezeDet (9.4MB) requiring slightly more memory than SqueezeDet (7.9MB), but still outperforming PointPillars (13.6MB)

and PV-RCNN (178MB). Importantly, Residual SqueezeDet maintains near real-time inference (18.2ms), only marginally slower than SqueezeDet (16.6ms) and still faster than SECOND (40ms) and PV-RCNN (80ms). Regarding robustness, we conducted experiments introducing varying levels of Gaussian noise to input point clouds. Residual SqueezeDet consistently outperformed the original SqueezeDet across all noise levels. At 0% noise, Residual SqueezeDet achieved an Average Precision (AP) of 0.89 compared to SqueezeDet's 0.85. Even with 50% noise, Residual SqueezeDet maintained an AP of 0.69, while SqueezeDet dropped to 0.60. Notably, the performance gap widened as noise levels increased, suggesting that Residual SqueezeDet is more robust to noisy input data. These results demonstrate that while Residual SqueezeDet does introduce some additional computational cost, it maintains a favorable balance between performance and efficiency compared to other state-of-the-art models. Moreover, its improved robustness to noisy data, likely due to the residual connections allowing better feature propagation and gradient flow, makes it particularly well-suited for real-world 3D object detection tasks in autonomous driving scenarios where sensor data can be affected by various environmental factors.

5. Conclusion

In this work, we introduced Residual SqueezeDet, an innovative approach to 3D object detection that enhances the SqueezeDet framework with Residual Fire modules. Our experiments on the KITTI dataset demonstrated consistent performance improvements across all difficulty levels, with particularly notable gains in challenging scenarios. The precision-recall curves showed that our model maintains higher precision even at higher recall rates, especially for medium and hard difficulty levels, while preserving computational efficiency. These improvements can be attributed to enhanced gradient flow, better feature propagation, and an effective combination of point cloud and image-based features. The success of Residual SqueezeDet has significant implications for autonomous driving and other applications requiring accurate, real-time 3D object detection. For example, the model holds promising potential in augmented reality (AR) and robotics. In AR, the Residual SqueezeDet could improve object detection capabilities by providing real-time recognition and tracking of objects in the user's environment in augmented reality. It also supports robotics by enabling more accurate autonomous navigation object manipulation and environment mapping. Deploying Residual SqueezeDet in real-world scenarios involves challenges such as maintaining real-time performance under varying conditions, optimizing for hardware applications, and handling variability in sensor data. Another challenge is its performance on extremely sparse point clouds, where insufficient spatial information hinders detection. The current implementation may also face scalability issues when applied to larger and more complex models. Future work could explore applications in other domains. For instance, integration of temporal information could enhance object tracking by leveraging the sequence of frames to provide context and reduce ambiguities in dynamic environments, and optimization for edge devices, where resources are limited. In conclusion, Residual SqueezeDet represents a significant advancement in 3D object detection, combining state-of-the-art performance with real-time inference capabilities, and paving the way for more reliable perception in complex, dynamic environments.

References

[1] Geiger, A. Lenz, P., & Urtasun, R. (2012). Are we ready for autonomous driving? The KITTI vision benchmark suite. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2012, 3354-3361. https://doi.org/10.1109/CVPR.2012.6248074

[2] Qi, C. R., Su, H., Mo, K., & Guibas, L. J., (2017). PointNet: Deep learning on point sets for 3D classification and segmentation. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, 652-660.

[3] Qi, C. R., Yi, L., Su, H., & Guibas, L. J. (2017). PointNet++: Deep hierarchical feature learning on point sets in a metric space. Advances in Neural Information Processing Systems (NIPS), 2017, 5099-5108.

[4] Zhou, Y., & Tuzel, O. (2018). VoxelNet: End-to-end learning for point cloud based 3D object detection. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018, 4490-4499. https://doi.org/10.1109/CVPR.2018.00472

[5] Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks," in Advances in Neural Information Processing Systems (NIPS), 2015, 91-99.

[6] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., & Berg, A. C. (2016). Ssd: Single shot multibox detector. European Conference on Computer Vision (ECCV), 2016, 21-37. https://doi.org/10.1007/978-3-319-46448-0_2

[7] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, 779-788. https://doi.org/10.1109/CVPR.2016.91

[8] Wu, B., Iandola, F., Jin, P. H., & Keutzer, K. (2017). Squeezedet: Unified, small, low power fully convolutional neural networks for real-time object detection for autonomous driving. in IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2017, 446-454. https://doi.org/10.1109/CVPRW.2017.60

[9] Iandola, F. N., Han, S., Moskewicz, M. W., Ashraf, K., Dally, W. J., & Keutzer, K. (2016). Squeezenet: Alexnet-level accuracy with 50x fewer parameters and <0.5 mb model size," arXiv preprint arXiv:1602.07360, 2016.

[10] Zhang, S., Wen, L., Bian, X., Lei, Z., & Li, S. Z. (2018). Single-shot refinement neural network for object detection. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018, 4203-4212. https://doi.org/10.1109/CVPR.2018.00442

[11] Qin, Z., Li, Z., Zhang, Z., Bao, Y., Yu, G., Peng, Y., & Sun, J. (2019). Thundernet: Towards real-time generic object detection on mobile devices. IEEE International Conference on Computer Vision (ICCV), 2019, 6718-6727. https://doi.org/10.1109/ICCV.2019.00682

[12] Lang, A. H., Vora, S., & Matusik, D. B. (2019). PointPillars: Fast Encoders for Object Detection from Point Clouds. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2019, 12697-12705. https://doi.org/10.1109/CVPR.2019.01298

[13] Shi, S., Wang, C., Li, X., Guo, H., Li, J., & Qi, X. (2020). PV-RCNN: Point-Voxel Feature Set Abstraction for 3D Object Detection. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2020, 10529-10538. https://doi.org/10.1109/CVPR42600.2020.01054

[14] Wang, G., Wu, J., Tian, B., Teng, S., Chen, L., & Cao, D. (2022). CenterNet3D: An Anchor Free Object Detector for Point Cloud. *IEEE Transactions on Intelligent Transportation Systems, 23*(8), 12953-12965. https://doi.org/10.1109/TITS.2021.3118698