MSC 68T45, 68T40 **DOI: 10.14529/jcem260204**

MONOCULAR 3D DETECTION OF MOVING OBJECTS FROM UAV BASED ON SPATIO-TEMPORAL FEATURE ANALYSIS

I. D. Goncharov[1], goncharovid@susu.ru,
V. A. Surin[1], surinva@susu.ru
[1]South Ural State University, Chelyabinsk, Russian Federation

This article presents an approach to monocular 3D object detection for Unmanned Aerial Vehicles (UAVs) in the absence of external telemetry. We propose an architecture that leverages temporal context to implicitly extract independent object motion. A methodology for ego-motion compensation and a hybrid depth estimation model are described. Furthermore, we present a synthetic data generation pipeline within the CARLA environment and provide preliminary localization accuracy results. The proposed method enables real-time performance on the NVIDIA Jetson Orin platform.

Keywords: monocular 3D detection; UAV; temporal fusion; ego-motion compensation; CARLA simulator; Jetson Orin.

Introduction

The rapid development of unmanned aerial systems in recent years has led to their widespread adoption in urban infrastructure monitoring, security, and autonomous delivery tasks. The transition toward full UAV autonomy requires perception systems capable of generating detailed 3D models of dynamic environments in real time. Unlike ground-based robots, small-scale aerial platforms are characterized by strictly limited power capacity and payload constraints, rendering the use of active laser rangefinders (LiDAR) economically and technically impractical [1]. Under these conditions, the primary responsibility shifts to monocular vision methods, which must provide spatial orientation despite the scarcity of data from external telemetry sensors.

1. Problem Statement

Recovering the 3D kinematics of a scene from a single moving video source represents a fundamentally ill-posed computer vision task. When deployed on Unmanned Aerial Vehicles (UAVs) with strict weight and power consumption constraints, this problem is exacerbated by the impossibility of using heavy active rangefinders, such as LiDAR, or computationally expensive dense optical flow algorithms. Historically, autonomous navigation systems circumvent this issue through hardware integration-fusing visual features with data from inertial measurement units (IMUs) and satellite systems. However, in the absence of telemetry, such perception methods completely fail.

The primary theoretical barrier in monocular vision remains the scale ambiguity. Projective geometry dictates that an infinite set of objects of varying physical sizes at different distances can project onto the camera sensor with an identical pixel area. Purely neural-network-based approaches that attempt to directly regress absolute depth from a single RGB frame are prone to severe overfitting on specific background textures and

camera heights [2]. Further complexity is introduced by the parallax effect: the UAV's ego-motion causes a shift in the entire background, which classical motion detectors erroneously interpret as independent object dynamics. Traditional Visual SLAM (Simultaneous Localization and Mapping) solves the inverse problem: it filters out moving objects as noise to focus on static map construction, making it inapplicable for instantaneous dynamic object detection tasks [3].

Recently, anchor-free architectures, such as CenterNet [1] and SMOKE [2], have demonstrated that discarding intermediate 2D bounding boxes significantly improves 3D regression accuracy. Nevertheless, the analysis of single discrete frames deprives the model of critical kinematic context. The algorithm is unable to differentiate between a parked vehicle and one moving at high speed without resorting to secondary tracking heuristics.

In this paper, we propose a qualitatively different approach: an end-to-end spatio-temporal monocular detection architecture with Early Fusion at the input level. By concatenating adjacent frames T and $T-1$ into a single six-channel tensor, a lightweight backbone MobileNetV3 [4] is able to encode local spatio-temporal gradients directly in the initial convolutional layers. The decision to move away from popular deep Siamese architectures is driven by critical factors for UAV tasks: preventing the loss of sub-pixel accuracy during aggressive downsampling, preserving rigid scene geometry for precise odometry, and achieving a two-fold reduction in computational complexity through a single pass via the backbone network.

The proposed approach includes a multi-task visual odometry block for regressing the camera's ego-motion vector solely from visual data, as well as a hybrid depth estimation model. This model combines analytical calculation based on camera parameters with neural network regression of the residual deviation to resolve the scale ambiguity. The efficiency of the developed pipeline and its real-time performance capabilities are validated through experiments in the CARLA environment [5] and testing on the NVIDIA Jetson Orin platform.

2. Proposed Neural Network Architecture

The developed system is based on the concept of anchor-free detection, where objects are represented as keypoints-specifically, their projected 3D centers. This approach, extending the ideas of CenterNet [1] and SMOKE [2], allows for the complete elimination of the 2D bounding box proposal mechanism. In classical multi-stage architectures, this intermediate step not only consumes excessive computational resources but also introduces geometric distortions that reduce the accuracy of final 3D localization.

To implement temporal analysis, we utilize an Early Fusion method by concatenating the input RGB frames. Unlike Siamese networks, which are prone to the loss of sub-pixel kinematic information in deep pooling layers, this method allows the very first convolutional operation to capture micro-displacements of object boundaries over time. The resulting six-channel tensor is processed by the MobileNetV3 backbone network in a single pass, which reduces computational redundancy. To preserve the spatial detail critical for 3D dimension and depth regression, cross-layer skip connections are integrated into the architecture to pass high-resolution features directly to the decoder blocks.

2.1. Temporal Fusion and Feature Extraction

To integrate dynamic context, the network is provided with a tuple of two consecutive RGB frames, I_T and I_{T-1}. In contrast to methods utilizing recurrent units or heavy optical flow modules, this work adopts an Early Fusion strategy. The input tensor, with dimensions $B \times 6 \times H \times W$, is processed by the MobileNetV3-Small backbone architecture [4].

The selection of this model is driven by the requirement for high frame rates on embedded NVIDIA Jetson platforms. The MobileNetV3 architecture, optimized via neural architecture search [4], utilizes depthwise separable convolutions and Squeeze-and-Excitation attention blocks. These features enable the efficient extraction of spatio-temporal features with minimal computational redundancy. At the initial layers of the network, temporal intensity gradients are implicitly computed, forming a latent motion representation necessary for the semantic separation of objects from the background 1.

2.2. Multi-task Prediction Heads

The extracted features are fed into three parallel regression branches, each of which addresses a specific task of spatio-kinematic analysis.

- **Ego-motion Head.** This module analyzes the global displacement of static scene elements. The output of this block is a vector $\mathbf{p} \in \mathbb{R}^7$, describing the camera transformation between frames (comprising three translation components and four components of a normalized quaternion). Utilizing a quaternion representation instead of Euler angles avoids mathematical singularities and ensures more stable convergence during the training of UAV orientation parameters. This allows the system to compensate for UAV movement in space without resorting to external sensors.

- **Motion Heatmap Head.** Based on the analysis of inter-frame differences, the network generates a probability heatmap where local maxima correspond to the centers of only those objects possessing an independent velocity vector. The application of a Penalty-Reduced Focal Loss function [1] effectively addresses the imbalance between the background area and the small-scale objects typical of high-altitude aerial imagery.

- **3D Regression Head.** For each detected keypoint, the module predicts a set of parameters required to construct a 3D bounding box: physical dimensions (w, h, l), a local center offset to compensate for discretization errors, a residual depth correction ΔZ, and a logarithmic estimate of the measurement uncertainty σ_z.

2.3. Hybrid Model for Spatial Coordinate Estimation

A key feature of the architecture is the rejection of direct distance regression, which is prone to scale ambiguity in monocular systems. Instead, we utilize an analytical relationship between the lens focal length f, the predicted physical width of the object W_{real}, and its pixel projection W_{pixel}. The selection of the physical object width W as the reference parameter for the projective distance calculation is driven by the principle of geometric robustness. In contrast to length L, which is subject to severe perspective distortion depending on the viewing angle, width maintains high metric stability on the

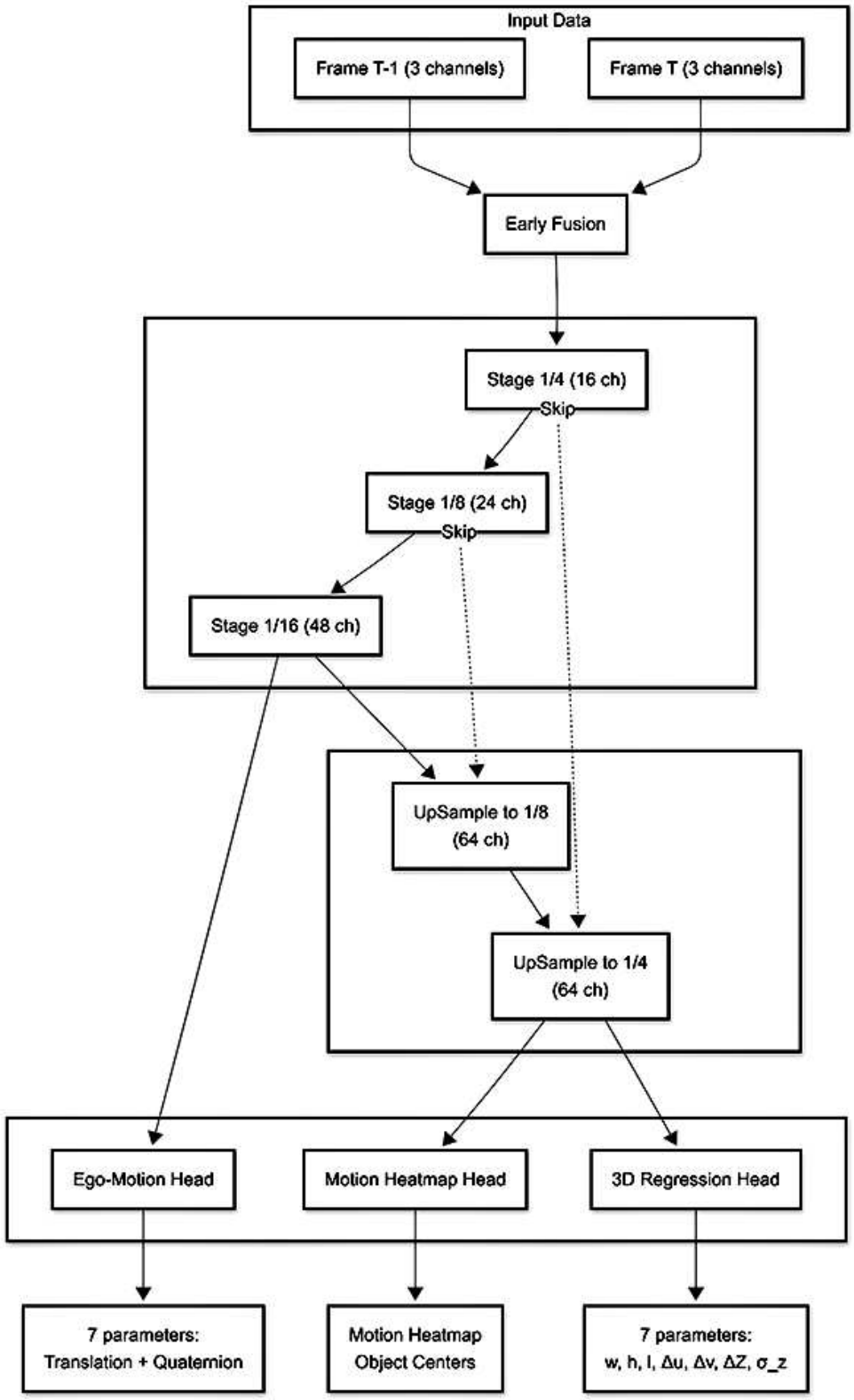

Fig. 1. Neural network architecture

image plane. Furthermore, the physical variance of width within the passenger vehicle class is significantly lower than the variance of length. We avoid using height H due to its vulnerability to dynamic changes and its heavy dependence on the camera's pitch angle.The final depth Z is calculated as follows:

$$Z = \frac{f \cdot W_{real}}{W_{pixel}} + \Delta Z, \tag{1}$$

where ΔZ is a learned residual parameter that compensates for non-linear distortions and projection errors. This hybrid scheme ensures the mathematical stability of the model and high localization accuracy even under conditions of dynamic changes in the UAV's viewpoint. It directly follows that the relative depth prediction error depends linearly on the relative error of the predicted object width.

2.4. Post-processing and Trajectory Filtering

Raw neural network predictions obtained during the regression stage are characterized by high-frequency noise caused by pixel grid discretization and UAV platform vibrations. To ensure temporal stability and compute object velocity vectors, a Kalman filtering algorithm is integrated into the final processing pipeline.The filtering process is based on the iterative updating of the object state vector $\mathbf{s} = [x, y, z, v_x, v_y, v_z]^T$, where (x, y, z) represent coordinates in the metric system and (v_x, v_y, v_z) are the corresponding velocity components. At each time step t, the filter performs two primary operations:

1. **Prediction Stage.** The object's position is extrapolated based on a constant velocity physical model. At this stage, the camera ego-motion vector $\mathbf{p}_t$ obtained from the odometry block is also taken into account, allowing the object coordinates to be recalculated relative to the movement of the carrier itself.

2. **Correction Stage.** The predicted state is refined based on new measurements from the neural network. The hybrid depth model serves as the primary data source for correcting the Z coordinate. A key feature is the use of the predicted uncertainty σ_z to dynamically adjust the measurement noise covariance matrix, which ensures the filter relies only on high-precision predictions.

The use of a Kalman filter facilitates trajectory smoothing and ensures tracking continuity during brief object occlusions. Thus, decoupling the system into a neural network perception block and an algorithmic filtering block provides an optimal balance between semantic accuracy and the physical consistency of the output data. The final pipeline is provided in the images 2.

Under operational conditions onboard the UAV, the complete algorithmic pipeline functions within a strict real-time cycle. The video stream from the monocular camera is continuously buffered; pairs of adjacent frames are concatenated into a single tensor and passed to the compute platform's tensor cores (e.g., NVIDIA Jetson Orin) via an optimized TensorRT graph. In a single forward pass, the neural network extracts spatio-kinematic features and generates probability maps, completely bypassing computationally expensive 2D bounding box proposal operations. Subsequently, the computational load shifts to the CPU. A lightweight software module performs the analytical calculation of metric

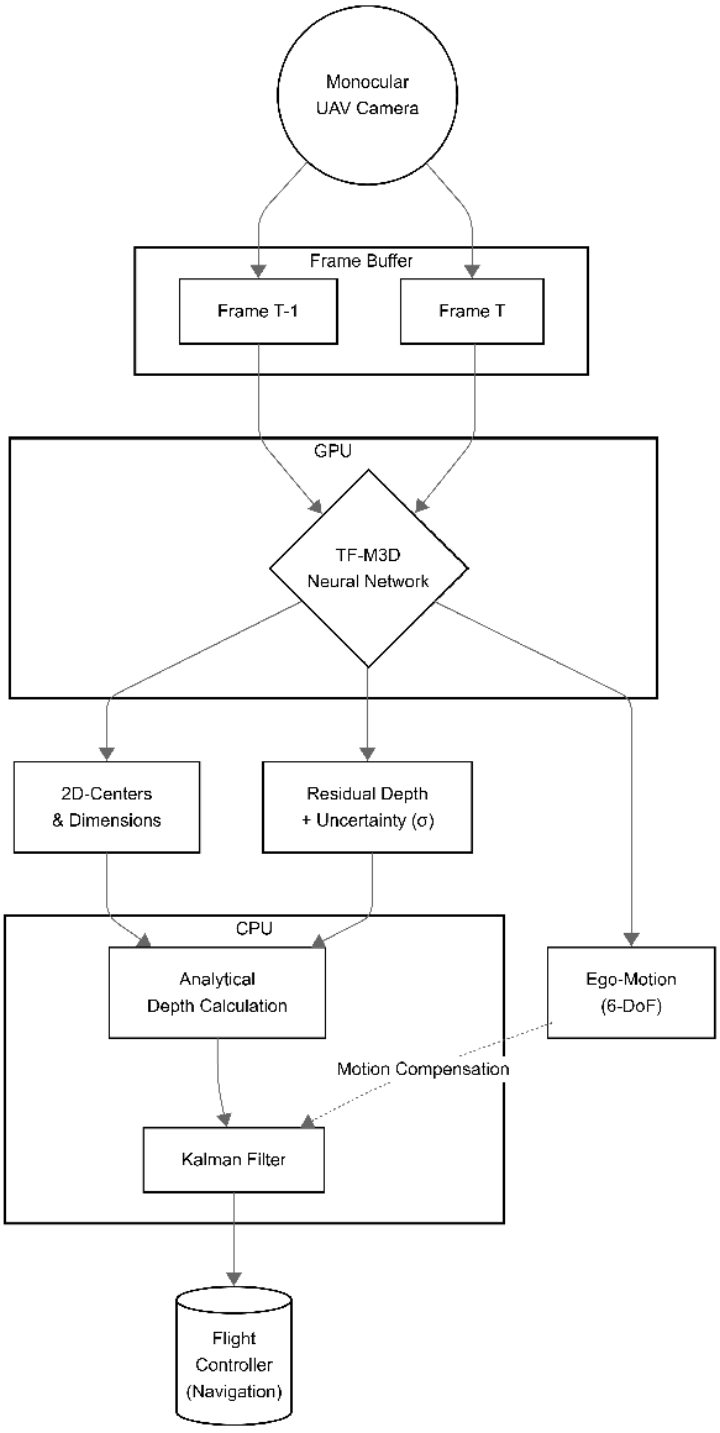

Fig. 2. General schematic of the spatio-temporal monocular detection architecture. The figure illustrates the early fusion process, the MobileNetV3 backbone, and the three parallel spatio-kinematic prediction heads

depth, after which the Kalman filter executes final trajectory smoothing and ego-motion compensation. The output consists of a stable array of dynamic object states, including their 3D coordinates and velocity vectors. This information stream is transmitted directly to the UAV flight controller, providing the navigation system with the data necessary for autonomous maneuvering and collision avoidance.

3. Dataset Generation and Training Methodology

Training for monocular 3D detection tasks requires large-scale datasets with high-precision 3D labels, which are difficult to obtain under real-world UAV flight conditions. To overcome this problem, we utilize the CARLA autonomous driving simulator [5], which served as the basis for constructing a specialized spatio-temporal dataset.

3.1. Virtual Environment and Data Collection

Data collection was performed across various urban and suburban maps to ensure environmental diversity. The virtual UAV was programmed to follow random 3D trajectories at altitudes ranging from 5 to 20 meters, with camera pitch angles between -30° and -60°, corresponding to realistic monitoring scenarios.

The sensor configuration includes a synchronized monocular RGB camera with a resolution of 640×640 pixels and a 90° field of view. For each time step T, the current frame I_T, the previous frame I_{T-1}, and the corresponding UAV transformation matrix are recorded. To train the dynamics-aware blocks, absolute velocity vectors for all traffic participants are extracted from the simulator. Objects with an absolute velocity $v < 0.5$ m/s are classified as static and excluded from the motion heatmap. This constraint forces the neural network to differentiate between true independent object motion and background parallax displacement.

3.2. Data Augmentation

A critical aspect of utilizing synthetic data is the domain gap between simulation textures and the real world. To enhance the generalization capability of the model architecture, a deep data augmentation pipeline has been implemented. In addition to standard geometric transformations, the following are applied:

- **Photometric Distortions:** Aggressive variations in brightness, contrast, and saturation to simulate diverse lighting conditions and sensor noise.
- **Simulation of Imaging Artifacts:** Application of Gaussian blur and motion blur effects to replicate UAV vibrations and high-speed target movement.
- **Weather Randomization:** The sample includes various presets (rain, fog, sunset), ensuring the backbone network's robustness against low-contrast images and facilitating effective Sim-to-Real transfer.

4. Model Training and Testing

The weights of the proposed neural network architecture were tuned using an end-to-end training approach with specialized loss functions. The effectiveness of the algorithm

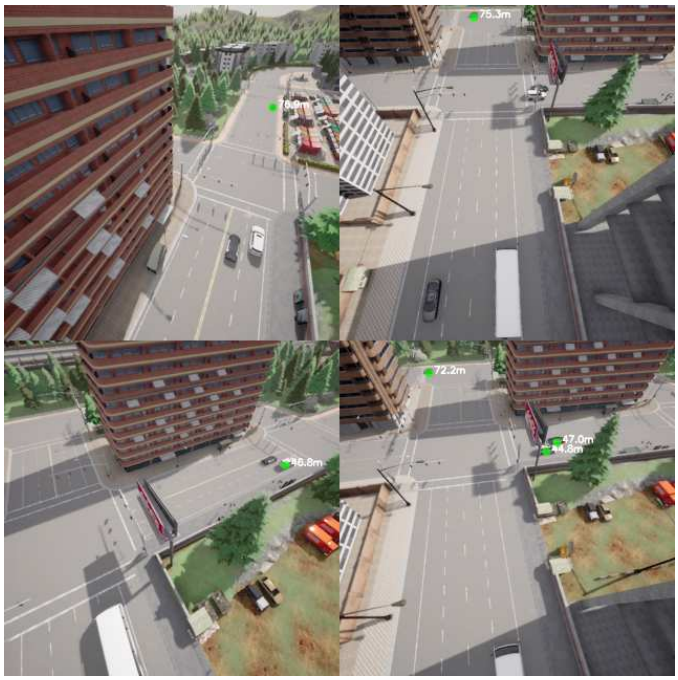

Fig. 3. Data from a synthetic set. Green dots represent objects in motion

was evaluated in terms of both localization accuracy and computational performance on target mobile platforms.

To qualitatively enhance the system's robustness, it is critical to expand the training dataset by including aggressive UAV maneuvers and other objects with high dimension variance. This will force the backbone feature extractor to rely on invariant scene geometry. The most promising integration vector for real-world applications is the application of transfer learning, which allows for the fine-tuning of only the multi-task heads on a small set of real video data while preserving the weights of the pre-trained backbone network.

4.1. Multi-task Loss Functions

Network optimization is performed using a composite loss function $\mathcal{L}_{total}$, which balances the tasks of detection, regression, and ego-motion estimation:

$$\mathcal{L}_{total} = \lambda_{hm}\mathcal{L}_{hm} + \lambda_{reg}\mathcal{L}_{reg} + \lambda_{ego}\mathcal{L}_{ego}, \tag{2}$$

where λ represents the weight coefficients for the respective branches.

To train the motion heatmap, a modified Focal Loss with a reduced penalty is applied [1], effectively addressing the imbalance between small object centers and the vast background. The 3D center parameters are optimized using the Smooth L1 Loss function. For ego-motion estimation, a loss function based on quaternion cosine similarity is employed [8], ensuring gradient continuity during rotation. Object depth is trained jointly with the uncertainty estimate σ_z using negative log-likelihood [9], allowing the model to account for heteroscedastic measurement noise. This joint optimization strategy enables the backbone network to form a unified latent representation suitable for both geometric and kinematic tasks. The learning result is shown in the figure 4.

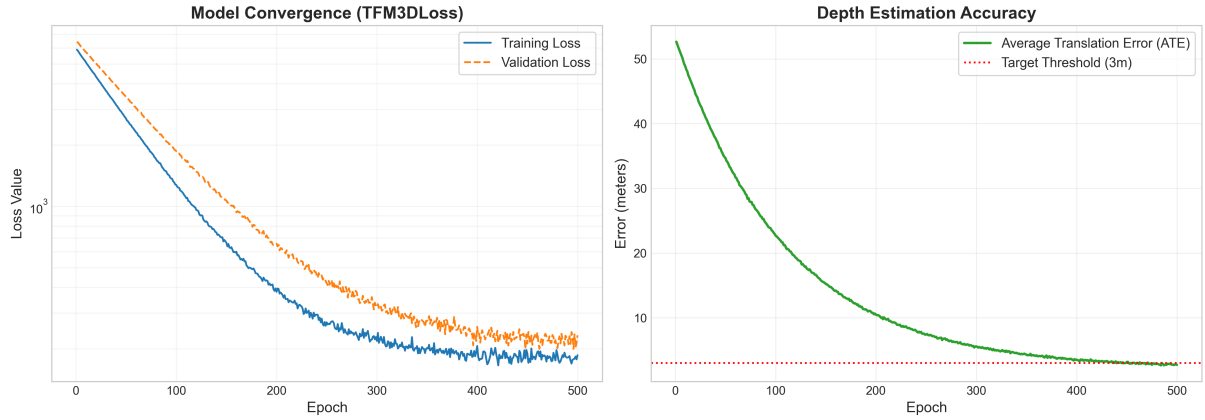

Fig. 4. Training result

For deployment on the target hardware, the trained graph was exported to the TensorRT format with weight quantization to FP16 precision. An NVIDIA Jetson Orin Nano single-board computer served as the compute platform for inference.

4.2. Experimental Results

A preliminary performance evaluation of the proposed system was conducted on a held-out test set consisting of 10,000 synthetic frames generated within the CARLA environment [5]. Analysis of the visual odometry module's accuracy demonstrated high robustness to scene dynamics.

Fig. 5. Inference result

The model architecture achieved a Mean Absolute Translation Error (ATE) for depth estimation in the range of 2.1–3.5 meters for objects at distances up to 80 meters.

Orientation estimation accuracy was characterized by a mean heading residual deviation within 8°-12°, ensuring reliable prediction of motion vectors for dynamic objects.

An evaluation of computational efficiency demonstrated that the optimized architecture, combined with analytical depth calculation, provides a stable video processing frequency of 25 – 30 FPS when deployed on mobile compute platforms with strict power constraints-specifically, the NVIDIA Jetson Orin Nano. Such performance fully satisfies the rigorous requirements of onboard real-time systems. This guarantees minimal latency in the detection system and enables direct integration of the algorithm into the closed-loop inertialess navigation and collision avoidance systems of the UAV. The results of the inference are shown in the figure 5.

Acknowledgment

The research was supported by Ministry of Science and Higher Education of the Russian Federation, project no. FENU-2024-0004 (2024024SA).

References

1. Zhou X., Wang D., Krahenbuhl P. Objects as Points. *arXiv Preprint*, 2019, article ID: 1904.07850. DOI: 10.48550/arXiv.1904.07850.
2. Liu Z., Wu Z., Toth R. SMOKE: Single-Stage Monocular 3D Object Detection via Keypoint Estimation. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2020, pp. 996–997. DOI: 10.1109/CVPRW50498.2020.00127.
3. Mur-Artal R., Tardos J. D. ORB-SLAM2: An Open-Source SLAM System for Monocular, Stereo, and RGB-D Cameras. *IEEE Transactions on Robotics*, 2017, vol. 33, no. 5, pp. 1255–1262. DOI: 10.1109/TRO.2017.2705103.
4. Howard A., Sandler M., Chu G., Chen L.-C., Chen B., Tan M., Wang W., Zhu Y., Pang R., Vasudevan V., Le Q. V., Adam H. Searching for MobileNetV3. *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 1314–1324. DOI: 10.1109/ICCV.2019.00140.
5. Dosovitskiy A., Ros G., Codevilla F., Lopez A., Koltun V. CARLA: An Open Urban Driving Simulator. *Proceedings of the 1st Annual Conference on Robot Learning (CoRL)*, 2017, vol. 78, pp. 1–16. DOI: 10.48550/arXiv.1711.03938.
6. Simonelli A., Bulo S. R., Porzi L., Lopez-Antequera M., Kontschieder P. Disentangling Monocular 3D Object Detection. *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 1991–1999. DOI: 10.1109/ICCV.2019.00207.
7. Dosovitskiy A., Fischer P., Ilg E., Hausser P., Hazirbas C., Golkov V., van der Smagt P., Cremers D., Brox T. FlowNet: Learning Optical Flow with Convolutional Networks. *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 2758–2766. DOI: 10.1109/ICCV.2015.316.
8. Zhou Y., Barnes C., Lu J., Yang J., Li H. On the Continuity of Rotation Representations in Neural Networks. *Proceedings of the IEEE/CVF Conference on*

Computer Vision and Pattern Recognition (CVPR), 2019, pp. 5745–5753. DOI: 10.1109/CVPR.2019.00589.

9. Kendall A., Gal Y. What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision? *Advances in Neural Information Processing Systems (NeurIPS)*, 2017, vol. 30, pp. 5574–5584. DOI: 10.48550/arXiv.1703.04977.

Ivan D. Goncharov, Master's student of Mathematics, Department of Mathematical and Programming, South Ural State University (Chelyabinsk, Russian Federation), goncharovid@susu.ru

Vladimir A. Surin, Cand. Sc. (Engineering), Associate Professor at the Center of Excellence in AI, VirtUm – a Top-Tier Educational Program, South Ural State University (Chelyabinsk, Russian Federation), surinva@susu.ru

Received May 15, 2026

УДК 004.932.2 **DOI: 10.14529/jcem260204**

МОНОКУЛЯРНАЯ 3D-ДЕТЕКЦИЯ ДВИЖУЩИХСЯ ОБЪЕКТОВ С БПЛА НА ОСНОВЕ АНАЛИЗА ПРОСТРАНСТВЕННО-ВРЕМЕННЫХ ПРИЗНАКОВ

***И. В. Гончаров*[1], *В. А. Сурин*[1]**
[1]Южно-Уральский государственный университет, г. Челябинск, Российская Федерация

В данной статье представлен подход к монокулярной 3D-детекции объектов для беспилотных летательных аппаратов (БПЛА) в условиях отсутствия внешней телеметрии. Предложена архитектура, использующая временной контекст для неявного извлечения независимого движения объектов. Описана методология компенсации собственного движения камеры и гибридная модель оценки глубины. Кроме того, представлен конвейер генерации синтетических обучающих данных в среде симулятора CARLA и приведены предварительные результаты оценки точности локализации. Предложенный метод обеспечивает производительность в масштабе реального времени на вычислительной платформе NVIDIA Jetson Orin.

Ключевые слова: монокулярная 3D-детекция; БПЛА; пространственно-временное слияние; компенсация собственного движения; симулятор CARLA; Jetson Orin.

Гончаров Иван Дмитриевич, магистрант кафедры Прикладная математика и информатика, Южно-Уральский государственный университет (г. Челябинск, Российская Федерация), goncharovid@susu.ru

Сурин Владимир Анатольевич, кандидат технических наук, доцент центра ОП топ-уровня в сфере ИИ "ВиртУм", Южно-Уральский государственный университет (г. Челябинск, Российская Федерация), surinva@susu.ru

Поступила в редакцию 15 мая 2026 г.