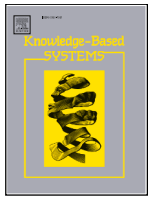
Knowledge-Based Systems 340 (2026) 115763

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys



YOLO-MOTF: Motion-temporal fusion for dynamic object detection with a moving camera for assistive wheelchairs

Shanelle Tennekoon [a,*], Nushara Wedasingha [b,c], Anuradhi Welhenge [a], Nimsiri Abhayasinghe [a], Iain Murray [a]

[a] School of Electrical Engineering, Computing & Mathematical Sciences, Curtin University, Bentley, Perth, 6102, WA, Australia
[b] Department of Electrical and Electronic Engineering, Faculty of Engineering, Sri Lanka Institute of Information Technology, Malabe, 10115, Sri Lanka
[c] Center for Excellence in Intellegent, Electronics and Telematics (CIET), Malabe, 10115, Sri Lanka

ARTICLE INFO

Keywords:
Autonomous navigation
Motion compensation
Occlusion handling
Optical flow
YOLOv8
Dynamic object detection

ABSTRACT

Dynamic object detection is fundamental to advancing vision-based navigation systems, particularly in environments where the camera itself is in motion. Despite progress in detection algorithms, existing approaches often struggle with challenges such as egomotion, short-term occlusions, temporal discontinuities, and computational cost. This paper presents YOLO-MOTF, a novel knowledge-based model that integrates spatial features with motion cues, especially for operation under moving camera conditions. The framework incorporates a hybrid motion compensation strategy to suppress camera-induced distortions and an occlusion handling buffer to preserve object trajectories through discontinuities. Additionally, a motion attention gating mechanism selectively reinforces moving object predictions by intersecting fused motion masks with semantic outputs. The proposed system achieves an F1 score of 88.6% and a 93% reduction in flow processing compared to dense flow methods, underscoring its robustness and efficiency in dynamic environments. Beyond theoretical contributions, the model demonstrates direct applicability in real-world knowledge-based decision systems, including healthcare applications such as assistive wheelchair navigation, as well as assistive robotics, autonomous navigation, and surveillance.

1. Introduction

When humans encounter unfamiliar or changing environments, they instinctively rely on visual perception to interpret their surroundings and make context-aware decisions for safe adaptation. This fundamental cognitive process has inspired the integration of object detection capabilities into digital systems such as autonomous driving [1,2], robotic navigation [3,4], assistive technologies [5,6], and surveillance [7]. In these applications, accurately identifying dynamic objects is crucial to avoid collisions, ensure safe navigation, and achieve a proper understanding of the environment, as the positions of these objects continuously change. Therefore, context awareness plays a vital role in navigation and dynamic object recognition, especially when both the camera and the objects are in motion [8]. However, accurately detecting and tracking moving objects in complex real-world environments while maintaining computational efficiency remains a challenging open problem, underscoring the need for robust models that can jointly capture object identity and motion for reliable operation.

Conventional object detection algorithms [9] are constrained by their dependence on appearance-based features and inherently lack temporal awareness [10]. Although several dynamic object detection frameworks employ dense optical flow [11], these methods are computationally demanding, restricting their applicability in real-time embedded systems such as assistive wheelchairs. More critically, motion estimation techniques often fail when both the camera and the scene are in motion, as camera-induced background motion (egomotion) can be misinterpreted as genuine object movement, resulting in a high rate of false positives for static objects. Furthermore, the absence of temporal memory renders existing models vulnerable to short-term occlusions and discontinuities, leading to fragmented object trajectories. Hence, there remains a pressing need for an integrated framework capable of efficiently analyzing both spatial semantics and motion cues under moving-camera conditions.

To address these limitations, we propose a novel architecture for dynamic object detection that effectively suppresses false positives arising from static objects. The proposed method integrates motion cues

* Corresponding author.
E-mail addresses: h.tennekoon@postgrad.curtin.edu.au, shentennekoon@gmail.com (S. Tennekoon), nushara.w@sliit.lk (N. Wedasingha), anuradhi.welhenge@curtin.edu.au (A. Welhenge), k.abhayasinghe@curtin.edu.au (N. Abhayasinghe), i.murray@curtin.edu.au (I. Murray).

https://doi.org/10.1016/j.knosys.2026.115763
Received 3 September 2025; Received in revised form 17 February 2026; Accepted 9 March 2026
Available online 11 March 2026

derived from optical flow, appearance features extracted using YOLOv8, and keypoint information obtained through Oriented FAST and Rotated BRIEF (ORB) [12]. In contrast to computationally demanding multi-object tracking (MOT) systems, our approach employs packet-based processing and shared flow computations to enhance efficiency. By incorporating motion-aware attention and memory-based tracking mechanisms, the system enables robust detection and tracking of dynamic objects in complex real-world environments. This joint motion-semantic fusion allows the model to reliably differentiate genuine object motion from camera-induced egomotion and to maintain consistent trajectories under short-term occlusions. Consequently, the proposed framework achieves the benefits of both detection and tracking systems while maintaining high computational efficiency. The core contributions of this work are summarized as follows:

- Develop a packet-based processing framework to reduce computational complexity through shared optical flow computation and selective ORB feature updates in motion-active regions.
- Construct a hybrid motion compensation (HMC) strategy that combines a feature fusion mechanism to effectively distinguish object motion from egomotion.
- Implement a motion-attention gating (MAG) mechanism to suppress static object detections by reweighting confidence scores based on optical flow consistency.
- Incorporate occlusion handling and motion prediction to handle short-term occlusions.

Collectively, these contributions transform YOLO-MOTF from a conventional appearance-based detector into a motion aware, temporally consistent framework that directly overcomes the core limitations of existing dynamic object detection methods namely, computational inefficiency, poor egomotion separation, static false positives, and vulnerability to occlusion.

The remainder of this paper is organized as follows. Section 2 reviews key developments across five domains relevant to object detection. Section 3 presents the proposed YOLO-MOTF model, detailing its mechanisms for compensating egomotion and reliably distinguishing genuine dynamic objects from background motion. Section 4 reports and discusses the experimental results on both benchmark and custom datasets, emphasizing performance improvements and practical implications. Finally, Section 5 concludes the paper by summarizing the key contributions and outlining potential directions for future research, with a particular focus on applications in healthcare contexts.

2. Related work

Dynamic object detection requires the integration of both spatial semantics and temporal motion cues, especially under moving-camera conditions. Research in this field has gradually evolved from static object recognition toward motion-aware detection and tracking. Accordingly, this section reviews key developments across four major domains relevant to this study: static object detection and backbone architectures (Section 2.1), dynamic object detection under camera motion (Section 2.2), dense optical flow and motion compensation (Section 2.3), and motion-aware multi-object tracking (MOT) frameworks (Section 2.4).

2.1. Static object detection and efficient backbones

Recognizing the importance of object recognition, early research primarily focused on detecting static objects in controlled environments. Traditional approaches [13,14] relied on hand-crafted features and statistical classifiers. Although computationally efficient, their performance degraded under conditions of occlusion, illumination variation, and background clutter. The foundation of modern visual perception was later established through convolutional neural networks (CNNs). Early two-stage detectors, such as R-CNN [15] and Faster R-CNN [16], achieved high precision. To further enhance adaptability and reduce annotation costs, recent studies have extended these architectures using cyclic self-training with proposal weight modulation for cross-supervision [17] and holistic hierarchical feature alignment for cross-domain adaptation [18]. While these methods focus on bridging domain or supervision gaps in static settings, their principles of robust feature alignment and uncertainty management are highly relevant to the YOLO-MOTF framework, which must maintain consistent feature representation despite the spatial distortions and noise introduced by dynamic camera movement. However, these models remained computationally demanding and thus unsuitable for real-time applications [10]. This limitation led to the emergence of efficient one-stage detectors, such as the YOLO family [9], which performs direct regression of bounding boxes and class probabilities in a single pass. Despite these advancements, static object detectors depend solely on appearance-based cues and therefore lack sensitivity to motion and temporal continuity [19]. Consequently, when both the camera and the objects are in motion, these models frequently misinterpret background changes as dynamic foreground motion, producing false detections. This limitation is particularly critical in navigation scenarios, where misclassifying static objects as moving entities results in false positives and reduces system reliability.

Recent studies have focused on optimizing detection backbones for resource-constrained environments to mitigate computational inefficiency. Architectures such as IRSAM [20] introduce residual aggregation modules to enhance feature reuse for small targets, while IRPrune [21] applies structured pruning to eliminate redundant convolutional channels. Similarly, ISNet [22] emphasizes multi-scale feature fusion and inter-layer connectivity to support robust contextual learning. However, these designs remain confined to static image domains and fail to address temporal variations or camera-induced motion. Building upon these foundations, our model extends the principles of spatial efficiency into the temporal domain by embedding a motion fusion layer and a hybrid compensation module within a YOLO-based framework. This design enables spatially optimized backbones to adapt dynamically to moving-camera environments, effectively transforming static object detection networks into motion-adaptive architectures.

2.2. Dynamic object detection in video sequences

Recognizing the need for motion-aware detection, recent research has increasingly focused on dynamic object detection in video sequences. Early approaches such as background subtraction, frame differencing, and motion saliency extracted motion cues from temporal variations. Traditional background subtraction methods modeled pixel distributions using Gaussian mixtures [23], later improved through non-parametric neighbor sampling [24], color-texture integration [25], and word consensus modeling [26]. While effective for static camera setups, these methods deteriorated under camera motion due to their reliance on fixed background assumptions. Deep learning approaches improved robustness to illumination changes and subtle motion [27,28], yet many remained constrained by simple camera dynamics [29,30]. More recent efforts, such as 3D Hough transform-based alignment for sidereal imagery [31], enhanced motion detection under parallax but struggled with faint objects and overlapping trajectories.

These challenges highlight the persistent need for models that can robustly disentangle object motion from camera motion in dynamic environments.

2.3. Dense optical flow and motion compensation

Addressing these challenges, advanced methods have employed dense optical flow [11] for pixel-wise motion estimation, enabling dynamic object detection under camera motion. The advent of deep learning further advanced this field, beginning with FlowNet [32], the first end-to-end CNN for optical flow estimation. While effective on synthetic

datasets, FlowNet struggled with fine-grained details and large displacements. FlowNet2.0 [33] alleviated these limitations through stacked modules and warping operations. Pyramid, Warping, and Cost Volume (PWC-Net) [34] improved computational efficiency by leveraging feature pyramids, cost volumes, and warping, thereby reducing processing time while preserving accuracy. RAFT [35] further enhanced precision using dense all-pairs correlation volumes with recurrent refinement. More recently, Gehrig et al. [36] proposed a method for continuous-time pixel trajectory regression from event-based data using Bézier curves and sequential correlation volumes. Despite these advancements, most dense-flow models remain computationally intensive due to exhaustive matching and multi-stage refinement, limiting their suitability for real-time deployment.

To mitigate this, hybrid and sparse alternatives have emerged. Combining dense flow with sparse keypoints (e.g., ORB, FAST, or Shi-Tomasi) reduces redundancy and enhances computational efficiency. However, most existing motion compensation models still treat camera motion as a separate preprocessing step rather than integrating it into the detection pipeline [37,38]. This separation reduces temporal consistency and introduces cumulative alignment errors. Consequently, the trade-off between precision and efficiency persists, underscoring the need for lightweight hybrid solutions that jointly estimate and compensate for motion within the detection process.

2.4. Motion-aware multi-object tracking (MOT) frameworks

Beyond object detection, MOT frameworks have been developed to explicitly model temporal associations between consecutive detections. Tracking-by-detection approaches such as DeepSORT [39] and ByteTrack [38] associate bounding boxes across frames based on appearance and motion similarity. ByteTrack, in particular, enhances occlusion recovery by considering low-confidence detections; however, it assumes a linear motion model and lacks compensation for camera-induced motion. Joint detection-tracking paradigms such as CenterTrack [40] and ReMOTSv2 [41] partially address motion consistency but remain sensitive to background drift. More recent trackers, including UCMC-Track [42], attempt sequence-level motion compensation through uniform transformations. Yet, this approach fails to capture localized, non-linear, or non-rigid motion, as relying solely on a global transformation matrix (e.g., homography) is insufficient for complex, non-planar, or 3D dynamic scenes. Transformer-based models [43,44] achieve strong temporal reasoning but incur substantial computational costs.

In contrast, the proposed YOLO-MOTF framework integrates motion reasoning directly within the detection stage through motion-attention gating and occlusion-aware temporal fusion. This unified design eliminates the need for a separate tracking module, enabling the model to distinguish genuine object motion from camera-induced egomotion and maintain temporal continuity under short-term occlusions. Consequently, YOLO-MOTF achieves the advantages of both detection and tracking frameworks while maintaining high computational efficiency.

3. Methodology

In this section, we describe the development of a model designed to detect dynamic objects irrespective to the motion of the camera by compensating for ego-motion and accurately identifying true dynamic objects. The overall architecture of the proposed model (Fig. 1) consists of four sub-modules:

1. **Packet Grouping:** This sub-module prepares the input video for spatio-temporal feature analysis and motion tracing by organizing its frames (f) into overlapping temporal packets (P_m), (Section 3.1). Unlike frame-by-frame dynamic detectors that redundantly compute motion fields for each frame, this design reduces computational complexity while maintaining temporal continuity, enabling real-time operation.
2. **Semantic and Spatial Feature Extraction:** This module is responsible for extracting semantic object regions and frame-to-frame motion cues by leveraging YOLOv8 segmentation [45], (Section 3.2).
3. **Temporal Feature Extraction:** This submodule captures dynamic motion cues across the frames in each packet. It integrates ORB regions of interest, optical flow vectors, and a hybrid motion compensation mechanism to eliminate egomotion (Section 3.3). In contrast to prior methods that treat motion compensation as a separate preprocessing step, this integrated approach performs feature fusion within the detection pipeline, effectively mitigating egomotion drift and enhancing detection stability in moving camera scenarios.
4. **Motion Attention Gating (MAG):** The final sub-module is designed to filter the semantic detections based on motion relevance, achieved by filtering out low-motion pixels using a gating mask (Section 3.4). Unlike conventional attention mechanisms that rely solely on spatial salience, this motion-guided gating explicitly suppresses static object detections, reducing false positives arising from appearance-only cues.

3.1. Packet grouping

Current dynamic object classification algorithms [46,47] that use frame-by-frame processing struggle to capture real-time object dynamics due to redundant computations and lack of short-term temporal coherence. To overcome these issues, this section presents a packet-based framework that reduces computational load by sharing optical flow calculations and selectively updating ORB features in motion-active regions.

The model first receives a video containing both dynamic and static objects as input. The frames (f) are then grouped into packets (P_m), based on the total length of the video. This temporal segmentation facilitates efficient spatiotemporal analysis while preserving motion continuity (Eq. 1).

$$P_m = \{f_{5m}, f_{5m+1}, \dots, f_{5m+4}\} \tag{1}$$

where, m denotes the number of packets that depends on the length of the video.

This P_m allows for inter-frame motion trajectory analysis within a localized temporal window while preserving contextual continuity. Subsequently, each P_m is simultaneously forwarded to both the Semantic-Spatial Feature Extraction module and the Temporal Feature Extraction module.

3.2. Semantic-spatial feature extraction

Traditional vision-based object detection models often struggle to accurately detect and track objects due to their limited ability to extract semantic object regions and temporal motion cues from video sequences. Recognizing the importance of incorporating both spatial semantics and motion information, this section outlines our approach to integrate the YOLOv8-seg model as a pre-trained semantic segmentation backbone as shown in Fig. 2.

First, the YOLOv8-seg model receives the input packet P_m, and uses a pretrained CNN backbone to extract multiscale feature maps, capturing low, mid, and high level semantic representations, from each frame (Eq. 2).

$$SF_{P_m} = CNN(P_m) \tag{2}$$

where, SF_{P_m} denotes the spatial features of the objects contained within a single input packet P_m.

Next, to merge the extracted spatial features SF_{P_m}, the model employs a Feature Pyramid Network (FPN), which integrates multi-scale features to enhance object representation across different resolutions (Eq. 3).

$$MF_{P_m} = FPN(SF_{P_m}) \tag{3}$$

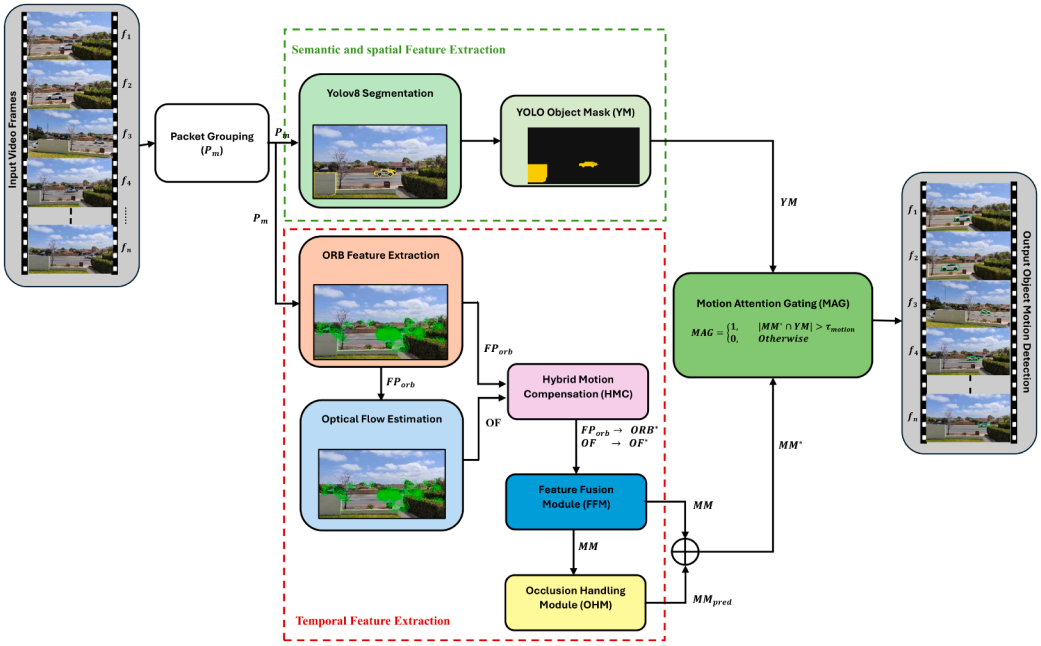


Fig. 1. Integration of temporal-motion fusion within the YOLO-MOTF framework bridging high-level semantic segmentation with global motion compensation, enabling the system to differentiate between true object dynamics and apparent background shifts caused by camera movement, by synchronizing spatial masks with temporal motion vectors, the model effectively isolates dynamic targets in complex, non-stationary environments.

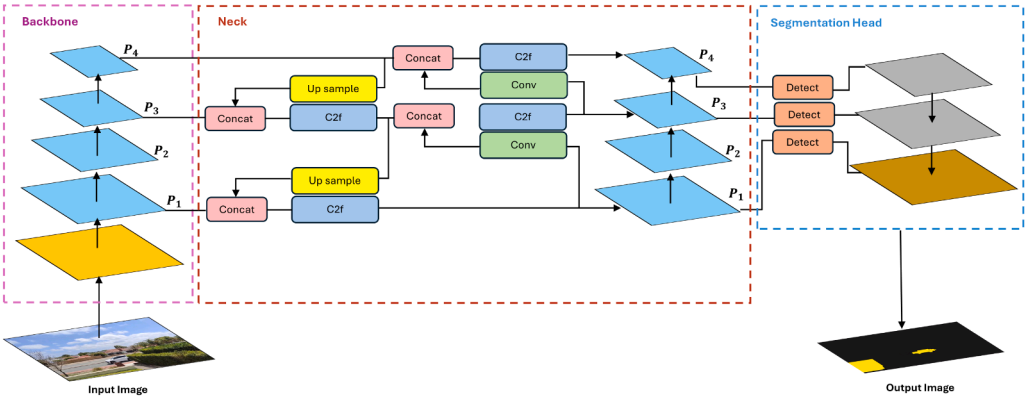


Fig. 2. Structural configuration of the YOLOv8-based segmentation backbone, that generates high-precision object-level masks that serve as the geometric foundation for motion validation. The use of multi-scale feature aggregation ensures that object boundaries remain sharp and accurate, providing the necessary spatial context for the subsequent filtering of stationary background clutter.

where, MF_{P_m} denotes the merged feature representation obtained after applying the FPN.

Then, MF_{P_m} are passed through the detection head, which consists of a one-stage dense prediction model, which predicts the segmentation mask. This delineates the spatial extent of each object and the object confidence score, which quantifies the certainty of each detection (Eq. 4).

$$(M_{P_m}, C_{P_m}) = Dense(MF_{P_m}) \tag{4}$$

where, M_{P_m} denotes the segmentation mask of each object, and C_{P_m} represents the corresponding confidence scores for dynamic object detections.

Afterwards, all pairs of (M_{P_m}, C_{P_m}) are stored in an array YM (YOLO Mask) that represents the spatial features of the detected objects across P_m (Eq. 5).

$$YM = \{(M_{P_1}, C_{P_1}), \ldots, (M_{P_m}, C_{P_m})\} \tag{5}$$

Finally, YM is forwarded to the MAG module for temporal processing.

3.3. Temporal feature extraction

Object detection algorithms that rely solely on spatial features [14, 48] often misclassify static objects as dynamic in the presence of camera motion, due to their inability to differentiate true object motion from apparent motion caused by egomotion [49]. To address this limitation, this section presents a structured pipeline for extracting temporal motion cues. The process begins with regions of interest (ROI) keypoint tracking using ORB (Section 3.3.1), followed by pixelwise motion estimation via optical flow (Section 3.3.2). Finally, a hybrid motion compensation strategy is then applied to correct for camera-induced egomotion (Section 3.3.3).

3.3.1. ORB Feature extraction

Traditional dense feature extraction methods [50] and simple corner detectors [51] often fail to accurately identify ROIs in dynamic environments due to their high computational cost and limited robustness to scale and rotation variations. To address these limitations, our

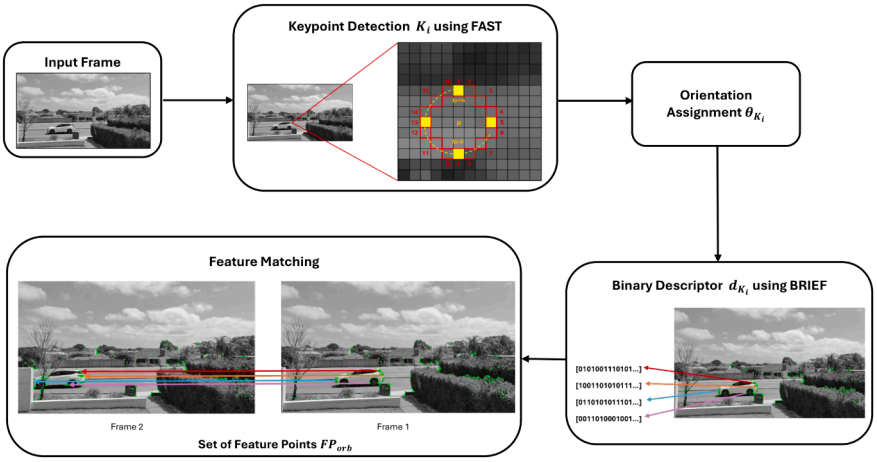


Fig. 3. Homography-based ego-motion compensation using ORB feature extraction. Rather than simple point-matching, this process establishes the global camera motion matrix (H) used to align consecutive frames. By calculating the residual displacement of keypoints relative to this compensated background, the system generates the motion-based evidence required to validate the dynamic status of detected segments.

model integrates the ORB algorithm [12], which provides a lightweight, rotation-invariant approach for detecting ROIs.

Firstly, Features from Accelerated Segment Test (FAST) [52] algorithm is applied to the frames (f) in a P_m to identify keypoints (K_i) comparing the intensity levels of pixels (Eq. 6).

$$K_i = FAST(f_{P_m}) \tag{6}$$

Eq. 6 classifies a pixel as a K_i if there exists a contiguous arc of 12 pixels within a 16 pixel Bresenham circle [53] of radius 3 (Fig. 3), where the pixel intensities are consistently brighter or darker than the central pixel. This intensity contrast indicates high local variation, which is characteristic of informative regions in f_{P_m}. To efficiently eliminate non-corner candidates, Eq. 6 evaluates only four specific pixels located at positions 1, 5, 9, and 13 on the circle highlighted in yellow in Fig. 3 before proceeding to a full arc comparison.

Next, we derive the orientation θ of K_i, since FAST is not inherently rotation-invariant (Eq. 7).

$$\theta_{K_i} = arctan\left(\frac{\sum I(x,y)(y-y_i)}{\sum I(x,y)(x-x_i)}\right) \tag{7}$$

where, $I(x, y)$ represents the intensity of a random pixel located in x and y along the horizontal and vertical axes, respectively.

Next, to compute the rotation-invariant binary descriptor (d_{K_i}), the Binary Robust Independent Elementary Features (BRIEF) algorithm [54] is applied. It performs a series of intensity comparisons between predefined pairs of pixel locations within the ROI, using the keypoint K_i and its associated orientation θ_{K_i} as inputs to generate the descriptor (8).

$$d_{K_i} = BRIEF(K_i, \theta_{K_i}) \tag{8}$$

Afterwards to track the motion of the feature points (FP_{orb}) across frames, the dot product between K_i and d_{K_i} is computed (Eq. 9).

$$FP_{orb} = K_i \cdot d_{K_i} \tag{9}$$

Finally, the computed ORB-based feature points (FP_{orb}) are simultaneously forwarded to both the Hybrid Motion Compensation (HMC) and the Optical Flow Estimation modules for further analysis.

3.3.2. Optical flow estimation

While ORB identifies salient ROI's through K_i, in each frame of P_m, it does not directly provide information about how these keypoints move over time. To understand the temporal dynamics of motion between frames, we employ optical flow estimation. Optical flow captures the apparent pixel-wise motion by analyzing changes in intensity across consecutive frames in P_m, allowing us to estimate how each keypoint and its surrounding region have shifted in space. In this study, we adopt the Lucas-Kanade method, a sparse optical flow approach that is ideal for real-time processing and robust in dynamic scenes with low texture noise.

The objective is to compute the motion vector $\vec{v}$ for each K_i in ORB (Eq. 10), such that the brightness constancy constraint (Eq. 11) is satisfied.

$$OF = [\vec{v}]_{FP_{orb}} \tag{10}$$

$$I(x, y, t) = I(x + u, y + v, t + 1) \tag{11}$$

Where, OF is the derived optical flow of feature points. Taking the Taylor series expansion of Eq. 11 and ignoring higher-order terms leads to the optical flow constraint equation (Eq. 12).

$$I_x u + I_y v + I_t = 0 \tag{12}$$

where I_x, I_y, and I_t denote the partial derivatives of the image intensity with respect to space and time.

For a small window around each frame, a system of equations is solved in the least squares sense (Eq. 13).

$$A \cdot \vec{v} = -\vec{b} \tag{13}$$

where A is the spatial gradient matrix and $\vec{b}$ is the temporal gradient vector as follows,

$$A = \begin{bmatrix} I_{x_1} & I_{y_1} \\ I_{x_2} & I_{y_2} \\ \vdots & \vdots \\ I_{x_N} & I_{y_N} \end{bmatrix} \qquad b = \begin{bmatrix} I_{t_1} \\ I_{t_2} \\ \vdots \\ I_{t_N} \end{bmatrix}$$

Solving this gives us $\vec{v}$, the motion vector for each K_i. To represent the motion information extracted, we denote the aggregated flow field over all tracked points in a packet by OF. These motion vectors OF, are later combined with ORB feature points in the feature fusion stage (Section 3.3.4).

3.3.3. Hybrid motion compensation (HMC)

In a setup where the camera is in motion, background motion induced by egomotion is often mistaken for object motion, leading to false positives [31]. Our HMC strategy addresses this by combining global homography alignment with local relative motion estimation. This hybrid approach removes background shifts induced by camera motion, while preserving the trajectories of independently moving objects. This module consists of two components:

Firstly, for two consecutive frames, a homography transformation matrix $\mathcal{H}_m$ is estimated using matched ORB feature points (Eq. 14).

$$\mathcal{H}_m = \text{findHomography}(FP_{orb}) \tag{14}$$

Once $\mathcal{H}_m$ is computed, it is used to warp the optical flow field to simulate the expected motion that would occur if the scene were entirely static (Eq. 15)

$$AF = Warp\,(OF, \mathcal{H}_m) \tag{15}$$

Where, OF represents the actual optical flow of a feature point and AF denotes the aligned (expected) flow induced solely by camera motion. The residual flow ΔRF is then computed as the absolute difference (Eq. 16).

$$\Delta RF = |OF - AF| \tag{16}$$

This residual compensates for egomotion and isolates changes caused by independently moving objects.

Although the homography approach compensates for planar background motion, it is insufficient for 3D dynamic scenes. Therefore, we introduce a relative motion analysis of tracked feature points. For each point, we calculate motion with respect to the moving camera by subtracting the estimated global camera motion, M_{cam} from the observed motion of tracked points, M_{obs} (Eq. 17)

$$RM = M_{obs} - M_{cam} \tag{17}$$

Where, RM is the relative motion, M_{obs} is the observed displacement of the feature point, and M_{cam} is the expected motion of that point due to camera motion. A non-zero relative motion vector indicates that the feature has moved independently of the camera, suggesting true object motion. These are visualized in 3D (Fig. 4), where the x-axis is the ID of the feature, the y-axis is the magnitude of motion, and the z-axis is the frame index.

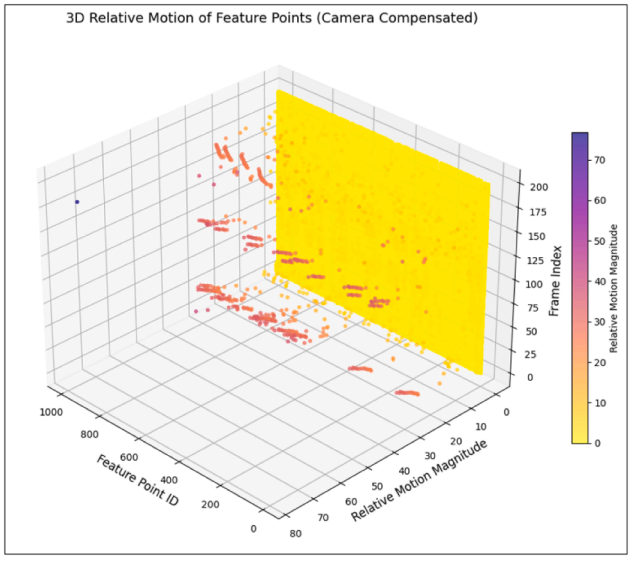


Fig. 4. 3D visualization of relative motion magnitudes of ORB feature points across time. Dense yellow points indicate static background, while outliers represent true dynamic objects.

This hybrid strategy improves robustness by compensating for egomotion using geometric warping, while also identifying dynamic elements through point-level motion inconsistency.

3.3.4. Feature fusion module (FFM)

Most prior dynamic detection methods merge motion and semantic cues using uniform or sequential fusion [55], where motion information is first estimated and then appended to visual features. This decoupled process often causes temporal misalignment and allows background motion to dominate object cues [56].

To overcome this, after extracting motion cues using optical flow and ORB features, and compensating them for camera induced motion, we perform feature fusion to generate a unified spatio-temporal representation. Given,

- Motion compensated ORB feature points: ORB^*
- Motion compensated optical flow motion vectors: OF^*

We define the fused feature map (Eq. 18),

$$MM = ORB^* + OF^* \tag{18}$$

where MM is the motion mask of objects in the fused feature map. This fusion module captures both how a point has moved and what it looks like, enabling downstream modules to more robustly distinguish between static and dynamic regions by integrating features into a single motion-aware representation.

3.3.5. Occlusion handling module (OHM)

A common limitation in dynamic object detection is the inability to maintain consistent tracking when an object undergoes temporary occlusion or becomes visually indistinguishable. Conventional segmentation models fail to detect such objects in consecutive frames, resulting in fragmented trajectories and missed detections [57,58].

To address this, we incorporate an OHM that preserves temporal continuity by leveraging motion memory. The motion masks MM are stored across the last 5 frames in a temporary occlusion buffer. If a current frame detection is missing, a predicted motion mask MM_{pred} is generated based on recent movement of the object.

The updated motion representation MM^* is derived using the union of current and predicted motion masks (Eq. 19).

$$MM^* = MM \cup MM_{pred} \tag{19}$$

This strategy allows the system to retain object awareness during short-term occlusions, ensuring robust motion continuity.

3.4. Motion attention gating (MAG)

Even after compensation, residual noise or static objects with low motion can be misclassified as dynamic objects [59]. To mitigate this, we apply a MAG module, which filters the detections to isolate regions of significant motion. This helps suppress false positives from minor visual variations and ensures that only true dynamic objects are retained.

$$MAG = \begin{cases} 1, & \text{if } |MM^* \cap YM| > \tau_{\text{motion}} \\ 0, & \text{otherwise} \end{cases} \tag{20}$$

where τ_{motion} is a predefined motion threshold. This gating ensures that only regions exhibiting significant motion are passed to the final output.

This results in a clean binary mask that highlights only the independently moving objects in the scene. By integrating object awareness with motion selectivity, MAG enables the model to operate robustly in visually complex, dynamic environments.

4. Results and discussion

This section presents the experimental evaluation of the proposed dynamic object detection framework using a camera in motion. We evaluated the model in terms of detection accuracy, temporal consistency, and computational efficiency, with comparative analyses against state-of-the-art methods and an ablation study. Each result directly supports one of the four core contributions described in Section 1.

All experiments carried out for the proposed model were based on five main datasets. These include three publicly available datasets: The Multi-Object Tracking and Segmentation (MOTS) dataset [60], the KITTI dataset [61], the DAVIS dataset [62], the VisDrone dataset [63], and a custom dataset created specifically for the testing of this research.

4.1. Dataset description

Table 1 presents a detailed description of the five datasets. These datasets were selected to reflect challenging real-world scenarios that include low-light environments, dynamic backgrounds, partial occlusions, and fast moving objects in egomotion.

Table 1
Summary of datasets used in evaluation, including the total number of frames, resolution, and their applications.

Dataset	Total Frames	Resolution	Application	Description
MOTS	5724	1080x1920 480x640	Tracking, segmentation	Urban traffic dense crowds
KITTI	4328	1280x384	Driving scenes, egomotion	Outdoor sequences moving camera
DAVIS	2764	1080x1920 480x640	Video object segmentation	Fast motion complex scenes
VisDrone	8957	1080x1920 480x640	Drone captured scenes	Crowded urban environments
Custom	1680	1080x1920 480x640 720x1280	Real-world scenes	Poor lighting corridor turns occlusions
Total	**23,453**			

The diversity of these datasets facilitated a thorough evaluation of the robustness of the proposed model in dynamic and complex environments.

4.2. Evaluating efficient packet-based processing

To overcome the computational inefficiency of frame-by-frame processing, we introduced a packet-based grouping method. To assess the effectiveness of this module, we compared the average per-frame flow computation time of YOLO-MOTF with that of the dense Farneback optical flow baseline.

Fig. 5 illustrates the average flow computation time per frame across the five datasets for both the dense optical flow (Farneback) and the proposed packet-based YOLO-MOTF model. As shown, the proposed model consistently achieves substantially lower computation times, reducing the latency per frame from approximately 0.40 - 0.55 seconds (of dense flow) to just 0.02 - 0.04 seconds.

This represents a reduction of 93-95% in flow processing time across all datasets. This improvement is attributed to two key factors: (1) packet-based grouping, which enables shared computation across frames, and (2) the use of sparse Lucas-Kanade flow guided by ORB keypoints, instead of dense pixel-wise estimation. These findings demonstrate that the proposed approach effectively addresses the limitation of high computational overhead in dynamic object detection systems, making it suitable for real-time applications.

To support the real-time performance objectives of the model, we evaluated several keypoint-based feature extractors with respect to their processing time and the number of detected keypoints. Fig. 6 summarizes this comparison across five extractors: ORB [12], Scale-Invariant Feature Transform (SIFT) [64], FAST [52], Binary Robust Invariant Scalable Keypoints (BRISK) [65], and Accelerated KAZE (AKAZE) [66].

As shown in the left plot, ORB consistently achieves the lowest processing time per frame with minimal variance, significantly outperforming SIFT, BRISK, and AKAZE in terms of speed. Although FAST offers competitive speed, it lacks a descriptor component, making it unsuitable for motion compensation and tracking. The right plot shows the number of keypoints extracted, where ORB provides a stable and sufficient number for motion estimation, balancing rich features with computational efficiency.

These results validate the selection of ORB for our feature extraction pipeline, offering an optimal trade-off between speed and feature richness which is critical for maintaining lightweight, real-time performance in navigation tasks.

4.3. Distinguishing object motion from camera motion using HMC

One of the primary challenges in dynamic object detection using a camera in motion is differentiating the actual motion of an object from the background changes induced by camera movement. To address this limitation, the proposed YOLO-MOTF model incorporates a HMC strategy. This hybrid approach enables the system to suppress egomotion induced background movement while preserving independently moving objects for accurate motion detection. The model was compared with several models under the hyperparameter settings in Table 2.

As shown in Table 3, YOLO-MOTF achieved a notable balance between detection accuracy and computational efficiency. Within the

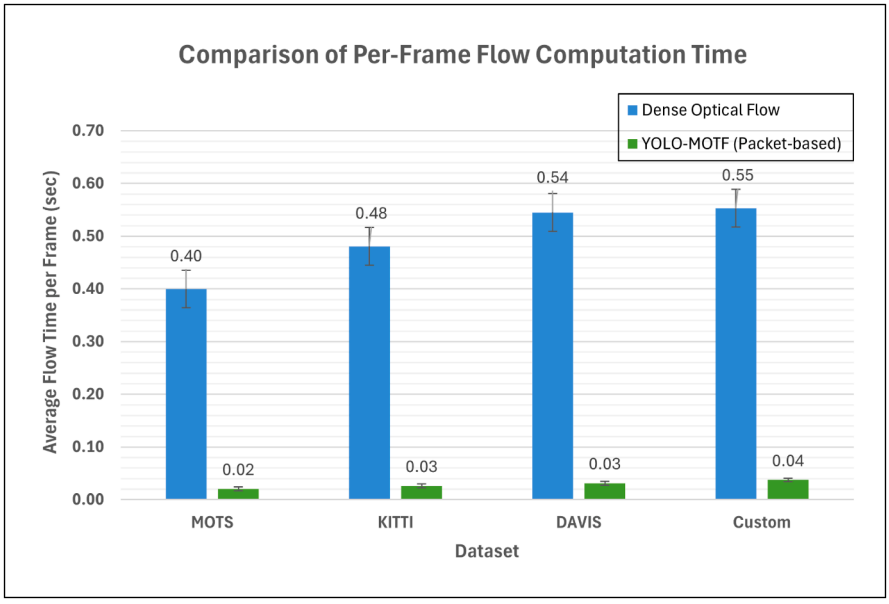


Fig. 5. Comparison of per-frame flow computation time between YOLO-MOTF and dense optical flow.

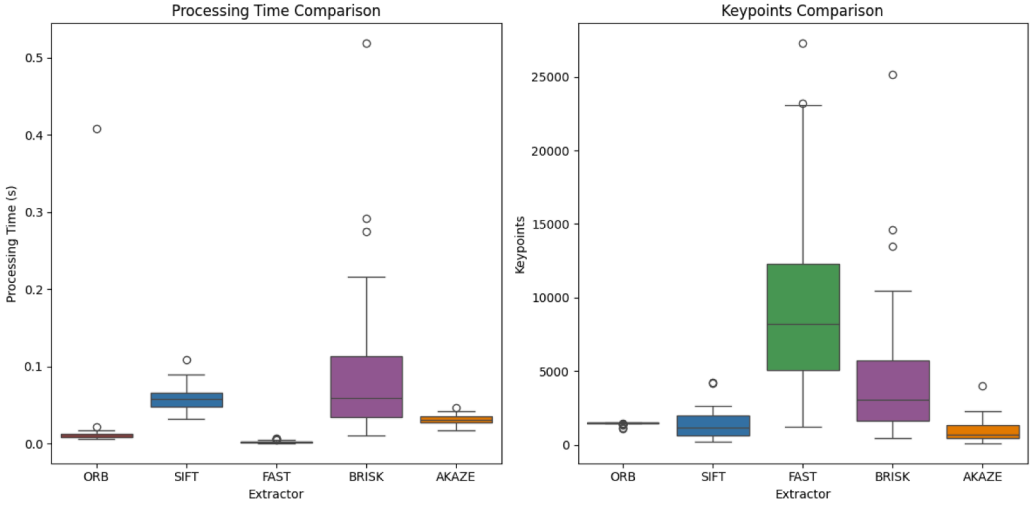


Fig. 6. Performance comparison of feature extractors where ORB demonstrates the best balance between computational efficiency and feature stability.

Table 2
Hyperparameter settings.

Parameters	Value
Epochs	150
Ir0	0.002
Irf	0.002
Momentum	0.9
Batchsize	16
Cache	False
Input image size	640 × 640
Optimizer	AdamW

Table 3
Performance comparison of models used for both appearance and motion, and motion only. The motion only methods have been tested on the MOT20 dataset and some detection results obtained from Yi et.al., [42].

Model	Precision (%)	F1 score (%)	Model size (MB)	Params (M)
appearance and motion:				
YOLOv8-seg [48]	63.5	73.56	6.45	3.66
YOLOv7-seg	62.9	71.28	78.1	37.98
YOLOv5 [48]	59.08	70.76	6.43	7.74
Faster R-CNN [16]	52.59	61.44	159.7	41.75
TrackRCNN [60]	94	57.3	605.3	44.6
YOLO12n-seg [67]	66.8	76.2	5.8	2.6
AMF-MOT [68]	91.4	84.2	128	32
RT-DETRv2-s [69]	82.5	81.5	31	5.2
motion only:				
ReMOTSv2* [41]	69.7	70.4	7.2	6.5
ByteTrack* [38]	72.16	82.1	6.54	3.24
C-BIoU* [70]	87.9	80.7	3.92	4.76
MotionTrack* [71]	78.2	80.1	6.1	3.45
SparseTrack* [72]	54.9	56.7	6.4	3.7
YOLO-MOTF (Ours)	**95.6**	**88.62**	**5.6**	**2.98**

*parameters and model sizes for motion only methods are reported based on YOLOv8n detection backbone.

appearance and motion category, YOLO-MOTF achieved an F1 score of 88.62% and precision of 95.6%, outperforming state-of-the-art detectors. This improvement demonstrates the benefit of integrating HMC and temporal feature fusion, which allows the model to maintain robust performance under camera egomotion. Despite these gains, the model remains compact with only 5.6MB and 2.98M parameters, making it more practical for edge and embedded deployment than other models.

In the motion only category, YOLO-MOTF also surpasses recent trackers such as ReMOTSv2, ByteTrack, and MotionTrack, achieving an F1 score of 88.62% at a real-time speed of 37 FPS. Competing algorithms have reported lower frame rates ($\lesssim$ 31 FPS) or reduced segmentation quality, reflecting trade-offs between temporal precision and runtime. This performance gain can be attributed to its effective temporal integration, a strategy that aligns with recent advancements in broader video analysis tasks. For instance, in the field of language-guided video segmentation, [73] demonstrates that aggregating local and global temporal context is essential for resolving ambiguities in dynamic scenes. While this work focuses on aligning linguistic features with video context, our results suggest that a similar fusion philosophy is highly effective for geometric ego-motion compensation. The ability of YOLO-MOTF to maintain high accuracy while sustaining real-time throughput emphasizes its suitability for autonomous navigation and other resource limited intelligent vision systems.

In addition to the average precision and F1 scores, the distribution of model performance across frames is visualized in Fig. 7. The box plot highlights both the higher median F1 score and the lower variance of the proposed model compared to other baselines. This reflects more consistent detection performance across diverse frames and scenarios, further validating the effectiveness of the motion compensation module.

Nevertheless, a small number of outliers are observed for YOLO-MOTF (See Fig. 7). These correspond to frames with abrupt egomotion, non-rigid background motion, or long-term occlusions, where motion estimation becomes unreliable and the temporal fusion buffer momentarily fails to recover consistent trajectories. Such cases expose the model's dependence on optical-flow accuracy under highly dynamic scenes. Future research will focus on adaptive temporal filtering and depth-aware flow regularization to further reduce the remaining sensitivity to severe motion and occlusion artifacts.

These results collectively demonstrate that the proposed HMC approach significantly enhances the model's ability to suppress false positives due to egomotion, leading to more reliable detection of dynamic objects. This confirms the effectiveness of the motion compensation strategy in improving both the accuracy and the temporal stability of the system's detection output.

4.4. Suppressing residual false positives with motion-attention gating

Even after accounting for camera motion, static or low-motion regions in the scene may still be misclassified as dynamic objects, leading to persistent false positives. This becomes particularly challenging when distinguishing between static and dynamic objects becomes ambiguous. To address this limitation, YOLO-MOTF incorporates a MAG mechanism that filters out detections based on optical flow magnitude and

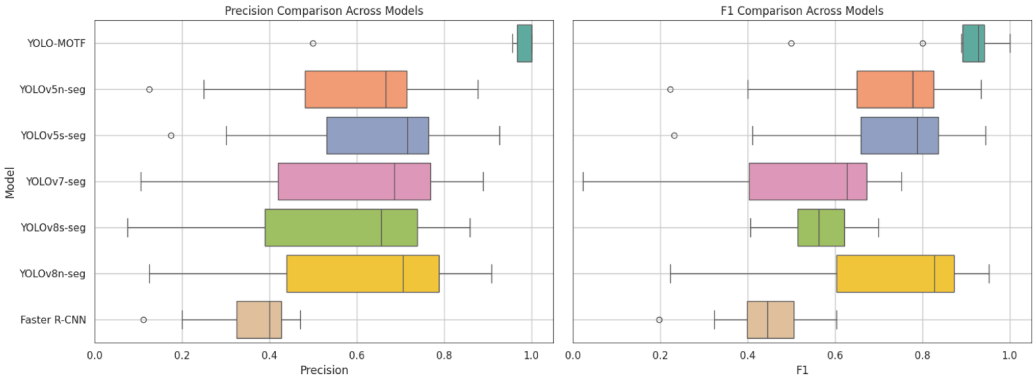


Fig. 7. Box plot comparison of Precision and F1 scores across different models. The proposed YOLO-MOTF model demonstrates higher median F1 and lower variability compared to state-of-the-art baselines, indicating improved consistency and reliability in dynamic object detection.

temporal consistency. This method ensures that only truly dynamic regions are retained for downstream decision making.

Fig. 8 presents frame-by-frame intersection over union (IoU) comparisons between YOLO-MOTF and YOLOv8 across four datasets: (8a) MOTS, (8b) KITTI, (8c) DAVIS, and (8d) a custom navigation dataset. In all four scenarios, YOLO-MOTF consistently maintains higher IoU values, reflecting improved segmentation accuracy and mask alignment with ground truth. The proposed YOLO-MOTF model demonstrates particular robustness in scenes with dense occlusions and subtle motion, where motion gating has effectively suppressed noise from static regions.

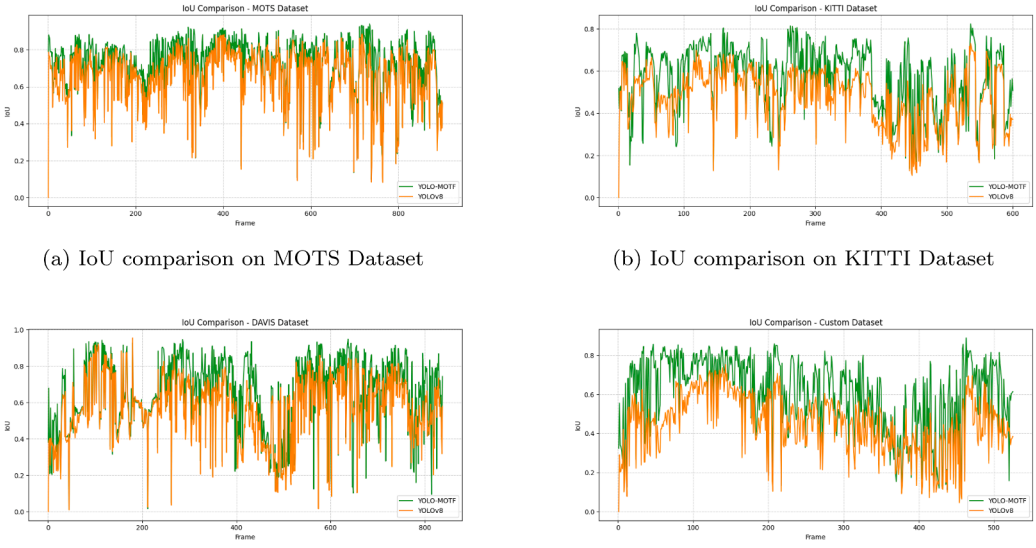
To complement the quantitative analysis, Fig. 9 illustrates qualitative comparisons of the precision of the segmentation in several diverse frames. In each example, red outlines denote ground truth masks, green outlines correspond to YOLO-MOTF predictions, and the blue outlines demonstrate the YOLOv8 detections. As seen, YOLOv8 produces broader and imprecise masks, particularly in static or partially occluded regions. In contrast, YOLO-MOTF consistently provides cleaner and tighter segmentation masks that closely align with the ground truth. This qualitative analysis reinforces the effectiveness of the MAG module in enhancing object-level precision while reducing background interference.

Together, these results confirm that the MAG module plays a critical role in reducing false positives and improving the reliability of dynamic object detection, particularly in real-world navigation environments where segmentation precision directly impacts decision making.

4.5. Motion prediction and occlusion handling

In real-world navigation scenarios, especially in crowded environments, dynamic objects are frequently subjected to short-term occlusions. These occlusions disrupt the continuity of detection, leading to missed detections that compromise tracking reliability [74,75]. To address this, our model comprises a motion prediction mechanism coupled with an occlusion handling module. This allows the system to maintain the estimated trajectory of moving objects even when they are temporarily obstructed from view.

Fig. 10 demonstrates the effectiveness of this approach using a sequence of three consecutive frames from a crowded pedestrian scene. The green contours represent the current detections, while the orange contours indicate predicted locations of objects temporarily lost due to occlusion. The cyan lines trace the motion trajectories of the tracked objects across the frames.



(a) IoU comparison on MOTS Dataset

(b) IoU comparison on KITTI Dataset

(c) IoU comparison on DAVIS Dataset

(d) IoU comparison on Custom Dataset

Fig. 8. Frame-wise IoU comparison across the four datasets where YOLO-MOTF consistently outperforms YOLOv8 in segmentation alignment with ground truth, in cluttered and dynamic scenes.

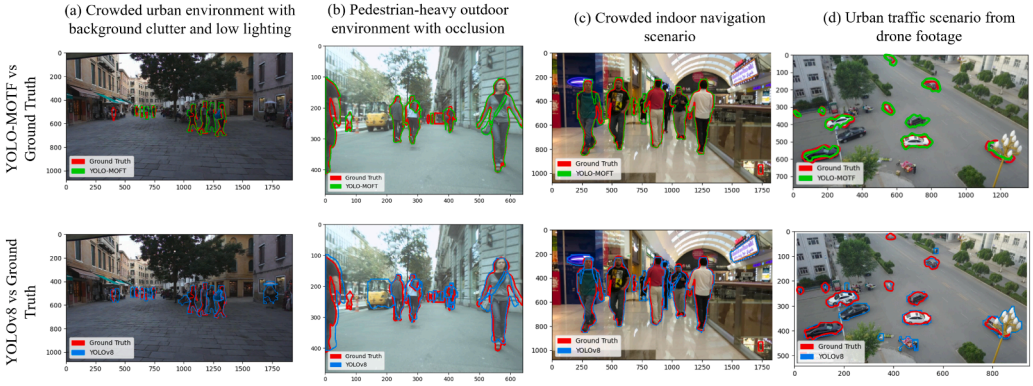


Fig. 9. Qualitative results across 4 sample frames in diverse environments illustrating motion prediction and occlusion handling.

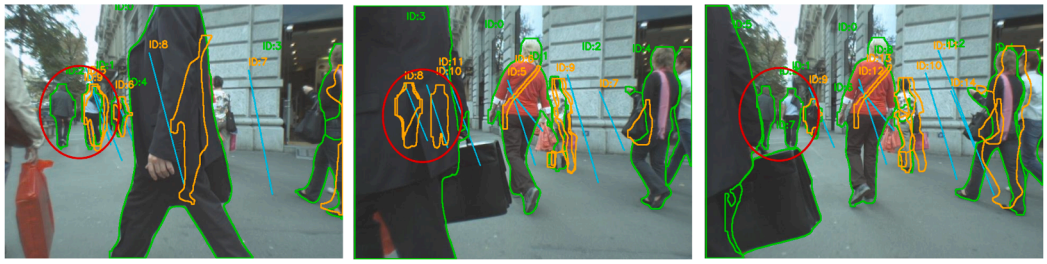


Fig. 10. Qualitative results across 3 sample frames (MOTS Dataset) illustrating motion prediction and occlusion handling.

The red circled region in each frame highlights a case of occlusion. In frame 1, the two pedestrians are clearly visible. In frame 2, the same pedestrians become occluded by another pedestrian and is no longer detected directly. However, their location is accurately predicted (orange contour), preserving its identity. By frame 3, the pedestrians reappear and is once again detected, allowing seamless re-association with its original track. This example illustrates how the integration of motion-aware prediction and temporal memory has enabled robust tracking continuity through occlusions.

These results highlight the ability of the model to handle challenging occlusions while preserving trajectory consistency. This is critical for navigation systems, where false negatives or detection loss can lead to unsafe path planning. This OHM not only improves detection continuity but also improves downstream decision making by ensuring stable object awareness over time.

4.6. Ablation study

To systematically validate the contribution of each component in the proposed YOLO-MOTF framework, an ablation experiment was conducted using sequentially added modules: Packet grouping, ORB + HMC, MAG, and OHM. Fig. 11 illustrates the qualitative effect of each module, and Table 4 presents the quantitative evaluation in terms of precision, F1 score, and average flow time.

As shown in Table 4, each component contributed to both accuracy and efficiency. Packet grouping reduced computational overhead by sharing optical-flow operations within temporal windows, achieving a 37% improvement in runtime. Incorporating HMC yielded the largest accuracy gain, confirming its effectiveness in mitigating camera-induced background motion. MAG further refined segmentation by filtering low-motion regions, resulting in a 15-point increase in the F1 score compared to the baseline YOLOv8. Finally, the OHM maintained detection stability during short-term occlusions without additional latency.

The visual comparisons in Fig. 11 demonstrate a progressive improvement in detection stability and precision as each module is introduced. The baseline YOLOv8-seg model performs well in static scenes, but fails to isolate dynamic objects in motion-rich environments. Adding packet grouping reduces redundant optical flow computations by processing frame packets, improving runtime efficiency (as seen in Table 4). The inclusion of HMC, which compensates for camera ego-motion, significantly improves precision by filtering out background movements and aligning motion vectors to the scene geometry. The MAG further enhances mask quality by suppressing low-motion regions and maintaining temporal consistency, as is evident in the cleaner silhouettes. Finally, the OHM effectively maintains object continuity across short occlusions, producing stable detection trajectories without adding computational latency. This stage shows well-bounded silhouettes for pedestrians and vehicles even under motion blur or partial visibility.

Overall, the ablation study confirms that each component contributes distinctively to both detection accuracy and computational efficiency. The proposed modules achieve a 32.1% increase in precision

Table 4
Ablation experiment on each improved module.

Configuration	Precision (%) ↑	F1 Score (%) ↑	Average Flow Time (s/frame) ↓	Remarks
YOLOv8-seg (baseline)	63.5	73.6	0.5	No motion or temporal context.
+ Packet grouping	70.8	77.4	0.25	Reduced redundancy via shared flow computation.
+ HMC	82.5	84.2	0.23	Suppressed egomotion-induced distortions.
+ MAG	94.3	88.2	0.21	Eliminated static false positives using motion consistency.
+ OHM	95.6	88.6	0.21	Preserved trajectories through short-term occlusion.

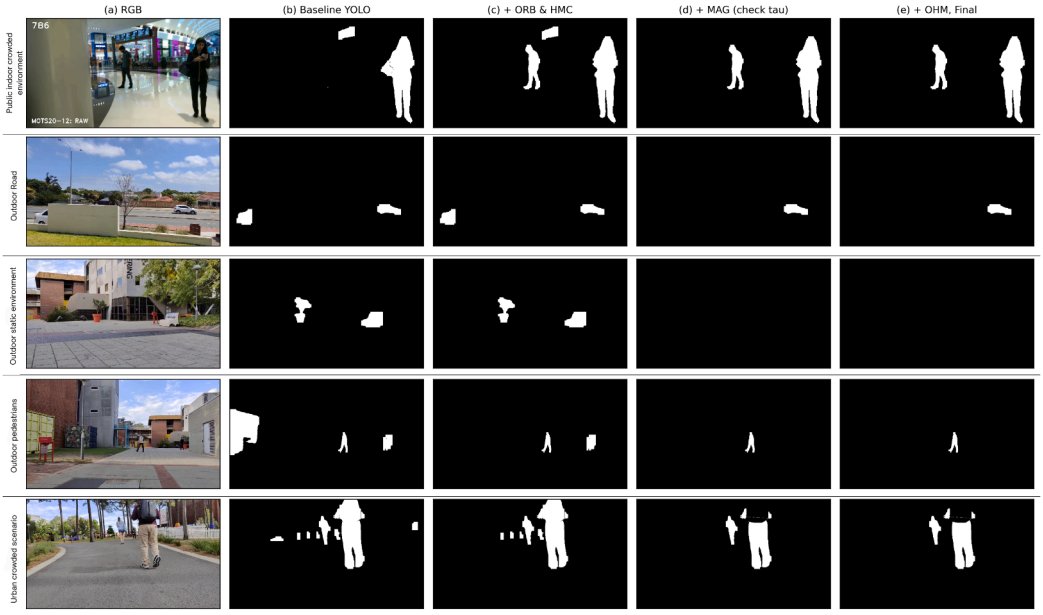


Fig. 11. Qualitative ablation results showing the effect of each module on dynamic object detection in diverse environmental conditions. Columns represent (a) RGB input, (b) baseline YOLOv8 segmentation, (c) + ORB & HMC, (d) + MAG, and (e) + OHM, Final. White regions denote detected moving objects.

and a 15% gain in F1 score relative to the YOLOv8-seg baseline model, while reducing the average flow computation time by more than 50%. These results verify the synergy between spatial detection and temporal motion compensation, validating the design of YOLO-MOTF as a lightweight and temporally aware model suitable for real-time dynamic object detection.

5. Conclusion

In this research, we propose a novel model designed to detect dynamic objects in environments with high motion using a lightweight, temporally guided framework. By combining optical flow, ORB-based feature extraction, and motion-guided feature fusion, our model effectively identifies dynamic objects from a mobile camera while reducing static false positives. Importantly, it introduces MAG and HMC strategies to distinguish actual object movement from background motion caused by egomotion, one of the key challenges in dynamic visual perception. The novelty of YOLO-MOTF lies in its integrated motion-temporal fusion architecture, which embeds motion reasoning directly into the detection pipeline addressing temporal misalignment in uniform and sequential fusion models.

Across benchmark datasets such as MOTS, KITTI, DAVIS, VisDrone, and a custom navigation set, our model consistently outperforms state-of-the-art detectors and MOTS models in terms of segmentation accuracy, F1 score, and robustness in cluttered scenes. In particular, the inclusion of an occlusion handling buffer allows our system to preserve object detections even through brief occlusions, achieving smoother and more reliable tracking in real-world navigation scenarios. Compared to dense flow-based systems, our packet-based method yields a 93-95% reduction in computation time per frame, making it ideal for real-time deployment in assistive technologies. Our comprehensive ablation study confirmed the distinct value of each component in the pipeline, resulting in a 32.1% increase in precision and a 15% gain in F1 score over the baseline model.

Beyond technical performance, the proposed model offers significant practical value. Its computational efficiency and robustness to egomotion make it well suited for integration into resource-constrained platforms such as assistive wheelchairs, mobile robots, and low-power embedded systems. By enabling accurate and continuous detection of moving obstacles in dynamic scenes, our approach contributes to safer and more intelligent navigation. Most importantly, its applicability in healthcare settings, such as supporting visually impaired users in wheelchair navigation and rehabilitation, demonstrates how knowledge-based dynamic perception can directly enhance autonomy, safety, and decision making in real-world intelligent mobility systems.

6. Limitations and future directions

Despite the performance gains demonstrated by YOLO-MOTF, its architectural dependencies introduce specific operational constraints. Firstly, the reliance on sparse feature matching (ORB) for homography estimation assumes a 'rigid world' with sufficient texture. In scenarios involving homogeneous surfaces (e.g., snowy landscapes, featureless asphalt, or clear sea surfaces), the sparsity of valid keypoints can lead to unstable ego-motion compensation, potentially causing the temporal filter to misidentify background segments as dynamic targets. Secondly, while the 2.98M parameter footprint is optimized for edge deployment, the current implementation handles long-term occlusions through linear motion extrapolation. This means, if an object is occluded for a significant duration during a sharp camera maneuver, the non-linear displacement may cause the system to lose the object's identity. To mitigate these issues, we plan to investigate hybrid motion estimation that combines traditional keypoint geometry with deep optical flow to handle regions with no texture. Furthermore, future work will focus on a transformer based temporal aggregator that can maintain long range dependencies, ensuring tracking stability even during prolonged occlusions and extreme motion blur. Finally, we aim to evaluate the framework's adaptability to resource constrained edge deployment. Given the model's optimized parameter footprint, future research will focus on hardware level acceleration and the implementation of TensorRT or OpenVINO optimizations. This will facilitate the integration of YOLO-MOTF into real-time embedded systems, such as UAV flight controllers and mobile robotics platforms, where high accuracy motion awareness must be achieved within a limited power budget.

Funding Sources

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

CRediT authorship contribution statement

Shanelle Tennekoon: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Resources, Methodology, Investigation, Formal analysis, Data curation, Conceptualization; **Nushara Wedasingha:** Writing – review & editing, Supervision, Conceptualization; **Anuradhi Welhenge:** Writing – review & editing, Supervision, Conceptualization; **Nimsiri Abhayasinghe:** Writing – review & editing, Supervision, Conceptualization; **Iain Murray:** Writing – review & editing, Supervision, Conceptualization.

Data availability

All data used have been cited in the paper and is publicly available. The authors do not have permission to share the custom data used.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] L. Liang, H. Ma, L. Zhao, X. Xie, C. Hua, M. Zhang, Y. Zhang, Vehicle detection algorithms for autonomous driving: a review, Sensors 24 (10) (2024) 3088.
[2] M.M. Kabir, J.R. Jim, Z. Istenes, Terrain detection and segmentation for autonomous vehicle navigation: a state-of-the-art systematic review, Inf. Fusion 113 (2025) 102644.
[3] L. Chen, G. Li, W. Xie, J. Tan, Y. Li, J. Pu, L. Chen, D. Gan, W. Shi, A survey of computer vision detection, visual SLAM algorithms, and their applications in energy-Efficient autonomous systems, Energies (19961073) 17 (20) (2024).
[4] H. Le, S. Saeedvand, C.-C. Hsu, A comprehensive review of mobile robot navigation using deep reinforcement learning algorithms in crowded environments, J. Intell. Rob. Syst. 110 (4) (2024) 1–22.
[5] V. Gallo, I. Shallari, M. Carratù, V. Laino, C. Liguori, Design and characterization of a powered wheelchair autonomous guidance system, Sensors 24 (5) (2024) 1581.
[6] I. Patel, M. Kulkarni, N. Mehendale, Review of sensor-driven assistive device technologies for enhancing navigation for the visually impaired, Multimed Tools Appl 83 (17) (2024) 52171–52195.
[7] O. Elharrouss, N. Almaadeed, S. Al-Maadeed, A review of video surveillance systems, J. Vis. Commun Image Represent. 77 (2021) 103116.
[8] M. Jamali, P. Davidsson, R. Khoshkangini, M.G. Ljungqvist, R.-C. Mihailescu, Context in object detection: a systematic literature review, Artif. Intell. Rev. 58 (6) (2025) 1–89.
[9] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You only look once: unified, real-time object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 779–788.
[10] Z. Zou, K. Chen, Z. Shi, Y. Guo, J. Ye, Object detection in 20 years: a survey, Proc. IEEE 111 (3) (2023) 257–276.
[11] L. Alvarez, J. Weickert, J. Sánchez, Reliable estimation of dense optical flow fields with large displacements, Int. J. Comput. Vis. 39 (2000) 41–56.
[12] E. Rublee, V. Rabaud, K. Konolige, G. Bradski, ORB: An efficient alternative to SIFT or SURF, in: 2011 International Conference on Computer Vision, 2011, pp. 2564–2571. https://doi.org/10.1109/ICCV.2011.6126544
[13] P. Viola, M. Jones, Rapid object detection using a boosted cascade of simple features, in: Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001, 1, Ieee, 2001, pp. I.
[14] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), 1, Ieee, 2005, pp. 886–893.
[15] R. Girshick, J. Donahue, T. Darrell, J. Malik, Rich feature hierarchies for accurate object detection and semantic segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 580–587.
[16] S. Ren, K. He, R. Girshick, J. Sun, Faster r-cnn: towards real-time object detection with region proposal networks, Adv. Neural Inf. Process Syst. 28 (2015).
[17] Y. Xu, C. Zhou, X. Yu, Y. Yang, Cyclic self-training with proposal weight modulation for cross-supervised object detection, IEEE Trans. Image Process. 32 (2023) 1992–2002.
[18] Y. Xu, Y. Sun, Z. Yang, J. Miao, Y. Yang, H2fa R-cnn: holistic and hierarchical feature alignment for cross-domain weakly supervised object detection, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 14329–14339.
[19] S. Tennekoon, N. Wedasingha, A. Welhenge, N. Abhayasinghe, I. Murray, Advancing object detection: a narrative review of evolving techniques and their navigation applications, IEEE Access (2025).
[20] M. Zhang, Y. Wang, J. Guo, Y. Li, X. Gao, J. Zhang, IRSAM: Advancing segment anything model for infrared small target detection, in: European Conference on Computer Vision, Springer, 2024, pp. 233–249.
[21] M. Zhang, H. Yang, J. Guo, Y. Li, X. Gao, J. Zhang, IRPruneDet: efficient infrared small target detection via wavelet structure-regularized soft channel pruning, in: Proceedings of the AAAI Conference on Artificial Intelligence, 38, 2024, pp. 7224–7232.
[22] M. Zhang, R. Zhang, Y. Yang, H. Bai, J. Zhang, J. Guo, ISNet: Shape matters for infrared small target detection, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 877–886.
[23] C. Stauffer, W.E.L. Grimson, Adaptive background mixture models for real-time tracking, in: Proceedings. 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No PR00149), 2, IEEE, 1999, pp. 246–252.
[24] O. Barnich, M. Van Droogenbroeck, Vibe: a powerful random technique to estimate the background in video sequences, in: 2009 IEEE International Conference on Acoustics, Speech and Signal Processing, IEEE, 2009, pp. 945–948.
[25] P.-L. St-Charles, G.-A. Bilodeau, R. Bergevin, Flexible background subtraction with self-balanced local sensitivity, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2014, pp. 408–413.
[26] P.-L. St-Charles, G.-A. Bilodeau, R. Bergevin, A self-adjusting approach to change detection based on background word consensus, in: 2015 IEEE Winter Conference on Applications of Computer Vision, IEEE, 2015, pp. 990–997.
[27] L.A. Lim, H.Y. Keles, Foreground segmentation using convolutional neural networks for multiscale feature encoding, Pattern Recognit Lett 112 (2018) 256–262.
[28] L.A. Lim, H.Y. Keles, Learning multi-scale features for foreground segmentation, Pattern Analysis and Applications 23 (3) (2020) 1369–1380.
[29] P. Rodriguez, B. Wohlberg, Incremental principal component pursuit for video background modeling, J Math Imaging Vis 55 (1) (2016) 1–18.
[30] M.-N. Chapel, T. Bouwmans, Moving objects detection with a moving camera: a comprehensive review, Computer Sci. Rev. 38 (2020) 100310.
[31] B. Zhang, S. Hu, J. Du, X. Yang, X. Chen, H. Jiang, H. Cao, S. Feng, Detecting moving objects in photometric images using 3D hough transform, Publ. Astron. Soc. Pac. 136 (5) (2024) 054502.
[32] A. Dosovitskiy, P. Fischer, E. Ilg, P. Hausser, C. Hazirbas, V. Golkov, P. Van Der Smagt, D. Cremers, T. Brox, Flownet: learning optical flow with convolutional networks, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 2758–2766.
[33] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, T. Brox, Flownet 2.0: evolution of optical flow estimation with deep networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 2462–2470.
[34] D. Sun, X. Yang, M.-Y. Liu, J. Kautz, Pwc-net: cnns for optical flow using pyramid, warping, and cost volume, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 8934–8943.
[35] Z. Teed, J. Deng, Raft: recurrent all-pairs field transforms for optical flow, in: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16, Springer, 2020, pp. 402–419.
[36] M. Gehrig, M. Muglikar, D. Scaramuzza, Dense continuous-time optical flow from event cameras, IEEE Trans. Pattern Anal. Mach. Intell. 46 (7) (2024) 4736–4746.
[37] P. Tokmakov, K. Alahari, C. Schmid, Learning motion patterns in videos, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 3386–3394.
[38] Y. Zhang, P. Sun, Y. Jiang, D. Yu, F. Weng, Z. Yuan, P. Luo, W. Liu, X. Wang, Bytetrack: multi-object tracking by associating every detection box, in: European Conference on Computer Vision, Springer, 2022, pp. 1–21.
[39] A. Pujara, M. Bhamare, DeepSORT: real time & multi-object detection and tracking with YOLO and tensorflow, in: 2022 International Conference on Augmented Intelligence and Sustainable Systems (ICAISS), IEEE, 2022, pp. 456–460.
[40] X. Zhou, V. Koltun, P. Krähenbühl, Tracking objects as points, in: European Conference on Computer Vision, Springer, 2020, pp. 474–490.
[41] F. Yang, Z. Wang, Y. Wu, S. Sakti, S. Nakamura, Tackling multiple object tracking with complicated motions-re-designing the integration of motion and appearance, Image Vis. Comput. 124 (2022) 104514.
[42] K. Yi, K. Luo, X. Luo, J. Huang, H. Wu, R. Hu, W. Hao, Ucmctrack: multi-object tracking with uniform camera motion compensation, in: Proceedings of the AAAI Conference on Artificial Intelligence, 38, 2024, pp. 6702–6710.
[43] X. Chen, B. Yan, J. Zhu, D. Wang, X. Yang, H. Lu, Transformer tracking, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 8126–8135.
[44] Q. Yu, Y. Ma, J. He, D. Yang, T. Zhang, A unified transformer based tracker for anti-uav tracking, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 3036–3046.
[45] H. Lou, X. Duan, J. Guo, H. Liu, J. Gu, L. Bi, H. Chen, DC-YOLOv8: Small-size object detection algorithm based on camera sensor, Electronics 12 (10) (2023) 2323.
[46] M.A. Arefeen, S.T. Nimi, M.Y.S. Uddin, Framehopper: selective processing of video frames in detection-driven real-time video analytics, in: 2022 18Th International Conference on Distributed Computing in Sensor Systems (DCOSS), IEEE, 2022, pp. 125–132.
[47] I. Surenther, K.P. Sridhar, M.K. Roberts, Enhancing data transmission efficiency in wireless sensor networks through machine learning-enabled energy optimization: a grouping model approach, Ain Shams Eng. J. 15 (4) (2024) 102644.
[48] M. Hussain, YOLOv5, YOLOv8 and YOLOv10: The Go-To Detectors for Real-time Vision, 2024. arXiv:2407.02988

[49] Z. Wang, C. Dang, R. Zhang, L. Wang, Y. He, R. Wu, MDDFA-Net: Multi-Scale dynamic feature extraction from drone-Acquired thermal infrared imagery, Drones 9 (3) (2025) 224.
[50] H. Bay, A. Ess, T. Tuytelaars, L. Van Gool, Speeded-up robust features (SURF), Comput. Vision Image Understanding 110 (3) (2008) 346–359.
[51] C. Harris, M. Stephens, et al., A combined corner and edge detector, in: Alvey Vision Conference, 15, Citeseer, 1988, pp. 10–5244.
[52] E. Rosten, T. Drummond, Machine learning for high-speed corner detection, in: Computer Vision–ECCV 2006: 9th European Conference on Computer Vision, Graz, Austria, May 7–13, 2006. Proceedings, Part I 9, Springer, 2006, pp. 430–443.
[53] M. Cao, S. Liu, F. Cao, Midpoint distance circle generation algorithm based on midpoint circle algorithm and bresenham circle algorithm, in: Journal of Physics: Conference Series, 1438, IOP Publishing, 2020, p. 012017.
[54] M. Calonder, V. Lepetit, M. Ozuysal, T. Trzcinski, C. Strecha, P. Fua, BRIEF: Computing a local binary descriptor very fast, IEEE Trans Pattern Anal Mach Intell 34 (7) (2011) 1281–1298.
[55] H. Guan, W. Diao, A sequential fusion method of multi-Radar tracking orbital target based on EKF, in: 2023 6Th International Conference on Information Communication and Signal Processing (ICICSP), IEEE, 2023, pp. 579–584.
[56] L. Ren, W. Yin, W. Diao, K. Fu, X. Sun, SuperMOT: decoupling motion and fusing temporal pyramid features for UAV multi-Object tracking, IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. (2025).
[57] K. Saleh, S. Szénási, Z. Vámossy, Occlusion handling in generic object detection: a review, in: 2021 IEEE 19Th World Symposium on Applied Machine Intelligence and Informatics (SAMI), IEEE, 2021, pp. 000477–000484.
[58] R. Liu, M. Huang, L. Wang, C. Bi, Y. Tao, PDT-YOLO: A roadside object-detection algorithm for multiscale and occluded targets, Sensors 24 (7) (2024) 2302.
[59] K. Zeng, Y. You, T. Shen, Q. Wang, Z. Tao, Z. Wang, Q. Liu, NCT: Noise-control multi-object tracking, Complex & Intelligent Systems 9 (4) (2023) 4331–4347.
[60] P. Voigtlaender, M. Krause, A. Osep, J. Luiten, B.B.G. Sekar, A. Geiger, B. Leibe, Mots: multi-object tracking and segmentation, in: Proceedings of the Ieee/cvf Conference on Computer Vision and Pattern Recognition [Dataset], 2019, pp. 7942–7951.
[61] A. Geiger, P. Lenz, C. Stiller, R. Urtasun, Vision meets robotics: the kitti dataset, The International J. Robotics Res. [dataset] 32 (11) (2013) 1231–1237.
[62] J. Pont-Tuset, F. Perazzi, S. Caelles, P. Arbeláez, A. Sorkine-Hornung, L. Van Gool, The 2017 davis challenge on video object segmentation [dataset], (2017), arXiv:1704.00675.
[63] P. Zhu, L. Wen, D. Du, X. Bian, H. Fan, Q. Hu, H. Ling, Detection and tracking meet drones challenge, IEEE Trans. Pattern Anal. Mach. Intell. (2021) 1. https://doi.org/10.1109/TPAMI.2021.3119563
[64] D.G. Lowe, Distinctive image features from scale-invariant keypoints, Int. J. Comput. Vis. 60 (2004) 91–110.
[65] S. Leutenegger, M. Chli, R.Y. Siegwart, BRISK: Binary robust invariant scalable keypoints, in: 2011 International Conference on Computer Vision, 2011, pp. 2548–2555. https://doi.org/10.1109/ICCV.2011.6126542
[66] P.F. Alcantarilla, T. Solutions, Fast explicit diffusion for accelerated features in nonlinear scale spaces, IEEE Trans. Patt. Anal. Mach. Intell 34 (7) (2011) 1281–1298.
[67] Y. Tian, Q. Ye, D. Doermann, Yolov12: Attention-centric real-time object detectors, (2025), arXiv:2502.12524.
[68] M. Cao, C. Wang, W. Zhao, Z. Zhang, AMF-MOT: Multi-Object tracking based on motion-Appearance feature fusion for object vehicle loss and occlusion, IEEE Trans. Veh. Technol. (2025).
[69] W. Lv, Y. Zhao, Q. Chang, K. Huang, G. Wang, Y. Liu, Rt-detrv2: Improved baseline with bag-of-freebies for real-time detection transformer, (2024), arXiv:2407.17140.
[70] F. Yang, S. Odashima, S. Masui, S. Jiang, Hard to track objects with irregular motions and similar appearances? make it easier by buffering the matching space, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2023, pp. 4799–4808.
[71] C. Xiao, Q. Cao, Y. Zhong, L. Lan, X. Zhang, Z. Luo, D. Tao, Motiontrack: learning motion predictor for multiple object tracking, Neural Netw. 179 (2024) 106539.
[72] Z. Liu, X. Wang, C. Wang, W. Liu, X. Bai, Sparsetrack: multi-object tracking by performing scene decomposition based on pseudo-depth, IEEE Trans. Circuits Syst. Video Technol. (2025).
[73] C. Liang, W. Wang, T. Zhou, J. Miao, Y. Luo, Y. Yang, Local-global context aware transformer for language-guided video segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 45 (8) (2023) 10055–10069.
[74] L. Van Ma, T.T.D. Nguyen, C. Shim, D.Y. Kim, N. Ha, M. Jeon, Visual multi-object tracking with re-identification and occlusion handling using labeled random finite sets, Pattern Recognit. 156 (2024) 110785.
[75] X. Zhang, X. Tan, Y. An, Y. Li, Z. Fan, OATracker: Object-aware anti-occlusion 3D multiobject tracking for autonomous driving, Expert Syst. Appl. 252 (2024) 124158.