Air-to-Ground Real-Time Temporal Small Object Detection from a Flying Platform

Ella Fokkinga[a], Martin van Leeuwen[a], and Hugo Kuijf[a]

[a]TNO - Intelligent Imaging, Oude Waalsdorperweg 63, the Hague, the Netherlands

ABSTRACT

Real-time small object detection in air-to-ground scenarios is important for military applications, enabling unmanned aerial vehicles (UAVs) and other airborne platforms to monitor large areas. The detection of small objects on the ground remains a significant challenge because of their lack of distinctive features and the low signal-to-noise ratio. Temporal-YOLO has demonstrated that exploiting temporal information in video data improves small object detection across diverse environments. However, the current implementation of Temporal-YOLO is not suitable for real-time deployment on a moving aerial platform.

In this study, we extend Temporal-YOLO by addressing two key limitations. First, because Temporal-YOLO assumes a static camera over frames, it requires compensation for the platform motion. We implement and evaluate various (global) motion compensation techniques to assess their impact on detection performance. Second, the original Temporal-YOLO model introduces latency by incorporating a future frame. To enable real-time inference, we propose a modification that relies solely on past frames. Through evaluation on an aerial dataset featuring small military-relevant objects, we demonstrate that global motion compensation significantly enhances detection accuracy and that the low-latency historical approach achieves comparable performance.

Keywords: Small object detection; air-to-ground imaging; temporal object detection; real-time inference, EDF STORE

1. INTRODUCTION

The modern battlefield is increasingly characterized by its transparency. Manned and unmanned airborne platforms are more and more used to monitor large areas with electro-optical (EO) and infrared (IR) sensors.[1,2] These platforms generate vast amounts of video data, often in real time, which can quickly lead to information overload for human operators when entirely left to manual monitoring. As a result, automated analysis tools are essential to support timely and accurate decision-making.

Automated object detection is a key capability for military intelligence, surveillance, reconnaissance, and threat assessment tasks. Early detection of objects of interest is particularly valuable, because it provides commanders with more time to respond. Achieving this requires recognition of objects from afar, when they are still very small, often just a few pixels in size. Such targets, including distant vehicles, personnel, or drones, typically lack distinctive features, have limited spatial resolution, and are easily lost in cluttered or low-contrast backgrounds. Traditional object detectors often struggle when observing under these long-range conditions.[3,4] As a result, small object detection remains an ongoing challenge despite advances in computer vision.[5–8]

The incorporation of temporal information in deep learning-based object detectors can significantly improve small moving object detection performance.[9–15] Temporal-YOLO, an extension of the YOLOv8 architecture,[16,17] leverages motion cues across video frames by stacking multiple frames as input.[18] Temporal-YOLO shows promising performance when tested on a large-scale dataset representative for various military scenarios with annotated small moving objects.[10]

However, the current implementation of Temporal-YOLO is not yet suitable for real-time deployment on airborne platforms. First, it assumes a static camera viewpoint across frames, which limits its applicability in real-world aerial scenarios where the platform is in motion. Without compensating for this motion, the temporal

Corresponding author: Ella Fokkinga. E-mail: ella.fokkinga@tno.nl

consistency that Temporal-YOLO relies on is disrupted, leading to degraded detection performance. Second, the original Temporal-YOLO architecture introduces latency by incorporating a future frame into its input, which is not feasible for real-time deployment.

In this work, we address both of these limitations. We introduce an image registration module that aligns frames prior to detection. We evaluate a range of global image registration techniques, including classical intensity- and feature-based methods as well as deep learning-based approaches, and assess their impact on detection performance. We propose a low-latency variant of Temporal-YOLO that relies solely on past frames, making it suitable for real-time inference. Through experiments on an air-to-ground dataset featuring small, military-relevant objects, we demonstrate that image registration significantly improves detection accuracy, and that the historical-only approach achieves comparable performance to the original Temporal-YOLO while enabling real-time operation.

2. RELATED WORKS

2.1 Small object detection

The detection of small objects remains a challenging task, particularly in aerial and defense applications, where targets often appear at low resolution, lack distinctive features, and are surrounded by background clutter.[2,5–8] Traditional approaches for small object detection relied on classical computer vision techniques based on handcrafted features. These methods typically fall into two categories: *single-frame* methods, which enhance small targets using specialized filters such as top-hat transforms, and *multi-frame* methods, which leverage temporal cues through background subtraction, optical flow, or frame differencing.[19–25] While effective in constrained settings, these classical methods often struggle to generalize across dynamic or complex environments.

The limited adaptability of handcrafted features motivated the transition to deep learning, which can learn discriminative representations directly from data. Models such as R-CNN, Grounding DINO, and the YOLO family have demonstrated strong performance on regular object detection benchmarks.[26–28] However, these detectors are typically optimized for medium to large objects, and their performance drops significantly when applied to small or low-resolution targets.[7] To address this, various adaptations have been proposed, including multi-scale representation, contextual reasoning, super-resolution techniques, and region proposal refinement.[2,6,8]

Most of these methods operate on *single frames*, ignoring temporal information available in video data. While temporal information has long been used in classical *multi-frame* methods, it is now being reintroduced in deep learning-based small object detection frameworks.[29] Techniques such as optical flow estimation[30] have been integrated into neural networks to enhance detection robustness across frames in both visual and infrared domains.[9,31,32] More recently, deep learning approaches have been proposed to directly incorporate temporal data by providing multiple frames as input. For instance, Yan et al. (2023)[33] developed a spatio-temporal network for infrared SOD that extracts both appearance and motion cues across frames.

Within the YOLO family of detectors, several extensions have been introduced to exploit temporal information for small moving object detection.[14,15,18] The Temporal-YOLO framework modifies the standard YOLO architecture to accept multiple time steps as input channels.[18] Initial implementations of Temporal-YOLO demonstrated strong performance improvements over both non-deep-learning methods and single-frame baselines.[34] In our recent work,[10] we extended this approach by introducing a bounding box augmentation strategy tailored for small objects and a balanced mosaicking technique to mitigate foreground–background imbalance. Additionally, we showed the benefits of training on a specialized small object detection dataset, emphasizing the need for diverse and representative training data in this domain. Temporal-YOLO has also been shown to maintain performance under adverse weather conditions,[35] further supporting its potential for real-world deployment.

2.1.1 Datasets

The development of effective small object detectors strongly depends on the availability of high-quality annotated datasets. Many existing datasets lack sufficient representation of small targets or fail to capture the complexity of real-world environments.[7,10] In this work, we focus on the detection of small moving objects from a moving aerial platform, which is a scenario that presents unique challenges not adequately addressed by existing datasets. While

the VisDrone dataset[36] is widely used for aerial object detection, it lacks environmental diversity and does not does not provide consistent annotations of small objects. Other datasets such as DOTA[37] and SODA[38] provide high-resolution imagery with small objects from aerial perspectives, but they consist solely of still images, which precludes leveraging temporal information. XS-VID[39] includes video sequences and smaller objects, but mixes static and moving object annotations, complicating efforts to isolate and optimize for motion-based small object detection.

2.2 Image registration

When the sensing platform or camera is in motion, the background shifts across frames, making it difficult to distinguish moving targets from camera-induced motion. To ensure that only objects of interest are tracked, image registration can be applied. Image registration is a well-established problem in fields such as medical imaging, remote sensing, and surveillance.[40–42] Its objective is to align two or more images captured at different times, viewpoints, or by different sensors, enabling consistent analysis across frames or modalities. Mathematically, this involves estimating a spatial transformation that maps pixel locations from a moving image to a fixed reference image. Registration methods are typically categorized into *global* and *local* approaches. Global approaches estimate a single perspective transformation matrix to align the entire frame, offering computational efficiency and robustness to local noise. These characteristics make them suitable for real-time applications. However, global methods assume a consistent motion model across the scene and therefore struggle with parallax effects, where foreground and background objects shift differently because of depth variation. Local registration methods address this by estimating a dense motion field that can vary across the image. This field is often smoothed using filters such as median or bilateral filtering, scaled to the image resolution, and then the image is transformed accordingly.[43–46] While local methods provide more flexibility in dynamic or layered scenes, they are more sensitive to noise and outliers in the motion estimate and often come with higher computational costs. In this study, we therefore focus on global image registration methods. The remainder of this section will describe several global registration techniques. In many global approaches, the process is split into two steps: *motion estimation*, where corresponding points between the two images are identified, and *motion compensation*, where these correspondences are used to compute a transformation matrix (e.g. using RANSAC), which is then used to transform the image. The motion estimation can be performed using *intensity-based*, *feature-based* and *deep-learning-based* approaches. Alternatively, some methods estimate the transformation matrix directly based on the images, combining the motion estimation and compensation into a single step. The various approaches to global image registration are illustrated in Figure 1.

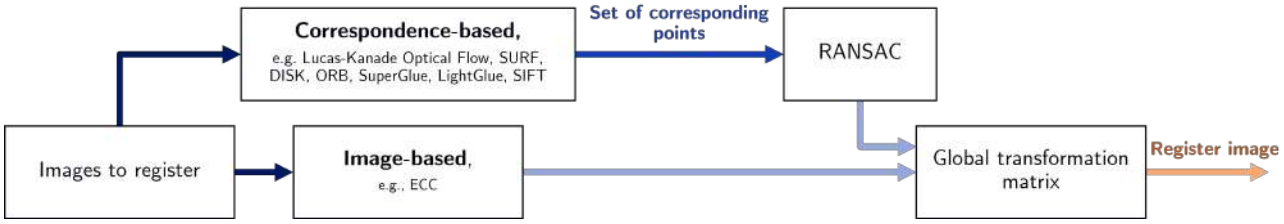


Figure 1: Schematic overview of image registration methods. In this work, image registration is used to align frames captured from a moving platform, enabling the use of temporal information for small object detection. Global image registration involves computing a transformation matrix that aligns the entire image. This can be done either by estimating the matrix from the whole image, or by splitting the process into two steps: motion estimation and motion compensation. In the latter case, corresponding point sets are first identified, and the transformation matrix is then estimated. Finally, the resulting matrix is used to warp the image.

2.2.1 Motion estimation

The goal of motion estimation is to identify a set of corresponding points between two frames, which form the basis for computing the transformation that aligns them. Intensity-based methods detect motion between frames by directly comparing pixel intensities, without relying on feature extraction. Many intensity-based techniques are based on an estimation of *optical flow*, where the apparent motion of objects or surfaces between frames

is computed.[30,47] Two traditional well-known optical flow approaches are the Lucas-Kanade method,[48] which estimates motion locally by assuming constant flow within small regions of the image, and the Horn-Schunk method,[30] which computes a dense flow field over the whole image. Another common subclass of intensity-based approaches is area-based registration, which evaluates similarity over the images using a window or region rather than individual pixels.[40] This can be advantageous in air-to-ground imagery, where texture is often sparse and keypoints are difficult to detect. By considering entire regions, more stable alignment can be achieved for scenes with uniform textures, such as fields or dense forests. A key limitation of intensity-based methods is their sensitivity to lighting changes and their dependency on consistent intensity values across frames. To address this, Evangelidis and Psarakaris (2008)[49] proposed to use the *Enhanced Correlation Coefficient (ECC)*, which estimates a global motion model by maximizing the correlation coefficient between image intensities and is therefore more robust to photometric distortions.

Feature-based methods detect and match keypoints across frames to determine motion between frames. These methods are generally more robust to illumination changes and computationally more efficient, because they operate on a sparse set of keypoints rather than processing all pixel intensities across the image. Popular classical methods include SIFT,[50] SURF,[51] and ORB,[52] which are robust to scale, rotation, and lighting variations. However, their performance degrades when no distinct features can be found, such as in low-contrast or repetitive-pattern environments.

More recently, deep learning-based methods have been proposed as a promising alternative to classical approaches.[53,54] These models learn motion representations directly from training data, often achieving higher accuracy and robustness to complex distortions. Notable methods include SuperPoint,[55] which jointly learns keypoint detection and description, and SuperGlue,[56] which uses graph neural networks with attention to improve feature matching. LightGlue[57] offers a much faster alternative for matching that is adaptive to the difficulty of the alignment. In DISK,[58] reinforcement learning is used to train both the keypoint detector and descriptor. While these deep learning-based methods often achieve high accuracy, they often involve higher computational costs than classical methods.

2.2.2 Motion compensation

To estimate a robust global motion model for image registration, RANSAC (Random Sample Consensus) can be applied[59] (Figure 1). The found correspondences between the images to register frequently include outliers because of noise, occlusions, or incorrect feature matches. RANSAC addresses this by iteratively selecting random subsets of point correspondences to estimate a candidate transformation matrix, typically a homography. For each iteration, RANSAC computes the transformation and then evaluates how many of the remaining correspondences fit the model within a specified tolerance. The candidate transformation matrix with the most keypoints within this tolerance is selected as the final solution. This approach ensures that the final transformation matrix used to align the images is robust to incorrectly matched keypoints, leading to accurate and reliable registration results, even in challenging conditions. As can be seen from Figure 1, this iterative filtering step is only required for methods that result in a set of point correspondences, such as Lucas-Kanade sparse optical flow,[48] SIFT,[50] and LightGlue.[57] In contrast, direct estimation methods such as ECC[49] compute the transformation directly from image intensities and therefore do not require a separate RANSAC stage.

2.3 Temporal information: past, present, or future?

Temporal-YOLO, like many temporal models, combines frames from the past, present, and future. A similar design is used by for instance Alqaysi et al. (2021), stacking frames from $t-1$, t, and $t+1$ as input.[14] This approach introduces a latency because of its reliance on future frames. Several recent works demonstrate that temporal information can be effectively incorporated using only historical data instead. For instance, Luesutthiviboon et al. (2023)[15] enhance target-to-background contrast using preprocessing methods based solely on preceding frames. Sun et al. (2024)[9] propose Multi-YOLOV8, which integrates temporal information through both optical flow and background suppression computed from the current and previous frames. Similarly, Guo et al. (2024)[39] present the YOLOFT model, which extracts local motion features between t and $t-1$ using optical flow to guide temporal feature fusion. Poplavskiy (2024)[60] adopts a tracking-by-detection pipeline that stacks the current frame with an aligned previous frame using deep optical flow, entirely without future input. Based on these

promising examples from recent literature, this study introduces a modified version of Temporal-YOLO in this study, that excludes future frames and instead uses the current frame and two preceding frames.

3. METHODS

Within this section, we provide implementation details for the image registration techniques, the object detector, the dataset and the evaluation procedure.

3.1 Image registration

As described in Section 2.2, image registration methods can be categorized into classical intensity-based, optical flow-based (a subclass of intensity-based approaches), feature-based, and deep learning-based approaches. To evaluate their suitability for our Temporal-YOLO pipeline, we selected one representative method from each category:

- *ECC*: Enhanced Correlation Coefficient for intensity-based alignment.[49]
- *Optical flow*: Lucas-Kanade for sparse optical flow-based motion estimation.[48]
- *SIFT*: SIFT for feature extraction with FLANN for feature matching.[50]
- *LightGlue*: SuperPoint for feature extraction and *LightGlue* for deep learning-based matching.[55,57]

Each of these methods offers distinct trade-offs in terms of accuracy, robustness, and computational cost. Our goal is not to perform an exhaustive benchmark of all available image registration techniques, but rather to identify a method that provides sufficient alignment quality for effective multi-frame small object detection, while remaining feasible for real-time or near-real-time deployment. To that end, we did not optimize each method for minimal runtime.

All image registration methods are applied as a pre-processing step, after converting the frames to grayscale, and before providing the frames as input to the model. For all three classical approaches, we use an OpenCV implementation[61] to estimate the motion. For *ECC*, we use a three-level pyramid implementation to estimate the global transformation matrix in a coarse-to-fine manner, with a maximum of 400 iterations and a convergence threshold of 1×10^{-5}. Lucas-Kanade *optical flow* is computed on a sparse, regularly spaced grid centered in the image, with spacing set to one-tenth of the image height. This ensures an even distribution of flow vectors across the frame, reduces computational cost, and allows focusing on large-scale motion. For *SIFT*, we extract a maximum of 500 features per frame from downscaled images (50%) and match them using OpenCV's FLANN-based matcher. SuperPoint is used to extract a maximum number of 1024 keypoints, and *LightGlue* performs the matching with a depth confidence of 0.9 and width confidence 0.95. For *optical flow*, *SIFT*, and *LightGlue* the resulting set of point correspondences (p_1, p_2) are passed to a RANSAC-based homography estimator to compute a transformation matrix that aligns the frames, while filtering out outliers.[59] Homographies between the correspondences were estimated using OpenCV's RANSAC-based method with a reprojection threshold of 5 pixels, a confidence level of 0.995, and a maximum number of 2000 iterations.

3.2 Small object detection and temporal information

For our object detector, we build upon the Temporal-YOLO architecture introduced by Corsel et al. (2023)[18] and further improved in Leeuwen et al. (2024),[10] which adapts YOLOv8 for small object detection by exploiting temporal context (Figure 2).

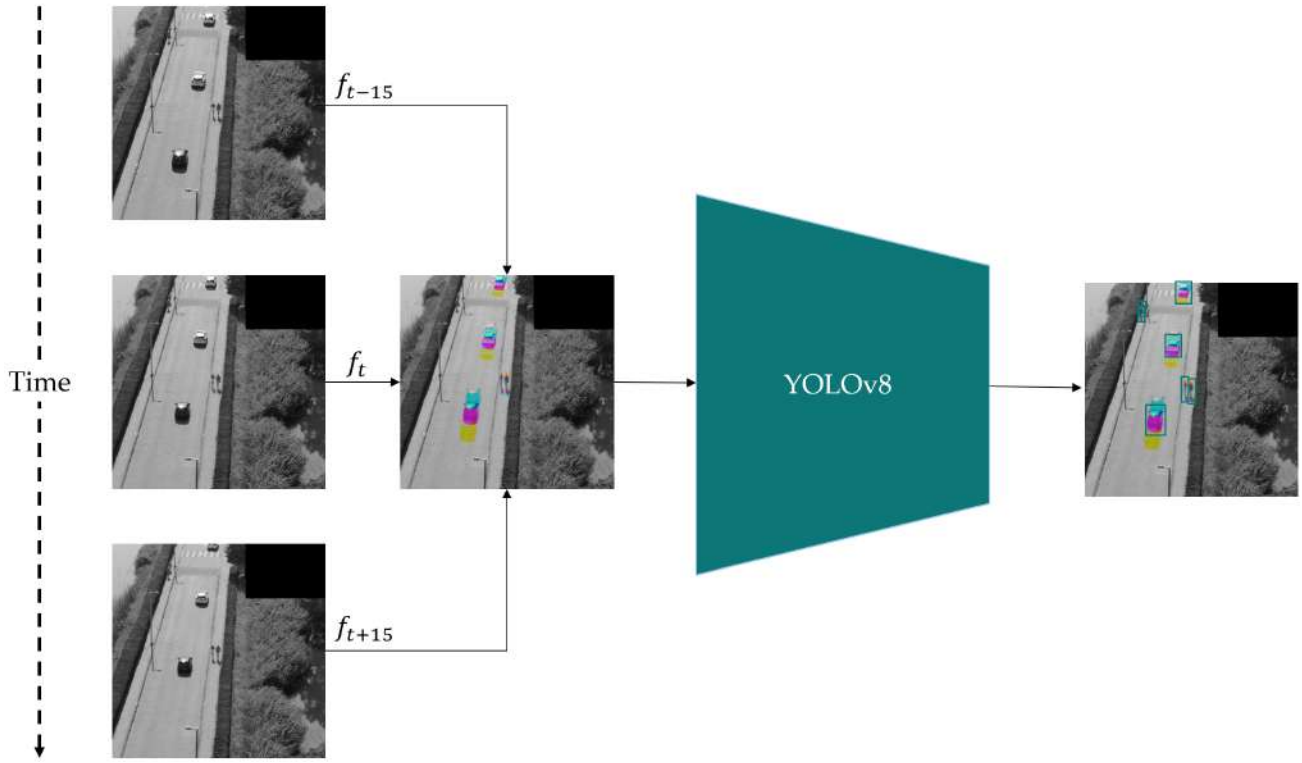


Figure 2: An illustration of the Temporal-YOLO concept. Instead of using a single video frame as input, multiple frames are stacked from different time steps. The RGB channels are replaced with three grayscale frames. This combined 3-channel image is provided to the model, allowing temporal context to be exploited. Assuming a 30-frames-per-second (FPS) source video and time offsets of -0.5 and $+0.5$ seconds, around 15 frames are sampled before and after the current frame in the originally proposed method.[10,18]

Instead of standard RGB inputs, Temporal-YOLO stacks grayscale frames from different time steps across the input channels. To address challenges specific to small object detection,[7] the improved Temporal-YOLO incorporates several enhancements: (1) balanced mosaicking to mitigate foreground-background imbalance; (2) bounding box augmentations, which scale small ground truth boxes to avoid disproportionate penalization of errors in small objects, (3) training on a high-quality, in-house annotated datasets (Nano-VID) with small, moving objects across varied environments and conditions. For the full methodology and ablation analysis, see the original paper.[10] To enhance performance in air-to-ground settings, we retrained the model using an additional in-house dataset comprising footage captured from a plane. This supplementary dataset includes annotations for 766 objects.

To reduce inference latency in Temporal-YOLO, we introduce a *past* variant that selects frames at temporal offsets of $[0s, –0.5s, –1s]$, compared to the original *bidirectional* setup using $[+0.5s, 0s, -0.5s]$. Since all input frames originate at or before the current frame, no additional delay is introduced during inference on top of the model execution. With this approach, we maintain the level of temporal context but shift some of this context somewhat further away from the potential target position. A potential drawback is that this *past* variant could require a larger perceptive field. However, in practice, faster movements result in stronger brightness variations, making them more distinct. As such, no strong performance reduction is expected in practice.

In the *past* variant of our pipeline, the motion models are estimated in two sequential steps: from frame -1 to -0.5, and then from -0.5 to the reference frame at time 0. These are then composed to derive a direct transformation from -1 to 0. This two-step approach ensures that each motion estimation covers a maximum interval of 0.5 seconds, maintaining consistency with the *bidirectional* variant, which also uses frames spaced no more than 0.5 seconds apart.

3.3 Datasets

To evaluate both the image registration methods and the low-latency approach, we assembled a dataset featuring air-to-ground scenarios with a variety of small targets. Since Temporal-YOLO is designed for detecting small *moving* objects, we only annotate those, while static objects are marked as ignored. Figure 3 shows a histogram with the bounding box size ($\sqrt{\text{bbox}_w \cdot \text{bbox}_h}$) of annotated objects in our dataset, which is typically around 16 pixels. Note that we did not focus on fitting the annotated bounding boxes as tightly as possible around objects, to simplify the annotation process.

The evaluation dataset consists of two parts. The first was recorded during the EDF STORE trial at the Fontevraud firing range site in September 2024, which included moving persons and vehicle maneuvers at varying speeds. Vehicles were provided by the Musée de Saumur and included T72 and T62 battle tanks, BMP-1 infantry fighting vehicles, MTLB armored personnel carriers, and 2S1/2S3 self-propelled artillery. Data was captured using drones flying at different velocities and heights, recording both IR and visual modalities under diverse environmental conditions across forests and plains. The second part was included to increase diversity in scenarios, sensors configurations, and platform speeds. It consists of an in-house dataset again featuring military vehicles and personnel, primarily in forested environments, recorded from both a fixed-wing aircraft and drone at various speeds.

To give an indication of the speed at which the camera is moving, we compute the L2-norm of the translation component of the homography matrix estimated by the *LightGlue*-based method (Section 3.1), between frames spaced 1 second apart. While these results are not guaranteed to be accurate at all times and errors can naturally be present in the motion estimation, they can be used to gain insights into the camera movement in the dataset.

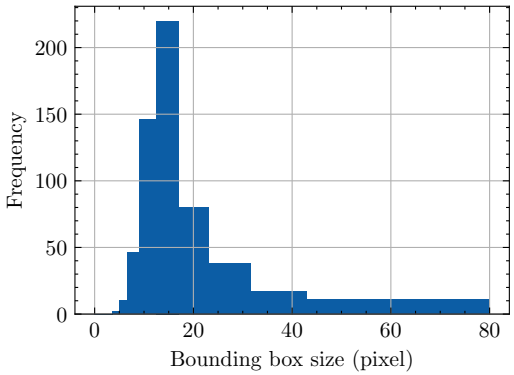


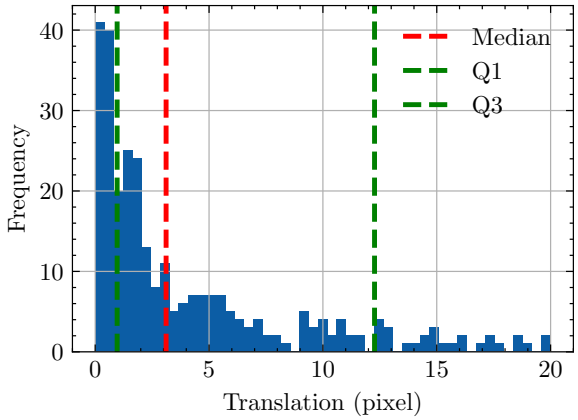
Figure 3: Histogram of the bounding box sizes in the test set computed by $\sqrt{\text{bbox}_w \cdot \text{bbox}_h}$.

3.4 Evaluation

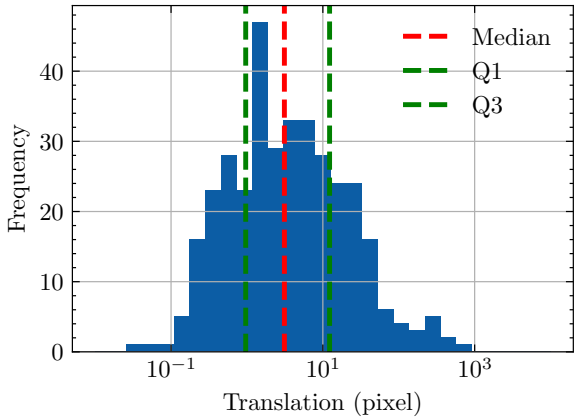
A total of ten experiments will be executed, testing the baseline Temporal-YOLO as well as Temporal-YOLO combined with the four image registration methods, *ECC, Optical flow, SIFT*, and *LightGlue*, each evaluated under two temporal variants: *past* and *bidirectional*. Throughout the results section, we will refer to these experiments using this format. Overall detection performance of the model reflects both the accuracy of Temporal-YOLO and the quality of the image registration method used as pre-processing step. As a primary tool for evaluation, the precision–recall (PR) curves are computed, which offer insights into the trade-off between precision and recall across varying confidence thresholds. In addition, we report the mean Average Precision (mAP), using the modified formulation proposed by van Leeuwen et al.,[10] which uses a more lenient intersection-over-union (IoU)-threshold of 1% instead of the conventional 50%. This accounts for the fact that minor deviations in bounding box predictions can lead to disproportionately low IoU scores for small objects, even when detections are visually useful. Additionally, the evaluation metric is adapted to avoid penalizing multiple detections within a single annotation or a single detection overlapping multiple annotations—reflecting the practical goal of drawing operator attention rather than perfect box alignment. To get an idea of the computational efficiency of the image registration methods, we report the total end-to-end inference time per frame including the image registration task.

4. RESULTS

Figure 4 shows the distribution of the L2-norm of the translations between frames spaced one second apart, estimated in the *LightGlue*-based transformation matrices. The median translation magnitude is approximately 3.11 pixels ($Q_1 = 0.96$, $Q_3 = 12.3$), indicating a moderate displacement of the camera platform. As highlighted in Figure 4b, the distributions show a long tail, revealing substantial variation in motion across frames.



(a) Histogram on a linear scale, showing that most frame translations are below 10 pixels.



(b) Histogram on a logarithmic scale, revealing a long tail of larger translations ranging from 10^1 to 10^3 pixels.

Figure 4: Distribution of estimated translations in the evaluation set, computed from the translation component of the transformation matrix using the *LightGlue* deep-learning-based method. The median translation is 3.11 pixels ($Q_1 = 0.96$, $Q_3 = 12.3$).

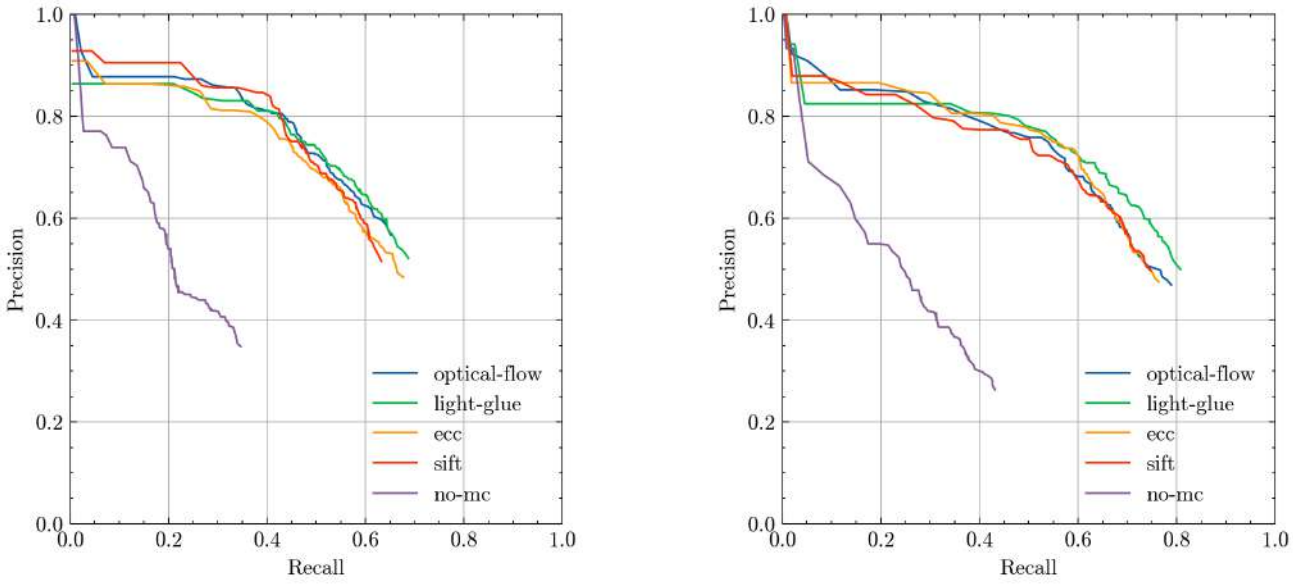
Frames with evaluated detections for the *optical flow-past* method are shown in Figure 5. Temporal-YOLO successfully detects very distant moving objects, including both pedestrians and vehicles. False positives, visible in the right frame, are in this case caused by dead pixels in the IR-imagery.

Figure 5: Frames from the evaluation dataset with detections. Blue boxes indicate the zoomed-in region shown in the top-left corner, green boxes indicate true positives, and orange boxes indicate false positives.

4.1 Image registration

Figures 6a and 6b show the precision-recall curves for the baseline Temporal-YOLO without image registration alongside the four image registration methods, for the *bidirectional* and *past* variant respectively. All methods

exhibit an immediate drop in precision as recall increases. The baseline without registration shows the most significant decline, with precision dropping rapidly and failing to exceed a recall of 41%. In contrast, the detector with *LightGlue* achieves the highest recall, with the *past* variant exceeding 80% and attaining a maximum mAP of 0.62 (Table 1). All registration methods yield a substantial improvement in performance across both variants, producing similar precision-recall curves, with *past* mAP scores ranging between 0.57 and 0.62. Although differences are small, *LightGlue* exhibits a slight advantage in both precision and recall, as confirmed by the mAP reported in Table 1.



(a) *Bidirectional* variant. (b) *Past* variant.

Figure 6: Precision–recall curves computed for baseline Temporal-YOLO without image registration and Temporal-YOLO with four image registration methods, leveraging the *bidirectional* (left) and *past* (right) contexts.

Table 1: The mAP results for the baseline Temporal-YOLO without image registration and the four image registration methods.

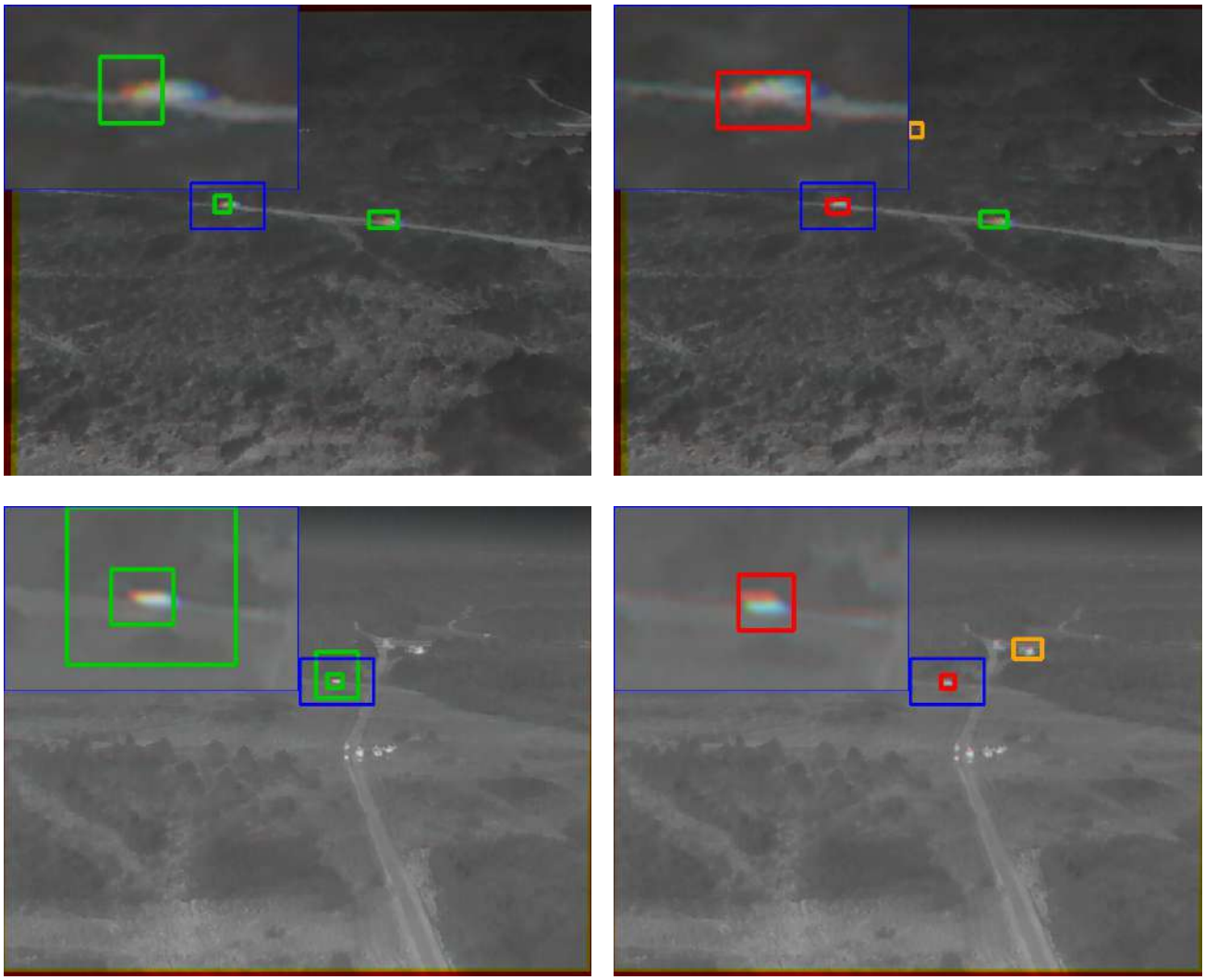
	Baseline	**ECC**	**Optical flow**	**SIFT**	**LightGlue**
Bidirectional	0.21	0.52	0.52	0.52	0.54
Past	0.23	0.59	0.60	0.57	0.62

The median execution time for each of the methods is reported in Table 2. These results cover the non-optimized settings of the algorithms, since optimizing the algorithm's efficiency was beyond the scope of this work. The *optical flow*-based method is the fastest method, achieving a median execution time of 0.08 seconds even at the highest resolution frames present in our dataset. Given that the *optical flow*-based method shows promise in terms of inference speed, and *LightGlue* demonstrates a slight advantage in terms of detection accuracy, we focus on these two methods for the remainder of the experiments.

To illustrate the impact of registration accuracy on detection performance, Figure 7 shows two examples where small misalignments affect the outcome. Even minor registration errors can result in missed objects or production of false positives. In these examples, *LightGlue* (Figure 7a) achieves accurate alignment, resulting in successful detections, whereas *optical flow* leads to missed detections. These examples were selected to highlight the effect of registration quality, since overall, the difference between the two methods on our evaluation dataset is relatively small, as reflected in the quantitative results (Table 1).

Table 2: Median execution time in seconds for each image registration method, at different resolutions present in the evaluation dataset.

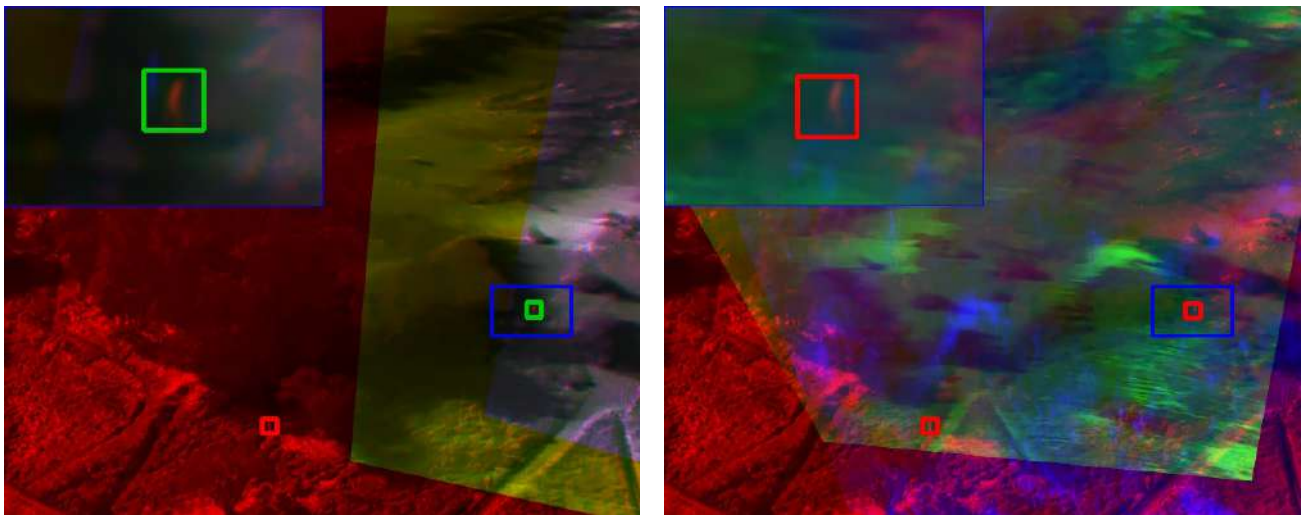
Resolution	ECC	Optical flow	SIFT	LightGlue
3840x2160	10.62	0.08	6.25	0.55
1920x1080	4.22	0.02	1.48	0.41
640x512	1.42	0.01	0.21	0.37



(a) *LightGlue*-based image registration method. (b) *Optical flow*-based image registration method.

Figure 7: Examples of the *past* variant of Temporal-YOLO, illustrating how slight registration errors can impact detection accuracy. The prediction frame at t is shown in gray, with registered frames from one step back ($t-0.5$ s) and two steps back ($t-1$ s) are overlaid in yellow and red, respectively. When registration is accurate, static regions remain gray, while moving objects appear in color. Blue boxes indicate the zoomed-in region shown in the top-left corner; green boxes indicate true positives, orange boxes false positives, and red boxes missed detections (false negatives). In these examples, *LightGlue* (left) achieves accurate alignment and correct detections, while the *optical-flow*-based method (right) introduces a small misalignment that result in missed detections and false positives.

Figure 8 illustrates a case with one of the largest camera displacements in the dataset: a translation of 599 pixels between the prediction frame on t and frame $t-1$ s (see Figure 4). In such extreme cases, registration becomes challenging. Here, the *LightGlue*-based method still achieves sufficient alignment for Temporal-YOLO to correctly detect the moving object, whereas the *optical-flow*-based method fails completely, resulting in a missed detection. This examples suggests that *LightGlue* might be more robust under large displacements, although such cases are rare in our dataset, and do not translate into a considerable difference in overall detection performance.

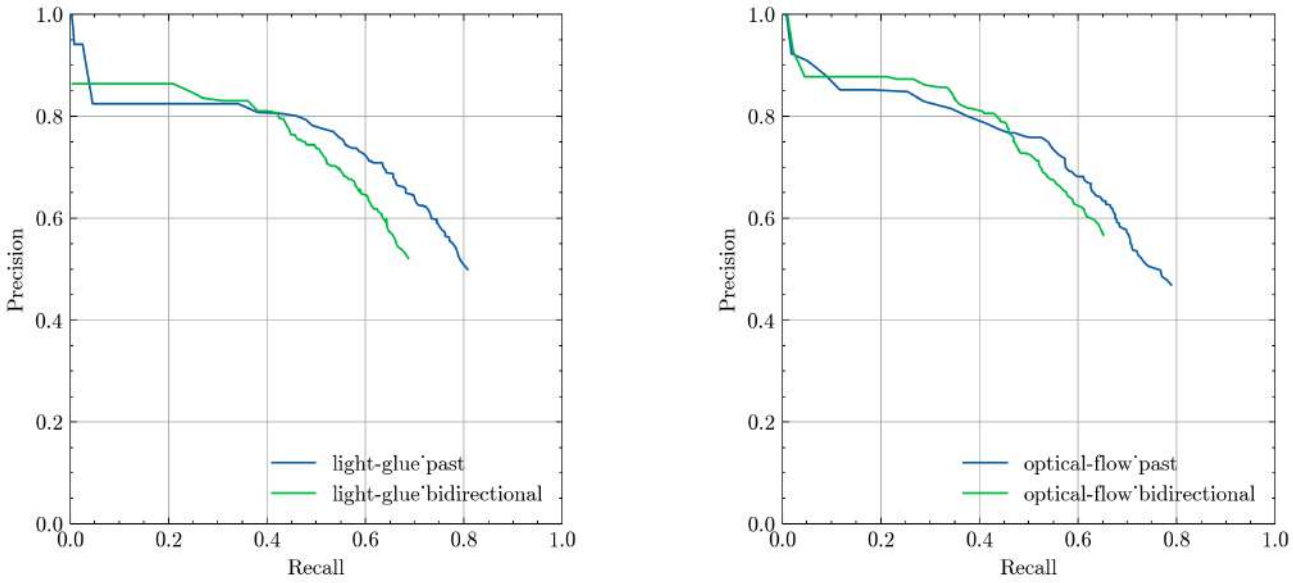


(a) *LightGlue*-based image registration method. (b) *Optical flow*-based image registration method.

Figure 8: Examples of the *past* variant of Temporal-YOLO under large camera displacement: 599 pixels in the *LightGlue*-based transformation matrix between frame t and $t-1$ s). The prediction frame is shown in gray, with registered frames from one step back ($t-0.5$ s) and two steps back ($t-1$ s) overlaid in yellow and red, respectively. When registration is accurate, static regions should remain gray, while moving objects should appear in color, highlighting their displacement. Blue boxes indicate the zoomed-in region shown in the top-left corner; green boxes denote true positives, and red boxes missed detections (false negatives). In this example, *LightGlue* (left) maintains sufficient alignment for correct detection, while the *optical-flow*-based method (right) fails, causing no image region to remain gray, and the object to be missed.

4.2 Past versus bidirectional

In Figure 9, the PR-curves for the *bidirectional* and the *past* variants of Temporal-YOLO are reported. For both image registration methods, the *past* variant achieves the highest recall. The mAP results in Table 1 support this, with the average mAP of the *past* variants with image registration (0.57) higher than those of the *bidirectional* variants (0.52).



(a) *LightGlue*-based image registration method. (b) *Optical flow*-based image registration method.

Figure 9: Precision–recall curves computed for the *bidirectional* and *past* variants of Temporal-YOLO, for two of the image registration methods: *LightGlue* (left) and *optical flow* (right).

5. DISCUSSION

In this paper, we addressed two limitations that hinder the use of Temporal-YOLO for air-to-ground real-time object detection from a flying platform. First, we mitigated the dependency on future frames by introducing a setting that relies only on past frames. Second, we corrected for platform motion by integrating image registration into the detection pipeline. We implemented and evaluated various classical global image registration techniques (intensity-, optical flow-, and feature-based) and one deep learning-based approach, to identify a technique that provides sufficient alignment quality for Temporal-YOLO to perform accurately while remaining feasible for real-time deployment. We found that image registration for frames recorded from moving platforms substantially improves small object detection performance, as visible in Figure 6. Upon visual inspection, we found that Temporal-YOLO indeed misses objects under unregistered camera displacements. We observed minimal performance differences between the evaluated registration methods. *LightGlue* shows a slight advantage in recall and appears to be more robust under large camera displacements.

To enable real-time detection, we evaluated a variant of Temporal-YOLO that relies solely on past frames. Both the precision-recall curves and mAP scores show that this *past* variant performs comparably to the *bidirectional* variant, with a slight improvement in both recall and mAP. One possible explanation is a data sampling bias in the evaluation dataset: since objects are annotated only sparsely at selected points, more annotated instances may coincide with frames where past motion is visible rather than future motion. If all frames were annotated sequentially, both variants would eventually access the same temporal context, and this difference would likely diminish. Alternatively, this discrepancy could be explained by the difference in temporal offsets to the current frame in the *past* and *bidirectional* setting. Although both variants span the same overall temporal window (1 second), the distribution of frames differs: in the *past* variant, the prediction frame t is provided as input alongside reference frames with a maximum offset of 1 second, while in the *bidirectional* the largest offset between t and the reference frames is 0.5 seconds. For slowly moving objects, the smaller offset in the *bidirectional* variant may result in minimal apparent displacement, while the larger gap could provide more distinguishable motion cues. Further investigation is needed to confirm this hypothesis.

The overall mAP is notably lower than reported in our previous work,[10] which is likely a result of the increased difficulty of the evaluation dataset. This dataset includes challenging conditions: very small objects, complex backgrounds, and a moving recording platform. In addition, dead pixels in the IR footage often result

in false positives. These factors explain the precision drop in the PR-curves (Figure 6). While image registration helps in mitigating the effect of camera motion, even slight misalignments occasionally lead to false positives. Nevertheless, the model often demonstrates robustness and successfully detects very distant object even under camera motion (Figure 7). To further improve performance, additional air-to-ground training data should be incorporated. Preferably, this data is captured from a moving camera platform and preprocessed with image registration. By providing the registered frames as input to the detector, the model can learn to handle registration-induced artifacts and prevent resulting false positives.

The *past* variant of Temporal-YOLO reduces latency by avoiding future frames, making it suitable for real-time applications. In combination with the implementation of a Lucas-Kanade *optical flow*-based method, we are able to achieve low-latency small moving object detection. While *LightGlue* achieves higher mAP scores, its efficiency should be improved to facilitate real-time operation. We did not optimize all image registration methods for speed, since their detection performance was already comparable to the other two methods. Our goal was not to exhaustively benchmark all image registration techniques, but rather to identify methods that are sufficiently accurate and fast for Temporal-YOLO.

Currently, we do not analyze the relationship between camera motion and registration accuracy in detail. Although ground-truth platform motion annotations are available for part of the dataset (EDF STORE), they were not used in this study. Instead, we analyzed camera motion indirectly by examining the translation component of the transformation matrix produced by the *LightGlue*-based registration method (Figure 4). However, this measure is not always fully reliable since it depends on registration accuracy. Visual inspections of frame displacement and alignment confirmed that the dataset includes a wide range of motion intensities. Future work should analyze the limits of image registration effectiveness in relation to platform speed and altitude.

The proposed image registration approach relies on global transformations, which assume a planar background. This assumption holds reasonably well at high altitudes but becomes problematic at lower altitudes or in ground-to-ground scenarios, where parallax effects are more significant. Such effects can lead to local misalignments and, consequently, false positives or missed detections. Temporal-YOLO mitigates some of these issues because, unlike classical frame differencing methods, misaligned regions do not necessarily trigger detections. However, registration errors may still degrade performance, and it could be valuable to explore local image registration strategies or parallax filtering techniques to better handle depth variations.

Finally, integrating semantic information could further enhance small object detection performance, particularly in complex scenes where distinguishing objects from background is challenging. Semantic segmentation could be leveraged as an additional input modality to provide spatial context or applied as a post-processing step to refine detections.

In conclusion, we have demonstrated that image registration enables the alignment of frames from moving platforms, allowing Temporal-YOLO to effecctively exploit temporal information for small object detection. Leveraging past frames ensures the feasibility of real-time deployment.

ACKNOWLEDGMENTS

This work received funding from the the European Defence Fund through the project STORE (Shared daTabase for Optronics image Recognition and Evaluation), grant agreement № 101121405.

REFERENCES

[1] Tang, G., Ni, J., Zhao, Y., Gu, Y., and Cao, W., "A survey of object detection for uavs based on deep learning," *Remote Sensing* **16**(1), 149 (2023).

[2] Hua, W. and Chen, Q., "A survey of small object detection based on deep learning in aerial images," *Artificial Intelligence Review* **58**(6), 1–67 (2025).

[3] Cheng, Y., Lai, X., Xia, Y., and Zhou, J., "Infrared dim small target detection networks: A review," *Sensors* **24**(12), 3885 (2024).

[4] Wang, W., Xu, J., and Zhang, R., "Optimized small object detection in low resolution infrared images using super resolution and attention based feature fusion," *PLoS One* **20**(7), e0328223 (2025).

[5] Nikouei, M., Baroutian, B., Nabavi, S., Taraghi, F., Aghaei, A., Sajedi, A., and Moghaddam, M. E., "Small object detection: A comprehensive survey on challenges, techniques and real-world applications," *arXiv preprint arXiv:2503.20516* (2025).

[6] Chen, G., Wang, H., Chen, K., Li, Z., Member, S., Song, Z., Liu, Y., Chen, W., Knoll, A., and Member, S., "A Survey of the Four Pillars for Small Object Detection: Multiscale Representation , Contextual and Region Proposal," **52**(2), 936–953 (2022).

[7] Cheng, G., Yuan, X., Yao, X., Yan, K., Zeng, Q., Xie, X., and Han, J., "Towards large-scale small object detection: Survey and benchmarks," *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2023).

[8] Mirzaei, B., Nezamabadi-Pour, H., Raoof, A., and Derakhshani, R., "Small object detection and tracking: a comprehensive review," *Sensors* **23**(15), 6887 (2023).

[9] Sun, S., Mo, B., Xu, J., Li, D., Zhao, J., and Han, S., "Multi-yolov8: An infrared moving small object detection model based on yolov8 for air vehicle," *Neurocomputing* **588**, 127685 (2024).

[10] Leeuwen, M. V., Heslinga, F. G., Pereboom, W., and Baan, J., "Towards versatile small object detection with Temporal-YOLOv8," 1–12 (2024).

[11] Fischer, N. M., Kruithof, M. C., and Bouma, H., "Optimizing a neural network for detection of moving vehicles in video," **10441**, International Society for Optics and Photonics, SPIE (2017).

[12] Bosquet, B., Mucientes, M., and Brea, V. M., "STDnet-ST: Spatio-temporal ConvNet for small object detection," *Pattern Recognition* **116**, 107929 (2021).

[13] Hajizadeh, M., Sabokrou, M., and Rahmani, A., "STARNet: spatio-temporal aware recurrent network for efficient video object detection on embedded devices," *Machine Vision and Application* **35**(23), 1–11 (2024).

[14] Alqaysi, H., Fedorov, I., Qureshi, F. Z., and O'Nils, M., "A temporal boosted yolo-based model for birds detection around wind farms," *Journal of Imaging* **7**(11) (2021).

[15] Luesutthiviboon, S., de Croon, G. C. H. E., Altena, A. V. N., Snellen, M., and Voskuijl, M., "Bio-inspired enhancement for optical detection of drones using convolutional neural networks," in [*Artificial Intelligence for Security and Defence Applications*], **12742**, 127420F, International Society for Optics and Photonics, SPIE (2023).

[16] Redmon, J. and Farhadi, A., "Yolov3: An incremental improvement," *CoRR* **abs/1804.02767** (2018).

[17] Jocher, G., Chaurasia, A., and Qiu, J., "YOLO-v8 by Ultralytics," tech. rep. (2023).

[18] Corsel, C. W., van Lier, M., Kampmeijer, L., Boehrer, N., and Bakker, E. M., "Exploiting temporal context for tiny object detection," in [*2023 IEEE/CVF Winter Conference on Applications of Computer Vision Workshops (WACVW)*], 1–11 (2023).

[19] Garcia-Garcia, B., Bouwmans, T., and Silva, A. J. R., "Background subtraction in real applications: Challenges, current models and future directions," *Computer Science Review* **35**, 100204 (2020).

[20] Shi, B., Gu, W., and Sun, X., "Xdmom: A real-time moving object detection system based on a dual-spectrum camera," *Sensors* **22**(10), 3905 (2022).

[21] Yang, D., Bai, Z., and Zhang, J., "Infrared weak and small target detection based on top-hat filtering and multi-feature fuzzy decision-making," *Electronics* **11**(21), 3549 (2022).

[22] Elgammal, A., Harwood, D., and Davis, L., "Non-parametric model for background subtraction," in [*European conference on computer vision*], 751–767, Springer (2000).

[23] Xiao, J., Cheng, H., Sawhney, H., and Han, F., "Vehicle detection and tracking in wide field-of-view aerial video," in [*2010 IEEE computer society conference on computer vision and pattern recognition*], 679–684, IEEE (2010).

[24] Benezeth, Y., Jodoin, P.-M., Emile, B., Laurent, H., and Rosenberger, C., "Comparative study of background subtraction algorithms," *Journal of Electronic Imaging* **19**(3), 033003–033003 (2010).

[25] Bouwmans, T., "Traditional and recent approaches in background modeling for foreground detection: An overview," *Computer science review* **11**, 31–66 (2014).

[26] Girshick, R., Donahue, J., Darrell, T., and Malik, J., "Rich feature hierarchies for accurate object detection and semantic segmentation," in [*IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*], 580–587 (2014).

[27] Liu, S., Zeng, Z., Ren, T., Li, F., Zhang, H., Yang, J., Li, C., Yang, J., Su, H., Zhu, J., and Zhang, L., "Grounding DINO: Marrying DINO with grounded pre-training for open-set object detection," (2023).

[28] Redmon, J., Divvala, S., Girshick, R., and Farhadi, A., "You only look once: Unified, real-time object detection," in [*2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*], 779–788 (2016).

[29] Yang, X., Wang, G., Hu, W., Gao, J., Lin, S., Li, L., Gao, K., and Wang, Y., "Video tiny-object detection guided by the spatial-temporal motion information," in [*Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*], 3054–3063 (2023).

[30] Horn, B. K. and Schunck, B. G., "Determining optical flow," *Artificial intelligence* **17**(1-3), 185–203 (1981).

[31] Fan, L., Zhang, T., and Du, W., "Optical-flow-based framework to boost video object detection performance with object enhancement," *Expert Systems with Applications* **170**, 114544 (2021).

[32] Zhao, F., Wang, T., Shao, S., Zhang, E., and Lin, G., "Infrared moving small-target detection via spatiotemporal consistency of trajectory points," *IEEE Geoscience and Remote Sensing Letters* **17**(1), 122–126 (2020).

[33] Yan, P., Hou, R., Duan, X., Yue, C., Wang, X., and Cao, X., "Stdmanet: Spatio-temporal differential multiscale attention network for small moving infrared target detection," *IEEE transactions on geoscience and remote sensing* **61**, 1–16 (2023).

[34] Heslinga, F. G., Ruis, F. A., Ballan, L., van Leeuwen, M. C., Masini, B., van Woerden, J. E., den Hollander, R. J. M., Berndsen, M., Baan, J., Dijk, J., and Pereboom-Huizinga, W., "Leveraging temporal context in deep learning methodology for small object detection," (October), 17 (2023).

[35] van Lier, M., van Leeuwen, M., van Manen, B., Kampmeijer, L., and Boehrer, N., "Evaluation of spatio-temporal small object detection in real-world adverse weather conditions," in [*Proceedings of the Winter Conference on Applications of Computer Vision*], 844–855 (2025).

[36] Zhu, P., Wen, L., Du, D., Bian, X., Fan, H., Hu, Q., and Ling, H., "Detection and tracking meet drones challenge," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **44**(11), 7380–7399 (2021).

[37] Xia, G. S., Bai, X., Ding, J., and Zhu, Z., "Dota," (2025).

[38] Cheng, G., Yuan, X., Yao, X., Yan, K., Zeng, Q., Xie, X., and Han, J., "Towards Large-Scale Small Object Detection: Survey and Benchmarks," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **45**(11), 13467–13488 (2023).

[39] Guo, J., Xu, Z., Wu, L., Gao, F., Liu, W., and Wang, X., "Xs-vid: An extremely small video object detection dataset," *arXiv preprint arXiv:2407.18137* (2024).

[40] Zitova, B. and Flusser, J., "Image registration methods: a survey," *Image and vision computing* **21**(11), 977–1000 (2003).

[41] Brown, L. G., "A survey of image registration techniques," *ACM computing surveys (CSUR)* **24**(4), 325–376 (1992).

[42] Nag, S., "Image registration techniques: a survey," *arXiv preprint arXiv:1712.07540* (2017).

[43] Brownrigg, D. R., "The weighted median filter," *Communications of the ACM* **27**(8), 807–818 (1984).

[44] Brinkmann, R., [*The art and science of digital compositing: Techniques for visual effects, animation and motion graphics*], Morgan Kaufmann (2008).

[45] Davies, E. R., [*Machine vision: theory, algorithms, practicalities*], Elsevier (2004).

[46] Ko, S.-J. and Lee, Y. H., "Center weighted median filters and their applications to image enhancement," *IEEE transactions on circuits and systems* **38**(9), 984–993 (1991).

[47] Beauchemin, S. S. and Barron, J. L., "The computation of optical flow," *ACM computing surveys (CSUR)* **27**(3), 433–466 (1995).

[48] Lucas, B. D. and Kanade, T., "An iterative image registration technique with an application to stereo vision," in [*IJCAI'81: 7th international joint conference on Artificial intelligence*], **2**, 674–679 (1981).

[49] Evangelidis, G. D. and Psarakis, E. Z., "Parametric image alignment using enhanced correlation coefficient maximization," *IEEE transactions on pattern analysis and machine intelligence* **30**(10), 1858–1865 (2008).

[50] Lowe, D. G., "Object recognition from local scale-invariant features," in [*Proceedings of the seventh IEEE international conference on computer vision*], **2**, 1150–1157, Ieee (1999).

[51] Bay, H., Ess, A., Tuytelaars, T., and Van Gool, L., "Speeded-up robust features (surf)," *Computer vision and image understanding* **110**(3), 346–359 (2008).

[52] Rublee, E., Rabaud, V., Konolige, K., and Bradski, G., "Orb: An efficient alternative to sift or surf," in [*2011 International conference on computer vision*], 2564–2571, Ieee (2011).

[53] Jiang, H., Sun, D., Jampani, V., Yang, M.-H., Learned-Miller, E., and Kautz, J., "Super slomo: High quality estimation of multiple intermediate frames for video interpolation," in [*Proceedings of the IEEE conference on computer vision and pattern recognition*], 9000–9008 (2018).

[54] Poplavskiy, D., "The winning solution for the airborne object tracking challenge," *Airborne Object Tracking Challenge, GitHub* (2021).

[55] DeTone, D., Malisiewicz, T., and Rabinovich, A., "Superpoint: Self-supervised interest point detection and description," in [*Proceedings of the IEEE conference on computer vision and pattern recognition workshops*], 224–236 (2018).

[56] Sarlin, P.-E., DeTone, D., Malisiewicz, T., and Rabinovich, A., "Superglue: Learning feature matching with graph neural networks," in [*Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*], 4938–4947 (2020).

[57] Lindenberger, P., Sarlin, P.-E., and Pollefeys, M., "Lightglue: Local feature matching at light speed," in [*Proceedings of the IEEE/CVF international conference on computer vision*], 17627–17638 (2023).

[58] Tyszkiewicz, M., Fua, P., and Trulls, E., "Disk: Learning local features with policy gradient," *Advances in neural information processing systems* **33**, 14254–14265 (2020).

[59] Fischler, M. A. and Bolles, R. C., "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM* **24**(6), 381–395 (1981).

[60] Poplavskiy, D., "The winning solution for the airborne object tracking challenge," (2024).

[61] Bradski, G., "The OpenCV Library," *Dr. Dobb's Journal of Software Tools* (2000).