electronics

MDPI

Article

A Benchmark for UAV-View Natural Language-Guided Tracking

Hengyou Li, Xinyan Liu and Guorong Li *

School of Computer Science and Technology, University of Chinese Academy of Sciences, Beijing 100049, China; lihengyou21@mails.ucas.ac.cn (H.L.); liuxinyan19@mails.ucas.ac.cn (X.L.)
* Correspondence: liguorong@ucas.ac.cn; Tel.: +86-135-8181-6611

Abstract: We propose a new benchmark, UAVNLT (Unmanned Aerial Vehicle Natural Language Tracking), for the UAV-view natural language-guided tracking task. UAVNLT consists of videos taken from UAV cameras from four cities for vehicles on city roads. For each video, vehicles' bounding boxes, trajectories, and natural language are carefully annotated. Compared to the existing data sets, which are only annotated with bounding boxes, the natural language sentences in our data set can be more suitable for many application fields where humans take part in the system for that language, being not only more friendly for human–computer interaction but also capable of overcoming the appearance features' low uniqueness for tracking. We tested several existing methods on our new benchmarks and found that the performance of the existing methods was not satisfactory. To pave the way for future work, we propose a baseline method suitable for this task, achieving state-of-the-art performance. We believe our new data set and proposed baseline method will be helpful in many fields, such as smart city, smart transportation, vehicle management, etc.

Keywords: UAV; tracking; tracking by language

1. Introduction

With the advancement of industrial technology, unmanned aerial vehicles (UAVs) have become increasingly prevalent and have seen significant development across multiple fields [1–5]. Compared with traditional fixed cameras, UAVs offer unique advantages in the video field due to their distinctive high-altitude perspective, high mobility, and the ability to move vertically. UAVs can cover a wide area, possess flexible and variable postures, and have adjustable flying heights, making them uniquely valuable in security monitoring [6], facility inspection [7], aerial photography [8] and object detection [9,10]. Moreover, UAVs also perform well in robot arms [11], grippers [12], rescue and delivery [13], and so on. Therefore, in terms of single-object tracking, tracking carried out by UAVs is currently a hot research topic [14–16].

Despite the significant potential for research in UAV tracking, it still faces many challenges. Just like traditional single-object tracking, UAV tracking often starts by using a bounding box (bbox) to identify the tracking target in the first frame of the video. However, the tracking with the bbox setting suffers from two limitations. (i) Sometimes, the objects are hard to annotate with bounding boxes, as the UAV flies fast and objects can be very small from a UAV view. (ii) Only the appearance features of the target can be delineated in the bbox, leaving actions and contextual elements such as the environment undescribed. This limitation can lead to misjudgments in UAV videos, which often contain multiple similar objects.

Many researchers [17–19] have recently added natural language descriptions to single-object tracking videos, introducing the concept of tracking with the language specification. Research in this area can be divided into two main categories, based on the tasks that they address: Tracking by natural language (TNL) and vision-language tracking (VLT). The TNL system provides natural language descriptions for the target. Natural language is more convenient and natural for humans than bounding boxes, as it can provide details

check for updates

Citation: Li, H.; Liu, X.; Li, G. A Benchmark for UAV-View Natural Language-Guided Tracking. *Electronics* **2024**, *13*, 1706. https://doi.org/10.3390/electronics13091706

Academic Editor: Kostas Karpouzis

Received: 3 April 2024
Revised: 12 April 2024
Accepted: 25 April 2024
Published: 28 April 2024

about the object's category, color, actions, and more attributes. Moreover, natural language descriptions can identify the target to be tracked, avoiding the ambiguity that bounding boxes may introduce. The VLT combines natural language with bounding boxes, treating the natural language as auxiliary information to the bounding boxes. Therefore, we propose the incorporation of natural language into UAV tracking: namely, natural language-based UAV tracking.

Natural language-based UAV tracking effectively solves key issues in the field of UAV tracking: (1) Describing targets with natural language avoids the problem of fixed initialization in practical applications. Using natural language descriptions allows for accurate target localization, greatly enhancing the practicality of UAV tracking in real-time scenarios. (2) Natural language tracking can provide information on the target's movement direction and surrounding environment while describing the target's appearance. This approach helps to avoid the interference caused by numerous objects with similar appearances from a high-altitude perspective. Although various researchers have introduced several tracking-by-language data sets (OTB-Lang [17], LaSOT [18], TNL2K [19]), these primarily focus on video data that are readily available from real-world scenarios, with almost no coverage of the UAV video data. Moreover, the information provided in natural language descriptions in these data sets is insufficient and lacks a unified standard. In complex UAV tracking, these simplistic language descriptions often fail to accurately characterize the targets, limiting the development of natural language-based UAV tracking technologies.

For this work, we collected a UAV video sequence of 2000 videos named UAVNLT (Unmanned Aerial Vehicle Natural Language Tracking). These videos are composed of two parts. One part consists of data collected from existing UAV tracking data sets such as UAV123 [15], UAVDT [14], and VisDrone [16]. The other part consists of new videos we captured using a UAV in different cities. We offer 4K high-resolution video sequences, ensuring that the captured images are clear enough for target recognition, even when the UAV is at a higher flying altitude. We performed frame-by-frame intensive annotation of the tracking targets in each video sequence. Furthermore, we provide natural language descriptions of the tracking targets in the videos. Given the presence of numerous similar targets in UAV videos, we provide information about the category, color, and position within the video and the direction of movement of the targets in our natural language descriptions.

Additionally, we offer a simple yet effective baseline for the UAVNLT data set. This baseline comprises a visual grounding module responsible for locating objects based on natural language, an object tracker module that handles the object tracking, and a global–local switcher module that determines when to switch modules. This baseline provides a reference for future research. The data set and baseline method will be publicly available at https://github.com/Lich-King000/UAVNLT (accessed on 2 April 2024).

The contributions of this paper can be summarized in the following aspects:

- We propose the UAVNLT data set, providing bounding box and language annotations for 2000 video sequences. This data set aims to offer robust support for developing and testing natural language-based UAV tracking technologies.
- We propose a benchmark method that utilizes the global–local switcher module to alternate between the visual grounding and object tracker components flexibly. The method paves the way for future research.
- We conduct extensive experiments using state-of-the-art trackers on our proposed natural language-based UAV tracking data set, aiming to provide a comprehensive benchmark for future research.

2. Related Work

2.1. Tracking with Bounding Box

In object tracking, classical methods have been instrumental in tracking the position of target objects within a video sequence using a template annotated with a bounding box. Among these, Siamese network-based trackers are notable for using annotated templates to locate target objects by matching these templates against regions within video frames.

Techniques such as those introduced in SiamFC [20], SiamRPN++ [21], SiamRPN [22], and SiamFC++ [23] have demonstrated effectiveness in adapting to changes in appearance and motion, setting competitive standards in the field.

A novel approach within this domain is SiamPIN [24] (Siamese Parallel Interaction Network). It leverages a Siamese parallel interaction network to intricately capture global and local scene information, enhancing the distinction between the target and its surroundings. Furthermore, RTUSC [25] (Robust Tracking via Uncertainty-Aware Semantic Consistency) introduces an uncertainty-aware strategy for visual tracking, employing data-driven techniques to generate features cognizant of uncertainty, thereby achieving more robust tracking outcomes.

AES [26] proposes a simple and robust reliable memory model. In this method, an adaptive evaluation strategy (AES) is proposed to combine the confidence of the tracker predictions and the similarity distance between the current predicted result and the existing tracking results. Based on the reliable results of AES selection, we designed an active–frozen memory model to store reliable results.

Recent innovations have seen the advent of transformer-based trackers, such as TRTR [27] and TransT [28], which utilize encoder–decoder architectures equipped with self- and cross-attention mechanisms. These models, including STARK [29], MixFormer [30], and OSTrack [31], employ transformer-based techniques for tracking, enabling the capture of global feature dependencies through a unified framework for feature extraction and fusion.

Despite these advancements, bounding box-based tracking methods face challenges, including the need for human calibration and the potential for deviation from user habits. These limitations have inspired research into alternative strategies to refine tracking performance and overcome conventional methods' inherent constraints.

2.2. Tracking with Natural Language Description

Tracking objects in videos using natural language descriptions challenges models to not only extract pertinent image features but also comprehend and integrate language descriptions. This task demands a sophisticated blend of visual and linguistic understanding, along with the capability to fuse these modalities effectively in later stages. Research in this area can be divided into two main categories, based on the tasks they address: tracking by natural language (TNL) and vision-language tracking (VLT).

For the TNL task, the pioneering work TNLS [32] introduced the concept and developed a linguistic specification attention network, marking the beginning of targeted efforts in this direction. RTTNLD [33] adopted a tracking-by-detection strategy, using the language description to generate relevant proposals during the detection phase. Building on this, GTI [34] approached the problem by breaking down the TNL task into tracking, grounding, and integration sub-tasks, employing a grounding model for object location and the RT-score method for deciding when to switch to tracking for improved predictions. TNL2K [19] introduced Adaswitcher, optimizing the selection process between grounding and tracking model outputs, while CTRNLT [35] proposed a unified framework that integrates local and global search methods in cross-modal retrieval.

In the VLT task, various methodologies have been explored to leverage linguistic descriptions alongside visual information for enhanced tracking. SNLT [36] utilizes language descriptions as convolutional kernels for feature aggregation, whereas DAT [37] (Describe and Attend to Track) focuses on generating proposals driven by combined visual and linguistic cues. VLTTT [38] developed ModaMixer for a unified approach to vision-language representation learning. OVLM [39] (One-stream Vision-Language Memory Network for Object Tracking) introduces a model that augments visual features with linguistic inputs, and MMtrack [40] re-conceptualizes vision–language tracking by treating it as a token generation task, combining language and bounding box information into a cohesive model.

Despite the advancements in both TNL and VLT tasks as highlighted by works such as [19,32–35], a gap remains in optimizing these methods specifically for the TNL chal-

lenge. Our approach leverages the CLIP model's robust feature extraction and alignment capabilities to handle complex scenarios, aiming to set a new benchmark in the field.

2.3. UAV Video Data Sets

The utilization of UAVs for visual tracking has garnered significant attention in recent years, leading to the development and employment of various UAV video data sets. These data sets serve as benchmarks to evaluate and enhance the performance of tracking algorithms under diverse conditions and scenarios specific to aerial footage. Notable data sets in this domain include VisDrone, UAV123, UAVDT, and UAV20L, each offering unique challenges and characteristics.

VisDrone, introduced by Zhu et al. [41], comprises images and video sequences captured over various urban and rural areas. It is designed to evaluate object detection, object tracking, and single-object tracking in drone-based surveillance. This data set stands out due to its diversity in scenarios, including varying weather conditions, lighting, and object densities, making it a comprehensive benchmark for UAV-based vision tasks.

UAV123, presented by Mueller et al. [15], consists of 123 video sequences totaling more than 110K frames. This data set emphasizes low-altitude UAV operations, featuring a wide range of real-world scenarios that challenge both the detection and tracking algorithms due to the high dynamics of the UAVs and the small size of the objects being tracked.

UAVDT, by Du et al. [14], focuses on object detection and tracking, offering a rich set of aerial videos. It includes attributes such as scale variation, occlusion, and illumination changes, specifically tailored for evaluating algorithms in urban environments. The UAVDT data set is instrumental in understanding how environmental factors influence tracking and detection performance in UAV footage.

DTB70, introduced by Li et al. [42], is a data set specifically designed for evaluating the performance of UAV tracking algorithms. It comprises 70 high-quality video sequences, encompassing challenges such as fast motion, scale changes, occlusion, and low resolution. This diversity makes DTB70 an essential benchmark for testing the robustness of UAV tracking algorithms under various dynamic and complex conditions.

UAVTrack112, proposed by Fu et al. [43], is another valuable data set for the UAV-tracking community. It includes 112 video sequences with over 100,000 frames, covering various scenarios, including urban landscapes, vehicles, and people. The data set is characterized by its emphasis on small object tracking, a common challenge in UAV surveillance and monitoring applications. The comprehensive range of scenarios of UAVTrack112, from densely populated areas to challenging weather conditions, makes it a robust benchmark for advancing UAV tracking technologies.

Lastly, UAV20L, a subset of the UAV123 data set, is specifically curated by Mueller et al. [15] to focus on 20 long-duration sequences. This subset challenges tracking algorithms over extended temporal spans, testing their endurance and ability to maintain consistent tracking performance over time.

3. Proposed Benchmark

Our benchmark aims to benefit research focused on applying natural language-based tracking technologies in the UAV domain by integrating natural language tracking with UAV tracking technologies. To this end, we introduce the UAVNLT data set, which combines detailed natural language descriptions with high-quality UAV video data to foster the advancement and widespread application of natural language-based UAV tracking technologies. In the following sections, we will provide a detailed overview of the composition and annotation details of the UAVNLT data set.

3.1. Data Collection and Annotation

The UAVDT data set comprises 6 h of original video material, from which 2000 video sequences are extracted, totaling approximately 900,000 frames. The number of frames per video sequence ranges from 117 to 1461. The construction of the data set involved two main

steps. First, 280 video sequences were selected from existing UAV tracking data sets, including UAV123 [15], UAVDT [14], and VisDrone2019-SOT [16], which already provide tracking data based on bounding boxes. To these sequences, we added natural language descriptions of the tracking targets to enrich the UAV data sets. Second, an additional 1720 video sequences were collected through UAV filming over traffic arteries in various cities. All videos were shot and recorded at 30 frames per second (fps) with a resolution of 3840 × 2160 (4K), ensuring high image clarity.

As illustrated in Figure 1, this paper introduces a novel UAV video tracking data set, UAVNLT, enriched with natural language descriptions of tracking targets. The UAVNLT data set includes annotations for various types of vehicles, such as cars, SUVs, motorcycles, etc. In the UAVNLT, intensive bounding box annotations are made on videos captured by UAVs, ensuring the precise annotation of every tracking target in each video frame. For each frame's annotation, the left upper corner point (x_1, y_1), width (w), and height (h) of the target's bounding box are utilized as the ground truth, recorded in the format $[x_1, y_1, w, h]$. Furthermore, detailed natural language descriptions are provided for each tracking target in the videos. These descriptions include information such as the category, color, specific location, and direction of movement, offering a comprehensive data foundation for the application and research of natural language-driven UAV tracking technologies.

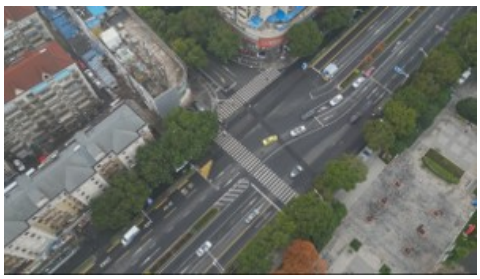
In order to enhance the diversity of scenarios in our UAVNLT data set as shown in Figure 2, we selected four cities with distinct geographical environments for video shooting: Nanjing, Qinhuangdao, Anyang, and Taizhou. With the help of UAV123, UAVDT, and VisDrone-2019SOT, the locations of UAVNLT not only vary in their geographic features but also allow us to capture the essence of different seasons—spring, summer, fall, and winter—thus ensuring a broad temporal span in our data set. During the process of bounding box annotation, three experts were invited to carry out detailed and intensive data annotation, aiming to ensure the accuracy and reliability of the data. All annotation tasks were performed in high resolution (4k) to guarantee the highest clarity of the collected data. In order to minimize annotation errors, another expert reviewed and proofread each frame's annotation to ensure absolute accuracy. Any errors detected were corrected and supplemented by the expert, followed by a final check to ensure the high quality of the bounding box annotations.

(**a**) The gray car traveling from right to left on the right side of the picture.

Figure 1. *Cont.*

(**b**) The black car traveling from right to left on the right side of the picture.

Figure 1. Presentation of the UAVNLT (Unmanned Aerial Vehicle Natural Language Tracking) data set in this paper. In addition to providing bounding box annotations for UAV tracking targets, we also detail the category, color, location, and direction of movement of the tracking targets using natural language. The red box represents the groundtruth label.



(**a**) Nanjing: Fall.

(**b**) Qinhuangdao: Winter.

(**c**) Wenzhou: Spring.

(**d**) Other: Summer.

Figure 2. The figures illustrate the diversity of video data in our data set, captured across different cities and seasons. Sub-figures (**a**–**c**) showcase video data from Nanjing in fall, Qinhuangdao in winter, and Wenzhou in spring, respectively. Sub-figure (**d**) represents additional video data acquired from UAV123 [15].

Using densely annotated bbox and natural language description, we successfully constructed a UAV-view natural language-guided tracking data set named UAVNLT. This data set effectively addresses the gap in natural language tracking within the UAV domain, contributing to advancing the practical application of natural language-based tracking technologies in UAV applications.

3.2. Comparison with Existing Data Sets

While there are some data sets available in both the UAV tracking [14–16,42,43] and tracking by language fields [17–19], there are still some shortcomings when it comes to joint UAV and natural language tracking. As shown in Table 1, UAV-tracking data sets such as UAV123, UAV20L, UAV123@10FPS, UAVDT, and VisDrone2019-SOT include a variety of weather and traffic scenarios, providing diverse training data for object tracking from a UAV perspective. However, their shortcomings in terms of data set size pose a significant barrier to further development in the field. Moreover, the relatively low video resolution offered by these data sets limits the possibility of developing and testing tracking algorithms in high-resolution scenarios. As for TNL data sets [17–19], the TNL data set offers a large-scale single-object tracking data set with a wealth of natural language annotations, significantly advancing the TNL task. However, the lack of UAV-related video data in the TNL data set hinders the development of natural language-based UAV tracking. Additionally, the absence of a uniform standard for natural language annotation could impede accurate target identification. Similarly, the low resolution of the videos provided by these data sets is a common flaw shared with UAV tracking data sets.

Table 1. Comparison with other UAV tracking data sets and TNL data sets. # denotes the number of corresponding items, and NL denotes natural language annotations. The ✗ denotes without NL annotations, while the ✓ indicates data set with NL annotations.

Data Sets	Resolution	#Video	#Min	#Max	#Total	NL
UAV123	720p	123	109	3085	113 K	✗
UAV20L	720p	20	1717	5527	59 K	✗
UAV123@10FPS	720p	123	36	1362	38 K	✗
UAVDT	540p	100	82	2969	80 K	✗
VisDrone2019-SOT	756p	132	329	2789	109 K	✗
DTB	720p	70	68	699	15 K	✗
UAVTrack112	400p	112	-	1000	-	✗
OTB-99	432p	99	71	3872	59 K	✓
LaSOT	360p	1400	1000	11,397	3.52 M	✓
TNL2K	720p	2000	21	18,488	1.24 M	✓
UAVNLT	2160p	2000	117	1461	900 K	✓

Our proposed UAVNLT data set specifically addresses the challenges within the current UAV tracking and tracking using language technology domains. By offering 2000 videos totaling approximately 90,000 frames of UAV tracking data, this data set significantly expands the scale of training data for UAV tracking technology, providing robust support for its development. Moreover, the standardized natural language annotations for UAV videos—including detailed descriptions of the tracking target's category, color, specific location, and direction of movement—facilitate more accurate and convenient target identification and localization in complex scenarios. The introduction of natural language improves the tracking initialization process and overcomes the limitations encountered in practical applications of UAV tracking. Most importantly, all videos are provided in 4K ultra-high resolution, contributing to the progression of UAV tracking technology toward higher definition and practicality.

3.3. Data Set Analysis

In order to explore the characteristics of the UAVNLT data set further, we conducted a detailed statistical analysis of the data set. As shown in Figure 3, we analyzed the following aspects:

1. Object aspect: As shown in Figure 3a, the distribution of the aspect ratios of objects is displayed. Most objects have an aspect ratio of 0.5 to 1.5, indicating that smaller vehicles, such as cars or SUVs, are common in the UAVNLT data set. Additionally,

a portion of objects have an aspect ratio of around 2, signifying the presence of larger vehicles, such as buses and trucks, in the data set.

2. Object scale: Figure 3b illustrates the distribution of object sizes, defined as the proportion of the video frame occupied by the object. It can be observed that most objects are of a smaller scale, primarily concentrated below 0.05. This is mainly because our videos are mainly shot from heights of greater than 70 m, thus including many small-sized targets and bringing new challenges to UAV tracking.
3. Object position: As shown in Figure 3c, the heatmap illustrates the spatial distribution of target bounding boxes within video frames. The bright areas indicate locations with a high frequency of target appearances. The primary directions of target movement are either upwards, downwards, left, or right, a characteristic that indicates our data set was mainly captured in areas such as intersections and other traffic hubs, capturing typical movement patterns within these regions.
4. Video length: Figure 3d presents the distribution of tracking sequence lengths. Most of the videos are centered around the 400 to 800 frame range, but a few videos are shorter than 200 or longer than 1000 frames. The distribution indicates that our data set offers a wide range of sequence lengths, from short to medium, which is beneficial for evaluating the performance of tracking algorithms across different temporal spans.

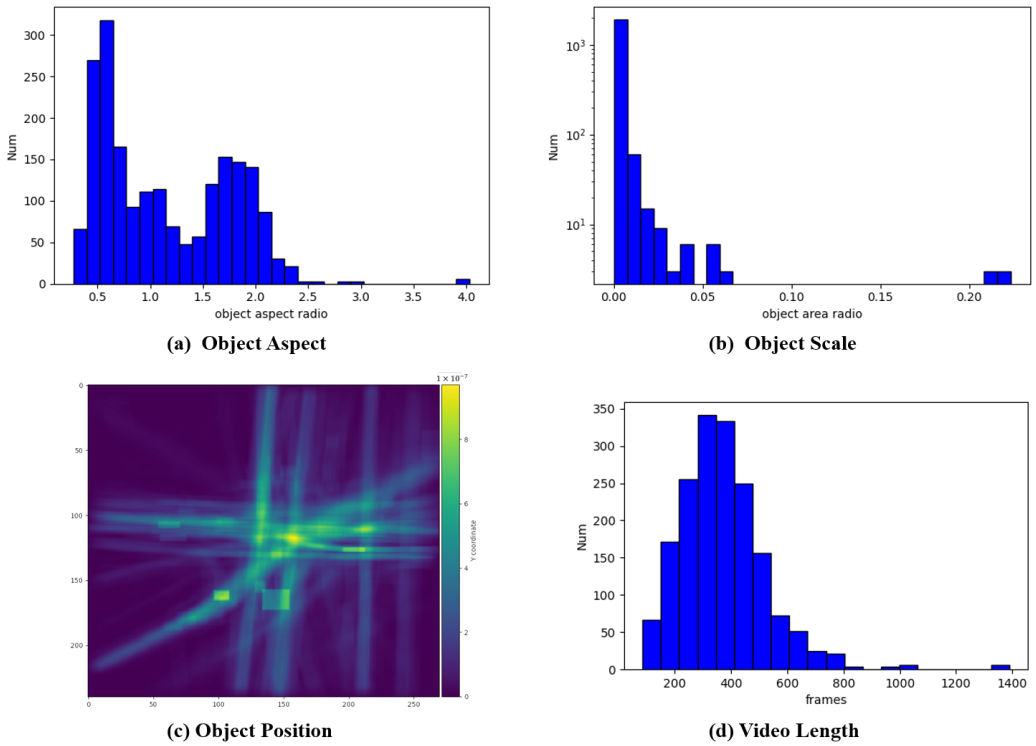


Figure 3. Sub-figures (**a**–**c**) illustrate the distribution of tracking targets in the video sequences concerning aspect ratio, scale ratio, and positional distribution. Sub-figure (**d**) details the distribution of video lengths within the UAVNLT data set.

Furthermore, when providing natural language descriptions for videos, our data set utilizes a standardized language pattern, specifically detailing the object's type, color, location, and direction of movement within a single sentence. For instance, "The blue truck at the bottom traveling from bottom to top". Thus, as shown in Figure 4, our proposed UAVNLT contains 65 English words that detail the object's category, color, location, and direction of movement.

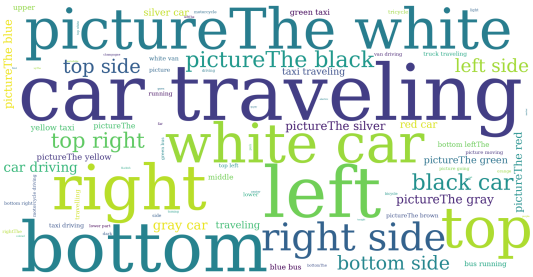


Figure 4. Word cloud for UAVNLT data set descriptions.

In summary, the characteristics of the UAVNLT data set benefit the training and evaluation of UAV tracking algorithms across various scenarios. It encompasses targets ranging from small to medium sizes with diverse aspect ratios. Moreover, the data set includes sequences from short to relatively long lengths, allowing researchers to test the persistence of tracking algorithms. The high-altitude footage results in a wealth of small-sized targets, expanding the applicability of UAVs and presenting new challenges to the field of UAV tracking. Moreover, we provide standardized natural language annotations, which enable a clearer description of targets in natural language-based UAV tracking tasks.

3.4. Evaluation Protocols

In order to provide a large-scale UAV tracking training and test set simultaneously, we divided the 2000 UAV tracking video sequences into two parts: 1300 videos for training and 700 videos for testing. In order to maintain diversity as much as possible, we proportionally selected UAV video sequences based on different filming locations and scenarios, ensuring that the test set includes all locations and scenarios in the data set. Furthermore, we ensured an even distribution of the three seasons across the training and test sets as much as possible.

Regarding the evaluation metrics, we maintained consistency with previous studies [17–19]. Specifically, we utilized the success plot, represented by the area under the curve (AUC), and precision.

AUC is the Area Under Curve (AUC) of the success plot. Specifically, dividing threshold t into multiple equidistant points $t_0, t_1, ..., t_n$ across the range from 0 to 1. For each threshold t_i, compute the Intersection over Union (IoU) between the bounding box and the ground truth. Count the frames where the IoU is greater than or equal to the threshold t_i as a proportion of the total number of frames, yielding the accuracy ratio $S(t_i)$ at that threshold. By calculating the accuracy ratio at various thresholds, the success plot can be generated with thresholds on the horizontal axis and the accuracy ratio on the vertical axis. AUC is represented by the following formula:

$$\text{AUC} = \sum_{i=0}^{n-1} \frac{S(t_i) + S(t_{i+1})}{2} \times (t_{i+1} - t_i), \tag{1}$$

where t_i denotes the discrete thresholds, and $S(t_i)$ represents the accuracy ratio at threshold t_i.

Precision typically refers to the proportion of frames in which the average Euclidean distance between the center of the tracking box and the center of the true target box is less than a certain threshold (for example, 20 pixels). It is an intuitive metric used to measure the accuracy of tracking. The calculation formula is as follows:

$$\text{Precision} = \frac{1}{N} \sum_{i=1}^{N} \mathbb{I}\Big(\| p_{\text{pred},i} - p_{\text{gt},i} \|_2 < \text{threshold} \Big) \tag{2}$$

where $\mathbb{I}(\cdot)$ is the indicator function, which is 1 when the condition inside is met and 0 otherwise; $\|\cdot\|_2$ denotes the Euclidean distance; $p_{\text{pred},i}$ and $p_{\text{gt},i}$ represent the center coordinates of the tracking box and the true target box for the ith frame, respectively; and threshold is the predefined distance threshold.

4. Method

The UAVNLT task aims to generate bounding boxes to track the designated target across each frame in a video sequence, leveraging the descriptions provided in natural language. To elaborate, for a given sequence of video frames denoted as $\{I_1, I_2, \ldots, I_n\}$ accompanied by a textual description L of the target, the UAVNLT approach is tasked with forecasting the bounding boxes that encapsulate the target in every frame, represented as $\{o_1, o_2, \ldots, o_n\}$:

$$\{o_1, o_2, \ldots, o_n\} = UAVNLT(\{I_1, I_2, \ldots, I_n\}, L) \tag{3}$$

In this study, we propose a simple yet effective baseline for the UAVNLT task, providing a reference for future research. The framework introduces an integrated model consisting of a visual grounding module for target location based on natural language, an object tracker for subsequent tracking and a global–local switcher for switching between the two based on spatio-temporal and language features. The framework is shown in Figure 5. In the following sections, we will introduce the visual grounding module, the object tracker, and the global–local switcher module.

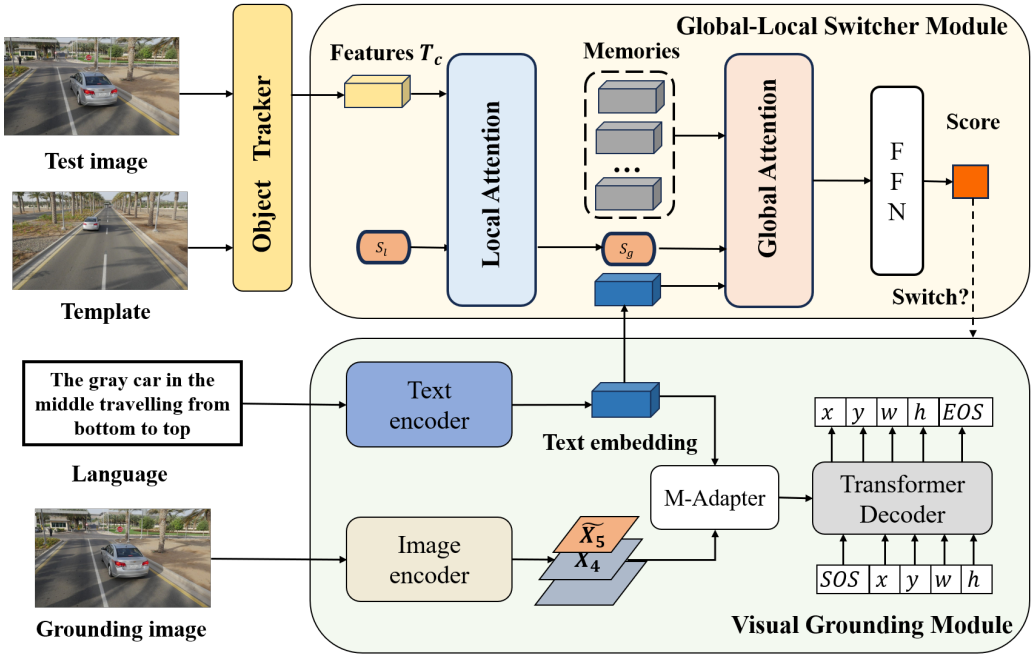


Figure 5. The overall framework of our method. It contains three modules: (1) The visual grounding module, which locates the target based on natural language descriptions. For the first frame of the video, the visual grounding module locates the target and crops it to serve as a template. (2) The object tracker, which utilizes the template to perform tracking on subsequent images. (3) The global–local switcher module evaluates the tracking results by integrating spatio-temporal and linguistic features to assign a score. Moreover, based on the score, it chooses whether to switch modules.

4.1. Visual Grounding Module

The visual grounding module is structured to employ a pre-trained CLIP model, incorporating both image and text encoders, to pinpoint the target's location within a specified frame. As depicted in Figure 5, this module comprises the CLIP image and text encoders, a multi-modal feature adapter known as M-Adapter, and an auto-regressive head that is responsible for predicting the target's position.

This way, we adopt the transformer-based CLIP text encoder for natural language processing. Given a language sentence L, the CLIP text encoder first tokenizes the language description L to obtain language tokens. Subsequently, it adds positional embeddings to these tokens, incorporating information about the position of each token within the sentence. Finally, the tokens are fed into a transformer encoder, resulting in the language embeddings F_l.

In the image encoding sector, we employed the architecture of the CLIP image encoder. We selected ResNet50 [44] as the backbone architecture to balance the processing speed and computational efficiency. When processing the input image I_z, the image encoder is primarily responsible for extracting spatial features from the image. Inspired by the design philosophy of the feature pyramid network (FPN) [45] to capture more refined multi-scale spatial features, we further integrated an FPN structure. For the last three feature maps generated by the FPN, $\{z_3, z_4, z_5\}$, we performed up-sampling and conducted a deep feature fusion, aiming to optimize the effectiveness of the final feature representation:

$$F_z = FPN(z_3, z_4, z_5) \tag{4}$$

The primary function of the visual grounding module is to utilize input from both image and language modalities to achieve cross-modal localization within the image. Therefore, this module must possess excellent multi-modal semantic alignment capabilities to accurately understand and match the relationships between image content and language descriptions. In the transformer decoder [46], the cross-attention mechanism facilitates the effective interaction of multi-modal information, thereby significantly enhancing the model's multi-modal semantic alignment capabilities. Therefore, we employed a standard transformer decoder as our multi-modal fusion model, which we refer to as M-Adapter. M-Adapter utilizes cross-attention to process and integrate information from image and language inputs, incorporating the language priors into the visual features.

More specifically, we input the image features F_z obtained from the image encoder and FPN, and the language features F_l into the M-Adapter. In this setup, F_z serves as the query, while F_l acts as both the key and value, facilitating effective interaction and fusion between image and language features to obtain fusion features F_M. This design enables M-Adapter to precisely locate and interpret image content while retaining language features, enhancing multi-modal semantic alignment capabilities:

$$F_M = M_{Adapter}(q = F_z, k = F_l, v = F_l) \tag{5}$$

Inspired by the Pix2Seq [47] model, we propose an autoregressive transformer decoder as our prediction head named ARhead. This autoregressive model treats tracking as a task of interpreting coordinate sequences, where the prediction of the current state is influenced by the previous state, forming a sequence decoding process. Compared to traditional template matching-based trackers, this approach is more straightforward, eliminating the need for complex customized localization heads and post-processing steps. The fused features F_M and query embeddings q_h are fed into the transformer decoder, aiming to predict the sequence of target bounding boxes. After the predictions are completed, these sequences are input into a feed-forward network (FFN) to predict the final coordinates O_z:

$$O_z = ARhead(q = q_h, k = F_M, v = F_M) \tag{6}$$

4.2. Object Tracker

While the visual grounding model can achieve multi-modal alignment and locate the target, solely relying on the grounding model may not be sufficient due to potential interference from global background information. Hence, we also employed a traditional object tracker. The object tracker uses the result located using the visual grounding module in the first frame of the video as a template, then begins tracking the target in subsequent video

frames. Considering both model performance and tracking speed, we used MixVIT [30] as our tracking module.

4.3. Global–Local Switcher Module

The visual grounding model locates the target from a global linguistic perspective, while the object tracker tracks the target using local image spatial features. Combining both is essential for effectively completing tracking based on natural language descriptions. We propose the global–local switcher model to facilitate a more effective switch between global and local models. This model is designed to integrate the advantages of global grounding and local tracking, ensuring that the tracking process is accurate and contextually relevant to the natural language input.

As shown in Figure 5, our proposed global–local switcher module comprises three components: local attention, global attention, and FFN. We introduce a learnable token S_l, which initially learns the local image features of the test image through the local attention module. Subsequently, it learns the historical temporal features from memory and global features from language by using the global attention module. Finally, S_g predicts the tracking score S of the tracking image via the FFN. Based on the score S and a predetermined threshold, we decide whether to switch the tracking module to the grounding module.

More specifically, after the object tracker completes tracking the target, we collect the final feature map of image features f_c, along with the tracking result $B = [x, y, w, h]$. Subsequently, we apply the ROI pooling [48] algorithm to crop f_c based on the position and size of B, thereby obtaining a token T_c, corresponding to the specific region of the test image targeted by the tracking result. Subsequently, we use a learnable score S_l as the query to compute self-attention with T_c, thereby capturing the local appearance features in the current prediction result.

In the global attention module, we employ memory techniques to enhance the model's efficiency in utilizing historical information. We established a memory that stores a set of ROI pooling features from historical frames $T_H = \{T_{n_1}, T_{n_2}, \cdots, T_{n_h}\}$. These features represent the appearance information of the target captured at different time points, providing the model with rich temporal context information. In addition to temporal features, natural language features also serve as a crucial source of global feature information. Therefore, we incorporate the natural language embeddings F_l into the global attention module to further provide semantic guidance. Through the global attention mechanism, the token S_g, carrying local feature information of the test image, can further integrate the historical temporal features and semantic features provided by natural language.

Finally, a token that integrates local image features with global temporal and language features is input into an FFN to predict the quality score S of the object tracker's result on the test image. This score reflects the accuracy and reliability of the tracking result, providing a basis for determining whether to switch to the grounding module:

$$S = FFN(GAM(LAM(S_l, T_c), [T_H, F_l])) \tag{7}$$

In the end, if S is below a set threshold, we activate the visual grounding module to localize the test image. This approach allows for corrections using the global grounding model when local tracking proves inaccurate, thereby enhancing the accuracy and robustness of tracking. After this process, the object tracker resumes its tracking task.

4.4. Implementation Details

In the visual grounding module, we utilized the pre-trained models provided by CLIP [49] as the image and text encoders. Initially, the images were resized to 224 $\times$ 224 pixels before being input into the image encoder. Subsequently, the features extracted from the feature maps are $\{z_3 \in \mathbb{R}^{49\times2048}, z_4 \in \mathbb{R}^{196\times1024}, z_5 \in \mathbb{R}^{784\times512}\}$. Through the up-sampling process, we obtained multi-scale features F_z. Meanwhile, the text encoder is responsible for extracting features from natural language, obtaining the language embedding $F_l \in \mathbb{R}^{1024}$. Subsequently, we input the image features F_z and language features F_l into the M-Adapter for

feature fusion. Through this fusion process, we obtained the fused feature F_M with dimensions $\mathbb{R}^{49\times768}$. Lastly, the F_M was input into the autoregressive head (ARhead), sequentially predicting the target's coordinates $[x_g, y_g, w_g, h_g]$.

In the object tracker, once we obtained the predicted co-ordinates $[x_g, y_g, w_g, h_g]$, for the first frame, we cropped the image by enlarging it 1.5 times around the center point of these co-ordinates to create the tracking template. Both the template and the test image were resized to 224 × 224 pixels. After the object tracker completes tracking, it outputs the target coordinates $[x_t, y_t, w_t, h_t]$.

In the global–local switcher module, we set the learnable token S_l as $S_l \in \mathbb{R}^{1\times768}$. The features obtained from the object tracker are $T_c \in \mathbb{R}^{36\times768}$. After processing using local attention, we obtained $S_g \in \mathbb{R}^{1\times768}$. In the global attention phase, we concatenated the historical temporal features to form $T_H \in \mathbb{R}^{9\times768}$. The natural language feature $F_l \in \mathbb{R}^{1\times768}$ was also input into the global attention module. After processing using global attention and an FFN, we ultimately derived a quality score $S \in \mathbb{R}^{1\times1}$. This score reflects the quality of the tracking result. We set the threshold for the quality score at 0.3. When S is lower than 0.3, it indicates that the object tracker's tracking quality on the test image is poor. Thus, we would then activate the visual grounding model to correct and enhance the tracking process.

During the training phase, we adopted a strategy of training the visual grounding module and the global–local switcher module separately. In the visual grounding module, we implemented a simple cross-entropy loss:

$$\mathcal{L}_g = -\sum_{i=1}^{4} \omega_i log P(T_i|F_M, Q_{1:i-1}) \tag{8}$$

where Q is the input queries of the decoder, and T is the output bounding box $[x_g, y_g, w_g, h_g]$ of the decoder. We set the initial learning rate for the grounding model at 1e-5. In order to preserve the weights of the pre-trained model, we adjusted the learning rate for the image encoder to one-tenth of that of the other parameters. The text encoder remained frozen during training to preserve its pre-trained weights.

In the global–local switcher module, we set the memory length to 9, and the switching threshold θ was 0.3. For the memory method, a first-in-first-out (FIFO) approach was utilized. This approach means that the oldest historical features are replaced when the number of stored historical features exceeds the memory capacity. In the training phase, for each image in the training data set, we exploited the intersection over the union (IoU) score between the tracking result and the ground truth as the training label. Moreover, we employed the smooth L_1 loss for training our global–local switcher module. The learning rate in this phrase was set to 1×10^{-5}.

5. Experiments

5.1. Performance on UAVNLT

In order to further demonstrate the value of our proposed UAVNLT data set, we conducted tests with many advanced trackers on this data set. These tests aimed to evaluate the performance of various tracking algorithms in handling UAV perspective target tracking tasks based on natural language descriptions. Moreover, to facilitate a comprehensive and quantitative comparison for the UAVNLT data set, we also incorporated large-scale tracking from language data sets such as LaSOT and TNL2K. This approach enabled a broader evaluation of the performance of object trackers across different data sets, highlighting the unique challenges and opportunities for improvement that UAVNLT presents in the context of language-driven tracking tasks. We categorized the tracking tasks on the UAVNLT data set into three types: tracking by bbox, tracking by language, and tracking by joint language and bbox. For tracking by bbox, the initialization only utilizes the manually annotated bbox from the first frame. In tracking by language, the initialization relies solely on the natural language description of the first frame. As for tracking by joint language and bbox,

the first frame is initialized using the bbox to activate the object tracker. When the module switches to the visual grounding module, it locates the target based on the natural language description.

Tracking by bbox: In order to further evaluate the UAVNLT data set, we first selected object trackers based on bounding boxes for testing. For a comprehensive comparison, we chose several trackers based on CNN, including ATOM [50], DiMP [51], and KeepTrack [52]. Additionally, we considered object-tracking algorithms based on the transformer architecture, such as STARK [29], MixFormer [30], ODTrack [53], and ARTrack [54]. Based on the performance of these trackers on the UAVNLT data set, we can infer that, when compared to other data sets, existing object trackers still have room for improvement in the comprehensive metric of AUC. This indicates that our UAV data set provides new object-tracking challenges, suggesting further research and development opportunities to enhance tracking algorithms.

Tracking by natural language: As for the tracking methods initiated by natural language, as most existing methods have not been made open source, we show the results of TNL2K-1 [19] and CTRNLT [35] on three data sets. Furthermore, to facilitate extensive comparisons on the UAVNLT data set, we selected state-of-the-art object trackers such as STARK [29], MixFormer [30], ARtrack [54], and ODtrack [53], denoted by an asterisk (*) in Table 2. The tracking templates these trackers use are derived from the localization of the video's first frame, accomplished through our proposed visual grounding approach. While existing natural language-based tracking methods perform well on other data sets, their effectiveness on the UAVNLT data set requires improvement. In the LaSOT and TNL2K data sets, objects within the videos are generally larger, and the tracking scenarios tend to be simpler. However, in the UAVNLT data set, objects viewed from a drone perspective are smaller and present a higher difficulty level. State-of-the-art trackers achieve good results based on bounding boxes, but the performance is not as satisfactory when tracking with templates located by the visual grounding model. This indicates that using natural language to locate targets in the UAVNLT data set poses significant challenges.

Table 2. The performance of state-of-the-art tracking models on the UAVNLT (Unmanned Aerial Vehicle Natural Language Tracking), LaSOT, and TNL2K data sets. * indicates that the template in the first frame for these trackers is predicted by our visual grounding module.

Methods	Source	Initialization	UAVNLT		LaSOT		TNL2K	
			AUC	PRE	AUC	PRE	AUC	PRE
ATOM [50]	CVPR2019	BB	0.429	0.587	0.510	0.510	0.390	0.400
DIMP [51]	CVPR2019	BB	0.462	0.611	0.569	-	-	-
KeepTrack [52]	ICCV2019	BB	0.483	0.643	0.671	0.702	-	-
STARK [29]	ICCV2021	BB	0.623	0.811	0.671	0.712	-	-
MixFormer [30]	CVPR2022	BB	0.666	0.894	0.692	0.747	0.552	0.558
ODtrack [53]	AAAI2024	BB	0.657	0.919	0.731	0.757	0.609	0.723
ARtrack [54]	CVPR2024	BB	0.689	0.731	0.803	0.747	0.603	0.766
TNL2K-1 [19]	CVPR2021	NL	-	-	0.510	0.490	0.110	0.060
CTRNLT [35]	CVPR2022	NL	-	-	0.520	0.510	0.140	0.090
STARK * [29]	ICCV2021	NL	0.062	0.071	-	-	-	-
MixFormer * [30]	CVPR2022	NL	0.077	0.094	-	-	-	-
ARtrack * [54]	CVPR2024	NL	0.097	0.102	-	-	-	-
ODtrack * [53]	AAAI2024	NL	0.103	0.113	-	-	-	-
Ours	-	NL	0.105	0.105	0.522	0.513	0.461	0.422
SNLT [36]	CVPR2021	NL + BB	0.234	0.372	0.540	0.576	0.276	0.419
VLTTT [38]	NIPS2022	NL + BB	0.452	0.418	0.673	0.721	0.531	0.533
Ours	-	NL + BB	0.467	0.421	0.613	0.688	0.547	0.529

Tracking by joint natural language and bbox: As tracking methods that are initialized by both natural language (NL) and bounding box (BB), SNLT [36] and VLTTT [38] were

selected for comparative analysis. The findings indicate that the initialization using NL+BB leads to significant performance improvements, compared to relying solely on NL. This highlights the sensitivity of current tracking algorithms to bounding boxes. Compared to initialization using only natural language, our method achieves better results when combining natural language with bounding boxes. This suggests that existing methods are more sensitive to the bounding box; once the target is accurately located using the bounding box, subsequent tracking can be performed more effectively.

Furthermore, to qualitatively demonstrate the efficacy of our proposed method, we provide visual comparisons of the UAVNLT data set as shown in Figure 6. In the visualizations, in the simpler scenario depicted in the first row, our methods based on NL or NL+BB achieve good results, accurately locating the moving white vehicle. The second row shows a more complex scenario. In this scenario, the method based solely on NL exhibits some inaccuracies. However, our approach of integrating NL with BB still precisely locates the target.

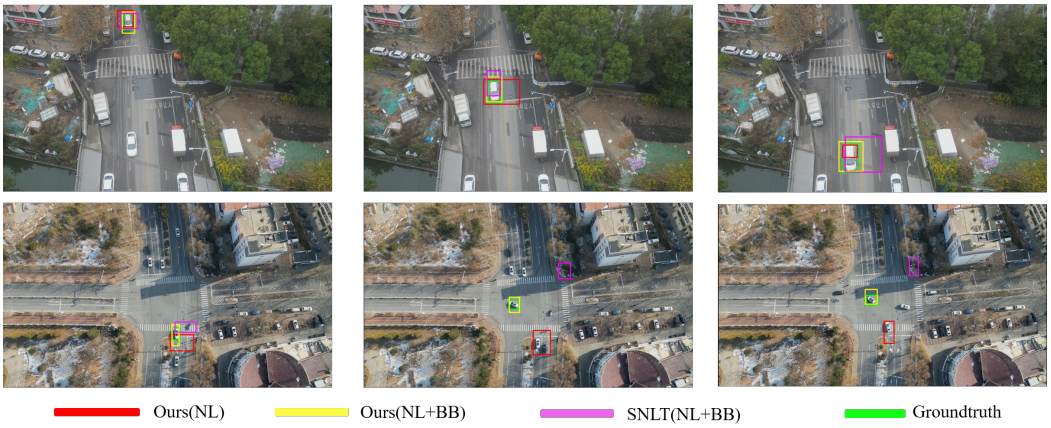


Figure 6. Qualitative comparison of two challenging scenarios of UAVNLT. The first row depicts a scenario with fewer vehicles and a lower flying altitude, characterizing a simpler setting. The second row, on the other hand, showcases a complex scene captured at a higher flying altitude over a crossroads.

5.2. Ablation Study

In order to verify the effectiveness of our proposed method, we conducted ablation studies on our approach for the TNL2K data set as shown in Table 3. For the baseline scenario indicated by ①, we replaced the M-Adapter, local attention module (LAM) and global attention module (GAM) with a straightforward concatenation method.

Table 3. Ablation studies on TNL2K. The symbol ✓ indicates the module is used in the framework.

Method	M-Adapter	LAM	GAM	AUC	Precision
①				0.443	0.396
②	✓			0.452	0.412
③	✓	✓		0.456	0.420
④	✓	✓	✓	0.461	0.422

The baseline method achieved scores of 0.443 and 0.396 for AUC and precision, respectively. When we incorporated the M-Adapter into our method, the performance improved to 0.452 for AUC and 0.412 for precision, indicating enhancements in both metrics. This improvement signifies that the proposed M-Adapter significantly promotes multi-modal alignment and fusion, demonstrating its effectiveness in enhancing overall tracking accuracy.

In addition, when we added the local attention module (LAM) and global attention module (GAM) to our approach, there was an improvement in our method's performance. This confirms that the proposed LAM and GAM contribute to the switcher module's ability to integrate local and global features effectively.

Efficiency of proposed switcherOur approach includes the visual grounding model, the object tracker model, and the switcher model. Compared to conventional target tracking methods, the main expense in our method is associated with the visual grounding model. However, the application of visual grounding is primarily focused on the initial localization in the first frame and re-localization after model switches. Based on our analysis and statistics across multiple data sets, as shown in Table 4, the additional overhead of the visual grounding model is about 11.4%, which does not lead to significant resource consumption. The overall method occupies 4.5 GB of VRAM, allowing it to be deployed on popular edge devices such as the Jetson Orin Nano (8 GB) and Xavier NX (8 GB) (Nvidia, located in Madison, Alabama, USA), with both operating at around 10 watts of power.

Table 4. The frequency of switch on the UAVNLT, TNL2K and LaSOT data sets.

Data Sets	Num Frames	Num Switch	Cost Rate
TNL2K	512,783	61,576	12.0%
LaSOT	684,688	69,933	10.2%
UAVNLT	295,607	38,731	13.1%
Total	1,493,078	170,240	11.4%

6. Conclusions and Future Works

In this study, we introduced a benchmark for UAV natural language-guided tracking named UAVNLT. The introduction of UAVNLT fills a gap in the domain of UAV tracking based on natural language, contributing to the broader applicability of UAV tracking. In our UAVNLT data set, we offer 2000 video sequences, with each target within the videos having been densely annotated to ensure the presence of a bounding box for the tracking target in every frame. Moreover, we provided standardized natural language descriptions, enabling precise localization of the target based on natural language.

In order to improve the tracking performance further and make full use of our data set, we propose a baseline for UAV-view natural language-guided tracking. In this baseline, we delineate three distinct modules: the visual grounding module, the object tracker, and the global–local switcher module. The incorporation of the switcher model allows our approach to dynamically switch between the visual grounding module and the object tracker, ensuring an effective balance between specificity and versatility in tracking performance.

In the future, we will consider further enhancing the robustness and accuracy of tracking in diverse and challenging environments, such as urban landscapes with high-density objects and natural terrains with varying weather conditions. Moreover, we will focus on improving the framework to achieve high performance.

Author Contributions: Conceptualization, H.L. and G.L.; Data curation, H.L. and X.L.; Formal analysis, H.L. and G.L.; Funding acquisition, G.L.; Investigation, H.L. and X.L.; Methodology, H.L.; Project administration, G.L.; Resources, G.L.; Software, H.L.; Supervision, G.L.; Validation, H.L., X.L., and G.L.; Visualization, H.L.; Writing—original draft, H.L.; Writing—review and editing, G.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work and the APC were supported in part by Key Deployment Program of the Chinese Academy of Sciences under Grant KGFZD145-23-18, and in part by the Fundamental Research Funds for Central Universities (E2ET1104).

Data Availability Statement: The data set and will be publicly available at https://github.com/Lich-King000/UAVNLT (accessed on 2 April 2024).

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Shao, Y.; Yang, Z.; Li, Z.; Li, J. Aero-YOLO: An Efficient Vehicle and Pedestrian Detection Algorithm Based on Unmanned Aerial Imagery. *Electronics* **2024**, *13*, 1190. [CrossRef]
2. Hu, Q.; Li, L.; Duan, J.; Gao, M.; Liu, G.; Wang, Z.; Huang, D. Object Detection Algorithm of UAV Aerial Photography Image Based on Anchor-Free Algorithms. *Electronics* **2023**, *12*, 1339. [CrossRef]
3. Yamani, A.; Alyami, A.; Luqman, H.; Ghanem, B.; Giancola, S. Active Learning for Single-Stage Object Detection in UAV Images. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA, 4–8 January 2024; pp. 1860–1869.
4. Rizzoli, G.; Barbato, F.; Caligiuri, M.; Zanuttigh, P. SynDrone-Multi-Modal UAV Dataset for Urban Scenarios. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Paris, France, 2–3 October 2023; pp. 2210–2220.
5. Javed, S.; Hassan, A.; Ahmad, R.; Ahmed, W.; Ahmed, R.; Saadat, A.; Guizani, M. State-of-the-Art and Future Research Challenges in UAV Swarms. *IEEE Internet Things J.* **2024**. [CrossRef]
6. Ren, H.; Zhao, Y.; Xiao, W.; Hu, Z. A review of UAV monitoring in mining areas: Current status and future perspectives. *Int. J. Coal Sci. Technol.* **2019**, *6*, 320–333. [CrossRef]
7. Moore, J.; Tadinada, H.; Kirsche, K.; Perry, J.; Remen, F.; Tse, Z.T.H. Facility inspection using UAVs: A case study in the University of Georgia campus. *Int. J. Remote Sens.* **2018**, *39*, 7189–7200. [CrossRef]
8. Li, X.; Yang, L. Design and Implementation of UAV Intelligent Aerial Photography System. In Proceedings of the 2012 4th International Conference on Intelligent Human-Machine Systems and Cybernetics, Nanchang, China, 26–27 August 2012; Volume 2, pp. 200–203. [CrossRef]
9. Zhao, H.; Zhang, H.; Zhao, Y. Yolov7-sea: Object detection of maritime uav images based on improved yolov7. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA, 2–7 January 2023; pp. 233–238.
10. Wang, G.; Chen, Y.; An, P.; Hong, H.; Hu, J.; Huang, T. UAV-YOLOv8: A small-object-detection model based on improved YOLOv8 for UAV aerial photography scenarios. *Sensors* **2023**, *23*, 7190. [CrossRef] [PubMed]
11. Paul, H.; Martinez, R.R.; Ladig, R.; Shimonomura, K. Lightweight multipurpose three-arm aerial manipulator systems for uav adaptive leveling after landing and overhead docking. *Drones* **2022**, *6*, 380. [CrossRef]
12. Lieret, M.; Lukas, J.; Nikol, M.; Franke, J. A lightweight, low-cost and self-diagnosing mechatronic jaw gripper for the aerial picking with unmanned aerial vehicles. *Procedia Manuf.* **2020**, *51*, 424–430. [CrossRef]
13. Nguyen, V.S.; Jung, J.; Jung, S.; Joe, S.; Kim, B. Deployable hook retrieval system for UAV rescue and delivery. *IEEE Access* **2021**, *9*, 74632–74645. [CrossRef]
14. Du, D.; Qi, Y.; Yu, H.; Yang, Y.; Duan, K.; Li, G.; Zhang, W.; Huang, Q.; Tian, Q. The unmanned aerial vehicle benchmark: Object detection and tracking. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 370–386.
15. Mueller, M.; Smith, N.; Ghanem, B. A benchmark and simulator for uav tracking. In Proceedings of the European Conference on Computer Vision (ECCV) 2016: 14th European Conference, Amsterdam, The Netherlands, 11–14 October 2016; Proceedings, Part I 14; Springer: Berlin/Heidelberg, Germany, 2016; pp. 445–461.
16. Zhu, P.; Wen, L.; Du, D.; Bian, X.; Fan, H.; Hu, Q.; Ling, H. Detection and tracking meet drones challenge. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *44*, 7380–7399. [CrossRef]
17. Wu, Y.; Lim, J.; Yang, M.H. Online object tracking: A benchmark. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Portland, OR, USA, 23–28 June 2013; pp. 2411–2418.
18. Fan, H.; Lin, L.; Yang, F.; Chu, P.; Deng, G.; Yu, S.; Bai, H.; Xu, Y.; Liao, C.; Ling, H. Lasot: A high-quality benchmark for large-scale single object tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 5374–5383.
19. Wang, X.; Shu, X.; Zhang, Z.; Jiang, B.; Wang, Y.; Tian, Y.; Wu, F. Towards more flexible and accurate object tracking with natural language: Algorithms and benchmark. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021; pp. 13763–13773.
20. Bertinetto, L.; Valmadre, J.; Henriques, J.F.; Vedaldi, A.; Torr, P.H. Fully-convolutional siamese networks for object tracking. In Proceedings of the European Conference on Computer Vision (ECCV) 2016, Amsterdam, The Netherlands, 11–14 October 2016; Springer: Cham, Switzerland, 2016; pp. 850–865.
21. Li, B.; Wu, W.; Wang, Q.; Zhang, F.; Xing, J.; Yan, J. Siamrpn++: Evolution of siamese visual tracking with very deep networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, 16–20 June 2019; pp. 4282–4291.
22. Li, B.; Yan, J.; Wu, W.; Zhu, Z.; Hu, X. High performance visual tracking with siamese region proposal network. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition,CVPR 2018, Salt Lake City, UT, USA, 18–22 June 2018; pp. 8971–8980.
23. Xu, Y.; Wang, Z.; Li, Z.; Yuan, Y.; Yu, G. Siamfc++: Towards robust and accurate visual tracking with target estimation guidelines. In Proceedings of the AAAI Conference on Artificial Intelligence, AAAI 2020, New York, NY, USA, 7–12 February 2020; Volume 34, pp. 12549–12556.
24. Zheng, Y.; Zhong, B.; Liang, Q.; Tang, Z.; Ji, R.; Li, X. Leveraging Local and Global Cues for Visual Tracking via Parallel Interaction Network. *IEEE Trans. Circuits Syst. Video Technol.* **2023**, *33*, 1671–1683. [CrossRef]

25. Ma, J.; Lan, X.; Zhong, B.; Li, G.; Tang, Z.; Li, X.; Ji, R. Robust Tracking via Uncertainty-Aware Semantic Consistency. *IEEE Trans. Circuits Syst. Video Technol.* **2023**, *33*, 1740–1751. [CrossRef]
26. Ge, D.; Liu, R.; Li, Y.; Miao, Q. Reliable Memory Model for Visual Tracking. *Electronics* **2021**, *10*, 2488. [CrossRef]
27. Zhao, M.; Okada, K.; Inaba, M. Trtr: Visual tracking with transformer. *arXiv* **2021**, arXiv:2105.03817.
28. Chen, X.; Yan, B.; Zhu, J.; Wang, D.; Yang, X.; Lu, H. Transformer tracking. In Proceedings of the Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, 19–25 June 2021; pp. 8126–8135.
29. Yan, B.; Peng, H.; Fu, J.; Wang, D.; Lu, H. Learning spatio-temporal transformer for visual tracking. In Proceedings of the IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, 10–17 October 2021; pp. 10448–10457.
30. Cui, Y.; Jiang, C.; Wang, L.; Wu, G. Mixformer: End-to-end tracking with iterative mixed attention. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, 18–24 June 2022; pp. 13608–13618.
31. Ye, B.; Chang, H.; Ma, B.; Shan, S.; Chen, X. Joint feature learning and relation modeling for tracking: A one-stream framework. In Proceedings of the In Proceedings of the Computer Vision–ECCV 2022, Tel Aviv, Israel, 23–27 October 2022; pp. 341–357.
32. Li, Z.; Tao, R.; Gavves, E.; Snoek, C.G.; Smeulders, A.W. Tracking by natural language specification. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, Honolulu, HI, USA, 21–26 July 2017; pp. 6495–6503.
33. Feng, Q.; Ablavsky, V.; Bai, Q.; Li, G.; Sclaroff, S. Real-time visual object tracking with natural language description. In Proceedings of the Winter Conference on Applications of Computer Vision, WACV 2020, Snowmass Village, CO, USA, 1–5 March 2020; pp. 700–709.
34. Yang, Z.; Kumar, T.; Chen, T.; Su, J.; Luo, J. Grounding-Tracking-Integration. *IEEE Trans. Circuits Syst. Video Technol.* **2021**, *31*, 3433–3443. [CrossRef]
35. Li, Y.; Yu, J.; Cai, Z.; Pan, Y. Cross-modal Target Retrieval for Tracking by Natural Language. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2022, New Orleans, LA, USA, 19–20 June 2022; pp. 4931–4940.
36. Feng, Q.; Ablavsky, V.; Bai, Q.; Sclaroff, S. Siamese natural language tracker: Tracking by natural language descriptions with siamese trackers. In Proceedings of the Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, 19–25 June 2021; pp. 5851–5860.
37. Wang, X.; Li, C.; Yang, R.; Zhang, T.; Tang, J.; Luo, B. Describe and attend to track: Learning natural language guided structural representation and visual attention for object tracking. *arXiv* **2018**, arXiv:1811.10014.
38. Guo, M.; Zhang, Z.; Fan, H.; Jing, L. Divert more attention to vision-language tracking. *NeurIPS* **2022**, *35*, 4446–4460.
39. Zhang, H.; Wang, J.; Zhang, J.; Zhang, T.; Zhong, B. One-stream Vision-Language Memory Network for Object Tracking. *IEEE Trans. Multimed.* **2023**, *26*, 1720–1730. [CrossRef]
40. Zheng, Y.; Zhong, B.; Liang, Q.; Li, G.; Ji, R.; Li, X. Towards Unified Token Learning for Vision-Language Tracking. *IEEE Trans. Circuits Syst. Video Technol.* **2023**, *34*, 2125–2135. [CrossRef]
41. Zhu, P.; Wen, L.; Bian, X.; Ling, H.; Hu, Q. VisDrone-DET2018: The Vision Meets Drone Object Detection in Image Challenge Results. In Proceedings of the European Conference on Computer Vision (ECCV) Workshops, Munich, Germany, 8–14 September 2018.
42. Li, B.; Yan, J.; Wu, W.; Zhu, Z.; Hu, X. Visual Object Tracking for Unmanned Aerial Vehicles: A Benchmark and New Motion Models. In Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017; Springer: Berlin/Heidelberg, Germany,2017; pp. 445–461.
43. Fu, C.; Cao, Z.; Li, Y.; Ye, J.; Feng, C. Onboard Real-Time Aerial Tracking with Efficient Siamese Anchor Proposal Network. *IEEE Trans. Geosci. Remote Sens.* **2021**, *60*, 5606913. [CrossRef]
44. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
45. Lin, T.Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature pyramid networks for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, 21–26 July 2017; pp. 2117–2125.
46. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; Volume 30.
47. Chen, T.; Saxena, S.; Li, L.; Fleet, D.J.; Hinton, G. Pix2seq: A language modeling framework for object detection. *arXiv* **2022**, arXiv:2109.10852.
48. Girshick, R. Fast r-cnn. In Proceedings of the International Conference on Computer Vision, ICCV 2015, Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
49. Radford, A.; Kim, J.W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. Learning transferable visual models from natural language supervision. In Proceedings of the 38th International Conference on Machine Learning, ICML 2021, Virtual Event, 18–24 July 2021; pp. 8748–8763.
50. Danelljan, M.; Bhat, G.; Khan, F.S.; Felsberg, M. Atom: Accurate tracking by overlap maximization. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, 16–20 June 2019; pp. 4660–4669.

51. Bhat, G.; Danelljan, M.; Gool, L.V.; Timofte, R. Learning discriminative model prediction for tracking. In Proceedings of the IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea, 27 October–2 November 2019; pp. 6182–6191.
52. Mayer, C.; Danelljan, M.; Paudel, D.P.; Van Gool, L. Learning target candidate association to keep track of what not to track. In Proceedings of the IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, 10–17 October 2021; pp. 13444–13454.
53. Zheng, Y.; Zhong, B.; Liang, Q.; Mo, Z.; Zhang, S.; Li, X. ODTrack: Online Dense Temporal Token Learning for Visual Tracking. *arXiv* **2024**, arXiv:2401.01686.
54. Wei, X.; Bai, Y.; Zheng, Y.; Shi, D.; Gong, Y. Autoregressive visual tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, 17–24 June 2023; pp. 9697–9706.