



Article

Detection-Free Object Tracking for Multiple Occluded Targets in Plenoptic Video

Yunjeong Yong¹, Jiwoo Kang^{2,*}  and Heeseok Oh^{1,*} ¹ Department of Applied AI, Hansung University, Seoul 02876, Republic of Korea; dbswjd9698@hansung.ac.kr² Division of Artificial Intelligence Engineering, Sookmyung Women's University, Seoul 04310, Republic of Korea

* Correspondence: jwkang@sookmyung.ac.kr (J.K.); ohhs@hansung.ac.kr (H.O.)

Abstract: Multiple object tracking (MOT) is a fundamental task in vision, but MOT techniques for plenoptic video are scarce. Almost all 2D MOT algorithms that show high performance mostly use the detection-based method which has the disadvantage of operating only for a specific object. To enable tracking of arbitrary desired objects, this paper introduces a groundbreaking detection-free tracking method for MOT in plenoptic videos. The proposed method deviates from traditional detection-based tracking methods, emphasizing the challenges of tracking targets with occlusions. The paper presents specialized algorithms that exploit the multifocal information of plenoptic video, including the focal range restriction and dynamic focal range adjustment schemes to secure robustness for occluded object tracking. To the improvement of the spatial searching capability, the anchor ensemble and the dynamic change of spatial search region algorithms are also proposed. Additionally, in terms of MOT, to reduce the computation time involved, the motion-adaptive time scheduling technique is proposed, which improves computation speed while guaranteeing a certain level of accuracy. Experimental results show a significant improvement in tracking performance, with a 77% success rate based on intersection over union for occluded targets in plenoptic videos, marking a substantial advancement in the field of plenoptic object tracking.

Keywords: multiple object tracking; plenoptic video; occluded target; multifocal information; time scheduling



Citation: Yong, Y.; Kang, J.; Oh, H. Detection-Free Object Tracking for Multiple Occluded Targets in Plenoptic Video. *Electronics* **2024**, *13*, 590. <https://doi.org/10.3390/electronics13030590>

Academic Editor: Beiwen Li

Received: 29 December 2023

Revised: 17 January 2024

Accepted: 29 January 2024

Published: 31 January 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Plenoptic imaging is one of the light field capturing techniques that reproduces the distribution of light rays emitted from an object. By placing the camera array, a sampling of the light field with the directions and the intensities of outgoing radiances from a scene can be captured. Because a plenoptic video includes both spatial and angular information in a 3D space, functions such as changing a viewpoint, refocusing, and adjusting a sense of depth can be given by processing focal information delivered from rays within 3D volume [1,2]. Through calibration and refocused rendering by using multiple images, a plurality of focal regions called focal planes constitute a focal stack which makes such post-processing possible [3]. Previous plenoptic-related research mainly focused on image acquisition, visualization, and display, that is, implementation of the hardware side [4]. However, in the recent field of plenoptic imaging, research is conducted in a wide range of various fields of computer vision that infer higher-level information as human perception, such as detection and semantic segmentation, but research on object tracking is inactively conducted [5,6]. As the demand for realistic and immersive content representing 3D space is emerging, the advanced tracking technique for editing the object and scene (e.g., deletion, completion, synthesis, emphasizing salient region, etc.) in producing and post-processing is required.

In recent, most existing visual object tracking (VOT) algorithms have been limited to general 2D videos, and the development that utilizes the focal information of a plenoptic sequence is scarce. In the 2D scenario, high-reliability tracking is still impossible in relation

to scenes including occlusions in which the target object is obscured by the occluder, and there are technical huddles in which it is difficult to easily derive a solution [7]. Whereas, in the case of a plenoptic sequence reconstructed through parallax between images captured by multiple cameras. Hereby, there are appearance features of the object that are weakly present in the focal stack, and whose focal information can be applied to track the occluded target as shown in Figure 1. Based on such characteristics, we adopted that focal information to track the occluded target and proposed the focal range restriction and dynamic focal range adjustment algorithms, which improve the robustness of tracking performance in the plenoptic sequence.

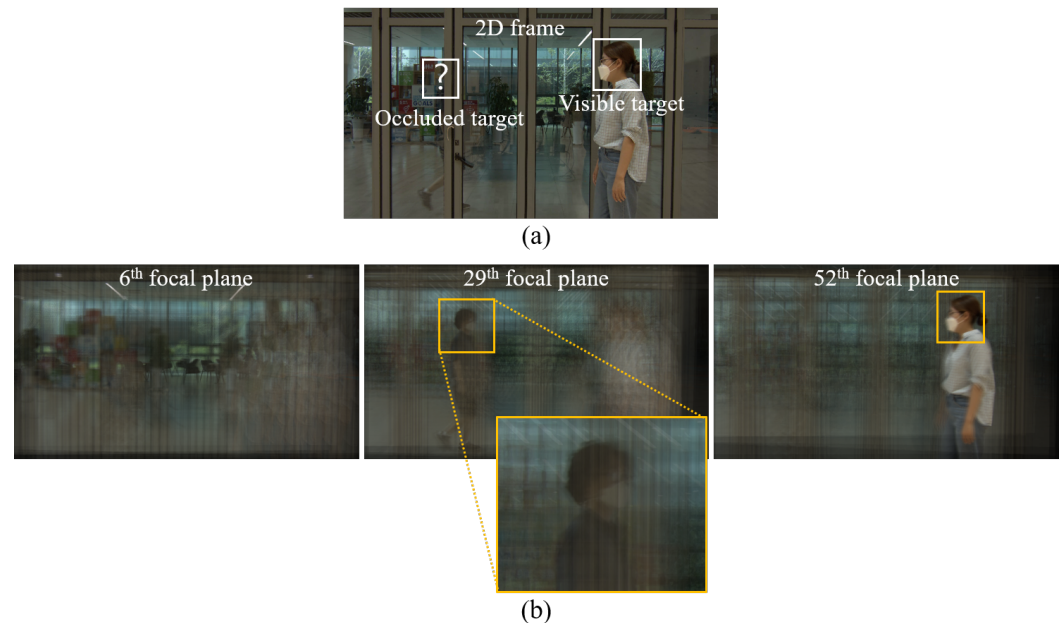


Figure 1. Comparison of 2D and plenoptic multiple object tracking when occlusion occurs. (a) One of the target objects is invisible in a 2D sequence due to an occluder. (b) Weak appearance of a corresponding target object exists in a plenoptic video constituted of multifocal planes along the depth.

As the object detection technique develops, most multiple object tracking (MOT) techniques have been performed through the detection-based tracking (DBT) approach [8]. However, the DBT methods can only deal with specific targets included in the training labels (e.g., pedestrians, vehicles, faces, etc.), thus there are limitations in tracking user-defined objects. From a practical post-processing point of view, a producer needs to track the desired target and correct it with the manual selection of the interested region. Therefore, we investigated the MOT approach by using multiple single-object trackers aiming for detection-free tracking (DFT) without an additional pre-detection overhead.

Towards this, our approach is to develop plenoptic VOT first and then utilize multiple VOT trackers to plenoptic MOT. For this purpose, we built a plenoptic baseline tracker using the modified 2D VOT model to track the target by calculating the similarity over all of the focal planes. When using the developed plenoptic baseline tracker naïvely, plenoptic VOT encounters problems such as exploring excessive focal regions and not being able to track sudden movements of objects along the depth. Such problems are resolved by the proposed algorithms, the restriction of the focal range, and dynamically adjusting it based on similarity. In addition, to further improve tracking performance, the spatial search corresponding focal information has also to be developed. In this paper, we propose a method that takes a mean ensemble of the results of the inferred boxes of anchors and a method to dynamically change the spatial search region at each time.

To deal with multiple objects, a bundle of single-object plenoptic trackers has to be processed simultaneously and long-term tracking as fast as possible. When MOT is per-

formed by simply copying VOTs, a long processing time is consumed due to computational complexity. To cope with this, we propose a method that dynamically skips tracking less important targets based on their movement at every frame called motion-adaptive time scheduling. By leveraging the motion adaptive time scheduling algorithm with consideration of multiple trackers, we verified that the tracking speed is improved compared to the naïve scheduling approach. In this paper, we compare MOT by using naïve parallelization, fairtime, and motion adaptive scheduling techniques for investigating effective ways of MOT. In the naïve parallelization, the number of single trackers is set to the number of objects to be tracked. Although each tracker can perform stable tracking without any dependency on each other, this requires heavy computation and long processing time. The fair-time scheduling is to have each tracker dominate a frame so that for N trackers, each tracker tracks once per N frames. This is designed to dramatically improve speed by reducing the number of tracker executions, but it is not realistic because it assumes that objects do not move much. The motion adaptive scheduling is to perform tracking based on the movement speed of the object. This method detects an abrupt movement change and applies it to the tracking, and which approach allows us to increase speed while maintaining tracking accuracy.

The main contribution of this work is three-fold: (1) We newly introduce the DFT MOT approach targeting plenoptic video. (2) We demonstrate that the proposed focal range restriction and dynamic focal range adjustment algorithms lead to improved tracking performances on the occluded target by utilizing multifocal information implied in the plenoptic video. (3) We propose motion-adaptive time scheduling which achieves improved MOT speed while guaranteeing a competitive performance.

2. Related Work

2.1. Visual Object Tracking

The purpose of VOT is to build a model that robustly responds to changes in the object's appearance and the object has to be tracked by deriving the relationship between the current frame and the previous frame. Bertinetto et al. [9] presented SiamFC, a Siamese network-based method that tracks an object by calculating the cross-correlation between features embedded through a weights-sharing backbone. Since then, a number of trackers developed using the Siamese neural network base have been proposed. He et al. [10] introduced SA-Siam which utilized an ensemble technique that increases the success rate of tracking by reflecting the inference results from a pre-trained semantic embedder along with the existing SiamFC. Wang et al. [11] proposed SiamMask, which simultaneously performs object segmentation as well as object tracking through multi-task learning. SiamRPN [12] and SiamRPN++ [13] added a region proposal network (RPN) in order to improve tracking performance by using anchors and learning start from default boxes. Guo et al. [14] proposed SiamCAR that tracks objects by performing regression and classification of bounding boxes in a pixel-wise manner to compensate for the disadvantages of vague parameter setting in anchor setting, which is a chronic problem on hyperparameter tuning lied in RPN. More recently, transformer-based tracking schemes have been introduced. Wang et al. [15] leveraged a Siamese-like tracking pipeline by combining with transformers to learn the distance between target and search region based on cross-attention. MixFormer [16] proposed the mixed attention module aiming for the estimation of the template's similarity. SeqTrack [17] removed the prediction head, and adopted transformer encoder-decoder architecture to generate a sequence of bounding boxes autoregressively. However, such single-object trackers inevitably encountered the occlusion problem of the limited visual information underlying 2D video; thus, in this paper, we design a more robust tracking scheme by utilizing the focal stack information given in the plenoptic sequence.

2.2. Multiple Object Tracking

Almost all MOT techniques detect objects first and then maintain the identity of each object frame-by-frame, but their main purpose was to establish a versatile mid-level model for developing higher-level tasks, e.g., behavior analysis, visual surveillance, and action

recognition. Hwang et al. [18] presented a technique for multi-target and multi-camera scenarios, a multi-pedestrian detection dataset for surveillance systems, and promoted the development of re-identification. Bewley et al. [19] proposed a simple online and real-time tracker SORT, a technique that combines the Kalman filter and the Hungarian algorithm for MOT. Zhang et al. [20] introduced a FairMOT that composes object detection and re-identification into one integrated neural network for MOT infers it and showed remarkable performance. MixSort [21] integrated both motion-based association and appearance-based association models for responding to both fast motion and undistinguishable appearance.

As shown in previous works, most MOT techniques employed the DBT approach, which implies that detection quality has a great influence on tracking performance. Above all, the MOT techniques of the DBT approach cannot track a desired target object which is impractical in commercial scenarios. Moreover, such methods make it difficult to utilize the focal information given as independent planes in the plenoptic sequence since the tracking process cannot be separated from a unified framework. Consequently, previous MOT methods can only deal with specific targets, there are limitations in applying them to user-defined objects for plenoptic videos pursued in our goal.

3. Plenoptic Object Tracking

3.1. Baseline Tracker

In the proposed plenoptic tracker, the foundation model for object tracking uses SiamRPN++ [13]. At the first frame, after the user manually designates the region of interest on the 2D frame, the target object is tracked by performing similarity calculation using the Siamese network for all focal planes in the focal stack for the subsequent frame. Figure 2 depicts the extraction of features in the focal planes that make up plenoptic videos. The ResNet50-based backbone extracts both encoded representations of the target object template and the focal plane, and after extracting and concatenating three intermediate features, the similarity was used as the final feature map for prediction. The feature map extracted here consists of 256 channels, and in the target object exemplar, two $4 \times 4 \times (4 \times 5 \times 256)$ size feature maps for bounding box coordinate prediction and for objectness classification, respectively. For each focal plane used for a search image, a feature map of size $20 \times 20 \times 256$ is extracted; thus, if there are K focal planes in the focal stack, a total of K feature maps are fed into the RPN.

Figure 3 shows the estimation of the similarity for bounding box prediction and objectness classification through cross-correlation between the feature maps finally extracted. The feature maps of the target object template and the focal plane extracted from the backbone are refined through convolution with 1×1 kernels. Then, similarity maps for box prediction having the relationship between the features extracted from the target and search region are estimated through a cross-correlation operation (utilizing Conv2D for implementation) as used in SiamRPN++ [13]. By performing the depth-wise cross-correlation operation between the final features, a map of size $17 \times 17 \times (4 \times 5)$ is predicted, containing the information of each of the four coordinates (x, y, h, w) for the representation of the five bounding boxes per anchor. Then, a $17 \times 17 \times (2 \times 5)$ similarity score map is computed for each of the five bounding boxes per anchor to perform the classification of the object and background. Consequently, by selecting the maximum value in the similarity maps, the region with the highest similarity is determined as the location of the object, and then object tracking works for the plenoptic sequence.

3.2. Focal Range Restriction

When the similarity is calculated for all focal planes in the focal stack based on the target object template through the baseline model and the region with the highest similarity is determined as the tracked object, object afterimages are generated due to reconstructed rendering from multiple images to the focal stack, which results in failure to track. Since it exists across multiple focal planes, the problem of being tracked to a position unrelated to the actual object arises. Moreover, if the search region for all focal planes in the focal

stack is specified, not only the amount of computation is excessive, but also the accuracy of predicting the coordinates of the bounding box is reduced. Because the maximum similarity score is yielded in the vast amount of focal information irrelevant to the target object and tracking fails, this paper proposes a restriction algorithm for the focal range.

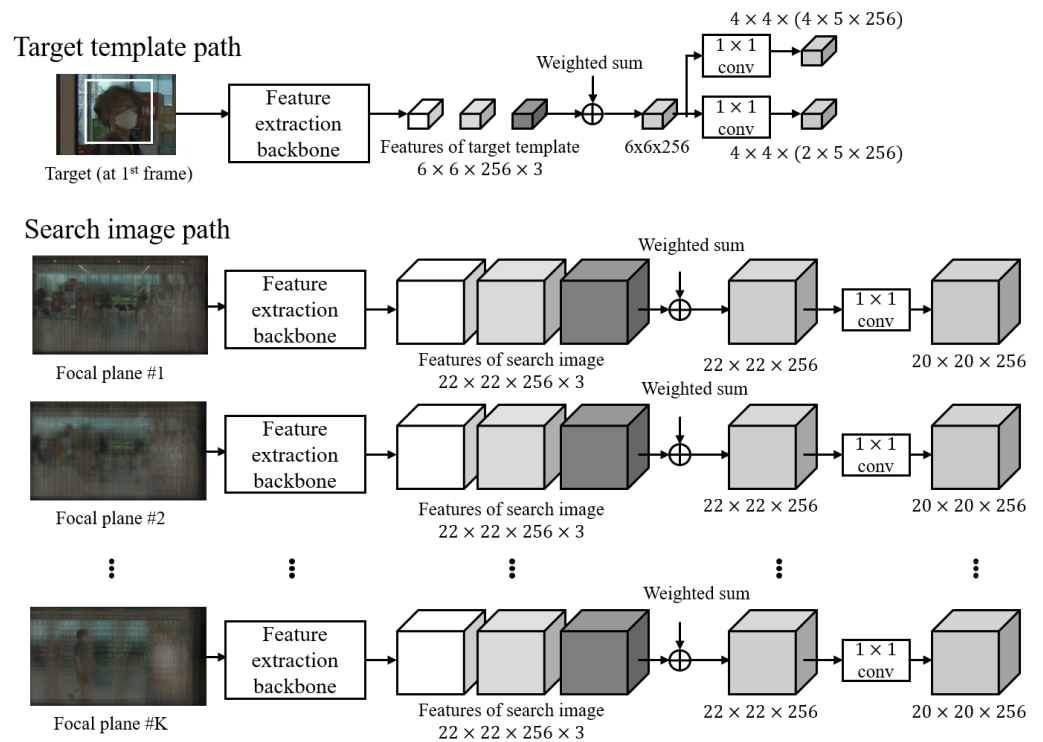


Figure 2. Feature extraction for plenoptic object tracking over the multiple focal planes.

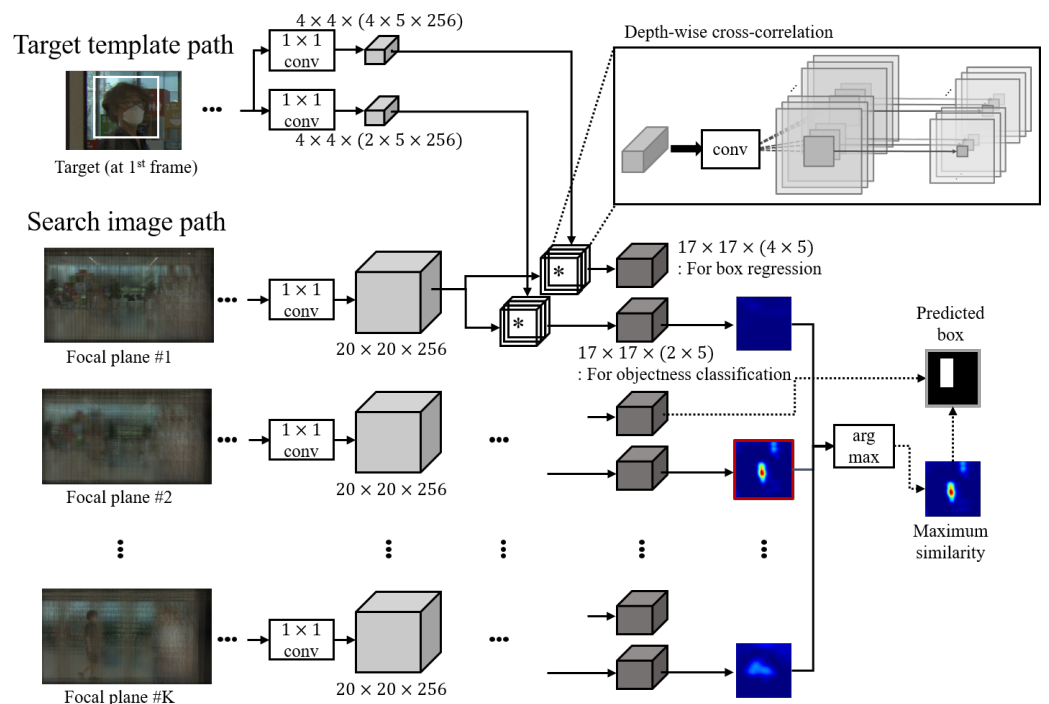


Figure 3. Bounding-box regression and objectness classification through similarity estimation for plenoptic object tracking.

For example, if the \hat{k}^1 -th focal plane in F_1^r shows the highest similarity, r focal plane candidate groups are regenerated centering on the \hat{k}^1 -th focal plane in the second frame, and these are formed into the F_2^r focal stack. And in the second frame, the similarity for all focal planes in the F_2^r focal stack is calculated, and the \hat{k}^2 -th focal plane with the highest similarity is configured as the search area in the third frame. This process is schematized in Schematic Figure 4, which shows the algorithm for limiting the focus range in each frame.

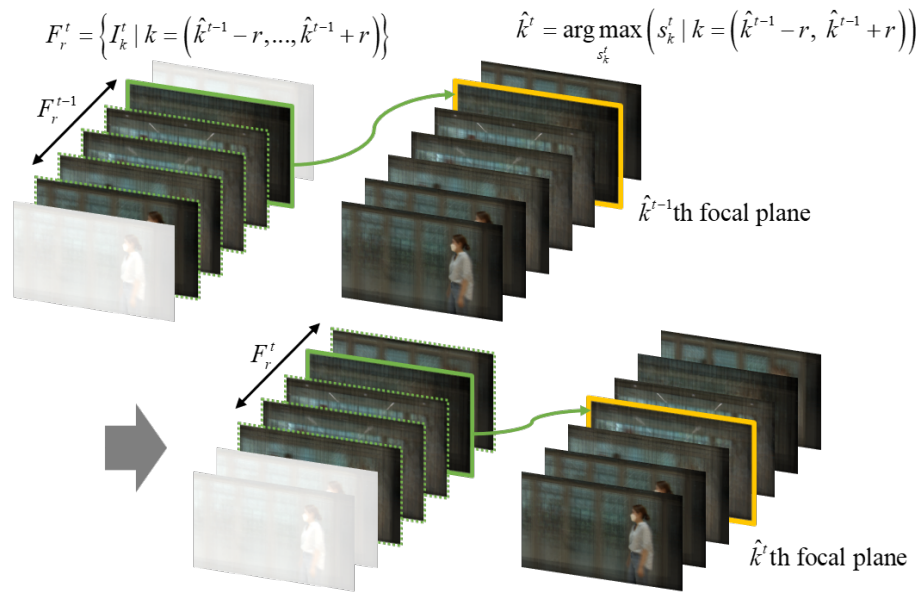


Figure 4. Focal range restriction algorithm at each frame. The method constrains the search area along the depth around the \hat{k} -th focal plane having the maximum similarity in the immediately preceding frame.

When the optimal focal plane I_k^{t-1} has the maximum similarity and whose index \hat{k}^{t-1} is selected from the focal stack F_r^{t-1} of the $(t - 1)$ -th frame (i.e., previous frame), the operation is performed considering only the focal planes of r centered on the \hat{k}^{t-1} -th focal plane, and then a new focal stack having a limited range is formed.

$$F_r^t = \{I_k^t \mid k = (\hat{k}^{t-1} - r, \dots, \hat{k}^{t-1} + r)\}. \tag{1}$$

After that, the tracker works on the newly constructed F_r^t for the t -th frame (i.e., current frame), and then the \hat{k}^t -th focal plane containing the target object region is obtained by calculating the similarity score s_k^t as stated in Section 3.1 for the corresponding restricted focal stack F_r^t .

$$\hat{k}^t = \operatorname{argmax}_{s_k^t} \{s_k^t \mid k = (\hat{k}^{t-1} - r, \dots, \hat{k}^{t-1} + r)\}. \tag{2}$$

Note that, since the restricted focal stack is undefined at $t = 0$ (that is, \hat{k}^{-1} cannot exist), every focal plane in the focal stack F is used to track the target in the first frame.

This method seeks to achieve more accurate object tracking by extracting features by reconstructing the candidate group within the focal stack at frame t , focusing on the focal plane with the highest similarity to the target object template at frame $t - 1$.

3.3. Dynamic Focal Range Adjustment

In order to perform robust tracking, features are extracted from the reconstructed focal stack in t -th frame, with the focal plane having the maximum similarity between the target object template and the focal plane in frame $t - 1$. As stated in (2), by setting the focal range r , the search range is composed in the range of $-r$ to $+r$ based on \hat{k}^{t-1} -th focal plane where an object

was tracked in the previous frame. Here, the default focal range r in this paper was empirically set to 5. It means that the object was tracked over a total of 11 focal planes for each frame.

However, when the search region is maintained to an unchangeable range, there is a limit in object tracking due to the restricted small focal range when object movement along the depth direction changes rapidly. Also, when the focal range is wider than necessary might result in the tracker being capable of detecting irrelevant features and missing weak target feature variation. Additionally, if the target object is occluded by the search occluder and reappears in the next frame, accurate search is difficult because it still refers only to the location at the time of disappearance, that is, specific focal planes. Therefore, by increasing the focus range, you can respond to objects wherever they appear.

Therefore, the focal range adjustment is required based on the estimated similarity between feature maps dynamically as shown in Figure 5. The excessively wide or small search range of the focal plane decreases the performance in plenoptic object tracking; thus, the level of range r is adjusted according to the similarity score s . The details were empirically set as shown in Table 1. The proposed dynamic adjustment of the focus range according to similarity aims for more accurate tracking and ensures robustness in object tracking. However, if the similarity score continues to be below a certain level, the object is judged to have been missed and the focal plane search area is quickly expanded. It is also an expansion technique.

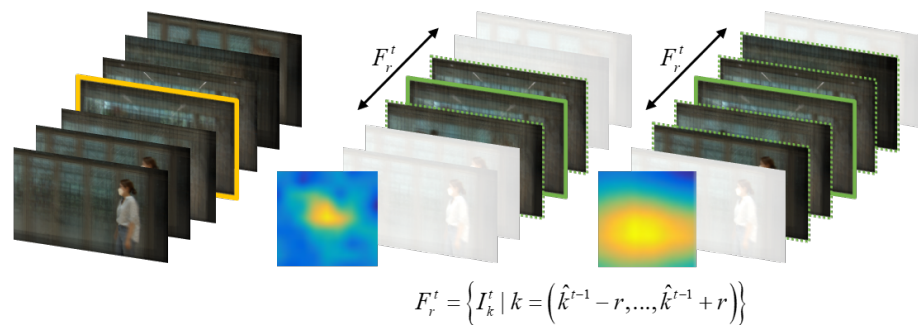


Figure 5. Illustration of dynamic focal range adjustment scheme based on the degree of similarity score.

In this study, we experimentally added a search area initialization algorithm that expands the search area r to 30 when the similarity value over 3 frames is 0.5 or less.

Table 1. Focal range settings based on similarity score.

Similarity Score	Focal Range r
$s > 0.8$	3
$0.8 \geq s \geq 0.2$	5
$0.2 > s$	7

Here, we add a search range initialization step that expands the range r to 30 when the similarity score is less than 0.5 for three or more frames in sequence.

3.4. Anchor Ensemble

In general, similar to SiamRPN++, which is employed for our baseline tracker, tracking algorithms including RPN require heuristic tuning for several hyperparameters of anchors such as default bounding boxes and scales [14]. When detecting the highest similarity with the target object, the possibility of error is low because only the spatial region is covered. However, the region is expanded to the spatial region times focal planes in the plenoptic sequence. In a plenoptic sequence, the $\max(\cdot)$ operation is computed on predicted bounding boxes of each focal plane, and then the $\max(\cdot)$ operation is calculated again for as many focal planes as the number of focal planes. Here, the $\max(\cdot)$ voting is repeated, and errors accumulate due to increased the number of combinations. Figure 6a illustrates the

problem of the existing method, as the $\max(\cdot)$ operation between the predicted bounding boxes of the maximum similarity anchors selected for each focal plane is performed again in all focal regions; errors caused by the $\max(\cdot)$ operation accumulate.

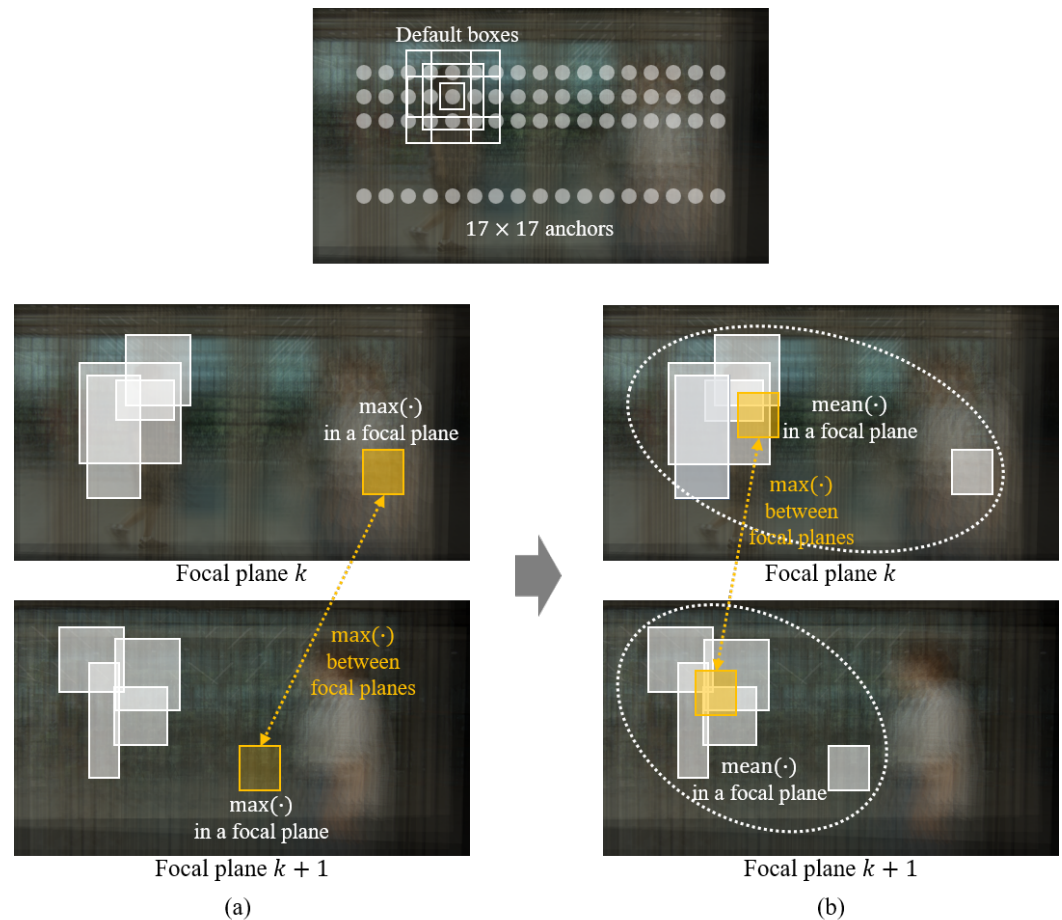


Figure 6. Anchor ensemble technique through $\text{mean}(\cdot)$ operation. (a) When the predicted bounding box having maximum similarity is separately far from the other anchor predictions in one focal plane, the $\max(\cdot)$ operation on all focal planes leads accumulation of errors. (b) Since multiple focal planes are adopted for inference, by utilizing the $\text{mean}(\cdot)$ operation instead of $\max(\cdot)$ depending on the similarity score in a focal plane, the predicted box coordinates are compensated.

To prevent such kind of error accumulation, as shown in (b), a method of ensemble predictions of anchor bounding boxes through $\text{mean}(\cdot)$ operation is devised. If the maximum similarity in one focal plane is 0.7 or higher, the object is tracked by $\max(\cdot)$ operation on anchor prediction results as before (i.e., five anchors in the proposed baseline tracker). Whereas, in the case of a similarity score less than 0.7, a method was adopted to limit the change so that the change is not large by performing the $\text{mean}(\cdot)$ operation on the coordinate values of the inference result of the bounding box derived from the other anchors.

3.5. Dynamic Change of Spatial Search Region

In SiamRPN++, the spatial searching used an algorithm that restricts region by cropping and resizing in the next t -th frame centered on the (x, y) coordinates of the tracked object in the $t - 1$ th frame. This assumption was based on the fact that the position of an object in the current frame is not significantly different from its position in the previous frame. Here, the crop-resize ratio was set as a hyperparameter based on the tracking coordinates of the previous frame [13]. However, when moving an object, the bounding box and spatial search range became excessively large or small depending on the size of the target object due to that heuristic ratio. As a result of tracking, the size of the bounding

box also continued to grow smaller or larger, consequently resulting in tracking failure. We observed that such a phenomenon has occurred frequently.

To resolve this problem, we utilized an additional approach to dynamically change the spatial search area, just as recent research has utilized dynamic neural networks to increase accuracy and computational efficiency [22], as shown in Figure 7. We set the criterion for expanding the spatial area to be searched to whether a similarity of 0.5 or less lasts for more than 3 frames. When this criterion is met, the spatial search area is increased by a single pixel. Empirically, when the increment is designated as larger than a single pixel, the predicted box area that is inferred thereafter tends to decrease since the target size gradually decreases relative to the search area. Hereby, more stable object tracking regarding spatial search is possible with the introduction of this algorithm.



Figure 7. Dynamic change of spatial search region according to similarity score, which prevents an excessive change of search region for the next prediction.

4. Scheduling for Faster Plenoptic MOT

4.1. Plenoptic MOT via Baseline Tracker Parallelization

As explained in Section 2, we aim to apply the baseline plenoptic tracker to multiple user-specified objects by creating tracker instances for each target as shown in Figure 8. At this time, if the feature extraction backbones are shared between trackers, the feature maps of each tracker will affect the prediction of other trackers, leading to unintended results. Therefore, in this paper, a deep copy is used to allow the properties of each tracker to be instantiated without sharing memory despite a bit higher resource consumption. This enables independent tracking of multiple objects without interference, relying on the baseline tracker’s performance, and exhibiting linearly increasing time consumption with the number of targets.

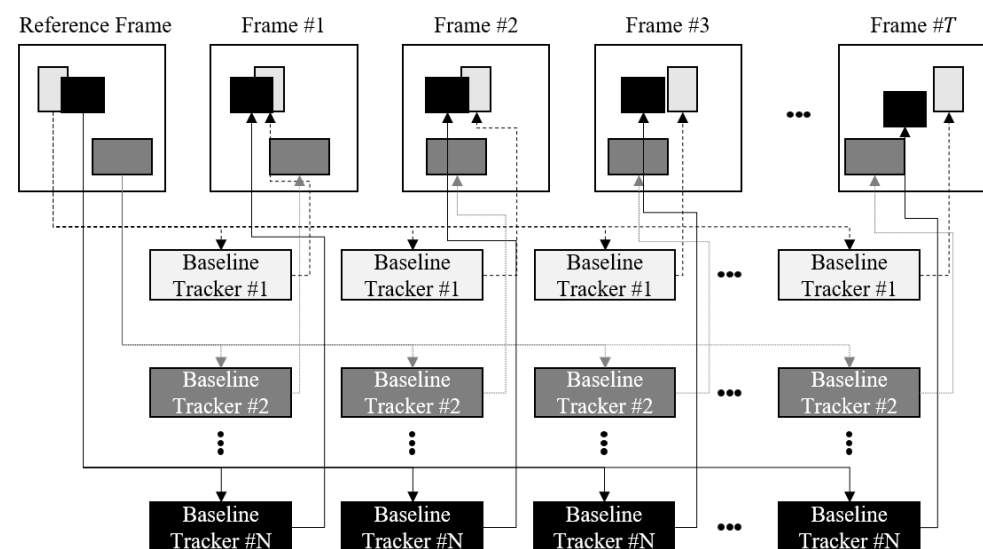


Figure 8. Naïve scheduling for MOT based on utilizing the multiple baseline trackers.

4.2. Fair-Time Scheduling

The parallelization of MOT provides strong tracking performance, but the computational complexity is high because there are as many baseline trackers as there are objects and they operate simultaneously. The basic method for improving speed in MOT is to split the work evenly as depicted in Figure 9. Here, individual trackers are assigned to objects, with only one tracker operating per frame, while others wait for their turn. If there are N trackers, tracking is performed once per N frames, which results in a speed improvement of N times compared to a situation in which all trackers are operating.

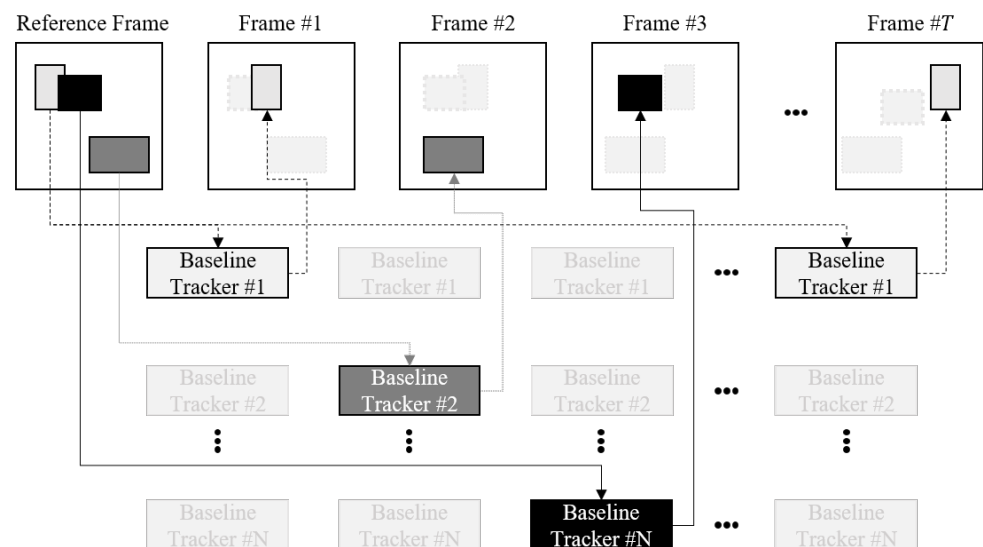


Figure 9. Fair-time scheduling technique for MOT which boosts speed by tracking only a single target per frame, but whose performance is in inverse proportion to the number of targets.

However, the fair-time scheduling technique is based on the assumption that there is only a single target object per frame and that the movement speed of each target object is stationary, which is an unrealistic scenario. Additionally, if there are many objects to be tracked, the number of skipped frames increases, and the tracking success rate becomes inversely proportional to the number of target objects.

4.3. Motion Adaptive Time Scheduling

To reduce computational complexity in object tracking and increase speed while maintaining accuracy, this paper introduces a motion adaptive time scheduling method that assigns trackers to multiple objects based on motion speed. This approach is inspired by the existing method, which dynamically skips less important frames and then quickly selects valuable regions from the remaining frames [23].

When the moving direction of each object changes, a tracker is applied to prevent tracking in the wrong direction. If object tracking is in progress in the same direction as the current moving direction, it can be viewed as a form of skipping, such as fair-time scheduling. Since there is an assumption that no sudden rapid change in speed occurs, tracking for that frame can be skipped. The direction of movement here can be known based on the amount of change in speed. In Figure 10, a conceptual diagram for motion adaptive time scheduling that skips the tracking of a certain object among a plurality of target objects based on the moving speed is visualized.

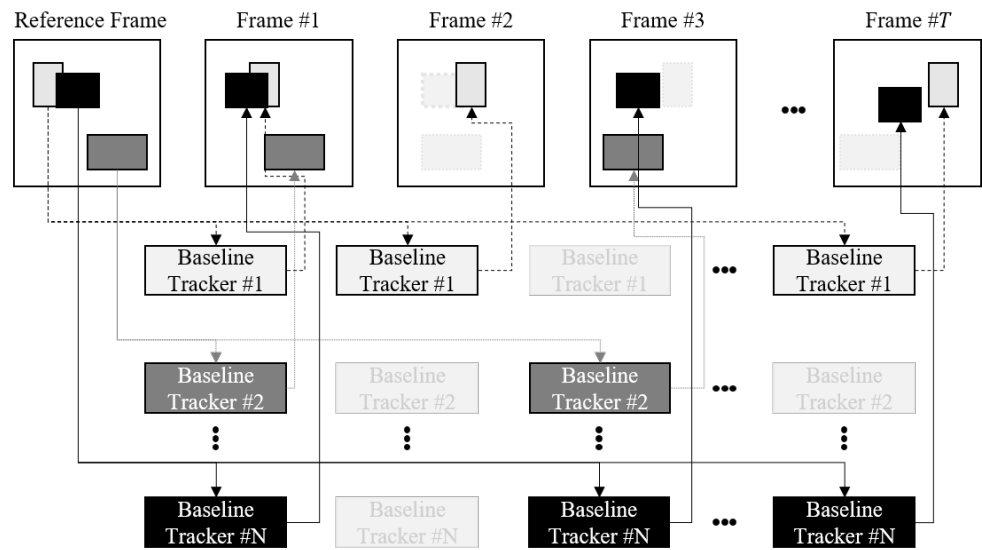


Figure 10. Motion adaptive time scheduling in multi-object tracking.

Figure 11 visualizes the criteria for skipping the tracking of a specific target. The main idea is that if the target movement varies dynamically, the corresponding tracker continuously tracks the target as depicted in (a). On the other hand, if the motion of a target is small enough, the corresponding tracker does not track its target in that frame as shown in (b). Towards this, at first, each tracker in a frame calculates the central distance of n -th object coordinates between frames:

$$m_t^n = \sqrt{(x_{t-1-\eta}^n - x_t^n)^2 + (y_{t-1-\eta}^n - y_t^n)^2}, \tag{3}$$

where η and $m_{t-\eta}^n$ represent a skipped frame number increment (default $\eta = 0$) and motion speed, respectively. Let m_{t-2}^n and m_{t-1}^n be the distances between the center coordinate of the previous second frame and the center coordinate of the previous frame, and between the center coordinate of the previous frame and the center coordinate of the current t -th frame, then the differential motion speed can be calculated:

$$\Delta m_t^n = m_{t-2}^n - m_{t-1}^n. \tag{4}$$

Based on (4), the n -th object in the corresponding frame O_t^n is skipped being tracked when the differential motion speed of a specific object below a certain level γ and the similarity score in the previous frame s_{t-1}^n exceeds 0.5:

$$\begin{cases} O_t^n \notin \Omega_t & |\Delta m_t^n| \leq \gamma, \text{ and } s_{t-1}^n > 0.5 \\ O_t^n \in \Omega_t & \text{otherwise,} \end{cases} \tag{5}$$

where Ω_t represents the set of targets at t -th frame. Here, we set γ to 5% of the frame size [24]. When the n -th object tracking is skipped in the previous frame, the central distance m_t^n in (3) and the motion speed Δm_t^n in (4) are calculated with η incremented as the number of skipped frames to determine whether the object is tracked or not in the current frame as shown in (b). Additionally, when O_t^n is skipped by two consecutive frames, the spatial search region is expanded by 30 pixels to minimize tracking errors.

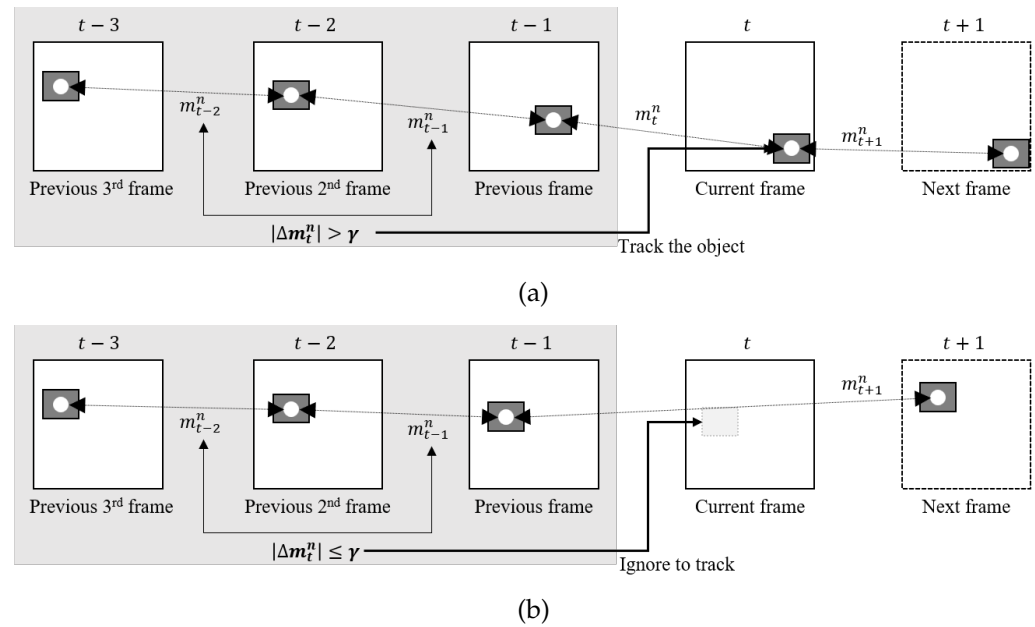


Figure 11. Criterion to adaptively skip certain object tracking based on motion. (a) Visualize cases where the motion speed of an n -th object exceeds a certain level of γ , where the corresponding tracker keeps track of the target. (b) Visualize cases where the motion speed of an n -th object is below a certain level of γ , where the corresponding tracker determines to ignore the tracking target.

5. Experiments

5.1. Plenoptic Video Sample

To evaluate the performance of the proposed scheme, the *NV1* sequence is used which is an unstructured plenoptic video taken by 16 UHD cameras with parallax at an arbitrary position and is composed of a focal stack including 101 focal planes per frame. Since the plenoptic sequence used for MOT in this experiment is not a public dataset and there is no ground truth (i.e., target bounding boxes) for evaluating the accuracy, four subjects manually annotated it.

5.2. Performance Metrics

In this paper, the performance of MOT is calculated through bounding boxes drawn around the target objects to be tracked. The first used metric is the central distance [25], which is measured Euclidean distance based on the coordinates of the centers of the two bounding boxes, ground truth, and prediction. That is, object tracking error is expressed as a distance in pixels, and success or failure is determined by calculating whether an error exists within a specific distance criterion compared to the size of the entire scene. The second utilized metric is intersection over union (IoU), which is an indicator that considers not only the center coordinates (x, y) but also the width and height (w, h) of the bounding box. It is calculated by dividing the overlapping area of the two bounding boxes by the area of the sum of the widths of the bounding boxes derived as a result of tracking the plenoptic object. Generally, in object tracking competitions such as VOT challenges, the object tracking success rate is judged based on IoU 50% [26].

5.3. Performance Evaluation on Plenoptic VOT

At first, we verified the performance of plenoptic VOT (i.e., tracking a single target). Experiments were performed in an ablation study manner; thus, the results show the improvement level of plenoptic tracking following adding each proposed module. Additionally, to show the superiority of our proposed method, the previous plenoptic VOT model introduced by Bae et al. [25] was also compared. The results of evaluated central distance and IoU are shown in Figure 12 and Table 2, and Figure 13 and Table 3, respectively.

Table 2. Average central distance for different algorithms of VOT in plenoptic sequence NV1.

Algorithm	Central Distance (Pixels)
2D tracking	118
Bae et al. [25]	358
A: Plenoptic tracker baseline	162
B: A + focal range restriction & adjustment	53
C: B + anchor mean ensemble	38
D: C + search region change	28

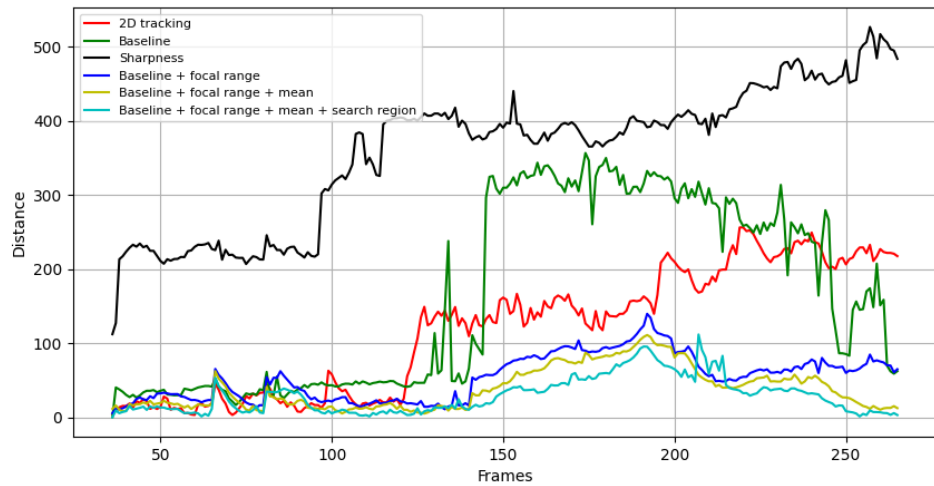


Figure 12. Central distance measure for different algorithms of VOT in plenoptic sequence NV1.

Table 3. Average IoU for different algorithms of VOT in plenoptic sequence NV1.

Algorithm	IoU (%)
2D tracking	3
Bae et al. [25]	1
A: Plenoptic tracker baseline	24
B: A + focal range restriction & adjustment	41
C: B + anchor mean ensemble	55
D: C + search region change	72

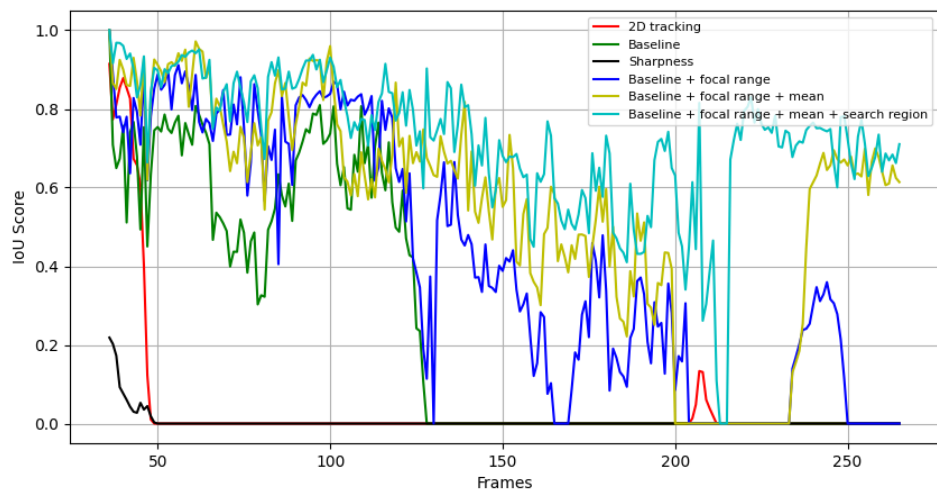


Figure 13. IoU measure for different algorithms of SOT in plenoptic sequence NV1.

As can be seen from the results, the performance of the 2D tracker (i.e., SiamRPN++) on the plenoptic sequence was notably low since the target object in NV1 is occluded at

approximately the 100th frame and fails to track from then. What is noteworthy is that the previous plenoptic tracking method showed poor performance on *NV1*, because the focal regions selection by using sharpness metric did not work on the *NV1* which contains many edges in the scene. On the other hand, the proposed plenoptic baseline tracker (A in Tables) showed that it can track the desired target to some extent, which is particularly represented in the IoU score as tabulated in Table 3. Moreover, it can be shown that overall performance significantly increased when the proposed schemes were embodied. The plenoptic tracking using the focal range restriction and dynamic adjustment (B in Tables) shows an improvement of about 17% over the plenoptic baseline tracker. When the anchor ensemble (C in Tables) raised performance another 14%. Finally, when all of the proposed schemes were applied, the model resulted in a total improvement of about 48% over the plenoptic baseline tracker. As shown in Figures 12 and 13, the proposed plenoptic tracker briefly missed the target at about 220th frame but immediately started to track it normally differing from the other methods. The experimental results demonstrate that the proposed method operates with higher reliability than the existing VOT for plenoptic sequences.

5.4. Performance Evaluation on Plenoptic MOT

For three target objects within the plenoptic sequence *NV1*, four different methods were compared: MOT by the general 2D tracker (i.e., SiamRPN++), the proposed plenoptic MOT baseline (naïve scheduling), MOT with fair-time scheduling, and MOT with the motion adaptive time scheduling. In Figures 14–16 and Tables 4–6, the tracking performances on the unstructured plenoptic sequence *NV1* are quantitatively shown using central distance, IoU, and tracking speed, respectively.

Table 4. Average central distance for different algorithms of MOT in plenoptic sequence *NV1*.

Algorithm	Central Distance (Pixels)
2D tracking	570
Plenoptic MOT baseline (naïve scheduling)	66
Fair-time scheduling	78
Motion adaptive time scheduling	60

As shown in Figure 14 and Table 4, the general tracker failed to track after an occlusion occurred (~100th frame), and the central distance was constantly increased due to the accumulation of errors. Differing from the 2D scenario, the other plenoptic MOT algorithms showed they were able to achieve stable tracking performance in spite of occlusion. Similar to central distance results, as shown in Figure 15 and Table 5, IoU represents the matching ratio between the ground truth and predicted bounding box; thus, the higher the % value is the higher the tracking performance. In the case of the plenoptic MOT baseline algorithm, which performed tracking for three objects in each frame, it showed the best performance with 77% IoU, but the object tracking speed was the lowest as shown in Figure 16 and Table 6. When using the fair-time scheduling technique, there was a speed improvement of about three times (i.e., similar to the number of target objects), but the tracking performance decreased to 51%. When using the motion adaptive scheduling based on motion speed, the overall tracking speed was improved by approximately 18% compared to the naïve approach, while the tracking performance was relatively lower but acceptable. Note that the reason why the latency at the first frame is that it needs to set up a buffer to store past movements.

Table 5. Average IoU for different algorithms of MOT in plenoptic sequence *NV1*.

Algorithm	IoU (%)
2D tracking	29
Plenoptic MOT baseline (naïve scheduling)	77
Fair-time scheduling	51
Motion adaptive time scheduling	68

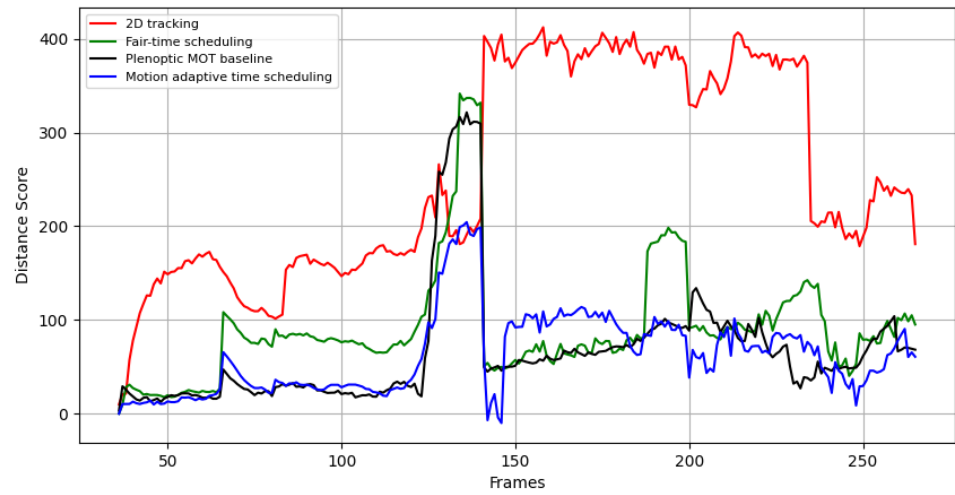


Figure 14. Central distance measure for different algorithms of MOT in plenoptic sequence NV1.

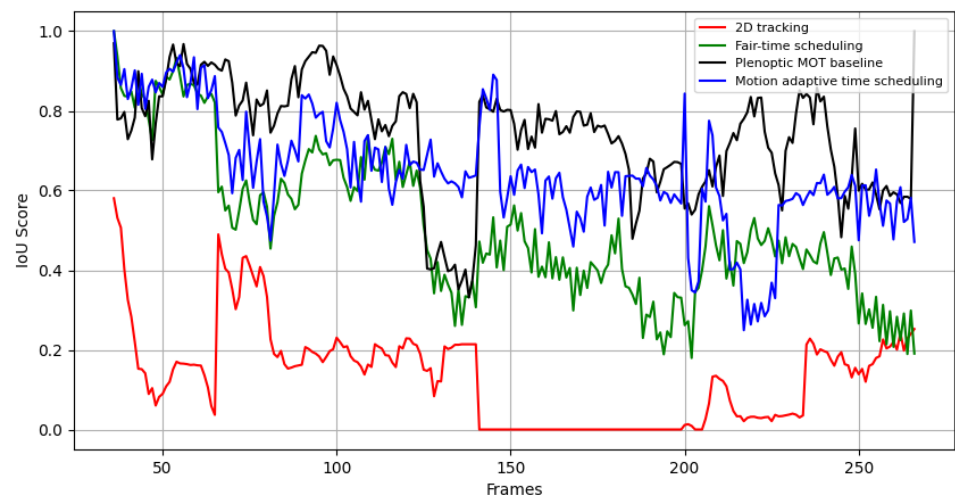


Figure 15. IoU measure for different algorithms of MOT in plenoptic sequence NV1.

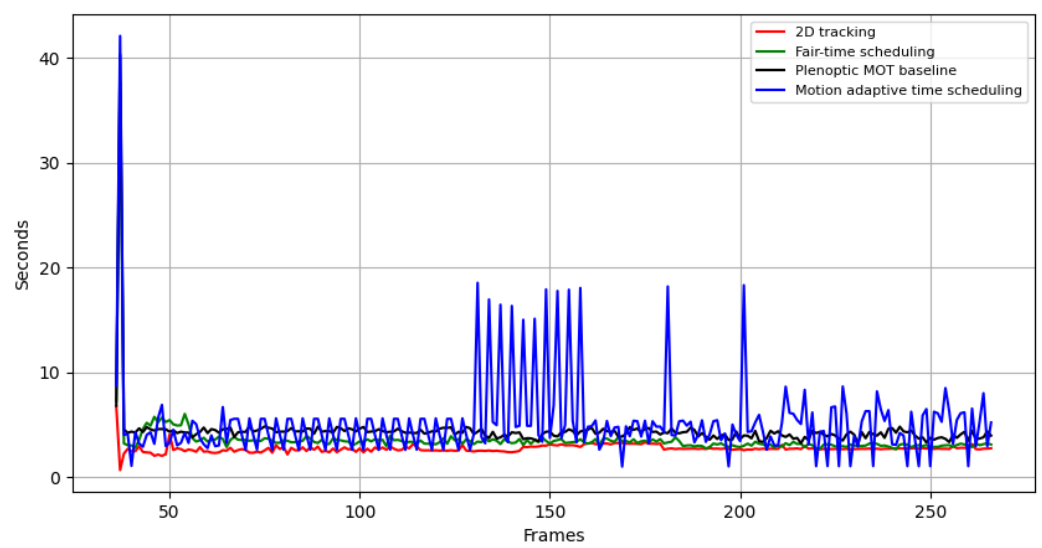


Figure 16. Tracking speed measure for different algorithms of MOT in plenoptic sequence NV1.

Table 6. Average speed for different algorithms of MOT in plenoptic sequence NV1.

Algorithm	Sec/Frame
2D tracking	0.3
Plenoptic MOT baseline (naïve scheduling)	4.23
Fair-time scheduling	1.25
Motion adaptive time scheduling	3.45

The quantitative results demonstrated that the MOT models based on the proposed plenoptic MOT baseline are able to cope with occlusion. Furthermore, although a trade off between tracking performance and speed inevitably exists in a DFT manner, the results showed that the proposed motion adaptive time scheduling algorithm achieved stable performance while reducing computational complexity.

Figures 17 visualize the results of the proposed plenoptic MOT for each algorithm in NV1. As shown in the qualitative results, it verified that tracking of objects with occlusion fails in the case of using only 2D information in (a), whereas the proposed plenoptic MOT algorithm performed successful tracking even when occlusion occurs as shown in (b). When the fair-time scheduling technique is applied as shown in (c), there is an advantage in tracking speed, but when occlusion occurs, a certain target object (e.g., foot) is missed. Whereas, when the motion adaptive time scheduling was applied in (d), there was a benefit in overall tracking speed, and reliable tracking was performed even when occlusion occurred for all objects.

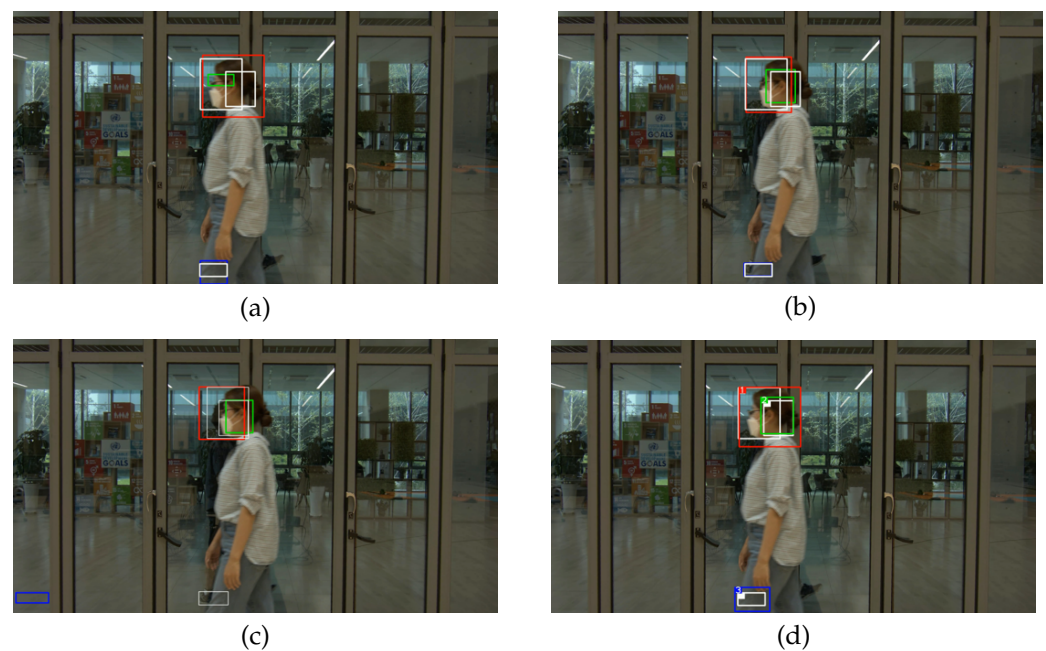


Figure 17. Results for each scheduling algorithm applied to Plenoptic MOT. The white boxes are ground truths, and the colored boxes are predictions: (a) 2D tracker applied to the plenoptic sequence. The tracking target fails due to occlusion. (b) Naïve parallelization. Object successfully tracked even when the target is unseen, but requires heavy computation. (c) Fair-time scheduling. Advantage in speed, but targets often missed during occlusion. (d) Motion adaptive scheduling. The target object was successfully tracked even with occlusion with an advantage in speed.

6. Conclusions

In this work, the plenoptic MOT scheme in terms of DFT is proposed that can be applied to arbitrary objects. The proposed method can track desired objects when they are not within a predefined category and even if they are invisible to the occluder. To this end, a robust baseline plenoptic tracker search over the focal stack is investigated, and focal

range restriction and adjustment techniques to prevent false positives are introduced. In addition to improving tracking performance regarding focal information, we developed anchor pooling and search methods over spatial regions to correctly track the target. In addition, through motion adaptive parallelization of the plenoptic tracker, the proposed method enables high-speed, high-precision MOT for the plenoptic sequence having vast information. As a result of applying the proposed techniques to MOT in unstructured plenoptic videos, it quantitatively showed superior performance compared to existing 2D-oriented MOT and achieved robust results even when occlusion occurs.

The method proposed in this paper can be used as a software plug-in of a platform in post-processing for plenoptic content creators, and producers since the desired modification of manually targeted objects is available. Moreover, it is expected to be used as an additional function for advertising or promoting products from a plenoptic video service provider through MOT. We now investigate the complete post-processing solution, including video inpainting, editing, or partial generation in conjunction with the tracked target.

Author Contributions: Conceptualization, J.K.; Methodology, H.O.; Software, Y.Y. and H.O.; Validation, Y.Y.; Formal analysis, J.K.; Investigation, Y.Y.; Resources, H.O.; Writing—original draft, Y.Y.; Writing—review & editing, J.K. and H.O.; Visualization, Y.Y.; Supervision, H.O.; Project administration, H.O.; Funding acquisition, J.K. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. 2020-0-00457, Development of ultra high resolution unstructured plenoptic video authoring and playback platform technology for large capture space) and National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2022R1F1A1068704).

Data Availability Statement: The data presented in this study are available on request from the corresponding author (accurately indicate status).

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

References

1. Li, Y.; Sjöström, M.; Olsson, R. Scalable coding of plenoptic images by using a sparse set and disparities. *IEEE Trans. Image Process.* **2016**, *25*, 80–91. [[CrossRef](#)] [[PubMed](#)]
2. Seitz, S.M.; Kutulakos, K.N. Plenoptic image editing. *Int. J. Comput. Vis.* **2002**, *48*, 616–625.
3. Moreno-Noguer, F.; Belhumeur, P.N.; Nayar, S.K. Active refocusing of images and videos. *ACM Trans. Graph.* **2007**, *26*, 67-1–67-9. [[CrossRef](#)]
4. Zhan, F.L.; Wang, J.; Shechtman, E.; Zhou, Z.Y.; Shi, J.X.; Hu, S.M. PlenoPatch: Patch-based plenoptic image manipulation. *IEEE Trans. Vis. Comput. Graph.* **2017**, *23*, 1561–1573. [[CrossRef](#)] [[PubMed](#)]
5. Chen, W.; Zhu, C.; Zhang, S. Piecewise segmentation occlusion model for image-based plenoptic spectral analysis. In Proceedings of the IEEE 23rd International Workshop on Multimedia Signal Processing (MMSp), Tampere, Finland, 6–8 October 2021.
6. Sabrina, Y.; Tayeb, M. Plenoptic imaging for object detecting and tracking: An edge detection approach. *Int. J. Appl. Eng. Res.* **2018**, *13*, 11392–11401.
7. Kwon, J.; Lee, K.M. Visual tracking decomposition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), San Francisco, CA, USA, 13–18 June 2010.
8. Luo, W.; Xing, J.; Milan, A.; Zhang, X.; Liu, W.; Kim, T.K. Multiple object tracking: A literature review. *Artif. Intell.* **2021**, *293*, 103448. [[CrossRef](#)]
9. Bertinetto, L.; Valmadre, J.; Henriques, J.F.; Vedaldi, A.; Torr, P.H.S. Fully-convolutional siamese networks for object tracking. In Proceedings of the European Conference on Computer Vision Workshop (ECCVW), Amsterdam, The Netherlands, 11–14 October 2016.
10. He, A.; Luo, C.; Tian, X.; Zeng, W. A twofold Siamese network for real-time object tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018; pp. 4834–4843.
11. Wang, Q.; Zhang, L.; Bertinetto, L.; Hu, W.; Torr, P.H.S. Fast online object tracking and segmentation: A unifying approach. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 1328–1338.
12. 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), Salt Lake City, UT, USA, 18–22 June 2018; pp. 8971–8980.

13. Li, B.; Wu, W.; Wang, Q.; Zhang, F.; Xing, J.; Yan, J. SiamRPN++: Evolution of Siamese visual tracking with very deep network. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 4277–4286.
14. Guo, D.; Wang, J.; Cui, Y.; Wang, Z.; Chen, S. SiamCAR: Siamese fully convolutional classification and regression for visual tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 14–19 June 2020; pp. 6269–6277.
15. Wang, N.; Zhou, W.; Wang, J.; Li, H. Transformer meets tracker: Exploiting temporal context for robust visual tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 19–25 June 2021; pp. 1571–1580.
16. 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), New Orleans, LA, USA, 18–24 June 2022; pp. 13608–13618.
17. Chen, X.; Peng, H.; Wang, D.; Lu, H.; Huu, H. SeqTrack: Sequence to sequence learning for visual object tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Vancouver, BC, Canada, 18–22 June 2023; pp. 14572–14581.
18. Hwang, S.; Kim, N.; Choi, Y.; Lee, S.; Kweon, I.S. Fast multiple objects detection and tracking fusing color camera and 3D LIDAR for intelligent vehicles. In Proceedings of the International Conference on Ubiquitous Robots and Ambient Intelligence (URAI), Xian, China, 19–22 August 2016.
19. Bewley, A.; Ge, Z.; Ott, L.; Ramos, F.; Upcroft, B. Simple online and realtime tracking. In Proceedings of the IEEE International Conference on Image Processing (ICIP), Phoenix, AZ, USA, 19 August 2016.
20. Zhang, Y.; Wang, C.; Wang, X.; Zeng, W.; Liu, W. FairMOT: On the fairness of detection and re-identification in multiple object tracking. *Int. J. Comput. Vis.* **2021**, *129*, 3069–3087. [[CrossRef](#)]
21. Cui, Y.; Zeng, C.; Zhao, X.; Yang, Y.; Wu, G.; Wang, L. SportMOT: A large multi-object tracking dataset in multiple sports scenes. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Paris, France, 2–6 October 2023; pp. 9921–9931.
22. Han, Y.; Huang, G.; Song, S.; Yang, L.; Wang, H.; Wang, Y. Dynamic neural networks: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *44*, 7436–7456. [[CrossRef](#)] [[PubMed](#)]
23. Wang, Y.; Chen, Z.; Jiang, H.; Song, S.; Han, Y.; Huang, G. Adaptive focus for efficient video recognition. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 16249–16258.
24. Held, D.; Thrun, S.; Savarese, S. Learning to track at 100 fps with deep regression networks. In Proceedings of the European Conference on Computer Vision (ECCV), Amsterdam, The Netherlands, 11–14 October 2016; pp. 749–765.
25. Bae, D.H.; Kim, J.W.; Heo, J.P. Content-Aware Focal Plane Selection and Proposals for Object Tracking on Plenoptic Image Sequences. *Sensors* **2019**, *19*, 48. [[CrossRef](#)] [[PubMed](#)]
26. Kristan, M.; Matas, J.; Leonardis, A.; Vojir, T.; Pflugfelder, R.; Fernandez, G.; Nebehay, G.; Porikli, F.; Čehovin, L. A novel performance evaluation methodology for single-target trackers. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *38*, 2137–2155. [[CrossRef](#)] [[PubMed](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.