FDA-SSD: Fast Depth-Assisted Single-Shot MultiBox Detector for 3D Tracking Based on Monocular Vision

Zihao Wang 1,2, Sen Yang 1,2, Mengji Shi 1,2 and Kaiyu Qin 1,2,*

1 School of Aeronautics and Astronautics, University of Electronic Science and Technology of China, Chengdu 611731, China; 201611191002@std.uestc.edu.cn (Z.W.); 2018100402009@std.uestc.edu.cn (S.Y.); maangat@126.com (M.S.)
2 Aircraft Swarm Intelligent Sensing and Cooperative Control Key Laboratory of Sichuan Province, Chengdu 611731, China
* Correspondence: kyqin@uestc.edu.cn; Tel.: +86-13981987921

Abstract: In this study, a set of benchmarks for object tracking with motion parameters (OTMP) was first designed. The sample images were matched with the spatial depth of the camera, the pose of the camera, and other spatial parameters for the training of the detection model. Then, a Fast Depth-Assisted Single-Shot MultiBox Detector (FDA-SSD) algorithm suitable for 3D target tracking was proposed by combining the depth information of the sample into the original Single-Shot MultiBox Detector (SSD). Finally, an FDA-SSD-based monocular motion platform target detection and tracking algorithm framework were established. Specifically, the spatial geometric constraints of the target were adapted to solve the target depth information, which was fed back to the detection model. Then, the normalized depth information of the target was employed to select the feature window of the convolutional layer for the detector at a specific scale. This significantly reduces the computational power for simultaneously calculating detectors of all scales. This framework effectively combines the two-dimensional detection model and the three-dimensional positioning algorithm. Compared with the original SSD method, the network model designed in this study has fewer actual operating parameters; the measured detection operation speed was increased by about 18.1% on average; the recognition rate was maintained at a high level consistent with that of the original SSD. Furthermore, several groups of experiments were conducted on target detection and target space tracking based on monocular motion platforms indoors. The root mean square error (RMSE) of the spatial tracking trajectory was less than 4.72 cm. The experimental results verified that the algorithm framework in this study can effectively realize tasks such as visual detection, classification, and spatial tracking based on a monocular motion platform.

Keywords: real-time object tracking; depth-assisted training; benchmark for tracking; monocular vision

1. Introduction

With a large number of applications of visual algorithms in autonomous navigation of unmanned vehicles and UAV positioning, real-time tracking and positioning of 3D space targets based on vision has become a popular direction in the field of computer vision. In recent years, methods based on neural networks and deep learning have exhibited high accuracy and strong generalization capabilities for target detection and tracking. Meanwhile, various types of neural-network models have been proposed, such as the Siamese network for improving the tracking speed [1], and the Feature Pyramid Network for high-precision detection [2]. The one-stage method SSD [3] considers both speed and performance, and the once-for-all (OFA) method is for lightweight computing [4]. Besides, the emergence of unmanned driving [5] and visual slam [6] has enabled researchers to gradually expand the focus of visual research from 2D planar-image problems to 3D spatial vision [7–9]. However, it is difficult for most neural networks to achieve both accuracy
and real-time due to their complex networks and a huge number of parameters. Therefore, how to effectively reduce the computational assistance of the target-detection model is an essential research direction.

In many specific target-tracking applications, the 2D detection method of the target and the 3D spatial-positioning problem are generally coupled. The plane information projected by the camera is frequently employed to solve the spatial-positioning problem, while the solved spatial position can be adapted to assist the tracking algorithm of the projection plane. Similarly, can the tracking result of the projection plane be merged with the spatial depth information (or the trajectory relative to the camera) obtained by the spatial-positioning algorithm? Can it be applied as a depth parameter to join the training of the network model and finally obtain a high-precision and high-speed detection model? Based on the above consideration, a set of benchmarks for object tracking with motion parameters (OTMP) matching the sample spatial information was first designed using the indoor motion-capture system in this study to assist our algorithm research. Then, an improved model based on Single-Shot MultiBox Detector (SSD) was proposed. This model integrates the spatial-depth information of the sample into the network training and adopts the normalized depth information of the target to select the feature window and the classifier under the convolutional layer of a specific scale on the original SSD for detection. This significantly reduces the computational burden of simultaneously calculating feature channels of all scales. Finally, a target-detection and tracking algorithm framework based on the FDA-SSD monocular sensor platform was established in the combination with the author’s previous target-location method based on geometric constraints [10]. Specifically, an efficient and robust geometric constraint equation method was taken to solve the target space; the normalized depth information of the target in the current frame was fed back to the FDA-SSD of the next frame to select a detector at the matching scale for target detection. Finally, the real-time target detection and tracking of the monocular vision platform was realized.

Considering that the term “target” refers to a wide range, the “target” that is mainly studied in our study refers to objects that need to be positioned in monocular-vision scenes, such as doors or windows that need to be passed, obstacles that need to be avoided, etc. In the experimental part, the circular and rectangular box-like samples in the OTMP dataset were used as the “targets” to be tracked, and the relevant details are explained in Section 3. Although the target types in practical applications are different, the core idea of this study is to combine the spatial parameters to realize the scale selection of the detection model. Therefore, the method proposed in our study applies to most non-deformed targets, unless the target is deformed violently, causing the normalized parameter-matching error. The main contributions of this study:

1. A method for establishing benchmarks for object tracking with motion parameters (OTMP) was proposed. This dataset is constructed by the image obtained by the monocular camera, the motion trajectory of the monocular camera, the motion trajectory of the sample to be trained, and camera internal parameters. This data set can be used for visual scientific research such as visual slam of indoor dynamic environments, spatial positioning of moving targets, and dynamic target-recognition and classification.

2. A network structure of a Fast Depth-Assisted Single-Shot MultiBox Detector was designed. With a data set with depth information (such as the OTMP in this article), the original SSD and the target-depth information were fused to train, enabling the network to learn the correlation between the depth of a target and the feature detectors at various scales. The normalized depth information obtained by combining target-positioning and -tracking algorithms can highly reduce the amount of computing on the SSD network when FDA-SSD is employed for actual detection.

3. Based on the FDA-SSD network and the author’s previous work [10], a target-detection-and-tracking algorithm framework for a monocular motion platform was established using the spatial-positioning method of geometric constraints. In this
study, multiple sets of indoor target-tracking experiments were conducted with the indoor dynamic capture system as the ground truth. The test data demonstrate that the average RMSE of the tracking trajectory was 4.72 cm, which verifies the effectiveness of the target-tracking algorithm framework.

The rest of this article is organized as follows. In Section 2, the research work related to this study is introduced. Then, benchmarks for object tracking with motion parameters are briefly described in Section 3. Next, the design and training method of the FDA-SSD neural network is detailed in Section 4. The methods of target depth extraction and spatial positioning based on geometric constraints are illustrated in Section 5. Afterward, the comparison of related algorithms and experimental verification is presented in Section 6. Finally, the conclusions are drawn in Section 7. Besides, the related fund projects that supported the research of this study are listed.

2. Related Works

2.1. Data Set

At present, the typical datasets based on visual-tracking research mainly include the short-term tracking dataset named OTB100 [11] for humans, cars, and other targets; the long-term tracking dataset named TLP [12]; the dataset named VOT2020 [13] with a large number of deformation, rotation, and occlusion characteristics; and difficult interference Tracking Net [14] involving camera movement, image blur, and video leaving. The OTMP dataset designed in this study aimed to provide an indoor dataset with camera-motion and 3D-motion trajectories in template space, as well as an experimental simulation and a verification basis for sample training with depth parameters, visual slam, target 3D positioning, and tracking problems.

2.2. Target-Detection Model in Real-Time 3D Tracking

Given the real-time problem of algorithms, there are two types of common methods based on deep-learning models: the Siamese network [1,15] and single-stage methods such as SSD [3] and YOLO v1 [16]. Regarding the problem of spatial target tracking, the Siamese network, though it is very fast when addressing spatial 3D targets, is usually unable to adapt to the target’s rotation and angle changes. Meanwhile, the SSD algorithm has a good data-augmentation module and stronger robustness. Its multi-scale detector inspired us to improve our ideas. In this study, the FDA-SSD method was to learn the depth information of the training-sample target in the front end of the classification detector of different channels. As a result, the FDA-SSD can obtain the information of the target related to the spatial depth of the camera coordinate system, to select the matched detector.

2.3. Depth Estimation in Monocular Vision

There are some typical methods of traditional monocular depth estimation [17–19]. The current works of depth estimation based on neural networks are still not mature enough, with a problem of a large amount of calculation and poor real-time performance, such as CNN-based monocular depth-estimation methods [20], multi-scale networks for DepthMap estimation methods [21], the method of using the Res-Net to solve the depth estimation [22], and the method of using the combination of CNN and a graphical model [23]. Considering the real-time problem, accurate positioning and stable tracking of 3D-space targets were achieved in this study based on the author’s previous research work and geometrically constrained spatial-positioning method [10]. Our method considers the computational burden and real-time issues while ensuring the tracking accuracy.

3. Data Set with Motion Parameters

The samples in the OTMP dataset were classified into two types: rectangular frame-shaped objects and circular frame-shaped objects. All samples were taken by a monocular fixed-focus pinhole camera. The trajectory information of multiple sets of sample targets in space and the pose information of the camera itself were simultaneously recorded using
an indoor motion-capture system. Besides, the camera internal parameter matrix $CM$, the sample calibration set $S$, the sample-space motion-trajectory parameters $T_s$, and the camera’s pose $[R_c|T_c]$ after calibration using the checkerboard are provided. Notably, this study focused on the spatial trajectory $T_s$ of the target and offered the rotation matrix $R_s$ of the rigid target for part of the data. Finally, $[T_s, R_s, R_c, T_c]$ corresponding to the synchronization is presented for each frame of data. The storage capacity of this dataset was 2.67 GB; the total number of frames was 10432; and the resolution was $1280 \times 720$. This dataset was uploaded to GitHub: https://github.com/6wa-car/OTMP-DataSet.git (accessed on 30 December 2021).

Figure 1 illustrates the indoor collection environment of this dataset. Figure 2 exhibits some typical sample types in the data set. Figure 3 shows the spatial motion trajectory of the monocular camera and the target measured by the motion-capture system.

Figure 1. Measurement of indoor environment. (a) Collection environment of datasets, (b) motion-capture system monitoring station perspective, (c) example of the tracking interface of the motion-capture system.

Figure 2. Samples in benchmark for object tracking with motion parameters. (a) Circular-like sample, (b) rectangle-like sample, (c) monochrome samples for classification.

Figure 3. The trajectory of the target and the camera obtained by the motion—capture system.
Our dataset can be used to study the problems of target tracking in 3D space, target classification, and visual SLAM in a motion-interference environment. In this study, a part of this dataset was selected as the training samples to investigate the training method of the FDA-SSD network. In the experimental part, the reliability of the 3D-space target-tracking algorithm based on FDA-SSD and geometric constraint methods was verified with other non-overlapping data as the experimental test data.

4. FDA-SSD Network

4.1. Improved Model

The original SSD design was to use multi-layer convolution to obtain feature extraction of different scales, perform a large number of anchors on each scale, and finally screen the optimal detection by non-maximum suppression. The advantage of this algorithm is that it can adapt to the detection of targets at different scales, while it also has the problem of a large amount of calculation. Regarding each type of target, the smallest-scale detector in SSD needs 4332 anchors; the second-smallest scale needs 2166 anchors, followed by 600, 150, 54, and 6, with a total of 8732 anchors to be detected.

Considering that the depth of the target related to the camera-coordinate system in the target-tracking problem can be solved, a Fast Depth-Assisted Single-Shot MultiBox Detector for 3D tracking based on monocular vision was designed in this study based on the strong correlation between adjacent frames in the video tracking problem. Figure 4 exhibits a schematic diagram of the Fast Depth-Assisted Single-Shot MultiBox Detector (FDA-SSD) proposed in this study. The part in the red box represents the depth-assisted training model. Regarding the FAD-SSD model, in addition to the image input, we increased the depth correlation between adjacent frames in the video tracking problem. For 3D tracking based on monocular vision was designed in this study based on the strong correlation between adjacent frames in the video tracking problem. Figure 4 exhibits a schematic diagram of the Fast Depth-Assisted Single-Shot MultiBox Detector (FDA-SSD) proposed in this study. The part in the red box represents the depth-assisted training model. Regarding the FAD-SSD model, in addition to the image input, we increased the depth value $D_{norm}$ of the target sample relative to the camera-coordinates. Besides, the scale correlation information of the sample with the depth value was learned through a Feedforward Neural Network (FNN) and a SoftMax module to determine the detection layer that obtains the most-reliable detection result.

![Figure 4. Structure of Fast Depth-Assisted Single-Shot MultiBox Detector.](image)

With respect to the target spatial-positioning algorithm, a suitable detector can be selected for target recognition and extraction instead of searching in every scale. This method can significantly reduce the amount of calculation in the detection part. The deep
conv-layer computing cost in the backbone structure can be saved. The FNN structure is provided in Figure 5.

![Figure 5: FNN structure of classification for depth value.](image)

Among them, the input layer $x$ corresponds to the depth information $D_{\text{norm}}$ obtained by the tracking algorithm; the number of hidden layer units was set to 20; the maximum confidence probability $p_i$, $i = 1, 2, 3, 4, 5, 6$ of the detection anchor frame obtained by the six different scale detectors of the SSD corresponding to the output layer $y_i$, $i = 1, 2, 3, 4, 5, 6$ after SoftMax is:

$$y_i = \frac{\exp(p_i)}{\sum_{j=1}^{6} \exp(p_j)}, \quad i = 1, 2, 3, 4, 5, 6,$$

(1)

The loss function $L$ used by FNN in this algorithm is cross-entropy:

$$L = \frac{1}{N} \sum_i L_i = \frac{1}{N} \sum_i - [y_i \times \log(p_i) + (1 - y_i) \times \log(1 - p_i)],$$

(2)

where $y_i$ denotes the probability that the detector correctly detects the sample under six detection scales, and $p_i$ represents the predicted probability under different input depth values. The FDA-SSD training method is consistent with the original SSD. The structure of FNN combined with the SoftMax decision-making scheme was adapted on the increased depth channel, allowing the model to self-learn from end to end if it matches the optimal scale. There was no need to manually mark which scale to use.

4.2. Issues Related to Depth Values

It is worth noting that our FDA-SSD model still needs to clarify three important issues:

1. For the FDA-SSD-based detection-and-tracking framework, the detection model has not yet detected the target sample to be tracked when the first frame of the image is given. Therefore, the target depth value cannot be obtained.
2. For the Nth frame image, the input depth value is the depth information calculated from the N − 1 frame instead of the current frame.
3. The scale of the training sample may not be consistent with the size of the tracking target during visual inspection. The same depth information does not match the same 2D-image scale. Therefore, the algorithm needs to normalize parameters such as the depth and the target scale.

The above three issues were discussed and analyzed as follows:

1. In most detection and spatial-positioning models, the basis of the 3D-spatial-positioning algorithm is to obtain the detection results of the 2D-projection surface. Therefore, during the process of using the FDA-SSD model, the depth-assisted module was not adapted by default when the first frame of the image was initialized for detection. Instead, detectors at all scales were directly employed to detect the target and obtain the initial-value 2D-tracking anchor frame. Then, the spatial-positioning-and-tracking algorithm in Section 5 was adapted to obtain the initial depth value $D_0$ of the target.
2. Given the tracking problem based on real-time video streams, adjacent images have strong relevance. In other words, the positions of the \( N-1 \)th frame target and the \( N \) frame target are very close in most cases. Therefore, the depth information of the \( N-1 \) frame can be adapted to assist the detection of the \( N \)th frame. There are some extreme situations. For example, the high-speed movement of the target in the radial direction of the camera results in a large gap between the depth values of frame \( N-1 \) and frame \( N \). In this case, we can obtain the movement trend of the target through the target-tracking algorithm and turn on the corresponding scale detector of the corresponding \( N-1 \) frame and its adjacent detectors to expand the scale-detection interval. Concerning general detection, when the depth information of the target moves between two scale detectors, the close FNN output-estimation probabilities \( y_i \) and \( y_{i+1} \) indicates that the target can be detected on both the \( i \)-th and \( i+1 \)-th scale detectors. At this point, we can open the two road detectors to perform directory detection.

3. Since there may be differences between the sample scale and the real target scale in practical applications, similar triangular properties of the projection should be employed to return the depth value to a unified standard, as illustrated in Figure 6.

We defined the normalized depth value of the sample in each frame of the image when training the FDA-SSD model:

\[
D_{\text{norm}} = \frac{D_s}{\lambda_s} = \frac{D_0}{\lambda_0}.
\]  

The normalized depth value of the FDA-SSD model when tracking targets in practical applications:

\[
D_{\text{norm}} = \frac{D_0}{\lambda_0}.
\]

It can be observed that when the ratio of the depth value \( D_s \) of the training sample to the scale value \( \lambda_s \) is equal to the ratio of the depth value \( D_0 \) of the actual detection target to the scale value \( \lambda_0 \), the normalized scale \( \lambda_{\text{norm}} \) projected on the 2D-image plane is also the same according to the nature of similar triangles. Therefore, the training depth information is actually normalized depth information obtained from the samples. In the test, the input is the normalized deep learning of the target obtained by the tracking algorithm.

5. 3D Target Tracking

5.1. Overall Framework

In Section 4, the FDA-SSD model construction with normalized depth information was completed. After the model network was used to obtain the target’s 2D anchor frame, the edge information of the sample and the required characteristic corner points can be obtained following specific needs. Given the target space positioning of geometric constraints, a
target-space-tracking framework based on monocular vision was proposed, as shown in Figure 7.

**Figure 7.** The overall framework diagram of the visual target-detection-and-tracking algorithm of the monocular moving platform.

The framework consists of three parts:

1. Two-dimensional-detection based on a neural network (red part): The FDA-SSD model was adapted to obtain the anchor frame area of the sample in real-time. Considering the real-time requirements, the key corner points can be extracted according to the typical geometric characteristics of the target, such as triangles, parallelograms, and circles, for use in subsequent positioning algorithms. Finally, the motion information of the target related to the camera in the current frame was employed to extract the depth information of the target relative to the camera, obtain the restored depth value based on the target scale and the sample scale, and feed it back to the input of the FDA-SSD as the reference depth value of the next frame of the image.

2. Three-dimensional-location method based on geometric constraints (blue part): The targets in this article were abstracted as a circle or a parallelogram, and the improved high-order Newton iterative algorithm was used to realize the efficient real-time numerical solution of the depth information of the target feature points. Then, the spatial motion of the target related to the camera was restored using depth information. Finally, the real-space motion trajectory of the target was calculated with the pose of the camera.

3. Camera-pose acquisition (gray part): The monocular camera pose can be obtained by open-source visual-slam methods such as VINS [24], ORB-SLAM3 [25], or refer to the author’s previous work [26,27]. This article focuses on two parts, part1 and part2. In the experimental verification of this study, the camera pose was acquired by the motion-capture system. In practical applications, the absolute scale information learning of the target can be obtained by multiple methods such as multi-frame initialization in the monocular vision slam, the baseline of the visual motion platform, and the prior information of the cooperative target.
5.2. Spatial-Positioning Method Based on Geometric Constraints

For a point \( P \) in three-dimensional space, it is expressed by \( P_w = [X_{wi}, Y_{wi}, Z_{wi}]^T \) in the world coordinate system \( W \). For the pixel coordinate system \( UV \), it is expressed by \( P_{ui} = [u_x, v_y, 1]^T \). For the camera coordinate system \( C \), it is expressed by \( P_c = [X_c, Y_c, Z_c]^T \). The input of the depth-estimation module is a typical pinhole projection model and the internal reference l relationship of the pinhole projection mode. The pixel coordinates of the extracted feature points, and the output is a depth-estimation equation set with the depth information of the feature points as unknowns. In this study, the targets were abstracted as geometric images such as parallelograms or circles.

As shown in Figure 8, the coordinates of the target in the world coordinate system were defined as \( P_i = [X_{wi}, Y_{wi}, Z_{wi}]^T, i = 1, 2, 3, 4 \ldots \). The coordinates in the monocular camera coordinate system were defined as \( P_{wi} = [p_{uxi}, p_{wyi}, p_{wzi}]^T, i = 1, 2, 3, 4 \ldots \). The coordinates of the target in the two-dimensional image coordinate system were defined as \( P_{ui} = [u_i, v_i, 1]^T, i = 1, 2, 3, 4 \ldots \). The normalized projection points of the normalized plane \( Z = 1 \) were defined as \( P_c = [p_{cxi}, p_{cyi}, 1]^T, i = 1, 2, 3, 4 \ldots \). The relationship between the image coordinate system and the camera coordinate system can be obtained with the internal reference model of the pinhole camera:

\[
\begin{bmatrix}
  u \\
  v \\
  1
\end{bmatrix} = CM \times \begin{bmatrix}
  x_c \\
  y_c \\
  1
\end{bmatrix} = 
\begin{bmatrix}
  f_x & 0 & c_x \\
  0 & f_y & c_y \\
  0 & 0 & 1
\end{bmatrix} \times \begin{bmatrix}
  x_c \\
  y_c \\
  1
\end{bmatrix},
\]

where \( CM \) is the internal parameter matrix. The camera can identify target feature points and obtain a set of 2D-image point positions through real-time image processing. We can obtain the coordinates of the projection with distortion \( P_c = [p_{cxi}, p_{cyi}, 1]^T \) in case of \( Z = 1 \) plane by using the internal reference 1 relationship of the pinhole projection mode.

\[
P_{wi} = CM^{-1} \times P_{ui},
\]

\[
P_{ui} = \begin{bmatrix}
  u_{xi} \\
  v_{yi} \\
  1
\end{bmatrix} = CM \times \begin{bmatrix}
  p_{uxi} \\
  p_{wyi} \\
  p_{wzi}
\end{bmatrix},
\]

\[
P_c = \begin{bmatrix}
  p_{cxi} \\
  p_{cyi} \\
  p_{czi}
\end{bmatrix} = \begin{bmatrix}
  p_{uxi} \\
  p_{wyi} \\
  p_{wzi}
\end{bmatrix},
\]

\[
\begin{align*}
  p_{uxi} &= p_{cxi} \times p_{wzi} \\
  p_{wyi} &= p_{cyi} \times p_{wzi}
\end{align*}
\]

Figure 8. Diagram of 3D to 2D projection.
Since $p_i$ was obtained from (8), we obtained the mathematical constraint relationship of $p_{wxi}$, $p_{wyi}$, $p_{wzi}$ in the feature point of $P_{wli} = [p_{wxi} \ p_{wyi} \ p_{wzi}]^T$:

The solution $p_{wzi}$ was to solve the scale $\lambda$ uncertainty problem in monocular-vision positioning.

So far, through Formula (9), we can express $P_{wli}$ as a three-dimensional vector with only one unknown quantity, $p_{wzi}$, that is, the depth value mentioned in Sections 3 and 4 $D_0$:

$$P_{wli} = \begin{bmatrix} p_{wxi} \\ p_{wyi} \\ p_{wzi} \end{bmatrix}.$$ (10)

If the target is abstracted as a parallelogram, then a reliable method to construct the constraint equations is presented. As seen in Figure 9, four vertices $P_{wji}$, $i = 1, 2, 3, 4$ are given for the parallelogram feature, and six distance constraints and two groups of parallel constraints can be given, among which the distance constraint is:

$$d_{ij}^2 = (p_{wxi} - p_{wxj})^2 + (p_{wyi} - p_{wyj})^2 + (p_{wzi} - p_{wzj})^2$$

$$= (p_{xci} \times p_{wzi} - p_{xci} \times p_{wxj})^2 + (p_{yji} \times p_{wzj} - p_{yji} \times p_{wxj})^2 + (p_{wzi} - p_{wzj})^2$$ (11)

Considering that there are four unknown depth-information values of four feature points, we needed to reasonably select and use at least four of the 12 geometric constraint equations, so as to construct the multiple quadratic well-posed or overdetermined equation set $F$ about $p_{wzi}$, which can be obtained by solving $F$. Note that $p_{wzi}$ must be positive, and all non-positive solutions in the space of $F$ solution are pseudo-solutions. From the author’s previous related work [24], we introduced the Improved Newton Iteration Method, which is an improved Newton–Raphson method with the fifth-order convergence adapted, whose iterative equation is as follows:

$$\gamma^n = x^n - \frac{F(x^n)}{F'(x^n)}.$$ (12)

$$z^n = x^n - \frac{2F(x^n)}{F'(x^n) + F''(x^n)}.$$ (13)
where $X^n$ is the vector to be solved composed of depth value $p_{wzi}$, and $Y^n$ and $Z^n$ are the intermediate variables of the iteration. By substituting the solved $p_{wzi}$ into Equation (10), the coordinates $P_{wzi}$ of the spatial-feature points of the target in the camera coordinate system can be obtained. $R$ and $T$ of the posture were obtained with the monocular VSLAM method:

$$R \times P_i + T = P_{wi} \Rightarrow P_i = R^{-1} \times P_{wi} - T,$$

where $R$ is the rotation matrix of the camera relative to the world coordinate system, and $T$ is the displacement vector of the camera relative to the world coordinate system. The coordinates $P_i$ of the target feature point in the world coordinate system can be obtained by solving the Equation (15).

In this study, the method was extended to the solution of circular targets. The properties of the circle projection are that the projection of the circle is a circle or an ellipse, and the projection of the circle is still the center of the projection. The center of the projection-surface image can be obtained with image algorithms. Following the projection properties, the line segment was obtained through the intersection of any two rays at the center, and the graph is radius $R$, as shown in Figure 10. According to the symmetry of the circle, the shape formed by the four intersections must be a parallelogram. Therefore, solving the depth problem essentially returns to the method of solving the parallelogram above.

3D Coordinate Space

![3D Coordinate Space](image)

2D Image Plane

![2D Image Plane](image)

Figure 10. Feature points projection of circular target.

In actual engineering, the side length scale of the quadrilateral obtained by this method is difficult to obtain. The diagonal distance is the radius $Rc$, which is easier to calculate:

$$|Rc| = \left| d_{ij}^2 \right| = (p_{wxi} - p_{wzj})^2 + (p_{wyi} - p_{wzj})^2 + (p_{wzi} - p_{wzj})^2$$

$$= (p_{cxi} \times p_{wzi} - p_{cxj} \times p_{wzj})^2 + (p_{cyi} \times p_{wzi} - p_{cyj} \times p_{wzj})^2 + (p_{wzi} - p_{wzj})^2$$

for $i = 1, j = 3$ or $i = 2, j = 4$.

Generally, the combination of the diagonal distance $Rc$ and the quadrilateral parallel condition equation is performed to obtain the solution depth of the equation group $F$ and thus the spatial position of the circular target. For most of the deep-solving problems of monocular vision, the target can be abstracted as basic shapes such as triangles, circles, and parallelograms. The robustness and anti-noise ability of this type of method can be significantly improved by using various geometric properties of the target (constraints of vector parallel, vector coplanar, vector vertical, etc.) to assist in constructing constraint equations. Finally, the improved gradient-descent optimization algorithm was employed to obtain the depth information of the target. Our method has the advantages of the small amount of calculation and clear physical meaning of the constructed model.
6. Experiment and Results

In this study, a series of target-detection-comparison experiments and indoor target-tracking experiments were designed for verification. A visual motion-capture system was employed to calibrate the true value of the target’s trajectory. The motion-capture system (MCS) was composed of eight cameras. It can cover the space experiment range of 4.7 m × 3.7 m × 2.6 m and can achieve posture tracking and data recording with a refresh rate of 260 Hz. The moving targets in the experiment were rectangular and circular targets, as exhibited in Figure 11.

![Figure 11](image1.png)

**Figure 11.** The type of testing targets: (a) Rectangle target with grid pattern, (b) circular target with grid pattern, (c) rectangle monochrome target, and (d) circular monochrome target. The size of various targets was about 40 cm.

6.1. Target = Detection Experiment

Experiment 1: The purpose of this experiment was to verify the target-detection performance of the FDA-SSD model. Specifically, 1200 images of the same type with depth information that had not been used for training were randomly selected from OTMP and put into the trained model. The test diagram is illustrated in Figure 12.

![Figure 12](image2.png)

**Figure 12.** Cont.
Table 1. Performance comparison of detection models based on neural network in experiment 1.

| Method    | mAP  | FPS  | Boxes | GPU      | Input Resolution |
|-----------|------|------|-------|----------|------------------|
| YOLO v1   | 97.7 | 21.2 | 98    | GTX1060  | 448 × 448        |
| YOLO v2   | 98.9 | 27.4 | 1521  | GTX1060  | 416 × 416        |
| SSD       | 98.6 | 24.3 | 8732  | GTX1060  | 300 × 300        |
| FDA-SSD   | 98.3 | 28.7 | ~dynamic | GTX1060 | 300 × 300        |

The experimental-result table suggests that the FDA-SSD can effectively match the detector under the appropriate feature scale for target detection due to the use of the depth information of the target during detection. The actual number of anchor frames used can dynamically change according to the normalized depth value provided by the 3D-positioning algorithm. Therefore, compared with the original SSD method that detects 8732 anchor frames each time, unnecessary detection processes were screened to effectively reduce the amount of calculation. This algorithm achieved an 18.1% increase in speed compared to the SSD while maintaining the same high level of detection accuracy as the original SSD.

Besides, comparing the performance of FDA-SSD and YOLO v2 on the OTMP dataset, it can be seen that they are very close in speed and accuracy. Considering that YOLO v2 uses Darknet-19 to achieve a smaller computational load, FDA-SSD uses the spatial information of samples to help the original SSD model to reduce the computational load of the entire detection model from another perspective and thus saves computational resources.

6.2. Target-Tracking Experiments

Experiment 2: The purpose of this experiment was to verify the comprehensive performance of the target-tracking algorithm combined with the FDA-SSD detection model for moving target recognition, positioning, and tracking under the static conditions of the camera. In this experiment, the monocular camera was fixed, and the circular grid target moved randomly indoors. The algorithm calculated the spatial-trajectory coordinates of the output target in real-time, which were compared with the measured values of the motion-
capture system. The following Figure 13 shows the comparison results of the experimental trajectory.

![Figure 13](image)

**Figure 13.** (a) Target tracking results of experiment 2: Comparison of motion trajectories under different depth values, (b) trajectory comparison in x-axis, (c) trajectory comparison in y-axis, (d) trajectory comparison in z-axis.

Figure 13 and Table 2 reveal that our mono-visual target-tracking framework in this study can well realize the dynamic detection and tracking of the indoor target by using the FDA-SSD model. Compared with the ground truth measured by the motion-capture system, the actual tracking RMSE of the tracking algorithm on the three coordinate axes x, y, and z was less than 4.17 cm. Experiment 2 demonstrated that the algorithm in this study can effectively realize the detection and 3D tracking of moving targets on a static observation platform.

**Table 2.** RMSE of target tracking in Experiment 2.

| Axis | X (m)  | Y (m)  | Z (m) |
|------|--------|--------|-------|
| RMSE | 0.0402 | 0.0326 | 0.0417 |
Experiment 3: The purpose of this experiment was to verify the comprehensive performance of the target-tracking algorithm combined with the FDA-SSD detection model for moving target recognition, positioning, and tracking under the condition of camera motion. In this experiment, the monocular camera faced the tracking target and moved randomly, and the rectangular solid-color target moved randomly indoors. The algorithm output the space trajectory coordinates of the target in real-time, which was compared with the measured value of the motion-capture system. Compared with the conditions of the camera in the second experiment, the motion of the monocular camera in this experiment often brought image jitter, resulting in higher requirements for detection. Additionally, the trajectory of the camera was provided by the indoor motion-capture system. The following Figures 14 and 15 show the results of related experimental data.

Figure 14. Pose of the camera obtained by motion−capture system.

Figure 15. Cont.
It can be concluded that the FDA-SSD can effectively save computing resources and improve real-time performance compared to the original method.

Table 3 shows the root mean square error of dynamic target tracking in experiment 3:

| Axis   | X (m)  | Y (m)  | Z (m)  |
|--------|--------|--------|--------|
| RMSE   | 0.0392 | 0.0372 | 0.0472 |

The data in Figures 14 and 15 and Table 3 indicate that our mono-visual target-tracking framework in this study can well realize the dynamic detection and tracking of the in-door target by using the FDA-SSD model. Compared with the ground truth of the motion-capture system, the actual tracking RMSE of the tracking algorithm on the three coordinate axes x, y, and z was less than 4.72 cm. The overall tracking process was stable, and there was no tracking failure. Experiment 3 reflects that the algorithm in this study can still effectively realize the detection and 3D tracking of moving targets on the dynamic detection platform.

We also analyzed the computational timing performance of the object-tracking experiments in Experiment 3. We conducted three sets of localization experiments, using YOLO v2 [28], SSD, and FDA-SSD to test the same samples in the OTMP dataset. Considering that the solution of the camera pose in this study was given by the indoor positioning system, we focused on analyzing the timing performance of the detection module and the spatial positioning module, where the Timing for Detection refers to the time-consuming of the neural-network-based detection model; the Timing for Location refers to the time-consuming of the 3D-positioning algorithms.

It can be seen from Table 4 that the 3D-positioning algorithm used in our framework takes only less than 3 ms, which only takes up very little computing resources compared to the detection model. Therefore, improving the operation speed of the detection model is particularly important to save the memory and computing resources of the entire system. Compared with the SSD, the FDA-SSD used in this study improved the operation speed by 16.5%, which is consistent with the conclusions in experiment 1. It can be concluded that the FDA-SSD can effectively save computing resources and improve real-time performance compared to the original method.
Table 4. Comparison of timing performance in experiment 3.

| Method      | Timing for Detection (ms) | Timing for Location (ms) | GPU          | Input Resolution |
|-------------|---------------------------|--------------------------|--------------|------------------|
| Yolo v2     | 37.15                     | 2.63                     | GTX1060      | 416 × 416        |
| SSD         | 41.73                     | 2.30                     | GTX1060      | 300 × 300        |
| FDA-SSD     | 34.82                     | 2.55                     | GTX1060      | 300 × 300        |

7. Conclusions

An efficient target-detection network structure, namely, FDA-SSD, was proposed in this study to achieve efficient and fast target-detection problems. The key idea of FDA-SSD is to train samples with the normalization depth. Firstly, we proposed a method for building OTMP datasets with spatial parameters and open-source datasets. Then, we proposed a fusion model of FFN and SSD, which realizes the data fusion of sample image information and spatial normalization parameters of the model in the training stage, so that the model can filter detectors of different scales. Finally, the appropriate scale in the SSD was selected for detection to significantly reduce the number of calculations. Based on FDA-SSD, a mono-visual positioning-and-tracking framework integrating target feature detection and dynamic target tracking was designed.

Indoor experiments verified that the speed under the computing platform based on GTX1060 was about 18.1% faster than the original SSD, while the accuracy of FDA-SSD maintained a consistently high level. The method proposed in this study was not limited to monocular-vision applications. In the future, it would be extended to binocular and depth-camera models and implanted in various applications such as obstacle avoidance, detection, and follow-up of various UAVs and self-driving cars.

Author Contributions: Conceptualization, Z.W.; Data curation, Z.W.; Formal analysis, Z.W. and S.Y.; Investigation, S.Y. and M.S.; Methodology, Z.W.; Project administration, K.Q.; Resources, Z.W.; Software, Z.W.; Supervision, K.Q.; Visualization, Z.W.; Writing—original draft, Z.W.; Writing—review and editing, Z.W. and S.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Science and Technology Department of Sichuan Province under Grant No. 2020YJ0044, 2021YFG0131, the Fundamental Research Funds for the Central Universities under Grant No. ZYGX2020J020, and the National Numerical Wind Tunnel Project, China under Grant No. NNW2021ZT6-A26.

Data Availability Statement: The complete dataset of benchmarks for object tracking with motion parameters (OTMP) is available at 201611191002@std.uestc.edu.cn. Part of the dataset has been uploaded to GitHub: https://github.com/6wa-car/OTMP-DataSet.git (accessed on 30 December 2021).

Conflicts of Interest: The authors declare that there are no conflict of interest regarding the publication of this study.

References

1. Guo, Q.; Feng, W.; Zhou, C.; Huang, R.; Wan, L.; Wang, S. Learning Dynamic Siamese Network for Visual Object Tracking. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 1763–1771.
2. Lin, T.-Y.; Dollar, 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, Honolulu, HI, USA, 21–26 July 2017; pp. 2117–2125. [CrossRef]
3. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; Berg, A.C. Ssd: Single shot multibox detector. In European Conference on Computer Vision; Springer: Cham, Switzerland, 2016; pp. 21–37.
4. Cai, H.; Gan, C.; Wang, T.; Zhang, Z.; Han, S. Once-for-all: Train one network and specialize it for efficient deployment. arXiv 2019, arXiv:1908.09791.
5. Ma, X.; Wang, Z.; Li, H.; Zhang, P.; Ouyang, W.; Fan, X. Accurate Monocular 3D Object Detection via Color-Embedded 3D Reconstruction for Autonomous Driving. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea, 27–28 October 2019; pp. 6851–6860.
6. Singandhupe, A.; La, H.M. A Review of SLAM Techniques and Security in Autonomous Driving. In Proceedings of the 2019 Third IEEE International Conference on Robotic Computing (IRC), Naples, Italy, 25–27 February 2019; pp. 602–607.

7. Chang, M.-F.; Ramanan, D.; Hays, J.; Lamb, J.; Sangkloy, P.; Singh, J.; Bak, S.; Hartnett, A.; Wang, D.; Carr, P.; et al. Argoverse: 3D Tracking and Forecasting with Rich Maps. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 8746–8757.

8. Hu, H.N.; Cai, Q.Z.; Wang, D.; Lin, J.; Sun, M.; Krahenbuhl, P.; Darrell, T.; Yu, F. Joint monocular 3D vehicle detection and tracking. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27–28 October 2019; pp. 5390–5399.

9. Luo, W.; Yang, B.; Urtasun, R. Fast and Furious: Real Time End-to-End 3D Detection, Tracking and Motion Forecasting with a Single Convolutional Net. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–20 June 2018; pp. 3569–3577.

10. Wang, Z.H.; Chen, W.J.; Qin, K.Y. Dynamic Target Tracking and Ingressing of a Small UAV Using Monocular Sensor Based on the Geometric Constraints. Electronics 2021, 10, 1931. [CrossRef]

11. Wu, Y.; Lim, J.; Yang, M.-H. Online Object Tracking: A Benchmark. In Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2013; pp. 2411–2418.

12. Moudgil, A.; Gandhi, V. Long-Term Visual Object Tracking Benchmark. In Asian Conference on Computer Vision; Springer: Cham, Switzerland, 2018; pp. 629–645.

13. Kristan, M.; Leonardis, A.; Matas, J.; Felsberg, M.; Pfugfelder, R.; Kämäräinen, J.-K.; Danelljan, M.; Zajeć, L.Č.; Lukežič, A.; Drbohlav, O.; et al. The Eighth Visual Object Tracking VOT2020 Challenge Results. In European Conference on Computer Vision; Springer: Cham, Switzerland, 2020; pp. 547–601.

14. Müller, M.; Bibi, A.; Giancola, S.; Alsabahi, S.; Ghanem, B. TrackingNet: A Large-Scale Dataset and Benchmark for Object Tracking in the Wild. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 300–317.

15. Dong, X.; Shen, J. Triplet Loss in Siamese Network for Object Tracking. In Proceedings of the Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support, Munich, Germany, 8–14 September 2018; pp. 459–474.

16. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788. [CrossRef]

17. Joglekar, A.; Joshi, D.; Khemani, R.; Nair, S.; Sahare, S. Depth estimation using monocular camera. Int. J. Comput. Sci. Inf. Technol. 2011, 2, 1758–1763.

18. Pizzoli, M.; Forster, C.; Scaramuzza, D. REMODE: Probabilistic, monocular dense reconstruction in real time. In Proceedings of the 2014 IEEE International Conference on Robotics and Automation (ICRA), Hong Kong, China, 31 May–7 June 2014; pp. 2609–2616.

19. Choi, S.; Min, D.; Ham, B.; Kim, Y.; Oh, C.; Sohn, K. Depth Analogy: Data-Driven Approach for Single Image Depth Estimation Using Gradient Samples. IEEE Trans. Image Process. 2015, 24, 5953–5966. [CrossRef] [PubMed]

20. Eigen, D.; Puhrsch, C.; Fergus, R. Depth map prediction from a single image using a multi-scale deep network. arXiv 2014, arXiv:1406.2283.

21. Eigen, D.; Fergus, R. Predicting Depth, Surface Normals and Semantic Labels with a Common Multi-scale Convolutional Architecture. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 2650–2658.

22. Laina, I.; Rupprecht, C.; Belagiannis, V.; Tombari, F.; Navab, N. Deeper Depth Prediction with Fully Convolutional Residual Networks. In Proceedings of the 2016 Fourth International Conference on 3D Vision (3DV), Stanford, CA, USA, 25–28 October 2016; pp. 239–248. [CrossRef]

23. Xu, D.; Ricci, E.; Ouyang, W.; Wang, X.; Sebe, N. Multi-scale Continuous CRFs as Sequential Deep Networks for Monocular Depth Estimation. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 5354–5362.

24. Qin, T.; Li, P.; Shen, S. VINS-Mono: A Robust and Versatile Monocular Visual-Inertial State Estimator. IEEE Trans. Robot. 2018, 34, 1004–1020. [CrossRef]

25. Campos, C.; Elvira, R.; Rodriguez, J.J.G.; Montiel, J.M.M.; Tardos, J.D. ORB-SLAM3: An Accurate Open-Source Library for Visual, Visual–Inertial, and Multimap SLAM. IEEE Trans. Robot. 2021, 37, 1874–1890. [CrossRef]

26. Wang, Z.H.; Qin, K.Y.; Zhang, T.; Zhu, B. An Intelligent Ground-Air Cooperative Navigation Framework Based on Visual-Aided Method in Indoor Environments. Unmanned Syst. 2021, 9, 237–246. [CrossRef]

27. Wang, Z.H.; Zhang, T.; Qin, K.Y.; Zhu, B. A vision-aided navigation system by ground-aerial vehicle cooperation for UAV in GNSS-denied environments. In Proceedings of the 2018 IEEE CSAA Guidance, Navigation and Control Conference (CGNCC), Xiamen, China, 10–12 August 2018; pp. 1–6.

28. Redmon, J.; Farhadi, A. YOLO9000: Better, Faster, Stronger. In Proceedings of the 30th IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7263–7271. [CrossRef]