*Computer Systems Science & Engineering*
DOI: 10.32604/csse.2023.032904
***Article***

Tech Science Press



# Visual Object Tracking Based on Modified LeNet-5 and RCCF

**Aparna Gullapelly and Barnali Gupta Banik***

Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Deemed to be University, Hyderabad, 500075, Telangana, India
*Corresponding Author: Barnali Gupta Banik. Email: barnali.guptabanik@ieee.org
Received: 01 June 2022; Accepted: 29 September 2022

**Abstract:** The field of object tracking has recently made significant progress. Particularly, the performance results in both deep learning and correlation filters, based trackers achieved effective tracking performance. Moreover, there are still some difficulties with object tracking for example illumination and deformation (DEF). The precision and accuracy of tracking algorithms suffer from the effects of such occurrences. For this situation, finding a solution is important. This research proposes a new tracking algorithm to handle this problem. The features are extracted by using Modified LeNet-5, and the precision and accuracy are improved by developing the Real-Time Cross-modality Correlation Filtering method (RCCF). In Modified LeNet-5, the visual tracking performance is improved by adjusting the number and size of the convolution kernels in the pooling and convolution layers. The high-level, middle-level, and handcraft features are extracted from the modified LeNet-5 network. The handcraft features are used to determine the specific location of the target because the handcraft features contain more spatial information regarding the visual object. The LeNet features are more suitable for a target appearance change in object tracking. Extensive experiments were conducted by the Object Tracking Benchmarking (OTB) databases like OTB50 and OTB100. The experimental results reveal that the proposed tracker outperforms other state-of-the-art trackers under different problems. The experimental simulation is carried out in python. The overall success rate and precision of the proposed algorithm are 93.8% and 92.5%. The average running frame rate reaches 42 frames per second, which can meet the real-time requirements.

**Keywords:** Object tracking; correlation filters; feature extraction; experimental results; semantic information

## 1 Introduction

In image processing and computer vision, visual object tracking is an important field of research with several applications in computer vision, human-computer interaction, and CCTV surveillance. The visual object tracking objective is to discover and localize a target of interest in the successive video frames [1]. In this field, advancements have been made through years of scientific research [2]. Due to this visual object tracking, a possibility of achieving multiple difficulties including blur, partial occlusion, camera

motion, scale variation, illumination variation, and background clutter. In unconstrained situations, achieving efficient and reliable tracking remains a difficult task [3].

Several tracking algorithms have been developed over the last few years. There are two types of approaches: discriminative and generative approaches [4,5]. As the target location, the tracker determines the best matching region using a generative method, which establishes a target appearance model [6]. The visual tracking is done by the discriminative method in contrast. The classifier is trained using target and background training samples with the vision of making separation among samples [7].

The Discriminative Correlation Filters (DCF) tracking paradigm was created and intensively investigated to reduce the tracking difficulties of standard visual object tracking systems [8,9]. The circulant matrix property ensures the performance of its training and localization phases, which involve entire circularly enhanced samples [10]. Discrete Fourier Transform (DFT) in the frequency domain with closed-form solutions speeds up the correlation operator development, which is represented as a ridge regression problem [11].

Multiple trackers in recent decades have achieved superior performance on benchmarking datasets and in competitions [12]. Furthermore, visual object tracking performance is improved by establishing feature extraction. Feature extraction is the most essential factor for object tracking [13]. When compared to the features of a deep Convolutional Neural Network (DCNN) with descriptors of the conventional images, the features are more effective and intuitive. In the DCF model, several deep features are merged for the object tracking in the video [14,15]. There has not been research into the relationships between several deep channels. In the training framework, the group relationships among deep image features are identified by researching the significance of multi-channel features with high dimensions to achieve adaptive channel selection as a result of this research [16].

In visual tracking, Correlation Filters (CFs) are now commonly utilized due to the significant computing power [17]. Multiple research has been explored that use CFs. Other than generative techniques, tracking algorithms use discriminative techniques such as CFS. The classification problem is considered by discriminative trackers [18]. The relationship between the last frame's target model and the present frame's unknown target samples is calculated by using the CFs in the visual object tracking and also the foreground and background are distinguished using CFS [19]. By using single CFs in the visual object tracking, it is ineffective for dealing with fast motion (FM) despite its advantages. Simultaneously, a linear method is used to combine some multiple correlation filter models, which has the drawback of making the tracking algorithms vulnerable to problems like as quick motion and target DEF. To solve these problems, proposed different trackers of deep learning-based but still it has a complex computation limit [20].

In the field of computer vision, visual tracking has attracted a lot of interest since the development of camera technology. A variety of DCF techniques are frequently utilized in tracking, however, the majority of them consistently fail to locate the target in difficult circumstances, failing during the period of the sequences. Some of the existing trackers fail due to motion blur (MB), FM, and occlusions. To handle these issues, propose a different tracking technique that depends on the Modified LeNet-5 feature extraction and RCCF for real-time object tracking for robust object tracking.

This research aims to enhance tracking accuracy while maintaining a reasonable tracking performance under several difficulties. We develop the visual tracking algorithm depending on the Modified LeNet-5 for feature extraction and RCCF. The tracking accuracy is improved by using the RCFF, and increasing the tracking performance by combining a feature fusion factor with a scale estimation factor. As a result, in situations of changes of scale, FM, and DEF, Modified LeNet-5-RCFF can achieve good tracking accuracy.

The novelty of the work is as follows,

- Features are extracted by using the Modified LeNet-5 model. The target appearance model is represented by feature selection and is based on the middle and high network layers. The different

information of the target is expressed by different features, the overall target appearance is represented by combining the different features, which has improved the tracking performance.

- Then propose a novel RCCF with complex visual changes, unlike previous DCF-based trackers. The function of RCFF is applying correlation filtering in extracted features and generating response maps to enhance visual object tracking.
- On the OTB50 and OTB100 benchmarks, we perform extensive experiments. The results of the experiment reveal that the proposed tracker performed successfully compared with other state-of-the-art trackers.

The remaining of this research is structured as follows. The related work relevant to the proposed system is discussed in Section 2. The detailed representation of the proposed model and the optimization algorithm is described in Section 3. The experimental results and the performance comparison with various representative and correlative trackers are described in Section 4. Section 5 concludes the paper.

## 2 Related Works

For real-time object tracking, a new dual-template adaptive correlation filter is proposed by Yan et al. [21]. Different size-level templates are developed first in this paper. Based on the target response confidence, the optimal template was established at second during target translation estimation. The template switching threshold is changed during the object tracking process under each frame's output response confidence. This can improve the tracker's ability to adapt to diverse video sequences. Lastly, the Kernelized Correlation Filter (KCF) is developed by integrating the feature fusion factor, scale estimation factor, and dual templates for increasing the tracking performance.

A novel DCF-based tracking system was investigated by Xu et al. [22]. The temporal consistent constraints and adaptive spatial feature selection are two fundamental novelties of the proposed technique, the joint spatial-temporal filter learning is also enabled by this tracker in a lower-dimensional discriminative manifold. More particularly, multi-channel filers were exposed to structured spatial sparsity requirements. As a result, the lasso regularization can approximate the process of learning spatial filters. The global structure is preserved in the manifold by limiting the filter model to a range of values around its historical value and is changed locally for ensuring temporal consistency. Finally, the temporal consistency is selected by proposing a unified optimization framework with preserves the spatial features and developing the discriminative filters with the augmented Lagrangian method.

For visual tracking, the different multiple feature fused model into a correlation filter framework is proposed by Yuan et al. [23] for improving the tracking effectiveness and reliability of the tracker. The model of correlation filter generates different response maps for each feature in different tracking conditions. The noise interference in different features is eliminated by using an adaptive weighting function depending on the corresponding response maps of each feature while maintaining their benefits. Moreover, the fast training and accurate location of the object are provided by using this correlation filter. Furthermore, the scale variation of the target is appropriately handled by developing the effective scale variation detection method in visual object tracking.

Siamese Region Proposal Network (SiamRPN) is proposed by Jain et al. [24] that was described as an offline network with a large dataset. The features are extracted by using multiple sub-networks that are present in this network, such as classification and regression. The visual tracking analysis is enhanced by optimizing the feature extraction model through the combination of the Siamese-RPN++ network and You Only Look Once, Version 3 (Yolo-v3). Previously used recognition frameworks repurposed classifiers or localizers to extract features. Even while object scaling, implements the model to various areas of an image.

An alternate description of visuals was proposed by Zhu et al. [25]. From a target with rapid appearance changes, the dynamic appearance information is extracted by proposing the collaborative representation

between successive frames specifically, this has the effect of reducing the background's undesirable impact. A spatially regularized DCF framework is used to combine the resulting collaborative representation coefficients with the original feature maps for better performance. In this paper, a new feature integration object tracker is proposed by Zhang et al. [26], called correlation filters and online learning (CFOL). The location of the target and their corresponding correlation score is estimated by using the CFOL with the same multi-feature DCF. For online learning, the tracking drifts are reduced by developing the new sampling and updating technique.

A cascade-based tracking algorithm is proposed by Lu et al. [27] for improving the tracker's robustness and reducing time consumption. Features are extracted by proposing a new deep network first. The feature extraction stage can achieve speeds of up to 50 frames per second due to the use of specific pruning algorithms. Moreover, using more spatial information, the performance of tracking is improved by introducing a Dense Connection-based Cascade Tracker (DCCT). The proposed DCCT tracker, like the cascade classifier, is made up of numerous weaker trackers. The false candidates of the tracked object are rejected by each weak tracker and integrated with these weak trackers for obtaining the final tracking results.

## 3 Proposed Methodology

The target object is tracked more accurately across the entire frame while avoiding high overflow, this is the main aim of our system. For tracking objects in a video frame, the system uses Modified LeNet-5 and RCCF methodologies with various challenge factors including Low Resolutions (LR), Out of Views (OV), Scale Variations (SV), MB, Out of Plane Rotations (OPR), Occlusions, Illumination Variations (IV), FM, DEF, In-Plane Rotations (IPR), and Background Clutter (BC). The schematic diagram of the proposed method is given in Fig. 1.

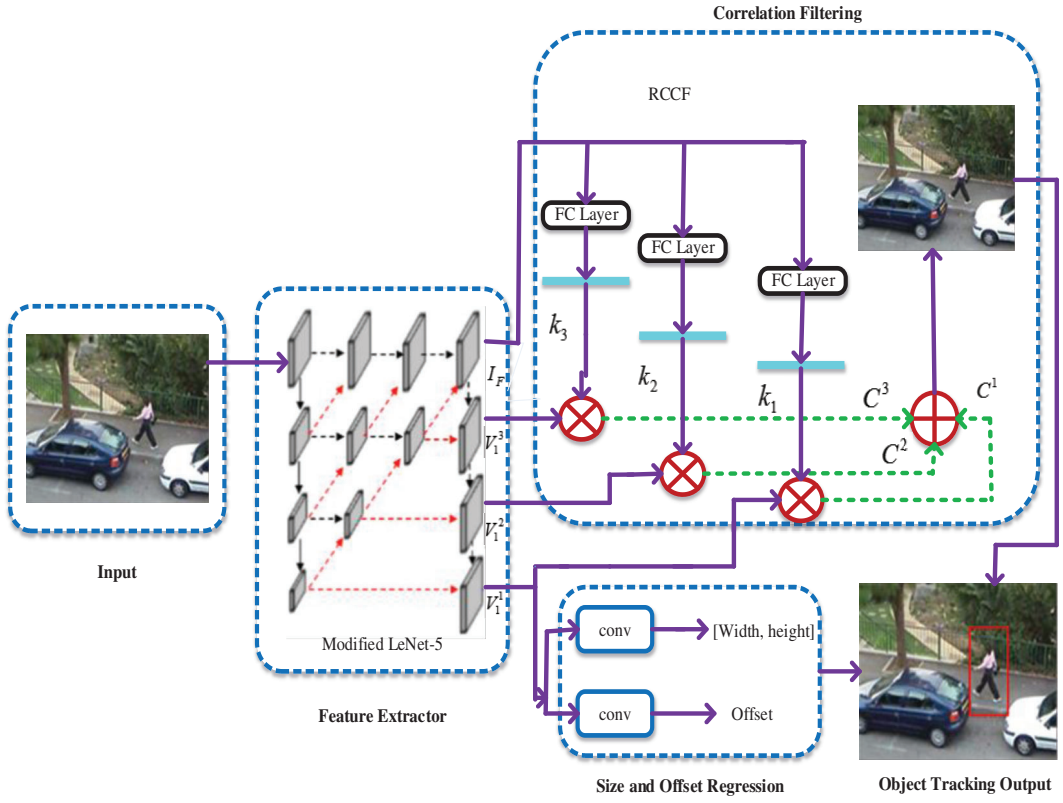


**Figure 1:** Schematic diagram of our proposed methodology

The target appearance model is represented by proposing a Modified LeNet-5 for feature extraction and the extracted features are selected based on the middle and high network layers. The different information of the target is expressed by different features and the overall appearance of the target is represented by combining these features for improving the performance of the tracking. For visual tracking with complex appearance changes, we propose a new selective correlation filter model termed RCCF for object tracking in the crowd that contributes to improved tracking. The correlation filter generates a response map corresponding to each feature in the RCCF algorithm. The noise interference in different features is eliminated by using an adaptive weighting function depending on the corresponding response maps of each feature while maintaining their benefits. It effectively improves the tracker's tracking performance and reliability.

After feature extraction, three different filter kernels are used to map the extracted feature by using the function of RCFF. With the corresponding kernel on three levels of image features, the three correlation maps are generated by performing correlation filtering. Lastly, we employ pixel-wise averaging to combine the three correlation maps. The fused heat map's peak value corresponds to the central point. Then performing the Size and Offset Regression, only the last-level image feature is used to regress the local offset and two-dimensional object size for the center point. Combining the local offset, estimated center point of the object, and size of the object provides the target object region.

### *3.1 Feature Extraction using Modified Lenet-5*

From the visual data, the object is extracted in the form of frame sequences, which is called Visual tracking. It is a popular method for identifying an object when the appearance of the object changes. And the target object is tracked across multiple frames without knowing anything about the target object. However, the extraction model recognizes the initial frame sequence and based on that, the model is effectively followed from the first to the last frame. An original video is divided into several frames, and the target area has been manually modified in the LeNet-5 network also known as a region of interest or an area of interest. This method depends on the model of cross-correlation which is the local feature extraction technique embedded with appropriate classes. Using LeNet-5, the proportional region is obtained by comparing the appropriate frames. This network generates feature templates, which are then compared to the object feature to improve model tracking.

The object is also tracked with different scores by using different bounding boxes. The object's specific score is generated by the system which is reciprocated with 2k logits. The object is classified by the trained network using full proportional segmentation and binarization and LeNet has been upgraded with LeNet-5, which eliminates network faults and the features are analyzed by exploring more strategic behavior for improved object detection.

The LeNet-5 was made up of seven layers such as two pooling layers, two fully connected layers, two convolution layers, an input layer, and an output layer. An input image $\{A_i,\ b_i\}_{i=1}^{N}$ is taken formally, where the original image data is represented by $A_i$, and the image's class category (i.e., 0 and 1) is represented by $b_i$. The categorical cross-entropy function is used to calculate the difference between the actual label $b_i$ and the predicted label $\hat{b_i}$, it is given by:

$$F(s,\ t) \triangleq -\frac{1}{N}\sum_{l=1}^{N} bl1\ \hat{\log}\ \hat{bl1} + \ldots . + blk\ \log\ \hat{blk} \tag{1}$$

The biases and weights of the conventional LeNet-5 network layers are represented by $t$ and $s$, respectively. The range of class category is depicted by K and the softmax value of the $k$'s class category corresponds to $\hat{blk}$, defined as:

$$\overset{\wedge}{blk} = \text{softmax}(z_k) = \frac{e^{zk}}{\sum_{i=1}^{K} e^{z_i}} \tag{2}$$

where the last fully connected layer's corresponding $i'th$ class category result is defined $z_i$. The loss derivative is back-propagated (BP) for factors over the whole network and was used to learn the convolutional operation's weight and bias parameters.

In the Modified LeNet-5 network, the fully-connected layer nodes and convolution layer strides are not suitable for feature maps operation. Therefore, we made the following changes to LeNet-5: (1) Rather than a 2-dimensional convolution operation, feature extraction in the network utilizing a one-dimensional convolution operation; (2) The over-fitting problem is solved by an additional dropout layer between the fully connected and the convolution layers; (3) The network complexity is reduced by maintaining only one fully connected layer (4) adjusting the fully-connected layer node's range and the convolution layer stride's size. In the modified LeNet-5, the feature maps were increased layer by layer, and all convolution layer strides were changed to two when compared to the standard LeNet-5. Between the fully connected layer and convolution layer, a 0.8 drop rate of the dropout layer was added in particular and the binary classification problem is solved by reducing the nodes in the output layer from ten to two.

### *3.2 Correlation Filtering using RCCF Method*

The discrimination between images and their changes is determined by performing correlation filtering through linear template training. In several computer vision fields, correlation filtering is generally implemented. The classification of objects can be considered as a correlation filtering operation, where the filter kernel is the image feature vector output that performs correlation filtering on the last multi-layer perceptron's weight matrix. It attempts to locate an object in a video based on its location in the first frame for single object tracking, when comparing the initial frame to the others, correlation filtering is helpful.

The proposed RCCF approach for visual object tracking is described in this section. Our goal is to determine the exact location of the object depicted in the image without the process of generating proposals. We first perform Image feature correlation filtering with an image-guided filter kernel to locate the object center point in the process of RCCF. After that, the size of the object and offset of the center point are then regressed using a regression module. The target bounding box is formed by combining the correlation heat map's peak value, regressed object size, and center point offset.

#### *3.2.1 Framework*

Assume Q is the object of the image and the image height H and image width W are denoted by the $I \in R^{S\times\ T\times\ 3}$. The object size $(s_u,\ t_u)$ and the center point $(a_u,\ b_u)$ represent the target object region. There are two modules in the proposed RCCF i.e., correlation filtering and regression modules of size and offset. $I_F$ and $V_1$ represents the middle-level and high-level features respectively. The $M(\cdot)$ shows the cross-modality mapping function then it maps the middle-level feature into the visual domain. The $Map(I_F)$ mapping result as the filter (kernel) with $V_1$ a visual feature map, is used to convolve the correlation filtering module and generates the heat map $Corr \in U\frac{T}{d}x\frac{S}{d}$, where the output stride is represented by d. The $(x,\ y)$ represents object point expressed by the target is indicated by the correlation peak value. Furthermore, the center point's local offset $(\alpha a,\ \ \alpha b)$ and object size $(s,\ t)$ is predicted by the regression module of the size and offset.

The three levels of visual features are used in this process. The three-level features $[V_I^1,\ V_I^2,\ V_I^3] = V(I)$ from Modified Lenet-5 are extracted and converted into a uniform size $\frac{T}{d} \times \frac{S}{d}$

from $\frac{T}{8d} \times \frac{S}{8d} \times \frac{T}{4d} \times \frac{S}{4d}$ and $\frac{T}{2d} \times \frac{S}{2d}$. All 3-level features are used, when computing the correlation map $\widehat{C}$ because the size $[V_I^1,\ V_I^2,\ V_I^3]$ is all $64 \times \frac{T}{d} \times \frac{S}{d}$. For computational efficiency, only the $V_I^1$ highest resolution is used during the regression process. The two modules will next be described in detail.

*3.2.2 Cross Modality Correlation Filtering*

The method is used to locate the target box center $(x,\ y)$. The image-guided kernel generation, correlation operation, and correlation map fusion are the major three steps in RCCF. At first, $[k_1,\ k_2,\ k_3] = [Map_1(I_F),\ \ Map_2(I_F),\ \ Map_3(I_F)]$ is generated from the expression feature $I_F$ using three different linear functions.

From the visual space of the target, the cross-modality mapping function projects using the three connected layers $[Map_1(I_F),\ \ Map_2(I_F),\ \ Map_3(I_F)]$. A feature vector represents each kernel with a vector size of 64 dimensions. After that, it's reshaped into a $64 \times 1 \times 1$ filter for further process. On the three levels of visual features, the correlation operations are performed by using mapping kernels $[C^1,\ C^2,\ C^3] = [k_1 * V_I^1,\ \ k_2 * V_I^2,\ \ k_3 * V_I^3]$, where the convolution operation is denoted as $*$. Finally, pixel-wise averages of the three correlation maps are input into an activation function $\overset{\wedge}{C} = Sig \bmod \left(\frac{C^1 + C^2 + C^3}{3}\right)$. C1, C2, and C3 are all $R^{\frac{T}{d} \times \frac{S}{d}}$ in size. The target object's center point is the location with the highest $\overset{\wedge}{C}$ score. An output stride d is used to calculate the low-resolution equivalent $(a^g,\ b^g) = \left[\frac{(\widehat{a}^g,\ \widehat{b}^g)}{d}\right]$ for the center point of the ground truth $(\widehat{a}^g,\ \widehat{b}^g)$. In the heat map $C \in [0,\ 1]^{\frac{W}{d} \times \frac{H}{d}}$, the ground truth center point is splatted by using the Gaussian kernel $C_{ab} = \exp\left(-\frac{(a - a^g) + (b - b^g)^2}{2\sigma_t^2}\right)$. At the spatial location $(a,\ b)$, the value of $C$ is denoted by $C_{ab}$ the object size's standard deviation $\sigma_t$. A regression process is done pixel-wise with focal loss that is penalty-reduced, which is the training objective.

$$L_C = -\sum_{ab} \left\{ \begin{array}{ll} (1 - C_{ab})^{\alpha}\ \log\ (\widehat{C}_{ab}) & \text{if } C_{ab} = 1 \\ (1 - C_{ab})^{\beta}\ (\widehat{C}_{ab})\ \log\ (1 - \widehat{C}_{ab}) & \text{otherwise} \end{array} \right\} \tag{3}$$

ReLU on a 3-dimensional convolutional layer is used in both the offset and size regression branches. During training, used the $Loss_1$ loss function. The $Loss_{off}$(local offset regression loss) and $Loss_{size}$ (object size regression loss) are defined as:

$$Loss_{off} = \left|\widehat{\Delta} a_{a^g b^g} - \delta a^g\right| + \left|\widehat{\Delta} b_{a^g b^g} - \delta b^g\right| \tag{4}$$

$$Loss_{size} = \left|\widehat{S}_{a^g b^g} - s^g\right| + \left|\widehat{T}_{a^g b^g} - t^g\right|$$

where the target box's height and width for ground truth are represented by $t^g$ and $s^g$, the ground truth offset vector is denoted by $\delta a^g = \left(\frac{a^g}{d} - a^g\right)$ and $\delta b^g = \left(\frac{b^g}{d} - b^g\right)$. While $\widehat{H}_{a^g b^g},\ \ \widehat{\Delta} x_{a^g y b^g},\ \ $ and $\widehat{\Delta} y_{a^g b^g}$ are defined similarly. All other locations are ignored by the regression loss, which processes at the center point location $(a^g,\ b^g)$. The weighted sum of three-loss terms provides the final loss, which is given in below equation

$$Loss = L_C + \lambda_{size} Loss_{size} + \lambda_{off} Loss_{off} \tag{5}$$

Both values of $\lambda_{size}$ and $\lambda_{off}$ are set to 1, and a normalized object size coefficient is represented by $\lambda_{size}$. The target center point $(a_t,\ b_t)$ in the heat map $\widehat{C}$ is selected based on the highest confidence score during inference. The target offset and size are calculated using the relevant location in the $\widehat{\Delta} a_{a_t, b_g}$, $\widehat{\Delta} b_{a_t, b_t}$, $\widehat{\Delta}_a$ and $\widehat{\Delta}_b$ as $\widehat{S}_{a_t,b_t}$, $\widehat{T}_{a_t,b_t}$, $\widehat{S}$, $\widehat{T}$. The bottom right and top left corner coordinates of the target box are calculated by:

$$a_t + \widehat{\Delta}\, a_{a_t, b_t} - \frac{\widehat{S}_{a_t,b_t}}{2},\ b_t + \widehat{\Delta}\, b_{a_t, b_t} - \frac{T_{a_t, b_t}}{2},\ a_t + \widehat{\Delta}\, a_{a_t, b_t} + \frac{\widehat{S}_{a_t,b_t}}{2},\ b_t + \widehat{\Delta}\, b_{a_t, b_t} - \frac{T_{a_t, b_t}}{2} \tag{6}$$

## 4 Experimental Results

We evaluate our tracker using a standard visual tracking benchmark to objectively and properly examine the proposed tracker. We describe the algorithm flow as well as the experimental details and specifics. Then we provide the experimental evaluation's details and standards. Finally, the OTB50/OTB100 benchmarks, which contain over 100 test video sequences, are used to validate the performance of our tracker. And our methodology is implemented in Python.

### *4.1 Datasets*

Well-known benchmarks such as OTB50 and OTB100 are examined to assess the performance of our tracking approach. OTB50 and OTB100 datasets are commonly utilized. The dataset contains 11 attributes including LR, OV, SV, MB, OPR, Occlusions, IV, FM, DEF, IPR, and BC. There are many sequences in each of them.

### *4.2 Evaluation Metrics*

The tracking algorithms are evaluated by the OTB dataset on two levels such as success plot and precision plot. The average Euclidean distance between the ground truth and target prediction center location is denoted by CLE. In the object tracking in videos, the percentage of the entire frame is used to calculate the Distance Precision (DP), with comparing the set threshold, the CLE was smaller in the tracking process. The DP threshold was set to 20 pixels for this research.

The one-pass evaluation (OPE) protocol function is starting with the first frame's ground truth position and it provides the performance of success rate and average precision. Based on bounding box overlap and center location error, success and precision charts are provided. The factors employed in the evaluation were the DP, Overlap Precision (OP), and Area under Curve (AUC).

The tracker's running speed is also an important factor to consider while evaluating its performance. Running speed is measured in frames per second (FPS).

### *4.3 Implementation Results*

In the first frame, the different features are first extracted by the specified initial bounding box, and then corresponding filters are trained using the proposed methodology. In the tracking sequences, the tracker is then iterated over each frame. The suitable scale size and the location of the target's center position are determined by using the several features fused model in each iteration successively. The correlation filter models are finally updated progressively. The proposed methodology section shows the entire process of the method. The average frame rate for a running frame is 42 FPS.

The proposed tracker outperformed the existing trackers with multiple attributes, i.e., LR, OV, SV, MB, OPR, Occlusions, IV, FM, DEF, IPR, and BC. And also the robustness of the proposed method is validated by determining the attribute-based evaluation. While analyzing the AUC and DP scores on these attributes, the proposed tracker performed better on these factors than all competing trackers. These results demonstrate that the proposed tracker is more robust than the other trackers.

The experimental results of the proposed visual object tracking model are shown in Figs. 2 and 3 for the OTB50 and OTB100 datasets, respectively.

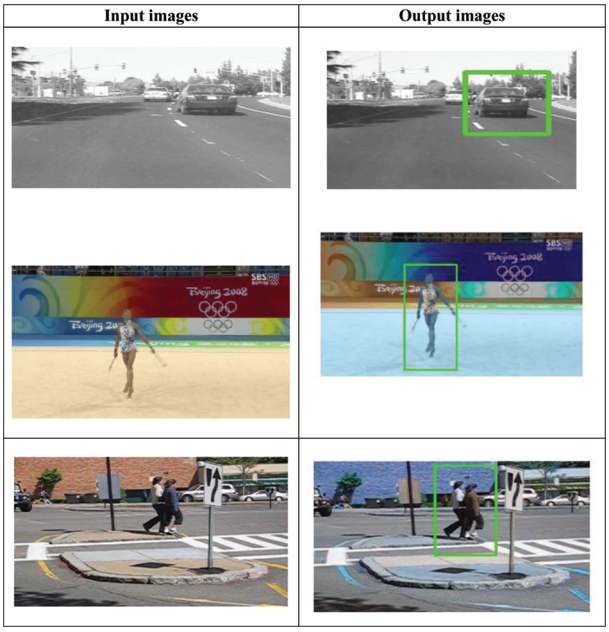


**Figure 2:** The experimental results of the proposed visual object tracking using the OTB50 dataset

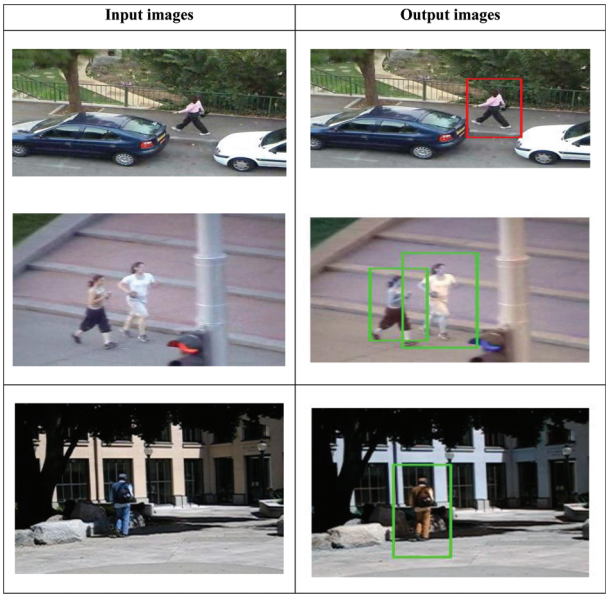


**Figure 3:** The experimental results of the proposed visual object tracking using the OTB100 dataset

### *4.4 Performance Evaluation*

Scale Adaptive with Multiple Features tracker (SAMF), KCF, Discriminative Scale Space Tracking (DSST), and Dimension Adaption Correlation Filters (DACF) are the recently proposed algorithms that are compared to our proposed methodology. For the OTB-50 and OTB100 datasets, Fig. 4 depicts the overall performance analysis, and it shows the performance of the success and precision plots. When the threshold is set to 20 pix, the algorithms are ordered by average DP in the precision plot. When the threshold is 0.5, the algorithms are sorted by AUC in the success plot.

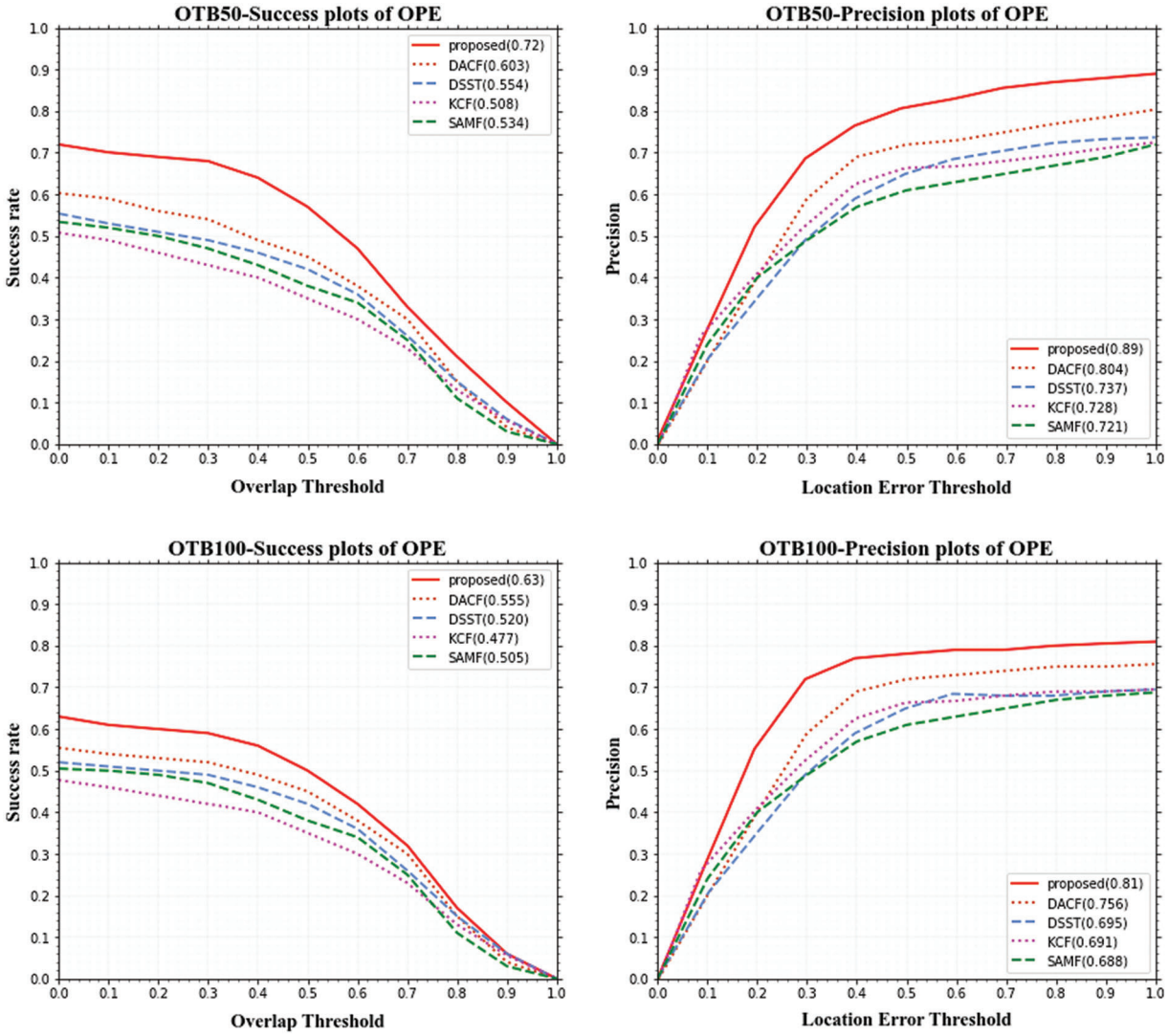


**Figure 4:** OPE success plot and precision plot on OTB-50 and OTB-100

For OTB50, the proposed method attains the success rate and precision is 0.72 and 0.89 and for OTB 100, the proposed method achieves the success rate and precision is 0.63 and 0.8, comparing the results of the proposed method with recent existing methods such as DACF, SAMF, KCF, and DSST, our tracker achieves better performance and the visual object is tacked efficiently.

The performance comparison of the proposed method and state-of-the-art tracking methods are shown in Table 1, On the OTB-50 and OTB-100 datasets, we compare the performance of our proposed tracker with state-of-the-art trackers in terms of overlap success rate (OSR) and at an overlap threshold of 0.5 and distance precision rate (DPR) at a threshold of 20 pixels in Table 1, We can observe from this table that the proposed tracker produces effective tracking results. Compared to the proposed tracker with different existing trackers SAMF, KCF, DSST, and DACF, the proposed tracker has attained enhanced performance of tracking than these trackers. These advantages are due to our hybrid Modified LeNet-5 and RCCF tracking methodology.

**Table 1:** Comparisons with state-of-the-art tracking methods on OTB50 and OTB1000

| Dataset | Evaluation criterion | State-of-the-art methods | | | | |
|---|---|---|---|---|---|---|
| | | SAMF | KCF | DSST | DACF | Proposed method (%) |
| OTB50 | Precision | 64.3 | 61.1 | 62.7 | 94.5 | 95.2 |
| | Success rate | 46.03 | 40.3 | 46.4 | 93.9 | 94.51 |
| OTB100 | Precision | 74.34 | 69.1 | 69.4 | 82.9 | 89.3 |
| | Success rate | 53.5 | 47.5 | 52 | 91.5 | 93.1 |

## 5 Conclusion

In computer vision, object tracking in visual is a popular topic. When the target deforms or moves rapidly, the existing correlation filter-based trackers frequently experience drift of the tracking box or the tracking target loss. Focusing on this problem, we proposed a tracking method based on the Modified LeNet-5 feature extraction and RCCF for real-time object tracking. The problems mentioned above can be solved with our tracking technique. The middle and high-level features of the object represent the target model, which shows it enhances the visual object tracking. Furthermore, the target is tracked by using the RCCF, which has been trained on a specific feature. These correlation filters provide complementary response maps. The final response map is improved by fusing several response maps with the RCCF method, where the final tracking result is the maximum value of the response map. A large-scale benchmark named OTB50 and OTB100 is used to conduct extensive research. The overall precision and success rate for OTB50 is 95.2% and 94.51%, and for OTB100 is 89.3% and 93.1%. The results of the experiments reveal that our proposed tracker outperforms other state-of-the-art tracking algorithms. We would refine our method in the future and investigate if this feature is suitable for other trackers.

**Funding Statement:** The authors received no specific funding for this study.

**Conflicts of Interest:** The authors declare that they have no conflicts of interest to report regarding the present study.

## References

[1] J. Yang, H. Ge, J. Yang, Y. Tong and S. Su, "Online multi-object tracking using multi-function integration and tracking simulation training," *Applied Intelligence*, vol. 52, no. 2, pp. 1268–1288, 2022.

[2] Y. Yu, L. Chen, H. He, J. Liu, W. Zhang *et al.,* "Second-order spatial-temporal correlation filters for visual tracking," *Mathematics*, vol. 10, no. 5, pp. 684, 2022.

[3] X. Cheng, Y. Zheng, J. Zhang and Z. Yang, "Multitask multisource deep correlation filter for remote sensing data fusion," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 13, no. 1, pp. 3723–3734, 2020.

[4] A. M. Roy, R. Bose and J. Bhaduri, "A fast accurate fine-grain object detection model based on YOLOv4 deep neural network," *Neural Computing and Applications*, vol. 34, no. 5, pp. 3895–3921, 2022.

[5] T. Yang, C. Cappelle, Y. Ruichek and M. El Bagdouri, "Multi-object tracking with discriminant correlation filter based deep learning tracker," *Integrated Computer-Aided Engineering*, vol. 26, no. 3, pp. 273–284, 2019.

[6] Z. Zhong, Z. Yang, W. Feng, W. Wu, Y. Hu *et al.,* "Decision controller for object tracking with deep reinforcement learning," *IEEE Access*, vol. 7, no. 1, pp. 28069–28079, 2019.

[7] Q. Liu, D. Chen, Q. Chu, L. Yuan, B. Liu *et al.,* "Online multi-object tracking with unsupervised re-identification learning and occlusion estimation," *Neurocomputing*, vol. 483, no. 1, pp. 333–347, 2022.

[8] F. Bastani, S. He and S. Madden, "Self-supervised multi-object tracking with cross-input consistency," *Advances in Neural Information Processing Systems*, vol. 34, no. 1, pp. 229–254, 2021.

[9] J. Zhang, J. Sun, J. Wang and X. G. Yue, "Visual object tracking based on residual network and cascaded correlation filters," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, no. 8, pp. 8427–8440, 2021.

[10] D. Elayaperumal and Y. H. Joo, "Aberrance suppressed spatio-temporal correlation filters for visual object tracking," *Pattern Recognition*, vol. 115, no. 1, pp. 107922, 2021.

[11] D. Yuan, X. Shu and Z. He, "TRBACF: Learning temporal regularized correlation filters for high performance online visual object tracking," *Journal of Visual Communication and Image Representation*, vol. 72, no. 1, pp. 102882, 2020.

[12] J. Fan, H. Song, K. Zhang, K. Yang and Q. Liu, "Feature alignment and aggregation siamese networks for fast visual tracking," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, no. 14, pp. 1296–1307, 2020.

[13] L. Zhou, J. Li, B. Lei, W. Li and J. Leng, "Correlation filter tracker with sample-reliability awareness and self-guided update," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 5, no. 1, pp. 23–56, 2022.

[14] Y. Huang, Z. Zhao, B. Wu, Z. Mei, Z. Cui *et al.,* "Visual object tracking with discriminative correlation filtering and hybrid color feature," *Multimedia Tools and Applications*, vol. 78, no. 24, pp. 34725–34744, 2019.

[15] J. Shin, H. Kim, D. Kim and J. Paik, "Fast and robust object tracking using tracking failure detection in kernelized correlation filter," *Applied Sciences*, vol. 10, no. 2, pp. 713, 2020.

[16] J. Zhang, X. Jin, J. Sun, J. Wang and A. K. Sangaiah, "Spatial and semantic convolutional features for robust visual object tracking," *Multimedia Tools and Applications*," vol. 79, no. 21, pp. 15095–15115, 2020.

[17] A. Bathija and G. Sharma, "Visual object detection and tracking using Yolo and sort," *International Journal of Engineering Research Technology*, vol. 8, no. 11, pp. 345–355, 2019.

[18] P. Nousi, D. Triantafyllidou, A. Tefas and I. Pitas, "Re-identification framework for long term visual object tracking based on object detection and classification," *Signal Processing: Image Communication*, vol. 88, no. 1, pp. 115969, 2020.

[19] S. Kanagamalliga and S. Vasuki, "Contour-based object tracking in video scenes through optical flow and gabor feature," *Optik*, vol. 157, no. 1, pp. 787–797, 2018.

[20] M. Yang, Y. Wu and Y. Jia, "A hybrid data association framework for robust online multi-object tracking," *IEEE Transactions on Image Processing*, vol. 26, no. 12, pp. 5667–5679, 2017.

[21] J. Yan, L. Zhong, Y. Yao, X. Xu and C. Du, "Dual-template adaptive correlation filter for real-time object tracking," *Multimedia Tools and Applications*, vol. 80, no. 2, pp. 2355–2376, 2021.

[22] T. Xu, Z. H. Feng, X. J. Wu and J. Kittler, "Learning adaptive discriminative correlation filters via temporal consistency preserving spatial feature selection for robust visual object tracking," *IEEE Transactions on Image Processing*, vol. 28, no. 11, pp. 5596–5609, 2019.

[23] D. Yuan, X. Zhang, J. Liu and D. Li, "A multiple feature fused model for visual object tracking via correlation filters," *Multimedia Tools and Applications*, vol. 78, no. 19, pp. 27271–27290, 2019.

[24] I. Jain and S. K. Sharma, "Convolutional siamese-RPN++ and Yolo-v3 based visual tracking regression," *Journal of Scientific Research*, vol. 66, no. 1, pp. 1–9, 2022.

[25] X. F. Zhu, X. J. Wu, T. Xu, Z. H. Feng and J. Kittler, "Complementary discriminative correlation filters based on collaborative representation for visual object tracking," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, no. 2, pp. 557–568, 2020.

[26] X. Zhang, G. S. Xia, Q. Lu, W. Shen and L. Zhang, "Visual object tracking by correlation filters and online learning," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 140, no. 1, pp. 77–89, 2018.

[27] Y. Lu, Y. Yuan and Q. Wang, "A dense connection based network for real-time object tracking," *Neurocomputing*, vol. 410, no. 1, pp. 229–236, 2020.