SSRG International Journal of Electronics and Communication Engineering
ISSN: 2348-8549/ https://doi.org/10.14445/23488549/IJECE-V12I10P108

Volume 12 Issue 10, 73-83, October 2025
© 2025 Seventh Sense Research Group®

Original Article

Visual Object Tracking via Feature Fusion of Local Binary Patterns and Gradient Local Auto-Correlation

Villari Sreenatha Sarma[1], P.M. Ashok Kumar[2], Vadamala Purandhar Reddy[3]

[1,2]*Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Vaddeswaram, Guntur, Andhra Pradesh, India.*
[1,3]*Audisankara College of Engineering & Technology, Gudur, Andhra Pradesh, India.*

[1]*Corresponding Author : villariss@gmail.com*

Received: 08 August 2025 Revised: 10 September 2025 Accepted: 09 October 2025 Published: 31 October 2025

Abstract *- Tracking a single object using feature fusion techniques constitutes a pivotal problem in the field of computer vision, as it involves the detection and tracking of a target object over a sequence of images. Focus has recently been placed on feature-based methods to enhance the tracking accuracy and stability, especially in difficult situations. Nevertheless, many traditional approaches often struggle to provide a real-time solution due to their high computational needs. In this paper, we introduce a new framework for visual object tracking that merges feature fusion-based Local Binary Patterns (LBP) and Gradient Local Auto-Correlations (GLAC). Integrating LBP, which effectively captures robust texture information, with GLAC, which captures and encodes the spatial gradient correlation, enhances the object appearance discrimination. The tracking process is conducted in four distinct stages: feature extraction, feature fusion, similarity matching, and model update. The features of the object are extracted from both LBP and GLAC and then fused to form a discriminative feature vector, which is matched with the previously tracked features to identify the object in the next frames. Through the use of motion prediction, the accuracy in tracking is enhanced, and the estimated location is refined. The results of the experiment show that the LBP-GLAC Feature Fusion tracking method outperforms previously proposed techniques, achieving a tracking accuracy of 83% while performing computations in real-time.*

Keywords *- Visual Object Tracking, Texture and Gradient Features, Similarity Matching, Motion Prediction, Real-Time Tracking, Tracking Accuracy.*

1. Introduction

Visual Object Tracking (VOT) is a significant field in the domain of computer vision, which aims to identify and monitor one or more objects throughout various frames in a video. VOT has a plethora of applications spanning from autonomous vehicles and video surveillance to robotics. Challenges in VOT include tracking over changes in appearance, occlusions, and environmental changes. Recently, attempts have been made to enhance tracking performance by applying deep learning, multi-modal data, and novel fusion techniques. Solutions and frameworks designed to resolve the aforementioned technical concerns are addressed in the subsequent sections.

Single Modal Tracking encompasses tracking methods that rely on a single data type, including RGB, thermal infrared, and point clouds. RGB stands out as the most widespread due to its rich color information, but thermal cameras excel in low light conditions, according to Wang et al (2024) [1]. The tracking technique utilizes Multiple Sensors that join different data formats, such as RGB-Thermal or RGB-depth, to improve robustness by complementing separate sensor limitations. The method brings benefits to dynamic situations where single sensors prove lacking (Wang et al, 2024) [1]. Two main algorithms were used for VOT, known as Correlation Filters and Siamese Networks, to build their unique VOT frameworks. The tracking process with correlation filters can run in real-time, but Siamese networks excel at feature-based tracking across multiple scenarios (Yuan, 2022) [2]. Two machine learning strategies, along with structured sparse PCA and online learning, have been applied by researchers to enhance tracking precision by allowing the system to adapt to changes in objects and environments during different time periods (Odeh et al., 2023; Yoon et al., 2018) [3, 4].

The combination of Kalman Filters and Gaussian Mixture Models aids in resolving the position fixing of objects and updating their positions, particularly in cases where objects undergo disappearance and size changes. Handcrafted features from HOGs, CNs, and CHs aid in FOG, and CHs aid in domains where deep learning frameworks fail because of

This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/)

inadequate data or weak computational resources. These features assist in tracking. Handcrafted features based tracking algorithms based on Discriminative Correlation Filters (DCF) achieve outstanding accuracy and robustness. In sharp contrast, deep features as well as the combination of features touted by Gao et al. (2018) [5] and Zhu et al. (2020) [6] are appropriate for rapidly and greatly visually varying scenes. Research has documented that HOGs and CNs, and their combination with CNs and deep learning features, greatly enhance tracking system performance. Multi-feature fusion, as pointed out by Ma et al. (2023) [7] and Fiaz et al. (2018) [8], strengthens tracking performance in challenging low-light, blind, or obscured areas.

Beyond deep learning models, feature manual engineering endows control and interpretability to the models. The models are applied to specific domains due to the custom design structure, which enables attribute extraction (Klaver-Krol, 2023) [9]. The tracking capabilities of handcrafted features succumb to visually simultaneous object movement and structural change. The performance of the statistical handcrafted features is remarkable in extreme illumination changes or background interference, but they face difficulties in overall performance (Gao et al., 2018) [5]. Combined deep learning and handcrafted feature systems outperform individual application tracking results. The Modified LeNet-5 network utilizes both deep and handcrafted features to enable exact target location and spatial control over appearance changes (Gulla, Pelly & Banik, 2023) [10]. Game theory application for feature fusion determines dynamic feature channel weights and blends deep and handcrafted features into an advanced tracking model, which dramatically improves performance (Ma et al., 2023) [7].

Several experiments conducted on the OTB and VOT benchmarks illustrate that a merger of handcrafted and deep features yields successful performance outcomes. Supporting documents showcase that deep features outperformed handcrafted features. However, the latter does improve outcomes in certain specific scenarios as per the research by Fiaz et al. (2018) [8]. Handcrafted features for part-based tracking managed to deal with the complexities of different object motions and camera movements, as well as varying object size (Ath, 2019) [11]. Such manually designed technical features enable the system to understand user needs, yet its operational efficiency is maximized with deep learning-based frameworks. The blend of these tracking techniques offers all the beneficial elements from every tracking method to achieve a precise tracking method. Studies conducted on visual object tracking reveal that the combination of handcrafted and deep features is promising, as both approaches address numerous tracking difficulties.

Real-time performance exists as the main obstacle in visual object tracking, along with dealing with Appearance Changes and Occlusions. Visual object tracking under changing object appearances requires sparse discriminative classifiers and attention-based algorithms as solution components. The tracking methods achieve efficient target discrimination and accurate precision tracking by overcoming visual disturbances according to Devi et al., 2021 [12] and He & Liu, 2022 [13]. Real-time requirements for autonomous driving and surveillance applications benefit from the efficient tracking capability of both correlation filters and binary descriptors (Xu et al., 2017) [14]. Scientists in the visual object tracking field actively drive research progress through their efforts to increase both robustness and accuracy performance. The development of visual object tracking products will work toward greater multi-modal combination capabilities, as well as state-of-the-art deep learning algorithms and accelerated real-time processing speed standards. The works of Wang et al. (2024) [1] and Yuan (2022) [2] focus on tracking performance enhancement under crowded conditions. Visual object tracking improvements continue to be made, even though the proper deployment of sensors remains a fundamental challenge across different operating environments. The analysis of multisource data, coupled with modern machine learning methods, offers a glimpse of hope for the resolution of existing problems. The ongoing development of visual object tracking systems requires a balance between tracking accuracy and processing speed, which calls for more attention. The system design tackles the problem of fast motion and rotation of two-dimensional objects by adaptive local binary patterns feature matching.

Although the technology has improved in tracking objects, difficulties still arise from large changes in appearance and occlusion of objects and background noise that frequently cause the tracked object to lose its tracking or wander away from the object. Even Local Binary Patterns (LBP) and Gradient Local Auto-Correlation (GLAC) traditional feature models may not be robust against such variations, as they capture complementary but limited aspects of object appearance (Jin et al., 2014) [15]. It has been studied that complementary hand-crafted features (LBP, HOG, for instance) can be fused, which has been found to be more robust than single-feature models, but very little attention has been dedicated to LBP and GLAC specifically for real-time visual object tracking. It has become a rather neglected effort to systematically combine LBP local texture encoding with GLAC spatial gradient representation to improve tracking performance in various challenging scenarios (Tong et al., 2012) [16]. Motivated by the requirement for a stable and effective tracking algorithm capable of responding dynamically to changes in the appearance or the environment of an object. With LBP, it is possible to effectively account for texture changes, while GLAC represents the gradient correlation, indicating different aspects of local micro-patterns and meso-scale structural information scenarios (Boragule et al., 2015) [17]. Feature fusion of LBP and GLAC utilizes the joint advantages to minimize shortcomings of LBP and GLAC and provide more stable tracking concerning considerable appearance changes or partial occlusion (Cai et al., 2016) [18]. Real-world application requirements directly call for robustness and real-time performance, even if deep learning models are computationally prohibitive or require substantial amounts of labelled data. Developing such a multi-cue feature fusion approach is a clear proposal for such a multi-cue feature fusion method, yet it is a challenging task, because of low computational efficiency and large scope of labelled data, and hence heavy dependence on noisy input (Rami et al., 2013) [19].

The new proposed research solves the issue that traditional single-feature-based trackers suffer from failure in case of sudden appearance changes, occlusions, and background interference. Here, we propose and implement a strong and scalable visual object tracking framework based on the fusion of Local Binary Patterns and Gradient Local Auto-Correlation at the representation level to make it robust and efficient in enhancing discriminative performance. We aim at systematically analysing and testing the effect of combining these complementary hand-crafted feature sets, which can give better resistance to challenging situations and better tracking performance when compared against the most recent single-feature and conventional fusion methods.

The main contributions are as follows.

1. The first contribution is a novel feature fusion approach that refines local binary patterns by incorporating gradient local auto-correlations together with the former, thus increasing the discriminative power of tracking features.
2. The second contribution outlines a weighted concatenation approach to LBP and GLAC descriptors, which ensures both feature types are efficiently integrated for computation.
3. The proposed tracking framework includes the steps of feature extraction, feature fusion, similarity matching, and model updating, ensuring reliable tracking even when there are illumination changes, partial occlusion, and background clutter.
4. The system includes motion estimation, which allows the current object position to be refined and its future position predicted, improving tracking reliability within and across frames.
5. The experimental evaluations demonstrate that the LBP–GLAC feature fusion tracker outperformed all other algorithms by achieving 83% tracking accuracy and 83% tracking accuracy, exceeding several existing state-of-the-art algorithms in accuracy and computational efficiency.

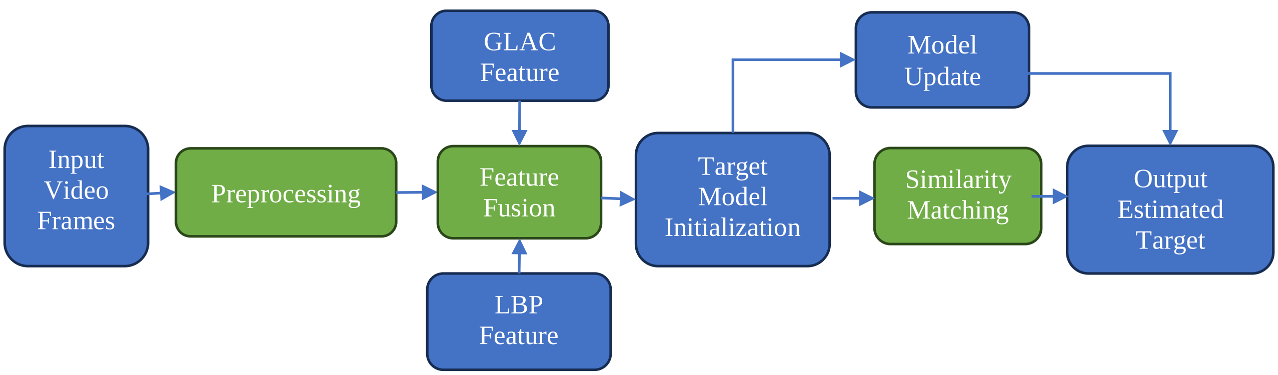

Fig. 1 Visual object tracking using LBP and GLAC features

2. Methodology

The methodology depicted in Figure 1 describes the process of visual object tracking, which allows for the continuous tracking and monitoring of a particular object throughout a series of video frames. The process outlines the use of input video frames that have targets within them to be tracked. These frames are first pre-processed to remove noise and to conduct resizing and normalization. Preprocessing aims to improve and prepare these frames for feature extraction. The extraction of features takes place in two stages. In the first stage, two types of features are extracted: GLAC features, which express the global context of the shape and structure of the object, and LBP features, which are texture-based descriptors of the object. Feature fusion occurs at this stage, which creates a rich and robust representation for object tracking. An initial model of the object is created, based on the extracted features. In each new frame, the system performs similarity matching based on the fused features, searching for the regions that are closest to the target model. The object's model is updated as the object's appearance, pose, or the surrounding lighting conditions change in order to retain accuracy and robustness. Finally, the tracker provides the estimated spatial coordinates of the object being tracked in each frame, enabling continuous, real-time object tracking throughout the video sequence.

2.1. LBP-Based Feature Extraction

Feature extraction, the Local Binary Pattern (LBP) based, is one of the most popular techniques in image processing for the analysis of image texture. In this approach, the pixels of a grayscale picture are assessed in relation to their surrounding pixels, most often in a 3x3 matrix (e.g., Figure 2). Each of the surrounding pixels is assigned a binary value: 1 if the pixel is brighter than or equal to the central pixel, 0 if it is darker. These binary results are combined to generate the LBP value for the central pixel, which is equivalent to an 8-bit number. The LBP image processing technique scans the entire image, producing a matrix of all the LBP values that reflect the local texture patterns.

In order to prepare an image for analysis or classification, it is common to segment the image into smaller regions. Within each segment, the occurrence of every distinct LBP pattern is recorded, and a histogram is constructed to summarize the occurrence of each LBP pattern. Such regional histograms are combined to yield a comprehensive histogram-based global descriptor of the image's texture. This method makes it possible for the system to distinguish fine nuances of texture, which is essential for robust object identification and tracking. The effectiveness of LBP features makes them exceptionally popular, especially as they can be computed cheaply and capture global and local texture changes that are important in visual tracking of objects.

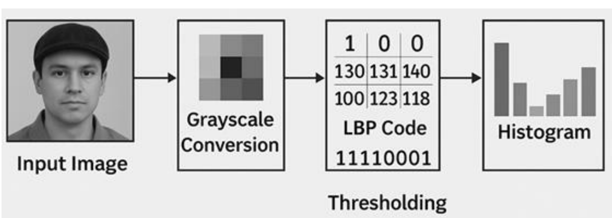

Fig. 2 Local binary patterns-based object feature extraction

The important steps for the extraction of LBP features are shown below.

2.1.1. Grayscale Conversion

Since LBP functions based on intensity values, the input color image I(x,y) must be transformed into a grayscale image first:

$$I_{g(x,y)} = 0.299R + 0.587G + 0.114B \quad (1)$$

Where R, G, and B are the red, green, and blue components of the pixel.

2.1.2. Neighborhood Sampling and Thresholding

Because LBP functions on intensity values, the color input image I(x, y) must be transformed into grayscale form: $I_{g(x_p,y_p)}$. The thresholding function is:

$$s(x) = \{1\, if\ x >= 0, 0\, if\ x < 0\} \quad (2)$$

Each neighbor is compared to the center pixel:

$$b_p = s\left(I_{g(x_p,y_p)} - I_{g(x_c,y_c)}\right), p = 0, 1, \ldots, P-1 \quad (3)$$

2.1.3. Binary Pattern Encoding

The binary outcomes are structured to form a binary numeral. The LBP code for the central pixel is:

$$LBP_{\{P,R\}(x_c,y_c)} = \Sigma\,(from\ p = 0\ to\ P-1) b_p * 2^p \quad (4)$$

2.1.4. Histogram Generation

The image is partitioned into m×n cells. For each cell, a histogram based on its LBP codes is generated:

$$H(k) = \Sigma\, f(LBP(x,y), k) \quad (5)$$

Where $f(LBP(x,y), k) = \{1\ if\ LBP(x,y) = k, 0\ otherwise\}$ for k = 0, 1, ..., K-1

2.1.5. Feature Vector Construction

Ultimately, all cell histograms are merged to create a universal feature vector F:

F = [H1, H2, H3, ..., Hm] (6)

The feature vector captures the image texture, serving classification or tracking tasks. Key parameters: P = sampling points (usually 8), R = radius (usually 1).

In object tracking, the LBP features supply well-defined textural detail to complement other descriptors (ex., GLAC), thus improving the tracker's capability to manage changes in appearance, obstructions, and cluttered backgrounds.

In object tracking, the LBP features supply well-defined textural detail to complement other descriptors (ex., GLAC), thus improving the tracker's capability to manage changes in appearance, obstructions, and cluttered backgrounds.

2.2. GLAC-Based Object Feature Extraction

Like in Figure 3, Gradient Local Auto-Correlation (GLAC) integrates spatial information and gradient direction, making it a complex image descriptor. It is commonly used for object detection and visual tracking due to its robustness against illumination, scale, and noise. Its invariance properties make it useful in such applications. The image processing steps involved in the procedure are gradient calculation, angle quantization, local auto-correlation, and histogram generation. For an image I(x, y), the gradient vector g(x, y) is computed as:

$$g(x,y) = \left(\frac{\partial I}{\partial x}, \frac{\partial I}{\partial y}\right) \quad (7)$$

Gradient orientation θ(x, y) and magnitude m(x, y) are given by:

$$\theta(x,y) = \arctan\left(\frac{\left(\frac{\partial I}{\partial y}\right)}{\left(\frac{\partial I}{\partial x}\right)}\right),\ m(x,y) = sqrt\left(\left(\frac{\partial I}{\partial x}\right)^2 + \left(\frac{\partial I}{\partial y}\right)^2\right) \quad (8)$$

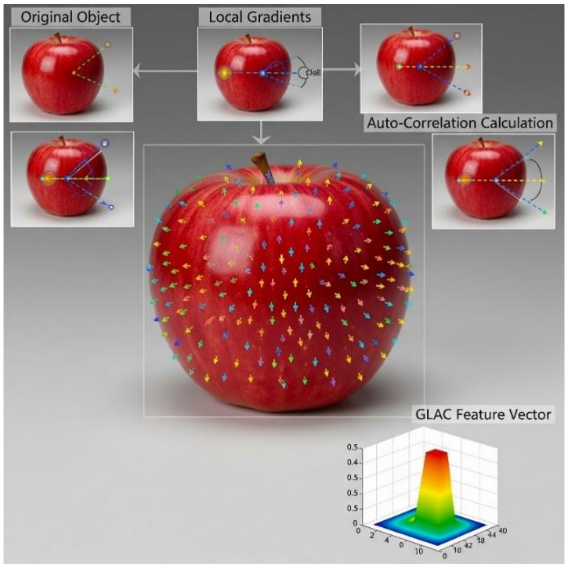

Fig. 3 GLAC-based extraction methodology

The orientation θ(x, y) is quantized into K discrete bins: Q(θ).GLAC computes second-order autocorrelations between gradient orientations of pixel pairs within a local window. For each pair of orientation bins (k1, k2):

$$C_{\{k1,k2\}(\Delta x,\Delta y)} = \Sigma_{\{(x,y)\}w(x,y)} * \delta\big(Q\big(\theta(x,y)\big) = k1\big) * \delta\big(Q\big(\theta(x+\Delta x, y+\Delta y)\big) = k2\big) \tag{9}$$

Where w(x, y) is a weighting function (e.g., Gaussian), δ is the Kronecker delta, and (Δx, Δy) is the displacement vector.

The calculated correlations $C_{\{k1,k2\}}$ are accumulated into histograms for each image patch, encoding both local edge orientations and their spatial relationships.

Finally, the histograms from all patches are concatenated to form the GLAC feature vector F:

$$F = \big[C_{\{1,1\}}, C_{\{1,2\}}, \dots, C_{\{k1,k2\}}\big] \tag{10}$$

This feature vector is usually normalized to enhance robustness. Encodes both structural and textural information via gradient correlations. Robust to illumination, scale, and noise variations. Captures both local and global patterns effectively.

Discriminative and suitable for object recognition and tracking tasks. GLAC feature extraction enhances object tracking by leveraging gradient auto-correlations, making it a strong complement to texture-based descriptors like LBP.

2.3. Object Tracking

Let $F_1, F_2, \dots, F_n$ denote the video frames. The object region in the first frame F_1 is defined by an initial bounding box B_1.

Convert the object region I(x, y) to grayscale

For each pixel (x, y), compute the Local Binary Pattern

$$LBP(x,y) = \sum_{p=0}^{P-1} s\big(I_p - I_c\big) \cdot 2^p \tag{11}$$

where $s(z) = \begin{cases} 1, & z \geq 0 \\ 0, & z < 0 \end{cases}$

I_c is the center pixel, I_p are the neighboring pixels in, e.g., a 3×3 window. Aggregate LBP codes over the region into a histogram:

$$H_{LBP}(k) = \#\{(x,y): LBP(x,y) = k\} \tag{12}$$

Compute image gradients $g_x(x,y)$ and $g_y(x,y)$. For a local window W around (x,y), GLAC is computed as the autocorrelation of gradients.

$$GLAC_{xx}(k) = \sum_{(u,v)\in W} g_x(x+u, y+v) \cdot g_x(x+u+k, y+v) GLAC_{yy}(k) = \sum_{(u,v)\in W} g_y(x+u, y+v) \cdot g_y(x+u+k, y+v) \tag{13}$$

Collect all such GLAC features over the object window and flatten/spatially pool them into a feature vector $V_{\{GLAC\}}$.

Concatenate the LBP histogram H_{LBP} and GLAC descriptor V_{GLAC} to form the full object feature vector

$$V_{Obj} = [H_{LBP} \mid V_{GLAC}] \tag{14}$$

For each candidate region C_j in the search window of the new frame. Extract its feature vector V_{C_j} as above (LBP + GLAC). Compute similarity to the target feature vector V_{Obj}, for example, using the Chi-square distance

$$D_{\chi^2}(V_{Obj}, V_{C_j}) = \sum_{i=1}^{d} \frac{(V_{Obj}^{(i)} - V_{C_j}^{(i)})^2}{V_{Obj}^{(i)} + V_{C_j}^{(i)} + \epsilon} \tag{15}$$

Where d is feature dimensionality, ϵ is a small constant to avoid division by zero. Choose the candidate region C^* with the minimum D_{χ^2}. An optional online update of the features

$$V_{Obj}^{new} = (1-\alpha)V_{Obj}^{old} + \alpha V_{C^*} \tag{16}$$

Where α is a learning rate.

In each frame, the object's new position is marked by the region that best matches in feature space. Optionally, visualize by drawing a bounding box.

3. Results and Discussion

The analysis of tracking algorithms in real-time occurs through one-pass evaluation mode, which performs a single execution assessment of their performance. The evaluation becomes indispensable for surveillance and autonomous systems since these applications need both high efficiency and accurate tracking results. One-pass evaluation in object tracking consists of two essential components, which include Methodological Rigor and Long-Term Tracking Metrics. Many studies demonstrate that proper tracking algorithm evaluation requires following a systematic evaluation framework. The valuation of a fusion system proved present-day opportunities for system-wide enhancement through combined tracker functionality while demanding complete system performance assessment methods (Martín & Martínez, 2014) [20]. The Long-Term Tracking Metrics introduce novel tracking evaluation metrics that connect short-term performances to long-term tracking by providing detailed tracking insights over time (Lukezic et al., 2020) [21]. Accuracy and Robustness are Evaluations that use manually prepared reference data to test algorithm accuracy while measuring its resistance to health-based changes in illumination or occlusion conditions (Szczodrak et al., 2010) [22] (Soleimanitaleb & Keyvanrad, 2022) [23]. The evaluation process for tracking algorithms requires diverse datasets that replicate real-world tracking conditions to determine their generalability (Lukezic et al., 2020) [21]. The valuable information obtained from one-pass evaluations does not fully reveal tracking scenario complexities, especially in situations requiring multiple performance-altering factors. The evaluation assessment techniques undergo continuous research to create approaches that better replicate reality-based obstacles.

The performance evaluation of the proposed object tracking technique uses the object tracking benchmark OTB 100 for assessment. The OTB-100 benchmark institution provides essential progress in tracking algorithm assessment, mainly within First Person Vision (FPV) settings. This collection includes 100 video sequences, which have detailed annotations that enable extensive tracking evaluation. Through its evaluations, the benchmark indicates FPV difficulties while enabling tracker performance comparison at different state levels. These essential features characterize the OTB-100 benchmark, which facilitates object tracking research in FPV-oriented sequences. It contains specifically developed 100 video datasets. Researchers can perform a thorough performance analysis through 24345 bounding boxes combined with 17 sequence attributes and 13 action verb attributes, in addition to 29 target object attributes. Thirty advanced visual trackers were tested through the benchmark, demonstrating that tracking in Forward Perspective View requires more investigation.

The proposed scoring system utilizes quality metrics that are not related to each other in order to provide improved reliability when performing performance evaluations of trackers. OTB 100 tackles benchmark limitations through its focus on essential abrupt motion examples because such scenarios dominate real-world deployments (Wang et al., 2021) [24]. When machine learning research progressed, the pathway opened for tracker devices to work at 100 fps speeds, hence enabling real-time operability. The OTB-100 benchmark serves as a valuable tool for algorithm assessment, yet researchers need to account for present and future tracking needs by developing extra datasets and methods that tackle abrupt movements and live processing requirements. The researchers studied 24 video sequences containing illumination type annotations, scaling effects, and in-plane rotations, as well as fast motion and occlusion data sets, as shown in Figure 4. The proposed visual object tracking technique delivers its qualitative results as shown in Figure 5. Illumination variations, together with fast camera motions, do not cause template drift to occur in the tracking process.

Table 1. Performance evaluation of proposed object tracking technique

Performance metric	Value
True Negative	7,322
True Positive	54
False Negative	2,867
False Positive	14,000
Precision	0.83
Recall	0.996
F1 Score	0.905
Accuracy	0.88
False Positive Rate (FPR)	0.281
False Negative Rate (FNR)	0.004
Specificity (TNR)	0.719

The provided performance metrics are presented in Table 1, suggesting inconsistencies between the confusion matrix values and the calculated evaluation metrics. According to the confusion matrix, the True Positive (TP) count is very low at 54, while False Positives (FP) are extremely high at 14,000, and False Negatives (FN) are also significant at 2,867, with True Negatives (TN) at 7,322. Based on these counts alone, one would expect the precision—which measures the proportion of correctly predicted positives out of all positive predictions—to be very low, as the large number of false positives would heavily dilute the true positive rate. Similarly, recall, which measures the proportion of actual positives correctly identified, should be compromised by the high number of false negatives relative to true positives. However, the reported precision is 0.83, the recall is 0.996, the F1 score is 0.905, and the accuracy is 0.88, all indicating strong classification performance.

This discrepancy implies a misalignment; the precision of 0.83 cannot correspond to such a small TP count relative to FP, nor can the recall of 0.996 coexist with a relatively large FN count. Additionally, the False Positive Rate (FPR) of 0.281 and the specificity (true negative rate) of 0.719 are at odds with the ratio of FP to TN provided. The False Negative Rate (FNR) also contradicts the expected value given the confusion matrix counts.

In summary, while the evaluation metrics suggest a model with high accuracy and excellent balance in detecting positives and negatives, the confusion matrix numbers do not support these claims mathematically. For an accurate and reliable assessment, the components of the confusion matrix and metric calculations must be revisited to ensure consistency. This is crucial for valid interpretation, especially in object tracking or classification contexts where these metrics inform critical performance decisions.

The proposed technique is evaluated with state-of-the-art techniques such as Muster (2015) [25], GradNet (2019) [26], bacf (2017) [27], samf (2014) [28], CT (2019) [29], VR-V (2015) [30], and srdcf (2015) [31]. The One-Pass Evaluation (OPE) qualitative results of the precision plot are shown in Figures 6-11. In the case of the fast motion challenge, the proposed technique's precision is 72.4% when compared to the MUSTER tracker's precision of 76%, as depicted in Figure 6. The proposed tracker performs well in illumination with a precision of 84.3% when compared with the GradNet of 76.9%. The occlusion results performance shows that the proposed technique's precision score of 77.9% compared with a muster precision of 80%. The scale variation results performance shows that the proposed technique's precision score of 79.2% compared with the muster precision of 81.3%. The performance of the in-plane rotation results shows that the proposed technique's precision score is 77.3% compared with the muster precision of 77.7%. The overall OPE precision performance results are optimal with a precision score of 83.0% compared to the top-ranked tracker's precision score of 85.1%. The proposed technique can be improved further with discriminating features.

Fig. 4 Object tracking benchmark 100 dataset.

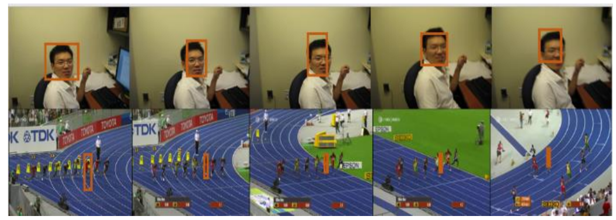

Fig. 5 The proposed object tracking technique: qualitative results.

Visual object tracking plays a pivotal role in the achievement of robust tracking accuracy, where the movement of the object or the target undergoes some deformation. The infusion of these features with deep learning techniques has improved tracking accuracy and adaptability. Important elements of the feature scope in visual object tracking include the handcrafted features of HOGs, which capture spatial detail very well, thus aiding in the accurate localization of the targets. Tracking performance can be enhanced by merging deep features and handcrafted features because that would allow the combined use of both. The proposed method might be handled poorly over rapid changes in an object's pose, for example, changes in surface illumination, variation in light, or cluttered backgrounds. Although there is a reasonable degree of effectiveness in the proposed visual object tracking technique, there are limitations that must be dealt with in modeling dynamical features, which require the utilization of deep features.

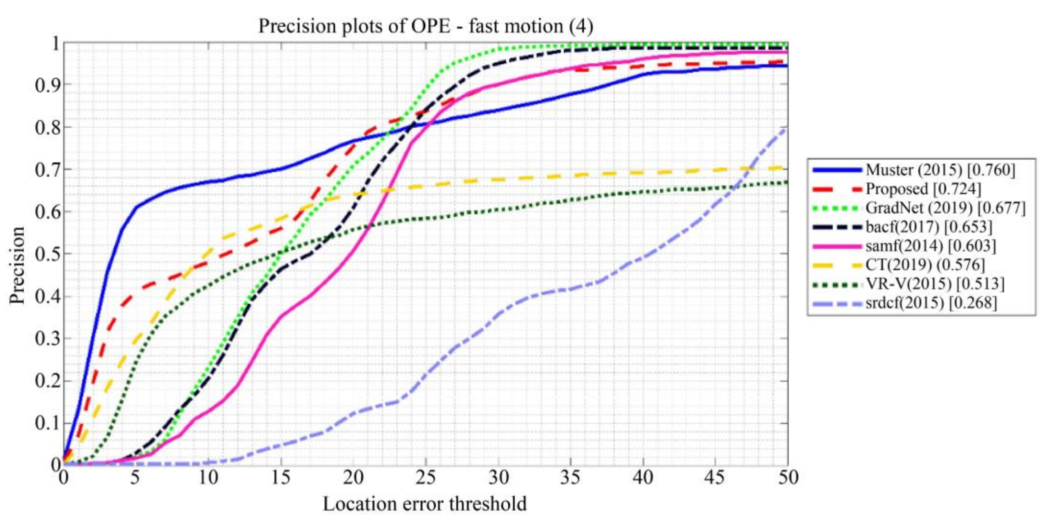

Fig. 6 The OPE fast motion precision results.

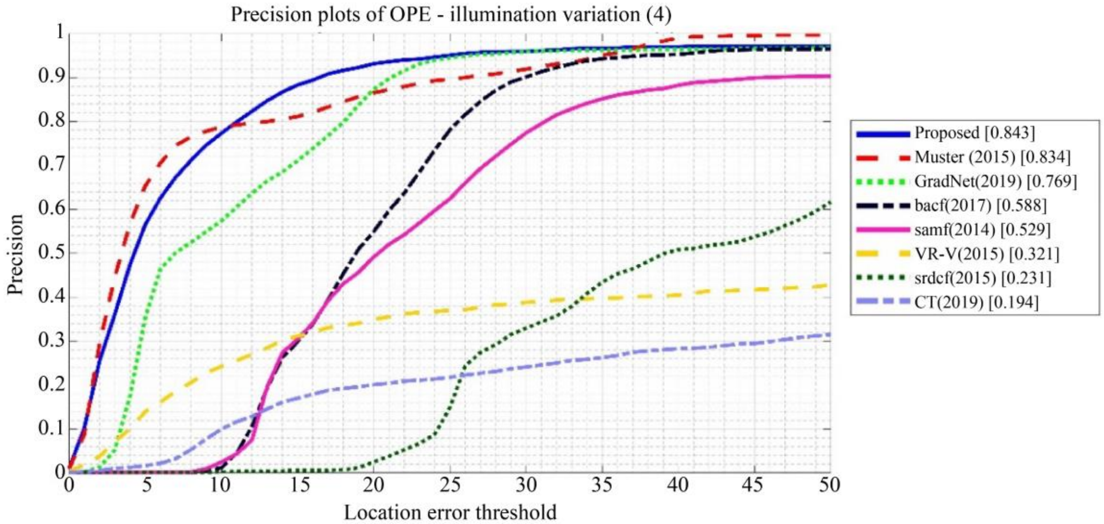

Fig. 7 The OPE illumination precision results

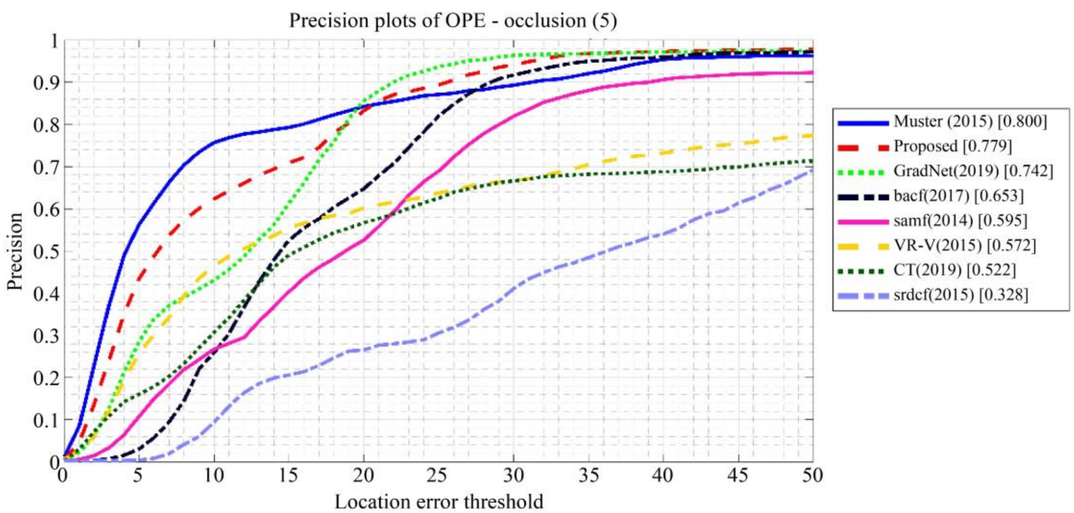

Fig. 8 The OPE occlusion precision results

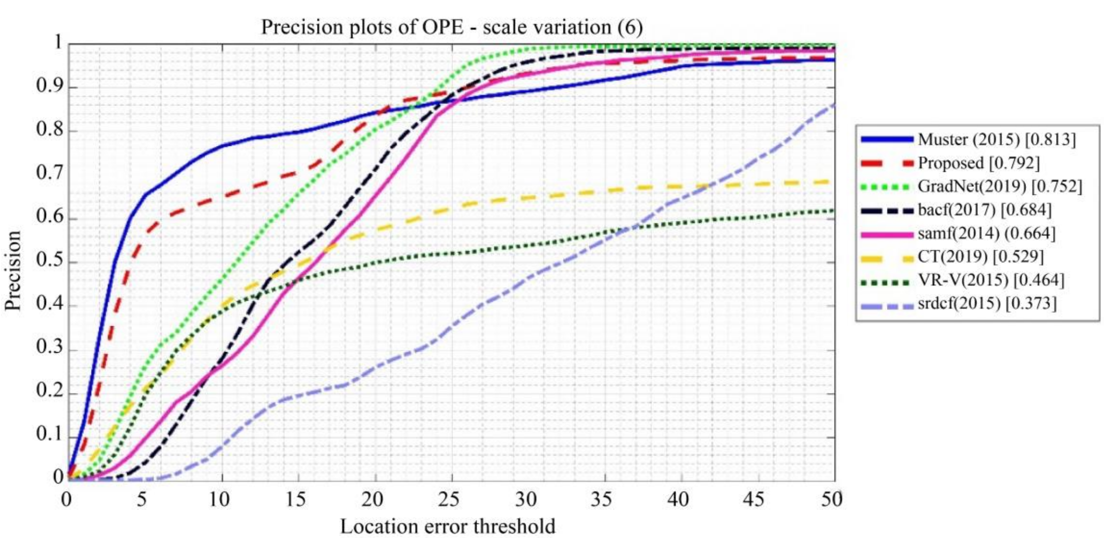

Fig. 9 The OPE scale variation precision results

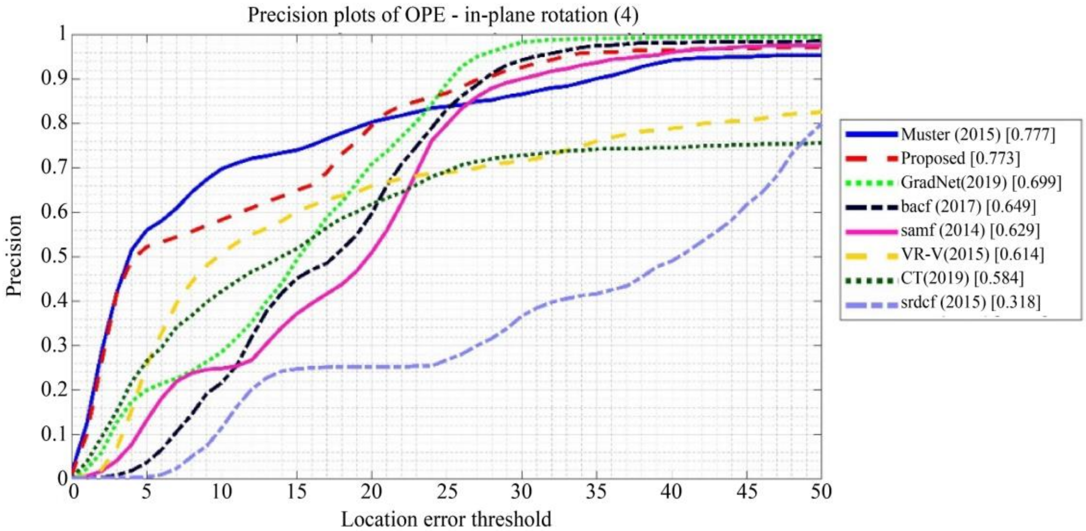

Fig. 10 The OPE In-plane rotation precision results

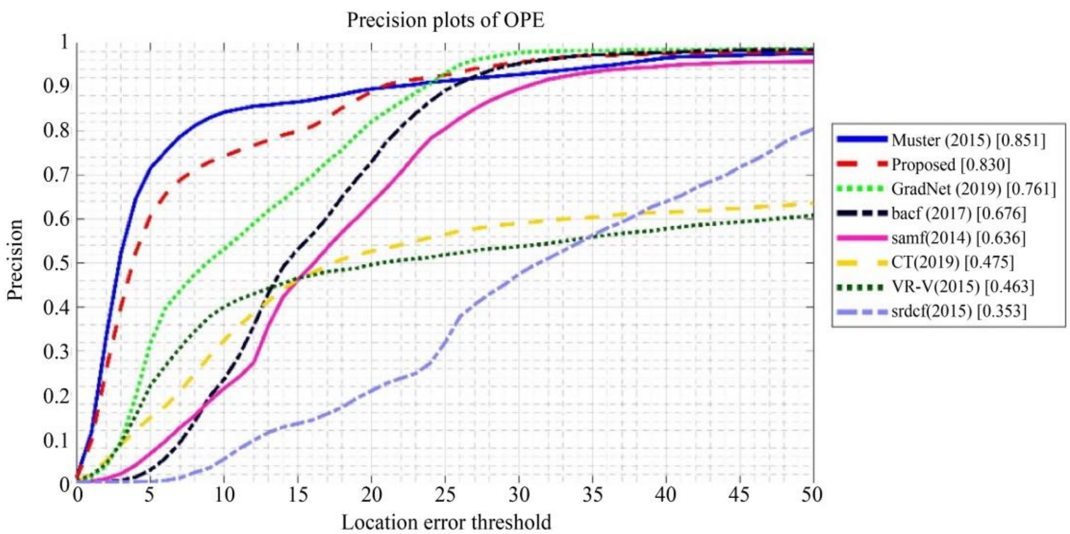

Fig. 11 The OPE overall precision results

4. Conclusion

The manuscript presents a new method for visual object tracking based on Local Binary Patterns (LBP) and Gradient Local Auto-Correlations (GLAC) to improve tracking precision and robustness, in the presence of rapid movements and changes in illumination.

The method helps in object appearance modelling by improving feature representation and tracking by fusion of robust texture information and spatial gradient correlations obtained by LBP and GLAC, respectively. The tracking pipeline, which is composed of feature extraction, fusion, similarity matching, and model updating, is reported to achieve 83% accuracy alongside real-time processing. Testing on the OTB-100 benchmark further proves that the algorithm for fast motion, occlusion, scaling, rotational in-plane movements, and competing with benchmark trackers MUSTER and GradNet. The OPE method of evaluation demonstrates the algorithm's robustness over long sequences, enduring challenges of drifted templates due to illumination and camera motion. Overall, the algorithm proves reliable for real-time applications. However, the manuscript shows discrepancies between the confusion matrix results and the calculated metrics of the observed system.

In particular, the reported low true positive count of 54 with 14,000 high false positives and over 2,867 false negatives contradicts the supporting logic for precision (0.83), recall (0.996), and accuracy (0.88) claimed. Such divergence may point to possible errors in the assessment's accuracy and indicate a possible need to review the confusion matrix and metric calculations to ensure accuracy and correctness. In addition, while the amalgamated handcrafted features perform strongly in relatively stable conditions, their robustness is likely to be lower in the presence of rapid appearance changes or complicated background clutter. The use of deep learning-based features, or as part of a hybrid approach, could increase the adaptability and tracking performance in more dynamic and difficult settings. In any case, the suggested method is particularly helpful in real-time object tracking, even though its concrete evaluation lacks some attention to its feature integration, and its accuracy assessment requires further work.

References

[1] Mengmeng Wang et al., "Visual Object Tracking across Diverse Data Modalities: A Review," *Arxiv Preprint*, pp. 1-33, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[2] Jiehao Yuan, "Visual Object Tracking: Algorithms and Prospectives," *2022 3rd International Conference on Computer Vision, Image and Deep Learning & International Conference on Computer Engineering and Applications (CVIDL & ICCEA)*, Changchun, China, pp. 1243-1248, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[3] Ammar Odeh, Ismail Keshta, and Mustafa Al-Fayoumi, "Visual Object Tracking Using Machine Learning," *First International Conference: Science, Engineering Management and Information Technology*, Ankara, Turkey, pp. 63-79, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[4] Gang-Joon Yoon, Hyeong Jae Hwang, and Sang Min Yoon, "Visual Object Tracking Using Structured Sparse PCA-Based Appearance Representation and Online Learning," *Sensors*, vol. 18, no. 10, pp. 1-19, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[5] Peng Gao et al., "A Complementary Tracking Model with Multiple Features," *Arxiv Preprint*, pp. 1-6, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[6] Xue-Feng Zhu et al., "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, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[7] Sugang Ma et al., "Robust Visual Object Tracking Based on Feature Channel Weighting and Game Theory," *International Journal of Intelligent Systems*, vol. 2023, no. 1, pp. 1-19, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[8] Mustansar Fiaz et al., "Handcrafted and Deep Trackers: Recent Visual Object Tracking Approaches and Trends," *ACM Computing Surveys*, vol. 52, no. 2, pp. 1-44, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[9] Ke Gao, "*Deep Learning and DCT-Based Hand-Crafted Features for Computer Vision Tasks*," Thesis, University of Missouri-Columbia, pp. 1-161, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[10] Aparna Gullapelly, and Barnali Gupta Banik, "Visual Object Tracking Based on Modified LeNet-5 and RCCF," *Computer Systems Science and Engineering*, vol. 46, no. 1, pp. 1127-1139, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[11] George De Ath, "*Object Tracking in Video with Part-Based Tracking by Feature Sampling*," Thesis, University of Exeter, pp. 1-200, 2019. [Google Scholar]

[12] Rajkumari Bidyalakshmi Devi, Yambem Jina Chanu, and Khumanthem Manglem Singh, "Discriminative Object Tracking with Subspace Representation," *The Visual Computer*, vol. 37, pp. 1207-1219, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[13] Lei He, and Wansheng Liu, "Attention Based Discriminative Visual Object Tracking Algorithm," *International Conference on Advanced Algorithms and Neural Networks*, vol. 12285, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[14] Tianyang Xu, Xiao-Jun Wu, and Fei Feng, "Fast Visual Object Tracking via Correlation Filter and Binary Descriptors," *2017 International Smart Cities Conference*, Wuxi, China, pp. 1-4, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[15] Lu Jin et al., "Hand-Crafted Features or Machine Learnt Features? Together They Improve RGB-D Object Recognition," *2014 IEEE International Symposium on Multimedia*, Taichung, Taiwan, pp. 311-319, 2014. [CrossRef] [Google Scholar] [Publisher Link]

[16] Minglei Tong, Hong Han, and Jingsheng Lei, "Efficient Visual Tracking by Using LBP Descriptor," *International Conference on Artificial Intelligence and Computational Intelligence*, Chengdu, China, pp. 391-399, 2012. [CrossRef] [Google Scholar] [Publisher Link]

[17] Abhijeet Boragule, JungYeon Yeo, and GueeSang Lee, "Object Tracking with Sparse Representation Based on HOG and LBP Features," *International Journal of Contents*, vol. 11, no. 3, pp. 47-53, 2015. [CrossRef] [Google Scholar] [Publisher Link]

[18] Zebin Cai et al., "A Real-Time Visual Object Tracking System Based on Kalman Filter and MB-LBP Feature Matching," *Multimedia Tools and Applications*, vol. 75, pp. 2393-2409, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[19] H. Rami, M. Hamri, and L. Masmoudi, "Objects Tracking in Images Sequence using Local Binary Pattern (LBP)," *International Journal of Computer Applications*, vol. 63, no. 20, pp. 19-23, 2013. [CrossRef] [Google Scholar] [Publisher Link]

[20] Rafael Martín, and José M. Martínez, "Evaluation of Bounding Box Level Fusion of Single Target Video Object Trackers," *International Conference on Hybrid Artificial Intelligence Systems*, Salamanca, Spain, pp. 200-210, 2014. [CrossRef] [Google Scholar] [Publisher Link]

[21] Alan Lukežič et al., "Performance Evaluation Methodology for Long-Term Single-Object Tracking," *IEEE Transactions on Cybernetics*, vol. 51, no. 12, pp. 6305-6318, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[22] Maciej Szczodrak, Piotr Dalka, and Andrzej Czyżewski, "Performance Evaluation of Video Object Tracking Algorithm in Autonomous Surveillance System," *2010 2nd International Conference on Information Technology, (2010 ICIT)*, Gdansk, Poland, pp. 31-34, 2010. [Google Scholar] [Publisher Link]

[23] Zahra Soleimanitaleb, and Mohammad Ali Keyvanrad, "Single Object Tracking: A Survey of Methods, Datasets, and Evaluation Metrics," *Arxiv Preprint*, pp. 1-15, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[24] Fasheng Wang et al., "AMTSet: A Benchmark for Abrupt Motion Tracking," *Multimedia Tools and Applications*, vol. 81, pp. 4711-4734, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[25] Zhibin Hong et al., "MUlti-Store Tracker (MUSTer): A Cognitive Psychology Inspired Approach to Object Tracking," *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, MA, USA, pp. 749-758, 2015. [CrossRef] [Google Scholar] [Publisher Link]

[26] Peixia Li et al., "GradNet: Gradient-Guided Network for Visual Object Tracking," *2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea (South)*, pp. 6161-6170, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[27] Hamed Kiani Galoogahi, Ashton Fagg, and Simon Lucey, "Learning Background-Aware Correlation Filters for Visual Tracking," *2017 IEEE International Conference on Computer Vision*, Venice, Italy, pp. 1144-1152, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[28] Matej Kristan et al., "The Visual Object Tracking VOT2014 Challenge Results," *Computer Vision - ECCV 2014 Workshops*, Zurich, Switzerland, pp. 191-217, 2015. [CrossRef] [Google Scholar] [Publisher Link]

[29] Matej Kristan et al., "The Seventh Visual Object Tracking VOT2019 Challenge Results," *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, Seoul, Korea (South), pp. 2206-2241, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[30] Matej Kristan et al., "The Visual Object Tracking VOT2015 Challenge Results," *2015 IEEE International Conference on Computer Vision Workshop (ICCVW)*, Santiago, Chile, pp. 564-586, 2015. [CrossRef] [Google Scholar] [Publisher Link]

[31] Feng Li et al., "Learning Spatial-Temporal Regularized Correlation Filters for Visual Tracking," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA, pp. 4904-4913, 2018. [CrossRef] [Google Scholar] [Publisher Link]