I.J. Image, Graphics and Signal Processing, **2022, 1, 64-81**
Published Online February 2022 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijigsp.2022.01.06

Object Tracking: An Experimental and Comprehensive Study on Vehicle Object in Video

Vo Hoai Viet
University of Science, Ho Chi Minh City, 700000, Viet Nam
E-mail: vhviet@fit.hcmus.edu.vn

Huynh Nhat Duy
University of Science, Ho Chi Minh City, 700000, Viet Nam
E-mail: 19C11003@hcmus.edu.vn

Received: 11 November 2021; Accepted: 05 January 2022; Published: 08 February 2022

Abstract: Tracking objects on camera or video is very important for automated surveillance systems. Along with the development of techniques and scientific research in object tracking, automatic surveillance systems have gradually become better. With the input of a frame including the object to be tracked and the location information of the object to be tracked in that video. The output will be the prediction of the position of the object to be tracked on the next frame. This paper presents the comparison and experiment of some traditional object tracking methods and suggestions for improvement between them. Firstly, we examined related studies, traditional object tracking models. Secondly, we examined image and video data sets for verification purposes. Thirdly, experimenting with some related research works in traditional object tracking problems, evaluation of the existing model, what has been achieved and what has not been achieved for the current models. Propose improvements based on the combination of traditional methods. Finally, we aggregate these results to evaluate for each type of object tracking model. The results show that Particles Filter method has the highest CDT with TO score of 0.907971 on VOT dataset and 0.866259 on UAV123 dataset. However, the most stable are the two hybrid methods, the Particle filter base on Mean shift method has a TF score of 31.1 on the VOT dataset and the Kalman Filter base on Mean shift method has a TME score of 28.8233 on the UAV dataset. Because low-level features cannot represent all the information of an object to be tracked during the completion of the experiment, we can conclude that combining deep learning network and using high-level feature into the tracking model can bring better performance in the future.

Index Terms: Object Tracking, Surveillance Systems, Single-Object Tracking, Mean-Shift, Kalman Filter, Particle Filter.

1. Introduction

Object Tracking is a problem in the field of computer vision. In recent years, along with the development of science and technology, people have more and more needs to use intelligent systems with an increasing degree of automation. In the field of security and surveillance, computer vision is widely used. The surveillance system includes three processes: Object extraction, object tracking and behavior recognition. From there, store the collected information in the database or detect anomalies to give timely warnings. In some practical applications such as abnormal detection, human activity recognition …The events happen in short or long time that objects are in the scene, so that the tracking algorithms are the most important phase to capture data and information for better performance in overall system. There is much research about tracking algorithms in the past [1, 5, 9] and benchmark datasets are published in computer vision community. However, when we apply these algorithms into practical application or research, the results still are not the same expectation. In this search, we will investigate these object tracking algorithms into vehicle or car domain and make some experiment on benchmarks datasets.

Currently, the research related to the object tracking problem [24, 26, 29, 30, 31] and as well as the reviews related [26, 15, 22, 25, 27] to this field are many. However, to our knowledge, there has not been any comprehensive review on the topic of vehicle object tracking. The purpose of this paper is to review and build traditional object tracking methods (Mean-Shift, Kalman Filter, Particle Filter) along with suggestions and improvements based on the combination of traditional methods for the object of vehicle. From researching, surveying object tracking models to building experiments and demonstrating research models to evaluate results and compare with experimental methods. The main contribution of this manuscript is a review of several object tracking methods and comparisons with improved methods based on a combination of current methods on vehicle datasets. Summary, the main contributions are summed:

- We describe our approach on object tracking models. Review literature of object tracking problems: i) problem definition; ii) challenges in research and evaluation; ii) metrics for measurement and testing.
- We present a perspective of evaluating both pros and cons of the models. Study traditional algorithms and make experiments on vehicle dataset. Then, we make evaluation for comparison for comprehensive perspective.
- Propose some hybrid algorithms for improvements base on current methods as well as feature extraction methods to provide stable solutions whenever an exception occurs and do experiments on vehicle datasets. Then, we summarize these findings for the best performance in use cased for practical application in vehicle tracking domain.

The remainder of the paper is organized as follows. In the section II, we present background (include challenges, evaluation metrics, traditional methods and methodology). Our proposed approach is explained in session III. Then, the experimental results and discussion are shown in the section IV. Finally, we draw conclusions in section V.

2. Background

2.1 Object Tracking Definition

Object tracking is a challenging task, that is the problem of locating objects moving over time on camera or on video sequences (An example of object correspondence is shown in Fig. 1). Object tracking is commonly used in automated surveillance systems, human-computer interaction, video communication and compression, augmented reality, traffic control, medical imaging and video editing, …. There are two directions of tracking objects, that is a single object tracking and multiple object tracking. In this report, we would like to present the evaluation of object tracking techniques in the approach track specific objects with a single object. The object to be tracked in this article is means of vehicle, the data sets [12, 13] for the experimental process include VOT Challenge Dataset and UAV123 Dataset.

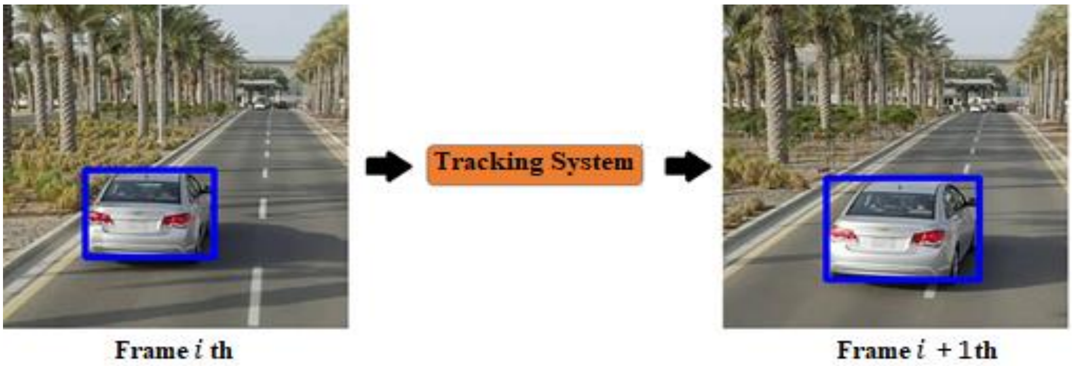
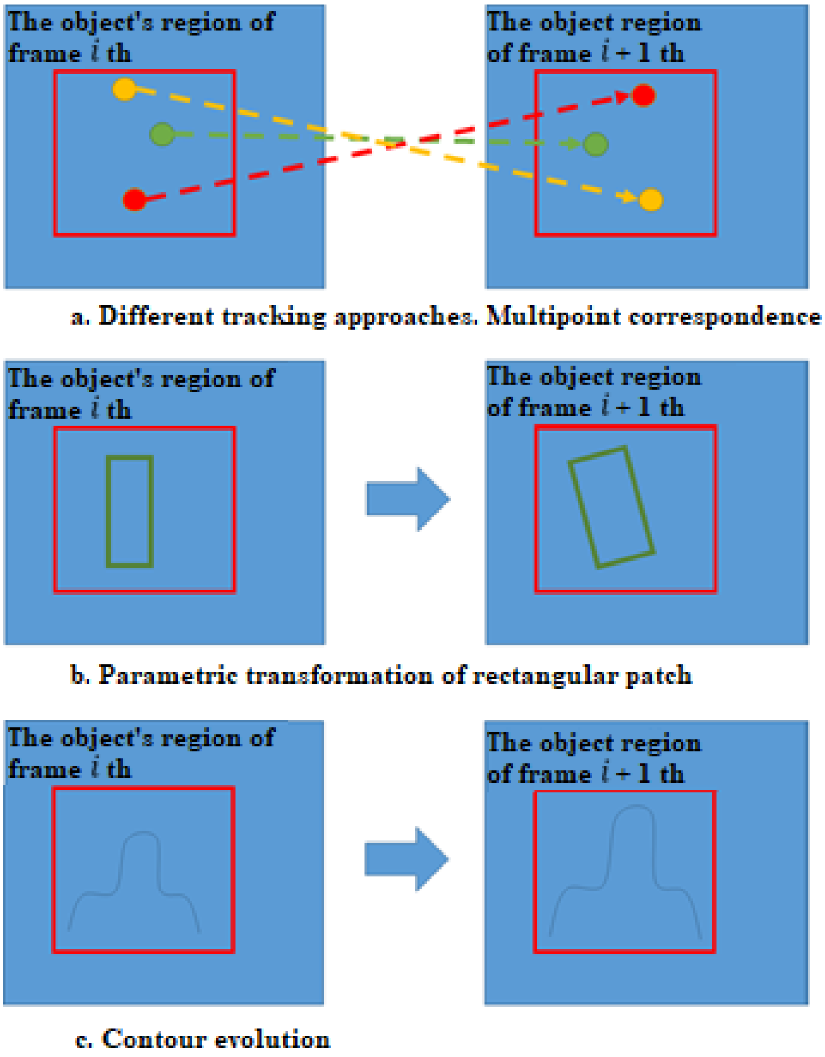


Fig. 1. Object’s position on frame $i + 1$ th has been predicted based on object’s region of frame i th.

Table I.The problem, the approach, and the goal of object tracking techniques

PROBLEM	The problem of estimating the trajectory of an object in an image plane.
APPROACH	• Point Tracking [14]: To predict the object's position in the next frame based on the object's position in the current frame, the association points of each frame are detected based on the object's state in the previous frame (including position and movement). (As shown in Fig. 2.a) • Kernel Tracking [14]: The object's motion is predicted on the next frame at which point the object shape and appearance may be changed. kernel predicts object motion such as translation, rotation, and affine (Fig. 2.b) with an associated histogram (kernel is usually represented as a rectangular template or an elliptical shape). • Silhouette Tracking [14]: To estimate the object container in each frame, this method is like a time domain object segmentation method using original segments generated from previous frames. object's silhouettes are tracked by either shape matching or contour evolution (Fig. 2.c).

GOAL	The main goal of the trackers in this category is to estimate the object motion. With the region-based object representation, computed motion implicitly defines the object region as well as the object orientation in the next frame since, for each point of the object in the current frame, its location in the next frame can be determined using the estimated motion model


The object's region of frame i th
The object region of frame $i+1$ th
The object's region of frame i th
The object region of frame $i+1$ th
The object's region of frame i th
The object region of frame $i+1$ th

a. Different tracking approaches. Multipoint correspondence

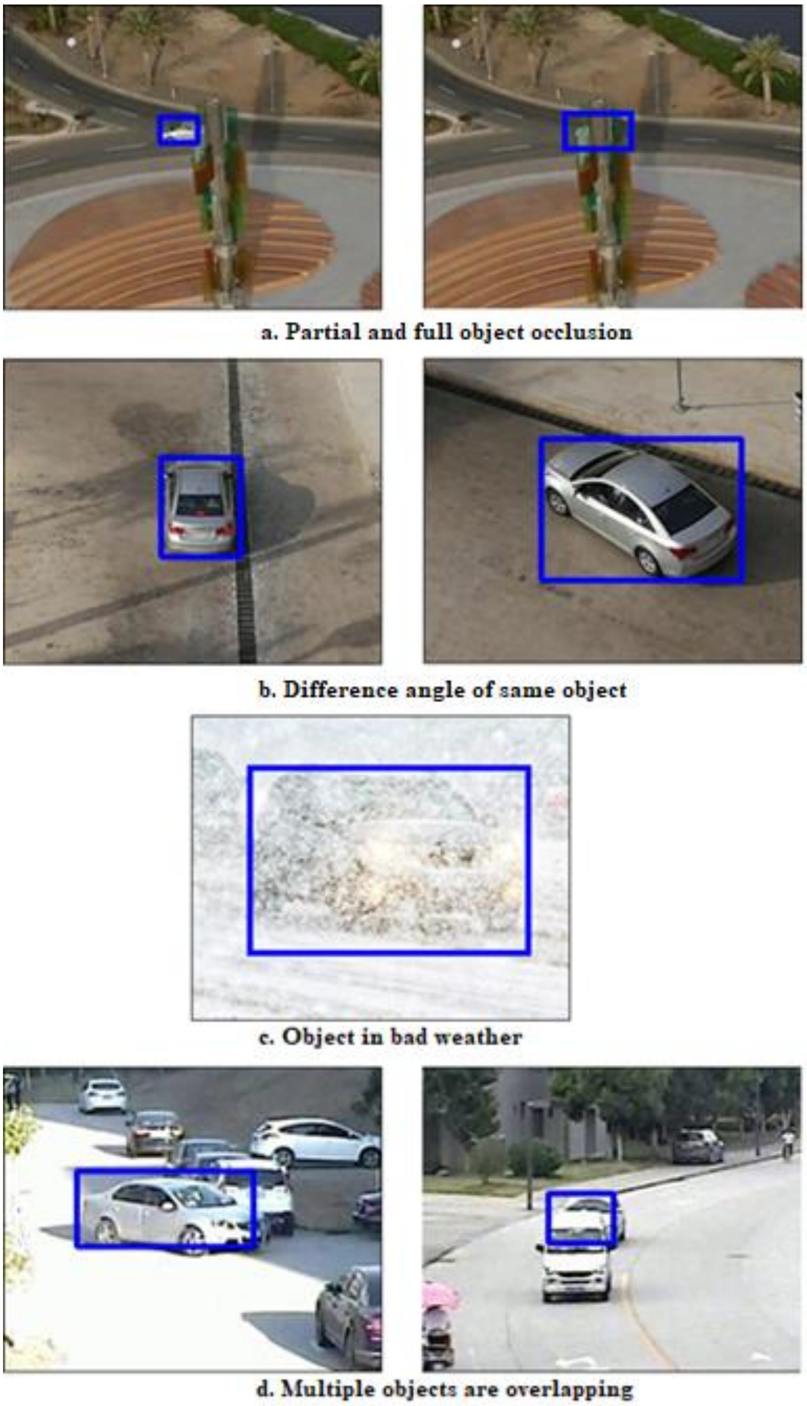
b. Parametric transformation of rectangular patch

c. Contour evolution

Fig. 2. Example for object tracking approaches

2.2 Challenges

The requirement of an object tracking problem that predicts the position of the object in the current frame based on the area that includes the object to be tracked in the previous frame. However, there are still many issues that can affect the tracking model when the tracked object can be occluded object (Fig. 3.a) or different angles (Fig. 3.b) of the same object in a real-time processing speed. Moreover, the image quality obtained from the camera is highly susceptible to physical changes such as weather if outdoors (Fig. 3.c). The object tracking problem is not only detected if there is only one object in the video sequence and calculates its displacement, but also there will be cases where situations contain multiple objects with the likelihood of overlapping between them (Shown in Fig. 3.d).



a. Partial and full object occlusion

b. Difference angle of same object

c. Object in bad weather

d. Multiple objects are overlapping

Fig. 3. Example for types of challenges

2.3 Evaluation Metrics

The evaluation metrics [20, 21, 23] is important to evaluate the performance of the systems, to define the concepts of spatial and temporal overlap [18] between tracks, which are required to quantify the level of matching between Ground Truth (GT) tracks and System (ST) tracks, both in space and time. The spatial overlap [17] is defined as the overlapping level A(GT_i, ST_j) between GT_i and ST_j tracks in a specific frame k define:

$$A\left(GT_{ik},ST_{jk}\right)=\frac{Area(GT_{ik}\cap ST_{jk})}{Area(GT_{ik}\cup ST_{jk})} \tag{1}$$

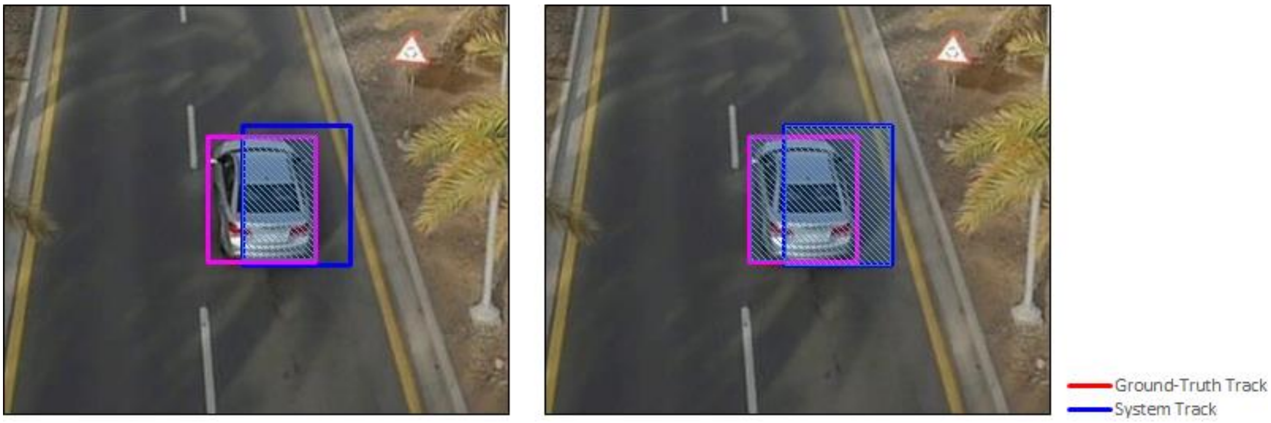


Fig. 4. Illustration of $interection\ area(GT_{ik} \cap ST_{jk})$ and union $area(GT_{ik} \cup ST_{jk})$

2.3.1 Correct Detected Track (CDT) or True Positive (TP)

If each of the GT tracks is detected correctly (by one or more system tracks), the number of CDT equals the number of GT tracks. A GT track is considered to have been detected correctly if it satisfies both of the following conditions:

The temporal overlap (TO) [18] between GT track i and system track j is larger than a predefined track overlap threshold TR_{OV}. In this paper, the threshold TR_{OV} is set to 15%.

$$\frac{Length(GT_i \cap ST_j)}{Length(GT_i)} \geq TR_{OV} \tag{2}$$

Assuming that the object needs to be tracked in a video of 100 frames. From the 51th frame to the 60th frame, the object is obscured by another object and the number of overlaps between GT and ST is 70 times (for one frame at time), so CDT score of TP is 70/90≅ 0.78 (greater than 0.15).

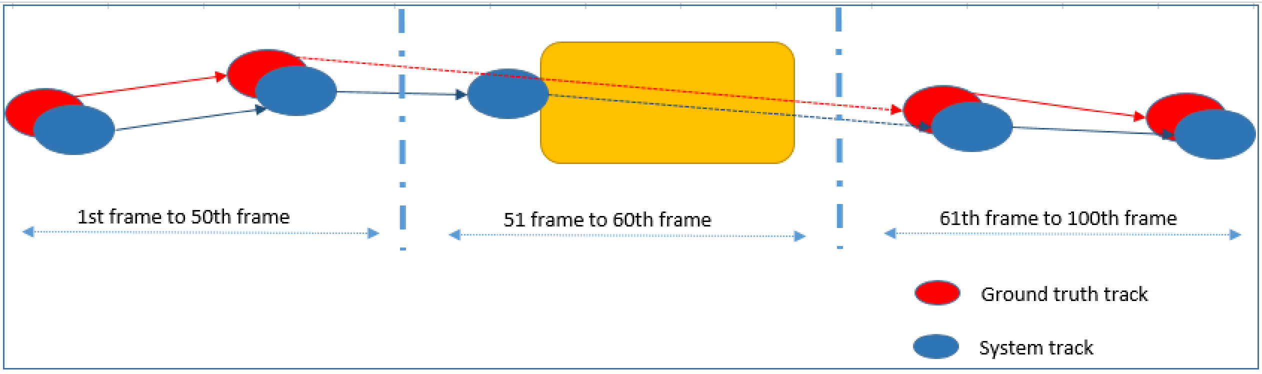


Fig. 5. Example for tracking objects in case obscured in the video chain

The system track j has sufficient spatial overlap (SSO) [17] with GT track i (Eq.1) and the theshold T_{OV} is set to 20%.

$$\frac{\sum_k^N A(GT_{ik}, ST_{jk})}{N} \geq T_{OV} \tag{3}$$

2.3.2 False Alarm Track (FAT) of False Positive (FP)

FAT is an important metric because it is consistently indicated by operators that a system which does not have a false alarm rate close to zero is likely to be switched off, not matter its TP performance and following conditions:

A system track j does not have temporal overlap [18] larger than TR_{OV} with any GT track i.

$$\frac{Length(GT_i \cap ST_j)}{Length(ST_j)} \leq TR_{OV} \tag{4}$$

Assuming that the object needs to be tracked in a video of 100 frames and object is always be detected for each frame by system. From the 51th frame to the 60th frame (Fig. 5), the object is obscured by another object and the number of overlaps between GT and ST is 70 times (for one frame at time), so FAT score of FP is $\frac{70}{100} = 0.7$.

A system track j does not have sufficient spatial overlap [17] with any GT track although it has enough temporal overlap [18] with GT track i. (Eq.1).

$$\frac{\sum_k^N A(GT_{ik}, ST_{jk})}{N} \leq T_{OV} \tag{5}$$

2.3.3 Track Detection Failure

A GT track is considered to have not been detected (i.e., it is classified as a track detection failure), if it satisfies any of the following conditions. GT track i does not have temporal overlap [18] larger than TR_{OV} with any system track j.

$$\frac{Length(GT_i \cap ST_j)}{Length(GT_i)} \leq TR_{OV} \tag{6}$$

A GT track i does not have any sufficient spatial overlap [17] with any system track, although it has enough temporal overlap [18] with system track j (Eq.1).

$$\frac{\sum_k^N A(GT_{ik}, ST_{jk})}{N} \leq T_{OV} \tag{7}$$

2.3.4 Track fragmentation

Fragmentation indicates the lack of continuity of system track for a single GT track. Below is an example of track fragmented error:

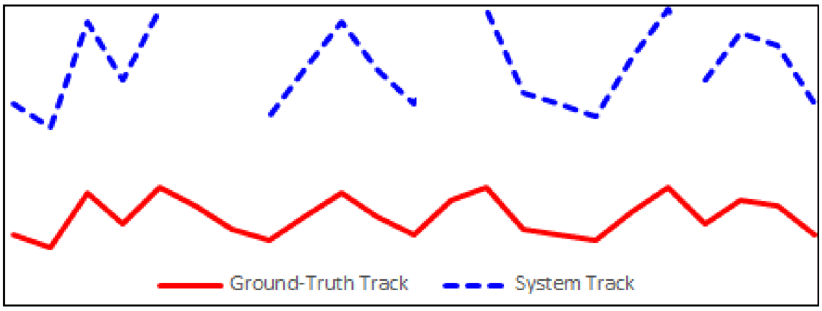


Fig. 6. The system tracks fragmented three times, then the number of tracks is TF = 3

For example, a video has 100 frames that include the objects have been tracked. The system has failure prediction at 41th frame and 46th frame, then track fragmentation is 2 because the system has 2 times tracking failue.

2.3.5 Latency of the system track

Latency [19] (time delay) of the system track (LT) is the time gap between the time that an object starts to be tracked by the system and the first appearance of the object.

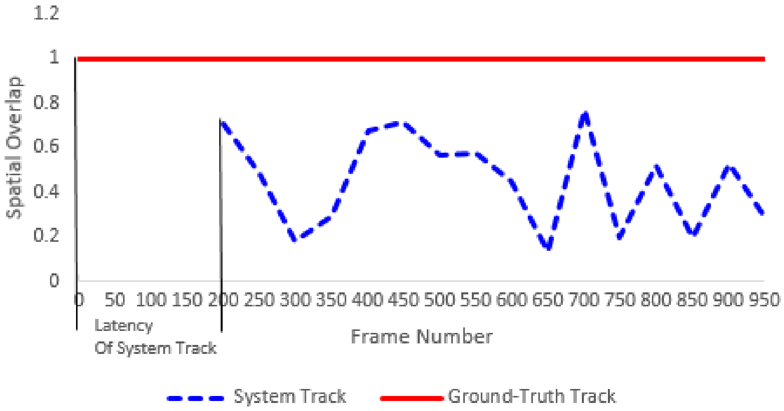


Fig. 7. Example of latency of system track

Latency of the system track is estimated by the difference in frames between the first frame of system track and the first frame of GT track:

$$\text{LT} = \text{start frame of } ST_j - \text{start frame of } GT_i \quad (8)$$

For example, a video has 30 FPS (Frame Per Second) and 1 object need to be tracked. At the first frame, the system needs 200ms to process and show the first frame to screen, the ground-truth track need 1ms to start the first frame. LT score will be 199.

2.3.6 Closeness of track

For a pair of associated GT track and system track, a dataset consisting of closeness of track pairs:

$$CT(GT_i, ST_j) = \{A(GT_{i1}, ST_{j1}), \cdots, A(GT_{iN}, ST_{jN})\} \quad (9)$$

$A(GT_{it}, ST_{jt})$ is formula in Eq.1.

To compare all M pairs in one video sequence, the closeness of this video as the weighted average of track closeness of all M pairs:

$$CMT = \frac{\sum_{t=1}^{M} CT_t}{\sum_{t=1}^{M} Length(CT_t)} \quad (10)$$

The weighted standard deviation of track closeness for the whole sequence:

$$CTD = \frac{\sum_{t=1}^{M} Length(CT_t) \times std(CT_t)}{\sum_{t=1}^{M} Length(CT_t)} \quad (11)$$

Where std(CT_t) is the standard deviation of CT_t.

2.3.7 Track matching error (TME)

TME is the average distance error between a system track and the GT track. The smaller the TME, the better the accuracy of the system track will be.

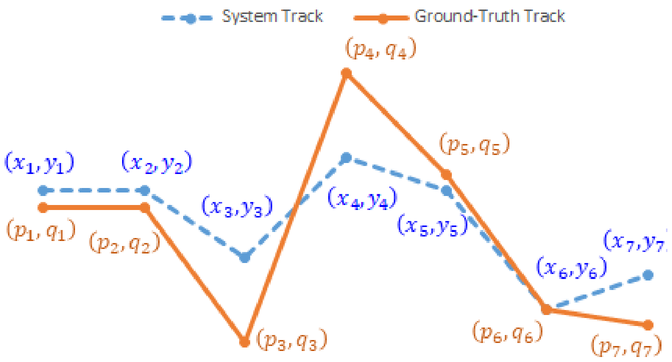


Fig.8. Example of a pair of trajectories

The estimate for TME as:

$$TME = \frac{\sum_{k=1}^{N} Dist(GTC_{ik}, STC_{jk})}{Length(GT_i \cap ST_j)} \quad (12)$$

Where $Dist(GTC_{ik},\ STC_{jk})$ is the Euclidean distance between the centroids of GT and the system track. GTC and STC are respectively the coordinates of the center of the ground-truth track and system track.

2.3.8 Track Completeness

Track Completeness (TC) is defined as the time span that the system track overlapped with GT track divided by the total time span of GT track:

$$TC = \frac{\sum_{k=1}^{N} O(GT_{ik}, ST_{jk})}{Number\ of\ GT_i} \quad (13)$$

Where:

$$O(GT_{ik}, ST_{jk}) = \begin{cases} 1 \; if \; A(GT_{ik}\,,\, ST_{jk}) > \; T_{OV} \\ 0 \; if \; A(GT_{ik}\,,\, ST_{jk}) \leq \; T_{OV} \end{cases} \quad (14)$$

The overlapping between GT_{it} and ST_{jt} is $A(GT_{ik}\,,\, ST_{jk})$ (Eq.1) and T_{OV} is track overlap threshold. When TC's value is 1 then it is a fully complete track.

2.4 Methodology

Although the object tracking methods have the same goal. However, different methods will give different results and the methodology of each method also differs based on a general assessment of the basic measurements during the test. Along with a set of measures that compare the outputs of each motion tracking system with the ground-truth to evaluate each method's performance. In that set of measures, it is important to define the concepts of spatial and temporal overlap [18] between tracks, which are required to quantify the level of matching between Ground Truth (GT) tracks and System (ST) tracks, both in space and time.

2.5 Traditional object tracking methods

For each video sequence captured from the camera, each frame has link points between the frame before and after it. Therefore, determining the link points belonging to the same object to be tracked for each frame is the goal of the problem. So, we need to use a tracking model that can track the movement and describe the image of the target changing frame by frame. In case of loss of track, if the tracked object is lost after 20 frames, re-detection of the object will be performed.

2.5.1 Mean Shift

Mean Shift [1, 2, 3] is an algorithm applied in data clustering. The reason this method is applied to the object tracking problem is because it is a hill climbing algorithm that continuously changes a data point to the average value of the data points in the vicinity of the object. Thus, when there is a change in the pixel value at the same object containing the object in each frame, it is possible to predict the position of the object in the next frame based on the average displacement point. Through our experiment, the threshold is set to 0.95.

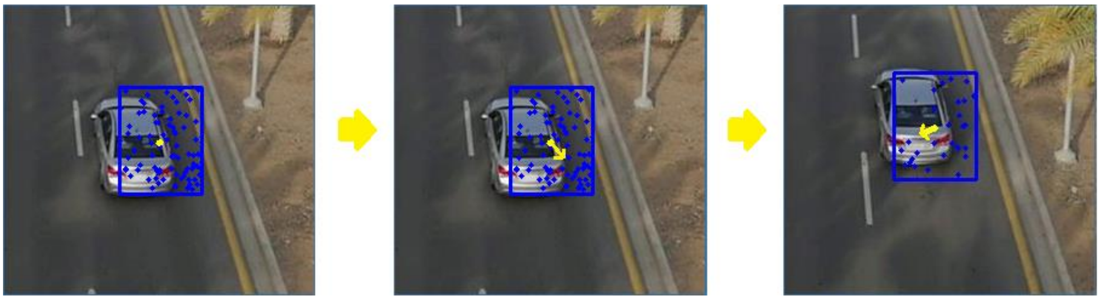


Fig. 9. Example of Mean Shift Tracking

ALGORITHM 1: Mean Shift

```
input first frame.
initialize target position y
set threshold
calculate histogram of object data with kernel h
while mean shift vector is greater than threshold do
        choose a candidate that has center at initial position y
        calculate histogram of candidate (new location) z with kernel h
        calculate weighting map w with histograms (object y and candidate z)
        calculate mean shift vector with weighting map w
while measure similarity between object model and candidate z smaller measure similarity between object model and target y do
        $z \leftarrow (z + y)\frac{1}{2}$
        end
        calculate histogram of object data with kernel h
end
```

Weaknesses: The selection of a window size is not trivial, inappropriate window size can cause modes to be merged, or generate additional “shallow” modes and often requires using adaptive window size.

2.5.2 Kalman Filter

The Kalman Filter [5, 28, 32] is used to track the points in the noisy image and assumes that the state variables are normally distributed (Gaussian noise). Tracking system based on the Kalman Filter consists of 2 steps, prediction and correction. In the prediction step, the new state variable is estimated based on the state model and the next step (correction step) is used to update the object's state based on the current observation. At model initialization, measurement noise covariance has value of 0.1 and process noise covariance has value of 0.01.

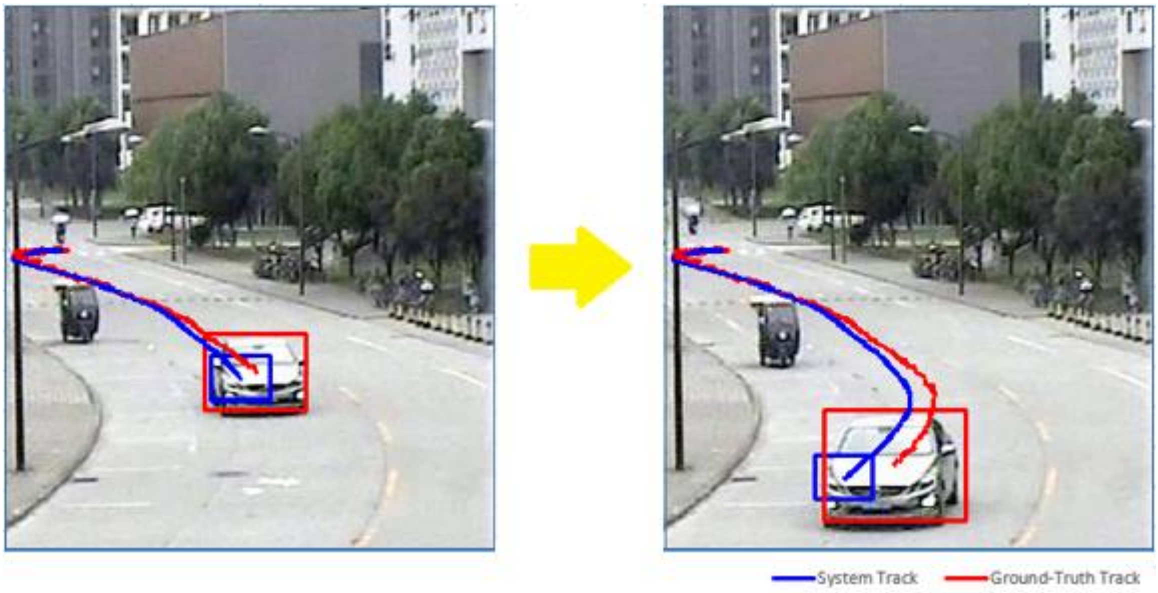


Fig. 10. Example of Kalman Filter

ALGORITHM 2: Kalman Filter

```
input first frame.
initialize target position $\hat{x}_{1,0}$ and covariance $P_{1,0}$
while true do
predict next state $\hat{x}_{n+1,n} \leftarrow F\hat{x}_{n,n} + G\hat{u}_{n,n}$ and covariance $P_{n+1,n} \leftarrow FP_{n,n}F^T + Q$
    update observation state
    $R_n \leftarrow E(v_n v_n^T)$
    $K_n \leftarrow P_{n,n-1}H^T(HP_{n,n-1}H^T + R_n)$
    $\hat{x}_{n,n} \leftarrow \hat{x}_{n,n-1} + K_n(z_n - H\hat{x}_{n,n-1})$
    $P_{n,n} \leftarrow (I - K_nH)P_{n,n-1}(I - K_nH)^T + K_nR_nK_n^T$
    $Q \leftarrow E(w_n w_n^T)$
End
```

Because the Kalman Filter assumes that the state variables are normally distributed (Gaussian), in the case of state variables that do not follow the Gaussian distribution, it is possible to give a poor estimate.

2.5.3 Particle Filter

Particle Filter [9] is a type of genetic simulation algorithm used to filter problems in signal analysis and time series analysis. To solve the case where the Kalman Filter [5] gives a poor estimate when the state variables do not follow a Gaussian distribution, the model will assume many random samples (following a certain distribution) around the tracked object in the current frame and move to the sample position that has the closest probability to the previous state. This model consists of three steps: selection, prediction, and correction. in the selection step, select random N samples by previous state. In the prediction step, for each selected sample, generate a new sample by zero mean Gaussian error and non-negative function, In the correction step, weights corresponding to the new samples are update. Experimentally, the particle filter parameter includes the standard deviation of the velocity (has value of 0.1), the standard deviation of the feature (set to 5), and the standard deviation of the change of position (has value of 25) with 4000 random samples.

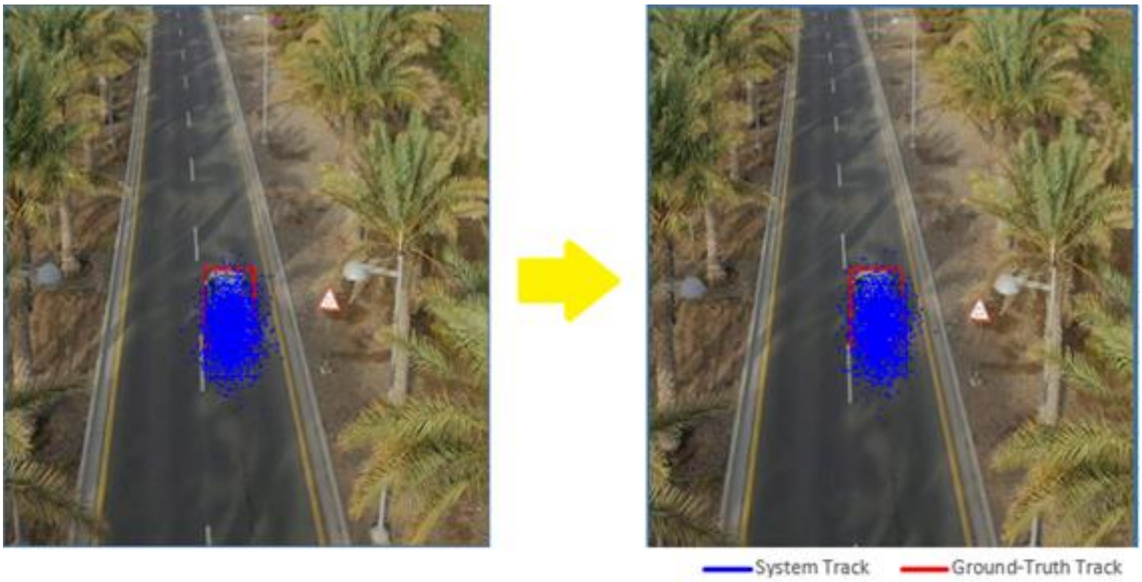


Fig. 11. Example of Particle Filter (random N samples by previous state)

ALGORITHM 3: Particle Filter

```
input first frame.
Generate samples set $\{x_0^i\}_{i=1}^N$ from the initial distribution $p(x_0)$, set $k = 1$
while true do
    predict sample $x_k^i \sim p(x_k \mid x_{k-1}^i),\ i = 1, \dots, N$
    update observation state
    Once of observation data $z$ is measure
    Evaluate weight of sample $\widetilde{w_k^i} \leftarrow w_{k-1}^i p(z_k \mid x_{k-1}^i),\ i = 1, \dots, N$
    Normalize weight $w_k^i \leftarrow \frac{\widetilde{w_k^i}}{\sum_t^N \widetilde{w_k^t}},\ i = 1, \dots, N$
    Generate new samples set $\{x_k^j\}_{j=1}^N$ by resampling (with replacement) $N$ times from $\{x_k^i\}_{i=1}^N$, where $\Pr(x_k^j = x_k^i) = w_k^i$ and set $w_k^i \leftarrow \frac{1}{N}$
end
```

With very high-dimensional systems, this approach doesn't work very well. on the other hand, if the current random samples do not belong to the tracked object, but they have a higher probability of being selected, this may lead to bad results.

3. Proposed Approach

In this section we propose to replace the color feature with the feature SURF to resolve shape variances when having large motion. Moreover, we propose the fusion of Mean Shift and Kalman Filter [33] as well as Mean Shift and Particle Filter to get the advantages of those. The goal of this fusion work is to want the system to run more stable in tracking performance.

3.1 Kalman filter base on Mean Shift

In order to solve the case of large motion tracking between 2 consecutively processed frames, the proposed combination of Mean Shift [1] and Kalman Filter [4, 5, 6] is a well point. First, the Mean Shift method is used to calculate the exact position of the object area for the current frame, then the Kalman Filter is used to predict the next position for Mean Shift iterations in the next frame. In Mean Shift method, robust features (SURF) histogram is used instead of color histogram. During the test of the parameters, the parameter for the Kalman filter is like section 5.3 but the threshold prediction is 0.7.

ALGORITHM 4: Kalman Filter based on Mean shift

```
input first frame.
initialize target position y
set threshold
calculate SUFT histogram of object data with kernel h
while mean shift vector is greater than threshold do
    choose a candidate that has center at initial position y
    Using Kalman Filter to predict next state
    calculate SUFT histogram of candidate (new location) z with kernel h
    calculate weighting map w with histograms (object y and candidate z)
    calculate mean shift vector with weighting map w
    then predict object's position if bhattacharyya distance between histogram of candidate z and histogram of
    target y is greater than threshold
    calculate histogram of object data with kernel h
    Updates the predicted state from object's position has been predict
end
```

The algorithm does not work well when there is an occlusion. In addition, different angles of the same object also affect the results because the SURF feature is detected incorrectly or not found in the area where the object is located.

3.2 Particle Filter based on Mean Shift

To solve the problem of not being able to track when partially obscured (Fig. 12) for the Kalman Filter [5] based on Mean shift [1] algorithm, the proposed method is Particle Filter [7, 8, 16] on Mean shift. Firstly, we use the Mean shift algorithm based on robust features to calculate an accurate location in current frame. The second, we use key points clustering for calculating the histogram. If the histogram match of the object region of the current frame is compared with region of the next frame reaching the threshold, the new position is updated for the object. otherwise randomizes regions according to a given distribution, constructs histograms for those regions and uses Particle Filter to predict the position of the object for the next frame. Experimentally, cluster size is 100 and the standard deviation of the velocity is set to 0.1, the standard deviation of position has value of 5, standard deviation of feature is set to 5 and the threshold prediction is 0.7 with 200 random samples for the system to work properly.

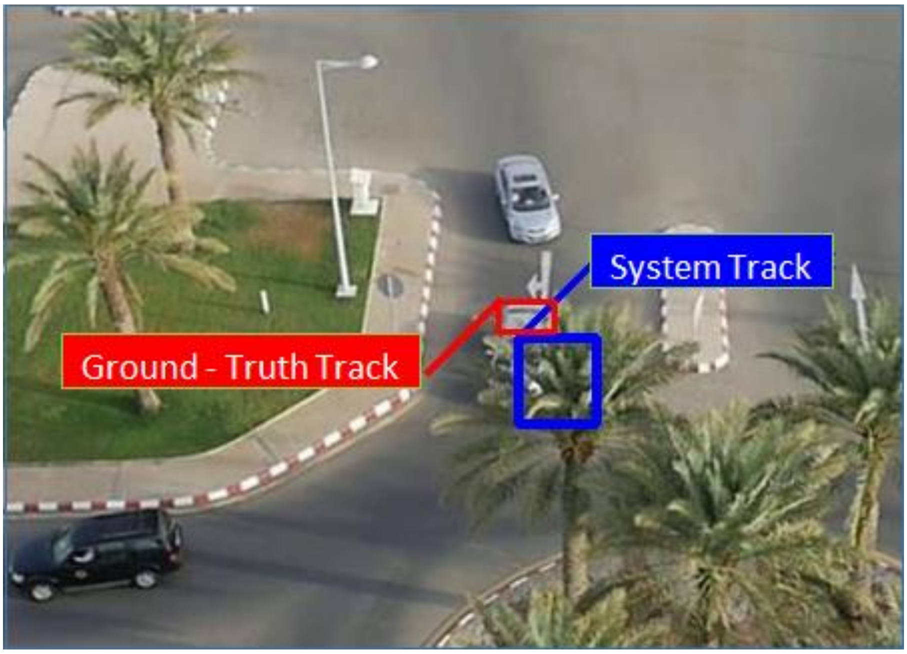


Fig. 12. Partially obscured for the Kalman Filter base on Mean shift algorithm

ALGORITHM 5: Particle Filter based on Mean shift

```
input first frame.
initialize target position y
set threshold
calculate SUFT histogram of object data with kernel h
calculate SUFT histogram for target' center for mean shift
Generate samples set $\{x_0^i\}_{i=1}^N$ from the initial distribution $p(x_0)$, set k = 1
while true do
    if mean shift vector is greater than threshold do
        choose a candidate that has center at initial position y
        calculate histogram of candidate (new location) z with kenel h
        calculate weighting map w with histograms (object y and candidate z)
        calculate mean shift vector with weighting map w
        then predict object's position if bhattacharyya distance between histogram of candidate z and histogram of target y is greater than threshold
        calculate histogram of object data with kernel h
         Updates the predicted state from object's position has been predict
    else
        predict sample $x_k^i \sim p(x_k \mid x_{k-1}^i),\ i = 1, \dots, N$
        update observation state
        Once of observation data z is measure
        Evaluate weight of sample $\widetilde{w_k^i} \leftarrow w_{k-1}^i p(z_k \mid x_{k-1}^i),\ i = 1, \dots, N$
        Normalize weight $w_k^i \leftarrow \frac{\widetilde{w_k^i}}{\sum_t^N \widetilde{w_k^t}},\ i = 1, \dots, N$
        Generate new sample set $\{x_k^j\}_{j=1}^N$ by resampling (with replacement) N times from $\{x_k^i\}_{i=1}^N$, where $\Pr(x_k^j = x_k^i) = w_k^i$ and set $w_k^i \leftarrow \frac{1}{N}$
    end
end
```

4. The Experimental Results And Discussion

4.1 Experimental Result

VOT Challenge Dataset [10]: VOT Challenge is a dataset used to test and evaluate single object (vehicles) tracking models, not necessarily linked video sequences, it means the current video is not sequel of the object in the previous video. In the normal case of the VOT dataset, the object can be tracked well when the object does not have many different angles of the same object and there is not much occlusion (shown in Fig. 13).

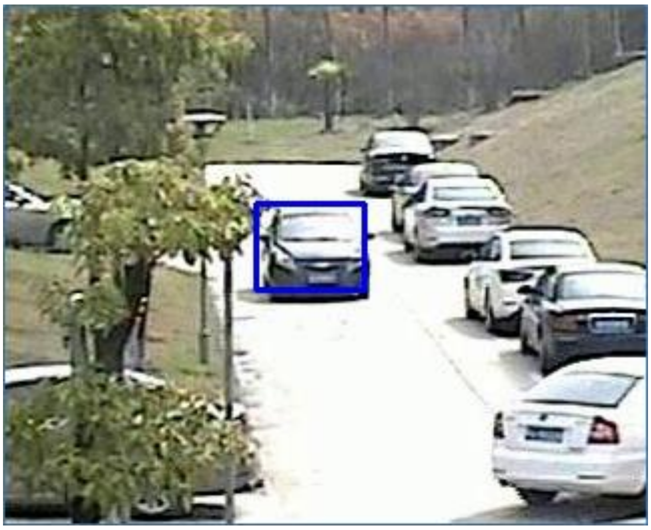
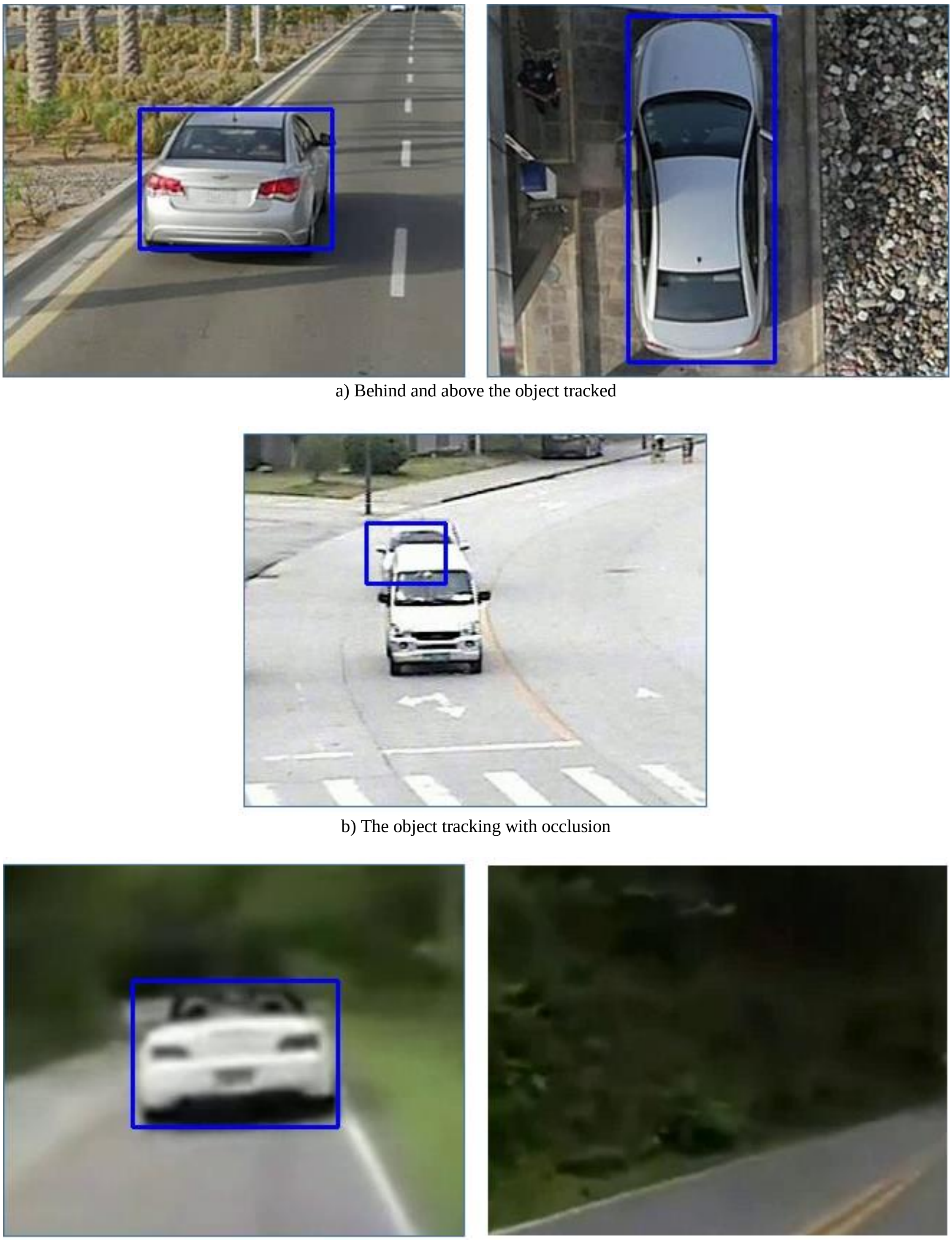


Fig. 13. Object tracked in normal case

However, there are still hard cases such as the angle of the tracked object being changed a lot (Fig. 14.a), occlusion (Fig. 14.b), the camera moves out of the scene with the object tracked (Fig. 14.c) and especially the influence of object's headlight or the outside light that changes the color of the object (shown in Fig. 15).

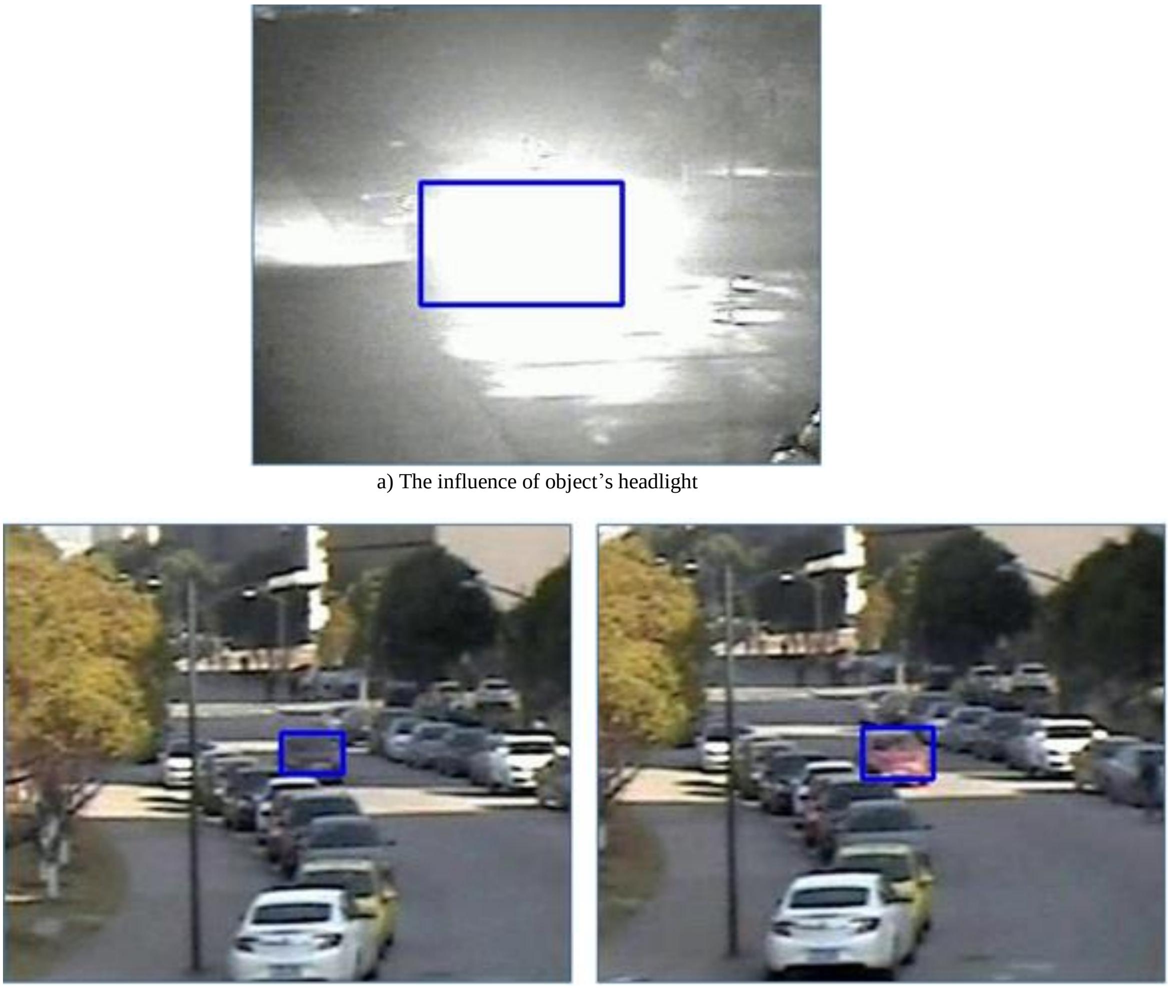


a) Behind and above the object tracked

b) The object tracking with occlusion

c) The image on the left is the image of the object tracked and the image on the right is the image that the camera moves out of the scene with the object tracked

Fig. 14. Example for hard case



a) The influence of object’s headlight

b) The influence of sunlight

Fig. 15. Example for hard case

Table II. The results of the experiment for VOT challenge dataset

Method	The Temporal Overlap (for the case of TP)	The Temporal Overlap (for the case of FP)	Sufficient Spatial Overlap	Track Fragmentation	Track Matching Error	Track Completeness	Latency of The System Track
Mean shift	0.756353	0.737282	0.183718	64	56.7355	0.366613	0.00434427
Kalman	0.83916	0.816722	**0.464464**	39.75	**33.7724**	**0.588341**	0.0104069
Particles Filter	**0.907971**	**0.884276**	0.293564	40.25	39.5286	0.328014	**0.00384504**
Kalman Filter base on Mean shift	0.857122	0.833882	0.293401	31.35	37.5002	0.334449	0.142119
Particle Filter base on Mean shift	0.872728	0.849475	0.299619	**31.1**	38.1011	0.341349	0.122689

UAV123 Dataset [11]: UAV123 dataset is the same as VOT dataset, which is used to test the results of the object tracking model and to track the single object (vehicles). However, mainly the video sequences have a common object tracking. UAV123 dataset also has normal cases and hard cases is like VOT dataset (the angle of the tracked object being changed a lot, occlusion, the camera moves out of the scene with the object tracked and the influence of object's headlight or the outside light that changes the color of the object). There are also cases where the camera is far away from the object tracked and the object too small to detect and track (shown in Fig. 16).

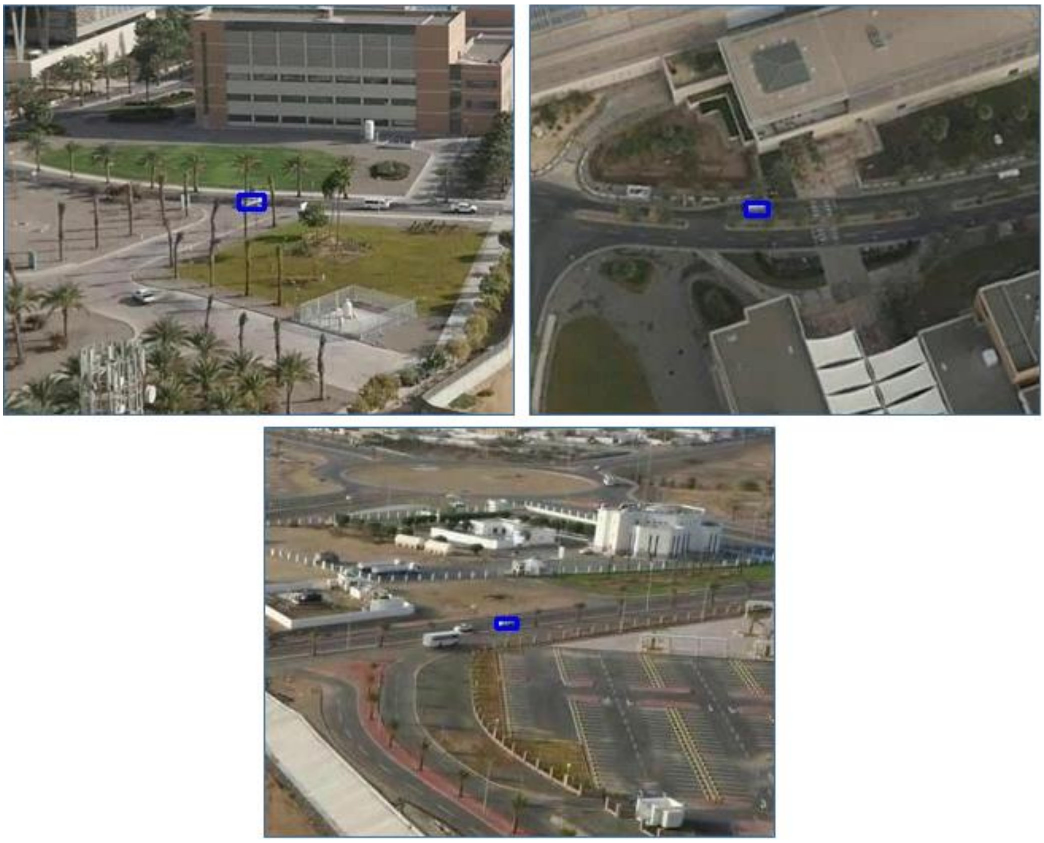


Fig. 16. Example for the case of object too small to detect and track

Table III. The results of the experiment for UAV123 dataset

Method	The Temporal Overlap (for the case of TP)	The Temporal Overlap (for the case of FP)	Sufficient Spatial Overlap	Track Fragmentation	Track Matching Error	Track Completeness	Latency of The System Track
Mean shift	0.79685	0.783789	0.26839	15.0455	46.1091	0.493041	0.00752362
Kalman Filter	0.788121	0.773512	**0.409057**	8.22727	35.7144	**0.511822**	0.0191109
Particles Filter	**0.866259**	**0.848228**	0.293303	**4.04545**	35.6452	0.353229	**0.00575516**
Kalman Filter base on Mean shift	0.832439	0.815819	0.349667	6.63636	**28.8233**	0.461452	0.654711
Particle Filter base on Mean shift	0.841045	0.824142	0.356746	6.54545	29.6678	0.474409	0.275997

4.2 Discussion

The purpose of this discussion is to provide an evaluation for each type of experiment-based model, explaining the strengths and weaknesses of each model that tracks the subject during experimentation. This discussion not only links to the introduction, but also implements the hypotheses and literature reviewed for each object tracking model. On the other hand, the current models can point out directions that need to be addressed in the future from the challenges and experimental results.

From the experimental results, we realize that the accuracy of the Particle Filter method is the highest with 0.907971 (Correct Detected Track) on VOT Challenge Dataset and 0.866259 (Correct Detected Track) on UAV123 Dataset, however, the results are unstable. because it's easy to make mistakes. Compared with the two proposed methods, although these two methods have lower accuracy than the Particle Filter method, they operate more stably.

Table IV. Comparison of vehicle tracking methods pros and cons

Method	Pros	Cons
Mean shift	• Invariant to pose and viewpoint • Often no need to update reference color model • Low computational cost (easily real-time)	• Position estimates prone to fluctuation • Scale and orientation not well captured • Sensitive to color clutter • Problems with sudden movements and occlusions
Kalman Filter	• Simple updates • Compact • Efficient	• Can be sensitive to process noise • Unimodal distribution, only single hypothesis • Restricted class of motions defined by linear model
Particle Filter	• Non-Gaussian and multi-modal distributions • Non-linear dynamic systems • Can be parallelized	• Degeneracy problem • High number of particles needed • Can be computationally expensive • Choice of importance density is crucial
Kalman Filter base on Mean shift	• Non sensitive to color clutter • Abrupt motion changes are resolved • Works more accurately than Mean shift method and Kalman Filter method • Low error rate	• Problems with candidate window has an unsatisfactory distance from the area that includes the current object • It is easy to fall into a situation where the SURF features are not found • Too slow processing
Particle filter base on Mean shift	• Improved processing speed compared to Kalman Filter base on Mean shift method • Low error rate	• When the SURF features are not found, it can fall into the bad situation of Particle Filter method • Slower than Particle Filter method

Furthermore, the comparison table (Table IV) of the advantages and disadvantages of each method and the experiment results show that the proposed tracking system which utilizes multi features to represent the target model has better performance evaluation results in contrast to other comparison algorithms. However, the proposed methods still have disadvantages, both are slow and hard to operate in real time system. For Kalman Filter base on Mean shift method, the cause of slow processing is because when SURF features are not found, the system needs to search for SURF features in a wider range. For Particle filter base on Mean shift method, when the SURF features are not found, it can fall into the bad situation of Particle Filter method. We need a different method or better feature usage to get high performance.

5. Conclusion

In this work, we do some literature reviewed to present surveys for traditional object tracking models and brief review of the topic related to object tracking problems based on these traditional object tracking methods We initially built traditional object tracking models including Mean Displacement, Kalman Filter and Particle Filter. Then, extend these methods by combining methods such as the Kalman filter based on the Mean shift and the Particle Filter based on the Mean shift. Based on the experimental results, the Mean-shift model shows that this model only works well for normal cases, but on 2 test data sets (VOT and UAV123 dataset) the results are lower than other methods. Experimentally, Mean-shift method has a TO score of 0.756353 on the VOT dataset and 0.79685 on the UAV123 dataset in the case of TP. The Kalman filter still has some highlights through the experimental results, the ability between ST (System Track) and GT (Ground Truth Track) to link together is very high with SSO score of 0.464464 on VOT dataset and 0.409057 on UAV123 dataset but does not provide stability in each case of the 2 test data sets. Particle filter is the same, if the object has a lot of change in feature or the feature of the outside of object tracked with higher confidence, the result is no longer true, Particle filter method has TME score of 39.5286 on VOT dataset and TME score of 35.6452 on UAV123 dataset. Two hybrid methods (Kalman filter base on Mean shift method and Particle filter base on Mean shift method) are more stable, although they do not achieve outstanding results on the metrics. However, these two methods (Kalman filter base on Mean shift method and Particle filter on Mean shift method) maybe fall into a state that the SURF feature of the object to be tracked cannot be detected or confused with the feature of the background or the object to be tracked at a different angle than the previous frame, since the feature is not found can lead to increased latency of the processing system. Specifically, LT score of 0.654711 on UAV123 dataset when experimenting with method Kalman filter base on Mean shift method. Particle filter on Mean shift method is more stable than Kalman filter based on Mean shift. From the above experimental results, we know that for the object tracking model to have a better set of features than the current ones (low-level features), we need a model to extract high-level features (can be a deep learning network), by using the extracted high-level features we can apply these features to the traditional object tracking model to get better performance. In the future, the combination of deep learning network with traditional methods can make the object tracking model work better.

References

[1] D. Comaniciu; P. Meer Mean shift: A Robust Approach Toward Feature Space Analysis. IEEE Transactions on Pattern Analysis and Machine Intelligence 24(5), 603-619, May 2002, doi: 10.1109/34.1000236.

[2] COMANICIU, D. AND MEER, P. 1999. Mean shift analysis and applications. In IEEE International Conference on Computer Vision (ICCV). Vol. 2. 1197–1203, doi: 10.1109/ICCV.1999.790416.

[3] H. Wang, X. Wang, L. Yu and F. Zhong, "Design of Mean Shift Tracking Algorithm Based on Target Position Prediction," 2019 IEEE International Conference on Mechatronics and Automation (ICMA), 2019, pp. 1114-1119, doi: 10.1109/ICMA.2019.8816295.

[4] Jinya Su; Baibing Li; Wen-Hua Chen (2015). "On existence, optimality and asymptotic stability of the Kalman filter with partially observed inputs". Automatica. 53: 149–154. doi:10.1016/j. automatica.2014.12.044.

[5] Lim Chot Hun, Ong Lee Yeng, Lim Tien Sze and Koo Voon Chet (June 8th, 2016). Kalman Filtering and Its Real ‐ Time Applications, Real-time Systems, Kuodi Jian, IntechOpen, DOI: 10.5772/62352.

[6] Feng Xiao; Mingyu Song; Xin Guo; Fengxiang Ge. Adaptive Kalman filtering for target tracking. 2016 IEEE/OES China Ocean Acoustics (COA), 2016, pp. 1-5, doi: 10.1109/COA.2016.7535797.

[7] Oğuzhan Gültekİn, Bilge Günsel, "Robust object tracking by variable rate kernel particle filter", 2018 26th Signal Processing and Communications Applications Conference (SIU), 2018, pp. 1-4, doi: 10.1109/SIU.2018.8404479.

[8] Marina A. Zanina, Vitalii A. Pavlov, Sergey V. Zavjalov, Sergey V. Volvenko, "TLD Object Tracking Algorithm Improved with Particle Filter". 2018 41st International Conference on Telecommunications and Signal Processing (TSP), 2018, pp. 1-4, doi: 10.1109/TSP.2018.8441515.

[9] Jung Uk Cho; Seung Hun Jin; Xuan Dai Pham; Jae Wook Jeon; Jong Eun Byun; Hoon Kang. A Real-Time Object Tracking System Using a Particle Filter. 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2006, pp. 2822-2827, doi: 10.1109/IROS.2006.282066.

[10] VOT Matej Kristan, Jiri Matas, Aleš Leonardis, Tomáš Vojíř, Roman Pflugfelder, Gustavo Fernández, Georg Nebehay, Fatih Porikli and Luka Čehovin, "A Novel Performance Evaluation Methodology for Single-Target Trackers", PAMI, vol. 38, no. 11, pp. 2137-2155, 1 Nov. 2016, doi: 10.1109/TPAMI.2016.2516982.

[11] UAV123 Matthias Mueller, Neil Smith, and Bernard Ghanem, "A Benchmark and Simulator for UAV Tracking", ECCV, 2016. 9905. 445-461. 10.1007/978-3-319-46448-0_27.

[12] C. Cuevas, E. M. Ynez, and N. Garca, Labeled dataset for integral evaluation of moving object detection algorithms: LASIESTA, Computer Vision and Image Understanding, vol. 152, pp. 103-117, 2016.

[13] A. Geiger, P. Lenz and R. Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite," 2012 IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 3354-3361, doi: 10.1109/CVPR.2012.6248074.

[14] Alper Yilmaz, Omar Javed, Mubarak Shah. Object tracking: A survey. ACM Computing Surveys: Vol 38, No 4, 2006, DOI: 10.1145/1177352.1177355.

[15] Mustansar Fiaz, Arif Mahmood, Sajid Javed, and Soon Ki Jung. 0000. Handcrafted and Deep Trackers: Recent Visual Object Tracking Approaches and Trends. ACM Comput. Surv. 0, 0, Article 0 (0000), 36 pages.

[16] Vaswani, N.; Rathi, Y.; Yezzi, A.; Tannenbaum, A. (2007). "Tracking deforming objects using particle filtering for geometric active contours". IEEE Transactions on Pattern Analysis and Machine Intelligence. 29 (8): 1470–1475, Aug. 2007, doi: 10.1109/TPAMI.2007.1081.

[17] L. M. Brown, A. W. Senior, Ying-li Tian, Jonathan Connell, Arun Hampapur, Chiao-Fe Shu, Hans Merkl, Max Lu, “Performance Evaluation of Surveillance Systems Under Varying Conditions”, IEEE Int'l Workshop on Performance Evaluation of Tracking and Surveillance, Colorado, Jan 2005.

[18] F. Bashir, F. Porikli. “Performance evaluation of object detection and tracking systems”, IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETS), June 2006.

[19] Sven Ubik; Jiří Pospíšilík. Video Camera Latency Analysis and Measurement. IEEE Transactions on Circuits and Systems for Video Technology (Volume: 31, Issue: 1, Jan. 2021): 140 - 147. DOI: 10.1109/TCSVT.2020.2978057.

[20] T. Ellis, “Performance Metrics and Methods for Tracking in Surveillance”, Third IEEE International Workshop on Performance Evaluation of Tracking and Surveillance, June, Copenhagen, Denmark, 2002, pp26-31.

[21] J. Nascimento, J. Marques, “Performance evaluation of object detection algorithms for video surveillance”, IEEE Transactions on Multimedia, vol. 8, no. 4, pp. 761-774, Aug. 2006, doi: 10.1109/TMM.2006.876287.

[22] N. Lazarevic - McManus, J.R. Renno, D. Makris, G.A. Jones, “An Object-based Comparative Methodology for Motion Detection based on the F-Measure”, in 'Computer Vision and Image Understanding', Special Issue on Intelligent Visual Surveillance TO APPEAR, 2007, Volume 111, Issue 1, 2008, Pages 74-85, ISSN 1077-3142, 10.1016/j.cviu.2007.07.007.

[23] C.J. Needham, R.D. Boyle. “Performance Evaluation Metrics and Statistics for Positional Tracker Evaluation” International Conference on Computer Vision Systems (ICVS'03), Graz, Austria, April 2003, pp. 278 – 289. 278-289. 10.1007/3-540-36592-3_27.

[24] J. F. Henriques, R. Caseiro, P. Martins and J. Batista, "High-speed tracking with kernelized correlation filters", IEEE Trans. Pattern Anal. Mach. Intell., vol. 37, pp. 583-596, Mar. 2015, doi: 10.1109/TPAMI.2014.2345390.

[25] Z. Soleimanitaleb, M. A. Keyvanrad and A. Jafari, "Object Tracking Methods: A Review," 2019 9th International Conference on Computer and Knowledge Engineering (ICCKE), 2019, pp. 282-288, doi: 10.1109/ICCKE48569.2019.8964761.

[26] Y. Ivanov et al., "Adaptive moving object segmentation algorithms in cluttered environments," The Experience of Designing and Application of CAD Systems in Microelectronics, 2015, pp. 97-99, doi: 10.1109/CADSM.2015.7230806.

[27] K. R. Reddy, K. H. Priya and N. Neelima, "Object Detection and Tracking -- A Survey," 2015 International Conference on Computational Intelligence and Communication Networks (CICN), 2015, pp. 418-421, doi: 10.1109/CICN.2015.317.

[28] Hamed Tirandaz, Sassan Azadi,"Utilizing GVF Active Contours for Real-Time Object Tracking", IJIGSP, vol.7, no.6, pp. 59-65, 2015.DOI: 10.5815/ijigsp.2015.06.08.

[29] Haocheng Le, Linglong Hu, Yuanjing Feng,"Momentum Based Level Set Method For Accurate Object Tracking", International Journal of Intelligent Systems and Applications (IJISA), vol.2, no.2, pp.10-16, 2010. DOI: 10.5815/ijisa.2010.02.02.

[30] Adithya Urs, Nagaraju C, "Object Motion Direction Detection and Tracking for Automatic Video Surveillance", International Journal of Education and Management Engineering (IJEME), Vol.11, No.2, pp. 32-39, 2021. DOI: 10.5815/ijeme.2021.02.04.

[31] Muhammad Tayyab, Muhammad Tahir Qadri, Raheel Ahmed, Maryam Ahmad Dhool,"Real Time Object Tracking using FPGA Development Kit", International Journal of Information Technology and Computer Science (IJITCS), vol.6, no.11, pp.54-58, 2014. DOI: 10.5815/ijitcs.2014.11.08.

[32] G. Mallikarjuna Rao, Ch. Satyanarayana,"Object Tracking System Using Approximate Median Filter, Kalman Filter and Dynamic Template Matching", International Journal of Intelligent Systems and Applications (IJISA), vol.6, no.5, pp.83-89, 2014. DOI: 10.5815/ijisa.2014.05.09.

[33] Ravi Kumar Jatoth, Sampad Shubhra, Ejaz Ali,"Performance Comparison of Kalman Filter and Mean Shift Algorithm for Object Tracking", IJIEEB, vol.5, no.5, pp.17-24, 2013. DOI: 10.5815/ijieeb.2013.05.03.

Authors' Profiles

Vo Hoai Viet is a Lecturer and Senior Researcher at the University of Science, VNU-HCMC, Vietnam from 2012. He is currently working in Computer Vision at University of Science, VNU-HCMC, Vietnam. His research interests include Digital Image Processing, Programming Language, Computer Graphics, Computer vision, and Machine Learning.

Huynh Nhat Duy graduated from the University of Science, VNU-HCMC, Vietnam in in 2015. Now, he is pursuing the master's degree of Computer Science at University of Science, VNU-HCMC. His research interests include Image Processing and Computer Vision.

How to cite this paper: Vo Hoai Viet, Huynh Nhat Duy, " Object Tracking: An Experimental and Comprehensive Study on Vehicle Object in Video", International Journal of Image, Graphics and Signal Processing(IJIGSP), Vol.14, No.1, pp. 64-81, 2022.DOI: 10.5815/ijigsp.2022.01.06