American Journal of Applied Sciences

Original Research Paper

Visual Tracking using Invariant Feature Descriptor

[1]Lee-Yeng Ong, [1]Siong-Hoe Lau and [2]Voon-Chet Koo

[1]*Faculty of Information Science and Technology, Multimedia University, Melaka, Malaysia*
[2]*Faculty of Engineering and Technology, Multimedia University, Melaka, Malaysia*

Article history
Received: 05-07-2017
Revised: 07-08-2017
Accepted: 16-09-2017

Corresponding Author:
Lee-Yeng Ong
Faculty of Information Science and Technology, Multimedia University, Melaka, Malaysia
E-mail: lyong@mmu.edu.my

Abstract: The process of identifying the state of an object in a video sequence is referred as visual tracking. It is mainly achieved by using the appearance information from a reference image to recognize the similar characteristics from the other images. Since a digital image is built-up with rows and columns of pixels that are represented with finite set of digital values, the appearance information is measured with a mathematical formulation that is known as image intensity. The problem of distinguishing the intensity of the object of interest from the other objects and the surrounding background is always the main challenge in visual tracking. In this study, a novel invariant feature descriptor model is introduced to address the aforesaid problem. The proposed framework is inspired by the theoretical model of local features that has been widely-used for image recognition. From the large number of diversified scenarios in the surveillance applications, the performance of the proposed model is demonstrated with the benchmarked dataset of single-target tracking. The experiment results shown the advantage of our proposed model for tracking non-rigid object in the changing background as compared to other state-of-the-art visual trackers. In addition, the important aspects of the proposed model are analyzed and highlighted as well in the experimental discussions.

Keywords: Surveillance, Single-Target Tracking, Non-Rigid Object

Introduction

Visual tracking is a process that imitates the visual perception of a human eye to observe the dynamic configuration of an object or target in the real-world. In order to perceive the dynamic configuration of a target from a sequence of video frames, an object of interest from the captured scenario is monitored closely to determine its state changes, such as position, color and shape (Hartley and Zisserman, 2004). The appearance information from the object of interest is used as a reference image to recognize the similar characteristics from the other images. The characteristics are extracted directly from the digital value of the image pixels, which can be denoted as image intensity.

Meanshift algorithm is one of the state-of-the-art visual trackers that uses the image intensity to compute the color probability distribution of a target. It applies a non-parametric approach, which iteratively seek the mode or local maxima from the probability distribution (Fukunaga and Hostetler, 1975). Bradski has modified the algorithm into a continuously adaptive meanshift (Camshift) algorithm. The simpler implementation of Camshift algorithm and its flexibility to track a varying target size have overtaken the popularity in visual tracking. A computationally efficient face tracker for games' and graphics' controls are demonstrated in Bradski (1998). Both algorithms are highly dependent on the color histogram that is computed from the intensity of the reference image. The mixture of foreground and background colors in the reference image may leads to an incorrect mode-seeking result. If the target is non-rigid or articulated, the reference image (bounded in a rectangle) includes some colors from the background area, as shown in Fig. 1. Mostly of the reference images in Fig. 1 contain higher percentage of intensity that belongs to the background rather than the object of interest. Although foreground detection can be carried out by using segmentation method, but the consistent performance is limited to the video sequences captured from a static camera. Thus, a successful visual tracker has to be non-sensitive to the multiple colors of reference image and is applicable in the video sequence with changing background.

Science Publications

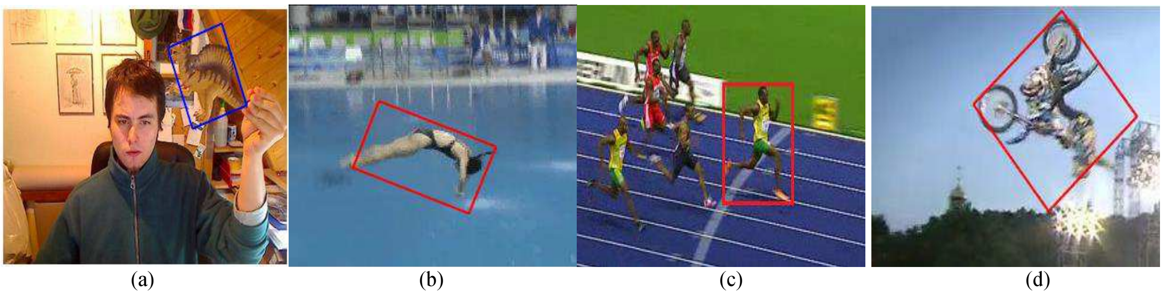


Fig. 1. Samples of non-rigid or articulated reference images: (a) dinosaur (b) diving (c) bolt (d) motocross

Instead of using the intensity of the reference image as a whole, the characteristics can be extracted from the distinctive appearance information resides in the reference image, which is also known as local features. A local feature is a special structure of image properties that is distinguishable among images. The local features ranging from point, edge, boundary, color, corner and motion have been commonly used for tracking object since decades ago (Dhome, 2009). A set of descriptor can be formulated by using the image pattern from the region surrounding a local feature in the reference image (Tuytelaars and Mikolajczyk, 2008). The set of descriptor formulated from the multiple local features are treated as the identity representation for tracking the object of interest in a sequence of video frames.

This paper presents a novel model that formulates an invariant feature descriptor from the local features (corner). Some preliminary results of this model have been published in Ong *et al*. (2014). A more detailed description of the proposed model will be described thoroughly in this study. From the large number of diversified scenarios in the surveillance applications, the performance and analysis of the proposed model is demonstrated with single-target tracking. The detailed experimental analysis and comparison are conducted with six benchmarked video sequences. Additional experiments are extended for investigating the properties and limitation of the proposed model.

The following section introduces the role of visual tracker and describes the research works done on different types of visual tracking. The existing algorithms that inspired the design and development of our proposed theoretical framework are also highlighted in the next section.

Related Work

Visual tracking algorithm is specifically designed to identify the state of a target throughout the video sequence in different types of video surveillance applications, such as robot navigation, traffic detection, sport analytics, gesture recognition and animal behavior analysis (Baltzakis *et al*., 2012; Diop *et al*., 2016; Linares-Sánchez *et al*., 2015; Mei and Ling, 2011; Santiago *et al*., 2011; Shvarts and Tamre, 2012). In the large number of diverse surveillance applications, visual tracking is generally divided into two categories, which are single-target tracking and multiple-target tracking. During the single-target tracking, the camera is actively following and monitoring a specific target from the beginning till the end of the video sequence. Therefore, a single-target visual tracker should be able to continuously detect the location of a specific target in the changing background, no matter how the motion and appearance changes happen to the target. On the other hand, multi-target tracking focuses on observing the state of a selected group of targets in the scene, such as people, vehicle or animal. A static camera is usually used to record a class of targeted entities that is entering and leaving the scene. Multi-target tracking algorithm discovers the existence of each entity with unique labeling, as well as registering their moving paths in the video sequence. Nevertheless, the responsibility of both visual trackers is focused on determining the position of the target in the video sequence.

To begin visual tracking, a Region Of Interest (*ROI*) is extracted from the reference image to obtain the intensity information. This information is normally processed as color probability distribution to assist the visual tracker to seek the most likely position of the target in the subsequent frames. The mixture colors from the foreground and background areas remains as the challenge for using color feature in a visual tracker. A series of continuous effort on the segmentation methods has been reported to eliminate the background disturbance from the reference image (Friedman and Russell, 1997; Pong and Bowden, 2001; Lee, 2005; Pnevmatikakis and Polymenakos, 2006; Hoseinnezhad *et al*., 2013; Kumar and Yadav, 2016). These methods can be successfully applied if the reference image is capturing in a static background with little changes of illumination and partial occlusion. However, this ideal situation is not always happening in the real-world surveillance. The tracking task becomes even complicated when the tracked target is non-rigid or articulated, such as human body parts.

Meanshift algorithm is the best known state-of-the-art visual tracker. The original meanshift algorithm is introduced as a non-parametric approach to solve the data clustering and noise filtering in the pattern recognition process. Due to the ability of meanshift algorithm to compensate noise and eliminate distractors (outliers) from the vision data, Camshift algorithm has modified it into a computationally efficient face tracker (Bradski, 1998). Camshift algorithm begins by computing the color histogram distribution from a *ROI* at the selected initial location. The *ROI* is dynamically located based on the changes of the color histogram distribution while the position of the target varies in time. An adaptive window size function has been suggested and proven to track human face efficiently. Yet, the Camshift algorithm still lacks of flexibility to accommodate the *ROI* for other non-rigid or articulated objects with non-specific color (Artner, 2008).

Instead of modeling the appearance information of the reference image as a whole, the *ROI* can be separated into smaller regions, where each region describes a part of the *ROI* with its own set of local descriptor. Figure 2 shows some samples of local descriptors that are formulated from the detected corner points in the *ROI*. Numerous part-based visual trackers become more popular partially due to their favorable property of adaptable *ROI* for various non-rigid or articulated objects (Adam *et al*., 2006; Nejhum *et al*., 2008; Izadinia *et al*., 2012; Liu *et al*., 2015). Part-based trackers can be generated from the simplest image pixels and image patches, as well as using the higher level interpretation from region descriptors. Region descriptors are selected from the smaller regions around the interest points within the *ROI*. These descriptors are distinguishable even though the foreground and background intensities are having a similar color distribution. In addition, descriptors can be used to dynamically adapt the *ROI* size for non-rigid or articulated objects (Lim and Kang, 2011).

The remaining of the paper is prepared as follows: Section 2 describes the algorithm of the proposed invariant feature descriptor for visual tracking. In Section 3, comparative experiments with the state-of-the-art visual trackers are reported and analyzed. Additional experiments are conducted to further investigate the properties of the proposed model. Concluded remarks are highlighted in the last section.

Proposed Invariant Feature Descriptor Model

An overview of the designated tasks in the proposed model is illustrated in Fig. 3. The model begins with a selected region of interest that contains a single target. Initially, distinctive feature points are detected from the *ROI* for the latter process, where a set of descriptor is formulating around the region of each feature point. Each set of descriptor is utilized to find the most likely position of the target in the consecutive frames. The following sections describe the methodology of each phase in the proposed model: Feature points detection, descriptor formulation and features matching.

Feature Points Detection

In the human visual system, corner point has been recognized as an important local feature to represent the context of an object in a scene. Experiments have been presented that human failed to recognize an object in a scene when the corner points are eliminated rather than removing the straight edges from the object (Biederman, 1987). A corner point in an image is defined as a pixel with the intensity value that appears to be different from its closest neighborhood. Corner is usually found at the edge (boundary between two regions) with a rapid change of intensity. Since decades ago, corner detection has been an attractive research topic for digital image processing (Rosenfeld and Johnston, 1973; Moravec, 1980; Shi and Tomasi, 1994; Harris and Stephens, 1988; Smith and Brady, 1997; Rosten and Drummond, 2006; Jian *et al*., 2015). This is due to the repeatability of the corner features in the same scene despite the changes of viewpoint. The massive increase of computing power to process live video streams also motivates the utilization of corner detection in visual tracking.

In order to determine a suitable corner detector, repeatability and efficiency are two of the important requirements to produce the same feature points correspond to multiple viewpoints within the video sequence. FAST detector has been proven to fulfill both requirements after comparing its performance with other popular corner detectors (Rosten and Drummond, 2006; Miksik and Mikolajczyk, 2012; Senst *et al*., 2012). FAST detector concludes a corner point if the intensities of the circular neighborhood pixels are significantly higher or lower than the intensity of the central point. A fixed radius of 16-pixels circulating a central point is illustrated in Fig. 4. The intensities of the pixels are selected from the location of north (1), south (9), east (5) and west (13) in a circle and compared with the intensity of the central point. The central point is deduced as a corner point if there are more than two of the selected pixels brighter or darker than the central point. Otherwise, the test criterion is continued on the remaining pixels until it satisfies the $n \leq 12$ contiguous pixels in the circle.

The performance of FAST detector by using different n values, ranging from 7 to 12 are depicted in Fig. 5. Test criterion is carried out on the n contiguous pixels before finalizing a corner point. The number of corner points are reducing whenever the n value increases because more contiguous pixels have to fulfill the test criterion. For the case of $n \leq 8$, FAST detector begins to respond strongly to edges and consumes greater processing time. FAST detector with $9 \leq n \leq 12$ shows an acceptable number of corner points and processing time for feature points detection task.

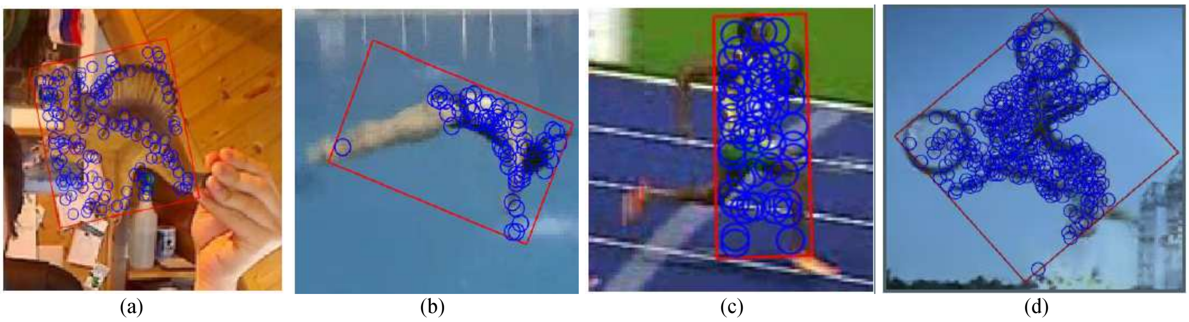


Fig. 2. Samples of local descriptors that are formulated from the detected corner points of non-rigid or articulated reference images for (a) dinosaur (b) diving (c) bolt (d) motocross

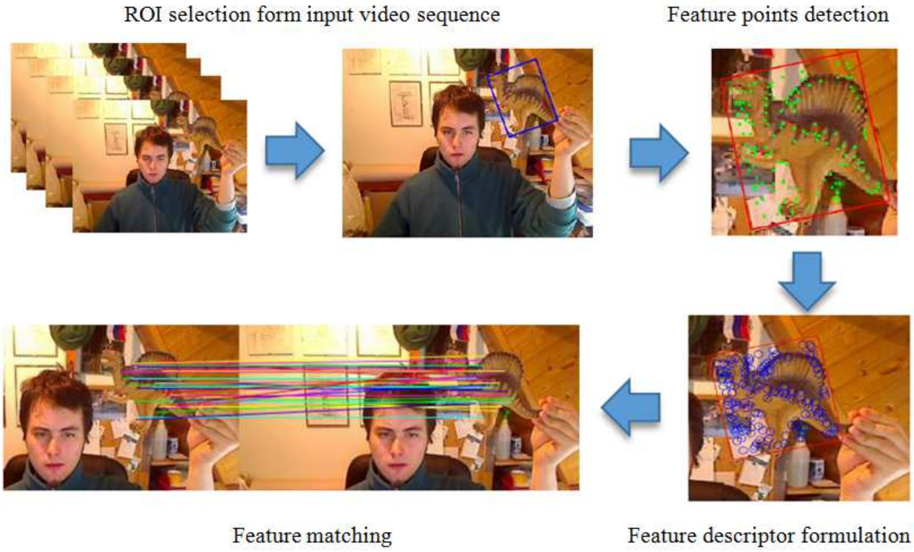

ROI selection form input video sequence
Feature points detection
Feature descriptor formulation
Feature matching

Fig. 3. Proposed framework of invariant feature descriptor model

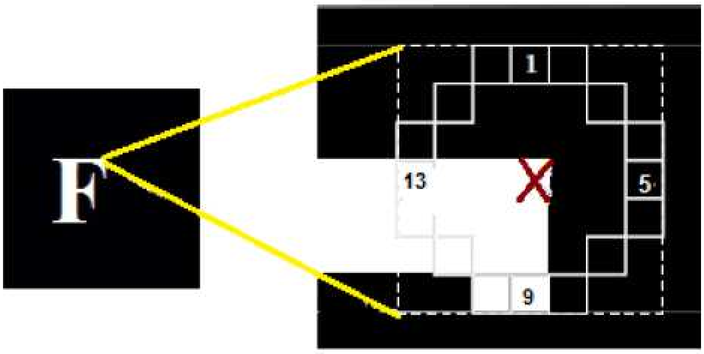


Fig. 4. Illustration of the circular neighborhood pixels of FAST detector

Since corner is a point defined in geometry without spatial extent, it would be difficult to localize various features in a video frame (Tuytelaars and Mikolajczyk, 2008). To provide an implicit spatial extent for each feature point, the intensity of local neighborhood is normally extracted and used as the identification for feature matching across video frames. The process of feature descriptor formulation has to determine the location, size and shape of the local neighborhood for each feature point. The next section describes the methodology to extract a set of descriptor for each feature point.

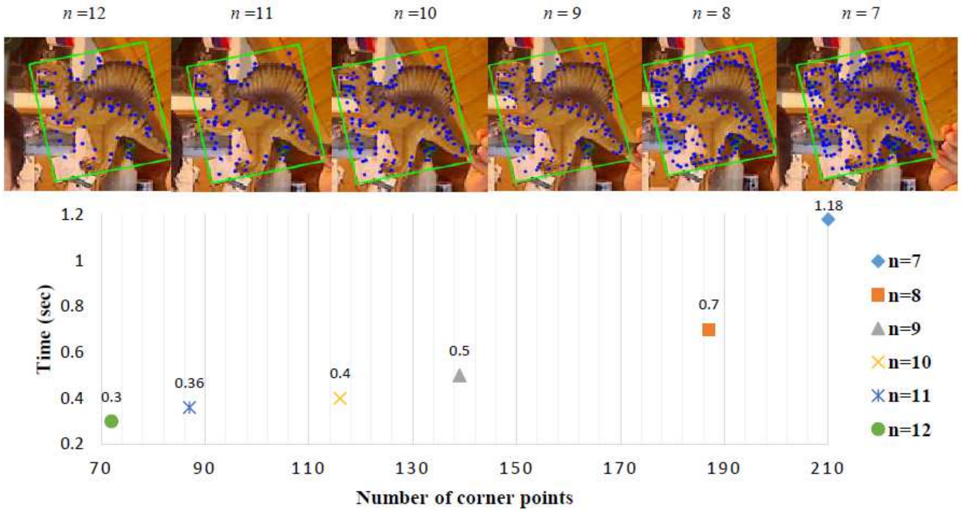


Fig. 5. The performance of FAST detector by using different

Feature Descriptor Formulation

A video frame is a static image taken from a video sequence. Note that a digital image is composed of a finite set of pixels, where each pixel contains the value of intensity at a set of spatial coordinates (x, y). Although a corner point is invariant against illumination and transformation, but it is lack of distinctive information content that is recognizable for feature localization in the consecutive frames. The ideal way to provide a salient description for each feature point is by extracting the informative content enclosed in the local neighborhood around the feature point (Tuytelaars and Mikolajczyk, 2008).

A salient descriptor has to provide an implicit spatial extent for each feature point, as well as to fulfill the transformation invariance property. Moment functions that have been widely used as global features for object recognition applications, are firstly introduced as a set of local feature descriptors in Ong *et al*. (2014). A set of moments function represents the geometrical properties of a two-dimensional digital image, such as shape, centroid, total mass, rotational inertia skewness and kurtosis (Mukundan and Ramakrishnan, 1998). These geometrical properties are extremely useful for describing the implicit spatial extent of a particular region around a feature point. The properties of an image can be generated from a set of geometric moments with the general definition given as:

$$G_{ij}=\iint_{\zeta} n^{i},y^{j}f(x,y)dxdy, i,j=0,1,2,3 \tag{1}$$

The moments function in Equation 1, G of order (i+j), is formulated with two-dimensional monomial functions in the image region of for $f(x, y)$. Geometric moments were firstly being introduced as the set of invariant descriptors for recognizing two-dimensional digital images (Hu, 1962). The presented set is able to eliminate the transformation factors, no matter in translation, scaling, reflection, skew and rotation. Since then, the research topic of moments function has been extensively explored for the past few decades (Almoosa *et al*., 2008; Chen *et al*., 2013; Costantini *et al*., 2011; Li *et al*., 2012). Every publication has reported its improved version of moments.

In order to eliminate the translation factor, geometric moments function is specified with respect to the image centroid (x_0, y_0) as the origin, which is stated in Equation 2. After that, Equation 3 is composed by eliminating the scale factor. Using the theory of algebraic invariants to derive rotation invariants with Hu's moments, a set of moment functions that are invariant with respect to scale, translation and rotation transformations is given in Equation 4 (Mukundan and Ramakrishnan, 1998):

$$\begin{aligned} &G_{ij}=\iint_{\zeta}(x-x_0)^{i}(y-y_0)^{j}f(x,y)dxdy \\ &i,j=0,1,2 \end{aligned} \tag{2}$$

$$\eta_{ij}\frac{C_{ij}}{C_{00}^{\rho}},\rho=\frac{i+j+2}{2} \tag{3}$$

$$\begin{aligned} M_1 &=\eta_{20}+\eta_{02} \\ M_2 &=(\eta_{20}-\eta_{02})^2+4\eta_{11}^2 \\ M_3 &=\eta_{20}\eta_{02}-\eta_{11}^2 \\ M_4 &=(\eta_{30}-3\eta_{12})^2+(3\eta_{21}-\eta_{03})^2 \\ M_5 &=(\eta_{30}+\eta_{12})^2+(\eta_{21}+\eta_{03})^2 \end{aligned} \tag{4}$$

Once a feature point has been detected with FAST detector, a set of local descriptor is formulated from the local neighborhood of each feature point. A local neighborhood region of 10x10 pixels is extracted and represented with the invariants descriptor of Equation 4. The proposed descriptor *ID*, as defined in Equation 5 stores the five different values of invariant moment functions for each feature point *p*:

$$ID(p) = [M_1 M_2 M_3 M_4 M_5] \tag{5}$$

Since each set of invariants descriptor constitutes the information of a region centered by a feature point, the combination of the entire sets of invariants descriptor depicts a unique representation of the *ROI* in the frame *n*:

$$ROI(n) = [ID(1)...ID(p)]^T, n, p = 1,2,3 \tag{6}$$

Equation 6 will be used to compute each *ROI* in the following experimental study of the single-target visual tracking. Table 1 listed some examples of descriptor that are computed with Equation 4 for single-target tracking from five different video sequences of VOT dataset (VOT, 2016). Each target in the *ROI* is bounded with a rectangle, which is depicted in Fig. 6. The first three samples of *ROI* in Fig. 6 are selected from the same video sequence (dinosaur) but each *ROI* undergoes different transformations, which are rotation and scaling. Based on Table 1, the variation values of descriptors computed for the similar *ROI* that undergoes different transformations are as low as 0.000016 and never exceeds 0.000308. Therefore, the proposed descriptor in Equation 4 has been proven to satisfy the invariants property against different transformations. In addition, the *ROI* from the other four video sequences are computed and compared with the *ROI* in video sequence *a*. The *ROI* from video sequence *g* is intentionally selected to show the discriminative power of the invariant descriptor since video sequence *a* and *g* are having similar background. The values of the calculated differences among descriptors have exceeded 0.05, which successfully shown the discrimination property of the proposed descriptor.

Feature Descriptor Matching

This section explains the methodology to estimate the most likely location of a target in the next video frame by using the position of *ROI* in the current frame. After getting the estimated *ROI* in next frame, a new set of feature descriptor is computed and match with the previous set of descriptor. Fig. 7 displays a concise pseudo code to estimate the most likely location of a target in the next frame by using the current *ROI*'s size and centroid, *ROI* (*n*). Feature points detection and descriptor formulation are carried out at the new location, *SA* (*n*+1) to compute a new set of descriptor, *ROI* (*n*+1).

In order to determine the matching pairs between two consecutive frames, the set of descriptor *ROI* (*n*-1) from the previous frame *Fr* (*n*-1) is associated in a certain criterion with the descriptor set of the current frame, *Fr*(*n*). To find out a suitable criterion for showing the correlation between two sets of descriptor, four types of matching criteria have been tested on the computed descriptors of Fig. 6. The matching result is illustrated in Fig. 8 to show the comparison between Pearson's Correlation Coefficient (CC), Mean Absolute Difference (MAD), Root Mean Squared Error (RMSE) and Sum of Absolute Difference (SAD). SAD criterion displays the most significant differences among the multiple video sequences and exhibits minor differences for the similar video sequence that undergoes various transformations. Hence, SAD is selected as the matching criterion, where a lowest value indicates a stronger association between two sets of descriptor.

Instead of using all the shortlisted pairs, only those highly reliable pairs are remained to improve the matching performance. An efficient way for filtering matching pairs is by using RANdom Sample and Consensus (RANSAC) algorithm. RANSAC algorithm estimates the possible homographies that elaborate the relation between descriptor pairs in different frames (Hartley and Zisserman, 2004). During the estimation process, the less reliable pairs or mostly known as the outliers are rejected. Figure 9 presents the procedure for descriptor association and matching pairs filtering. Based on the finalized set of descriptor from filtering process, the properties of the *ROI* in the new frame, such as location and centroid are redefined.

Experimental Evaluation

Experiments are conducted on six video sequences that are illustrated in Fig. 6 (a, d-h). The performance of the proposed model is evaluated based on different basic characteristics of the real-world objects. The basic characteristics of each video sequence are stated in Table 2. The resolution and total number of frames in each video sequence is clearly listed. In addition, the single-target is described in terms of type (rigid/ articulated), average size and size changes rate throughout each video sequence. The dinosaur, torus and fish sequences are composed of rigid object with minor changes in background and illumination. Whereas, the targets in bolt and diving sequences are moving body parts in different background and the *ROI*s are always positioned at the middle of the whole image frame. Motocross sequence is the most challenging among all, by experiencing the large size changes, high illumination and changing background.

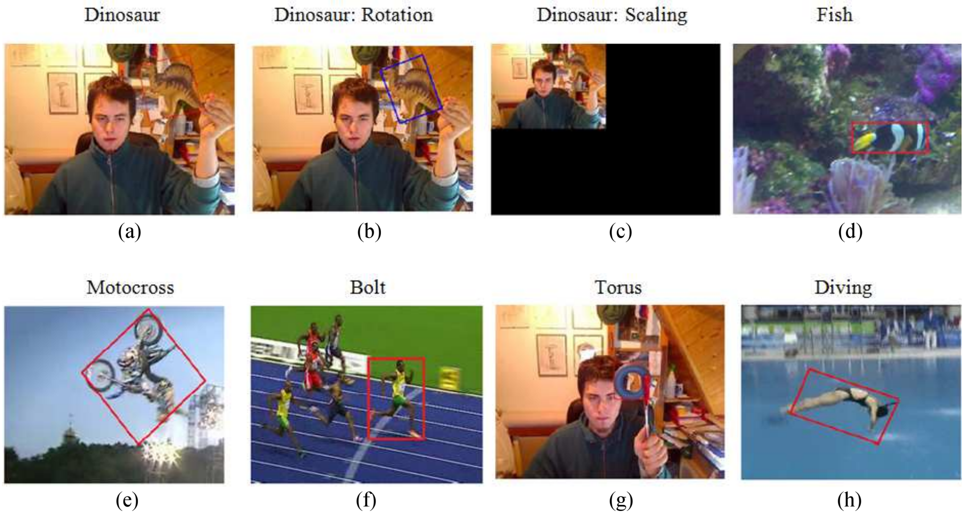
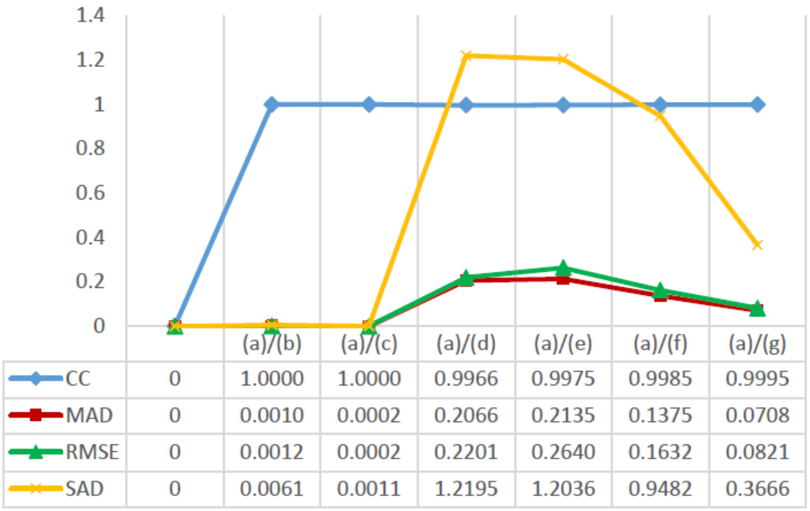


Fig. 6. Samples of video sequence selected from VOT database

```
Algorithm: Estimate the size of search area in the next video frame
Input:  Next frame ← Fr(n+1) = (widthFr, heightFr),
        ROI of target in the current video frame ← ROI (n) = (widthROI, heightROI, centroidROI)
Output: Search area size in the next frame ← SA (n+1) = (widthSA, heightSA)
Compute the location of target in next frame by using centroidROI ← (x, y)
if (widthROI x 2) > widthFr then
        widthSA = widthFr
else
        widthSA = widthROI x 2
end if
if (heightROI x 2) > heightFr then
        heightSA = heightFr
else
        heightSA = heightROI x 2
end if
```

Fig. 7. Pseudo code for searching the most likely location of target in the following video frame



		(a)/(b)	(a)/(c)	(a)/(d)	(a)/(e)	(a)/(f)	(a)/(g)
CC	0	1.0000	1.0000	0.9966	0.9975	0.9985	0.9995
MAD	0	0.0010	0.0002	0.2066	0.2135	0.1375	0.0708
RMSE	0	0.0012	0.0002	0.2201	0.2640	0.1632	0.0821
SAD	0	0.0061	0.0011	1.2195	1.2036	0.9482	0.3666

Fig. 8. Matching criteria of descriptors that are computed from Fig. 6, where Correlation Coefficient (CC), Mean Absolute Difference (MAD), Root Mean Squared Error (RMSE) and Sum of Absolute Difference (SAD)

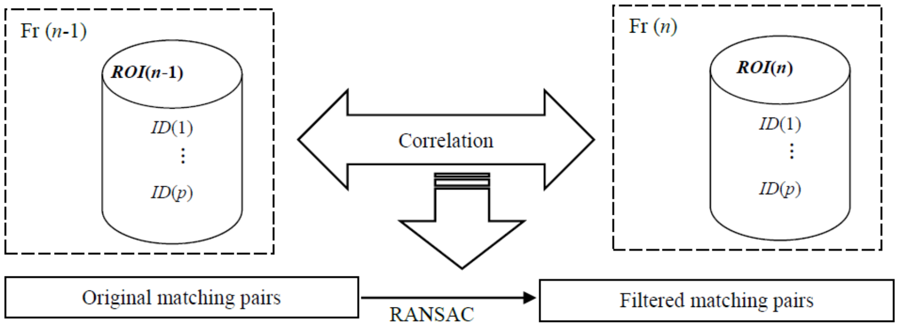


Fig. 9. Methodology to match the descriptors from two consecutive frames

Table 1. Examples of invariant descriptor computed for single-target tracking that is depicted in Fig. 6

Video sequence	Descriptor 1	Descriptor 2	Descriptor 3	Descriptor 4	Descriptor 5
(a) Dinosaur	2.83763718	6.43220496	10.005507850	9.89971412	6.360887056
(b) Dinosaur: Rotation	2.83754509	6.42870862	10.005079500	9.90128054	6.361408775
(c) Dinosaur: Scaling	2.83757730	6.43206610	10.006114530	9.89991911	6.360771349
Variance (δ/μ)	0.00001600	0.00030800	0.000052000	0.00008600	0.000053000
(d) Fish	2.74647456	6.01559222	9.417894528	9.91291528	6.250012111
(e) Motocross	2.86174249	6.34979751	10.280109300	10.64044220	6.442691427
(f) Bolt	2.68944380	5.95541496	9.958119233	9.87003711	6.114742046
(g) Torus	2.84872441	6.53332944	10.228156380	9.87506709	6.368011042
(h) Diving	2.83493682	6.21989446	9.566294378	9.52572635	6.415703213
Average of differences with video a (+/-)	0.05545000	0.25785000	0.314290000	0.23645000	0.100150000

Table 2. The basic characteristics of video sequences

Sequence	Frames	Resolution	Type	Target size (pixel2)	Size change	Target size w.r.t. Frame (%)	Features
Dinosaur	326	320×240	Rigid	Medium (8,448)	High	6.1-11.7	75-139
Fish	164	460×259	Rigid	Small (3,574)	Medium	1.5-3.9	12-23
Motocross	326	640×360	Rigid	Large (15,922)	High	2.3-8.6	109-345
Bolt	350	640×360	Articulated	Small (3,456)	Low	0.4-2	26-54
Torus	264	320×240	Rigid	Small (2,304)	Low	2.6-3.2	14-28
Diving	219	400×224	Articulated	Medium (8,960)	High	6.7-13	45-244

Performance Measures

Since a large variety of performance criterion are used in the existing research works of visual tracking, it is difficult to decide a standard criterion to show the comparative performance between the visual trackers. Čehovin *et al*. (2016) have made an effort to investigate the theoretical aspect of the existing criteria and have proven the correlation between all criteria with some systematic experimental analysis. Two most suitable criteria in terms of accuracy and robustness are identified as the guide to symbolize the trackers performance. Both criteria are able to measure the ability of a visual tracker to continuously and accurately locate the target throughout the video sequence. The overlap region criterion, *OR*(*n*) compares both position and size of the ground truth *ROI*, *R*(*gt*, *n*) and the estimated *ROI*, *R*(*est*, *n*) to observe the percentage of a successful tracking in a particular frame *n*. To conclude an overall overlap region over an entire video sequence, the average overlap region is deduced in Equation 7:

$$\overline{OR} = \sum_{k=1}^{TF} \frac{OR(i)}{TF}, where \qquad (7)$$
$$OR(n) = \frac{R(gt,n) \cap R(est,n)}{R(gt,n) \cup R(est,n)}, n = 1,...,TF$$

A visual tracker that scored a high value in accuracy is not necessarily showing a continuous tracking in every frame. Therefore, the number of tracking failures is recorded to annotate the robustness of a visual tracker. Note that the indicator of failure rate, *F* is increased whenever the overlap region, *OR* = 0. In order to have a better visualization of both criteria in the comparative performance, the failure rate is further normalized as an exponential representation in Equation 8. The value of *S* serves as a scaling factor to control the visualization of robustness criterion, *Rs*. When there is none tracking failures, the value of robustness criterion should increase to a maximum of value 1:

$$R_s = e^{-S\mu}, where\ \mu = \frac{F}{TF} \qquad (8)$$

In the common practice to setup an experiment for testing visual tracker, the *ROI* of target is initialized in the first frame and the tracker will locate the target in the subsequent frames. The performance is continuously measured even though the tracker failed in any time frame of the entire video sequence. This practice has been used by some publications to show comparative study in visual tracking (Ross *et al*., 2008; Kwon and Lee, 2009; Babenko *et al*., 2011). Other than that, some publications suggested that reinitialization has to be triggered once the tracker fails. Reinitialization represents the manual intervention in some supervised system that do not require autonomous real-time performance. If none of the region is overlap, the tracker is considered fail and the tracking process is terminated. The tracking failures that happened, whether at the beginning or in the middle of video sequence would affects the performance evaluation of the respective tracker. Thus, it is inappropriate to calculate the sum of overlap region with respect to total number of frames. The tracking should be continued until fully processed the entire sequence to show the overall evaluation of the performance (Čehovin *et al*., 2016; Kristan *et al*., 2016). Both practices are implemented in our experimental study to show the comparative performance among trackers. At the same time, we can further investigate whether the tracker improve its accuracy and robustness performances after reinitialization.

According to Kristan *et al.* (2016), it is difficult to point out a ‘best’ visual tracker that is able to accommodate a large variety of surveillance dataset with diversified characteristics. Some applications require a highly accurate tracking over robustness, such as sport analytics that is used to locate the position of ball during tournament or calculate the player’s acceleration and velocity. Yet, a robust tracker is more preferable than a highly accurate tracker for other applications that required a continuous tracking performance. The number of failure rate in the continuous tracking and accuracy have to be observed together to further explained a tracker’s performance. Apart from showing the applicability of our proposed model, the experimental results are also analyzed to find out the suitability of the proposed model towards different characteristics of dataset.

Two types of experiments are conducted on each tracker by using six video sequences from VOT dataset. The first type of experiment measured the tracking process, where the tracker was initialized in the first frame and the tracker has to locate the target in the subsequent frames until the end of the video sequence. Although reinitialization is not triggered, the number of failures of each tracker, *F* is also recorded. On the other hand, the second experiment would reinitialize the tracker whenever the tracking failed. The performance of the proposed and two state-of-art visual trackers are illustrated in Fig. 10. For a better understanding on the tracker’s performance in the accuracy-robustness visualization graph, kindly take notes on the following remarks:

- A zero-failure but totally inaccurate tracker will be displayed at the bottom right corner of the graph
- When the area of the estimated *ROI* overlapped with mostly area of the ground truth but the tracker experienced extremely high failure rate, it will be located towards the top left corner of the graph

For brevity, the proposed invariant feature descriptor is represented as IFD in the rest of the experimental results and discussions.

Results and Discussion

Among six of the video sequences, both of the proposed IFD (with and without reinitialization) achieved the highest average of overall accuracy and robustness, as illustrated in Fig. 11 but not all of the IFD is placed at the top right corner of the graph in each sequence of Fig. 10 since the accuracy is notably lower in the fish and torus sequences. Among four of the sequences (dinosaur, motocross, bolt and diving), both IFD have outperformed other trackers no matter in terms of accuracy or robustness as shown in Fig. 10. In the aspect of accuracy, both of the IFD achieved more than 0.6 till 0.8 in these sequences with the highest accuracy in dinosaur, motocross and followed by bolt and diving. However, only IFD with reinitialization always exceeds 0.8 in terms of robustness while maintaining at 0.7 to 0.9 accuracy. Tracker IFD without reinitialization always fall behind since the IFD tracker has included its false-tracking measurements although it has drifted away from the target or accidentally drifted back to the ground truth position at some other frames. For the fish and torus sequences, IFD with reinitialization also performed better in terms of accuracy and robustness even though the accuracy is notably lower than other trackers. Therefore, reinitialization is helpful to compensate the tracker’s performance in some challenging situations of the video sequence.

A few of the visual properties, such as rigidity, illumination, foreground and background colors are the conditions that can be analyzed to interpret the tracker’s performance. Although dinosaur, motocross and diving sequences experienced high changes in size with the changing background and motion but both IFD are able to perform superior results in terms of accuracy and robustness. Mainly because three of the sequences contributed more features as compared to the fish and torus sequences. The number of detected features are recorded in the last column of Table 2, where fish and torus sequences are having less than 20 features in mostly frames. Feature matching process is able to filter a set of more reliable matching pairs when more features are involved. This result has proven that IFD depends strongly on the number of detected features rather than size changes and rigidity of the target.

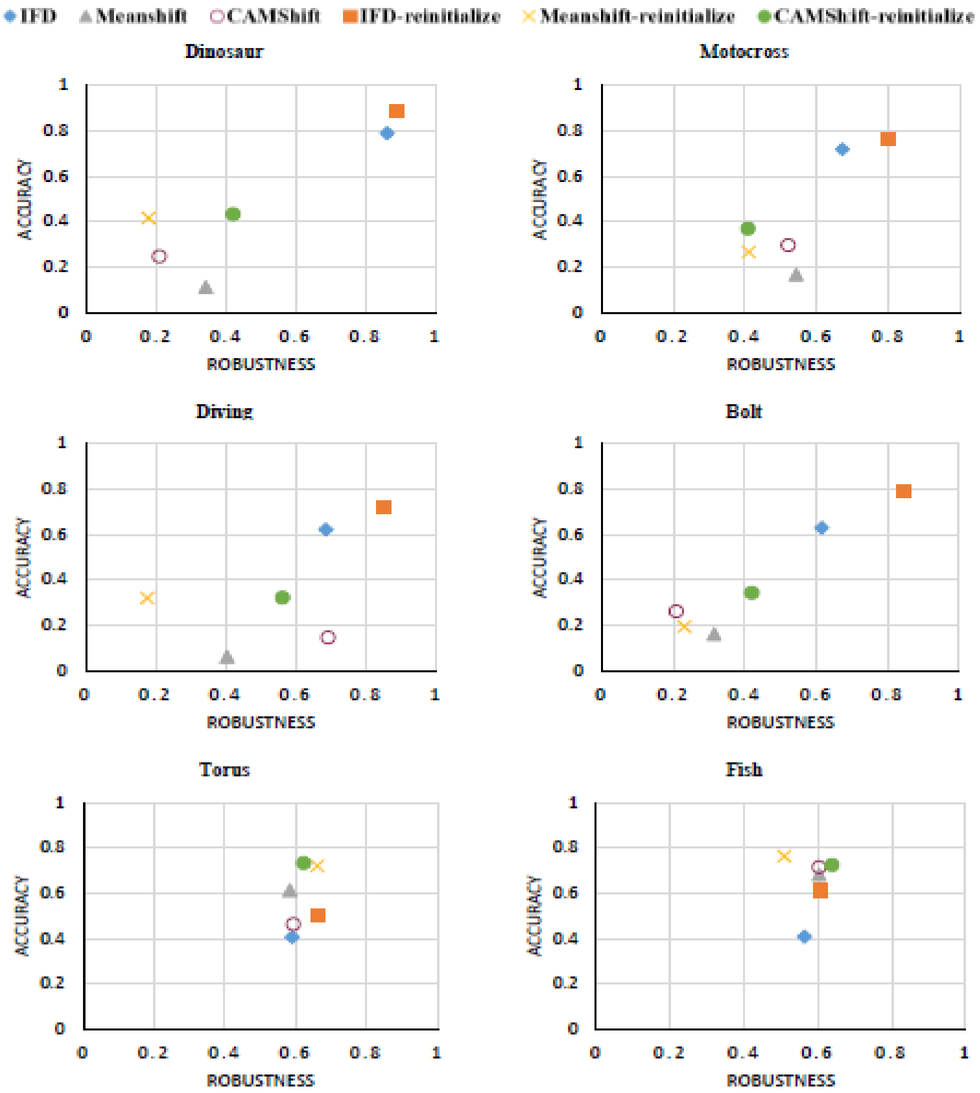


Fig. 10. Comparative performance for proposed and existing visual trackers over all video sequences

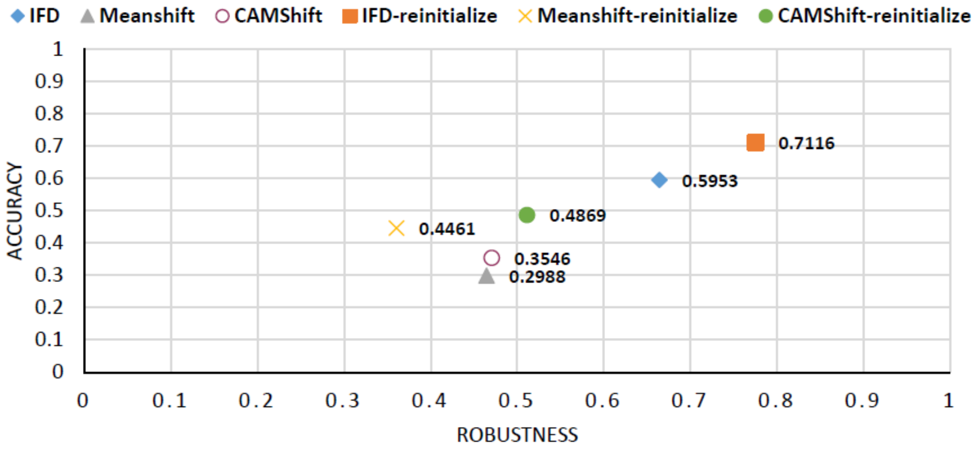


Fig. 11. Average performance of the overall accuracy and robustness between proposed and existing visual trackers

Unlike IFD tracker, the accuracy of meanshift and Camshift trackers are apparently low in the four aforementioned sequences but both trackers surpassed in the fish and torus sequences. Note that meanshift and Camshift trackers are sensitive to color ambiguity, illumination and non-rigid object (Lim and Kang, 2011). Since the foreground area of torus and fish are constituted from the specific colors that are almost similar with the background area, making both trackers that rely heavily on color to perform better than IFD. However, the color ambiguity between foreground and background colors in dinosaur sequence has increased the vulnerability of meanshift and Camshift trackers to drift away from the rigid target. The accuracy of both trackers also dropped in the motocross sequence because of the high illumination changes in the beginning to middle of sequence and the size changes in the rest of the sequence. Although diving sequence is having less illumination and color ambiguity problems, the background clutter and large shape deformation due to the articulated body parts have mixed up the specific color information used by both trackers. However, IFD tracker does not suffer from these problems and accomplish more than 0.6 accuracy in the dinosaur, motocross, bolt and diving sequences. The advantages of IFD are assured for processing an object that exhibits highly-distinctive features regardless of size, shape and illumination changes.

Conclusion

A new invariant feature descriptor model using moment invariants has been presented and compared with the state-of-the-art trackers on different characteristics of video sequences. A more detailed description of the proposed model has been described thoroughly in this study. The model begins with a selected region of interest that contains a single target. Feature points are detected from the *ROI*, where a set of descriptor is formulating around the region of each feature point. The set of descriptor extracted from *ROI* of two consecutive frames are matched and used to find out the most likely position of the target in the next frame.

From the large number of diversified scenarios in the surveillance applications, the performance and analysis of the proposed model has been demonstrated with single-target tracking. The detailed experimental analysis and comparison are conducted with six benchmarked video sequences. The experimental results shown that our proposed model on average outperforms the existing visual trackers by achieving higher accuracy and robustness. In addition, the results also revealed a few important aspects of the proposed model. Our proposed model is able to overcome the problems faced by meanshift and Camshift trackers, such as color ambiguity, illumination and non-rigidity. In addition, the proposed model is invariant regardless of object's size, rotation and translation changes throughout video sequence. Nevertheless, the experimental analysis also found out that the proposed model depends strongly on the number of detected features. Our model would fail if apply on the target that is particularly contained features that are not obvious and undetectable, such as a smooth surface with a single color or a tiny foreground area with less texture.

In our future research, the proposed model will be improved to manage the tracking task on the target with small and smooth surface. The video sequences will be selected from other add-on datasets of the future VOT challenges. Likewise, it would be interesting to conduct a comparative evaluation against a variety of existing trackers from the VOT challenges.

Acknowledgement

This research work is supported by the Fundamental Research Grant Scheme from Ministry of Education Malaysia (MMUE/140027).

Author's Contributions

Lee-Yeng Ong: Involving in carried out algorithm study, framework development, testing, compilation and manuscript writing.

Siong-Hoe Lau: Supervised the overall research and provided critical reviewing on the manuscript for significant intellectual content.

Voon-Chet Koo: Supervised the framework design, data analysis and provided critical reviewing on the manuscript for significant intellectual content.

Ethics

This article is original and contains unpublished material. The corresponding author confirms that all of the other authors have read and approved the manuscript and no ethical issues involved.

References

Adam, A., E. Rivlin and I. Shimshoni, 2006. Robust fragments-based tracking using the integral histogram. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Jun. 17-22, IEEE Xplore Press, New York, pp: 798-805. DOI: 10.1109/CVPR.2006.256

Almoosa, N.I., S.H. Bae and B.H. Juang, 2008. Toward robust moment invariants for image registration. Proceedings of IEEE IEEE International Conference on Acoustics, Speech and Signal Processing, Mar. 31-Apri. 4, IEEE Xplore Press, Las Vegas, pp: 1009-1012. DOI: 10.1109/ICASSP.2008.4517783

Artner, N.M., 2008. A comparison of mean shift tracking methods. Proceedings of the 12th Central European Seminar on Computer Graphics, (SCG' 08), pp: 197-203. DOI: 10.1.1.309.5140

Babenko, B., M. Yang and S. Belongie, 2011. Robust Object Tracking with Online Multiple Instance Learning. IEEE Trans Pattern Anal. Mach. Intell., 33: 1619-1632. DOI: 10.1109/TPAMI.2010.226

Baltzakis, H., M. Pateraki and P. Trahanias, 2012. Visual tracking of hands, faces and facial features of multiple persons. Mach. Vision Applic., 23: 1141-1157. DOI: 10.1007/s00138-012-0409-5

Biederman, I., 1987. Recognition-by-components: A theory of human image understanding. Psychol. Rev., 94: 115-147. DOI: 10.1037/0033-295X.94.2.115

Bradski, G.R., 1998. Computer vision face tracking for use in a perceptual user interface. Intel. Technol. J. DOI: 10.1.1.14.7673

Čehovin, L., A. Leonardis and M. Kristan, 2016. Visual object tracking performance measures revisited. IEEE Trans. Image Processing, 25: 1261-1274. DOI: 10.1109/TIP.2016.2520370

Chen, R., S. Wang, L. Gong and C. Liu, 2013. Hand gesture recognition for human-computer interaction using moment invariants and neural network classifier. Lecture Notes Comput. Sci., 8102: 661-667. DOI: 10.1007/978-3-642-40852-6_66

Costantini, L., L. Seidenari, G. Serra, L. Capodiferro and A. Del Bimbo, 2011. Space-time Zernike moments and pyramid kernel descriptors for action classification. Proceedings of the International Conference on Image Analysis and Processing Image Analysis and Processing, (IAP 11), Springer Berlin Heidelberg, pp: 199-208. DOI: 10.1007/978-3-642-24088-1_21

Dhome, M., 2009. Visual Perception Through Video Imagery. 1st Edn., John Wiley and Sons. ISBN-10: 1848210167.

Diop, M., C.H. Lim, T.S. Lim and L.Y. Ong, 2016. Vision-based real-time positioning and autonomous navigation system using mobile robot in indoor environments. Am. J. Applied Sci. DOI: 10.3844/ajassp.2016.593.608

Friedman, N. and S. Russell, 1997. Image segmentation in video sequences: A probabilistic approach. Proceedings of the 13th Conference Uncertainty in Artificial Intelligence, Aug. 01-03, Providence, pp: 175-181.

Fukunaga, K. and L. Hostetler, 1975. The estimation of the gradient of a density function, with applications in pattern recognition. IEEE Trans. Inform. Theory, 21: 32-40. DOI: 10.1109/TIT.1975.1055330

Harris, C. and M., Stephens, 1988. A combined corner and edge detector. Proc. Alvey Vision Conf., 15: 50-50.

Hartley, R. and A. Zisserman, 2004. Multiple View Geometry in Computer Vision. 1st Edn., Cambridge University.

Hoseinnezhad, R., B.N. Vo and B.T. Vo, 2013. Visual tracking in background subtracted image sequences via multi-Bernoulli filtering. IEEE Trans. Signal Processing, 61: 392-397. DOI: 10.1109/TSP.2012.2222389

Hu, M. 1962. Visual pattern recognition by moment invariants. IRE Transactions Information Theory, 8: 179-187. DOI: 10.1109/TIT.1962.1057692

Izadinia, H., I. Saleemi, W. Li and M. Shah, 2012. MT^{2}T: Multiple people multiple parts tracker. Proceedings of the European Conference on Computer Vision, (CCV' 12), pp: 100-114. DOI: 10.1007/978-3-642-33783-3_8

Jian, X., C. Xiaoyuan, S. Xiaoping and G. Wen, 2015. An improved Harris-FAST algorithm for underwater object corner detection. Proceedings of the 27th Chinese Control and Decision Conference, May 23-25, IEEE Xplore Press, Qingdao, pp: 5424-5428. DOI: 10.1109/CCDC.2015.7161763

Pong, K.T. and R. Bowden, 2001. An improved adaptive background mixture model for real-time tracking with shadow detection. Proceedings of the 2nd European Workshop on Advanced Video Based Surveillance Systems, (BSS' 01), Springer US, pp: 135-144. DOI: 10.1007/978-1-4615-0913-4_11

Kristan, M., J. Matas, A. Leonardis, T. Vojíř and R. Pflugfelder *et al*., 2016. A novel performance evaluation methodology for single-target trackers. IEEE Trans. Pattern Anal. Mach. Intelli., 38: 2137-2155. DOI: 10.1109/TPAMI.2016.2516982

Kumar, S. and J.S. Yadav, 2016. Video object extraction and its tracking using background subtraction in complex environments. Perspectives Sci., 8: 317-322. DOI: 10.1016/j.pisc.2016.04.064

Kwon, J.S. and K.M. Lee, 2009. Tracking of a non-rigid object via patch-based dynamic appearance modeling and adaptive basin hopping monte carlo sampling. Proceedings of the IEEE Conference of Computer Vision and Pattern Recognition, (VPR' 09), pp: 1208-1215. DOI: 10.1109/CVPRW.2009.5206502

Lee, D.S., 2005. Effective Gaussian mixture learning for video background subtraction. IEEE Trans. Pattern Anal. Mach. Intelligence, 27: 827-832. DOI: 10.1109/TPAMI.2005.102

Li, C., J. Li, B. Fu and X. Yang, 2012. Fingerprint verification based on DFB and Hu invariant moments. J. Computat. Inform. Syst., 4: 1407-1414.

Lim, H.Y. and D.S. Kang, 2011. Object tracking system using a VSW algorithm based on color and point features. EURASIP J. Adv. Signal Processing, 2011: 60-60. DOI: 10.1186/1687-6180-2011-60

Linares-Sánchez, L.J., J.L. Fernández-Alemán, G. García-Mateos, Á. Pérez-Ruzafa and F.J. Sánchez-Vázquez, 2015. Follow-me: A new start-and-stop method for visual animal tracking in biology research. Proceedings of the 37th Annual International Conference of the Engineering in Medicine and Biology Society, Aug. 25-29, IEEE Xplore Press, Milan, pp: 755-758. DOI: 10.1109/EMBC.2015.7318472

Liu, T., G. Wang and Q. Yang, 2015. Real-time part-based visual tracking via adaptive correlation filters. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, (VPR' 15), pp: 4902-4912. DOI: 10.1109/CVPR.2015.7299124

Mei, X. and H. Ling, 2011. Robust visual tracking and vehicle classification via sparse representation. IEEE Trans. Pattern Anal. Mach. Intelli., 33: 2259-2272. DOI: 10.1109/TPAMI.2011.66

Miksik, O. and K. Mikolajczyk, 2012. Evaluation of local detectors and descriptors for fast feature matching. Proceedings of the International Conference on Pattern Recognition, (CPR' 12), pp: 2681-2684. DOI: 10.1.1.301.6783

Moravec, H.P., 1980. Obstacle avoidance and navigation in the real world by a seeing robot rover. PhD Thesis, Carnegie-Mellon University, Pennsylvania.

Mukundan, R. and K.R. Ramakrishnan, 1998. Moment Functions In Image Analysis: Theory and Applications. 1st Edn., World Scientific, Singapore.

Nejhum, S.S., J. Ho and M.H. Yang, 2008. Visual tracking with histograms and articulating blocks. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, (VPR' 08), pp: 1-8. DOI: 10.1109/CVPR.2008.4587575

Ong, L.Y., S.H. Lau and V.C. Koo, 2014. A new approach of local feature descriptors using moment invariants. J. Comput. Sci., 10: 2538-2547. DOI: 10.3844/jcssp.2014.2538.2547

Pnevmatikakis, A. and L. Polymenakos, 2006. Kalman tracking with target feedback on adaptive background learning. Proceedings of the International Workshop of Machine Learning for Multimodal Interaction, (LMI' 06), Springer Berlin Heidelberg, pp: 114-122. DOI: 10.1007/11965152_10

Rosenfeld, A. and E. Johnston, 1973. Angle detection on digital curves. IEEE Trans. Comput., 100: 875-878. DOI: 10.1109/TC.1973.5009188

Ross, D.A., J. Lim, R. Lin and M. Yang, 2008. Incremental learning for robust visual tracking. IEEE Int. J. Comput. Vision, 77: 125-141. DOI: 10.1007/s11263-007-0075-7

Rosten, E. and T. Drummond, 2006. Machine learning for high-speed corner detection. Proceedings of the European Conference on Computer Vision, (ECC' 06), pp: 430-443.

Santiago, C.B., A. Sousa, L.P. Reis and M.L. Estriga, 2011. Real time colour based player tracking in indoor sports. Proceedings of the Computational Vision and Medical Image Processing: Recent Trends, Computational Methods in Applied Sciences, (MAS' 11), Springer Netherlands, pp: 17-35. DOI: 10.1007/978-94-007-0011-6_2

Senst, T., B. Unger, I. Keller and T. Sikora, 2012. Performance evaluation of feature detection for optical flow tracking. Int. Conf. Pattern Recognition Applic. Methods, 2: 303-309.

Shi, J. and C. Tomasi, 1994. Good features to track. Proceedings of the IEEE Computer Society Conference Computer Vision and Pattern Recognition, (VRP' 94), pp: 593-600. DOI: 10.1.1.36.2669

Shvarts, D. and M. Tamre, 2012. Local and global descriptors for place recognition in robotics. Proceedings of the 8th International DAAAM Baltic Conference, Apr. 19-21, Tallinn, Estonia, pp: 1-6.

Smith, S. M. and J.M. Brady, 1997. SUSAN-a new approach to low level image processing. Int. J. Comput. Vision, 23: 45-78. DOI: 10.1023/A:1007963824710

Tuytelaars, T. and K. Mikolajczyk, 2008. Local invariant feature detectors: A survey. J. Foundat. Trends Comput. Graph. Vision, 3: 177-280. DOI: 10.1561/0600000017

VOT, 2016. Visual object tracking.