Simple Online and Real-time Tracking with Feature Matching Enhancement for Re-identification after Occlusion

Koksal Chou, Natsuda Kaothanthong, Chawalit Jeenanunta

School of Management Technology, Sirindhorn International Institute of Technology,
Thammasat University, Pathumthani, Thailand
E-mail: m6022040353@g.siit.tu.ac.th, natsuda@siit.tu.ac.th, chawalit@siit.tu.ac.th

Received: May 13, 2019/ Revised: August 6, 2019 / Accepted: August 20, 2019

***Abstract*—Occlusion in people tracking in computer vision is the problem that affects the tracking continuity as the information of the object is lost when the tracked object is behind another object. This study proposes a tracking algorithm that is robust to occlusion. The algorithm is designed based on the algorithm called Simple Online and Real-time Tracking (SORT) which utilizes deep neural network detector, Kalman Filter, and Hungarian algorithm. The feature extraction method is used to capture the information of objects before and after occlusion to solve the problem of multi-object tracking in the presence of occlusion. This method can improve the lack of memory bottleneck of SORT which is a crucial requirement for robust multi-object tracking. The experiment is performed on 13 testing videos which contain multi-people walking pass each other with and without background noise. The result from the experiment shows that the proposed method can increase multi-object tracking accuracy of SORT. The algorithm can correctly re-identify the object after the occlusion event.**

***Index Terms*—Convolutional neural network, feature extraction, multiple object tracking.**

I. Introduction

Multi-object tracking is an application in computer vision that is used to track multiple moving objects in the video file or real-time video stream. It gives an ability for a computer to understand the context of the image by using an algorithm to process and analyze each image stream. The algorithm should be designed to keep tracking the identity and location of the same object throughout the duration that the object appears in a different frame within a video. At first, the algorithm must be able to detect the objects, their types, and locations when they appear in a certain frame in the video. Then the algorithm must be able to detect, associate, and differentiate objects in the following frames. For natural video, there exist noises in several forms that make tracking multiple objecnnts difficult. The problem such as occlusion makes it hard to differentiate multiple objects. In order to design a state-of-the-art algorithm, researchers must take these noises into consideration. Multi-object tracking can be used in various applications such as surveillance, robotic, industrial, self-driving car, shopping, etc.

The main purpose of this study is to develop a method that can effectively keep track of multiple objects in real time in the presence of partial and total occlusion.

This paper is organized as follow: Review on previous works and related literature are presented in section 2. Section 3 describes the proposed method, experiment setting, and key performance indicator. The result and comparison are demonstrated in section 4. Finally, section 5 discusses the conclusion of outcomes and future work.

II. Literature Review

A. Object Comparison and Detection

Multiple objects tracking is a two-stage problem. The first stage is the detection problem which tries to locate the coordinate of the object within one frame. The second stage is the tracking problem that is required to associate the same object between frames.

The early form of object detection is template matching. It compares the feature of source image, and the template image. In Image Correlation Matching, each pixel value is the feature of both source and template image [1]. Cross-correlation is used to calculate the image correlation which is the sum of pairwise multiplication for each corresponding pixel of the two images. Both images are supposed to have the same dimension. However, this method is not robust in the presence of changing in the global brightness of the images being compared [2]. Extended from Correlation Matching, S.-q. Chen [3] used a more advanced template matching algorithm. This algorithm used classic pyramid search by transform template image into different angles and different scales and then thoroughly compared to the source image. This algorithm returns both the position and orientation of the matched template. Similar approach enhanced the performance of Grayscale-based matching by limiting computation to the object edge areas. It is based on the assumption that the shape of an object is defined by the shape of its edge. This technique can, therefore, reduce some unnecessary computation.

Another method used for detection by comparing the object feature is called local feature extraction. It extracted key-point from an image in the pixel level. Key-points from the template are used to compare with key-point from different sources to find the matched pair of images. The well-known key-point descriptors are Scale Invariant Feature Transform (SHIF) [4] and Speeded-up Robust Features (SURF) [5]. However, both algorithms are patented. Oriented Features from Accelerated and Segments Test and Rotated Binary Robust Independent Elementary Features (ORB) is proposed by E. Rublee et al. [6] as an alternative to SHIF and SURF with superior performance. Viswanathan [7] use FAST (Features from Accelerated and Segments Test) to compare the brightness of a certain pixel in the picture to another 16 surrounding pixels. Then those 16 pixels are divided into three groups: brighter than, darker than or similar to the pixel in the middle. If more than half of those surrounding pixels are brighter or darker than the middle pixels, then that middle pixel is selected as key-point. Then Calonder, Lepetit, Strecha and Fua [8] used BRIEF (Binary Robust Independent Elementary Features) to convert all the key-point, found by FAST into a binary feature vector. Together, FAST and BRIEF can represent an object. BRIEF descriptor only contains 1 and 0. Each key-point is described by a feature vector of 256 bits string. By using a multiscale image pyramid over FAST and Rotation with BRIEF, ORB can archive scale and rotation invariant.

Convolutional Neural Network (CNN) is a Deep Learning algorithm designed to process image file, assign weights and biases to various features in the image, and then differentiate one image from another. For the learning process, the convolutional neural network observes the pattern of the group of pixels from the input image. Then find the common feature among the set of images from the same category. In the next step, CNN used the newly found feature to compare with a set of test images in the same and different categories. It then updated weight and biases based on the result of the comparison. This process happened for hundreds or thousands of iterations until the test accuracy is satisfied. After the final stage of learning, CNN can classify an image into different categories. This neural network can perform image matching better than traditional image matching such as template matching. There is one drawback that limits its performance to use as object detection. CNN can only tell if there is an object of interest inside the image, but it could not tell the specific location of that object.

To add the localization ability to CNN, Ren, He, Girshick and Sun [9] proposed the algorithm called Faster R-CNN. The algorithm composed of two modules, region proposal module and Fast R-CNN [10]. R-CNN used a selective search algorithm to identify the region proposal in order to detect the feature. This method can find the object of interest and the location in which it resides. Though, searching and comparing through all the sub-region inside the image would be too slow and time-consuming which will affect the performance of the detection network. Instead, a separate deep neural network is used to predict the region proposal. By applying detector inside a new proposed region, Faster R-CNN can outperform its predecessors and work in real-time at speed up to 5 frames per second.

B. Video Tracking

In the second stage of multi-object tracking, Video tracking is an assignment problem that is used to assign moving objects with unique IDs.

D. Comaniciu et al. [11] proposed a method to use a mean shift for tracking objects. Mean shift has been used to solve the clustering problem. In computer vision, Mean shift used image feature as the probability density function. Color histograms of the object in the first image are used to create a confident map of the next image. Then mean shift found the mean of a confident map around the old position of the object. Calculating iteration over iteration until it found the peak of a confident map. Mean shift gave the probability of the pixel color that occurs in the object in the previous image to each pixel of a new image. It outputted the new location that the object resides in the new video frame. This algorithm failed when two similar objects make a full occlusion or object is moving too fast.

Chirag I. Patel et al. [12] proposed another method that uses object contour for tracking. The algorithm uses centroid coordinates of the contour to define the state of the object along with its velocity and acceleration. Then the motion model is used to predict the next state of the object. This method is computationally inexpensive and can even track an object in the presence of partial occlusion. The limitation of this approach is that the color of the object must be different from the background.

Bewley, Ge, Ott, Ramos and Upcroft [13] proposed an algorithm called Simple Online and Real-time

Tracking (SORT) to solve a multi-object tracking problem. This algorithm extended the advantage of the efficiency of Deep Neural Network. Detection from CNN is represented as a state of the object. Then Bewley, Ge, Ott, Ramos and Upcroft [13] used SORT to apply linear motion model to the object state such that they can predict its new state in the next frame. This method can keep track of individual person accurately. Although this algorithm is robust with a distinguishable appearance of people, when there is full occlusion between multiple people, the algorithm tended to fail by switching ID or assign a new ID to the object after the occlusion. It happened because SORT lack of memory represents the property of the object.

Addressing the weakness of Convolutional Neural Network, Sadeghian, Alahi and Savarese [14], Xiang, Zhang and Hou [15], Fang, Xiang, Li and Savarese [16] and Fan and Ling [17] apply Recurrent Neural Network (RNN) to assist tracking process. CNN network performed best for interclass classification problem. But in multi-people tracking, people belonged to the same class that shares similar features. These similarities acted as distractors to the network. Fan and Ling [17] introduced Structure-aware network for visual tracking (SANet), which used object's own structure information to differentiate it from distractors. RNN is utilized to represent object structure and integrate with CNN to distinguish similar feature of different people. The method trained features of the object in real time in different stages using RNN which give the memory to the network. In the presence of noise such as full occlusion, SANet used learned features to re-identify the object. Benchmark result shows that this algorithm is robust to full occlusion and can effectively keep track of multiple people. However, RNN is computational expensive which make online training very slow. The whole algorithm can run less than 1 frame per second. This fact prevented the usage of SANet and another RNN based network in real-time object tracking.

III. Methodology

The objective of this study is to propose the Multi-object tracking algorithm that is robust to full occlusion. The study extends the method of SORT by incorporating the feature extraction algorithm to give instant memories of objects appearances right before the occlusion happen. Those memories are then used to associate the IDs of lost tracked objects after occlusion to the existing objects. This method is called Simple Online and Real-time Tracking with Feature Matching Enhancement (SORT-FME). SORT-FME is developed based on three main steps. The first step is detection, the second step is tracking and the third step where this study mainly focuses on is feature matching enhancement. Fig. 1 shows all the steps for SORT-FME.

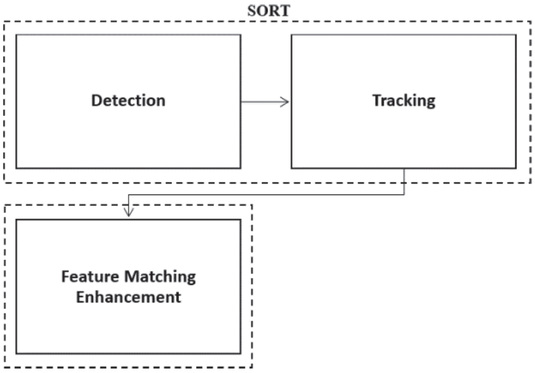


Fig. 1. SORT-FME Framework.

A. Detection

For the detection problem, as utilized by SORT, Faster Region CNN (FrRCNN) detection framework is used with default parameter. Output probability of more than 50% is selected to determine if a certain person is detected. Video file (or real-time video stream) is input through a detection network. The output from this detection is a set of coordinates of the bounding box that person resides in and its confident probability. It is in the form of [startX, startY, endX, endY, confident] where startX, endX are the range of coordinate in the vertical axis and startY, endY are the range of coordinate in the horizontal axis.

B. Tracking

Simple Online and Real-time Tracking is select as tracking algorithm framework. This algorithm augments the four coordinates of the rectangle of the detected object to model a state of that object. The state is represented as 7 components in each video frame for each object as below:

$$stateX = \left[x,\ y, s, r,\ \dot{x}, \dot{y}, \dot{s}\right]^T \tag{1}$$

Where in frame *T*, *x* and *y* are the vertical and horizontal location of the center of the box. *s* is the area of the box and *r* is aspect ratio which is constant. $\dot{x}, \dot{y}, \dot{s}$ are velocity component. Kalman filter is used in SORT to estimate new object's state and data association method to match the estimated box to the detection box [18]. Parameters from SORT are set as their default with minimum detect before it registers new ID to detected object to 3 consecutive frames and maximum age of 1 frame of lost detection before SORT terminate tracking of old IDs. SORT will then output new coordinates of the objects to create bounding boxes around objects in the new frame along with their IDs.

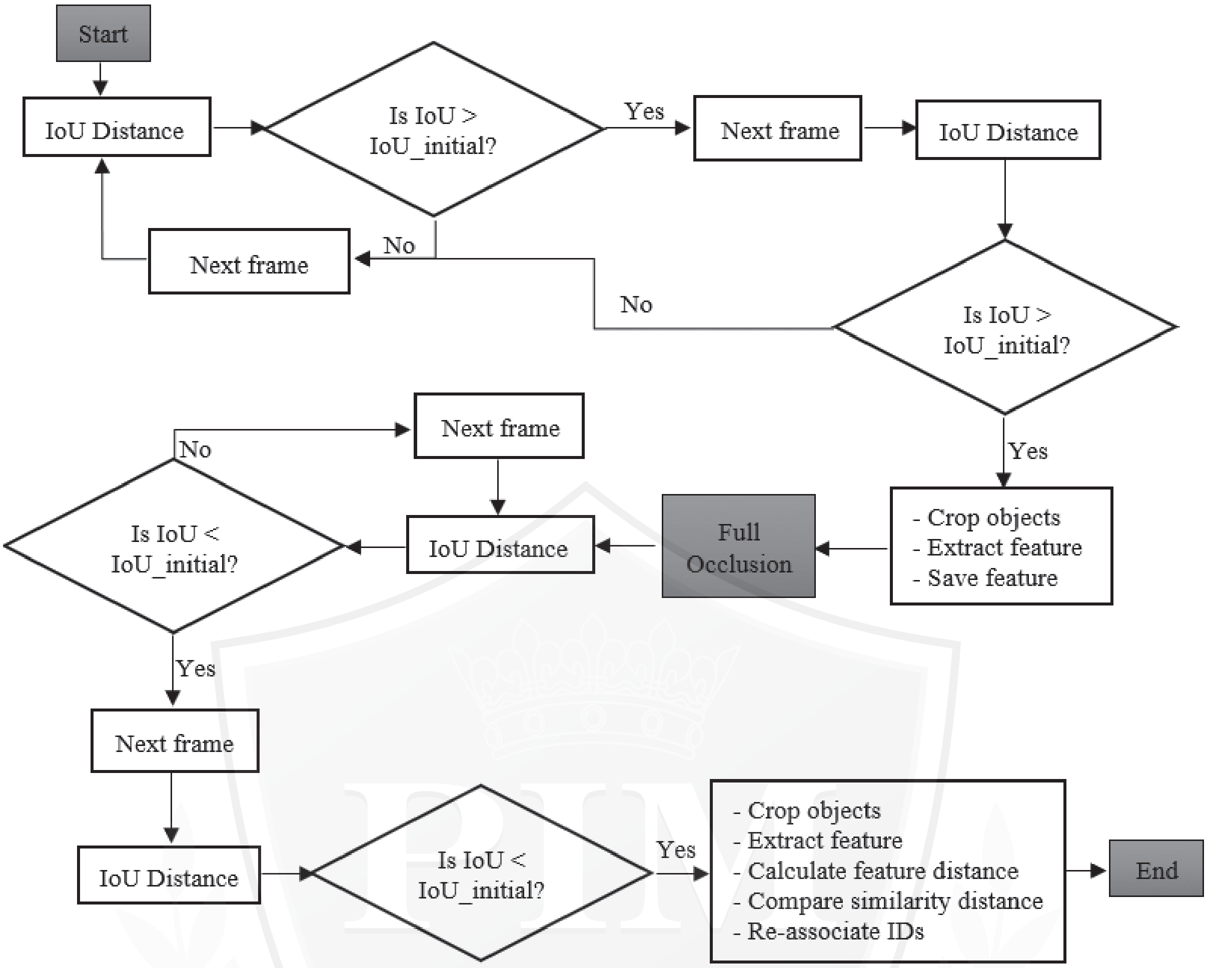


Fig. 2. Feature Matching Enhancement flow chart.

C. Feature Matching Enhancement

For the third step, SORT-FME is proposed to re-identify the object after occlusion. In order to solve the total occlusion problem, the algorithm must be able to remember the structure of the objects before two or more objects occlude with each other. The structure of the objects must be captured right at the start of occlusion. If the structure is captured after a large part of the objects has occluded, feature information can be lost by the occluded part. If it is captured long before occlusion happens, there is more chance that occlusion will not happen, or the features will be more different from those of object after occlusion. To detect the instance for the beginning of occlusion, intersection-over-union (IoU) distance is used between bounding box of the object.

$$IoU = \frac{Area\,of\ \ Overlap}{Area\ \ of\ \ Union} \tag{2}$$

Given the bounding box coordinate of object1 and object2:

object1= [startX1, startY1, endX1, endY1]
object2= [startX2, startY2, endX2, endY2]

Area of overlap and union are calculated by:

$$startx = \max\left(startX1,\ startX2\right)$$
$$starty = \max\left(startY1,\ startY2\right)$$
$$endx = \min\left(endX1,\ endX2\right)$$
$$endy = \min\left(endY1,\ endY2\right)$$
$$Area\ Overlap = \max\left(0, endx - startx\right)* \max\left(0, endy - starty\right) \tag{3}$$
$$Area\ of\ Union = Area\ object1 + Area\ object2 - Area\ overlap \tag{4}$$

Where *startx, starty, endx, endy* are the rectangle coordinates of the overlapped section.

Based on the experiment, the optimal *IoU* distance is set to a maximum of 0.15 for each box to contain full enough shape of the object to extract the features. At the frame where *IoU* is more than 0, algorithm triggers waiting moment. In the next frame if *IoU* increases, which mean that both objects converge, the bounding box of each object are cropped out from the picture. After that, feature extraction algorithm called Oriented Features from Accelerated and Segments Test and Rotated Binary Robust Independent Elementary Features (ORB) is used to extract the features from cropped images and save to memory with their respective ID. This algorithm uses a brightness at a pixel level to detect a key-point of the image. A group of key-points is representing the feature of an object. Feature extracted by ORB are the vectors of binary descriptors with the size of 256 bits.

During occlusion, only detection network and motion model are used. Feature extraction will not be applied as there is an incomplete structure of object in interest. End of occlusion is trigger by *IoU* distance smaller than 0.15. The proposed algorithm waits for another frame to see if *IoU* getting smaller than the previous frame which signifies that both objects are diverging. When the condition is true, it captures news images within the bounding boxes. Then features are extracted from those pair as vectors of binary descriptors. The next step, the feature of each object after the occlusion will be compared with the feature of each object before the occlusion. Since the feature of each object is represented by binary descriptor, brute force matching with Normalized Hamming distance is used for comparison. Normalized Hamming distance is the sum of XOR operation between 2 binary vectors.

$$Norm_{Hamming} = \sum_{i=1}^{n} V(i)\ XOR\ U(i) \tag{5}$$

Where *V(i)* and *U(i)* are binary vectors.

After finding the distance between each vector of each pair, the distance can be summed and divided by the total matching vector to get the average distance. The pair that has the least distance is the most similar one and could possibly be the same person. Hungarian algorithm will be used to optimally solve the assignment problem. So new IDs will be generated based on the IDs of the object before the occlusion happens. These IDs will replace false IDs provide by SORT. Fig. 2 gives the process flow chart of the algorithm.

D. Experiment

This study focuses on keep tracking of multiple people in the indoor environment where there is no lighting change. The room is set up with no clutter. A single camera is fixed overhead under the ceiling.

The experiment is done on videos recording with Lenovo workstation P320, OS Ubuntu 16.04 LTS, CPU Core i7-7700T, GPU Quadro P600, RAM 8GB, 1080p Webcam with autofocus.

Table I

Videos Experiment set 1 and 2

Test	Number of people	Re-Occlusion	Reverse Direction	Clothes	Background Noise
1	2	No	No	Different	No
2	2	No	No	Same	No
3	2	No	No	Same	No
4	2	No	Yes	Same	No
5	2	No	Yes	Different	No
6	3	No	Yes	Different	No
7	2	No	No	Different	Yes
8	2	Yes	No	Different	Yes
9	2	Yes	No	Same	Yes
10	2	Yes	No	Same	Yes
11	2	No	Yes	Different	Yes
12	3	No	No	Different	Yes
13	3	No	No	Different	Yes

The experiment is set to evaluate the performance of the proposed model against the visual similarity of the test subject and total occlusion. As shown in table I, the first set of videos (test 1 to test 6) contain plain background, the second set of videos (test 7 to test 13) contain some background noise.

E. Key Performance Indicator

Evaluation of Multi-Object Tracking proposed by Smith, Gatica-Perez, Odobez and Ba [19] consists of 3 measurements: recall, precision, and coverage test. The object that will be tracked is denoted by ground truth object (*GT*) with index *j*. The output from the tracking algorithm will be referred to as a tracker (*e*) and are indexed by *i*. Both *GT* and *e* are described by bounding boxes coordinates with respective index.

For *i* and *j* as an index of tracker and ground truth object, recall is defined as:

$$Recall_{i,j} = \frac{\left|e_i \cap GT_j\right|}{\left|GT_j\right|} \tag{6}$$

This value measures how much ground truth object (*j*) is covered by the respective tracker (*i*). It is within the range between 0 (no overlap) and 1 (fully overlap).

Precision is expressed as:

$$Precision_{i,j} = \frac{\left|e_i \cap GT_j\right|}{\left|e_i\right|} \tag{7}$$

Precision measure the coverage of tracker (*i*) over ground truth object (*j*). It also has a value between 0 (no overlap) and 1 (fully overlap).

Coverage Test will tell if *GT* is being tracked or not. The object being track required a high value of precision and recall. F-measure is used to formulate the coverage test.

$$F = \frac{2 * Recall * Precision}{Recall + Precision} \tag{8}$$

Coverage threshold (C_t) can be set so that if $F > C_t$, the object is correctly tracked. These KPI are calculated for each picture frame. For video tracking in this experiment:

$$Recall = \frac{number\ of\ correctly\ tracked}{number\ of\ ground\ truth} \tag{9}$$

$$Precision = \frac{number\ of\ correctly\ tracked}{number\ of\ tracked} \tag{10}$$

The Classification of Events, Activities, and Relationships (CLEAR) is proposed by Bernardin and Stiefelhagen [20] that applied accuracy and precision to define the performance of tracker. Multiple Object Tracking Precision shows the average dissimilarity between ground truth and all the correctly tracked object. It is computed as:

$$MOTP = \frac{\sum_{t,i} d_{t,i}}{\sum_t c_t} \tag{11}$$

Where c_t represents the total number of matches that generate by tracker and $d_{t,i}$ represent the correctly matched hypotheses to object *i* in frame *t*. It defines how precise the algorithm localizes objects. Another figure is called Multiple Object Tracking Accuracy. It is a combination of 3 mains errors.

$$MOTA = 1 - \frac{\sum_t (FN_t + FP_t + IDSW_t)}{\sum_t GT_t} \tag{12}$$

GT_t is the number of ground truth object in each frame *t*. FN_t is the number of missed assignments. For example, there is an object to track in the frame, but the algorithm failed to register the ID. FP_t is an error caused when tracker give object ID while there is no object to track. The last notation is $IDSW_t$ which stand for ID switch. During the tracking of objects, ID can change from one to another even the tracked object remains the same. It causes an error to the output and limits the performance of the tracking algorithm.

Wu and Nevatia [21] quantified multi-object tracking performance by introducing 5 quality measurements. Mostly Track (MT) count the number of objects that more than 80% of its trajectory is tracked. In contrast, mostly lost (MT) count the number of objects that more than 80% of its trajectory is lost or in other words, less than 20% of its trajectory is tracked. Fragments (Fgmt) counts the time that ground truth trajectory is interrupted. False alarm (FAT) as the name implies, count the number of tracking results that are not associated with any ground-truth object. The last one is ID switch (IDSW) which count the frequency of identity switch between a pair of an object.

SORT-FME focuses on re-associate the identification after the occlusion based on the object's structure before the occlusion happens. The performance evaluation is design in the way that it will reflect the ability of the proposed method to solve the occlusion problem. It is represented as multi-object tracking accuracy.

$$MOTA\% = 1 - \frac{\sum_v IDSG_v}{\sum_v GT_v} \tag{13}$$

Where *v* is the video file index and GT_v is the number of ground truth object in each video file. *IDSG* is the number of changing the ID in each video file.

IV. Performance Evaluation

The performance of SORT-FME is used to compare with the performance of SORT for the same video files by using MOTA% as a key performance indicator. Table II shows the comparison result of SORT and SORT-FME. The 2nd column shows the number of ground truth objects in each test scenario. The 3rd column represents the number of objects that are successfully tracked by SORT. The 4th column shows the successfully tracked results from SORT-FME. From the 5th column are the recall, precision, and F-measure of result from SORT and SORT-FME. Since the number of tracked objects are the same as the number of ground truth objects, the result of recall, precision, and F-measure are the same. The F-Measure of SORT-FME gives the value of 1 for most videos which indicate that the tracking algorithm has full coverage most of the objects. In contrast, F-measure of SORT gives value mostly between 0 and 0.5 which indicate that most of the tracking is lost after occlusion. These values show that the SORT-FME gives better result than SORT in correctly re-associating the ID after occlusion. Multi-object tracking accuracy (MOTA%) for all video experiments is also calculated based on tracking result from SORT and SORT-FME.

MOTA% for SORT is:

$$MOTA\%_{SORT} = 1 - \frac{18}{35} = 48.5\%$$

MOTA% for SORT-FME is:

$$MOTA\%_{SORT-FME} = 1 - \frac{11}{35} = 68.5\%$$

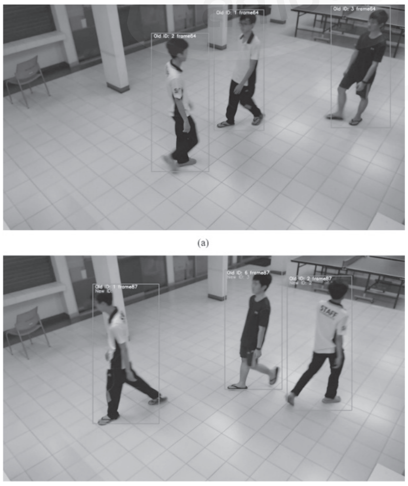
Looking at MOTA%, the proposed method can increase the accuracy of SORT up to 20%. In most case, SORT can keep track only one object in the event of occlusion. The object that is totally occluded by the foreground object will be lost. After occlusion, SORT will apply a new ID to the occluded object, hence the ID is inconsistency. By applying feature extraction and comparison to SORT, lost tracks ID of an object can be recovered and reassigned to that object. Fig. 3 (a) and (b) shows the tracking before occlusion and the successful tracking after occlusion by SORT_FME, respectively. In a certain case; however, the algorithm fails to reassign ID because the object changes perspective appearance during occlusion. The feature comparison step can also be confused by the visual similarity among objects and give false ID to the output.

TABLE II

Tracking Result of Video Experiments

Test	Number of Ground Truth	Correctly Tracked		Recall		Precision		F-Measure	
		SORT	SORT-FME	SORT	SORT-FME	SORT	SORT-FME	SORT	SORT-FME
1	2	1	2	0.5	1	0.5	1	0.5	1
2	2	1	2	0.5	1	0.5	1	0.5	1
3	2	0	2	0	1	0	1	0	1
4	2	2	0	1	0	1	0	1	0
5	2	1	2	0.5	1	0.5	1	0.5	1
6	3	0	0	0	0	0	0	0	0
7	2	1	2	0.5	1	0.5	1	0.5	1
8	4	1	2	0.25	0.5	0.25	0.5	0.25	0.5
9	4	3	4	0.75	1	0.75	1	0.75	1
10	4	3	3	0.75	0.75	0.75	0.75	0.75	0.75
11	2	2	2	1	1	1	1	1	1
12	3	2	3	0.66	1	0.66	1	0.66	1
13	3	0	0	0	0	0	0	0	0

In test 6 and test 13, there are 3 people in the videos. The people occlude for a long period of time and multiple times. The occlusion of the second pair happens before the occlusion of first pair finish. People in the video are walking with non-linear motion. For SORT, the tracking is failed because there are long occlusion and non-linear movements. These problems are supposed to be solved by SORT-FME but because there are 2 occlusions happen within the same time frame, the algorithm cannot match the new pairs to several old pairs of the object. SORT-FME is designed to match only one new pair to one old pair.



(a)

(b)

Fig. 3. Before occlusion (b) After occlusion

V. Conclusion And Future Work

The goal of object tracking is to make the computer understand the spatial and temporal context of video sequences. Computer processes object's information then keeps associating the same object throughout the video. Occlusion is one of the problems that make the tracking algorithm failed as the information of the object being tracked is replaced by the overlapped object. SORT-FME solves this problem by introducing a feature matching method. It enhances the ability of SORT to remember the feature of the tracked object. This new method can improve multi-object tracking accuracy of SORT up to 20% to become the object tracking algorithm that is more robust to occlusion.

Since the baseline model consumes less computational power, this algorithm can work in real-time with a live stream camera such as CCTV. However, for the current state of the model, it can only keep track of 2 people and 3 people from simple occlusion. If all 3 or more people occlude at the same time and at the same place, the method can face assignment problem and tends to fail.

For future work, non-linear motion model such Particle filter [22] can be used to fill a gap of non-linear movement of the object. Also, combinatorial optimization can be added on to solve the matching problem for tracking 3 and more people which occlude at the same time. More and larger testing data can be added so that the tracking performance can be improved.

References

[1] A. Gruen, "Adaptive least squares correlation: a powerful image matching technique," *South African Journal of Photogrammetry, Remote Sensing and Cartography,* vol. 14, no. 3, pp. 175-187, 1985.

[2] A. Nakhmani and A. Tannenbaum, "A new distance measure based on generalized image normalized cross-correlation for robust video tracking and image recognition," *Pattern recognition letters,* vol. 34, no. 3, pp. 315-321, 2013.

[3] S.-Q. Chen, "A corner matching algorithm based on Harris operator," ICIECS 2010, Wuhan, 2010, pp. 1-2.

[4] D. G. Lowe, "Object recognition from local scale-invariant features," ICCV, Corfu, 1999, pp. 1150-1157.

[5] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (SURF)," *Computer vision and image understanding,* vol. 110, no. 3, pp. 346-359, 2008.

[6] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An efficient alternative to SIFT or SURF," ICCV, Barcelona, 2011, pp. 2564-2571.

[7] D. G. Viswanathan, "Features from accelerated segment test (fast)," in *Proc*. ICISA 2014.

[8] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, "Brief: Binary robust independent elementary features," ECCV, Crete, 2010, pp. 778-792.

[9] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," NIPS, Montreal, 2015, pp. 91-99.

[10] R. Girshick, "Fast r-cnn," Proc. IEEE-ICCV, Santiago, 2015, pp. 1440-1448.

[11] D. Comaniciu, V. Ramesh, and P. Meer, "Real-time tracking of non-rigid objects using mean shift," CVPR, South Carolina, 2000, pp. 142-149.

[12] C. I. Patel and R. Patel, "Contour based object tracking," *International Journal of Computer and Electrical Engineering,* vol. 4, no. 4, pp. 525, 2012.

[13] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft, "Simple online and realtime tracking," ICIP, Phoenix, 2016, pp. 3464-3468.

[14] A. Sadeghian, A. Alahi, and S. Savarese, "Tracking the untrackable: Learning to track multiple cues with long-term dependencies," Proc. IEEE-ICCV, Venice, 2017, pp. 300-311.

[15] J. Xiang, G. Zhang, and J. Hou, "Online Multi-Object Tracking Based on Feature Representation and Bayesian Filtering within a Deep Learning Architecture," *IEEE Access,* vol. 7, pp. 27923-27935, Feb. 2019.

[16] K. Fang, Y. Xiang, X. Li, and S. Savarese, "Recurrent autoregressive networks for online multi-object tracking," WACV, California, 2018, pp. 466-475.

[17] H. Fan and H. Ling, "Sanet: Structure-aware network for visual tracking," CVPR, Hawaii, 2017, pp. 42-49.

[18] G. Bishop, and G. Welch, "An introduction to the kalman filter," *Proc of SIGGRAPH, Course,* vol. 8, no. 27599-23175.

[19] K. Smith, D. Gatica-Perez, J.-M. Odobez, and S. Ba, "Evaluating multi-object tracking," CVPR, San Diego, 2005, pp. 36-36.

[20] K. Bernardin and R. Stiefelhagen, "Evaluating multiple object tracking performance: the CLEAR MOT metrics," *Journal on Image and Video Processing,* vol. 2008, pp. 1, 2008.

[21] B. Wu and R. Nevatia, "Tracking of multiple, partially occluded humans based on static body part detection," CVPR, New York, 2006, pp. 951-958.

[22] R. Van Der Merwe, A. Doucet, N. De Freitas, and E. A. Wan, "The unscented particle filter," NIPS, Vancouver, 2001, pp. 584-590.

Koksal Chou is a Master Student in Management Technology at Sirindhorn International Institute of Technology (SIIT), Thammasat University. He received his Bachelor of Engineering (B.E) degree in Electrical and Electronic Engineering from Institute of Technology of Cambodia, Phnom Penh, Cambodia in 2017. In the same year he received the Excellent Foreign Student (EFS) scholarship award from Sirindhorn International Institute of Technology (SIIT), Thammasat University (TU), Thailand. His research interests are Computer Vision, Deep Learning, Optimization, and Supply Chain Management.

Natsuda Kaothanthong is a Lecturer of School of Management Technology (MT), Sirindhorn International Institute of Technology, Thammasat University, Thailand. She received a B.S. degree in Information Technology from Sirindhorn International Institute of Technology. She received her M.S and Ph.D. in Information Sciences from Graduate School of Information Sciences, Tohoku University, Japan. Her Research interests are Data Mining, Business Intelligence, Computer Vision, and Computational Geometry.

Chawalit Jeenanunta is an associate professor of School of Management Technology (MT), Sirindhorn International Institute of Technology, Thammasat University, Thailand. He received a B.S. degree in Mathematics and Computer Science, and M.Sc. in Management Science from University of Maryland and he received his Ph.D. in Industrial and Systems Engineering from Virginia Polytechnic Institute and State University. His Research interests are in area of applications of operations research, simulation, large-scaled optimization and supply chain management.