Region-Based Multiple Object Tracking with LSTM Supported Trajectories

Manish Khare[1(✉)], Manan Mapara[1], Noopur Srivastava[2], and Bakul Gohel[1]

[1] DA-IICT, Gandhinagar, Gujarat, India
mkharejk@gmail.com, {201911031,bakul_gohel}@daiict.ac.in
[2] Shri Ramswaroop Memorial University, Lucknow-Deva Road, Barabanki, Uttar Pradesh, India
noopurs6@gmail.com

Abstract. Object Tracking is the growing field in computer vision with its demands in various areas in monitoring and surveillance. Areas of surveillance can be improved with proper and efficient trackers that can ensure people's safety on roads, the safety of children in school, and in many other areas. Object tracking is a super-set of elements object classification, object detection, etc. Most of the work done consists of tracking based on visual features, so we have worked on region-based features. In this context, a method is proposed for tracking, based on region features extracted with the help of intersection over union and prediction of trajectories with the help of LSTM in case of occlusion. Comparison for the same is being carried out with the traditional centroid tracking algorithm.

Keywords: Multi-object tracking · LSTM · Dataset · Object detection

1 Introduction

Object tracking is detecting the object in a video, i.e. consecutive frames of images. It can be stated as tracking the object's trajectory throughout the sequences in which it is detected. Object tracking is the second stage, but the primary stage is object detection, so better detection accuracy will lead to better trajectory tracing. Object tracking is growing day by day. It is being used in several areas such as CCTV monitoring, traffic control, and self-driven vehicles to provide a good driving experience avoiding accidents with pedestrians and other objects on the streets. The naive idea is to use template matching throughout the video sequence to track a single object, which can be done using standard convolutional neural networks. Object detection is the classical problem of computer vision which is divided into two parts object classification and object localization i.e. finding the location of an object in the image. Object classification identifies the presence of an object in the image but fails when there are multiple objects present in the image.

Object localization is the process of enclosing the identified object with the help of a bounding box. The bounding box is a rectangle with four parameters: centre, height, width, and the class of objects, i.e., the enclosed object belongs, e.g. car, person. Going through the literature in object detection and object tracking, the depth of the problem, its

T. Zlateva and R. Goleva (Eds.): CSECS 2022, LNICST 450, pp. 60–72, 2022.
https://doi.org/10.1007/978-3-031-17292-2_5

importance, and the challenges were understood, which are open for research. Most of the work done in this domain involves standard deep learning and computer vision methodologies, mainly focusing on visual features. The growing demand for object tracking in traffic monitoring, CCTV surveillance, pedestrian detection shows an opportunity for research in this Area. It is being found that the accuracy of the tracker solely depends on object detection. So the main goal is proposing a method of tracking with proper object detection with region-based features, i.e. visualizing the effect of region-based features for tracking.

There are several challenges in multiple object tracking, but those to be considered of higher importance are occlusion handling and identifying objects in the required frame of reference. In this context, the problem of occlusion is tackled in the case of region-based analysis. Tracking of person takes place with the help of region they occupy in the respective frames and predicting the missing frames with the help of LSTM models, i.e. dealing with the case of occlusion.

The organization of the paper is as follows: Sect. 2 contains related work on Multiple object tracking. Section 3 presents the proposed methodology. Section 4 presented experimental results and analysis. Section 5 presents the conclusions and future work of the study.

2 Related Work

The goal of tracking is achieved by proper object detection; two methods used for object detection [8] are YOLO [9] and Faster R-CNN [10]. As YOLO is the state-of-the-art detection method, we have used it for detection purposes. Nig et al. [6] mainly discusses the idea of object tracking with the help of YOLO and LSTM [7] to preserve location histories. It works efficiently in both the temporal and spatial domains, but the prime focus is on the spatial domain to solve occlusion and motion blur issues. YOLO is used in the first stage for feature extraction and LSTM in the second stage for sequence processing. The algorithm is deep in both senses, spatial and temporal. It uses 4096 visual features, so it is a fusion of high-level visual features and location histories.

Yilmaz et al. [16] published a survey on object tracking, which discusses the state-of-the-art tracking methodologies and newly emerging trends. The first use of object tracking is in gesture recognition, and we can identify gestures with the help of contour detection and convexity defects. The bottom-up approach to designing an object tracker is discussed, starting from object representation, feature extraction/selection, object detection, and object tracking. Object representation is formulating an object in terms of points, contours, etc., and the kind of object representation is selected based on application. Feature selection is essential to identify the object because if accurate features are not obtained, the object might not be adequately detected in further processing. Color, edges, optical flow, and texture are the four kinds of features. The next stage is object detection. The techniques used are point detectors such as SIFT [13], background subtraction, and segmentation. The final stage is object tracking and is being divided into three categories: point tracking and kernel and silhouette tracking. All three methods are differentiated on the type of their object enclosing.

If many shapes are present, one should opt for silhouette tracking, while kernel tracking is best if motion is of significant concern. Chen et al. [4] discussed the idea of

intelligent mobility applications in traffic places. A comparison between Yolo and SSD is being made in the first part on datasets that contains a traffic environment. In the second part, distance estimation is done using the mono depth algorithm, an un-supervised CNN approach, and outputs a disparity map. Accurate disparity maps are obtained in the case of VGG as compared to ResNet. However, only obtaining the disparity maps is of no use as in most cases, the relation between the objects in the image and disparity maps is significantly less. So disparity maps are combined with the object detection approaches, which generate bounding boxes around detected objects. The main aim was to develop pedestrian tracking to avoid accidents.

YOLO v3 [5] turned out to be better than SSD on data that is highly filled up with objects in case of object detection. Tang et al. [14] discussed the idea of tracking multiple people with a graph-based approach. The main idea is that the person with an almost similar appearance in object detection may not be the same. Deep net-works are trained to re-identify the person with the help of lifted multi-cut problems multicut problems comprised of regular and lifted edges. Regular edges are the node connections that comprise the feasible solution, while lifted edges are long-range in-formation that connects nodes without changing the feasible solutions. Two objects can be placed in a single cluster if there exists a path between them. The primary goal of this paper is the re-identification of the object (person).

Tracking by detection is the idea of detecting objects and then associating those detections. The simple track by detection method does not handle the problem of occlusion. So there needs to be some relation between consecutive frames so that the object does not lose out to occlusion. To avoid the loss of objects, mean shift was used by searching over a larger region of interest using similarity measures. In contrast, the method of optical flow uses the idea of constant velocity and predict the trajectory. However, both methods are computationally costly and not able to per-form well in case of occlusion. On the other hand, deep sort [15] uses a Kalman filter for linear trajectory prediction and then associates the object with target and predict-ed trajectory with the help of a deep association metric.

The idea of centroid tracking [12, 17] is the traditional method that does not use visual features and only works with the help of region-based features, i.e. Euclidean distance. The first step is object detection over all frames and initialization of object list with objects of frame-1 and computing centroids of all the objects. The second step is the association based on Euclidean distance for objects of frame-2 onwards. So centroids for the objects of the current frame are computed, and then the Euclidean distance between the object list and current objects is being performed. The one which satisfies the threshold distance turns out to be the associated object in the object list. If the threshold distance is not satisfied, then a new object is created. After object association centroids of all the objects are updated, the same procedure continues for all other frames.

3 Proposed Methodology

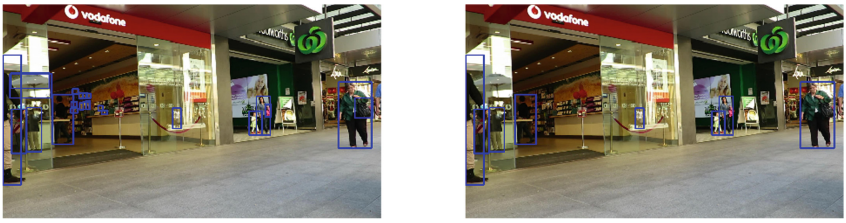
The simple IOU-based method applies IOU [11] only in consecutive frames and tries to find the association, but this leads to loss of objects in high number. As the traditional method is centroid tracking, we have tried to overcome all the drawbacks of the traditional method in our proposed method. Two methods are proposed for the same.

3.1 Proposed Method 1

Here, we set a goal to track objects based on their region over the frames and propose the methodology for multiple object tracking, specific person. The whole idea is divided into: Detection, Association, Trajectory prediction, Combining the frames, and association of Ids.

3.1.1 Object Detection

YOLO v4 [3] is used for object detection over all the frames. For YOLO v4, the standard weights of the paper that work over the coco dataset are used. A video sequence comprises 1000 frames, then on all 1000 frames, Yolo is applied, and the respective bounding box is obtained for the objects in every frame. As an output, we get objects of every class, but the main concern is person class, so we threshold the image with confidence as 25% and object class as a person. Figure 1 shows object thresholding, and Fig. 2 shows Object detection in consecutive frames.



(a) Unthresholded Image (b) Thresholded Image

Fig. 1. Object thresholding

(a) Objects in Frame 1 (b) Object in Frame 2

Fig. 2. Object detection

3.1.2 Object Association

For association, we use the thresholded images and initialize the object list with objects of frame-1. After that, every object in frame-2 is associated with all the objects from

the object list. However, the occlusion coefficient is kept as 15. As we are dealing with the problem of occlusion, we have defined the term occlusion coefficient. The occlusion coefficient is the numerical difference between the current frame and the frame of the object which is being associated. E.g., The current frame is 50, and the object list contains 15 objects and their trajectories among different frames. So, all the detected objects of frame-50 will be associated with those objects from the object list whose last trajectory has a frame difference of less than or equal to 15, like object-15 has the last trajectory of frame-40. It will be accepted while object-10 has the last trajectory of frame-25, then it will be rejected for the association.

The frame selected for the association will be the frame of the last trajectory in the object list. The Current frame is 50, but the object frame is 47; the association will be performed on frame-47 with bounding box coordinates. The intersection over union for all eligible objects will be calculated.

$$IOU = \frac{Area}{(Area + Area1 + Area2)} \tag{1}$$

In Eq. (1), Area is the intersection of two bounding boxes, and Area 1 is the unique region of bounding box 1 and Area 2 is the unique region of bounding box 2. After calculating it, the maximum intersection over union will be taken for the association.

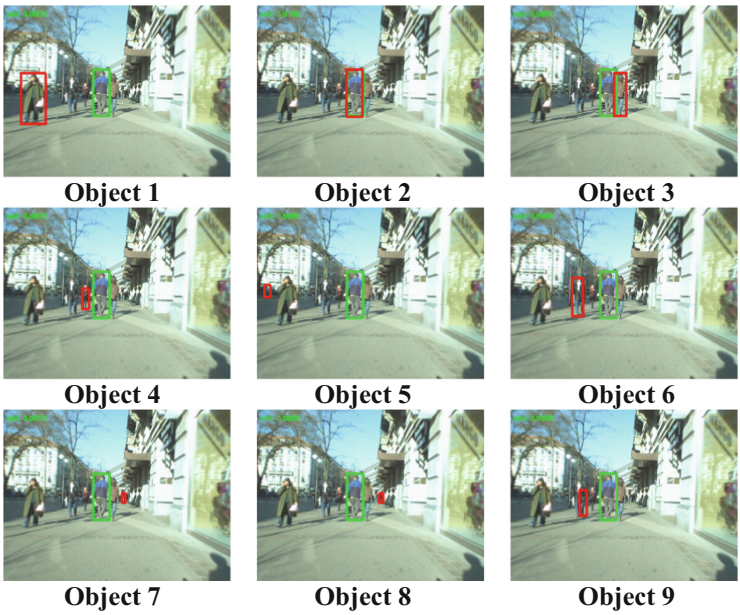


Fig. 3. Association of objects

If the maximum value is less than 20%, then a new object will be created; else, it will be appended to the corresponding object with maximum intersection over the union. Keeping the threshold of 20% is that the IOU takes place in the frame of the object list,

not in the current frame. So there are high chances that the shift might have occurred between the objects as the occlusion coefficient is kept as 15. Association can be well understood from Fig. 3. The green bounding box is the object to be associated, while the red bounding box is the object from the object list. The best association is found with object-2, as shown in Fig. 3(b). Here both the green and the red bounding box completely overlapped with an IOU of 91%, while the association with all other objects is 0 or some other value less than 91% making them false associations.

After performing association, we get a list of all objects with their trajectories. However, as YOLO v4 is a robust detection algorithm, there are chances that an object has multiple bounding boxes in the same frame; in that case bounding box of greater height is considered, and the other one is discarded.

3.1.3 Trajectory Prediction

As we have kept the occlusion coefficient as 15, there are chances that when the object is occluded in some frames, we will not have trajectories of that frame, so those trajectories are to be predicted. Two LSTM models of look back 1 and 3 are trained over trajectories of 570 different objects for predicting the trajectories.

For object's we have trajectories in the frame-70, 72, 74, 75, 76, 77. From this, we have two trajectories missing in frames 71 and 73. So trajectory for frame 71 will be predicted with a look back 1 with frame 70, and for prediction of frame 73, the previous frames are more than two, so in that case, the prediction will take place with a look back at 3 with frames 70, 71, 72. The trajectory list is updated after every prediction.

By proposed method-1, all the drawbacks of centroid tracks were cleared, but ID generation was a bit higher, so to reduce the number of ID's proposed method-2 is given which is a slight modification of proposed method-1 and reduces ID up to a certain extent.

3.2 Proposed Method 2

In this method, the complete architecture is kept the same as proposed method-1; only a single change is done when creating new objects. When the *IOU* turns out to be less than the threshold, we search those objects whose Euclidean distance is less than 50. If the search gives 0 objects, we create a new object; otherwise, we compute the difference between Euclidean distance and IOU of all objects whose Euclidean distance is less than 50. Append operation is being performed, i.e. appending trajectory to the object with whom the difference turned out to be minimum; by this, we avoided creating a new object. So proposed method-2 is an improvised version of proposed method-1, which gave better results with fewer IDs.

Figure 4 shows the image flow diagram, and Fig. 5 shows the block diagram, and from both figures proposed approach can be understood pictorially.

Let say two centroids are (x_1, y_1) and (x_2, y_2). Euclidean distance between them is computed as

$$e1 = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \tag{2}$$

The minimum value object is being selected for association using threshold check. Threshold check is the difference between IOU and Euclidean distance and is computed as $(|e1 - IOU|)$ for every object where $\max(IOU) \leq 20\%$.

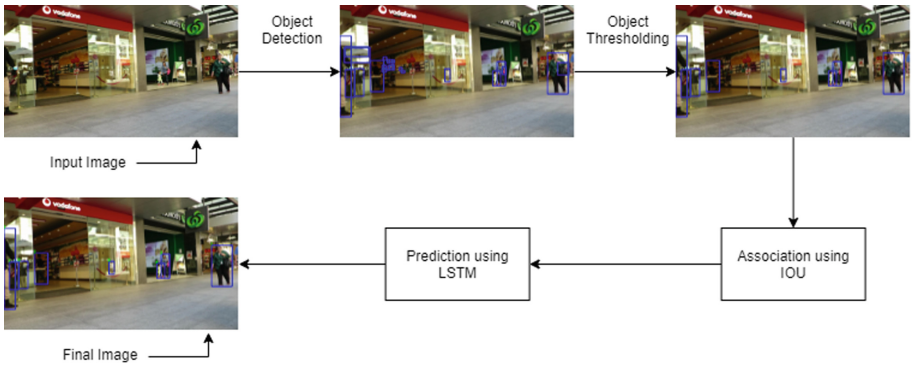


Fig. 4. Image flow diagram

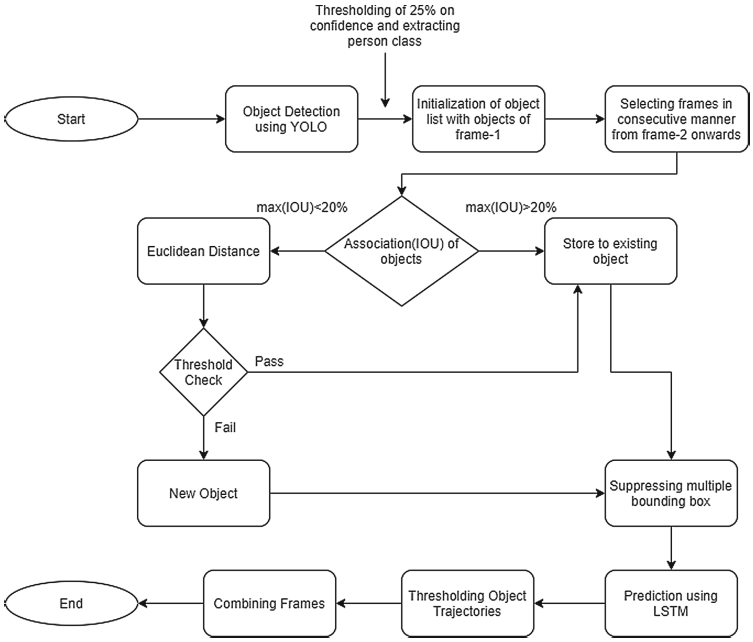


Fig. 5. Block diagram of the proposed method

4 Evaluation and Dataset Used in Experiments

Evaluation is being carried out with the help of standard CLEAR MOT [2] metrics. Some evaluations which we have used in our experimentation are given below:

MOTP: Mean error between the measurements obtained and the ground truth pro-vided. Less the value better the results and can also be represented in percentage by deducting the MOTP from 1 and multiplying by 100. In the case of percentage representation, the higher the value better is the result.
Matches: Object matches between the ground truth and measurement in every frame. The higher the number, was better the result.
Mismatches: Identity mismatch between ground truth and measurement. So a swap of identity in a single frame will lead to 2 mismatches. Lower the number better the result.
False Negatives: Un-identification of the ground truth object in measurement. i.e. the ground truth object is present but in measurement is not identified. Lower the number better the result.
False Positives: False identification of the ground truth object in measurement. i.e. the ground truth object is not present, but in measurement, it is identified. Lower the number better the result.
Object Count and ID: The total number of objects detected in all the frames while ID is the object identity among the total number of objects.

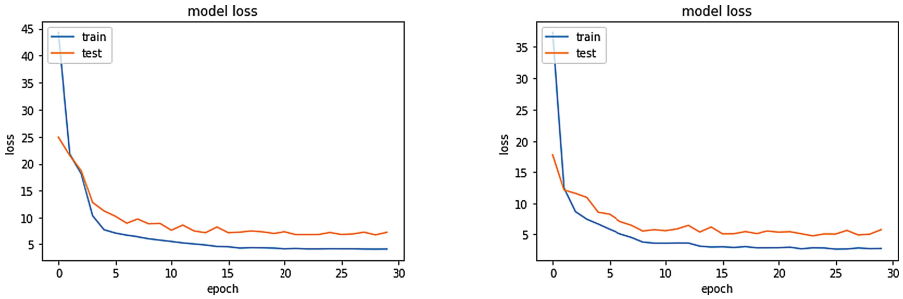
In this work, the MOT15 dataset [1] is used. It contains colored video sequences with different resolutions ranging from standard to full HD. The reason behind choosing this dataset is that it is filmed with both static and moving cameras with a prime focus on pedestrians on the streets. Video sequences with ground truth data are provided for evaluation. Table 1 shows the details of the dataset.

Table 1. Video details of 2D MOT15 dataset

	Video Name	Resolution	Number of frames
1	ADL-Rundle-6	1920 × 1080	525
2	ADL-Rundle-8	1920 × 1080	654
3	ETH-Bahnhof	640 × 480	1000
4	ETH-Pedcross2	640 × 480	837
5	ETH-Sunnyday	640 × 480	354
6	KITTI-13	1242 × 375	340
7	KITTI-17	1224 × 370	145
8	PETS09-S2L1	768 × 576	795
9	TUD-Campus	640 × 480	71
10	TUD-Stadtmitte	640 × 480	179
11	Venice-2	1920 × 1080	600

All experiments and results have been carried out on Google collab GPU. LSTM model is trained with 5 inputs; 4 inputs of the bounding box, i.e. x-center, y-centre, height, width, and 5th input is the frame's aspect ratio. Parameters for lookback = 1 are kept as epochs = 30, optimizer = adam, batch size = 128, loss = mean absolute percentage error, and.

3 Stacked LSTM units while for lookback = 3 only batch size = 64 is changed. Loss curves for lstm models are shown in Fig. 6.



(a) Model with Lookback-1 (b). Model with Lookback-3

Fig. 6. Loss curve for LSTM model

5 Experimental Results and Analysis

We have experimented on standard publicly available datasets as mentioned in Table 1. We have compared both proposed methods with centroid tracking methods. Here, we presented qualitative and quantitative results for one video (ETH-Bahnhof Dataset), as discussed in Table 1. We have shared results for other videos on the website https://sites.google.com/site/mkharejk/research due to page restriction as per conference guidelines.

Experiment 1: ETH-Bahnhof Dataset

The ETH-Bahnhof dataset contains 1000 frames with a resolution of 640 × 480. Experimental results for Centroid tracking and both proposed methods are given in Figs. 7, 8, and 9. Quantitative evaluation values are given in Table 2.

Fig. 7. Multi-object tracking results for ETH-Bahnhof video sequence using centroid tracking

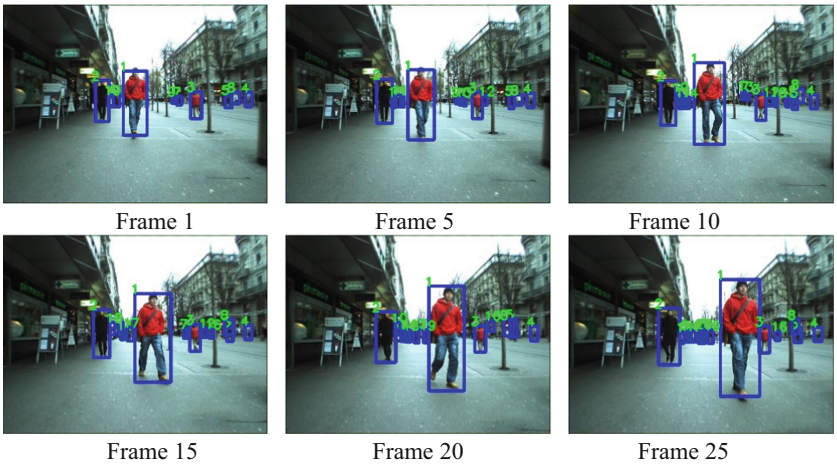


Fig. 8. Multi-object tracking results for ETH-Bahnhof video sequence using proposed method 1

From Fig. 7, we can see that the same object is losing its id as it moves forward. In frame-1, the object is labelled as id-1, in frame-5,10 labelled as id-6, and in frames 15,20,25 labelled as 13. Another evident issue is that the nearby objects have been labelled as the same object. From Fig. 8, we can see that the drawbacks of the traditional method shown in Fig. 7 are entirely resolved. From Fig. 9, we can see that proposed method 2 outperformed in comparison to proposed method 1. From the frames, we can see that occlusion is tackled well in the proposed method and fewer id switches.

We can see from Table 2 that both proposed methods have outperformed the baseline method while proposed approach-1 has also outperformed proposed approach-2 - 4

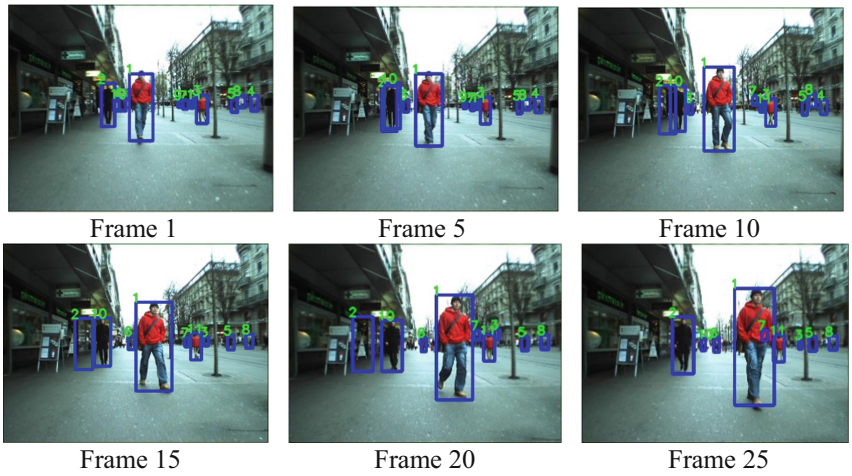


Fig. 9. Multi-object tracking results for ETH-Bahnhof video sequence using proposed method 2

Table 2. Quantitative analysis of ETH-Bahnhof video sequence

		MOTP	ID	False positive	False negative	Matches	Mismatches
Proposed method-1	Threshold-0	0.69	444	14710	7216	454	**28**
	Threshold-15	0.72	216	11565	5335	2335	**252**
	Threshold-25	**0.6**	169	10430	**5162**	**2508**	**237**
Proposed method-2	Threshold-0	**0.6**	97	**7599**	5822	1848	**104**
	Threshold-15	0.61	81	8363	6683	987	**47**
	Threshold-25	0.61	**72**	8300	6794	876	**36**
Centroid tracking		**0.63**	**950**	**12310**	**7579**	**91**	**40**

metrics (MOTP, False Negative, Matches, Mismatches) out of 6 favoured proposed approach-1.

Please visit my website https://sites.google.com/site/mkharejk/research, on which we shared all other video sequence experimental results and quantitative performance measures values.

By analyzing different video sequences and the nature of individual objects, we figured out that an object is occluded up to 15 frames in standard cases, so the occlusion coefficient is kept as 15. Trials have been carried out with different occlusion coefficients. However, as we increase the value of the occlusion coefficient, the number of object detection remains the same. However, the unique id association count reduces, which increases the false entries, i.e. different objects given the same id.

A similar observation is being drawn from the case of the IOU threshold being kept as 20%. The reason behind keeping it 20% is that in the worst case, the difference between the current frame and the object frame is 15, which can be associated, so there would be a deviation in the object's position. However, as the threshold is 20%, there are fewer chances of missing out on the object.

Thresholding of trajectories used in proposed methods keeps only those whose trajectory length is more significant than the threshold, i.e. for threshold-0, all objects will be considered. In contrast, threshold-5 objects with trajectory lengths greater than 5 will be considered. In general, on increasing the threshold, a positive sign is developed in MOTP, ID, False positives, False Negatives, and Matches while the negative sign for Mismatches.

The key factors were the IOU threshold, trajectory threshold, and the occlusion coefficient in the architecture, so they can be summarized as follows.

$$IOU\ Threshold \quad \alpha \quad \frac{1}{\mathrm{Re}sult} \tag{3}$$

$$Occulusion\ Coefficient \quad \alpha \quad \frac{1}{\mathrm{Re}sults} \tag{4}$$

$$Trajectory\ Threshold \quad \alpha \quad \mathrm{Re}sults \tag{5}$$

We can see that the IOU threshold and occlusion coefficient inversely affects the result while the trajectory threshold directly affects the result.

6 Conclusions and Future Directions

This work mainly focused on the idea of region-based multiple object tracking. So we proposed two methods to deal with this, which gave better results than the traditional method of region-based analysis. The proposed method is extending the evaluation metric IOU for tracking. Method examination is carried out using a mot-precision, false positives, false negatives, matches, mismatches, and ids. The algorithm's efficacy cannot be decided from a single metric, so we have taken the metrics count to favor the algorithm. All drawbacks of the standard method are resolved by occlusion tackling, a high number of ids and the same id to different objects in the same frame. All metrics have outperformed the traditional method in various videos tested. The problem of occlusion is being tackled up to a certain extent. For a steady object, occlusion is tackled well without loss of id and trajectory, while in the case of moving object, the trajectory is not lost, but the id switch takes place after some frames.

Both the proposed methods have outperformed the traditional method, but out of the 11 video sequences, an observation can be derived as follows. Proposed Approach-1 works well in case of more number frames. Proposed Approach-2 works well in case of less number of frames and works well where frames are more as well object crossings are high in number.

As the goal is to tackle the tracking problem with a region-based approach without taking into consideration the visual features, this work can be extended to tackle the problem of id switches in case of occlusion of moving objects by associating occlusion coefficient with some other parameter that deals with the region of interest.

Acknowledgment. This work was supported by the Science and Engineering Research Board (SERB), Department of Science and Technology (DST), New Delhi, India, under Grant No. CRG/2020/001982.

References

1. Taixe, L.I., Milan, A., Reid, I., Roth, S., Schindler, K.: MOTChallenge 2015: towards a benchmark for multi-target tracking. https://arxiv.org/abs/1504.01942 (2015)
2. Rainer, S., Bernadin, K.: Evaluating multiple object tracking performance: the CLEAR MOT metrics. EURASIP J. Image Video Process. **2008**, 10 (2008). Article No. 246309, https://doi.org/10.1155/2008/246309
3. Bochkovskiy, A., Wang, C.Y., Liao, H.Y.M.: Yolov4: optimal speed and accuracy of object detection https://arxiv.org/abs/2004.10934 (2020)
4. Chen, Z., Khemmar, R., Decoux, B., Atahouet, A., Ertaud, J.Y.: Real time object detection, tracking, and distance and motion estimation based on deep learning: application to smart mobility. In: Proceeding of Eighth International Conference on Emerging Security Technologies (EST) (2019)
5. Kathuria, A.: What's new in yolo v3? (2018). https://towardsdatascience.com/yolo-v3-object-detection-53fb7d3bfe6b
6. Ning, G., et al.: Spatially supervised recurrent convolutional neural networks for visual object tracking. In: Proceeding of IEEE International Symposium on Circuits and Systems (2017)
7. Olah, C.: Understanding LSTM networks (2015). https://colah.github.io/posts/2015-08-Understanding-LSTMs/
8. Zhao, Z.Q., Zheng, P., Xu, S.T., Wu, X.: Object detection with deep learning: a review. IEEE Trans. Neural Netw. Learn. Syst. **30**(11), 3212–3232 (2019)
9. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: unified, real-time object detection. In: Proceeding of IEEE Conference on Computer Vision and Pattern Recognition, pp. 779–788 (2016)
10. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. IEEE Trans. Pattern Anal. Mach. Intell. **39**(6), 1137–1149 (2017)
11. Rosebrock, A.: Intersection over union (IOU) for object detection (2016). https://www.pyimagesearch.com/2016/11/07/intersection-over-union-iou-for-object-detection/
12. Rosebrock, A.: Simple object tracking with opencv (2018). https://www.pyimagesearch.com/2018/07/23/simple-object-tracking-with-opencv/
13. Singh, A.: A detailed guide to the powerful sift technique for image matching (with python code) (2019). https://www.analyticsvidhya.com/blog/2019/10/detailed-guide-powerful-sift-technique-image-matching-python/
14. Tang, S., Andriluka, M., Andres, B., Schiele, B.: Multiple people tracking by lifted multicut and person re-identification. In: Proceeding of IEEE Conference on Computer Vision and Pattern Recognition, pp. 3701–3710 (2017)
15. Wojke, N., Bewley, A., Paulus, D.: Simple online and realtime tracking with a deep association metric. In: Proceeding of IEEE International Conference on Image Processing, pp. 3645–3649 (2017)
16. Yilmaz, A., Javed, O., Shah, M.: Object tracking: a survey. ACM Comput. Surv. **38**(4), 13–es (2006)
17. Zhang, R., Ding, J.: Object tracking and detecting based on adaptive background subtraction. Procedia Eng. **29**, 1351–1355 (2012)