Real Time Multi-Object Tracking based on Faster RCNN and Improved Deep Appearance Metric

Mohan Gowda V[1]
Dept. of Computer Science and Engineering
GITAM School of Technology
GITAM Deemed to be University
Bengaluru, India

Megha P Arakeri[2]
Dept. of Information Science and Engineering
Center of Imaging Technologies
Ramaiah Institute of Technology
Bengaluru, India

Abstract—**Computer Vision has set a new trend in image resolution, object detection, object tracking, and more by incorporating advanced techniques from Artificial Intelligence (AI). Object detection and tracking have many use cases such as driverless cars, security systems, patient monitoring, and so on. Various methods have been proposed to overcome the challenges such as long-term occlusion, identity switching, and fragmentation in real-time multi-object detection and tracking. However, reducing the number of identity switches and fragmentation remains unclear in multi-object detection and tracking. Hence, in this paper, we proposed a multi-object detection and tracking technique that involves two stages. The first stage helps to detect the multiple objects with high uniqueness using Faster RCNN and the second stage, Improved Sqrt cosine similarity, helps to track the multiple objects by using appearance and motion features. Finally, we evaluated our proposed technique using the Multi-Object Tracking (MOT) benchmark dataset with current state-of-the-art methods. The proposed technique resulted in enhanced accuracy and reduces identity switching and fragmentation.**

Keywords*—*Multi-object detection; tracking; faster RCNN; convolution neural network; data association

I. Introduction

MOT tracks moving objects with the regular time interval via camera as the input device. In 1998, Zenon Pylyshyn [1] was first developed multi-object tracking. Each detected object is assigned a unique identification number. This identity number retains its association with the object when changing the object's appearance or object moving and draw the motion trajectories of the object based on the unique identities. Fig. 1 shows the basic steps of object detection and tracking. Multi-object detection finds the objects under a unique frame and MOT is integrated with the detected objects in the sequence of frames.

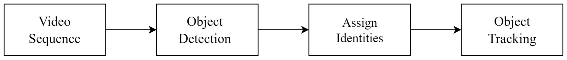


Fig. 1. Basic Steps of Object Detection and Tracking.

A wide range of real-time applications is implemented using MOT, with which it is having extraordinary significance nowadays [2]. Some real-time applications are Human tracking, monitoring Alzheimer's activities, monitoring security, autonomous driving, robotic vision, traffic control, medical images, and others. Gulraiz *et al.* [3] say that tracking is useful because of a specific reason as follows. In the video frame, multiple objects are detected, then establish the identity of the targeted objects in the frame while tracking the objects. Suppose object detection fails, it may be possible to track the object using appearance features and stored location of the previous frame. Local search is initiated instead of global search during tracking. Whenever the movements of the targeted objects are high, the tracking algorithm loses track of the target objects. Hence, the proposed system integrates the detection and tracking methods. The proposed hybrid system has two stages. In the first stage, multi-object detection performs every n^{th} frame of the input. The second stage, multi-object tracking, will take the target objects of n^{th} to $(n+1)^{th}$ frame based on the appearance features and object position in the different frames. Some applications of the proposed work are Alzheimer's patient tracking, monitoring the activities like cooking, hand washing, dressing and others, as shown in Fig. 2.

Fig. 2. Applications of MOT.

In recent years MOT has had many scopes. By the incorporation of AI techniques, the present object detection and tracking techniques are giving better performance than the traditional methods. Most of the traditional methods track the objects from frame to frame. It gives good tracking performance. However, traditional methods are facing computational problems in complex scenarios. They also face difficulty in handling noise and occlusion problems.

Most of the traditional methods are not suitable to adopt in real-time applications. Batch-based movement tracking [4] and probability-based systems [5] must complete batch video processing to track the targeted object. These methods take more time to convert a batch to a performance tracking process. Hence, it is not suitable for real-time scenarios.

Currently, researchers are concentrating on tracking performance by reducing the missing detection rate with the combination of detection and tracking methods. However, we are facing some challenges in tracking moving objects. The challenges are occlusion, identity switching, time efficiency, motion blurring, viewpoint variations, Background changing, low resolutions, etc. In this work we concentrate mainly occlusion in the long term, time efficiency, identity switching and fragmentation. The objective of this research work is given below.

1) Multiple object detection.
2) Develop a technique to reduce identity switching and fragmentation.
3) Compare the performance of the proposed technique with existing techniques.

The batch and probability-based tracking algorithms have some limitations in time efficiency to apply the real-time application. Much work has been done to overcome the above challenge.

Identity switch problem means detected object changing the identity number in the frame to frame. Identity switching problems mainly occur in two situations: failure of object detection and long-term occlusion occurs. To overcome the failure of object detection, we used the deep learning-based multi-object detection method to detect the person, Bread, Jam, Coffee Maker, Coffee cup and Spoon. To avoid the occlusion problem, we mainly focused on the localization features of the object's appearance. Every object has a different appearance and different locations of the frame. Hence we are implementing the tracking based on the appearance features.

Fragmentation of trajectories may fail when the identity switches have not occurred or object detection fails. With the help of motion and appearance features, a complete trajectory path of one fragmented frame to another fragmented frame can be obtained.

To overcome the above challenges, we proposed the novel multi-object detection and tracking technique. This work has two stages: the first stage detects multiple objects using Faster RCNN, and the second stage is tracking the multiple objects using unique appearance and motion features. The proposed method improves the tracking performance by making robustness in the occlusion.

Paper is organization is as followed: Related methodologies for object detection and object tracking are discussed under Section 2. Section 3 provides the architecture of the proposed system. Section 4 describes the evaluations and results. Finally, Section 5 gives the conclusions of the proposed work.

II. Literature Survey

Recent researchers have concentrated on tracking an individual object in various contexts with multi-object detection and tracking progress. The proposed multiple object tracking method is mainly focused on the association problem. The association problem is used to associate the detected object of one frame to another frame. Hence object detection is carried out before object tracking. The following section primarily focuses on the existing methods and methodologies for object detection and object tracking.

A. Object Detection Algorithms

In the 1990's Anil *et al.* [6] proposed object detection by object matching using deformable templates. These templates have prior knowledge of the object shape like edges and set of edge information. In the late 1990s, object detection was based on the associated geometric appearance feature [7]. The geometric appearance methods involve some geometric properties such as height, width, angle and so on.

Object recognition was moved to the low-level characteristics of the image in the 2000s, based on statistical classifiers. Ojala *et al.* [8] developed rotation invariant classification for grayscale images using the binary patterns locally. Dalal *et al.* [9] proposed Histogram oriented gradients method of object detection in static images. Lowe *et al.* [10] present a method to extract the invariant features of the images. The features are the uniform image scale and rotation, 3D viewpoint and addition of noise. Tuzel *et al.* [11] describe the covariance-based computation method on the internal images. Compared to other statistical methods, a covariance matrix is better to handle large rotations and illumination changes.

Handcrafted conventional features were adopted for object detection in the computer vision sector for many years. In 2012 the deep learning method gave terrific results for the image classification challenge [12]. After successful classification, the researcher concentrated on object detection using deep learning. A Convolution Neural Network (CNN) acts as a backbone network for object detection in deep learning. The CNN is used to extract the local and global feature maps of the input image. Researchers use different backbone networks such as VGG16, AlexaNet, MobileNet, ResNet, and others to achieve the best accuracy.

Now-a-days, research community has moved to region-based networks for object detection. Gulrciz *et al.* [13] proposed a video-based variety of object interaction and spatial relations of the objects. However, to solve a more complex object detection problem, we need to find the object's coordinates in the input image. Girshick *et al.* [14] developed a region Convolution neural network (RCNN) for object detection to overcome the above problem. Here instead of running the classification of many regions. Firstly, they use selective search to extract the region from the image. Then classification will run on the extracted regions. The RCNN has four steps as follows. The selective search algorithm passes on images to generate a region proposal network. Once the region of interest for each image is determined, then resize all proposed regions to match the predefined size of the classes. SVM classifier is used to classify the object and background of the image. Finally, train a linear regression model to generate the bounding boxes of the detected objects. The RCNN has some drawbacks that are as follows. RCNN consumes more time in the training process because the selective search generates the Region of interest. For classification purposes, they used a separate SVM classifier. It is expensive to extract the convolution feature maps for individual regions.

Spatial Pyramid Pooling (SPP-net) method [15] takes any size of the input image. In the SPP-net method, there is no need to compute the convolution feature maps of every region repeatedly. SPP-net generates the entire image features map at a time.

Girshick *et al.* [16] proposed the Fast RCNN to address the above problem. Fast RCNN is similar to RCNN, and Fast RCNN feeds full input images to CNN to generate the feature map. Then identify the different region proposals from the convolution feature map. The region proposals are different in size. Hence they add these region proposals to the Region of Interest (ROI). Pooling Layer to generate the fixed-size feature maps of the individual regions. The ROI feature vector is further split into two divisions that are classification and regression. Using softmax layer the image is classified into a predicted object and background object and regression is used to generate the bounding boxes. Compared to RCNN, the Fast RCNN has better performance.

Ren *et al.* [17] proposed the Faster RCNN object detection algorithm similar to Fast RCNN. To predict the region proposal, they used a separate network instead of a selective search algorithm. Redom *et al.* [18] proposed the YOLO object detection method. YOLO does not use the region proposal step. It simultaneously detects all bounding boxes of all classes in the input image. Hence YOLO can be an optimized, end-to-end training model. YOLO describes the image into S*S grids. Each grid has probabilities of B (Bounding boxes), C (classes). Yolo can predict the object at 45fps while running the real-time images. However, missed detection raises to identity switch and fragmentation issues.

B. Object Tracking

Some Researchers have investigated spatial features for multiple object tracking [19] and appearance-based approach to capture the association between previous and currently detected frames [20]. Motion-based multiple object target tracking with similar appearance features was proposed in [19]. This method recovers missing data efficiently during the long-term occlusion and also reduces misidentification. However, this method fails to maintain the association between different frames. JuHog *et al.* [21] constructed a Relative Motion Network (RMN) method to track a relative movement between the camera and objects resulting in better data association between different frames and accurate tracking during the camera movement.

Donald *et al.* [22] developed an algorithm for multiple target tracking and Fortmann*et al.* [23] introduced the Joint Probabilistic Data Association (JPDA) algorithm. Rezatofighi *et al.* [4] revisited the JPDA method that has better performance. They utilized some current methods to discover the m-best solution for linear programming. However, there is a delay in decision making. Hence these methods are not suitable for real-time scenarios.

Gabin *et al.* [24] proposed a method to track the multiple football player trajectories from the multiple cameras using a distributed scene algorithm. Improve the performance of online tracking of multiple objects using existing trajectories. Some researchers are using the correlated association for online detection purposes. Yang *et al.* [25] prepared the multi-person online tracking of dynamic appearance features. The temporal dynamic approach incorporates the spatial structure appearance features. This method gives an accurate approach using the data association technique. It also improves the affinity management between the detection and trajectories of the frames. However, it faces the difficulty of online tracking for complex scenes. Xiang *et al.* [26] used the Markow Decision Process (MDP) method to track multiple online objects.

Recently researchers worked on deep learning-based online multi-object tracking of live videos. Alex *et al.* [27] proposed the Simple Online and Real-time Tracking (SORT) method. SORT mainly focused on the association of the objects from one frame to another in online tracking. Firstly, identify the quality of the object detection. Then the Kalman filter is used to estimate the different positions like center, height, aspect ratio and linear velocity of one frame to another. The Kalman filter eliminates the duplicate track of the frame using the Hungarian algorithm. Finally, detect and create the track identities of the object. But, SORT is unable to handle the object re-identification and long-term occlusion problem.

To avoid problems with the SORT method, Nicolai *et al.* [28] integrate the SORT with the in-depth appearance information. In a deep sort, they change the association matrix with the integration of motion and appearance information. Then apply CNN for the target appearance to a large-scale object re-identification dataset. It reduces the loss track of the long-term occlusion. However, this system misses the track during the poster changes in real-time and detects the large frames.

Gulraiz *et al.* [3] proposed multi-person detection and tracking using state-of-art methods. Faster RCNN is used for accurate detection and the deep SORT method is used for tracking. It gives better accuracy in real-time human detection. In tracking, it misses re-identification of objects from one frame to another.

In this paper we propose a system to reduce the number of objects being missed from detection in multiple objects using Faster RCNN. Improved sqrt Cosine similarity method is applied to overcome the object re-identification problem. It uses to find the association matric between two consecutive frames.

III. Proposed Method

The Deep learning methods are the better choice for multiple object tracking. Object detection is the foundation of multiple object tracking. To solve object detection problems with greater accuracy in real-time, deep learning algorithms are preferable for detection. We have solved the multiple object tracking problems using state-of-the-art techniques. The projected method gives critical components of multiple object position prediction in the future frames, and tracking the association of the different frames and managing the lifespan of the tracked multiple objects.

The CNN methods are commonly used in object detection, such as Region-Based Convolutional Neural networks (R-CNN), Fast CNN, Faster R-CNN, etc. The CNN method divides the image into various regions and then classifies every region into different objects, but it needs many regions to predict the object. R-CNN method selectively searches together with the region, but it requires high computation time and prediction is carried using three different models. Fast R-CNN involves a single model that takes features from regions, classifies them, and delivers the border boxes for each class simultaneously. However, it is a slow and time-consuming process to find the Region of Interest. So we conducted a survey to choose the best detection algorithm that is Faster R-CNN. We have divided the proposed work into three models are shown in Fig. 3. The first one is object detection, the second one is tracking handling, and the third is an association.

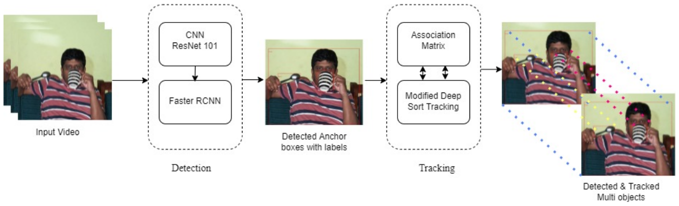

CNN ResNet 101
Faster RCNN
Input Video
Detected Anchor boxes with labels
Detection
Association Matrix
Modified Deep Sort Tracking
Tracking
Detected & Tracked Multi objects

Fig. 3. Block Diagram of Proposed System.

A. Multi-Object Detection using Faster RCNN

The Faster R-CNN involves two stages: Region Proposed Network (RPN) and object detection network. The region proposed network is further divided into three steps. Initially, to extract features map by using a convolutional neural network. After it generates anchor Boxes for using the sliding window approach. Finally, generated anchor boxes are reanalyzed using a tiny network that computes the loss function to select the containing object. The CNN is the backbone of the RPN and object detection network. It requires a step for extracting the convolution feature map.

1) Residual Network-101: The Object detection problem mainly depends upon the feature extraction process. So, we used the ResNet-101 Network model in our method to produce a feature map. ResNet-101 is a convolutional neural network that contains 101 layers between the residential connections. The main advantage of the ResNet-101 is to train the module efficiently without increasing the training error. It also helps to solve the vanishing gradient problem by adding a shortcut connection technique. The shortcut connection is skipped one or more layers to perform the identity mapping. The shortcut connection takes the input x to the output after a few weighted layers. This x is added to the output of the sketch layer. The pictorial representation of the residential network shows in Fig. 4. In different scenarios, the two types of shortcut connections are used in ResNet. When the input and output are the same dimensions then shortcut(x) is used directly.

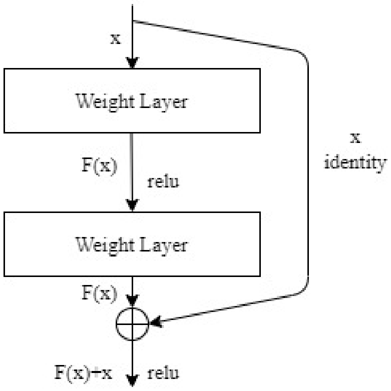


Fig. 4. Building Block of Residual Learning.

$$z = F(x, w_i) + x \tag{1}$$

When the input and output are different dimensions. Then identity mapping is preferred by padding extra 0 to make dimensions suitable.

$$z = F(x, w_i) + w_j x \tag{2}$$

In equations 1 and 2, where x is the feature map value of the previous layer, F is the convolution function and w is the weighted matrix.

2) Anchor Box Generation: The RPN takes the convolution feature map that the ResNet-101 generates as an input, and the output is anchor boxes generated by RPN using the Sliding window approach. This Sliding window approach adopts a 3 * 3 window size upon the future map. Sliding window traverse across the future map to generate the anchors. The sliding window generates a set of 9 anchors for each pixel, each pixel center point is (x,y). All 9 anchors are three different vertical scales such as 128 * 128, 256 * 256 and 512 * 512 and three different aspect ratios of 1:1, 1:2 and 2:1 as shown in Fig. 5. Then, determine the number of anchor boxes that is overloaded with Background Classes (Bc) using the intersection of union approaches. We set the threshold value to the intersection of the union approach. If the threshold value is greater than 0.8 then consider the object is present in the region. Suppose the threshold value is less than 0.2 then no object is present in the region.

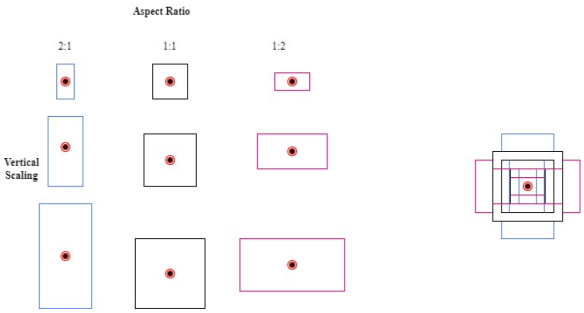


Fig. 5. Nine Anchor are Generted in Each Pixel.

$$IOU = \frac{anchor \cap Bc > 0.8 = ObjectPresent}{anchor \cap Bc > 0.2 = NoObjectPresent} \tag{3}$$

3) Loss Function: The loss function is used to fine-tuning the selected anchor boxes. The loss function mainly contains two tasks: regression and classification. We use binary classification to predict whether the concerned anchor boxes contain the object or background. The regression determines the position of the predicted anchor box. The loss function calculates for both the classification and regression to fine-tuning the anchor box. The loss function is shown in equation 4.

$$L(p_i, t_i) = \frac{1}{N_{cls}(\sum_i L_{cls}(p_i, p_i^*))} + \frac{\lambda}{N_{reg}(\sum_i p_i^* * L_{reg}(t_i, t_i^*))} \tag{4}$$

In equation 4, Where p_i means the predicted probability of anchors containing objects, p_i^* means the ground-truth value of anchors contains an object, t_i coordinates of predicted anchors, t_i^* is ground truth coordinates associated with anchors,

L_{cls} classification loss, N_{reg} is normalization parameter of regression, L_{reg} regression loss and λ is a constant value.

4) Region of Interest (ROI): The output generated from the RPN is the input of the ROI Pool layer. The output of RPN anchor boxes are different sizes, so the task of the ROI is to reduce the different size anchor boxes to the same or fixed size anchor boxes. For this purpose, we use classification and regression methods. The classification method identifies the object or background class of the image. Then regression gives the bounding box values (dx,dy,dh,dw) to cover the complete object. Where (dx,dy) is the center point of the bounding box, dh is the height of the bounding box and dw is the bounding box width. The performance and accuracy of the Faster R-CNN is good enough than all of the available traditional object detection algorithms. The framework of the Faster R-CNN is shown in Fig. 6. We have trained Faster R-CNN and 5000 annotated images like a person, coffee cup, coffee maker, bread, jam and spoon. The Faster R-CNN gives the improved detection accuracy effectively.

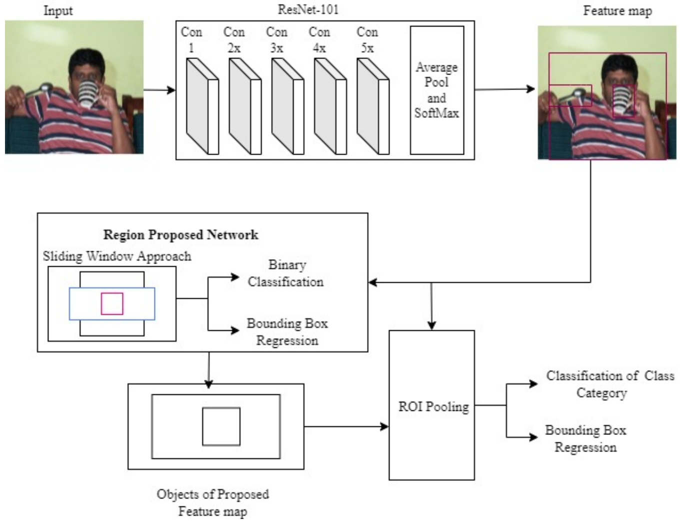


Fig. 6. Faster RCNN Complete Framework.

B. Multiple Object Track Handling of Feature Frames

We proposed a multi-object tracking method using the modified deep sort technique. The modified method takes the input as detected bounding boxes from Faster R-CNN. We are using the Kalman filter to extract the spatial and tracking information of the bounding boxes. The deep CNN model helps to extract the appearance feature of the frame. To find a track association between current and next frame of appearance feature using Mahabolish distance and improved sqrt-cosine similarity measures. If the track association's threshold value is equal to 1, the track is confirmed and updated; otherwise, delete the frame immediately. The modified deep sort is shown in Fig. 7. The frame-by-frame data association method and the Kalman filter are the essential components of the modified deep sort. The trackers scenario is based on the multidimensional state space(u,v,Y,h,a,b,c,d) that contains the center of bounding box is (u,v), expectation ratio is Y, height is h and the respective velocity coordinator are (a,b,c,d). The entire count of each track frame since the last successful measurement association of Tn is given by n for each track. The Kalman channel predictions occurred, the counter is incremented, or when the track has been assigned with the previous list, the counter is reset to 0. Suppose a new track prediction is started for each detection that cannot be assigned to any of the current lists. The first three frames of the new track are classified as tentative. These frames are kept for further processing when the association measurement is found at every timestamp, otherwise deleted.

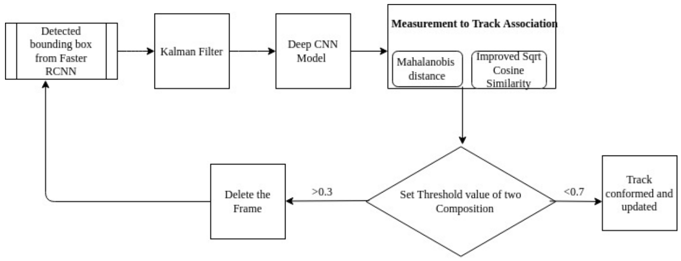

Detected bounding box from Faster RCNN
Kalman Filter
Deep CNN Model
Measurement to Track Association
Mahalanobis distance
Improved Sqrt Cosine Similarity
Set Threshold value of two Composition
>0.3
Delete the Frame
<0.7
Track conformed and updated

Fig. 7. Modified Deep Sort.

1) Association of the Current Frame and Predicted Frame: The Hungarian algorithm solves the measurement to track the association between the predicted Kalman state and the next frame. By creating two relevant measures, we were able to combine motion and appearance data. We utilized Mahalanobis Distance between the predicted frame and the next frame to get motion information.

$$d^{(1)}(p,q) = (d_p - y_p)^T S_i^{-1}(d_q - y_p) \qquad (5)$$

Equation 5 denotes the projection of the pth track distribution into measurement space (yp, Sp), and the detection of the qth bounding box by dq. The Mahalanobis Distance removes the state estimation uncertainty between the protected and newly arrived state mean track location. Further, it is possible to provide false Association by the threshold value at 95% confidence interval computed from the inverse distribution we denote

$$b^{(1)}_{(p,q)} = 1[d^{(1)}(p,q) \le t^{(1)}] \qquad (6)$$

Mahalanobis Distance is suitable only when motion uncertainty is less for the association metric. Our image space Kalman filtering framework provides the approximate value of the predicted object location. The Mahalanobis Distance is an uninformed metric for tracking in occlusion when the rapid displacement of the image plane. Therefore we integrated the second metric for tracking each bounding box. With the help of a protected CNN to extract the appearance feature of the bounding boxes, the architecture of the appearance feature network is shown in Fig. 8. To detect the space of the pth and qth track proposes most of the authors used cosine similarity. The cosine similarity is derived from the Euclidean distance. However, the Euclidean distance is not good for dealing with the probability-based approach. Zhu *et al.* [29] proposed the Sqrt cosine similarity. It is

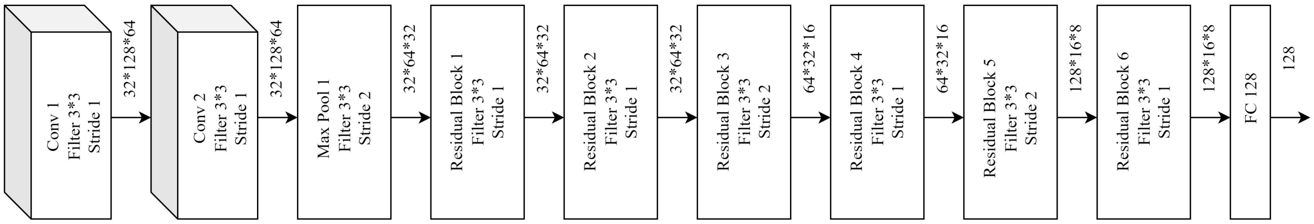


Fig. 8. Architecture of Appearance Feature Extractor Network.

derived from Hillinger distance, but there is conflict to define the similarity measures in some cases. Hence we adopted improved sqrt cosine similarity [30] to find the pth and qth track detection in appearance space.

$$d^{(2)}(p,q) = \frac{\sum_{i=1}^{T} \sqrt{p_i q_i}}{\sqrt{\sum_{i=1}^{T} p_i} \sqrt{\sum_{i=1}^{T} q_i}} \tag{7}$$

The binary variable is used to indicate if the association is good according to this metric.

$$b^{(2)}_{(p,q)} = 1[d^{(2)(p,q)} \leq t^{(2)}] \tag{8}$$

By addressing different aspects of the association method in a combination of both matrices. Mahalanobis Distance gives the information of the object position based on the motion for short-term prediction. The improved sqrt cosine distance is used to recover identities after long-term occlusion. Based on the Association problem, we combine what the metric is using a weighted sum.

$$C_{(p,q)} = \lambda d^{(1)}(p,q) + (1-\lambda) d^{(2)}(p,q) \tag{9}$$

Where we call the association appearance if it is within the range of both metrics.

$$b(p,q) = \prod_{i=1}^{2} b^{(i)}_{(p,q)} \tag{10}$$

In the equation 10, value is 1 it indicates both the metrics are equal, if zero metrics are not equal. It also indicates (p,q) is a true match between appearance and spatial Information. video sequence the next new frame detection is effectively associated with the present track OK then the track is continued as long as It is successfully associated and that track is confirmed and tracking update else deleted immediately.

IV. Evaluations and Results

The proposed system used a self-generated dataset to perform the multi-object detection and tracking. We use Google collaboratory for evaluation and python programming language and Detectron 2 for the experimental setup.

A. Dataset

To train the detection algorithm, we used a self-generated dataset containing 5000 pictures of people, jam, bread, spoons, coffee cups, and coffee makers. The dataset images are collected from different weather conditions, different viewpoints, different lighting of day and nights, blurring of images, crowded places and malls. Some images are collected from surveillance cameras and the internet. After collecting the images, we annotated images in different classes using the Labelimg tool. We classify images into 6 different classes: person, jam, bread, spoon, coffee cup, and coffee maker shown in Table I. The LabelImg tool after annotating it generates the XML files after we convert them into JSON format.

TABLE I. Proposed Dataset Classes

Classes	Number of Instances	Number of Images
Person	2100	1000
Bread	878	700
Jam	900	750
Spoon	1212	800
Coffee Maker	1278	900
Coffee Cup	1211	850

B. MOT Benchmark Dataset

We have evaluated our proposed system in the MOT benchmark dataset [35]. This dataset contains a combination of 21 different datasets. The dataset has contained 645 second video and it is the combination of 21 different sequences with proper annotation.

C. Results

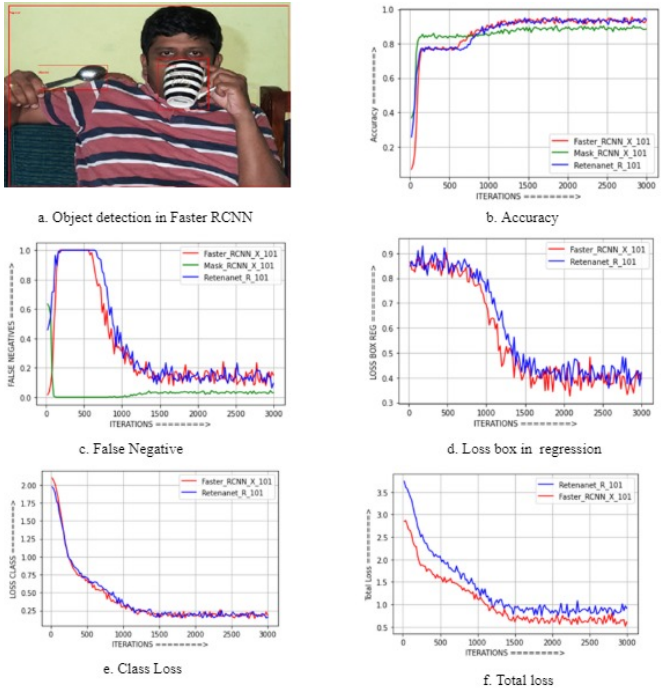
The proposed system has two stages: stage 1, Multi-object detection and stage 2, Multi-object tracking. Multi-object detection purpose we compared different Deep learning based object detection methods based on the accuracy and loss. After evaluating, we discovered that ResNet-101 based Faster RCNN is giving better performance. The evaluating different detection algorithm results are shown in Fig. 9. In Fig. 9(a) shows the multiple object detection output. (b) Gives an accuracy comparison of the Faster RCNN, Mask RCNN and Retinanet detection methods. Compared to all Faster RCNN gives better accuracy. (c) Shows the False negative results of Faster RCNN, Mask RCNN and Retinanet detection methods. Mask RCNN gives less False negative but Faster RCNN also gives better Results. (d) Shows the regression time box loss of Retinanet and Faster RCNN, (e) Gives the class name loss and (f) Gives a comparison of the total loss of Retinanet and Faster RCNN. Faster RCNN gives less detection loss compared to Retinanet and Faster RCNN. Compared to different detection methods, Faster RCNN gives good Accuracy and less detection

TABLE II. EVALUATED TRACKING RESULTS OF MOT BEHCHMARK DATASET

		MOTA	MOTP	MT	ML	ID	FM	FP	FN	RunTime
Batch	KBNT [31]	68.2	79.4	41.00%	19.00%	933	1093	11479	45605	0.7Hz
	LMPp [2]	71	80.2	46.90%	21.90%	434	587	7880	44564	0.5Hz
	MCMOT HDM [32]	62.4	78.3	31.50%	24.20%	1394	1318	9855	57257	35Hz
	NOMTwSDP16 [33]	62.2	79.6	32.50%	31.10%	406	642	5119	63352	3Hz
Online	EAMITT [34]	52.5	78.8	19.00%	34.90%	910	1321	4407	81223	12Hz
	POI [31]	66.1	72.5	34.00%	20.80%	805	3093	5065	55914	10Hz
	SORT[27]	59.8	79.6	25.40%	22.70%	1423	1835	8698	63245	60Hz
	DEEP SORT[28]	61.4	79.1	32.80%	18.20%	781	2008	12853	56668	40Hz
	Proposed System	71.2	80.1	33.40%	17.90%	825	1225	4115	54724	41Hz

loss. Hence we used Faster RCNN for detection purposes. It has variable performance on different classes of Self-Generated Dataset.

We proposed a Multi-object tracking method based on the appearance and motion features. Firstly we trained the dataset using the Faster RCNN and then we run modified deep sort on the same evaluation dataset. Fig. 10 shows the tracking of the different frames with identity numbers.



a. Object detection in Faster RCNN

b. Accuracy

c. False Negative

d. Loss box in regression

e. Class Loss

f. Total loss

Fig. 9. Object Detection Output and Comparision Graphs of Different Deep Learning Detection Methods.

In Table II shows the results of our proposed system's assessment on the MOT dataset. We used an object detection threshold of 0.8and further fine-tuned it with the different parameters to produce an efficient model. The following parameters are used for evaluation.

Multi-Object Tracking Accuracy (MOTA) - Summarizes the tracking accuracy terms of identity switches, false negative and false positive.

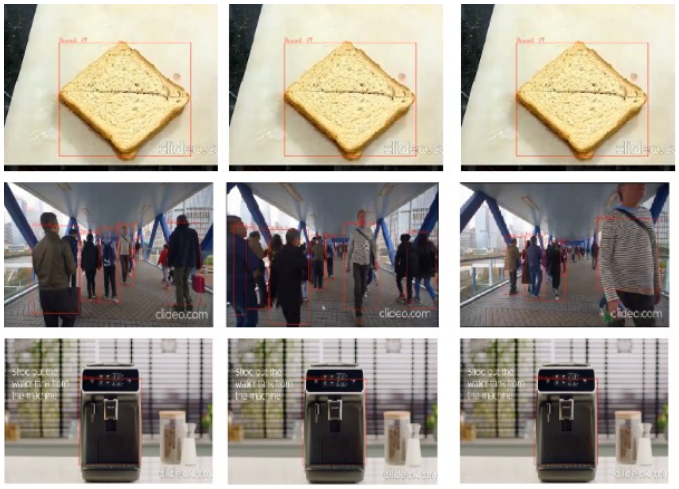


Fig. 10. Tracking with Unique Identity Number.

Multi-Object Tracking Precision (MOTP) - Provides the tracking precision for bounding boxes average dissimilarity between ground-truth value and predicted location.

Mostly Tracked (MT) - The percentage of ground-truth tracks has had the same label for at least 80% of their life span.

Mostly Lost (ML) - The proportion of ground-truth tracks that are tracked for at minimum 20% of their lifespan.

Identity Switches (ID) - It gives the number of times identity number changes the ground truth track.

Fragmentation (FM)- Identifies the number of times tracks have been interrupted due to missed detection.

Table II shows the results of our proposed work. It increases accuracy from 61.4 to 71.2 and also reduces the fragmentation problem. At the same time, identity switches increase slightly due to occlusion. We have seen a significant increase in the mostly tracked object. Hence our proposed model is suitable for online tracking.

Existing techniques were not efficient in reducing identity switching, fragmentation, and accuracy. However, using the modified deep sort technique it is possible to reduce identity switching and fragmentation reasonably better. Proposed research work contribute in reducing false identification due to the presence of frequent identity switching and fragmentation. It helps to track the multi objects in real time environment with better accuracy.

V. Conclusion

In this paper, we presented an improved multi-object detection and tracking technique that will reduce identity switches and fragmentation in the presence of occlusions. The proposed technique employed the Faster RCNN method to detect multiple objects and to overcome occlusion challenges. We also presented a modified deep sorting technique for multi-object tracking to reduce identity switch and fragmentation issues. The technique is simulated on a series of experiments using real and synthetic datasets has yielded better accuracy, reduced identity switching, and fragmentation compared to existing techniques. The specified objectives are achieved in this work. Improvement of performance in dark environmental conditions is yet to be investigated.

References

[1] R. W. Storm and Z. W. Pylyshyn, "Tracking multiple independent targets: Evidence for a parallel tracking mechanism," *Spatial Vision*, vol. 3, no. 3, pp. 179–197, 1988.

[2] M. Keuper, S. Tang, Y. Zhongjie, B. Andres, T. Brox, and B. Schiele, "A Multi-cut Formulation for Joint Segmentation and Tracking of Multiple Objects," 2016. [Online]. Available: http://arxiv.org/abs/1607.06317

[3] G. Khan, Z. Tariq, M. U. G. Khan, P. Mazzeo, S. Ramakrishnan, and P. Spagnolo, "Multi-person tracking based on faster r-cnn and deep appearance features," 2019.

[4] S. H. Rezatofighi, A. Milan, Z. Zhang, Q. Shi, A. Dick, and I. Reid, "Joint probabilistic data association revisited," *Proceedings of the IEEE International Conference on Computer Vision*, vol. 2015 Inter, pp. 3047–3055, 2015.

[5] A. Milan, K. Schindler, and S. Roth, "Detection- and trajectory-level exclusion in multiple object tracking," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 3682–3689, 2013.

[6] A. K. Jain, Z. Yu, and S. Lakshmanan, "Object matching using deformable templates," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 3, pp. 267–278, 1996.

[7] J. L. Mundy, "Object Recognition in the Geometric Era: A Retrospective," pp. 3–28, 2006.

[8] T. Ojala, M. Pietikäinen, and T. Mäenpää, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 971–987, 2002.

[9] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," vol. 1, pp. 886–893, 2005.

[10] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[11] O. Tuzel, F. Porikli, and P. Meer, "Region covariance: A fast descriptor for detection and classification," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 3952 LNCS, pp. 589–600, 2006.

[12] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, vol. 25, pp. 1097–1105, 2012.

[13] G. Khan, M. U. Ghani, A. Siddiqi, Z. ur Rehman, S. Seo, S. W. Baik, and I. Mehmood, "Egocentric visual scene description based on human-object interaction and deep spatial relations among objects," *Multimedia Tools and Applications*, vol. 79, no. 23-24, pp. 15 859–15 880, 2020.

[14] R. Girshick, J. Donahue, T. Darrell, J. Malik, U. C. Berkeley, and J. Malik, "1043.0690," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1, p. 5000, 2014. [Online]. Available: http://arxiv.

[15] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 9, pp. 1904–1916, 2015.

[16] R. Girshick, "Fast R-CNN," *Proceedings of the IEEE International Conference on Computer Vision*, vol. 2015 Inter, pp. 1440–1448, 2015.

[17] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *Advances in neural information processing systems*, vol. 28, pp. 91–99, 2015.

[18] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," pp. 779–788, 2016.

[19] C. Dicle, O. I. Camps, and M. Sznaier, "The way they move: Tracking multiple targets with similar appearance," *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2304–2311, 2013.

[20] A. Bewley, L. Ott, F. Ramos, and B. Upcroft, "Alextrac: Affinity learning by exploring temporal reinforcement within association chains," *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 2016-June, pp. 2212–2218, 2016.

[21] J. H. Yoon, M.-H. Yang, J. Lim, and K.-J. Yoon, "Bayesian multi-object tracking using motion context from multiple objects," pp. 33–40, 2015.

[22] D. Reid, "An algorithm for tracking multiple targets," *IEEE transactions on Automatic Control*, vol. 24, no. 6, pp. 843–854, 1979.

[23] T. E. Fortmann, Y. Bar-Shalom, and M. Scheffe, "Sonar Tracking of Multiple Targets Using Joint Probabilistic Data Association," *IEEE Journal of Oceanic Engineering*, vol. 8, no. 3, pp. 173–184, 1983.

[24] G. Kayumbi, P. L. Mazzeo, P. Spagnolo, M. Taj, and A. Cavallaro, "Distributed visual sensing for virtual top-view trajectory generation in football videos," pp. 535–542, 2008.

[25] M. Yang and Y. Jia, "Temporal dynamic appearance modeling for online multi-person tracking," *Computer Vision and Image Understanding*, vol. 153, pp. 16–28, 2016.

[26] Y. Xiang, A. Alahi, and S. Savarese, "Learning to track: Online multi-object tracking by decision making," *Proceedings of the IEEE International Conference on Computer Vision*, vol. 2015 Inter, pp. 4705–4713, 2015.

[27] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft, "Simple online and realtime tracking," *Proceedings - International Conference on Image Processing, ICIP*, vol. 2016-Augus, pp. 3464–3468, 2016.

[28] N. Wojke, A. Bewley, and D. Paulus, "Simple online and realtime tracking with a deep association metric," *Proceedings - International Conference on Image Processing, ICIP*, vol. 2017-Septe, pp. 3645–3649, 2018.

[29] S. Zhu, L. Liu, and Y. Wang, "Information retrieval using Hellinger distance and sqrt-cos similarity," *ICCSE 2012 - Proceedings of 2012 7th International Conference on Computer Science and Education*, no. Iccse, pp. 925–929, 2012.

[30] S. Sohangir and D. Wang, "Improved sqrt-cosine similarity measurement," *Journal of Big Data*, vol. 4, no. 1, 2017.

[31] F. Yu, W. Li, Q. Li, Y. Liu, X. Shi, and J. Yan, "POI: Multiple object tracking with high performance detection and appearance feature," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9914 LNCS, pp. 36–42, 2016.

[32] B. Lee, E. Erdenee, S. Jin, M. Y. Nam, Y. G. Jung, and P. K. Rhee, "Multi-class multi-object tracking using changing point detection," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9914 LNCS, no. Mcmc, pp. 68–83, 2016.

[33] W. Choi, "Near-online multi-target tracking with aggregated local flow descriptor," *Proceedings of the IEEE International Conference on Computer Vision*, vol. 2015 Inter, pp. 3029–3037, 2015.

[34] R. Sanchez-Matilla, F. Poiesi, and A. Cavallaro, "Online multi-target tracking with strong and weak detections," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9914 LNCS, pp. 84–99, 2016.

[35] L. Leal-Taixé, A. Milan, I. Reid, S. Roth, and K. Schindler, "MOTChallenge 2015: Towards a Benchmark for Multi-Target Tracking," pp. 1–15, 2015. [Online]. Available: http://arxiv.org/abs/1504.01942