A Novel Technique to Detect and Track Multiple Objects in Dynamic Video Surveillance Systems

M. Adimoolam[1], Senthilkumar Mohan [2], John A.[3], Gautam Srivastava[4,5]*

[1] Department of Computer Science and Engineering, Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences, Chennai (India)
[2] School of Information Technology and Engineering, Vellore Institute of Technology, Vellore (India) 3 School of Computer Science and Engineering, Galgotias University, Greater Noida (India)
[4] Department of Mathematics and Computer Science, Brandon University, Brandon, MB R7A 6A9 (Canada)
[5] Research Center for Interneural Computing, China Medical University, Taichung 40402 (Taiwan)

Received 14 April 2020 | Accepted 8 October 2021 | Published 18 January 2022

Abstract

Video surveillance is one of the important state of the art systems to be utilized in order to monitor different areas of modern society surveillance like the general public surveillance system, city traffic monitoring system, and forest monitoring system. Hence, surveillance systems have become especially relevant in the digital era. The needs of the video surveillance systems and its video analytics have become inevitable due to an increase in crimes and unethical behavior. Thus enabling the tracking of individuals object in video surveillance is an essential part of modern society. With the advent of video surveillance, performance measures for such surveillance also need to be improved to keep up with the ever increasing crime rates. So far, many methodologies relating to video surveillance have been introduced ranging from single object detection with a single or multiple cameras to multiple object detection using single or multiple cameras. Despite this, performance benchmarks and metrics need further improvements. While mechanisms exist for single or multiple object detection and prediction on videos or images, none can meet the criteria of detection and tracking of multiple objects in static as well as dynamic environments. Thus, real-world multiple object detection and prediction systems need to be introduced that are both accurate as well as fast and can also be adopted in static and dynamic environments. This paper introduces the Densely Feature selection Convolutional neural Network – Hyper Parameter tuning (DFCN-HP) and it is a hybrid protocol with faster prediction time and high accuracy levels. The proposed system has successfully tracked multiple objects from multiple channels and is a combination of dense block, feature selection, background subtraction and Bayesian methods. The results of the experiment conducted demonstrated an accuracy of 98% and 1.11 prediction time and these results have also been compared with existing methods such as Kalman Filtering (KF) and Deep Neural Network (DNN).

Keywords

Convolutional Neural Network, Machine Learning, Object Detection, Video Surveillance.

DOI: 10.9781/ijimai.2022.01.002

I. Introduction

The human visual system detects and recognizes objects within dense groups of multiple objects very efficiently. But this task proves to be difficult and is riddled with challenges when it comes to artificial systems. Modern surveillance systems have been used in public civil monitoring by implementing object detection and motion tracking. Object detection is a subsidiary topic under the field of Computer Vision which is a study of how computers detect and classify different types of objects in an image or a video. There are numerous applications of such systems in the modern world that detect and track objects in a region such as surveillance systems for military use, modernized traffic control systems, public weather observation systems, etc. [1], [2], [3]. Researchers are working on various techniques to increase the speed and overall accuracy of such object recognition and tracking. Recent advancements in the field of information technology have increased the need for more robust and intelligent surveillance systems with better speed and accuracy. Therefore, object detection and tracking have shown great potential while emerging as an important technology in the field of surveillance related to security.

Unlike humans, computers see images as several clusters. Each pixel in an image contains data corresponding to the colour values i.e. red, green and blue. If an image contains all three colour values then there are three channels present in that image. A grayscale image contains only one channel [4]. Determining the location of an object and the region of interest is a challenging task in the field of computer vision. In general, two methods are used for determining the location of the object and they are object detection and object

* Corresponding author.
E-mail address: srivastavag@brandonu.ca

tracking. To locate an instance of the object in images or videos, the object detection technique is used. The popular Convolutional Neural Networks (CNN) [5] was trained using a large set of labelled data and it is used to detect a region of interest within a completely new given image. Hence using this method one can determine the location of an object within a given image, whereas in the case of object tracking only the pixel information of the region of interest is provided and the region having the highest amount of similarity is searched.

During the process of object detection, objects are detected based on various points such as objects of interest, face, colour, shape, and skin. The process involves the extraction of frames from an image or video. Subsequently, various such features of objects are extracted for video surveillance systems and this is discussed in detail [6]. The detected object is then continuously tracked in the input video stream. Numerous factors make it difficult to track objects after their detection. Several times, the object is occluded by its surroundings which makes it difficult for the tracking algorithm to track the object in real-time. Additionally, sudden movements which lead to changes in the shape or size of the object or changes in the observed scene are a few of the factors that can affect object-tracking. Researchers are continuously working on improving these algorithms for the better tracking of objects despite the above-mentioned hindrances that affect the procedure.

Real-time object tracking algorithms are being studied where the detector learns about all the changes in the object and its environment and uses it to better track the object. Some of the popular object detectors are Region-based CNN (R-CNN), Faster R-CNN, Single Shot Detectors and You Only Look Once (YOLO). Among these object-detectors, Faster R-CNN and Single Shot Detectors have greater accuracy, while YOLO has better speed. In this paper, we look at the benefits and drawbacks of two-stage detectors and single-stage detectors. Moreover, we explore how to improve the speed and accuracy of modern security surveillance systems,. Furthermore we fine-tune object detection with direct comparison to state-of-the-art detecting techniques.

The topics discussed in this research paper in the subsequent sections are as follows. Section II discusses current and related work on object detection methods, single-stage and multistage as well as multi-object target methods. Section III delves into the working methodologies and the multi-object detection approaches in DFCN-HP. Section IV contains the implementation and performance analysis of the proposed work along with the existing works followed by the conclusion.

II. Related Works

This section discusses the technology related to object detection (detection of the object in the image), classification (classifying objects into different categories such as dog, cat, person), and tracking of the objects. The main body of work done related to object detection, tracking and classifications are the standard methods for the general one-stage object detection and multistage object detection which are expanded further.

A. Object Detection Methods

Various algorithms have been developed to detect objects in images or videos. The goal of every object detection algorithm is to improve the overall accuracy by improving the confidence level of the object detector while minimizing the time taken to detect the object in the image or video. One stage detector and two-state detector are the two key object detection algorithms in use today and they are extensively used in surveillance object detection. Fig. 1 shows the processes involved in an object detection flow. The surveillance image or video has to be pre-processed to detect the object and subsequently, the detected object has to undergo a feature extraction process.

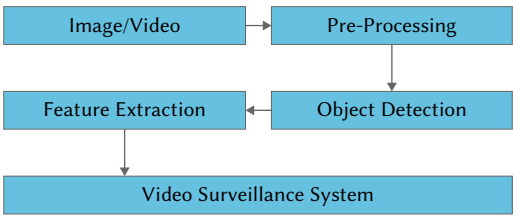

Fig. 1. Object Detection processes.

B. Single-Stage Detectors

The separate region proposal steps are not applied in single-state object detector algorithms. Instead, they consider every position on the image as a potential object and then try to classify each region of interest as an object or background. A few of the popular single-stage object detectors are discussed below.

YOLO: Joseph Redmon along with Ross Girshick and others proposed a new approach for object detection in which they framed object detection as a regression problem. This method divided the entire image into spatially separated bounding boxes with class probabilities associated with them. It could be an optimized end-to-end process since the whole framework was a single network. YOLO gained a remarkably faster speed than its predecessors by processing images in real-time at 45 frames per second [5].

Single-Shot Multibox Detector (SSD): W. Lue and others proposed a single-stage detector (2016) that used a single DNN for detecting objects in images. The output of space-bounding boxes for each grid cell was created to discretize after dividing the images into a grid cell. It has further been trained straightforwardly using SSD. SSD achieved 74.3% mAP(mean average precision) for the input size 300X300 using Visual Object Classes Challenge (VOC) 2007 at 59 frames per second [6].

YOLOv3: Joseph Redmon and Ali Farhadi (2018) proposed an improved version of YOLOv2, YOLOv3. They introduced a few design changes in YOLOv2 to make it better. The SSD runs 3 times faster for the input 320X320 using YOLOv3 and achieved 28.2 mAP just in 22ms [7].

C. Two-Stage Detectors

Two-stage detectors divide the detection of the object into two stages: in stage one, it identifies the subsets of the image that might contain an object (region proposal); and in-state two, it classifies the object for making predictions within the proposed region. The detector identifies the subset of the image which may potentially contain an object during the first state of two-stage detectors. This is done so that every object inside an image can belong to one of the proposed regions. The deep learning model has been applied further in these objects and labels are assigned based on object category and this is called the second stage of two-stage detectors. CNN before R-CNN mainly used a sliding window to generate regions individually with CNN classifiers to produce a set of probabilities. A general region-based CNN has the same approach but instead of selecting a huge number of regions to examine, this independently generates about 2000 regions of interest.

A few of the popular two-stage detectors are discussed below. Ross Girshick et al. proposed a novel two-stage detector using R-CNN (2014) [8]. When compared to the state-of-the-art traditional detectors (40.4% mAP), it obtained a 53.7% mAP performance and it significantly improved overall detection using R-CNN [9]. R-CNN involves three sequences in its pipeline: (i) proposal generation: to find regions in the image that might contain an object, and these regions are called region proposals, (ii) feature extraction: to extract all the CNN features from the region proposals, (iii) region classification: to classify all the objects using the extracted features.

Fast R-CNN [10]: This method is called a multi-task learning detector introduced by Ross Grishick et al. that overruns the R-CNN and also SPP-net [11]. It has R-CNN with ROI (Region of Interest) pooling layer to extract the feature of the region. The Fast R-CNN obtains an accuracy that is better than R-CNN and SPP-net and is of major significance. Fig. 2 shows the representation of Fast R-CNN objects.

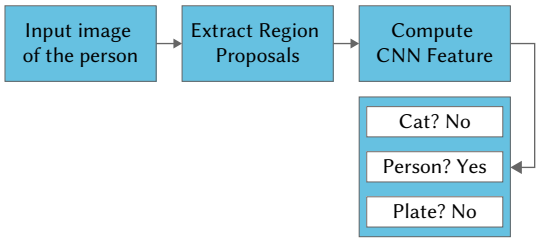

Fig. 2. R-CNN.

Faster R-CNN [12]: Subsequently, Ross Girshick and others proposed the state-of-the-art version of the R-CNN family in 2015. Here, the region proposals have been generated by Region Proposal Network (RPN) in Faster R-CNN. This method generates region proposals directly into the network instead of using an external algorithm. The frame rate of 5 fps for the Very Deep Convolutional Networks (VGG-16 model) [13] has been gained in Faster R-CNN. This performance is a remarkable achievement with an object detection accuracy of mAP 73.2% and 70.4% using Faster R-CNN's object proposal and this mAP has produced coloration with PASCAL 2007 and 2012 respectively. Fig.3 shows the Faster R-CNN working mechanism along with its object components.

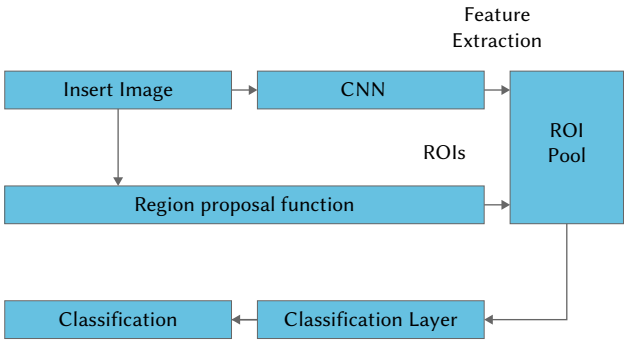

Fig. 3. Fast R-CNN.

Mask R-CNN [14]: The Faster R-CNN has generated a pixel-level mask of an object that has achieved state-of-the-art results. Further, the Mask R-CNN proposed by K. He, et al. has a branch for the prediction of an object which works parallel to the existing branch-box recognition and this parallel object prediction has proved to be more significant than the Faster R-CNN method and has outperformed it in all aspects concerning object detection and prediction.

D. Multi-Object Targets Methods

The geometric constraints method is used for target detection, recognition and tracking of objects using a distributed algorithm, [15]. This work is applied to different applications such as mobile cameras, multiple object detection in Multi-view, etc. The video surveillance techniques such as tracking and multiple object detection are discussed further. In this method, the Bayesian tracking multimodal framework is used without clearly associating object tracking and detection. It is observed to have an errorless performance, with missing detection problems also solved [16]. The real-time multiple objects tracked from the multiple camera surveillance systems were observed. In this work, object tracking and detection were performed using the feature selection parameter [17], with multi-object tracked from the multiple cameras put forward [18] from the surveillance of the video for which tracking and object detection was used [19]. This work uses the Pseudo motion algorithm, the Fourier shift theorem, and the two-stepped morphological operation is used to identify object properties such as region, size, etc. Here the Kalman Filtering (KF) is used for object tracking. The Local Maximal Occurrence Representation (LOMO) feature extraction algorithm is used to feature the representation of the objects and the Hankel matrix is used to manage the target objects, while the IHTLS algorithm is used to estimate the ranking of the objects. The real-time tracking of the objects from the multiple cameras was observed [20]. These works were applied to trace path tracking and trajectory finding using multiple cameras in different positions and multiple tracking of objects using the dual camera used to track it [21]. The geometry, homographic calibration was used for spatial mapping and a pan–tilt–zoom camera was used to detect the objects automatically. Multiple object detections continuously from the multiple cameras using single Target Track-Before-Detect, Particle Filter and predict and update methods are used to track the objects and were observed [22]. The video surveillance system computational cost for object detection was proposed by Rakesh Chandra Joshi et al. (2019) [23]. They use the Kalman Filter Assisted Occlusion Handling (KFAOH) technique for handling occasions. Table I shows the multi-targets tracking methods, with mention of the processing methods and algorithms, etc.

In the previous methods of object tracking, the detection was used for single or multiple objects from single and multiple cameras [35], [36], [37]. Here, it was proposed that multiple objects and multiple cameras be used to track and detect the objects automatically. Most of the works involved manual predictions with only the MGC algorithm having automatic tracking and detection. Hence, the work in question has multiple objects prediction from the multiple cameras atomically in different surveillance systems and from the sequence of videos. The work has also been compared with KF and DNN methods. Ahmad Jalal et al. (2017) [38] proposed a human activity recognition technique for a video surveillance system. The health care application of elder people monitoring was discussed in this work and used Hidden Markov Models (HMM) and robust multi-features model. This model recognized human activity in the experiment. Anahita Ghazvini et al. (2019) [39] discussed counting individuals in video surveillance as multiple object detection. The work used a Convolution Neural Network (CNN) to detect and count several objects in the surveillance video dataset.

III. The DFCN-HP

Multiple object detection from multiple cameras is called DFCN-HP. The DFCN-HP method consists of the following steps to detect multiple objects.

1. Pre-request information
2. Dense Block
3. Feature selection
4. Multiple Object detection and tracking
5. Hyperparameter Tuning
6. Data Acquisitions and Training

The first step is to pre-request information of multiple object detection from multiple cameras having detected multiple objects from multiple channels. Each channel has N objects and is selected. The channel and object combinations are shown in Fig. 4.

TABLE I. Multiple Object Target Tracking Methods

Algorithm	Features	Calibration	Multi-target	Limitation
KFAOH (Kalman Filter Assisted Occlusion Handling) (2019) [23]	Object detection	Manual	Yes	Only a single object detected with multiple cameras
GM (Graph matching) (2009) [24]	2D position, size, velocity	Manual	Yes	Only a single object detected with multiple cameras
CFI (Caratheodory-Fejer Interpolation) (2006) [25]	Pixels, manifold learning	Manual	Yes	Ambiguity occurs when detecting a single object with multiple cameras
GMPHD (Gaussian Mixture Probability Hypothesis Density filter) (2007) [26]	Position, size and colour histogram	Manual	Yes	Limited with object tracking infused video data
MGC (Minimum graph cut) (2009) [27]	Multiple planes occupancy map	Automatic	Yes	Plane view data object alone detected
VA (Viterbi algorithm) (2008) [28]	Colour and motion	Manual	Yes	Single object detection and probability is low
BT (Bayes tracker) (2008) [29]	Head position	Manual	Yes	Dense crowed single object detection and time is more to detect a single object
PF (Particle Filter) (2006) [30]	The vertical axis of the target, ground position	Manual	Yes	Guided particle filtering needed more dataset and time-consuming process
PF (Particle Filter) (2009) [31]	Signal intensity	Manual	Yes	More segmentation noise in object detection
NCA(neighbourhood components analysis) (2018) [32]	Posture change, Pedestrian tracking	Manual	Yes	Poor performance for low-quality videos object detection
KF(Kalman filtering) (2019) [33]	Multi-Object detection	Manual	Yes	Only Aerial Imagery data objects are detected
DNN (Deep Neural Network) (2019) [34]	Tracking multiple objects	Manual	Yes	A deep neural network takes more time to detect a blurred object

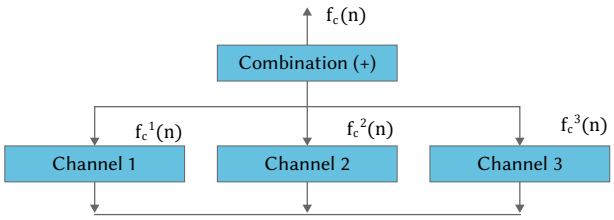

Fig. 4. Multiple channel combinations.

The n objects from each channel and the combination of objections are shown in Equation (1).

$$\mathrm{fc(n)} = \mathrm{fc^1(n)} + \mathrm{fc^2(n)} + \mathrm{fc^3(n)} + \cdots + \mathrm{fc^n(n)} \quad (1)$$

Let $D = \{D1, ..., Dk\}$ denote a set of k trained object boundary detectors of objects for a corresponding set of k situations $S = \{S1, ..., Sk\}$. Applying the j-th detector Dj to an image I give the boundary prediction $Dj\ (I)$. The final object detection equation is shown in Equation (2).

$$\mathrm{D(I)^k} = \sum_{\mathrm{j=1}}^{\mathrm{k}} \mathrm{P}\,(Sj(I))Dj(I) \quad (2)$$

Equation (2) denotes the sum of inter-product of probability between the set of various (k) situations of images and boundary detection of images.

The second step is a dense block and it is used to increase the prediction of the neural network. An important usage of the dense block is to increase the predictability of objects. Fig. 5 shows the dense block and Equation (3) represents the dense block target function in linear and nonlinear Equations.

$$\mathrm{f(x)} = \mathrm{f(\,w * y + b)} \quad (3)$$

The $f\,(x)$ is the target activation function and is used in the entire linear and non-linear prediction of the objects (y) concerning weight (w), and block (b).

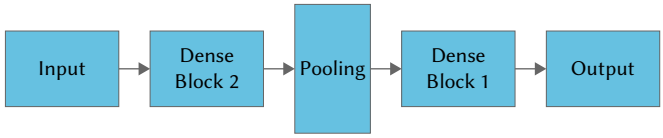

Fig. 5. Dense block.

The dense block connectivity between each layer receives the features of each input. The pooling layer operations are used to change the features from one layer to other layers between each block. Finally, each layer's features are concatenated for final operations. The concatenation operations of each layer's feature are shown in Equation (4).

$$\mathrm{C} = \mathrm{H}(c_1, c_2, c_3, \ldots, c_{n-1}) \quad (4)$$

C - Concatenation of layers, H - Histogram Values, M - Multiple Inputs, (c_1, c_2, c_3, ..., c_{n-1}) are the concatenations of the features.

The third step is object selection and tracking is based on the features of the objects. The important features of the object selection and tracking depend on the following parameters such as (s) similarity of the objects (a) appearance of the objects (c) structure of the objects. The similarity of the objects is measured based on the following Equation (5).

$$\mathrm{Sim(\,I, J)} = \mathrm{Wa.\,a(i,j)} + \mathrm{Ws\,.\,s(i,j)} + \mathrm{Wl.\,l(i,j)} + \mathrm{Wsd.\,Sd(i,j)} \quad (5)$$

Where $Sim(I, J)$ is the cost distance between the similarity of the objects, W is the weight of each attribute, a - appearance, s - structure, l - locations, sd - size difference. Equation (5) is used to manage each object in consecutive frames.

The appearance of an object is important to track and recognize the object continuously. The appearance depends on the viewpoint change, a correlation between objects (C), histogram (H) matching against the RGB, object orientation, transformation, keypoint features, etc. Equation (6) is used to match the appearance of the image for continuous detection and tracking in a dense block network from the sequence of frames. It reduces training time.

$$A(i,j) = \sum_{n}^{k} H * C(n,i)C(n.j) \tag{6}$$

Where k denotes the combined histogram data in memory from the past data, n denotes the number of frames detected from the tracking of frames. The structure of the features is another important clue to track the image and with it, the structure distance is also included. The structure distance is calculated based on the linear binary pattern [40] of the objects. The linear binary pattern captures the structure from the image. LPBH is used to recognize all the structures of the frames such as the face, nose, mouth, etc [41].

The fourth step is object detection and subtraction based on the shelf background subtraction method [42]. This shelf background method learns online information and undertakes foreground object subtraction and background information subtraction from the sequence of frames. The moving objects are detected from the frame regions pixel by pixel using Gaussians represented static sense. The fitting ellipse is used for foreground detection which combines the expectation from maximum methods used to estimate the number of the ellipse and the parameters [43] for the tracking of the objects traced using the Bayesian method. The general tracking of moving objects is represented in notation by Equation (7).

$$\mathrm{T} = [\mathrm{Tt}, \mathrm{n} | \mathrm{n} = 1, .., \mathrm{N}] \tag{7}$$

Where T is tracking, N is denoting the number of moving objects in each frame with time t. The n^{th} frame is denoted by Equation (8).

$$\mathrm{Tt}, \mathrm{n} = [\mathrm{P}, \mathrm{V}, \mathrm{E}] \tag{8}$$

Where P denotes object position, V denotes velocity, E denotes ellipse object position. Equation (9) represents the posterior probability density function to recursively measure the objects based on the current time slap.

$$\mathrm{P}(\mathrm{xt}|\mathrm{z1}:\mathrm{t}) = \mathrm{P}(\mathrm{zt}|\mathrm{xt})\,\mathrm{P}(\mathrm{xt}|\mathrm{z1}:\mathrm{t}-1)/\mathrm{P}(\mathrm{zt}|\mathrm{z1}:\mathrm{t}-1) \tag{9}$$

Where $P\,(xt\,|\,z1{:}\,t{-}1)$ denotes prior probability, $P\,(zt\,|\,xt)$ denotes likelihood, and $P\,(zt\,|\,z1{:}\,t{-}1)$ denotes normalization factors.

The fifth step is the optimization step using hyperparameter tuning to select a learning process. Before the learning process begins, the values of the hyperparameters are set. Tuning hyperparameters is often a difficult task and is used to train the dense block and their various parameters. Below there are a few of the hyperparameters that are considered. The optimization steps of hyperparameter turning are as following steps.

Step 1: An activation function introduces the non-linear functionality to our network. The activation function helps the neural network to understand something complicated and complex. The main purpose of the activation function is to change the input signal of the sequence of the frame into the output signal.

Step 2: The learning rate controls the rate of learning for each batch of iteration.

Step 3: The Number of Epochs is the number of times training sets are passed into the dense neural network.

Step 4: The batch size parameter denotes the size of the batches that are used during the training process. Mini batch size or frames is preferable in the training process.

Step 5: Step 5 is the backbone for the pre-processing of the dense block of the network.

Step 6: To train the maximum number of regions of interest.

Step 7: This step is the validation of every epoch in the training steps. If the value of the validation steps increases, then the accuracy of validation states will improve, but it will slow down the training.

Step 8: The confidence threshold step is determining how confident it can allow the correct detections to be. It will filter out the non-confident findings by the dense block. This threshold can increase its value to generate more proposals.

The sixth step is data acquisition. Training is used to train the multiple objects and the training data is an acquisition from the huge amount of data. The entire neural network is trained using a stochastic gradient descent using a dense batch size of 64 for 400 and 50 epochs, correspondingly. The starting learning rate is 0.1 divided by 20 at 60% and 85% of training epochs. The dense block network train models 90 epochs using 256 batch sizes. The learning rate start is set to 0.1 and lowered by 10. The graphic processor memory constrains the trained data to a mini-size batch of 156. To compensate for the small batches of frames, we increase the model for 100 epochs and divide the rate by 20 at 90 epoch. Based on the training, the objects are detected from a huge number of datasets.

Finally, the object tracking and detection of the entire process are shown in Algorithm 1.

Algorithm 1: Multiple object detection and tracking - DFCN-HP
Input: Sequence of frames from multiple cameras
Output: predicted objects and tracking
Initial: Capture the frames from the multiple cameras
Begin
If the objects are selected from the multiple frames
Combine the objects for tracking (Equation -1)
For each frame of multiple sources
    Use the features
    Predict the objects
    Add objects from multiple frames
End For
End If
For each
    The similarities of the objects are tracked (Equation -6)
    The appearance of the objects are tracked using various parameters (Equation -7)
    Objects are tracked using the Bayesian method (Equation 8)
End For
If Objects recursively traced on current time slap (Equation 10)
Return Optimal value
Performing hyperparameter tuning
Else
Continuously trained and Acquisition
End

IV. Implementation

The real-time datasets and CIFAR datasets [44] are used for implementations. The real-time datasets are captured from the multiple cameras for the implementation shown in Fig. 6.

Table II gives the configuration details used in image training and, based on the training, the DFCN-HP provides the results. The testing of the data is also associated with real-time and CIFAR datasets. Before testing the real-time and CIFAR datasets, its samples are fine-tuned. The fine-tuning does not increase the object tracking performance. The validation tests of a set of frames are used to verify the validity of the frames of objects. Accordingly, the static and dynamic objects are trained continuously using a dense block model.

Fig. 6. Set of images used for results.

TABLE II. Configurations Details for Training of DFCN-HP

Backbone of network	Dense block
Backbone Strides	16, 32, 64
Batch Size	64
The detection of max Instances	100
The detection of Min Confidence	0.9
Detection dense block Threshold	0.3
Frames per Graphics Processor Unit	1
Image Shape	[1024, 1024, 3]
Learning Momentum	0.9
Learning Rate	0.1
Min size of the image	156
Threshold	0.5
Max size of the image	256
Steps Per Epoch	1000
Train ROIs Per Image	200
Validation Steps	50

Most of the previous works used the small size of samples and it is difficult for detection and tracking. But using DFCN-HP, medium-size objects with fine-tuned objects are detected and tracked. Before tuning and after tuning results are shown in Table III. The proposed work of DFCN-HP consists of 50 sequences of frames which are grouped into 25 sequences of frames for training and 25 sequences of frames for testing. The targeted public real-time sequence of frames datasets from the different cameras are different in the following parameters such as viewpoint, camera motion, object density, target motion, object motion direction and objects movement direction, etc.

TABLE III. Parameters Details Before and After Tuning

Parameters	Before tuning	After tuning
Train Anchors Per Image	256	32
Detection Min Confidence	0.9	0.8
Learning Rate	0.1	0.01
Weight Decay	0.0001	0.1
RPN NMS Threshold	0.5	0.7

The implementation of DFCN-HP is similar to Mask R-CNN with ResNet-101 [19] and YOLOv3 for object tracking and detections from the sequence of frames. The fine-tuning of the process for the entire proposed work is according to the DFCN-HP needs and flow of the work. Here, it was run as a sequence of images similar to Mask R-CNN with ResNet-101 and YOLOv3, although DFCN-HP speed was taken into consideration. Based on the proposed work the object detection and tracking are shown in Fig. 7. For security surveillance systems, accuracy and details are far more important than is the case the field of security.

Fig. 7. Objects detection and tracking.

A. Improvement Via Tuning

While working with DFCN-HP, it was observed that reducing the batch size decreased the overall training time. Consequently, reducing the learning rate increased the confidence of predictions. With a minimum confidence threshold set to 0.7, our dense neural network identified a laptop with confidence above 95% but it also identified the display of the laptop as a "tv" with a confidence of 95%. After increasing the confidence threshold to 0.9, our DFCN-HP ignored the regions of the image of "tv" and only predicted the laptop. Using this example, similar object predictions also increased in the DFCN-HP.

In each trial of tuning more hyperparameters, the results were checked and the new results were compared to the old ones. All the tests were performed on a macOS with 8GB of RAM. The dense block per Image was reduced from 256 to 32. To increase the number of proposals, the value of Detection was decreased to a Min Confidence from 0.9 to 0.8 which enabled the detector to predict regions with lower confidence. The increased learning rate from 0.001 to 0.01 was to speed up the learning process. This helped a lot in our test runs. In weight decay, the weights were multiplied by a number slightly less than 1 to prevent the weights from growing too large. This changed weight decay from 0.0001 to 0.1 and worked well. Finally, the value of the dense block threshold was increased from 0.5 to 0.7 to generate more proposals. Many tests were performed on different hyperparameters to pick a certain value that was best for our detector.

Changing the backbone from *one dense block to another dense block* improves the accuracy and speed to a great extent. It was decided to stick to a *dense block* after performing some tests[47]. Here, two tests were performed on each image, one that did not change the hyperparameters and another that tuned the hyperparameters. It was observed that in most of the cases our detector performed well with high confidence and more proposals. The image in Fig. 7 and the chart in Table III and IV clearly show improvements in the detector.

Fig. 7 represents a real-time surveillance system using multiple sequences of frames. Each frame indicates that every object is

validated and compared to the trained dataset. If any new objects have been detected in the frames, the prediction and tracking take place.

The detector performs well with Multiple Object Tracking Accuracy (MOTA) for 98% of people, backpacks and handbags. In this, the single iterations for single object predictions rates are 98.187% for people, 97.719% for backpacks and 96.138% for handbags in public places. The Mostly traceable Object (MTO) rate is 99.2%. Overall, various comparison parameters [48] are IDP (ID precision), IDR (ID recall), IDF1 (ID F-score), MOTA-Multiple Object Tracking Accuracy, MOTP - Multiple Object Tracking Precision, Rcll -Recall, Prcn - Precision, MTO-Mostly tractable Object, ML -Mostly Lost Object. The predicted and traced results of DFCN-HP using real-time data are shown in Table IV.

TABLE IV. Results Ff DFCN-HP Prediction Using Real-Time Data

Method	IDF1	IDP	IDR	MOTA	MOTP	RcLL	Prcn	MTO	ML
RNN	-	-	-	82.9	80.3	92.3	95.3	85	15.2
KF	90	90	90	96.4	90.6	98.2	98.2	95.5	3.6
DNN	90.5	90.3	90.6	97.5	88.5	99	98.7	98.9	0
Proposed method	92.5	93.8	94	98	90.8	99.5	99.2	99.2	0

B. Performance and Comparison

The performance of our proposed method DFCN-HP is compared to previous works such as those based on KF [33], DNN [34] and Recurrent Neural Networks [45],[46]. The proposed work outputs are associated with all the detection and tracking scores. The proposed work obtains a recall (RcLL) and precision (Prcn) of 99.5% and 99.2%, respectively, and the prediction results are high when compared with those of KF and DNN methods. The tuning prediction is observed to increase in every frame in the image. The prediction performance is very high. The detection of false-positive alignment is 98.5% and negative detection tracking in the sample is 1.5%. Multiple object detection and distributions are shown in Fig. 8. The results show the prediction distributions of persons, backside bags and handbags, which are 99%, 98.4%, 98% respectively. As shown in Fig. 8, the person's prediction is higher than other objects in various iterations.

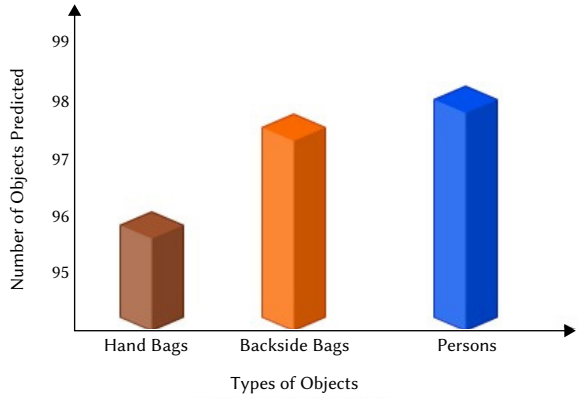

Fig. 8. The number of the objects predicted.

The precision value is associated with the true accuracy of predictions. The recall value is associated with true prediction and tracking found in the sequence of frames. The proposed work prediction is used in tuning and the tuning parameters are shown in Table III and the performance is shown in Fig. 9. The fine-tuning improvement showed a better performance compared to the other works. The time taken to predict an object is a measure in seconds. From Fig. 9, it is clear that the performance of the proposed DFCN-HP is better to predict objects than DNN and KF.

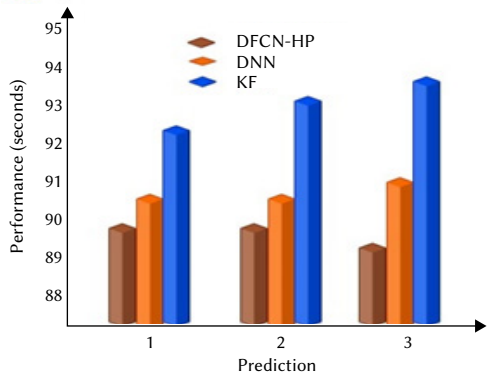

Fig. 9. Performance of Fine-Tuning.

The overall comparison parameters of the proposed work MOTA, MOTP, precision, Recall and MT are compared with those of existing works [33], [34], [45] and are shown in Fig. 10. The proposed work of MT is 99% and compared to the other methods it produces a high tracing rate. All the methods are including spatial information for tracking such as detection bounding areas, appearance, etc. and all the existing methods are not using temporal information. This proposed work considered the delay time and time slap also for prediction and tracing. The proposed work DFCN-HP and existing work DNN [34] are having online trackers with a similar learned motion model. The comparison of the results and multiple parameters is noted and it is shown in Table V and overall performance is shown in Fig. 10.

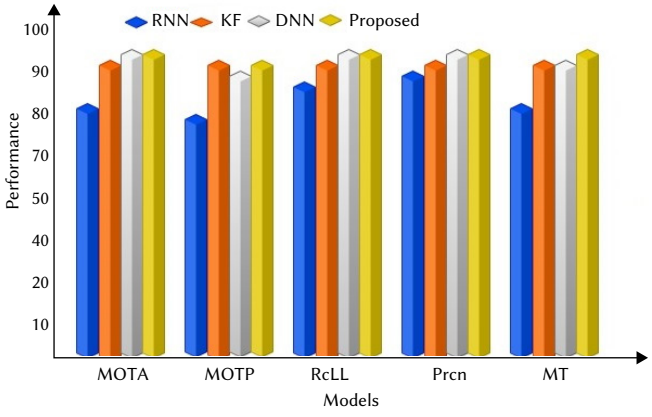

Fig. 10. Performance Comparison of the proposed work.

TABLE V. Comparison to Detection to Tracking

Tracking parameters with Methods											
Precision	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
All assignment (DNN)	0.3	0.422	0.92	0.92	0.93	0.92	0.93	0.93	0.88	0.7	0.5
Detection to track (DNN)	0.9	0.92	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.91
DFCN-HP - all assignment	0.4	0.8	0.93	0.95	0.95	0.95	0.94	0.94	0.93	0.8	0.7
DFCN-HP - Detection to track	0.98	0.98	0.99	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98

The speed of prediction of the proposed work is shown in Fig. 11. The speed of the prediction of work is compared to different standard object detection and tracking methods such as Fast R-CNN, R-CNN, and Faster R-CNN. The speed of prediction of the work DFCN-HP is 0.11 and, compared to the other existing work, this value is considered very low.

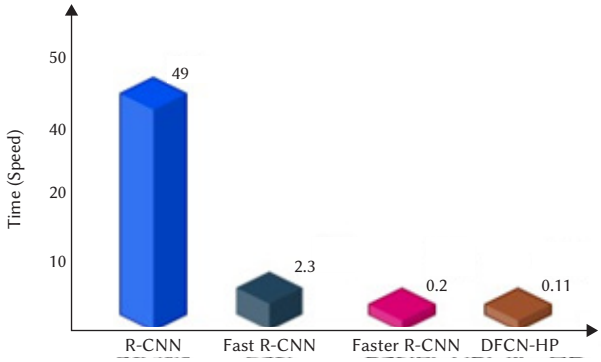

Fig. 11. Performance Comparison

V. Conclusion

Recently, there has been considerable advancement in the field of security and surveillance through different research projects that are being carried out by researchers. The proper utilization of all the new advanced techniques in object detection could dramatically change the field of object detection and open the doors to new research areas. In this research work, keeping surveillance systems for security in mind, the goal was to take a look at different types of static and dynamic object detection and tracking hybrid methods as have been introduced in this work. The main goal of the proposed hybrid DFCN-HP work is to increase the accuracy and decrease the training time to contribute to the area of human security systems. Furthermore, in this work, the hyperparameters have been fine-tuned to increase the speed and accuracy of the model. Several tests were performed to tune the hyperparameters and to evaluate the difference in performance thereof and consequently to pick certain new values of these hyperparameters for video surveillance systems. The proposed hybrid DFCN-HP method was also compared to the KF and DNN methods and was observed to produce better results in terms of multiple parameters such as MTO, ML and Accuracy.

References

[1] Ahn, H., and Cho, H., "Research of multi-object detection and tracking using machine learning based on knowledge for video surveillance system," *Personal and Ubiquitious Computing*, pp. 1-10, 2019, doi:10.1007/s00779-019-01296-z.

[2] A. Raghunandan, Mohana, P. Raghav and H. V. R. Aradhya, "Object Detection Algorithms for Video Surveillance Applications," *International Conference on Communication and Signal Processing* (ICCSP), 2018, pp. 0563-0568, doi:10.1109/ICCSP.2018.8524461.

[3] G. Chandan, A. Jain, H. Jain and Mohana, "Real Time Object Detection and Tracking Using Deep Learning and OpenCV," *International Conference on Inventive Research in Computing Applications (ICIRCA)*, 2018, pp. 1305-1308, doi: 10.1109/ICIRCA.2018.8597266.

[4] S. Ren, K. He, R. Girshick and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137-1149, 2017, doi:39. 10.1109/TPAMI.2016.2577031.

[5] Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 779-788, doi: 10.1109/CVPR.2016.91.

[6] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, A. C. Berg, "SSD: Single shot multibox detector," *Computer Vision in ECCV*, 2016, pp. 1-17, doi.org/10.1007/978-3-319-46448-0_2.

[7] Redmon, Joseph and Ali Farhadi. "YOLOv3: An Incremental Improvement" *ArXiv* abs/1804.02767, 2018, pp. 1-6.

[8] R. Girshick, J. Donahue, T. Darrell and J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," *IEEE Conference on Computer Vision and Pattern Recognition*, Columbus, OH, 2014, pp. 580-587, doi: 10.1109/CVPR.2014.81.

[9] Uijlings, Jasper Sande, K. and Gevers, T. and Smeulders, Arnold, "Selective Search for Object Recognition," *International Journal of Computer Vision*, vol. 104, pp. 154-171, doi:10.1007/s11263-013-0620-5.

[10] R. Girshick, "Fast R-CNN," *IEEE International Conference on Computer Vision (ICCV)*, Santiago, 2015, pp. 1440-1448, doi: 10.1109/ICCV.2015.169.

[11] Kaiming HeXiangyu and ZhangShaoqing RenJian Sun, "Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, pp. 1904-1916, 2014.

[12] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, 1137–1149, 2017, DOI:https://doi.org/10.1109/TPAMI.2016.2577031.

[13] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *In International Conference on Learning Representations*, 2015, pp.1-14.

[14] He, Kaiming. "Mask R-CNN," *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 2980-2988.

[15] Sankaranarayanan, Aswin Veeraraghavan, Ashok Chellappa, Rama, "Object Detection, Tracking and Recognition for Multiple Smart Cameras," *Proceedings of the IEEE*, vol. 96, no. 10, pp. 1606 – 1624, doi:10.1109/JPROC.2008.928758.

[16] C. R. del-Blanco, F. Jaureguizar and N. Garcia, "An efficient multiple object detection and tracking framework for automatic counting and video surveillance applications," *EEE Transactions on Consumer Electronics*, vol.58, no.3, pp.857-862, August 2012, doi: 10.1109/TCE.2012.6311328.

[17] K. S. Kumar, S. Prasad, P. K. Saroj and R. C. Tripathi, "Multiple Cameras Using Real Time Object Tracking for Surveillance and Security System," *3rd International Conference on Emerging Trends in Engineering and Technology*, 2010, pp. 213-218, doi: 10.1109/ICETET.2010.30.

[18] Ray, Kumar S. and Soma Chakraborty, "An Efficient Approach for Object Detection and Tracking of Objects in a Video with Variable Background," *ArXiv* abs/1706.02672, 2017, pp. 1-11.

[19] Wenqian Liu, Octavia Camps, and Mario Sznaier, "Multi-camera Multi-Object Tracking," *ArXiv* abs/1709.07065, 2017, pp.1-7.

[20] Kachhava, Rajendra, Shrivasta, Vivek, Jain, Rajkumar, Chaturvedi, Ekta, "Security System and Surveillance Using Real-Time Object Tracking and Multiple Cameras," *Advanced Materials Research*, vol. 403-408,4968-4973, doi: 10.4028/www.scientific.net/AMR.403-408.4968.

[21] Chen, Chung-Hao, Yao, Yi, Page, David, Abidi, Besma, Koschan, Andreas, Abidi, Mongi, "Heterogeneous Fusion of Omnidirectional and PTZ Cameras for Multiple Object Tracking. Circuits and Systems for Video Technology," *IEEE Transactions on Circuits and Systems for Video Technology*, vol.18, no.8, pp.1052-1063, doi: 10.1109/TCSVT.2008.928223.

[22] Taj, Murtaza Cavallaro, Andrea, "Simultaneous Detection and Tracking with Multiple Cameras," *Studies in Computational Intelligence*, 411, pp 197-214, 2013, doi:10.1007/978-3-642-28661-2_8.

[23] R, Y. Da Xu and M. Kemp, "Fitting multiple connected ellipses to an image silhouette hierarchically," *IEEE Transactions. on Image Processing*. vol.19, no. 7, 1673-1682, Jul 2010.

[24] Zhong-Qiu Zhao, Peng Zheng, Shou-tao Xu, and Xindong Wu, "Object Detection With Deep Learning: A Review," *IEEE Transactions on Neural Networks and Learning Systems, vol.* 30, pp. 3212-3232, 2019.

[25] Licheng Jiao, Fan Zhang, Fang Liu, Shuyuan Yang, Lingling Li, Zhixi Feng, and Rong Qu, "A Survey of Deep Learning-Based Object Detection," *IEEE Access* vol. 7, pp. 128837-128868, 2019.

[26] Anjum, Nadeem Cavallaro, Andrea . "Trajectory Association and Fusion across Partially Overlapping Cameras," *sixth ieee international conference on advanced video and signal based surveillanc*. pp 201-206,2009. 10.1109/AVSS.2009.65.

[27] V. Morariu and O. Camps, "Modeling Correspondences for Multi-Camera

Tracking Using Nonlinear Manifold Learning and Target Dynamics," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, New York, NY, USA, 2006, pp. 545-552. doi: 10.1109/CVPR.2006.189.

[28] Fleuret, F., Berclaz, J., Lengagne, R., Fua, P. "Multicamera people tracking with a probabilistic occupancy map," *IEEE Transactions on Pattern Analysis and Machine Intelligence* 30(2),267–282 (2008)

[29] Eshel, R., Moses, Y. "Homography-based multiple camera detection and tracking of people in a dense crowd," In: *Proc. of IEEE Int. Conf. on Computer Vision and Pattern Recognition,* Anchorage, AK, USA (June 2008).

[30] Kim, K., Davis, L.S.: "Multi-Camera Tracking and Segmentation of Occluded People on Ground Plane Using Search-Guided Particle Filtering," *European Conference on Computer Vision.* Springer, Berlin, Heidelberg, 2006.

[31] Taj, Murtaza Cavallaro, Andrea. "Multi-camera track-before-detect," *3rd ACM/IEEE International Conference on Distributed Smart Cameras*, ICDSC 2009. 1 - 6. 10.1109/ICDSC.2009.5289405.

[32] Tan, Yihua Tai, Yuan Xiong, Shengzhou. . "NCA-Net for Tracking Multiple Objects across Multiple Cameras," Sensors. 18. 3400. 10.3390/s18103400,2018.

[33] Hossain, Sabir, and Deok-Jin Lee. "Deep Learning-Based Real-Time Multiple-Object Detection and Tracking from Aerial Imagery via a Flying Robot with GPU-Based Embedded Devices," *Sensors (Basel, Switzerland)* vol. 19, 15 3371. 31 Jul. 2019, doi:10.3390/s19153371

[34] Yoon, Kwangjin. "Data Association for Multi-Object Tracking via Deep Neural Networks," *Sensors (Basel, Switzerland)* vol. 19, no. 3 pp. 559. 29 Jan. 2019, doi:10.3390/s19030559

[35] Sikora P, Malina L, Kiac M, Martinasek Z, Riha K, Prinosil J, Jirik L, Srivastava G. "Artificial Intelligence-based Surveillance System for Railway Crossing Traffic," *IEEE Sensors Journal.* 2020 Oct 16.

[36] Vallathan G, John A, Thirumalai C, Mohan S, Srivastava G, Lin JC. "Suspicious activity detection using deep learning in secure assisted living IoT environments," *The Journal of Supercomputing.* 2020 Jul 30:1-9.

[37] Wang X, Srivastava G. "The security of vulnerable senior citizens through dynamically sensed signal acquisition," *Transactions on Emerging Telecommunications Technologies.* 2020 Jul 14:e4037.T.

[38] Ahmad Jalal , Shaharyar Kamal and Daijin Kim, "A Depth Video-based Human Detection and Activity Recognition using Multi-features and Embedded Hidden Markov Models for Health Care Monitoring Systems," *International Journal of Interactive Multimedia and Artificial Intelligence*, Vol. 4, No. 4, pp. 54-62, 2017.

[39] Anahita Ghazvini, Siti Norul Huda Sheikh Abdullah, Masri Ayob, "A Recent Trend in Individual Counting Approach Using Deep Network," *International Journal of Interactive Multimedia and Artificial Intelligence*, Vol. 5, No. 5, pp. 7-14, 2019.

[40] T. Ojala, M. Pietikäinen, and D. Harwood, "A comparative study of texture measures with classification based on featured distributions," *Pattern Recognition,* vol. 29, no. 1, pp. 51–59, 1996.

[41] Ahonen, A. Hadid, and M. Pietikainen, "Face description with local binary patterns: Application to face recognition," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 12, pp. 2037–2041, 2006.

[42] Z. Zivkovic and F. van der Heijden, "Efficient adaptive density estimation per image pixel for the task of background subtraction," *Pattern Recognition Letters*, vol. 27, pp. 773–780, 2006.

[43] Rakesh Chandra Joshi, Adithya Gaurav Singh, Mayank Joshi, Sanjay Mathur, "A Low Cost and Computationally Efficient Approach for Occlusion Handling in Video Surveillance Systems," *International Journal of Interactive Multimedia and Artificial Intelligence*, Vol. 5, No. 7, pp. 28-38 2019.

[44] Krizhevsky, Alex. "Learning Multiple Layers of Features from Tiny Images," *Technical Report TR-2009, University of Toronto*, Toronto..

[45] Sadeghian, A.; Alahi, A.; Savarese, S. Tracking, "The Untrackable: Learning To Track Multiple Cues with Long-Term Dependencies," *arXiv 2017, arXiv*:1701.01909.

[46] Taj, Murtaza. "Tracking interacting targets in multi-modal," *sensors.* Diss. 2009.

[47] Chen, Muchun, et al. "Real-Time Multiple Pedestrians Tracking in Multi-camera System," *International Conference on Multimedia Modeling.* Springer, Cham, 2020.

[48] Xiaokai, Liu, Wang Hongyu, and Gao Hongbo. "Camera matching based on spatiotemporal activity and conditional random field model," IET Computer Vision 8.6 (2014): 487-497.

M. Adimoolam

Dr. M. Adimoolam received his Under Graduate Degree B.Tech in Computer Science and Engineering Discipline at Pondicherry University. He finished his Post Graduate Degree M.Tech in Information Security at Pondicherry Engineering College, Puducherry and it was sponsored by the Department of Information Technology, India under Project Information Security Awareness and Education. In 2019 he was completed a Ph.D. in Information Security – Computer Science and Engineering discipline at Manononmaniam Sundaranar University, Tirunelveli. Right now he is working as an Associate Professor at Saveetha School of Engineering in Institute of Computer Science. He is the life time member of ISTE, India. He has cleared the University Grant Commission conducting National Eligibility Test 8 times. He has also cleared the Tamilnadu Government conducting State Eligibility Test continuously 3 times. His research areas of interest are computer network, Information and Network Security, machine learning and deep learning.

John A

Dr. John A received his Under Graduate Degree B.Tech in Computer Science and Engineering Discipline at Pondicherry University. He finished his Post Graduate Degree M.Tech in Computer Science and Engineering at Pondicherry University, India. In 2019 he was completed a Ph.D. in Computer Science and Engineering discipline at Manononmaniam sundaranar University, India. Right now, he is working as an Assistant Professor at School of Computer Science in Galgotias University, India. He is the life time member of ISTE, India. His research areas of interest are real time applications, Data analysis and prediction, and Spatial and Temporal Database.

Senthilkumar Mohan

Dr. Senthilkumar Mohan was felicitated with a Ph.D. in engineering and technology from Vellore Institute of Technology in the year 2017. He has obtained his M.Tech in IT from VIT University in the year 2013. He earned his M.S (Software Engineering) degree in computer science and Engineering from VIT University Vellore, in the year2007. He is presently working in the rank of Associate Professor at the Department of Software and System Engineering, Vellore Institute of Technology, School of Information Technology and Engineering, Vellore, India. His area of research includes Artificial Neural Networks, Deep Learning, cloud computing. He has contributed to many research articles in various journals and conferences of repute. He is also a member of various professional societies like CSI, Indian congress, etc.

Gautam Srivastava

Dr. Gautam Srivastava was awarded his B.Sc. degree from Briar Cliff University in U.S.A. in the year 2004, followed by his M.Sc. and Ph.D. degrees from the University of Victoria in Victoria, British Columbia, Canada in the years 2006 and 2011, respectively. He then taught for 3 years at the University of Victoria in the Department of Computer Science, where he was regarded as one of the top undergraduate professors in the Computer Science Course Instruction at the University. From there in the year 2014, he joined a tenure-track position at Brandon University in Brandon, Manitoba, Canada, where he currently is active in various professional and scholarly activities. He was promoted to the rank Associate Professor in January 2018. Dr. G, as he is popularly known, is active in research in the field of Data Mining and Big Data. In his 8-year academic career, he has published a total of 243 papers in high-impact conferences in many countries and in high-status journals (SCI, SCIE) and has also delivered invited guest lectures on Big Data, Cloud Computing, Internet of Things, and Cryptography at many Taiwanese and Czech universities. He is an Editor of several international scientific research journals. He currently has active research projects with other academics in Taiwan, Singapore, Canada, Czech Republic, Poland and U.S.A. He is constantly looking for collaboration opportunities with foreign professors and students. Assoc. Prof. Gautam Srivastava received *Best Oral Presenter Award* in FSDM 2017 which was held at the National Dong Hwa University (NDHU) in Shoufeng (Hualien County) in Taiwan (Republic of China) on November 24-27, 2017.