Multiple Object Tracking using STMRF and YOLOv4 Deep SORT in Surveillance Video

[1]Kusuma T, [2]Dr. Ashwini K

[1]Research Scholer, Department of Computer Science & Engineering Global Academy of Technology, Bengaluru
[2]Professor, HOD, Department of Artificial Intelligence and Data Science, Global Academy of Technology, Bengaluru

***Abstract*: Multiple items tracking are a process of providing unique and consistent ownership of objects during video sequencing. This paper introduces a moving tracking sequence of video compressed H.264 / AVC using the 'Spatio Temporal Markov Random Field model (ST-MRF)'. The proposed method operates on a compressed domain and tracks moving vectors (MVs) and blocks coding modes (BCMs) from a compressed bitstream. The results presented in this paper suggest that the volume of the object detection algorithm reflects the overall performance of the tracking system. Finally, we investigate how the use of visual definitions in the tracking phase of a tracking system affects performance using Deep SORT. The results presented in this paper suggest that the volume of the discovery algorithm reflects the overall performance of the tracking-by detection system**.

Index Terms: Object Tracking, ST-MRF, Deep SORT, Block Coding Modes

1. INTRODUCTION

Video object tracking is an area of computer vision that deals with the localization of moving objects in video. There are many applications of video tracking in fields such as robotics, sports analysis, and video surveillance. These applications often require multiple objects to be tracked at the same time, which is referred to as "multiple object tracking". A popular approach to tracking of object is known as "tracking-by-detection (TBD)". TBD uses an object detection process to detect objects in a frame. These objects tracking by associating objects in the current frame with objects from previous frames using a tracking algorithm. Having a reliable method for object detection is crucial since the tracking algorithm is dependent on objects being detected in each frame. Lately, object detection algorithms based on convolutional neural networks have been able to achieve greater accuracy than traditional object detection methods. This improvement in object detection accuracy has facilitated the use of tracking-by-detection methods for multiple object tracking. Motion detection plans can be divided into two categories: (i) a video pixel domain approach; and (ii) a compression domain approach. The compressed domain approach [1-3] is based on compression bitstream artifact video codings, such as motion vector (MV), a macroblock section, and a quantization coefficient for motion detection. Compared to algorithms based on pixel domains, compressed domain methods generally require fewer computing resources since the input information can already be analyzed in the bitstream. Since the input analysis is already possible on bits, it provides the detection of objects moving by compression to apply data. Using only MVSs with video encoders in compressed streams, our approach can effectively determine moving objects. Furthermore, an experimental evaluation of our method is provided to compare the processing time with compression domain tracking methods.

. The primary focus of this paper has been on growing a current new and strong moving target monitoring model within the compressed area. This paper aims to build a novel and robust Spatio-Temporal Markov Random fiend (ST-MRF)[17] model for moving target tracking in a compressed domain. Tracking-by-detection system uses a detection algorithm called YOLO[5] to detect objects and a tracking algorithm called Deep SORT [4] to track detected objects. Deep SORT is an extension of the algorithm SORT [6], which does not use any appearance information in the tracking stage. This system is relevant since it is used as a test environment throughout this paper.

2. ST-MRF BASED OBJECT TRACKING

The purpose of this paper is to introduce a moving tracking framework for an MV-restricted domain only. Using the ST-MRF concept, the proposed model aims to enhance the accuracy of target acquisition and continuous tracking where different domains are located in Figure 1.

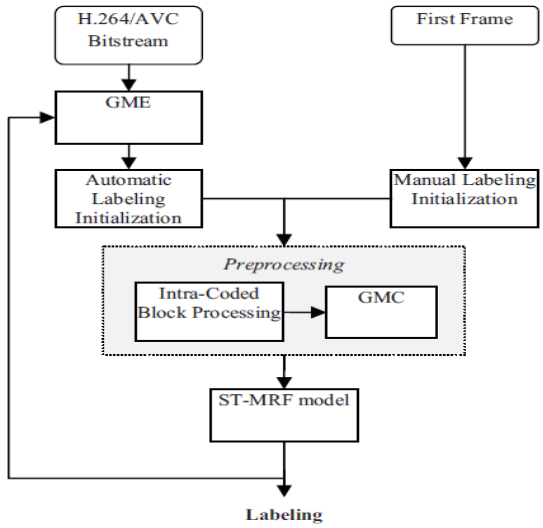


Figure 1. Flow chart of ST-MRF based Tracking

2.1. PVM and BCM assisted Moving target tracking in the compressed domain

A real-time CCTV system that can track many objects in real-time using Polar Vector Median (PVM) and Block Coding (BCM) mechanisms with Global Motion Compensation (GMC) has been developed [17]. This strategy works in a full environment and tracks an object in real time by using moving BCM vectors from a small squeezed transmission area. It is carried out in conjunction with adjacent motion vectors (MV) using a technique known as PVM. The proposed method is tested in a standard sequence and the results show its advantages in some of the current methods.The internal and in-image assumptions use correlations between local neighboring samples to obtain a global motion compensated (GMC) block of image sample block.This method is tested in several standard sequences and the results show its advantages in some of the current methods and in independent samples. Based on internal predictions, we assume that the neighboring block may appear as an object moving with the same motion.

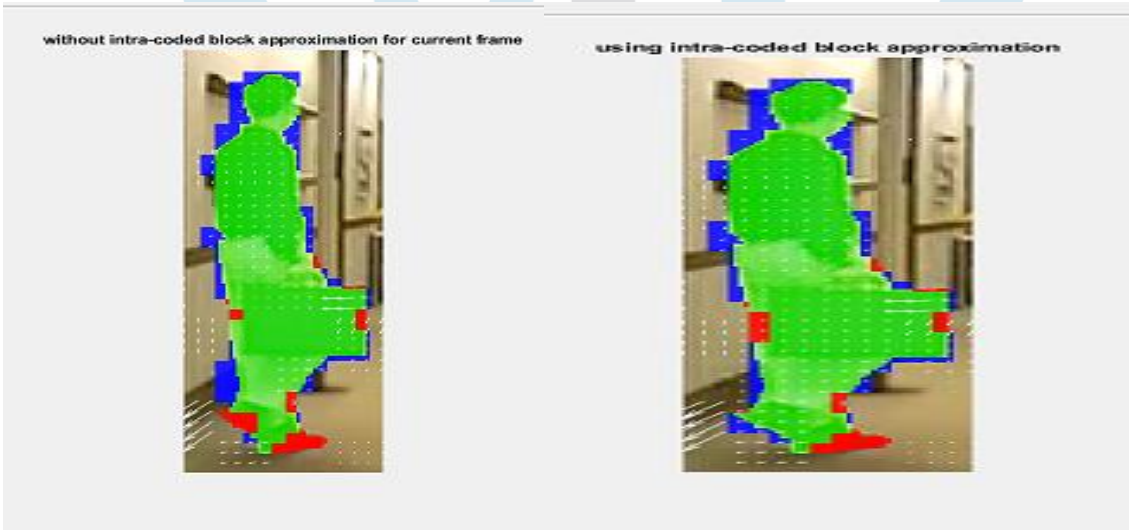


Figure 2. Impact of assigning PVM to coded blocks a) Trace tracking without PVM distribution. (b) PVM tracking effect.

3 DEEP LEARNING-BASED MOVING OBJECT DETECTION

3.1 Object tracking

Multiple objects tracking (MOT) is the characteristic of allocating a reliable and precise identifier to multiple objects within the video sequence. Object tracking is a two-step process: an algorithm for finding an object present in a frame[8]; these elements are related to the tracked gadgets with a tracking algorithm [8].Generally, object detection algorithms and tracking algorithms are completely separated and therefore can be individually analyzed. CNN-based object acquisition algorithms can be divided into two distinct groups: single-stage and two-stage detectors [9]. Two-phase receivers begin to produce binding boxes that can be split by separating the image into your favorite regions; these regions were then separated separately by CNN in the second phase. The tracking algorithm in a tracking-by-detection framework is responsible for assigning unique identities to tracked objects and making object associations between frames. This paper's main focus is on object detection algorithms and only two different tracking

algorithms will be considered, SORT [6] and Deep SORT [4]. SORT stands for Simple Online Realtime Tracking; it is a deliberately simple tracking algorithm that uses a Kalman filter [10] to estimate future positions of objects and makes frame-to-frame associations using the Hungarian method [11]. Deep SORT is an extension of SORT that incorporates appearance information when doing object associations between frames.

3.1 Object Detection Algorithms

3.1.1 R-CNN

It is a method for object detection introduced by Girschick et al. in [12]. The system is a pipeline with three main parts: a region proposal, a convolutional neural network, and a set of support vector machines (SVMs). Figure 3 below shows the interaction of the main parts of R CNN. First, the region proposal method segments an image into category-independent regions. This generates approximately 2000 regions per image. After segmenting the image, each region is warped to a fixed size to fit the required input size of the CNN. Next, the 2000 warped regions are separately fed through the CNN and a feature vector is extracted for each region. The feature vector is then classified by a set of linear SVMs, where each SVM is trained to classify one specific class. Finally, given the class predicted by the SVMs, ridge regression is used to improve the predicted shape of the bounding box. When all regions are scored, non-maxima suppression is applied to remove predicted bounding boxes that overlap with predictions with higher scores. R-CNN is not restricted to any specific segmentation method or a specific CNN architecture. In [12], a segmentation method called selective search [13] is used and results are demonstrated when CNN architectures presented in [14] and [22] are used.

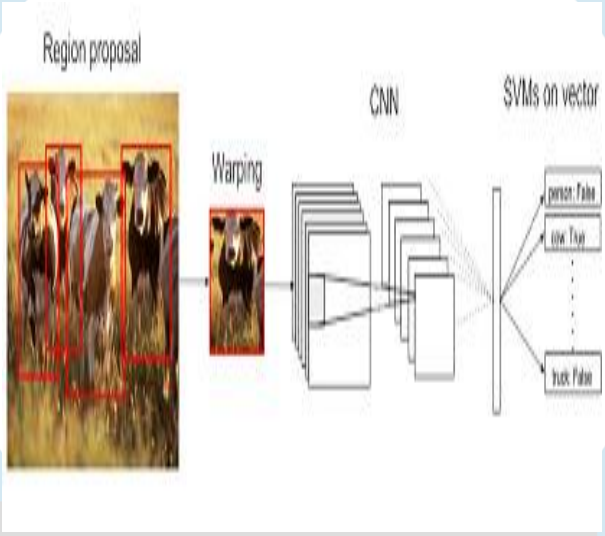


Figure 3. Schematic of the R-CNN pipeline.

The authors of R-CNN also showed that supervised pre-training on a similar problem is an effective way to initialize the weights of a CNN. In [12], initial weights of the CNN were obtained by pre-training the CNN to perform image classification on data.

3.2.2 YOLO

Redmon et al. introduced a novel approach to object detection in [15] called YOLO, You Only Look Once. Unlike R-CNN and its successors, YOLO does not use any region proposal method and instead uses a single CNN to predict both bounding boxes and classes. In YOLO, an input image is first split into an S×S grid. Each grid cell is then responsible for predicting B bounding boxes as well as a confidence score for every bounding box. The confidence score is calculated as $\Pr((\text{Object})) * \text{IoU}^{\text{gt}}_{\text{pred}}$ where$\Pr((\text{Object}))$ is the predicted probability that the box contains an object and $\text{IoU}^{\text{gt}}_{\text{pred}}$ is the estimated intersection over union (IoU) between the predicted box and a ground truth box. For each grid cell, C object class probabilities are also predicted, these probabilities are conditioned on the cell containing an object. The predicted boxes and class probabilities are then combined into a single score for each class and box. Equation 1 is taken from the introduction of YOLO in [15] and shows how the class predictions and box predictions are combined. As in the original paper [15], $\Pr(\text{Class}_i)$ is used as a simplified notation for $\Pr(\text{Class}_i, \text{Object})$.

$$\Pr(\text{Class}_i|\text{Object}) * \Pr(\text{Object}) * \text{IoU}^{\text{gt}}_{\text{pred}} = \Pr(\text{Class}_i) * \text{IoU}^{\text{gt}}_{\text{pred}} \quad (1)$$

The score accounts both for the probability that the box contains class i, $\Pr(\text{Class}_i)$, and for how the predicted box is estimated to fit a ground truth box, IoUgt pred. The predicted bounding boxes are then combined with class probabilities, which are also obtained from the image grid, to produce the final object detections. The illustration is kept simple so that it is easier to understand, in reality, there would be many more objects predicted by the grid

Similar to Faster R-CNN, YOLOv2 utilizes anchors when predicting bounding boxes. For each grid cell, YOLOv2 produces bounding boxes by predicting offsets to 5 anchors. Classes are now also predicted for each anchor instead of for each grid cell, each anchor is also given an objectness score. As in YOLO, classes are predicated on the condition that there is an

objectPr(Class$_i$|Object). Objectness is calculated as the estimated IoU between the predicted box and an estimated ground truth box IoU^{gt}_{pred}. YOLOv2 also employs a new method to determine anchor sizes; instead of hand-picking the anchors as in Faster R-CNN, YOLOv2 uses k-means clustering on the training data to produce anchors that are better fitted to the data.

3.3 Tracking Algorithms

The following section presents the theory behind the two tracking algorithms considered in this thesis: SORT and Deep SORT. As described in the limitations, these two are especially suited to answer the second research question without broadening the scope too much

3.3.1 SORT

Simple Online Tracking and Real-Time, SORT, tracking algorithm developed by Bewley et al. of [16]. SORT was developed to perform multiple object Tracking (MOT) on the recognition tracking system. It uses CNN-based finders to rely on finding something more accurate. In each new framework, SORT distributes pre-downloaded items to the current framework. A Kalman filter [10] is combined with a fixed-line line model to predict the new shape of these pre-tracked objects. After that, the object detection algorithm scans the current frame of objects. To create a cost matrix, these acquisitions are compared with previously tracked items. The IoU between each detection and each previously monitored object is used to calculate this value. Detections are then assigned to already tracked objects using the Hungarian method [11]. A new track is created when an object is detected in several consecutive frames while not overlapping with any of the already tracked objects.Figure 4 below shows how predicted positions are compared to object detections to assign identities in new frames. SORT, as it is used in this paper, does not have any memory and a tracked object is lost if SORT fails to detect it in a frame.

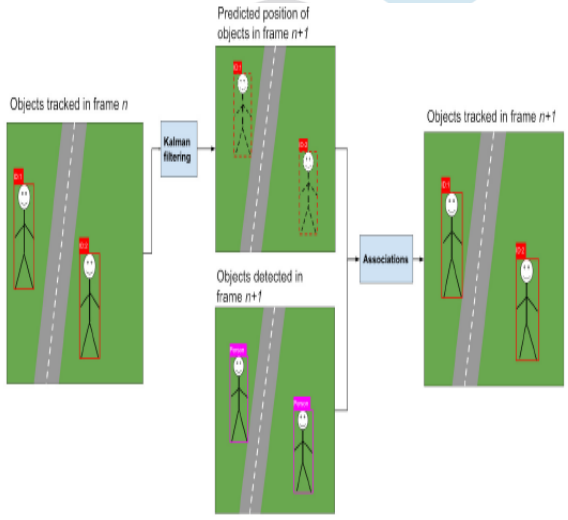


Figure. 4 Object associations between frames in SORT

3.3.2 Deep SORT

Designed to reduce the number of out-and-out exchanges, deep SORT incorporates visual information in the tracking process presented to SORT [4]. Designed to reduce the number of patent exchanges, Deep SORT incorporates visual information in the tracking process presented to SORT [4]. Similar to SORT, Deep SORT handles state estimations with a Kalman filter. Deep SORT differs from SORT in that it makes use of additional techniques when assigning detection to already track objects. Deep SORT utilizes two different distance metrics when comparing detection to already tracked objects: Mahalanobis distance [7] and cosine distance between\ appearance descriptors. The Mahalanobis distance measures how the position of a new detection differs from the positions of already tracked objects in terms of standard deviations from the tracked objects.

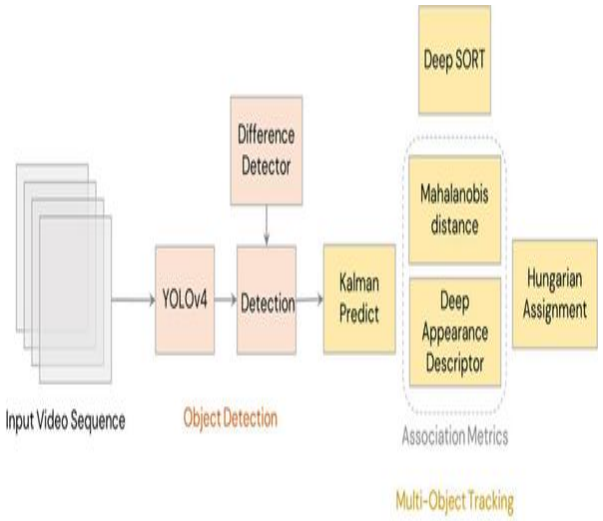


Figure.5 YOLOv4 Deep SORT

This metric allows Deep SORT to avoid giving a discovery to an existing track where the independent movement of the frame makes no sense. Appearance descriptions are calculated by forwarding each binding box with a pre-trained CNN in the database for personal identification. By calculating the cosine range between the two adjectives, the dictionary for the appearance of each new acquisition is compared with the visual descriptions of the previously traced objects. Tracks and appearance definitions are also stored for 30 frames after a loss, allowing Deep SORT to continue tracking lost ownership for most frames. By using the feature descriptors in this way, Deep SORT can get the pre-tracked object even if it was hidden in a few frames.

3.3.3 SORT with Deep Association Metric

This algorithm will be referred to as Deep SORT, and it is an extension of the SORT algorithm described in the previous section. SORT is fast and simple, while it performs very well in terms of precision and accuracy. However, it also brings the highest number of proprietary switches. This stimulus is developed using descriptive visual aids, which are used to compare the findings from one frame to another track video sequence identity. In deep matric learning methods, the concept of similarity is applied directly to the training objective. The Kalman filtering handles occlusion, but when an object has been occluded for several frames, the prediction becomes more uncertain.

Therefore, the majority of opportunities are widespread in the region and there is a risk that high uncertainty is prioritized due to the reduced range of general deviation of any acquisition to the description of the proposed track. Deep SORT solves this problem by using a matching cascade that prioritizes more frequently seen objects [6]. As mentioned earlier; the Deep SORT algorithm is an extension of the SORT algorithm. Comparison of cosine distance between feature vectors is consistent with measuring IoU distance and distances between Kalman district estimates. Thanks to this upgrade, Deep SORT gets better accuracy on standard benchmarks [6].

3.4 Classification of Predicted Bounding Boxes

The fundamental performance metric is the classification of bounding boxes. Table.1 below shows the different classes that a bounding box can be assigned. A predicted bounding box is considered a true positive (TP) if its intersection over union (IoU), or Jaccard Index [20], with a ground truth box is larger than 0.5. The table shows how the predicted box P and a ground truth box G are calculated. False positives (FP) are predicted bounding boxes without a corresponding ground truth case, and false negatives (FN) are ground truth boxes that the algorithm fails to detect.True negatives (TN) are irrelevant in this context since object detection algorithms do not produce any predictions on whether objects are absent. For all performance metrics defined in subsequent sections, TP, FP, and FN are used as shorthand notations for the number of bounding boxes that have been labeled as belonging to each class. GT is also used as a notation for the total number of ground truth bounding boxes that exist.

Table 1: Classification of bounding boxes

Ground Truth	**Prediction** Object (positive)	**Prediction** Background (negative)
Object (positive)	TP (True Positive) Correctly labeled as an object	FN (False Negative) Incorrectly labeled as background
Background (negative)	FP (False Positive) Incorrectly labeled as an object	TN(True Negative) Correctly labeled as background

Object Detection Evaluation

Recall [21]: Recall is measured as the ratio between well-received items and the total number of basic facts. Thus, the recall of the algorithm demonstrates its ability to obtain basic facts.

$$\text{Recall}=\frac{\text{TP}}{\text{TP+FN}}$$

Precision (Prcn) [21]: Precision describes the accuracy of predicted bounding boxes. It is calculated as the ratio between correctly predicted bounding boxes and the total number of predicted bounding boxes.

$$\text{Precision}=\frac{\text{TP}}{\text{TP+FP}}$$

F1 score (F1) [21]: F1 score is a metric that combines recall and precision into a single score by calculating the harmonic mean of precision and recall.

$$\text{F1}=\frac{\text{TP}}{\text{TP+FP+FN}}$$

Multiple Object Detection Accuracy (MODA) [39]: MODA measures the accuracy of predictions by looking at missed ground truth boxes and false positives.

$$\text{MODA}=\left(1-\frac{\text{FN+FP}}{\text{GT}}\right).100$$

Frames Per Second (FPS): FPS is a metric for comparing the speed of object detection algorithms. It is calculated as the ratio between the number of frames processed and the time that it takes to run the algorithm.

$$\text{FPS}=\frac{\#\text{frames}}{\text{total runtime}}$$

Even if Deep SORT uses both motion and visual appearance, the algorithm struggles when objects are occluded. In the Laboratory Sequences, the objects move back and forth in circles and their motion pattern is irregular and complex.

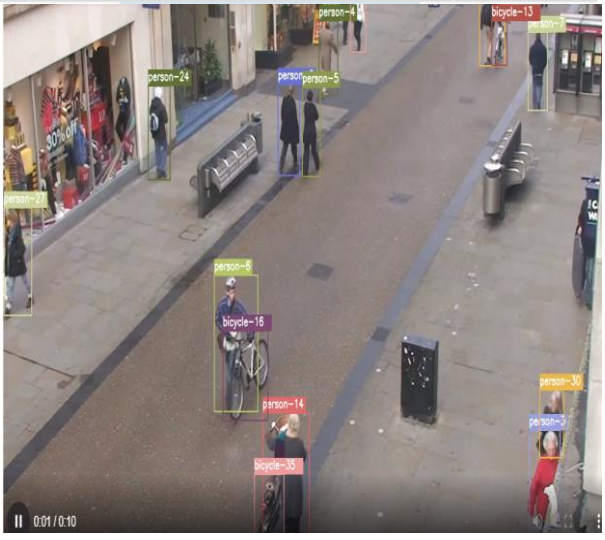


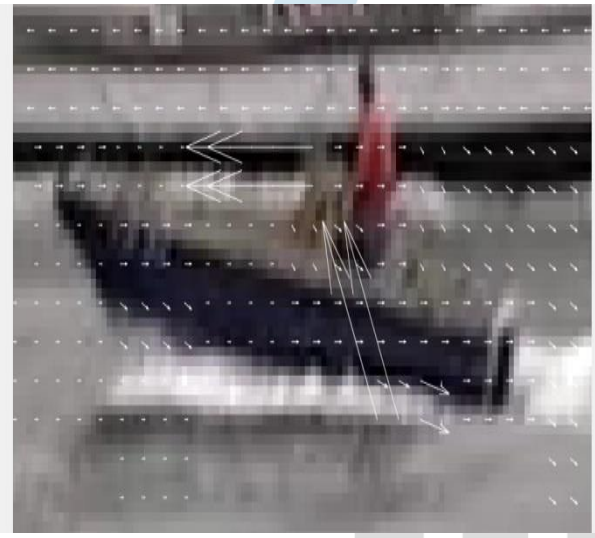
Figure.6 The result of YOLOv4 Deep SORT, with bounding boxes and ID for each person. For each object there are two bounding boxes, the red box is the detection in the current frame and the colored box is the predicted bounding box given by the Kalman filter.

When an object passes another object and is fully occluded, the tracker mostly assigns a new ID to the object when it is visible again. Several IDs can therefore be assigned to the same object throughout the sequence. An illustration of this is shown in figure 5.

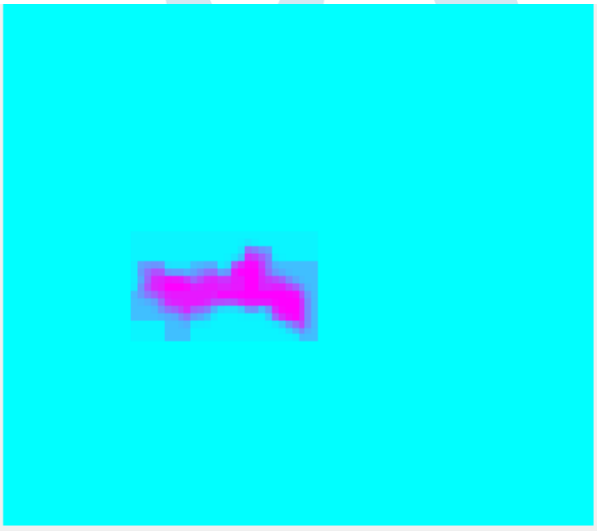
Table .2 Proposed method and state-of-the-arts on MOT-16 dataset

Method	MOTA	MOTP	MT	ML	IDs	FN	FM	FPS
SORT	59.8	79.6	25.4	22.7	1423	63245	1835	60
Deep SORT	61.4	79.1	32.8	18.2	781	56,668	2008	14
SORT - YOLOv4	63.4	81.7	33.8	16.7	707	21439	1888	21.5

Due to the apparent rapid camera motion in this part of the sequence, the MV field around the target (a small boat) is fairly erratic, as shown in Figure 7 (a). The proposed ST-MRF- based tracking algorithm calculates the power function of the target. MRF model as shown in Figure 7 (b). The darker the value, the lower the power as well as the backward opportunities. As a result, despite the negative MV field, the target seems to be getting better. Figure 7 (c) displays the target region obtained once the completion of the follow-up process. Different colored pixels represent TP, FP, and FN.



(a)



(b)

(c) Figure 7. ST-MRF-based tracking. (a) Target in Coastguard frame # 71 located above the standard MV stadium behind GMC. (b) MRF energy value — the darker the color, the stronger the energy. (c) Tracing of results in the proposed manner.

Table 3. Precision, Recall, F-measure (IN PERCENTAGE) of various QP values

QP	16	20	24	28	32	36	40
Precision	79.3	80.5	80.1	81.2	79.1	76.5	73.2
Recall	81	81.5	82.5	82.4	82.9	81.9	81
F-Measure	85.9	86.5	86.7	87.2	87	86.3	85.3
AVG	82.1	82.8	83.1	83.6	83.0	81.6	79.8

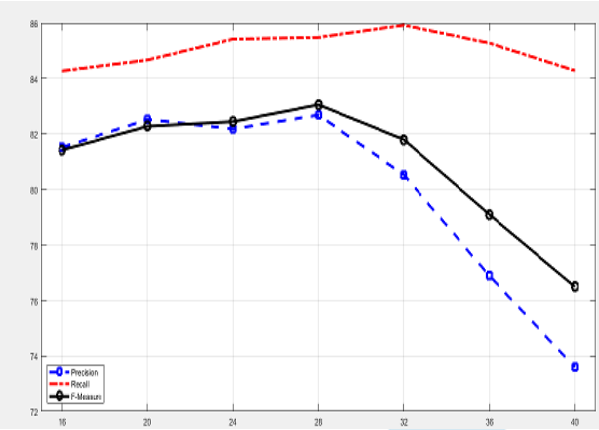


Figure 8. Comparison of several approaches in terms of Precision, Recall, and F- measure for Coastguard, where the tracking target is the boat .

CONCLUSION

In this paper, we carry an innovational way to track the moving object in an H. 264/ AVC- compressed the video. The method has a pretty equatorial processing time, yet motionlessly provided high accuracy. We retain the Spatio-Temporal Markov Random Field model to detect and track a target.. As for the first research questions, the results presented in this thesis show that the stand-alone performance of an object detection algorithm employed in a tracking by- detection system is highly correlated with the system's overall tracking performance. This correlation demonstrates the pivotal role of the object detection algorithm in a tracking-by-detection procedure. Further, it is also shown that modern single-stage detectors can achieve performance comparable to that of two-stage detectors. In general, single-stage detectors also seem to outdo two-stage detectors in terms of processing time, with YOLOv4 being the unquestionably fastest algorithm. The increase in performance is also shown to be correlated with how the object detection algorithms balance precision and recall, with DeepSORT improving the overall system the most when object detections are conservative but precise. A study has shown that the use of visual descriptors is dependent on the object detection algorithm's characteristics. This could be of aid in constructing future tracking-by-detection systems.

REFERENCES

[1] R. Huang, V. Pavlovic, and D. N. Metaxas, “A new Spatio-temporal MRF framework for video-based object segmentation,” in Proc. MLVMA Conjunct. Eur. Conf. Comput. Vis., 2008, pp. 1–12.

[2] Y. Wang, “A dynamic conditional random field model for object segmentation in image sequences,” in Proc. IEEE Comput. Vis. Pattern Recognit., vol. 1. Jun. 2005, pp. 264–270.

[3] A. Cherian, J. Anders, V. Morella's, N. Papanikolopoulos, and B. Mettler, "Autonomous altitude estimation of a UAV using a single onboard camera," in Proc. IEEE Int. Conf. Intell. Robots Syst., Oct. 2009, pp. 3900–3905. Z. Liu, Y. Lu, and Z. Zhang, “Real-time spatiotemporal segmentation of video objects in the H.264 compressed domain,” J. Visual Commun. Image Represent., vol. 18, no. 3, pp. 275–290, 2007.

[4] N. Wojke and A. Bewley. Deep cosine metric learning for person reidentification. pages 748–756, 2018. DOI: 10.1109/WACV.2018.00087.(44)

[5] J. Redmon and A. Farhadi. Yolov3: An incremental improvement. arXiv,2018.

[6] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft. Simple online and real-time tracking. In 2016 IEEE International Conference on Image Processing (ICIP), pages 3464–3468, 2016. DOI: 10.1109/ICIP.2016.7533003.4

[7] T. Hastie, R. Tibshirani, and J. Friedman. The Elements of Statistical Learning. Springer Series in Statistics. Springer New York Inc., New York, NY, USA, 2001.

[8] K. He, X. Zhang, S. Ren, and J. Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. CoRR, abs/1406.4729, 2014. URL http://arxiv.org/abs/1406.4729. W. Luo, X. Zhao, and T-K. Kim. Multiple object tracking: A review. CoRR, abs/1409.7618, 2014. URL http://arxiv.org/abs/1409.7618.-27.

[9] K.S. Chahal and K. Dey. A survey of modern object detection literature using deep learning. CoRR, abs/1808.07256, 2018. URL http://arxiv.org/abs/1808.07256.5

[10] R.E. Kalman. A new approach to linear filtering and prediction problems. Transactions of the ASME–Journal of Basic Engineering, 82(Series D):35–45,1960 19
[11] H.W. Kuhn and B. Yaw. The Hungarian method for the assignment problem. Naval Res. Logistics. Quart, pages 83–97, 1955.21
[12] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. CoRR, abs/1311.2524, 2013. URL http://arxiv.org/abs/1311.2524. 10
[13] J. Uijlings, K. Sande, T. Gevers, and A. Smeulders. Selective search for object recognition. International Journal of Computer Vision, 104:154–171, 09 2013. DOI: 10.1007/s11263-013-0620-5. 43
[14] A. Krizhevsky, I. Sutskever, and G. Hinton. Imagenet classification with deep convolutional neural networks. In Proceedings of the 25th International Conference on Neural Information Processing Systems – Volume 1, NIPS'12, pages 1097–1105, USA, 2012. Curran Associates Inc. URL http://dl.acm.org/citation.cfm?id=2999134.2999257.20
[15] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. CoRR, abs/1506.02640, 2015. URL http://arxiv.org/abs/1506.02640.34.
[16] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft. Simple online and real-time tracking. In 2016 IEEE International Conference on Image Processing (ICIP), pages 3464–3468, 2016. DOI: 10.1109/ICIP.2016.7533003.4
[17] Kusuma, T., Dr, A.: Real-Time Object Tracking in H.264/ AVC Using Polar Vector Median and Block Coding Modes. World Academy of Science, Engineering and Technology International Journal of Computer and Information Engineering p. 2018.
[18] Kusuma T, Dr.Ashwini K, "Performance Analysis of Motion Vector Entropy, Smoothed Residual Norm, Markov Random Field in the H.264/AVC Compressed Domain for Object Tracking", Journal of Advanced Research in Dynamical and Control Systems 10(13),366-372,2018.
[19] Chandan Hegde, Dr.Ashwini K, "Measuring the Performance of a model Semantic Knowledge-Base for Automation of Commonsense Reasoning", Intelligent System,253-263,2021.
[20] P. Jaccard. Etude de la distribution florale dans une portion des alpes et du jura. Bulletin de la Societe Vaudoise des Sciences Naturelles, 37:547–579,01 1901. doi: 10.5169/seals-266450.
[21] I. Goodfellow, Y. Bengio, and A. Courville. Deep Learning. MIT Press, 2016. http://www.deeplearningbook.org.
[22] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. CoRR, abs/1409.1556, 2014. URL http://arxiv.org/abs/1409.1556.