International Journal of Engineering Research and Development
e-ISSN: 2278-067X, p-ISSN: 2278-800X, www.ijerd.com
Volume 10, Issue 3 (March 2014), PP.48-54

Consensus algorithm for video tracking using camera networks

M. Sreedhar [1], V. Nagaraju [2], G. Sai Ganesh [3]
[1,2,3] *Department Of Electronics and Communication Engineering, NEC, Narasaraopeta, INDIA*

Abstract:- Networks of video cameras are being installed in many applications, like, surveillance and security, disaster response, environmental monitoring, etc. Currently, most of the data collected by such networks is analyzed manually, a task that is extremely tedious and reduces the potential of the installed networks. Therefore, it is essential to develop tools for analyzing the data collected from these cameras and summarizing the results in a manner that is meaningful to the end user. Tracking and activity recognition are two fundamental tasks in this regard. In this Paper, we developed the methods for tracking and activity recognition in a distributed network of cameras. It tested our approach for tracking in a real camera network with single camera looking over an indoor area of approximately 100 sq. feet. In the area under surveillance, there were two targets in total that were to be tracked using our distributed Kalman-Consensus filtering approach. In our experiment, the measurements are obtained using histogram of gradient (HOG) human detector. The association of measurements to targets is achieved based upon appearance (color) and motion information.

Keywords:- Activity recognition, camera networks, consensus, distributed image processing, tracking.

I. INTRODUCTION

Camera networks are being deployed for various applications like security and surveillance, disaster response and environmental modeling. However, there is little automated processing of the data. Moreover, most methods for multicamera analysis are centralized schemes that require the data to be present at a central server. For many applications, for a number of reasons it is desirable that the video analysis tasks be decentralized. For example, there may be constraints of bandwidth, secure transmission, and difficulty in analysing a huge amount of data centrally. In such situations, the cameras would have to act as autonomous agents making decisions in a decentralized manner. At the same time, however, the decisions of the cameras need to be coordinated so that there is a consensus on the state (e.g., position, activity) of the target even if each camera is an autonomous agent. Thus, the cameras, acting as autonomous agents, analyse the raw data locally, exchange only distilled information that is relevant to the collaboration, and reach a shared, global analysis of the scene. Although there are a number of methods in video analysis that deal with multiple cameras, and even camera networks, distributed processing in camera networks has received very little attention.

Here we will review the current state of the art in camera networks and will see that very few methods are capable of distributed analysis of video. On the other hand, distributed processing has been extensively studied in the multi agent systems and cooperative control literature. Methods have been developed for reaching consensus on a state observed independently by multiple sensors. However, there is very little study on the applicability of these methods in camera networks. In this paper, we show how to develop methods for tracking and activity recognition in a camera network where processing is distributed across the cameras. For this purpose, we show how consensus algorithms can be developed that are capable of converging to a solution, i.e., target state, based upon local decision making and exchange of these decisions (not sensed data) among the cameras. We focus on two problems. For distributed tracking, we show how the Kalman consensus algorithm [1] can be adapted to camera networks taking into account issues like network topology, handoff and fault tolerance. For activity recognition, we derive a new consensus algorithm based upon the recognized activity at each camera and the transition probabilities between various activities. Experimental results and quantitative evaluation for both these methods are presented. The review of scene analysis algorithms will be limited to those directly related to the application domain of camera networks. There have been a few papers in the recent past that deal with networks of video sensors. Particular interest has been focused on learning a network topology [2] [3] i.e., configuring connections between cameras and entry/exit points in their view.

In [4], a distributed target tracking approach using a cluster based Kalman filter was proposed. Here, a camera is selected as a cluster head which aggregates all the measurements of a target to estimate its position using a Kalman filter and sends that estimate to a central base station. Our proposed tracking system differs from this method in that each camera has a consensus-based estimate of the target's state and, thus, there is no need for additional computation and communication to select a cluster head. It apply in a special way the distributed Kalman-Consensus filter which has been shown to be more effective than other distributed Kalman filter schemes. Consensus schemes have been gaining popularity in computer vision applications involving multiple cameras. A related work that deals with tracking targets in a camera network with PTZ cameras is.

Here, the authors proposed a mixture between a distributed and a centralized scheme using both static and PTZ cameras in a virtual camera network environment.

II. INTRODUCTION TO CONSENSUS IN DISTRIBUTED CAMERA NETWORKS

In the multi-agent systems, consensus means that the agents reach an agreement regarding a certain quantity of interest that depends upon the measurements of all sensors in a network. The network may not be fully connected, so there is no central unit that has access to all the data from the sensors. Consequently, a consensus algorithm is an interaction rule that specifies information exchange between a sensor and its neighbours that guarantees that all the nodes reach a consensus.

The interaction topology of a network of sensors is represented using a graph with the set of nodes and edges. Each sensor node maintains an estimate of a quantity. Consensus is achieved when, which is an n-dimensional subspace of. A thorough review of consensus in networked multi-agent systems can be found.

A. Distributed Tracking using DKF

Distributed estimation and tracking is one of the most fundamental collaborative information processing problems in wireless sensor networks (WSN). Multi-sensor fusion and tracking problems have a long history in signal processing, control theory, and robotics. Moreover, estimation issues in wireless networks with packet-loss have been the center of much attention lately. Decentralized Kalman filtering, involves state estimation using a set of local Kalman filters that communicate with all other nodes. The information flow is all-to-all with communication complexity of O(n2) which is not scalable for WSNs. Here it focus on scalable or distributed Kalman Filtering algorithms in which each node only communicates messages with its neighbours on a network. Control-theoretic consensus algorithms have proven to be effective tools for performing network-wide distributed computation tasks such as computing aggregate quantities and functions over networks.

These algorithms are closely related to gossip-based algorithms in computer science literature. Recently, the author introduced a distributed Kalman Filtering (DKF) algorithm that uses dynamic consensus algorithms. The DKF algorithm consists of a network of micro-Kalman Filters (MKFs) [5] each embedded with a low-pass and a band-pass consensus filter. The role of consensus filters is fusion of sensor and covariance data obtained at each node. The DKF algorithm of the author suffers from a key weakness: the algorithm is only valid for sensors with identical sensing models. In other words, it is not applicable to heterogeneous multi-sensor fusion. To be more precise, let

$$zi(k) = Hi(k)x(k) + vi(k) \qquad \ldots\ldots\ldots\ldots\ldots(1)$$

Be the sensing model of node i in a sensor network. Here, x(k) denotes the state of a dynamic process

$$x(k + 1) = Akx(k) + Bkw(k) \qquad \ldots\ldots\ldots\ldots\ldots (2)$$

Driven by zero-mean white Gaussian noise w(k). Then, the DKF algorithm is applicable to sensors with identical Hi's. The reason is that under the assumption of identical observation matrices, or Hi = H, 8i, we get

$$zi(k) = H(k)x(k) + vi(k) = s(k) + vi(k) \qquad \ldots\ldots\ldots\ldots (3)$$

And zi's possess the structure necessary for the distributed averaging feature of the low-pass consensus filter. This limitation of the existing DKF algorithm motivates us to develop novel distributed Kalman filtering algorithms for sensor networks that have broader range of applications.

B. Distributed Activity Recognition

There have been methods on multi-view activity recognition, but the information of multiple views is fused centrally. In this thesis, we propose a framework for distributed activity recognition. Each camera determines a probabilistic measure of similarity of its own observed activities to a predefined dictionary, and then disperses this information to compute a consensus-based estimate with only point-to-point communication between the cameras. We show mathematically how to compute this consensus based upon the similarity score computed at each camera and the transition probabilities between activities (can be uniform if no prior information is available).

It consider the problem of recognizing actions using a priori unknown camera configurations. Action recognition has received considerable attention over the past decades, as a result of the growing interest for automatic and advanced scene interpretations shown in several applications domains, *e.g*. video-surveillance or human machine interactions. In this field, two main directions have been followed. Model based approaches, e.g. Assume a known parametric model, typically a kinematic model, and represent actions in a joint or parameter space. Unfortunately, recovering the parameters, e.g. the pose, of the model appears to be a difficult intermediate task without the help of landmarks. In contrast, template based or holistic approaches. Do not use such an intermediate representation and directly model actions using image information, silhouettes or optical flow for instance. Action templates are then spatio-temporal shapes either in a three-dimensional space, when a single camera is considered, or in a four dimensional space when multiple calibrated cameras are considered. In both cases, action recognition is achieved by comparing a motion template, built from observations, with

learned models of the same type. This limits recognition to situations where observed and learned models are obtained using similar camera configurations. In this work, we propose an approach that takes advantage of the template based methods but that does not constrain camera configurations during recognition. Instead, actions can be observed with any camera configuration, from single to multiple cameras, and from any viewpoint.
Our main motivation is to be able to cope with unknown recognition scenarios without learning multiple and specific databases. This has particularly clear applications in video surveillance where actions are often observed from a single and arbitrary viewpoint. To this purpose, we propose an exemplar-based hidden Markov model (HMM) inspired by the works of Frey and Jojic and Toyama and Blake. This model accounts for dependencies between three dimensional exemplars, *i.e*. representative pose instances, and image cues, this over time sequences. Inference is then used to identify the action sequence that best explains the image observations.

III. DISTRIBUTED TARGET TRACKING USING KALMAN CONSENSUS FILTERING

In this section, It present the first major result of this paper, how to track multiple targets in a camera network using a consensus algorithm that relies on the tracks obtained at individual cameras. For this purpose, it leverage upon the Kalman-Consensus algorithm in the distributed processing and multi-agent systems literature [2]. However, there are some major differences due to the nature of cameras, and we show how to handle them. Cameras are directional sensors and, thus, geographically neighboring cameras may be viewing very different portions of the scene. On the other hand, cameras that are geographically far away may be observing the same target. Therefore, we can define a target-based network topology, where the neighborhood structure is defined with respect to each target. Since targets are dynamic, this target-based topology changes over time.

A. Kalman filtering technique

In statistics, the Kalman filter is a mathematical method named after Rudolf E. Kalman. Its purpose is to use measurements observed over time, containing noise (random variations) and other inaccuracies, and produce values that tend to be closer to the true values of the measurements and their associated calculated values. The Kalman filter has many applications in technology, and is an essential part of space and military technology development. A very simple example and perhaps the most commonly used type of Kalman filter is the phase-locked loop, which is now ubiquitous in FM radios and most electronic communications equipment.

B. Distributed Kalman filter

First, we present a modified version of the DKF algorithm. The key modification is to replace the low-pass and band-pass consensus filters in the architecture of the microfilter by high-gain versions of the high-pass consensus filter. The resulting microfilter is shown in Figure.1.

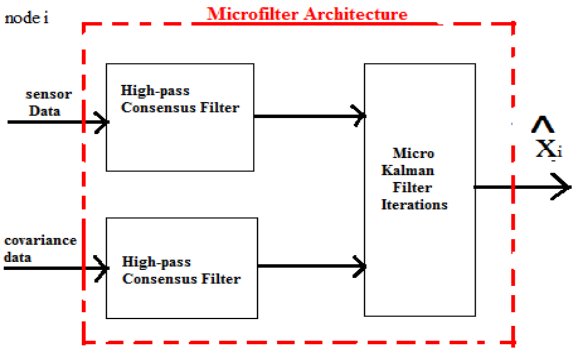
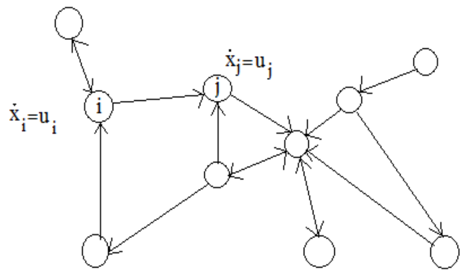


Fig.1: The architecture of the microfilter of the type-I DKF algorithm.

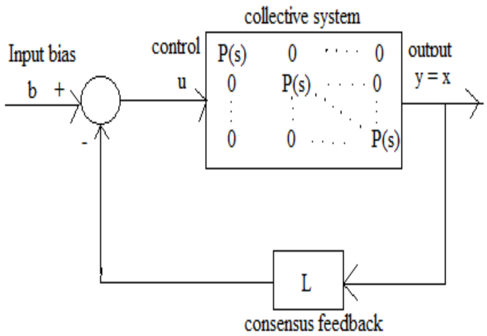
In the following, we provide the equations of the micro-Kalman filter (MKF) and the high-pass consensus filters (CFs).

C. Consensus and Cooperation for multi agent systems

Distributed computation over networks has a tradition in systems and control theory starting with the pioneering work of Borkar and Varaiya and Tsitsiklis and Tsitsiklis, Bertsekas, and Athans on asynchronous asymptotic agreement problem for distributed decision making systems and parallel computing. Further theoretical extensions of this work were presented and with a look toward treatment of directed information flow in networks as shown in Figure.2 (a).



a) A network of integrator agents



b) A network of interconnected dynamic systems

Fig.2: Two equivalent forms of consensus algorithms: (a) a network of integrator agents in which agent i receives the state xj of its neighbor, agent j, if there is a link ði; jÞ connecting the two nodes; and (b) the block diagram for a network of interconnected dynamic systems all with identical transfer functions PðsÞ ¼ 1=s. The collective networked system has a diagonal transfer function and is a multiple-input multiple-output (MIMO) linear system. The common motivation behind the work is the rich history of consensus protocols in computer science, whereas Jadbabaie et al. attempted to provide a formal analysis of emergence of alignment in the simplified model of flocking by Vicsek et al.

VI. SYSTEM DESIGN FOR TRACKING

The system design model having MATLAB SIMULINK design for the video tracking and detection.

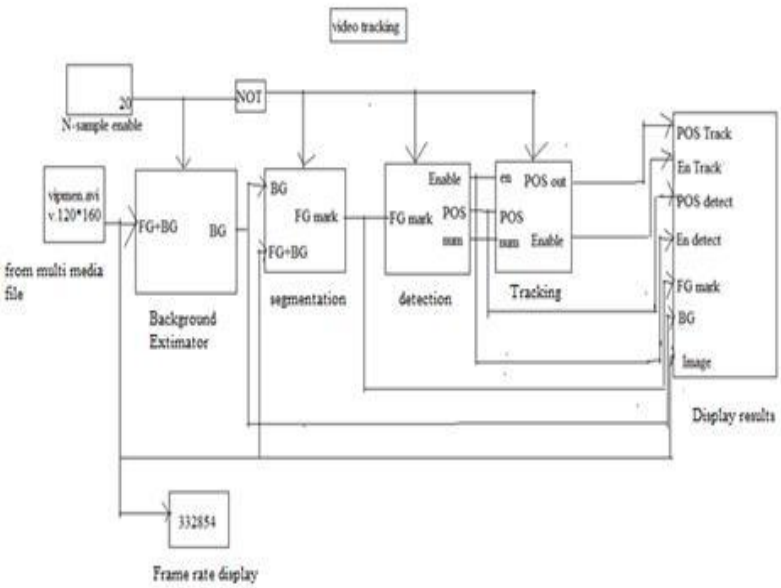


Fig.3: Shows the total design model

The design contains Simulink blocks for individual functionality. In a distributed target tracking approach using a cluster based Kalman filter was proposed. Here, a camera is selected as a cluster head which aggregates all the measurements of a target to estimate its position using a Kalman filter and sends that estimate to a central base station. Our proposed tracking system differs from this method in that each camera has a consensus-based estimate of the target's state and, thus, there is no need for additional computation and communication to select a cluster head. It present the first major result of this thesis how to track multiple targets in a camera network using a consensus algorithm that relies on the tracks obtained at individual cameras. For this purpose, it leverage upon the Kalman-Consensus algorithm in the distributed processing and multi-agent systems. However, there are some major differences due to the nature of cameras, and we show how to handle them. Cameras are directional sensors and, thus, geographically neighbouring cameras may be viewing very different portions of the scene. On the other hand, cameras that are geographically far away may be observing the same target.

V. TEST RESULTS

It tested our approach for tracking in a real camera network with single camera looking over an indoor area of approximately 100 sq. feet. In the area under surveillance, there were two targets in total that were to be tracked using our distributed Kalman-Consensus filtering approach. In our experiment, the measurements (i.e., the observed positions of targets) are obtained using histogram of gradient (HOG) human detector [6]. The association of measurements to targets is achieved based upon appearance (color) and motion information. Figure.4 to Figure.7 shows the tracking results as viewed by each camera at four times instant.

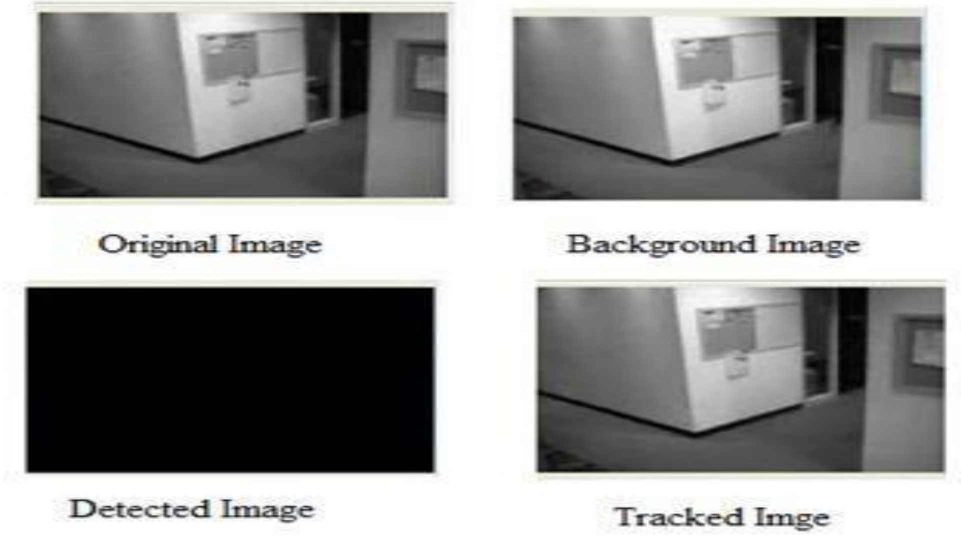


Fig.4: Each subfigure shows one camera at one of four time instants denoted by T. The track of one target, marked with a box. All targets are tracked using the Kalman-Consensus filtering approach, but are not marked for clarity. Tracked image at time T=0 sec.

Fig.5: Each subfigure shows one camera at one of four time instants denoted by T. The track of one target, marked with a box. All targets are tracked using the Kalman-Consensus filtering approach, but are not marked for clarity. Tracked image at time T=2.2 sec.

Fig.6: Each subfigure shows one camera at one of four time instants denoted by T. The track of one target, marked with a box. All targets are tracked using the Kalman-Consensus filtering approach, but are not marked for clarity. Tracked image at time T=5.5 sec.

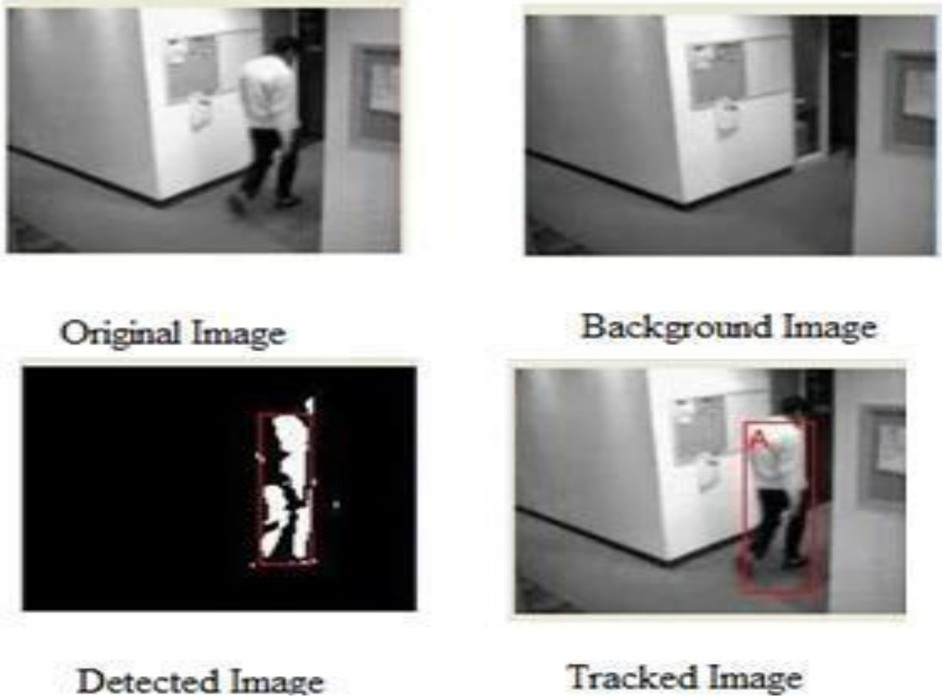


Fig.7: Each subfigure shows one camera at one of four time instants denoted by T. The track of one target, marked with a box. All targets are tracked using the Kalman-Consensus filtering approach, but are not marked for clarity. Tracked image at time T=8.2 sec.

The association of measurements to targets is achieved based upon appearance (color) and motion information. Figure.4 to Figure.7 shows the tracking results as viewed by each camera at four time instants. The results are shown on a nonstatic camera network. The cameras are controlled to always cover the entire area under surveillance through a game theoretic control framework it proposed in. As explained previously, the change of camera settings does not affect the procedure of the Kalman-consensus filter. As the targets are observed in this area, the single-view tracking module in each camera determines the ground plane position of each target in its FOV and sends that information to the Kalman-Consensus filter which processes it together with the information received from the Kalman-Consensus filters of neighboring.

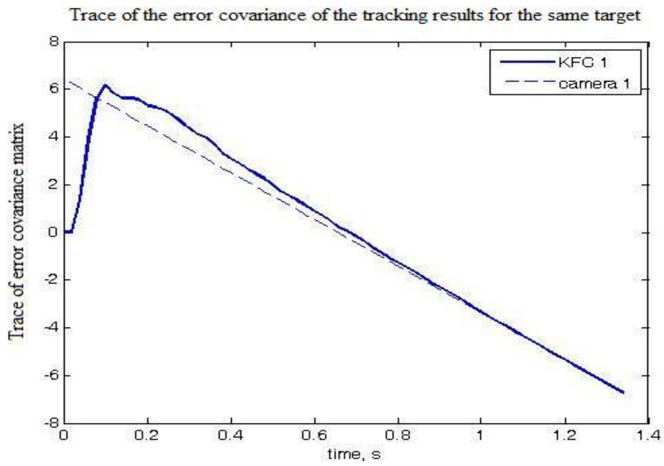


Fig.8: The traces of the covariance matrix of the tracking error for the same targets

In order to measure tracking performance, we compare the tracking results with the ground truth trajectory, which is shown in Figure.8. It can be seen that KCF1 performs best and KCF2 is better than individual camera tracks. It also look at the output error covariance matrix of the Kalman filter. The higher the trace of is, the lower the tracking accuracy is. Figure.8 shows the traces of the covariance matrix of the tracking error for the same targets. The colored lines with symbols correspond to tracking results from different cameras using their own measurements only (as each camera runs an independent Kalman filter), while the solid black line is the result of consensus-based estimate for the fully connected case (which will be the same for the centralized case) and dashed purple line is for the partially connected one. As can be seen clearly, the Kalman-Consensus filter with full connection performs the best, and partially connected one does better than individual Kalman filters without consensus.

VI. CONCLUSION AND FUTURE EXTENSION

It investigated in this thesis distributed scene analysis algorithms by leveraging upon concepts of consensus. We addressed two fundamental tasks, tracking and activity recognition in a distributed camera network. It proposed a robust approach to distributed multi-target tracking in a network of cameras. A distributed Kalman-Consensus filtering approach was used together with a dynamic network topology for persistently tracking multiple targets across several camera views. A probabilistic consensus scheme for activity recognition was provided, which combines the similarity scores of neighboring cameras to come up with a probability for each action at the network level. In the future, It integration of tracking and activity recognition into a single framework and more complex activities that span a larger area.

REFERENCES

[1]. R. Olfati-Saber, “Distributed Kalman filtering for sensor networks,” *in Proc. IEEE Conf. Decision and Control, 2007, pp. 5492–5498.*

[2]. D. Markis, T. Ellis, and J. Black, “Bridging the gap between cameras,” *in Proc. IEEE Conf. Computer Vision and Pattern Recognition, 2004, vol. II, pp. 205–210.*

[3]. K. Tieu, G. Dalley, and W. Grimson, “Inference of non-overlapping camera network topology by measuring statistical dependence,” *in Proc. Int. Conf. Computer Vision, 2005, pp. 1842–1849.*

[4]. H. Medeiros, J. Park, and A. Kak, “Distributed object tracking using a cluster-based Kalman filter in wireless camera networks,” *IEEE J. Sel. Topics Signal Process., vol. 2, no. 4, pp. 448–463, Aug. 2008.*

[5]. S. Khan and M. Shah, “A multiview approach to tracking people in crowded scenes using a planar homography constraint,” *in Proc. Eur. Conf. Computer Vision, 2006, pp. 133–146.*

[6]. N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” *in Proc. IEEE Conf. Computer Vision and Pattern Recognition, 2005, vol. I, pp. 886–893.*