2020

# Selective Subtraction: An Extension of Background Subtraction

Adeel Bhutta
*University of Central Florida*

# *SELECTIVE SUBTRACTION*: AN EXTENSION OF BACKGROUND SUBTRACTION

by

ADEEL ASLAM BHUTTA
M.S., Computer Engineering, University of Central Florida, 2012
M.S., Computer Science, University of Central Florida, 2006
B.S., Electronic Engineering, Ghulam Ishaq Khan Institute of Engr Sc and Tech, Pakistan, 1999

A dissertation submitted in partial fulfilment of the requirements
for the degree of Doctor of Philosophy
in the Department of Electrical and Computer Engineering
in the College of Engineering and Computer Science
at the University of Central Florida
Orlando, Florida

Summer Term
2020

Major Professor: Hassan Foroosh

# ABSTRACT

Background subtraction or scene modeling techniques model the background of the scene using the stationarity property and classify the scene into two classes of foreground and background. In doing so, most moving objects become foreground indiscriminately, except for perhaps some waving tree leaves, water ripples, or a water fountain, which are typically "learned" as part of the background using a large training set of video data. Traditional techniques exhibit a number of limitations including inability to model partial background or subtract partial foreground, inflexibility of the model being used, need for large training data and computational inefficiency. In this thesis, we present our work to address each of these limitations and propose algorithms in two major areas of research within background subtraction namely single-view and multi-view based techniques. We first propose the use of both spatial and temporal properties to model a dynamic scene and show how Mapping Convergence framework within Support Vector Mapping Convergence (SVMC) can be used to minimize training data. We also introduce a novel concept of background as the objects *other than* the foreground, which may include moving objects in the scene that cannot be learned from a training set because they occur only irregularly and sporadically, e.g. a walking person. We propose a "selective subtraction" method as an alternative to standard background subtraction, and show that a reference plane in a scene viewed by two cameras can be used as the decision boundary between foreground and background. In our definition, the foreground may actually occur *behind* a moving object. Our novel use of projective depth as a decision boundary allows us to extend the traditional definition of background subtraction and propose a much more powerful framework. Furthermore, we show that the reference plane can be selected in a very flexible manner, using for example the actual moving objects in the scene, if needed. We present diverse set of examples to show that: (i) the technique performs better than standard background subtraction techniques without the need for training, camera calibration, disparity map estimation, or special camera con-

figurations; (ii) it is potentially more powerful than standard methods because of its flexibility of making it possible to select in real-time what to filter out as background, regardless of whether the object is moving or not, or whether it is a rare event or a frequent one; (iii) the technique can be used for a variety of situations including when images are captured using stationary cameras or hand-held cameras and for both indoor and outdoor scenes. We provide extensive results to show the effectiveness of the proposed framework in a variety of very challenging environments.

Dedicated to my caring and loving family especially my kind parents whose love and prayers keep me going every day, and to my loving wife who has been there with me through everything.

# ACKNOWLEDGMENTS

I want to thank my advisor Dr. Hassan Foroosh who always believed in me and was there for me to provide guidance and support through thick and thin. This would not have been possible without his help and counsel.

I would also like to thank all members of my dissertation committee for their help and constant support. In specific, I would like to acknowledge Dr. Parveen Wahid (for her kindness), Dr. Charles Hughes (for his invaluable advising), Dr. Samuel Richie (for giving me many teaching opportunities and invaluable feedback), and Dr. Yuanwei Qi (for his support and countless fun moments on and off badminton court) throughout my time at UCF. I also want to acknowledge Dr. Max Poole not only for his ever-present counsel but also for his dedication to all graduate students in UCF and beyond.

I want to acknowledge countless others at UCF for helping me get a full learning-experience during my long years at this wonderful institute. I was fortunate to be part of so many great experiences in and out of school during my time in Orlando and I will cherish these moments forever!

Finally, I want to thank Khawar Hassan, Dr. Ashar Ahmed and Dr. Imran Junejo for their constant encouragement and ever-present friendship.

# TABLE OF CONTENTS

# LIST OF FIGURES

# LIST OF TABLES

# CHAPTER 1: INTRODUCTION

Background subtraction is the fundamental step used in many applications including object detection, tracking, gesture and action recognition, activity recognition, and user interfaces. Background subtraction or scene modeling techniques traditionally use one or more views to classify the objects (or image pixels) as either foreground or background. However, standard methods have a rigid definition of what constitutes a background - pixels or objects that remain static, stationary or don't change over a period of time - which often leads to classifying almost all moving objects as foreground, except for small persisting motions that can be learned from a training set. This loss of 'intra-class separability' results in inability to model partial background or partial foreground and thus the notion of a background object being *in front* of a foreground object. Moreover, none of the current techniques allow 'inter-changeability' of classification where an object (or pixel) classified as foreground can later be classified as background or vice versa. If scene modeling is to be made more effective, the background subtraction techniques need to offer a framework that ensures that the statistical models not only allow learning partial backgrounds or foregrounds and thus preserving of intra-class taxonomy but also allow backgrounds to be classified as foregrounds when desired. Such framework can prove very useful in many real world applications such as object detection, action and activity recognition in surveillance videos, tracking in crowds, and accident prevention.

Existing background subtraction techniques can be classified into two main categories: techniques using monocular sequences and those using stereo sequences. The work presented in this dissertation contributes to both areas of research. In our first work, we studied scene modeling problem for single view based sequences as a Single-Class Classification (SCC) problem and proposed the use of single-class SVM. Our primary motivation has been to reduce the burden of extensive training required in most background subtraction techniques. In our later work, we proposes Selective

Subtraction, a novel method that works on multi-view sequences and is free from a number of limitations such as rigid cameras configuration or use of disparity maps and only requires only two frames as training data.

Most of the existing literature focuses on aspects such as the statistical approach used to model the background, type of scene used (dynamic or static), the learning method applied to the training set, and the model used for the background or foreground. The background of a scene is generally defined as being *motionless* for static scenes (e.g., video conference) and *almost-motionless* for dynamic scenes (e.g., scenes which include changes such as illumination, shadows, waving tree leaves, water ripples, or fountains). Most single-view background subtraction techniques try to model the background (and the dynamic changes) either by modeling each pixel or different regions statistically, and then use those statistical models to detect the moving objects, known as foreground. This modeling requires large amount of training data for learning the statistical properties of the background. Alternatively, stereo-based techniques rely on estimating disparity maps by rectifying the views and using similarity measures in order to estimate the background. Such disparity maps are, in practice, difficult to estimate in real-time and very error prone. Also, these techniques require special camera setup and are computationally expensive. Furthermore, all background subtraction techniques classify moving objects as foreground indiscriminately. Consider a case when you have a street with multiple objects moving across the camera in both directions. The object closer to the camera occludes the object crossing behind it which, in turn, is occluding another object crossing behind it and so on. Any standard background subtraction technique will consider all of the moving objects as foreground thus will not be able to selectively distinguish which moving object should be kept as foreground and which ones discarded. What if you are only interested in the first two objects closest to the camera, or only one object at the back, and all other objects are irrelevant. Thus, the *foreground-of-interest* is now the partial foreground while *background-of-interest* is a combination of traditional background and partial foreground. In

this context, the standard definition of background is insufficient. Current background subtraction techniques fail to model such backgrounds.

The work presented here has six novel contributions. Firstly, most background subtraction techniques require training or learning of the background model using the data consisting of background alone. Even when such data is available, these techniques cannot learn the partial background as defined above. We challenge the requirement of extensive training and propose techniques that either use features that minimize the need for large training data or propose the use of a *reference plane* inducing a *base homography*, estimated using only two frames. This base homography can be used in the background subtraction of the scene when traditional technique fail, because they cannot classify an infrequently occurring moving object as background. This allows us to have a notion of background being *in front of* the foreground which is not possible in traditional techniques. Secondly, we propose to use the actual moving objects in the scene to estimate the base homography and show how a simple *walk* (or an object in motion) can be used to define a reference plane. Thirdly, most background subtraction techniques need sufficient amount of data to model the background (which usually ranges hundreds to several hundreds of frames). We first propose and show how Mapping Convergence framework within Support Vector Mapping Convergence (SVMC) can be used to minimize the need for training data. We further propose and show that the base homography can be estimated using an object in motion viewed only in two frames. Thus the presence of large amount of training data is no longer required in our methods. Fourthly, standard background subtraction techniques fail to change the background model once it is learned. Only some minor dynamic changes are incorporated in the updating of the background model. In our technique, the base homography can be modified using a different moving object or a plane in the scene in real time, and can be replaced altogether with a new base homography, thus providing flexibility in the background subtraction. Fifthly, we avoid the explicit use of depth map and the requirement of rectifying two views for calculating depth as in other stereo-based methods,

and propose a solution based entirely on projective depth which makes our technique more flexible, reliable and computationally efficient. Lastly, we show that our proposed frameworks can be used with any other classical background subtraction approach thus making this framework truly exciting.

# CHAPTER 2: BACKGROUND AND RELATED WORK

In this section, we provide a brief overview of Background Subtraction and summarize the most relevant work done over the last few decades.

## 2.1 Background Subtraction

Background subtraction or scene modeling is the process of discriminating foreground from background. It is a fundamental step used in a wide range of applications including object detection, recognition, tracking, activity recognition, surveillance, video transmission, and many more and is often considered an essential pre-processing step. Much of the background subtraction research overlaps the research done in areas including motion detection, background modeling, motion segmentation, and object detection. Almost all techniques see background subtraction as a binary classification problem where each pixel (or region) is either foreground or background. Foreground is defined as pixels (or region) that are moving (or non-static) and background is everything else as seen in Figure 2.1. This rigid definition of foreground and background has been universally adopted even in cases that involve dynamic moving objects that appear rarely (such as moving tree leaves or water waves). This approach has given birth to the notion that the foreground object is always *in front* of a background object.

Background subtraction is inherently a highly challenging task and has been an active area of research over many decades. A major factor in the success of any background subtraction algorithm is its ability to capture significant changes in the video as well as ignoring 'noisy' changes. Some of these changes are due to factors such as shadows, illumination, weather, motion (including dynamic motion), camera position and configuration. A large portion of relevant research

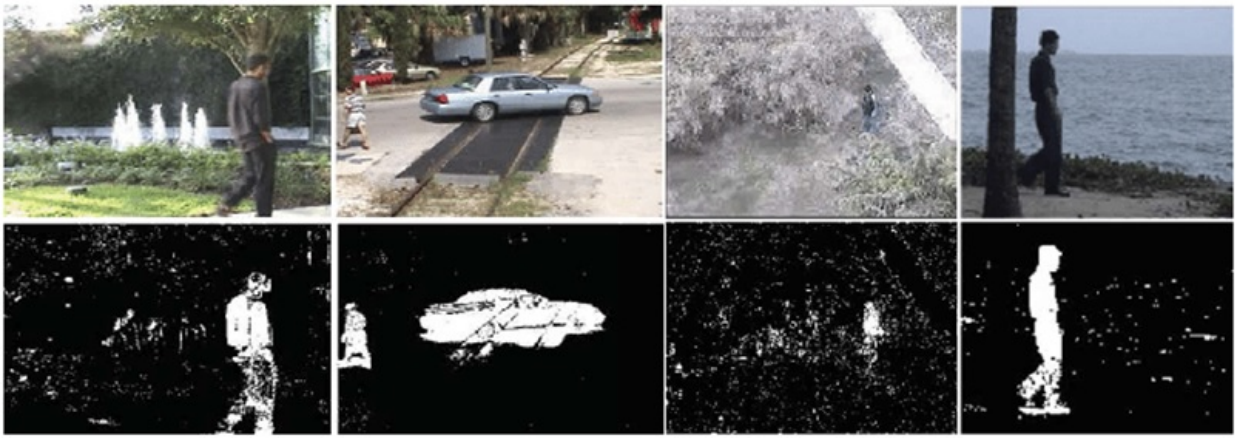


Figure 2.1: Examples of Background Subtraction.

focuses on designing algorithms with better accuracy despite prevalent changes in videos. It is beyond the scope of this work to review all the methods and techniques, hence we refer the reader to [6, 9, 29, 52, 54, 62] for a good review of the related work in this area. A quick review of highly cited background subtraction techniques (for both single camera and multiple cameras has been presented below.

## 2.2 Single-View Background Subtraction

The earliest research in background subtraction focused on images and videos from single camera. The idea of defining foreground as non-moving object in a static scene has been used in background subtraction, object tracking [28, 82], action and activity recognition in surveillance videos [37, 39, 65, 66], video summarization and image/video context description [67, 68] for a long time. In order to improve the results in real-world scenarios, dynamic background subtraction techniques use a single three-dimensional Gaussian distribution to model each pixel in the scene [75] or a Mixture of Gaussian (MoG) [23] or a non-parametric kernel density estimation (KDE) [21]. Other

techniques take a region based approach or utilize low-rank subspace decomposition which has also been used, more recently, in deep learning based approaches for modeling backgrounds.

[23] was the first to propose learning a Mixture-of-Gaussian classification model (GMM) from fixed number of components for each pixel using an unsupervised technique i.e., an efficient, incremental version of EM. This approach was also very successful in eliminating shadows. [64] later improved this work by modeling each pixel as a Mixture-of-Gaussian and using an on-line approximation to update the model. This approach produced highly stable and real-time outdoor tracker that was able to reliably deals with lighting changes, repetitive motions from clutter, long-term scene changes through rain and snow. [85] later proposed automatic selection of the number of components of the mixture model for each pixel and this approach was faster and was able to process gradual changes more effectively. [71] extended the prior work to model each pixel by a layer of 3D multivariate Gaussian (i.e., a multi-layer Gaussian mixture model) and this approach was able to process dynamic scenes more effectively. KDE models have also been proposed to address parametric estimation and update in GMMs. [21] proposed the use of fixed size window of most recent frames along with Normal function to estimate probability density function for each pixel. [86] combined the use of variable size windows with adaptive kernel sizes for density estimation. Their use of recursive equations was later adapted by [84] to estimate the local maximum in density function using mean shift method. This approach also improved the segmentation results by incorporating local texture information from neighboring pixels.

Region based techniques have also been proposed to improve background subtraction which try to use a covariance matrix from a region around a pixel [81] or auto-regression models [51] or propose the use of temporal persistence with single probability density in a Maximum A Posteriori in the Markov Random Field (MAP-MRF) selection framework [60] to model the spatial and appearance attributes. Other techniques have used principal component analysis (PCA) to reduce dimensionality of the space in order to estimate the mean background image [19, 53]. See [5, 23,

24, 41, 42, 50, 55, 58, 69, 83] for review of other single view methods.

Deep learning, and convolutional neural network (CNN) in general, has made its impact on background subtraction in recent years. Based on deep learning networks, [11] performs semantic segmentation on the images where pixel level information is leveraged for motion detection in the video sequence. Pixels with low semantic probability are deemed as background. In order to reduce any false negatives, a semantic background model is maintained at each pixel as well. In case of ambiguity, any background subtraction method can be used in their method as the final step. Other approaches have proposed dividing an input image into patches [3] and have used SuBSENSE algorithm [63] combined with Flux Tensor algorithm [74] to create a background image where CNNs are often fed with matching pairs of patches from background and the input image. [17] combined a standard background subtraction method with features learned from CNNs for applications in the field of agriculture. It is often hoped that these feature would be robust to camera motion and view changes, and sensitive to any new elements in the area.

Another approach proposed by [18] has computed pixel-wise segmentation map and used an encode-decoder framework where the input image is temporally aligned to the reference image. [77] introduced an atrous convolution to expand the receptive field of the network and added shortcut connections Mimicking res-net, to reduce training complexity. [43] proposed a triplet convolutional neural network along with an encoder-decoder type network and they utilized pretrained VGG-16 Net where each branch of this triplet network operates on different scale to perform feature encoding and the decoding is performed by the transposed convolutional network. This method, however, works on an image at a time, not utilizing any temporal information. In order to utilize the temporal information, [16] proposes a deep end-to-end framework where pixel-wise semantic features are extracted using an encoder-decoder network and Long Short-Term Memory networks (LSTMs) are used next to model pixel-wise changes overtime. In order to reduce sensitivity to camera motion, Conditional Random fields (CRF) are used in the last layer.

In order to fully capture the temporal information of a scene, a 3D CNN is proposed by [59]. Their specific 3D-CNN consists of 6 convolutional layers and the input is a window of 10 consecutive frames. These 10 frames are divided into a group of 4 frames and fed to 4 convolutional layers. Up-sampling is performed using kernels of various strides to retain the fine information from the input images, these layers are then concatenated to produce the final predication layer. See [10] for review of other deep-learning based methods.

## 2.3 Multi-View Background Subtraction

An alternate approach and the one most related to the technique presented in this dissertation, is based on stereo, which attempts to recover dense disparity maps in real time for segmenting the scene. [27] used stereo cameras and their disparity maps to perform background subtraction by checking the color intensity values of corresponding pixels. Each pixel was warped to the corresponding pixel in the reference image and the color and luminance was used to decide if the pixel belongs to the foreground or background. This method suffers from false and missed detections. [44] proposed the use of a stereo configuration, in which the cameras are vertically aligned, to improve the background subtraction. A multi-view approach is proposed by [20] to remove static background using two methods, one with rough camera localization and other with accurate camera localization. For the first method, they used scene-specific pre-trained background model (using SVMs) to perform foreground extraction. For their second approach, multi-view stereo approach is employed to perform a dense matching (using Structure from Motion technique) of the scene with data-set of existing images to remove static background. Major limitation of this approach is that scene-specific trained & labeled data-set is very expensive to acquire and SfM is known to be noise prone.

[25] proposed detecting out of plane objects. First, a stereo image pair is used offline to compute

the planar homography between them. During the test phase, one image is super-imposed on the other using the pre-computed homography and then a similarity map is created. A similarity map is created to detect out of plane objects, as pixels corresponding to a background have specific values (close to 1). The background pixels, on the other hand, have low values in the similarity map. Another two-view based hierarchical algorithm is proposed by [45] where stereo images are decomposed using the Discrete Wavelet Transform (DWT). Adaptive models are built over sub-bands at each level and a depth based model is also created, which is applied to pixels that do not conform to the adaptive model. However, DWT is an expensive process and is known to be effected by noise.

There are several major limitations of these techniques: color and luminance is not sufficient to decide if the pixels belong to foreground or background especially when objects are roughly similar in color. Furthermore, the cameras need to be in strict configurations to have sufficient accuracy or a dense disparity map is required for most techniques. We challenge the use of color and luminance values as well as the requirement of strict camera configurations or use of disparity maps and show that the background subtraction can be performed using our proposed technique without these requirements or limitations.

# CHAPTER 3: SCENE MODELING USING SUPPORT VECTOR MACHINE

A large amount of literature in this area of research uses single camera views. For single camera views, we studied scene modeling problem as a Single-Class Classification (SCC) problem and proposed the use of single-class SVM. SCC aims to distinguish one class of data from the universal set of multiple classes. Our primary motivation has been to reduce the burden of extensive training required in most background subtraction techniques. Without requiring a large amount of data, Single-Class SVM classify one class of data from the rest of feature space given only positive data by drawing a optimum non-linear boundary of the positive data set in the feature space. In addition, we use a novel set of region based features to capture the dynamics of the background. These features not only capture the dynamics at each pixel, but also capture the spatial context of the region surrounding a pixel. In essence we combine the use of both spatial and temporal properties to model a dynamic scene [30, 35].

## 3.1 Dynamic Scene Modeling

In a dynamic scene, every pixel in the image is undergoing a certain periodic or a repetitive change in intensities at each time instance. It is too simplistic to assume that a pixel intensity varies independently of its neighbors [60]. For example, in a typical scene with swaying trees or water ripples, such as Figure 2.1 a larger region of the image, not just a single pixel, is involved in the same type of motion. At the same time, there is a temporal continuity in the motion, as in the case of swaying trees, where branches or the leaves move back and forth. Thus it is essential that both the spatial and the temporal context be captured for an accurate scene modeling.

Let $\{\mathbf{I}(t)\}^{t=1...k}$ be a set of images. In order to model the background, we first compute the optical flow by using Lucas and Kanade method [47] on the whole image using two consecutive frames and generate their representations, i.e. the $v_x$ and $v_y$ components of optical flow such that: $\mathscr{F} = \{v_x, v_y\}$. The idea is to extract a set of features that uniquely capture the dynamics of the scene by using these representations.

### *3.1.1 Feature Set*

Once we have computed the optical flow, for every pixel $p_i^t$ in the image, i.e. the $i^{th}$ pixel in image $t$, a rectangular region of the size $M \times N$ is used to compute the following set of simple features:

**Entropy:** The standard way of defining entropy is,

$$h_i = -\sum_{m=1}^{k} \mathscr{P}_i \log(\mathscr{P}_i) \tag{3.1}$$

where $k$ refers to the number of histogram bins and $\mathscr{P}_i$ refers to the histogram count of $\mathscr{F}_i$ for $M \times N$ region around the $i^{th}$ pixel $p_i^t$. Generally this is set to be $5 \times 5$ in our experiments. The entropy $h$ is a statistical measure of the *randomness* that can be used to characterize the flow vectors.

**Energy:** The energy of flow vectors in an $M \times N$ region surrounding the $i^{th}$ pixel is computed as:

$$e_i = \sum_{u=1,v=1}^{M,N} (\mathscr{F}_i)^2 \tag{3.2}$$

where $\mathscr{F}_i$ refers to flow vectors as defined above and $u$,$v$ refer to the pixel location. $e$ measures the energy presented in the flow vectors in an $M \times N$ region around a pixel.

**Inertia:** Finally, we define the inertia as,

$$j_i = \sum_{u=1,v=1}^{M,N} (u-v)^2 \mathscr{F}_i \tag{3.3}$$

where $\mathscr{F}_i$ refers to flow vectors and $u$,$v$ refer to the pixel location. $j$ measures an object's resistance to changes in its rotation rate.

The features defined above are unique, and yet simple to compute. Entropy, inertia and energy are relatively immune to *rotation*, since the order is not important. These measures are *scale* invariant, and are inherently invariant to linear change in *illumination* as well. The output at this stage is a 6-dimensional feature vector

$$H^{p_i^t} = \{\{h_i, e_i, j_i\}_{v_x}, \{h_i, e_i, j_i\}_{v_y}\} \tag{3.4}$$

for every pixel $p_i^t$ in the frame $t$. This feature vector represents a set of features that uniquely capture the dynamics of the scene.

### *3.1.2 Single-Class Classification*

The scene modeling problem involves observing a scene which is assumed to contain an acceptable behavior. During this phase, which is generally termed as the training phase, it is possible to only gather the *positive data* that describes what belongs to the scene. However, during this phase it is not possible to include the negative data which is to be detected at a later time. This scenario is a good candidate for applying the Single Class Classification techniques.

Given a limited amount of training data, the *optimal* class boundary function is the one that gives

**Algorithm 1:** Scene Modeling using Single-Class SVM

1. **Train SVM**:
   * Using training sequence, generate flow components, $\mathscr{F}_i = \{v_x, v_y\}$ for each pixel.
   * For each pixel $p_i^t$, compute $H^{p_i^t}$ as defined in (3.4)
2. **Testing**:
   * For a test sequence, generate flow components, $\mathscr{F}_i = \{v_x, v_y\}$ for each pixel.
   * In a $5 \times 5$ window around each pixel $p_i^t$, compute $H^{p_i^t}$ as defined above.
   * Detect foreground and background pixels using SVMC framework.

the best generalization performance representing the performance on unseen examples. For supervised learning, SVM tries to maximize the generalization by maximizing the margin and supports nonlinear separation using advanced kernels; thus avoiding under-fitting and over-fitting [78].

More specifically, we adopt the Support Vector Mapping Convergence (SVMC) as proposed by [78], which employs the Mapping Convergence framework where the algorithm generates the boundary close to the optimum. As the sample size increases, SVMC prevents training time from increasing dramatically, and the running time is shown to be asymptotically equal to that of a SVM. The approach is to use minimally required data at each iteration so that the data does not degrade the accuracy of the boundary. In their work, [78] prove that the training time is $O(n^2)$, where $n$ is the size of the training data. Thus for training on data set of size $K$ images, we compute the feature vector $H^{p_i} = \{H^{p_i^1}, H^{p_i^2}, ..., H^{p_i^K}\}$ for each pixel location. This feature vector is used to train the SVMC at each pixel location. The complete algorithm is given in Algorithm 1.

SVMC has been shown to have a good accuracy for single class classification by computing accurate classification boundary around the positive data (during the training phase) using the unlabeled data in a systematic way. Moreover, SVMC does not require a large amount of positive training data while still maintaining performance close to that of original SVM while providing good gen-

eralization, as the results in the next section show.

## 3.2 Experiments and Results

We tested our method on two dynamic natural scenes from [60]; from here on we will refer to them as the *fountain* and the *railway* sequence. These sequences contain nominal camera motion, significant dynamic textures and cyclic motion. In the *fountain* sequence, the dynamic texture is induced in the scene by the moving trees while the fountain in the background induces constant cyclic motion. The *railway* sequence contains periodic self-occlusion of a walking person followed by occlusion by a passing car. We compare our method with the Mixture of Gaussian (MoG) approach [64], Principle Component Analysis based approach (Eigen) [53], Median filtering based approach (Mediod) [12, 56], Adjacent Frame Difference (FD) approach [70], and color distributions based approach [42]. We train MoG model using three (3) color components and use 400 and 270 frames for *fountain* and *railway* sequences respectively. For our method, we only used 75 frames for feature extraction and training the Single-class SVM, as described in the Section 3.1. FD method was implemented as proposed in [70] and uses threshold value equal to 0.06. Eigen and Medoid methods are implemented as described in [56] and [53] respectively. The images have a resolution of $360 \times 240$.

### *3.2.1 Qualitative Analysis*

Qualitatively, the results are an improvement over the methods [12], [53], [64] and [70], as shown in Figure 3.1. The camera is mounted on a tall tripod, and the wind causes the tripod to sway back and forth; and in the background is a water fountain and swaying trees. The first row shows the original images from the test sequence. The figure depicts a person coming in from the left

of the image and walking to the right. The second row shows the results obtained from the MoG method and it becomes evident that the nominal motion caused by the camera and the presence of the water fountain, causes substantial degradation of the results qualitatively. A large number of moving background pixels are detected as foreground pixels. Some portions of the foreground object are also classified as background. The third and fourth rows show the results obtained from Median Filtering Based approach (Mediod) and Principle Component Analysis based approach (Eigen) respectively. The results from both Mediod and Eigen approaches show the same behavior as MoG where large number of background pixels are classified as foreground. The fifth row shows the results obtained by our method, showing a considerable improvement over MoG [64], Medoid [12], and Eigen [53]. The last row shows the ground truth frames obtained by manually labeling some frames from the image sequence.

Qualitative results for the *railway* sequence are shown in Figure 3.2. This sequence, where camera also moves due to the wind, contains periodic self-occlusion of a walking person followed by occlusion by a passing car; with trees swaying due to the wind in the background. For the *railway* sequence, our method demonstrates the qualitative improvements as well which can be seen in the fifth row of Figure 3.2. It is important to highlight that the results from FD method were consistently worse than MoG and hence have not been shown here. We, however, have provided the comparison of all six methods in our quantitative analysis.

### *3.2.2 Quantitative Analysis*

Quantitative analysis is performed on both sequences and the results obtained from our method are compared to [12], [53], [64] and [70]. We compute the following two measures for assessing the

Table 3.1: Number of frames used for training

| | Support Vector Machine | Mixture of Gaussian |
|---|---|---|
| fountain | 75 | 400 |
| railway | 75 | 270 |

quality of our results:

$$\texttt{Precision} = \frac{\texttt{\# of true positives detected}}{\texttt{total \# of positives detected}}$$

$$\texttt{Recall} = \frac{\texttt{\# of true positives detected}}{\texttt{total \# of true positives}}$$

The detection accuracy, in terms of both the precision and the recall is considerably higher than FD, MoG, and color distributions based approaches as seen in Figure 3.3 and Figure 3.4. The recall rate for our method is also consistently high for both sequences, whereas in some instances the precision decreases due to strong motion in the input image sequence. This indicates that the localized foreground is larger than the labeled ground truth, however, the background pixels such as the fountain and the swaying trees are not detected as foreground objects at all. Moreover, we are not using any post-processing techniques, such as graph cuts [60] to improve the boundaries of the foreground objects, which would improve the precision considerably.

It is important to highlight that the results were generated using SVM which was trained using a very small number of training images as opposed to MoG. As shown in Table 3.1, we use 75 frames for training of SVM as opposed to 400 frames for *fountain* and 270 frames for *railway* sequence. It underlines the distinct advantage of our technique over MoG, in cases when the amount of available training data is limited.

## 3.3 Discussion

Scene modeling is a very significant initial step for various vision based systems. The existing methods often fail for scenes with dynamic textures or cyclic background motion. We propose treating the scene or the background modeling problem as a Single-Class classification problem, and propose using single-class SVM that is able to create the optimal class boundary from a *very limited* set of training examples. We also employ a novel, yet simple region based features, extracted at each pixel location for training the single-class SVMs. The proposed features not only capture the dynamics at each pixel over time, they also capture the spatial context of the region surrounding a pixel. We have presented results on challenging sequences that contain considerable amount of sensor motion, in addition to dynamic backgrounds.

We compare our results with five standard techniques including Mixture of Gaussian, Principle Component Analysis, Mediod filtering and Adjacent Frame Difference, Kernel Density Estimation and Color Distribution based methods and notice a very significant improvement. Our method has successfully minimized false positives and shows considerably higher recall and precision compared to all five approaches without using any post-processing. In addition, we have the distinct advantage of using considerably *less training data* as opposed to other methods. These encouraging results indicate the practicality and effectiveness of our method.

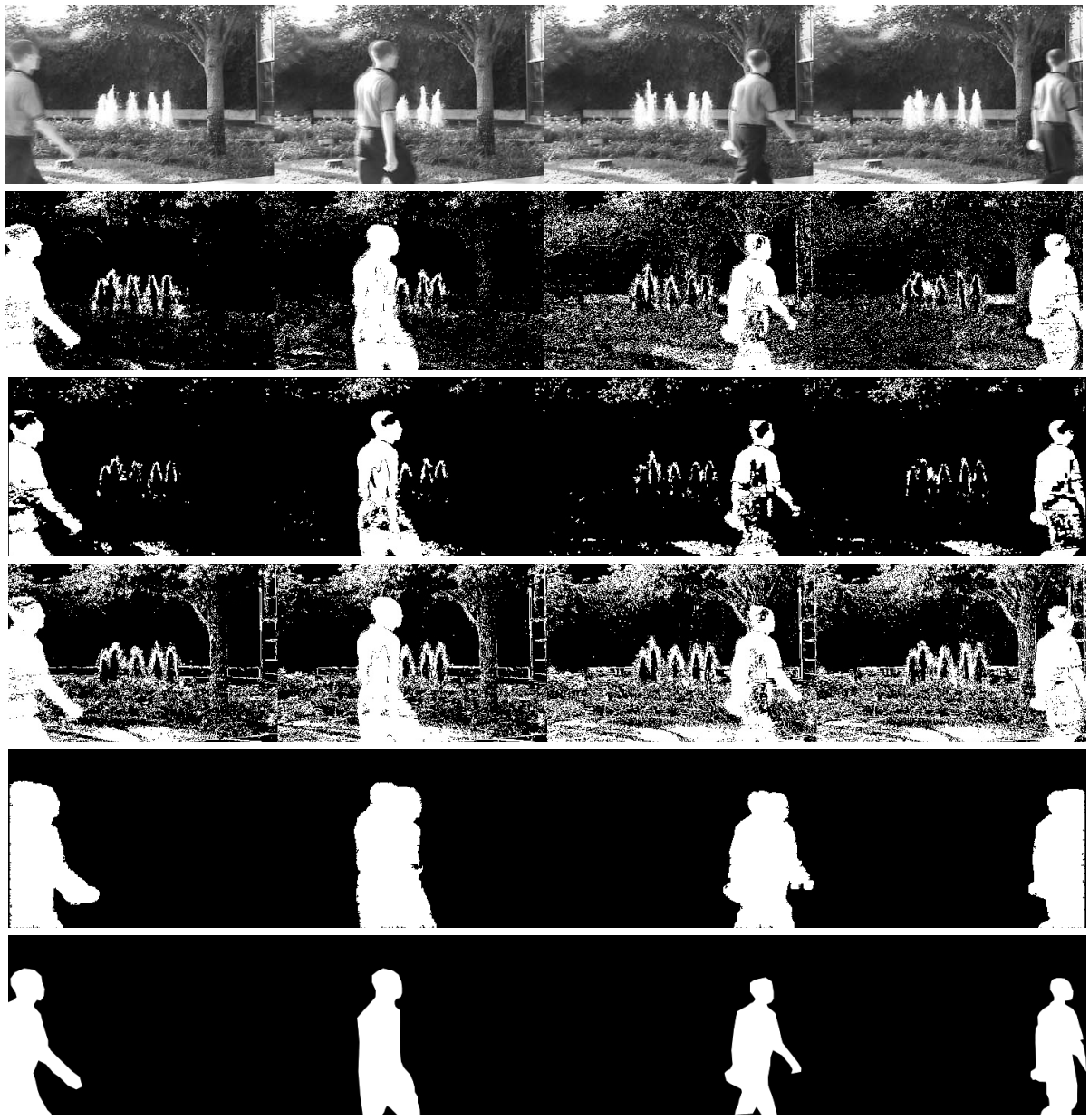


Figure 3.1: Experimental results obtained from the *fountain* sequence: The first row shows the original images which are followed by results obtained from Mixture of Gaussian approach shown in row 2 (training was done using 400 frames). The results from Median based and Principle Component Analysis based approaches are shown in rows 3 and 4 respectively. The results obtained from our method are shown in row 5 (using 75 frames only for training) and the ground truth subtraction results are shown in the last row. No post-processing was performed on these results.

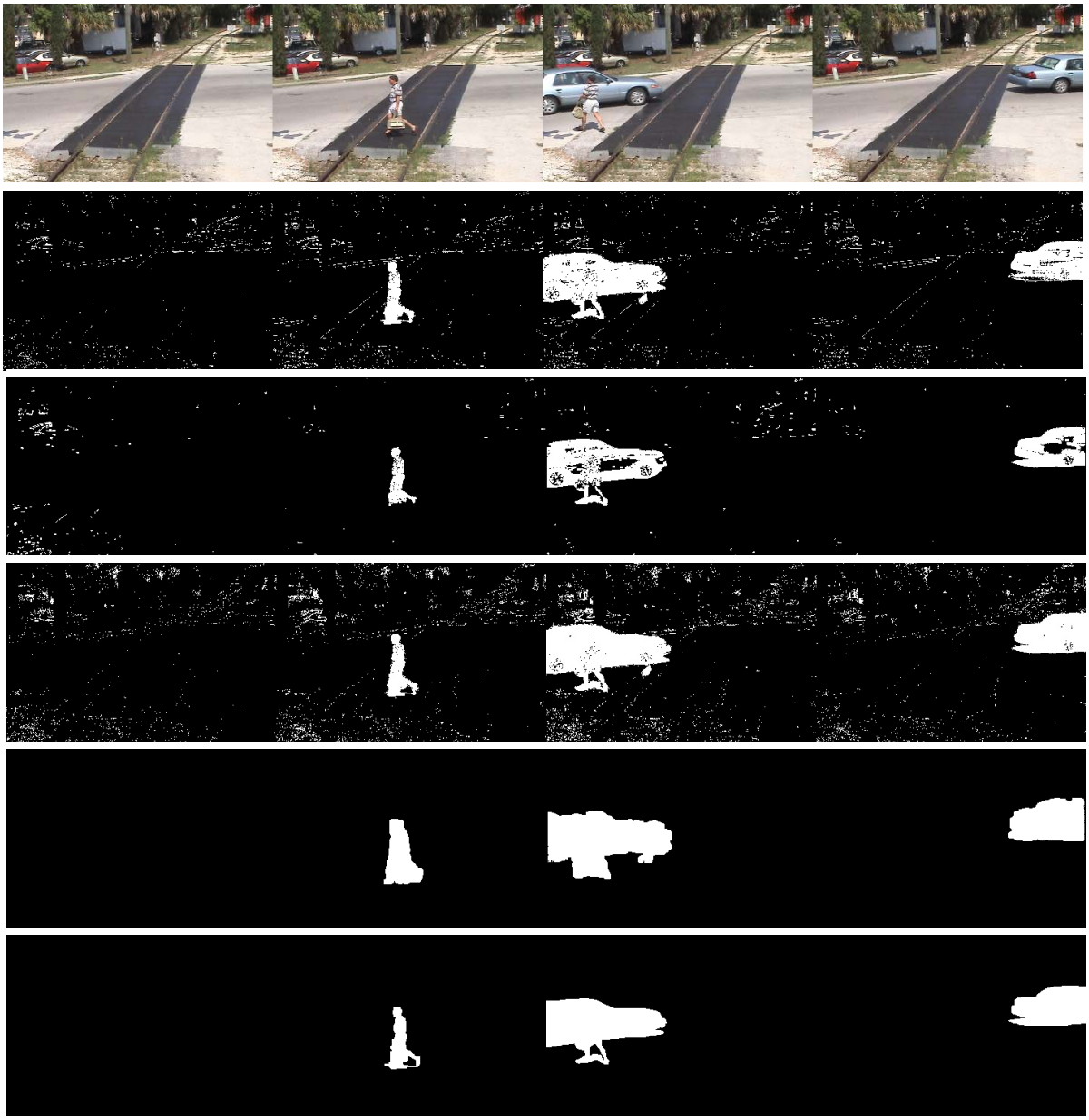


Figure 3.2: Experimental results obtained from the *railway* sequence: The first row shows the original images which are followed by results obtained by Mixture of Gaussian approach shown in row 2 (trained on 270 images). The results from Median filtering and Principle Component Analysis based approaches are shown in rows 3 and 4 respectively. The results obtained from our method are shown in row 3 (using 75 frames only for training) and the ground truth subtraction results are shown in the last row.

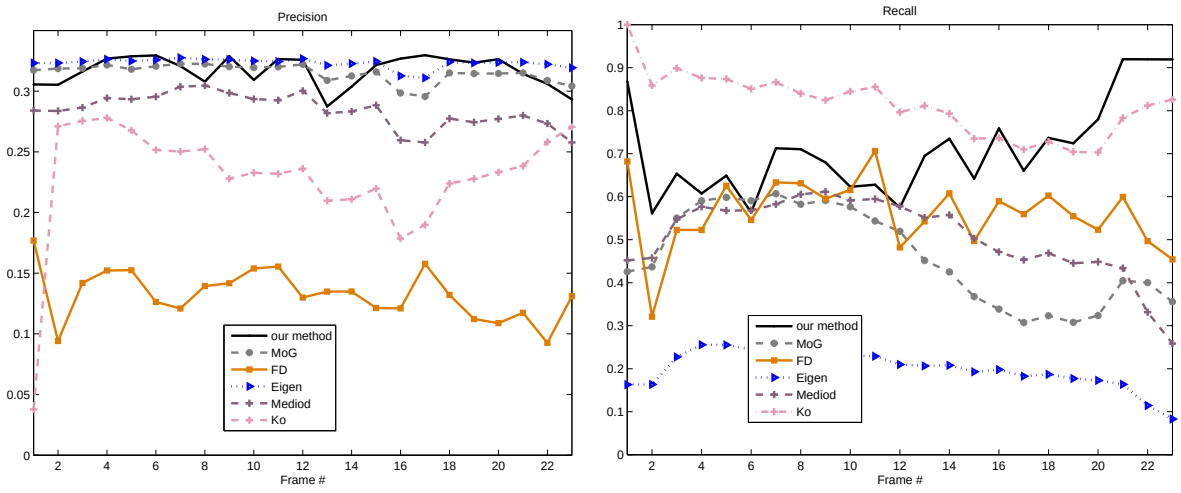



Figure 3.3: Comparison of our method with the state of the art MoG method [64] as well as Eigen [53], Medoid [12, 56], FD [70], and Ko [42] methods using *fountain* sequence: The figure on the left shows the calculated precision for all six methods while the right figure shows the computed recall for these methods.

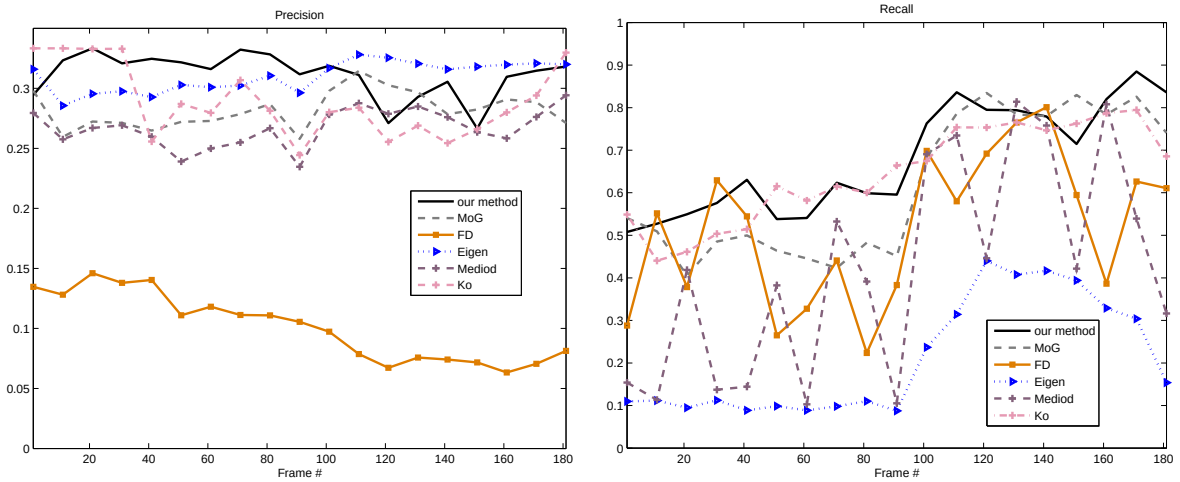



Figure 3.4: Comparison of our method with the state of the art MoG method [64] as well as Eigen [53], Medoid [12, 56], FD [70], and Ko [42] methods using *railway* sequence: The figure on left shows the calculated precision for all six methods while the right figure shows the computed recall for these methods.

# CHAPTER 4: REVIEW OF MULTI-VIEW GEOMETRY

A brief review of the basic concepts related to Multi-View Geometry are presented here. A more comprehensive study can be found in [1, 26, 79, 80]

## 4.1 Projective Geometry

Projective transformations are part of our everyday life. A 3D real world object is 'transformed' into a picture using these transformations. For example, circles may appear as ellipses and squares may not be squares any more. Some properties are preserved (such as straight lines) while others are not (such as angles, distances, ratio of distance). Euclidean geometry is used to describe our world but is insufficient to describe ideal points (or points at infinity). That is why we use projective space which is an extension of Euclidean space. A point $(x, y)$ in Euclidean space is represented as $(x, y, 1)$ in projective space which is same point in homogeneous coordinate as $(kx, ky, k)$ when $k$ is non-zero. Projective space allows us to define points at infinity $(x, y, 0)$ which would map to real-world points $(x/0, y/0)$, commonly referred as points at infinity. In general, Euclidean space $\mathbb{R}^n$ can be extended to a projective Space $\mathbb{P}^n$ by representing points as homogeneous vectors.

In affine transformations (a linear transformation to $\mathbb{R}^n$) are seen in the form of rotation, translation, scaling & shear and these transformations preserve points, straight lines and planes. The result of affine transformation is that parallel lines remain parallel and points at infinity remain at infinity. Mathematically, a linear transformation of Euclidean Space $\mathbb{R}^n$ is represented by matrix multiplication applied to the coordinates of the point. In similar manner, projective transformation of the projective space $\mathbb{P}^n$ is a mapping of the homogeneous coordinates representing a point (an $(n+1)$-vector), in which the coordinate vector is multiplied by a non-singular matrix (called

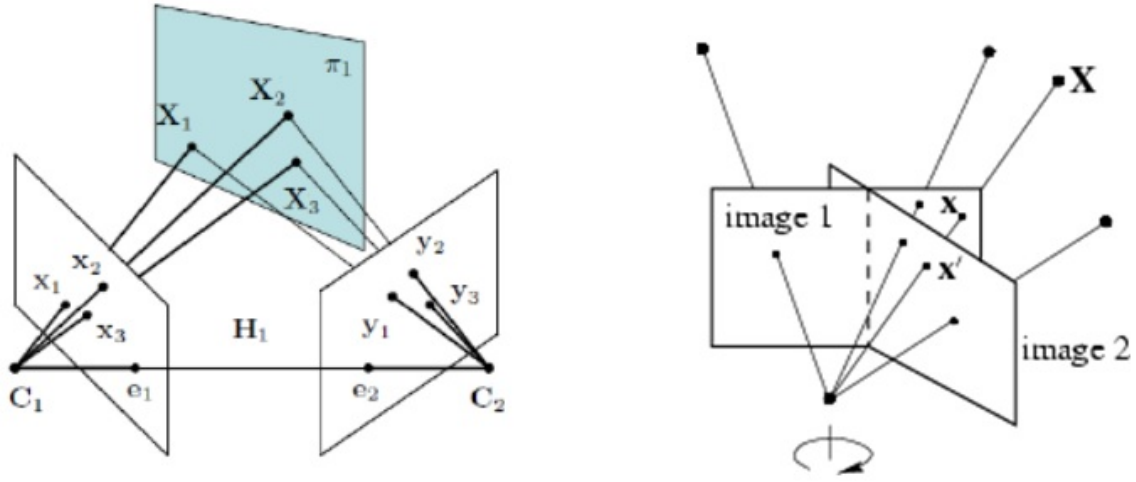



Figure 4.1: Homography: A projective transformation that defines the relationship between points on a planar surface when viewed from two cameras. In the first image on the left, a point $y_1$ in the right camera view is transferred via the homography $H_1$ to a matching point $x_1$ in left camera view. Similarly in the image on the right, a point $X$ in image 1 maps to $X'$ in image 2 via another homography. Here image 2 could be a rotated version of image 1. Notice that these images may be taken from a camera that rotates around its axis of projection and is equivalent to looking at the points that are on a plane at infinity.

Homograpy) and is represented by

$$\mathbf{X}' = \mathbf{H}_{(n+1)\times(n+1)}\mathbf{X} \tag{4.1}$$

Intuitively, homography defines relationships between points on planar surfaces or simply planes as shown in Figure 4.1. This is common when looking at scenes from a camera that is far away. Note that homography is a 3x3 matrix with 8 degrees of freedom (DOF) and can be calculated using 2 constraints (or 4 points).

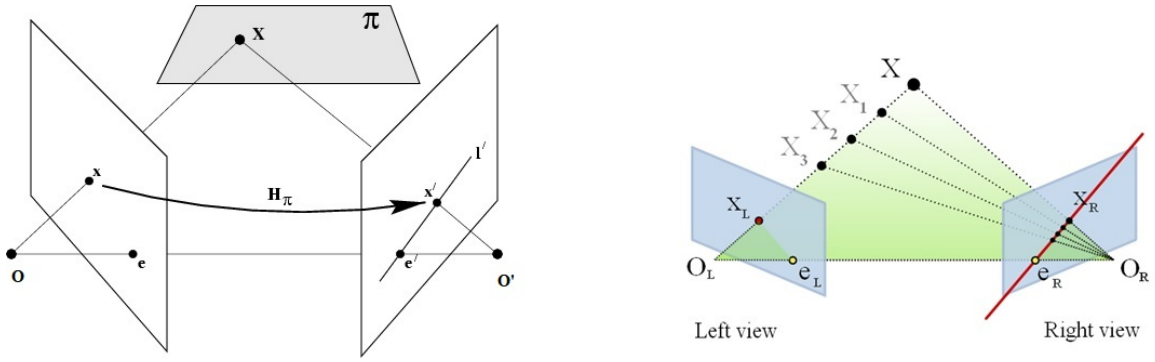



Figure 4.2: Epipolar Geometry: Any world point $X$ or $X_1$ is seen in the left view of the image on the right as $X_L$ and is constraint by epipolar line. It is also related to a matching point $X_R$ in the right view by Fundamental Matrix $F$. This relationship can be written as $X_L^T = FX_R$.

## 4.2 Epipolar Geometry and Fundamental Matrix

In a multi-view geometry, a world point $X$ on a planar surface is imaged at $x$ in the first view and is related to the corresponding point $x'$ imaged in second view. They are related by Homography $H_\pi$ as shown in the left image of Figure4.2. A line joining two camera centers ($O$ and $O'$) is called baseline. Baseline intersects each image plane at points known as epipoles and depicted as ($e$ and $e'$) in the left image and ($e_L$, $e_R$) in the right image. The line joining epipole $e'$ and point $x'$ is called epipolar line $l'$ as shown in the left image. The same line can also be seen in the right image. These three points ($O$, $O'$ and $X$ are co-planar) and can be used to derive relationship for points that are no longer on a planar surface. This relationship is called Fundamental Matrix (or $F$) which is a 3x3 matrix of rank 2. Notice that the world point $X$ may be closer to the left camera and could be at location $X_1$ but will still be imaged at $X_L$ in the left view as seen in the right image. For complete derivation, see [26].

## 4.3 Projective Depth

In a multi-view geometry, a world point $X = (x^T, \tau)^T$ imaged at $x$ in the left view is related to the same point imaged in second view by

$$\mathbf{x}' = Hx + \tau\mathbf{e}' \tag{4.2}$$

This world point introduces a parallax relative to the plane $\pi$ as illustrated in Figure 4.3. Since $x'$, $e'$, and $Hx$ are collinear, the scalar $\tau$ is the parallax relative to the plane $H$. When $\tau$ is 0, it implies the point $X$ is on the plane $\pi$. Otherwise, the sign of $\tau$ indicates which side of the plane $\pi$ the point X lies.

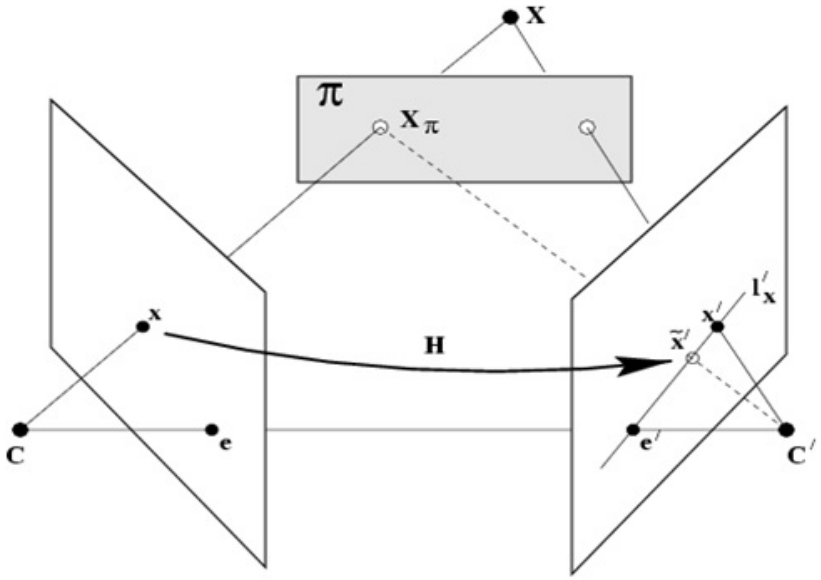



Figure 4.3: Geometric Depth: A world point $X$ which may not be on a planar surface $\pi$ is imaged at $x$ in the left view. Such point $x$ is transferred via a Homography $H$ to a matching point $x'$ in right view. Intuitively $\tau$ is the depth of point $X$ from the plane $\pi$ which currently is behind the plane.

# CHAPTER 5: SELECTIVE SUBTRACTION

An alternate approach to background subtraction and the one most related to the techniques presented in this section, is based on stereo geometry, which attempts to segment the scene using multiple views. There are a number of limitations of classical multi-view based techniques namely the cameras need to be in strict configurations or a reliable dense disparity map is required or a lot of training data is needed. Most of these techniques also lack any flexibility in selecting background or foreground objects. In Section 3.1.2, we presented results where we were able to use considerably *less training data* as opposed to other single-view based techniques. This section presents our works to address other limitations of multi-view based techniques mentioned above. We will start by re-imagining background subtraction in a way that allows us to use projective depth.

## 5.1 Using Human Walk as Reference Plane

Consider a sequence of images $\{\mathbf{I_t}\}^{t=1\ldots n}$, where multiple objects are moving across the scene as shown in Figure 5.1. A simple change detection algorithm can be used to detect the moving objects (or blobs) and their head and feet positions can be obtained by using the approach described in [48]. Let $\mathbf{P}_1$ and $\mathbf{P}_2$ be the two $3 \times 4$ camera projection matrices of two arbitrary cameras observing the scene. Since we do not require any calibration or a specific configuration, without loss of generality, we will model the two cameras as canonic cameras, i.e. $\mathbf{P}_1 = [\mathbf{I}, \mathbf{0}]$ and $\mathbf{P}_2 = [[\mathbf{e}']_\times \mathbf{F}, \mathbf{e}']$, where $\mathbf{F}$ is the fundamental matrix, $\mathbf{e}'$ is the epipole in the second camera view, and for any vector

$\mathbf{v} = (a,b,c)$ the notation $[\mathbf{v}]_\times$ denotes the skew symmetric matrix defined as:

$$[\mathbf{v}]_\times = \begin{pmatrix} 0 & -c & b \\ c & 0 & -a \\ -b & a & 0 \end{pmatrix} \tag{5.1}$$

Next, define the head and feet positions of a person viewed by these two cameras at a given instant in time as $\mathbf{p_1^t}$ (top), $\mathbf{p_1^b}$ (bottom) and $\mathbf{p_2^t}$ (top), $\mathbf{p_2^b}$ (bottom) points, respectively. These corresponding pair of points define a one parameter family of planes given by

$$\pi_\alpha = \alpha \mathbf{P}_1^T [\mathbf{p_1^t}]_\times \mathbf{p_1^b} + \mathbf{P}_2^T [\mathbf{p_2^t}]_\times \mathbf{p_2^b} \tag{5.2}$$

$$= \alpha \begin{pmatrix} c[\mathbf{p_1^t}]_\times \mathbf{p_1^b} \\ 0 \end{pmatrix} + \begin{pmatrix} c\mathbf{F}^T [\mathbf{e}']_\times^T [\mathbf{p_2^t}]_\times \mathbf{p_2^b} \\ \mathbf{e}'^T [\mathbf{p_2^t}]_\times \mathbf{p_2^b} \end{pmatrix} \tag{5.3}$$

where $\alpha$ is a scalar parameter.

The homography induced by this family of planes is then given by

$$\mathbf{H}_\alpha = \left( \mathbf{e}'^T [\mathbf{p_2^t}]_\times \mathbf{p_2^b} \mathbf{I} - \mathbf{e}' \mathbf{p_2^b}^T [\mathbf{p_2^t}]_\times^T \right) [\mathbf{e}']_\times \mathbf{F}$$

$$- \alpha \mathbf{e}' \mathbf{p_2^b}^T [\mathbf{p_2^t}]_\times^T \tag{5.4}$$

$$= [[\mathbf{p_2^t}]_\times \mathbf{p_2^b}]_\times [\mathbf{e}']_\times [\mathbf{e}']_\times \mathbf{F} + \alpha \mathbf{e}' \mathbf{p_2^b}^T [\mathbf{p_2^t}]_\times^T \tag{5.5}$$

$$= [[\mathbf{p_2^t}]_\times \mathbf{p_2^b}]_\times \mathbf{F} + \alpha \mathbf{e}' \mathbf{p_2^b}^T [\mathbf{p_2^t}]_\times^T \tag{5.6}$$

Now, let $\mathbf{m}$ and $\mathbf{m}'$ be two corresponding points of a 3D point $\mathbf{M}$ viewed by the two cameras. The

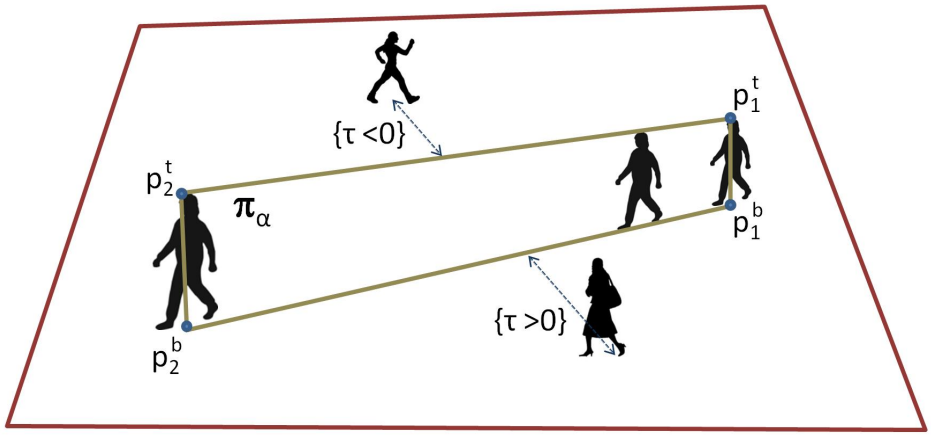



Figure 5.1: Reference plane: The reference plane is defined by a moving object or human *walk* where the head and the feet positions provide necessary constraints to define a plane. The projective depth ($\tau$) is defined as the distance between the *reference plane* and the objects in the scene.

homography $\mathbf{H}_\alpha$ would map $\mathbf{m}$ from the left image to the right image as

$$\mathbf{H}_\alpha \mathbf{m} = [[\mathbf{p_2^t}]_\times \mathbf{p_2^b}]_\times \mathbf{F}\mathbf{m} + \alpha \mathbf{e}' \mathbf{p_2^b}^T [\mathbf{p_2^t}]_\times^T \mathbf{m} \tag{5.7}$$

$$= [[\mathbf{p_2^t}]_\times \mathbf{p_2^b}]_\times [\mathbf{e}']_\times \mathbf{m}' + \beta \mathbf{e}' \tag{5.8}$$

$$= (1-\gamma)\mathbf{m}' + \gamma \mathbf{e}' + \beta \mathbf{e}' \tag{5.9}$$

where $\beta = \frac{\alpha}{\mathbf{p_2^b}^T [\mathbf{p_2^t}]_\times^T \mathbf{m}}$, $\gamma$ is a scalar parameter, and the last equation follows from the fact that the point $[[\mathbf{p_2^t}]_\times \mathbf{p_2^b}]_\times [\mathbf{e}']_\times \mathbf{m}'$ is on the epipolar line $[\mathbf{e}']_\times \mathbf{m}'$ and hence can be written as a linear combination of $\mathbf{e}'$ and $\mathbf{m}'$.

Therefore by proper scaling of the last equation we can get

$$\mathbf{H}_\alpha \mathbf{m} = (1-\tau)\mathbf{m}' + \tau \mathbf{e}' \tag{5.10}$$

Here the scalar parameter $\tau$ may be interpreted as the projective depth of the point $\mathbf{M}$ from the

plane $\pi_\alpha$, because we can readily verify that if $\mathbf{M} \in \pi_\alpha$, then $\tau = 0$. Otherwise, $\tau$ will be either positive or negative depending on which side of the plane, $\mathbf{M}$ lies.

Rearranging (5.10), we can determine $\tau$ from either $x$ or $y$ coordinates of the points $\mathbf{m}$, $\mathbf{m}'$, and $\mathbf{e}'$. For instance using $x$ coordinates we have:

$$\tau = \frac{(\mathbf{H}_\alpha \mathbf{m})_x - (\mathbf{m}')_x}{(\mathbf{e}')_x - (\mathbf{m}')_x} \tag{5.11}$$

where $(\cdot)_x$ denotes the $x$ coordinate of the vector. Note that you can rewrite (5.11) for $y$ coordinate of the vector as well.

One last issue before we describe how we can use (5.11) for selective subtraction: The base homography $\mathbf{H}_\alpha$ as derived above is parameterized in terms of a scalar $\alpha$. There are several ways we can determine $\alpha$. One simple way is to use a pair of corresponding points between the two camera views to solve for $\alpha$ using (5.6). For instance, either the head or feet point correspondences of the person in the two cameras in a later frame can be used to determine $\alpha$. In this way, a walking person would establish a reference plane as depicted in Figure 5.2.

## 5.2 Selective Subtraction Framework

We use the reference plane as the decision boundary between foreground and background objects. *Any plane in the scene can be chosen as the reference plane and thus it gives us the flexibility of selectively keeping or subtracting the objects on either side of the plane*. For instance, if the reference plane chosen is the farthest plane in the scene then all moving objects fall in front of the reference plane and thus the approach can be used as a traditional background subtraction technique. As shown in Figure 5.1, the projective depth ($\tau$) for any moving object in the scene can

be estimated and based on the sign of $\tau$, the object can be classified as being on the foreground or the background effectively producing a binary classification. Moreover, the rate of change of $\tau$ over time may be interpreted as 'projective speed' of the object relative to the reference plane. For instance, when an object moves, the rate of change of $\tau$ can be estimated and can be used in several applications including calibration [31, 34], video summarization and image/video context description [67, 68], vehicle navigation or detecting anomalies in pedestrian paths [37, 39, 65, 66]. Furthermore, the idea of a single reference plane and the estimation of projective depth defines a framework that can be extended to use multiple reference planes thus resulting in multi-class classification. This allows us to classify a scene as layers of foreground or background [7, 8]. It is important to highlight that this technique can also be used even when an object is fully or partially occluded (full occlusion can be detected as the object disappearing from the foreground).

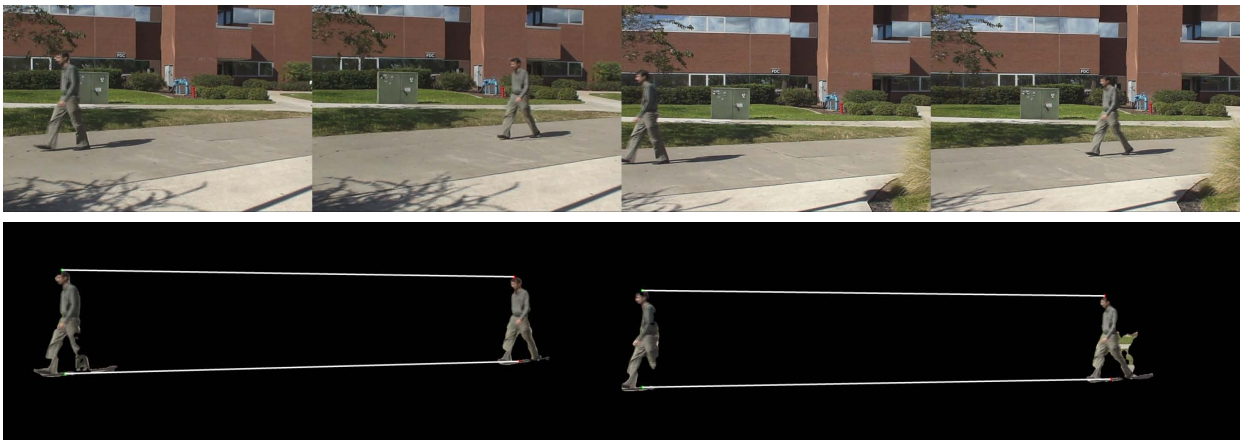


Figure 5.2: Base homography: First row shows the images used to estimate the base homography from reference plane. The pair of images on the left shows the first and the last image of the walk (from first camera view) and the pair of images on the right shows the first and the last images of the walk (from second camera view). The second row shows the head and feet positions of the object used to estimate the base homography. It is clear that the correspondences of head and feet positions from first and the last frames alone are sufficient to estimate the base homography. Please note that any change detection algorithm can be used to detect objects or blobs.

## 5.3 Results and Discussion

The algorithm was tested on a set of challenging sequences with multiple moving objects with significant occlusions and illumination changes. The comparative results with the Mixture of Gaussian method [64] have also been presented. The first sequence contains an outdoor scene with several moving objects along with shadows and dynamic motions including moving tree leaves. A simple frame difference algorithm with threshold along with connected component analysis was used to detect the changes (or blobs) in the scene. The reference *walk* from a moving object was selected as *reference plane* and *base homography* was estimated using head and feet positions in the first and the last frames as shown in Figure 5.2. It should be highlighted that only four point correspondences are used to calculate the *base homography* and we do not require any additional training data. An alternate approach would be to track the head and feet positions throughout the reference *walk* and use curve fitting techniques to improve the precision of head and feet positions. Moreover, numerous complex change detection algorithms can be used to detect the blobs with varying degree of success. The discussion on these algorithms is outside the scope of this dissertation.

We use Scale Invariant Feature Transform (SIFT) [46] to find point correspondences. SIFT is known as state of the art feature matching algorithm and provides reliable matching results. Once the blobs are detected, we use the algorithm as described in Section 5.1 to estimate the projective depth ($\tau$). In our experiments we first performed the blob detection followed by feature matching for point correspondences using SIFT. Notice that these two steps can be reversed, as to finding point correspondences on the entire image followed by eliminating the ones outside the blobs. Figure 5.3 shows two views of the input images used for blob detection and feature matching for point correspondences. For each corresponding point, we calculate $\tau$ using (5.11) and use a majority voting scheme to classify the blob as foreground or background (i.e., as being on one

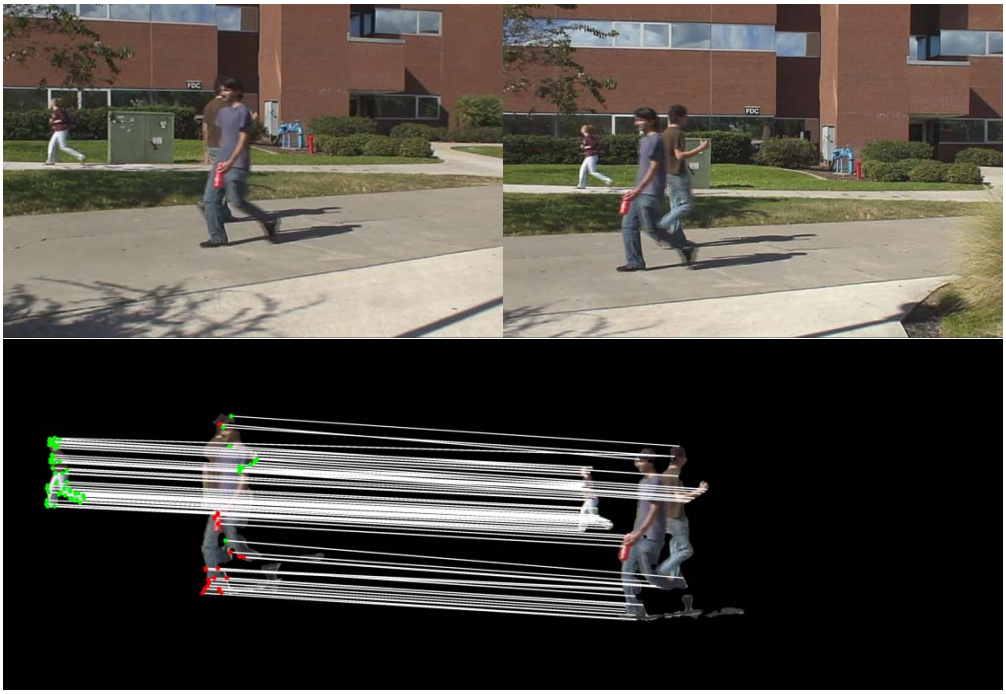


Figure 5.3: Occlusion handling by selective subtraction method: First row shows the input images from two views where two objects are occluding each other. The *reference plane* used in these results lies in the middle of both occluding objects as seen in Figure 5.2 and thus both objects must fall on the opposite sides of the *reference plane*. The correspondences between feature points are shown in the second row. The projective depth of each point was calculated using the proposed technique and the points belonging to front-side are shown in red while the points lying on the other side of the *reference plane* are shown in green. The results show that the proposed technique was correctly able to estimate the projective depth even when the objects are occluded especially near the head and leg positions. For the sake of simplicity we have shown the point correspondences on the first view only.

side of the reference plane or the other). Figure 5.5 depicts results that show that our algorithm can correctly separate the foreground from background. It should also be highlighted that any other background subtraction algorithm can be used as first step for blob detection. In doing so, Selection Subtraction becomes the second step in this process, thus allowing any other background subtraction technique to be used in Selective Subtraction framework.

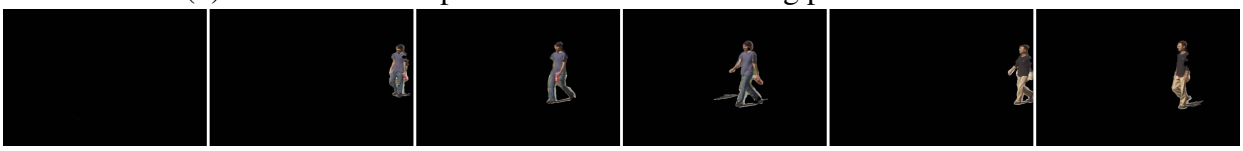
Figure 5.4: Input images: First row shows the selected images as seen from first view and the second row shows the input images from second view. These images (from left to right) show multiple moving objects which (in the order of increasing distance from the far wall) include a girl walking from left to right, followed by a boy walking from left to right holding water bottle, another boy moving from right to left, and finally another boy moving from left to right.

One of the most unique aspects of our technique is the flexibility it provides in selecting the *reference plane* of choice. Figure 5.5 shows how the foreground detection changes when different *reference planes* are selected for selective subtraction. Figure 5.5(a) shows the results when the *reference plane* is the far wall and hence all moving objects are considered foreground as in traditional background subtraction technique. When the *reference plane* is changed to a moving object, the foreground changes accordingly as seen in Figure 5.5(b). Figure 5.5(c) shows the results when the selected *reference plane* is in the middle of pathway thus, detecting the objects in front as foreground. We also selected our *reference plane* as the object walking closest to the camera and found that all moving objects were detected as background. Second test sequence containing the indoor scene with significant illumination changes was also used and the results are shown in Figure 5.6. These results indicate that selective subtraction is effective and provides flexibility in selectively subtracting the objects of choice from the scene.

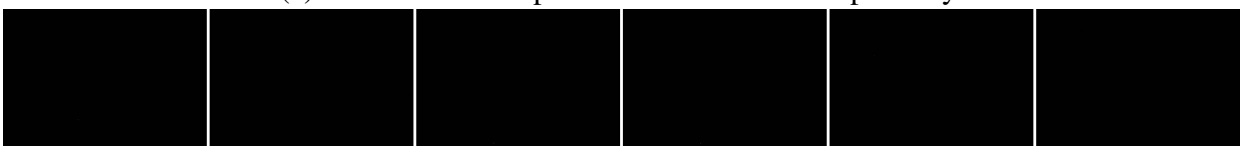
Figure 5.6, 5.7 depict the qualitative analysis of the results showing that our technique performs better than mixture of Gaussian [64]. We also performed the quantitative analysis of the pixel-level detection accuracy. The per frame detection rates are calculated in terms of sensitivity and

(a) When reference plane is the farthest wall in the scene.

(b) When reference plane is the farthest walking person in the scene.



(c) When reference plane is in the middle of pathway.



(d) When the reference plane is the walking person closest to the camera.

(e) Input images without the foreground blobs for (c).

Figure 5.5: Selective subtraction results for outdoor sequence with different *reference planes*: (a) First row shows the blobs found in foreground when the *reference plane* is the farthest wall in the scene. All moving objects are detected as foreground. (b) Second row shows the blobs detected as foreground if the farthest moving object (girl) is used as *reference plane*. All moving objects excluding the girl are now detected as foreground. (c) Third row shows the blobs detected as foreground when the *reference plane* used is in the middle of the pathway. Notice that the girl walking to the left and the boy walking to the right are both on the other side of the *reference plane* and are detected as background. Furthermore, two boys walking to the left are correctly detected as foreground. (d) Fourth row shows the results when the *reference plane* is the moving object closest to the camera and thus none of the moving objects are detected as foreground. (e) Last row shows the input images excluding the foreground blobs detected when the *reference plane* was in the middle of pathway shown in (c).

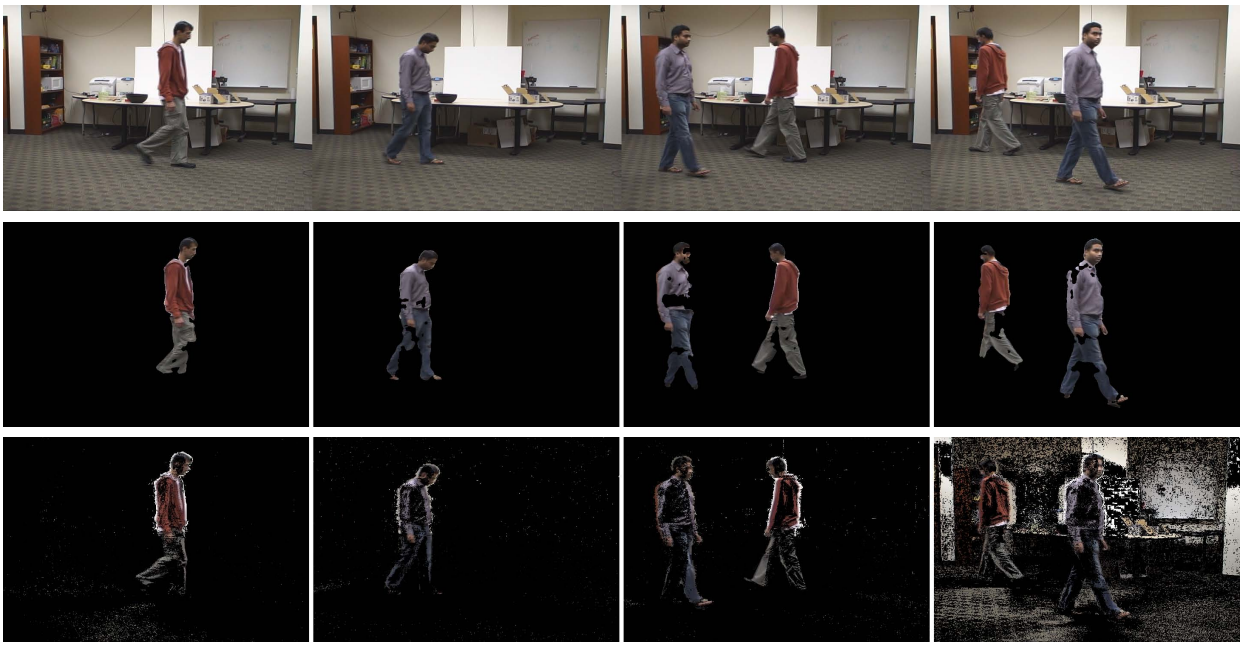


Figure 5.6: Selective subtraction results for indoor sequence: The results of the selective subtraction method are shown here. First row shows the input images from the first view. Objects found in front of the *reference plane* using selective subtraction are shown in the second row and the results of mixture of Gaussian method [64] are shown in the bottom row. The *reference plane* used in these results is the farthest wall in the scene. The results indicate that our technique can effectively detect foreground objects in indoor environments.

specificity, where

$$Sensitivity = \frac{\#\ of\ true\ positives\ detected}{total\ \#\ of\ true\ positives}$$
$$Specificity = \frac{\#\ of\ true\ negatives\ detected}{total\ \#\ of\ true\ negatives}$$

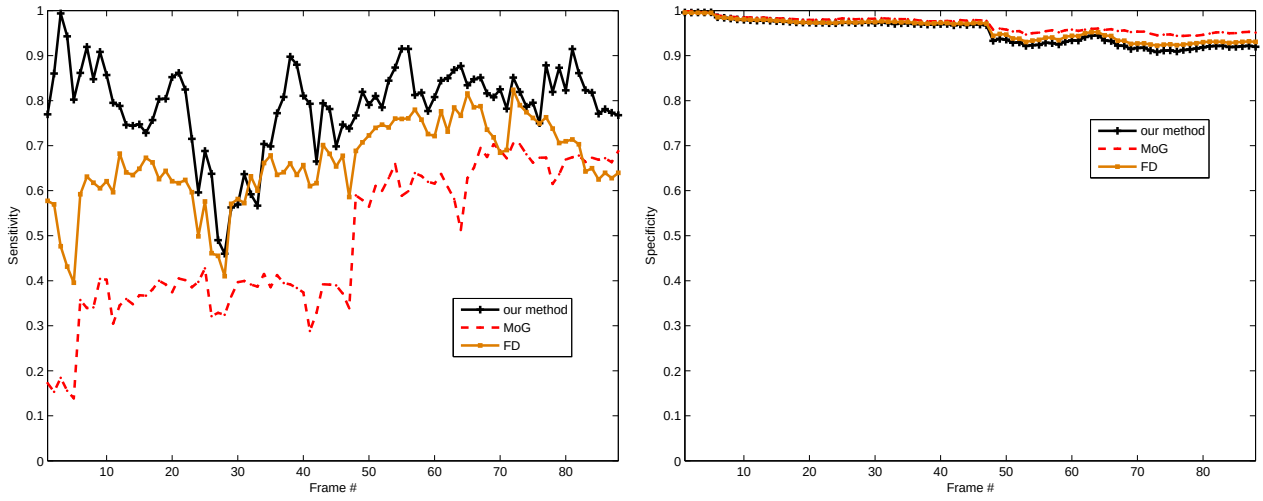
Figure 5.5 shows the sensitivity and specificity of our technique as compared to [64] and [70]. Clearly, the detection accuracy in terms of sensitivity is consistently higher than [64] and [70] while specificity is comparable to both techniques. One of the major advantages of our technique is that it does not require any special camera setup or configuration as needed in other two-view background

Figure 5.7: Selective subtraction as background subtraction: The results of the selective subtraction method when the *reference plane* is the far wall and hence all moving objects are considered foreground as in the traditional background subtraction techniques. First row shows the results obtained from our method and the second row shows the results from state of the art mixture of Gaussian method [64]. The results indicate that our technique can be used as background subtraction and gives better qualitative results.

subtraction techniques. We also do not use the disparity map and thus our algorithm is fast. The average computation time per frame ($480 \times 720$ pixels) is 0.0029 seconds on Intel Core2 Extreme CPU with 4GB RAM (excluding the time needed for blob detection and the feature matching). It should be noted that we have not performed any shadow removal or other post-processing, such as graph cuts [60] to improve the boundaries of foreground objects.

## 5.4 Selective Subtraction as an Extension of Background Subtraction - A New Approach

Traditional background subtraction approaches allow only the binary classification of any scene (i.e., either foreground or background objects) where all moving objects will be chosen as background. Selective subtraction approach provides a new framework which is much more powerful and offers greater flexibility. It offers several unique advantages: (1) Any plane in the scene can be chosen as a reference plane. In traditional background subtraction techniques, all moving objects are considered in the foreground, essentially making the farthest plane or horizon as reference




(a) (b)

Figure 5.8: Quantitative analysis of detection accuracy: (a) shows the sensitivity of our algorithm (Average values: Ours 79%, [64] 49%, [70] 64%), (b) shows the specificity (Average values: Ours 95%, [64] 96%, [70] 95%). The results show that the average detection sensitivity of our technique is consistently better than [64] and [70] and specificity is comparable to these techniques.

plane; (2) In the absence of any plane, the start and end of human walk can be used as a reference plane. Traditional approaches don't allow any such choice; (3) The projective depth of any moving object can be estimated with reference to any chosen plane or walk; (4) A foreground object (which falls on one side of reference plane) can move to the other side of the reference plane and thus may be classified as background object. At a later time, this background object may move back to the original side thus being classified as foreground object again. Traditional approaches do not allow foreground objects to be classified as background and then foreground again. The selective subtraction approach can be seen as an extension of background subtraction as it offers more than what other techniques offer, due to its use of projective depth. Please note that when used with PTZ cameras that are either rotating or zooming, the selection subtraction approach degenerates to standard background subtraction. This is due to the fact that the reference plane reduces to the plane at infinity and the depth information is lost [14, 36, 40].

# CHAPTER 6: SELECTIVE SUBTRACTION FOR HAND-HELD CAMERAS

Selective Subtraction Framework has proved effective in applications that involve fixed cameras, i.e., where camera center doesn't change. We studied the effectiveness of our new framework in more difficult environments such as moving cameras. In the first stage, we focused on hand-held cameras with small motions and evaluated the performance of our approach. Object detection for hand-held cameras was more challenging as the motion may be caused by moving object or camera motion. Even though, camera might be moving due to hand motion, we are assuming that there is no significant forward camera motion. In situations where the forward camera motion is known, Selection subtraction approach should be able to compensate for that. We used same steps as described in 5.3 and did not use any image stabilization techniques.

## 6.1 Reference Plane

Our approach offers tremendous flexibility in selecting the reference plane. Initially, we selected our reference plane as described in Section 5.1. In addition, we also tested our approach with reference planes selected using an area of the scene as shown in Figure 6.2. The results show that Selective Subtraction is effective when used with *any* reference plane in the scene.

## 6.2 Results and Discussion

The algorithm was tested on three a new data-set called **Cellphone** data-set which consisted of three challenging sequences taken from hand-held cameras where each sequence included multiple

moving objects with significant occlusions and illumination changes. The comparative results with the Mixture of Gaussian method [64] have also been presented. Table 6.1 summarizes the data-sets used for testing our method.

**Cellphone** dataset: This dataset comprises of three separate recordings, which we denote at **Cellphone-A**, **Cellphone-B** and **Cellphone-C**. These datasets were captured with two *handheld* SAMSUNG Galaxy S7 and Note 4 cellphones with an image resolution of $1080 \times 1920$. Cellphone-A was captured inside a cafe, where baristas are seen brewing coffee and taking orders for the customers. We see some customers passing in front of the staff from the left and move to the right of the scene. This is a very challenging sequence where a lot of moving objects can be found in small area as shown in Figure 6.2. Each row shows the images captured from both cellphone cameras along with the foreground and background points detected by our algorithm when different reference planes are chosen. The first column of the figures shows the images captured from left cellphone camera and the second column shows images captured from the right cellphone camera. The remaining columns show the results obtained from our method when different *reference planes* are chosen. The third column shows the results when the farthest wall or plane is used as reference plane. The fourth column shows the results when the middle plane is used as reference plane and the fifth column shows the results when foremost area is chosen as reference plan. In most results, objects are correctly classified as foreground objects. The average accuracy scores of 84%, 71.8% and 82.2% were observed for correct classification of each point shown in last three columns.

Table 6.1: Summary of the datasets for hand-held cameras.

| Dataset | Total Number of Frames | Image Resolution |
|---|---|---|
| **Cellphone-A** | 140 | $1080 \times 1920$ |
| **Cellphone-B** | 277 | $1080 \times 1920$ |
| **Cellphone-C** | 224 | $1920 \times 1080$ |

Figure 6.3 shows some of the frames in the **Cellphone-B** data-set. This sequence captures a food court in a shopping mall where both cameras are not at the same depth from the objects. People are seen moving in the background and helping themselves with food. Each row of the figure shows results obtained from our method. The first and second columns show two views of images captured from cellphone cameras. The remaining 4 plots in each row show the results obtained from our method when different *reference planes* are chosen. The top-left plot shows the results when the farthest wall or plane is used as reference plane. The bottom-right plot shows the results when the closest plane (i.e., closest to camera) is used as reference plane. The bottom-left and top-right plots show the results when different middle planes are used as reference planes. In most results, objects are correctly classified as foreground and background objects. The average accuracy scores of 94%, 94%, 85.3%, and 76% were observed for correct classification of each point shown in last two columns. The reference planes used are shown in Figure 6.2. Notice that these reference planes were selected by choosing specific areas of the scene rather than any specific human walk.

Finally, Figure 6.4 shows some of the frames in the **Cellphone-C** data-set. This sequence captures most challenging scene which includes dynamic moving objects (i.e., bushes are moving due to strong wind) as well as shadows. People are seen moving in both directions. The top two rows show some of the input frames from two views of cellphone cameras. Third row show results from our algorithm when the chosen reference plane is in the middle. The girl in green shirt walking from the left is selectively subtracted due to being in the background. Fourth row shows results from our algorithm when the chosen reference plane is the farthest plane (i.e., wall) in the scene hence this approach becomes a traditional background subtraction approach. All moving objects are correctly classified as foreground. The remaining rows show results from other approaches. Please note that we used Selective Subtraction approach as an add-on to [64] in the results shown in Figure 6.4.

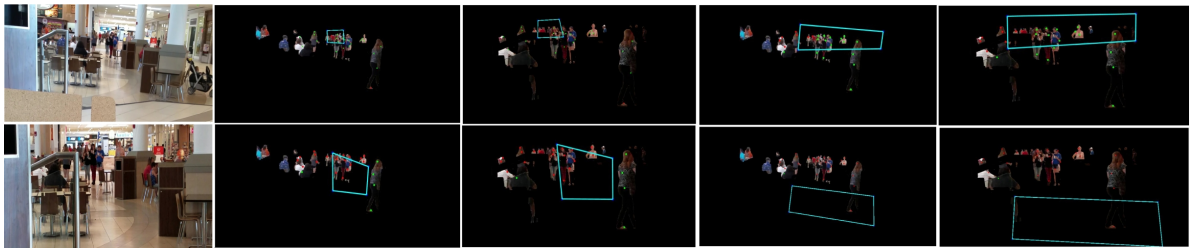


Figure 6.1: Reference planes: The reference plane used in Cellphone-B are shown here. First row shows images from left camera and second row shows the corresponding images from right camera. The first column shows the two frames used for SIFT matches. The remaining columns show selected reference planes as follows: (from left to right) when plane is farthest from camera, when plane is in the middle - one farther from camera and one closer, and when plane is closest to the camera.

These results presented here indicate that selective subtraction approach is effective and provides flexibility in selectively subtracting the objects of choice from the scene. The results are qualitatively demonstrated and compared to other methods, as shown here. The qualitative analysis of these results clearly shows that our technique performs very well in challenging environments even when used with data-sets captured with hand-held cameras.

Table 6.2: **Quantitative Analysis** This table shows results obtained from our method and also comparisons to other methods, tested on the same data.

| | **Ours Ours** | | **[64] [64]** | | **[70] [70]** | |
|---|---|---|---|---|---|---|
| | Sensitivity | Specificity | Sensitivity | Specificity | Sensitivity | Specificity |
| **outdoor** | 79 | 95 | 49 | 96 | 64 | 95 |
| **cellphone**-C | 74 | 99.8 | 14 | 99.3 | 73 | 98.5 |

## 6.3 Quantitative and Qualitative Analysis

Table 6.2 shows the results obtained from our method and we also compare these results with the standard methods of [64] and [70]. The first column shows different datasets that we have tested and the second column shows the specificity and sensitivity measurements obtained from selective subtraction approach, whereas the third and the fourth columns show the results obtained from [64] and [70], respectively. As can be seen from the table, results obtained from our approach are much higher and better. For the **outdoor** data-set, we obtained 79% and 95% for specificity and sensitivity, respectively. Similarly, for the **Cellphone-C** data-set, we obtain 74% and 99.8%, where the best results obtained from the competition is that of 73% from [70] and 98.5% from [64] for specificity and sensitivity, respectively. These results show that our method is robust and applicable. Moreover, our method performs simple blob detection and then computes projective depth, both operations are fast and thus resulting in our approach as computationally efficient. The above encouraging results demonstrate the practicality and viability of our method.

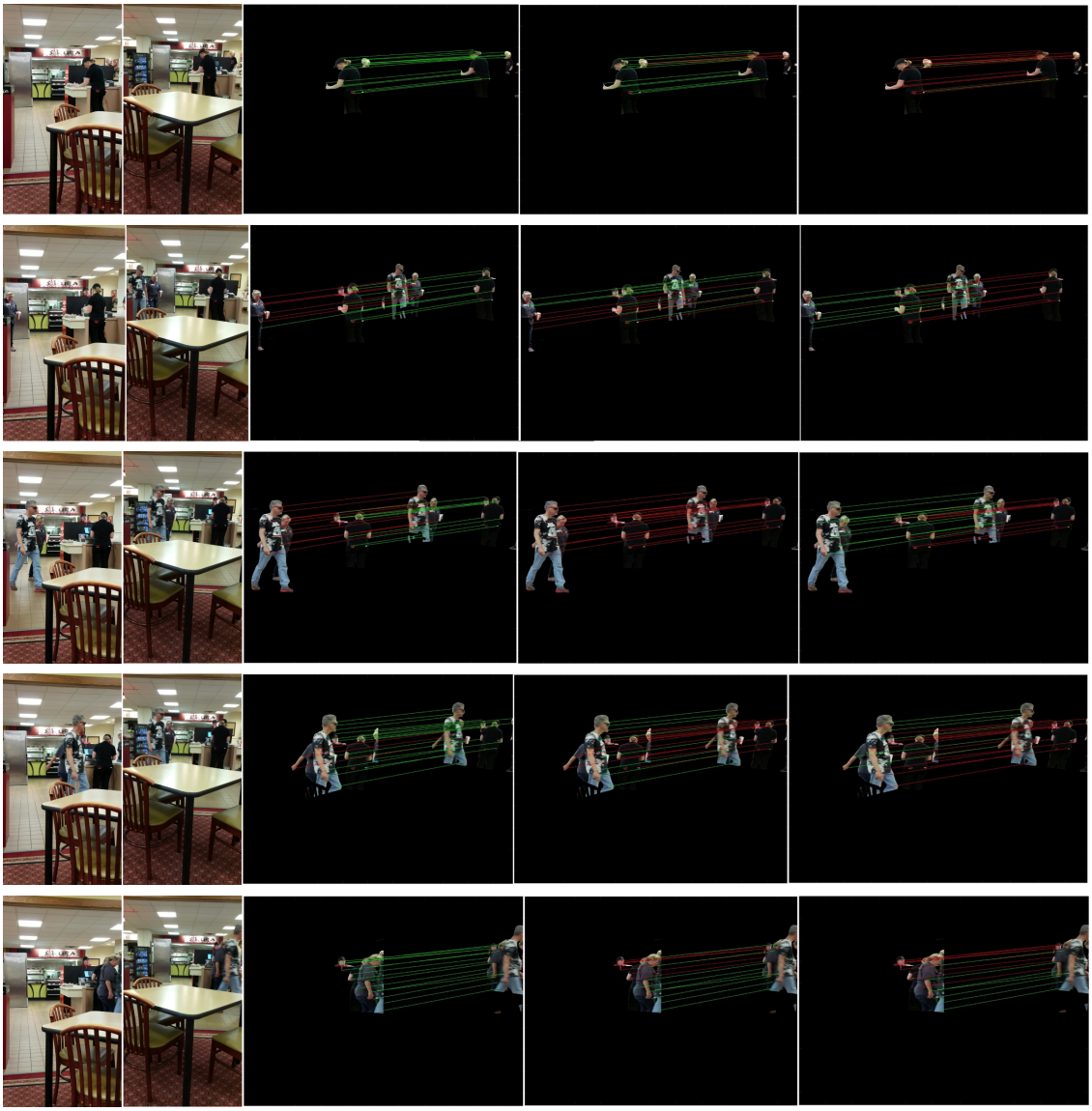


Figure 6.2: Selective subtraction results for **Cellphone-A** sequence: Baristas are seen brewing the coffee and taking orders for the customers. We see some customers walking and pass in front of the staff from the left and move to the right of the scene. Each row shows the images captured from both cellphone cameras along with the foreground and background points detected by our algorithm when different reference planes are chosen. The first column of the figures shows the input images captured from one cellphone camera and the second column shows input images captured from the second camera. The remaining columns show the results obtained from our method when different *reference planes* are chosen. The third column shows the results when the farthest wall or plane is used as reference plane. The fourth column shows the results when the middle plane is used as reference plane and the fifth column shows the results when foremost area is chosen as reference plan.

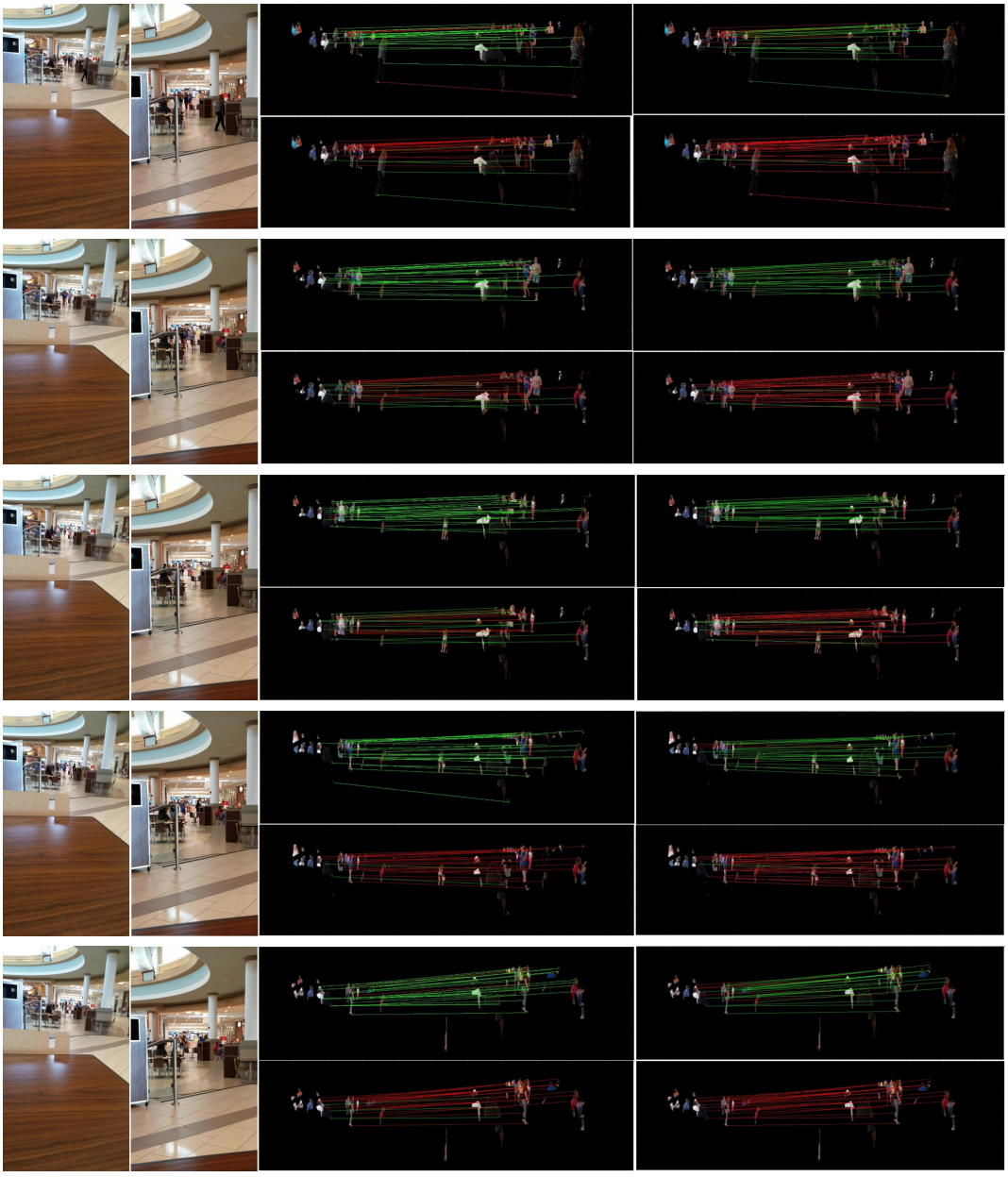
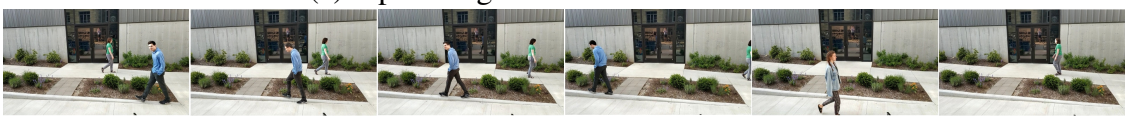


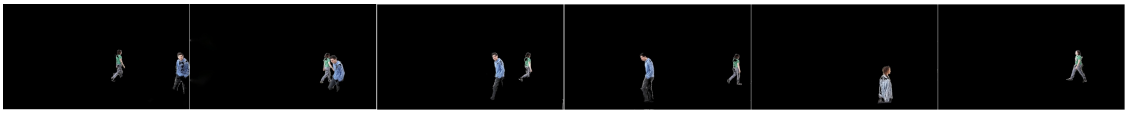
Figure 6.3: Selective subtraction results for **Cellphone-B** sequence: This data-set captures a food court in a shopping mall. People are seen moving in the background and helping themselves with food. The first and second columns show two views of input images captured from cellphone cameras. The remaining 4 plots in each row show the results obtained from our method when different *reference planes* are chosen. The top-left plot shows the results when the farthest wall or plane is used as reference plane. The bottom-right plot shows the results when the closest plane (i.e., closest to camera) is used as reference plane. The bottom-left and top-right plots show the results when different middle planes are used as reference planes.

(a) Input Image: as seen from first view.



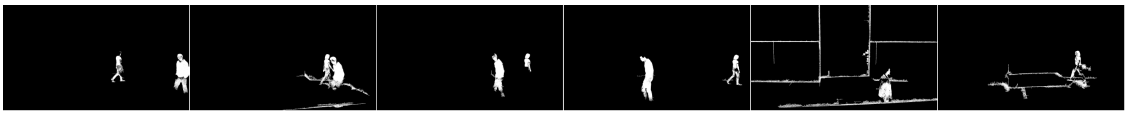
(b) Input Image: as seen from second view.

(c) When the reference plane is the walking person between farthest wall and the camera.

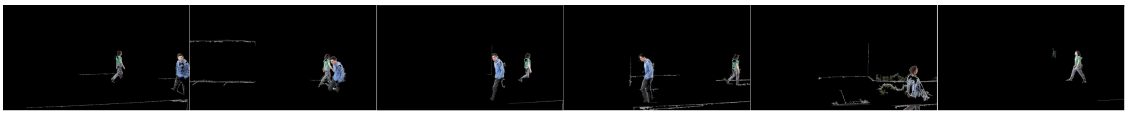


(d) When the reference plane is the farthest wall from the camera.

(e) Results obtained from [70].



(f) Results obtained from [49].



(g) Results obtained from [64].

(h) Results obtained from [57].

Figure 6.4: Selective subtraction results for **Cellphone-C** outdoor sequence with different *reference planes*: The top two rows (a) and (b) show some of the input frames from two views of cellphone cameras. Third row (c) shows results from our algorithm when the chosen reference plane is in the middle. The Fourth row (d) shows results from our algorithm when the chosen reference planes is the farthest plane in the scene. In this case, our approach is similar to standard background subtraction. The remaining rows show results from other traditional approaches.

# CHAPTER 7: CONCLUSION AND FUTURE WORK

This work presents a number of fundamental innovations in the context of background subtraction. In our quest to reduce the requirements of large amount of training data, we initially proposed utilizing temporal and spatial features around each pixel to model the background in dynamic scenes within Single-class classification. We also presented a novel concept of background as objects *other than* foreground which may include moving objects from the scene that cannot be learned from a training data. Our method, "Selective Subtraction", is as alternative to standard background subtraction, and we show that a *reference plane* in a scene is sufficient as the decision boundary between foreground and background. Furthermore, the flexibility in selecting the *reference plane* using the actual moving object in the scene or an arbitrary plane in the scene, is truly unique to this method and is not available in existing background subtraction techniques. We also show that our technique enables us to select multiple reference planes and thus relaxes the strict binary classification-based paradigm as shown in Figure 7.1. This flexibility enables us to use the proposed framework as an extension of standard background subtraction. We present promising results on a challenging set of image sequences to show that the selective subtraction approach performs effectively and has applications in background subtraction and can further can useful in vehicle navigation, path anomaly detection, and detecting objects in crowds. We also present results on images sequences from hand-held cameras to show that this technique is relatively immune to camera motion and is robust. Furthermore, we provide recommendations to improve the results of selective subtraction approach.

To the best of our knowledge, no other background subtraction approach allows an object to be classified as background and then later as foreground (or vice versa). We feel that Selective Subtraction approach has the potential to be a pioneer in this context. We should be able to classify a foreground object as foreground initially and then later as background and then as foreground

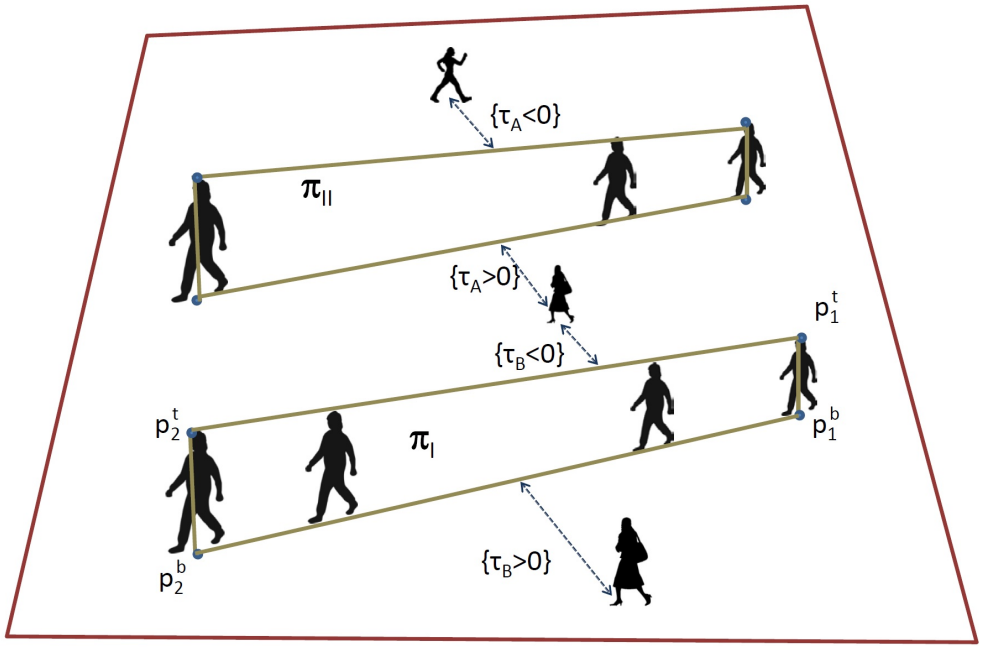



Figure 7.1: Multiple Reference planes: Two reference planes are defined by two moving objects or human *walks* and the projective depths ($\tau_A$) and ($\tau_B$) are defined as the distance between each *reference plane* and the objects in the scene.

object again at much later time. Furthermore, our approach is not dependent on any change detection algorithm and can in fact use any other classical background detection technique as the initial step. The ability to use our Selective Subtraction framework as an add-on feature to any other background subtraction technique provides flexibility that traditional techniques have been lacking to date. All background subtraction techniques should now be able to selectively classify objects as foreground or background whenever needed (or selected).

## 7.1 Future Work

The work presented in this dissertation uses a reference plane which is often facing towards the camera. In future we plan to study different orientations of the reference plane especially when it is orthogonal to the camera direction. Working with image sequences from PTZ cameras may be another possibility. We also plan to study the effects of forward camera motion (where camera center is moving towards the object) on selective subtraction. Another possible avenue of research is performing measurements which can be beneficial in automatic emergency braking systems for vehicles.

The ideas presented in this thesis may further contribute to research in action recognition where one could use this method at a finer scale by using a reference plane within the object of interest, e.g. a human body, to separate motions in different parts, i.e., head, shoulder and arms versus legs which may also be used for fine-grain or piecewise recognition of actions [2, 15, 61, 79]. Another possible application could be in shadow removals where we can use a walking person for geo-localization, assuming that human is walking on a planar surface and hence the moving shadow can be readily separated using the proposed method [13, 22, 32, 33, 38, 76]. Finally deep learning approaches could also benefit from the idea of depth-dependent subtraction [4,72,73] and we invite researchers to explore the possibility of designing novel architectures that exploit these ideas for better scene modeling and understanding.

# LIST OF REFERENCES


[1] N. Ashraf and H. Foroosh. Motion retrieval using consistency of epipolar geometry. In *2015 IEEE International Conference on Image Processing (ICIP)*, pages 4219–4223, 2015.

[2] N. Ashraf, C. Sun, and H. Foroosh. View invariant action recognition using projective depth. *Computer Vision and Image Understanding*, 123, 06 2014.

[3] M. Babaee, D. T. Dinh, and G. Rigoll. A deep convolutional neural network for background subtraction. *CoRR*, abs/1702.01731, 2017.

[4] Baoyuan Liu, Min Wang, H. Foroosh, M. Tappen, and M. Penksy. Sparse convolutional neural networks. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 806–814, 2015.

[5] Y. Benezeth, P. Jodoin, B. Emile, H. Laurent, and C. Rosenberger. Review and evaluation of commonly-implemented background subtraction algorithms. In *Proc. ICPR*, pages 1–4, 2008.

[6] Y. Benezeth, P.-M. Jodoin, B. Emile, H. Laurent, and C. Rosenberger. Comparative study of background subtraction algorithms. *J. Electronic Imaging*, 19:033003, 2010.

[7] A. A. Bhutta, I. N. Junejo, and H. Foroosh. Selective subtraction when the scene cannot be learned. In *2011 18th IEEE International Conference on Image Processing*, pages 3273–3276, 2011.

[8] A. A. Bhutta, I. N. Junejo, and H. Foroosh. Selective subtraction for handheld cameras. *IEEE Access*, 8:36556–36568, 2020.

[9] T. Bouwmans. Traditional and recent approaches in background modeling for foreground detection: An overview. *Computer Science Review*, 11-12:31–66, 2014.

[10] T. Bouwmans, S. Javed, M. Sultana, and S. K. Jung. Deep neural network concepts for background subtraction:a systematic review and comparative evaluation. *Neural Networks*, 117:8 – 66, 2019.

[11] M. Braham, S. Pierard, and M. V. Droogenbroeck. Semantic background subtraction. In *2017 IEEE International Conference on Image Processing (ICIP)*, pages 4552–4556, 2017.

[12] S. Calderara, R. Melli, A. Prati, and R. Cucchiara. Reliable background suppression for complex scenes. In *Proceedings of the 4th ACM International Workshop on Video Surveillance and Sensor Networks*, VSSN '06, page 211–214, 2006.

[13] X. Cao and H. Foroosh. Camera calibration and light source orientation from solar shadows. *Computer Vision and Image Understanding*, 105:60–72, 01 2007.

[14] X. Cao, J. Xiao, H. Foroosh, and M. Shah. Self-calibration from turn-table sequences in presence of zoom and focus. *Computer Vision and Image Understanding*, 102:227–237, 06 2006.

[15] S. Chen, L. Liang, W. Liang, and H. Foroosh. 3d pose tracking with multitemplate warping and sift correspondences. *IEEE Transactions on Circuits and Systems for Video Technology*, 26(11):2043–2055, 2016.

[16] Y. Chen, J. Wang, B. Zhu, M. Tang, and H. Lu. Pixel-wise deep sequence learning for moving object detection. *IEEE Transactions on Circuits and Systems for Video Technology*, pages 1–1, 2018.

[17] P. Christiansen, L. N. Nielsen, K. A. Steen, R. N. Jorgensen, and H. Karstoft. Deepanomaly: Combining background subtraction and deep learning for detecting obstacles and anomalies in an agricultural field. In *Sensors*, 2016.

[18] L. P. Cinelli, L. A. Thomaz, A. F. da Silva, E. A. B. da Silva, and S. L. Netto. Foreground segmentation for anomaly detection in surveillance videos using deep residual networks. 2017.

[19] F. De la Torre and M. J. Black. A framework for robust subspace learning. *International Journal of Computer Vision*, 54(1):117–142, Aug 2003.

[20] R. Díaz, S. Hallman, and C. C. Fowlkes. Detecting dynamic objects with multi-view background subtraction. *2013 IEEE International Conference on Computer Vision*, pages 273–280, 2013.

[21] A. Elgammal, R. Duraiswami, D. Harwood, L. Davis, R. Duraiswami, and D. Harwood. Background and foreground modeling using nonparametric kernel density for visual surveillance. In *Proceedings of the IEEE*, pages 1151–1163, 2002.

[22] Fei Lu, Xiaochun Cao, Yuping Shen, and H. Foroosh. Camera calibration from two shadow trajectories. In *18th International Conference on Pattern Recognition (ICPR'06)*, volume 2, pages 1–4, 2006.

[23] N. Friedman and S. Russell. Image segmentation in video sequences: A probalistic approach. *Uncertainty in Artificial Intelligence*, 1997.

[24] W. Grimson and C. Stauffer. Adaptive background mixture models for real time tracking. In *Proc. CVPR*, 1999.

[25] W. Guan and P. Monger. Real-time detection of out-of-plane objects in stereo vision. In *Advances in Visual Computing*, volume 4291, pages 102–111, 11 2006.

[26] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, New York, NY, USA, 2 edition, 2003.

[27] Y. Ivanov, A. Bobick, and J. liu. Fast lighting independent background subtraction. In *Proc. ICCV, Workshop on Video Surveillence*, 1998.

[28] O. Javed and S. M. Tracking and object classification for automated surveillance. In *Proc. ECCV*, 2002.

[29] P.-M. Jodoin, S. Piérard, Y. Wang, and M. Droogenbroeck. *Overview and Benchmarking of Motion Detection Methods*. 06 2014.

[30] I. Junejo, A. A. Bhutta, and H. Foroosh. "scene modeling for object detection using single-class svm". In *2010 17th IEEE International Conference on Image Processing*, pages 1541–1544, 2010.

[31] I. Junejo and H. Foroosh. Robust auto-calibration from pedestrians. In *2006 IEEE International Conference on Video and Signal Based Surveillance*, pages 92–92, 2006.

[32] I. Junejo and H. Foroosh. Using solar shadow trajectories for camera calibration. In *Proceedings International Conference on Image Processing (ICIP)*, pages 189 – 192, 11 2008.

[33] I. Junejo and H. Foroosh. Gps coordinates estimation and camera calibration from solar shadows. *Computer Vision and Image Understanding*, 114:991–1003, 09 2010.

[34] I. N. Junejo, N. Ashraf, Y. Shen, and H. Foroosh. Robust auto-calibration using fundamental matrices induced by pedestrians. In *2007 IEEE International Conference on Image Processing*, volume 3, pages III – 201–III – 204, 2007.

[35] I. N. Junejo, A. A. Bhutta, and H. Foroosh. Single-class svm for dynamic scene modeling. *Signal, Image and Video Processing*, 7(1):45–52, Jan 2013.

[36] I. N. Junejo, X. Cao, and H. Foroosh. Autoconfiguration of a dynamic nonoverlapping camera network. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 37(4):803–816, 2007.

[37] I. N. Junejo and H. Foroosh. Trajectory rectification and path modeling for video surveillance. In *2007 IEEE 11th International Conference on Computer Vision*, pages 1–7, 2007.

[38] I. N. Junejo and H. Foroosh. Estimating geo-temporal location of stationary cameras using shadow trajectories. In D. Forsyth, P. Torr, and A. Zisserman, editors, *Computer Vision – ECCV 2008*, pages 318–331, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg.

[39] I. N. Junejo and H. Foroosh. Euclidean path modeling for video surveillance. *Image Vision Comput.*, 26(4):512–528, Apr. 2008.

[40] I. N. Junejo and H. Foroosh. Optimizing ptz camera calibration from two images. *Mach. Vision Appl.*, 23(2):375–389, Mar. 2012.

[41] K. Karmann and A. v. Brandt. Moving object recognition using and adaptive background memory. *Time-Varying Image Processing and Moving Object Recognition, Elsevier Science Publishers*, 1990.

[42] T. Ko, S. Soatto, and D. Estrin. Background subtraction on distributions. In *Proc. ECCV*, pages 276–289, 2008.

[43] L. A. Lim and H. Y. Keles. Foreground segmentation using convolutional neural networks for multiscale feature encoding. *Pattern Recognition Letters*, 112:256–262, 2018.

[44] S. Lim, A. Mittal, D. L., and N. Paragios. Fast illumination-invariant background subtraction using two views: Error. analysis, sensor placement and applications. In *Proc. CVPR*, 2005.

[45] T. Liu and G. Wang. A hierarchical approach for robust background subtraction based on two views. In *2009 WRI Global Congress on Intelligent Systems*, volume 4, pages 325–329, May 2009.

[46] D. Lowe. Object recognition from local scale-invariant features. In *Proc. ICCV*, 1999.

[47] B. D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proceedings of the 7th International Joint Conference on Artificial Intelligence - Volume 2*, IJCAI'81, page 674–679, 1981.

[48] F. Lv, T. Zhao, and R. Nevatia. Self-calibration of a camera from video of a walking human. In *Proc. ICIP*, 2002.

[49] J. Malik, S. Belongie, T. Leung, and J. Shi. Contour and texture analysis for image segmentation. *International Journal of Computer Vision*, 43(1):7–27, Jun 2001.

[50] A. Mittel and P. N. Motion-based background subtraction using adaptive kernel density estimation. In *Proc. CVPR*, 2004.

[51] A. Monnet, A. Mittal, N. Paragios, and V. Ramesh. Background modeling and subtraction of dynamic scenes. In *Proc. ICCV*, pages 1305–1312, 2003.

[52] N. Ogale. A survey of techniques for human detection from video. 08 2008.

[53] N. M. Oliver, B. Rosario, and A. P. Pentland. A bayesian computer vision system for modeling human interactions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):831–843, Aug 2000.

[54] M. Piccardi. Background subtraction techniques: a review. In *2004 IEEE International Conference on Systems, Man and Cybernetics (IEEE Cat. No.04CH37583)*, volume 4, pages 3099–3104 vol.4, Oct 2004.

[55] R. Pless, J. Larson, S. S., and B. Westover. Evaluation of local models of dynamic backgrounds. In *Proc. CVPR*, 2003.

[56] A. Prati, I. Mikic, M. Trivedi, and R. Cucchiara. “detecting moving shadows: Algorithms and evaluation". *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(7):918–923, 2003.

[57] J. Redmon, S. K. Divvala, R. B. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. *CoRR*, abs/1506.02640, 2015.

[58] Y. Ren, C. Chua, and Y. Ho. Motion detection with nonstationary background. *Machine Vision and Applications*, 13(5-6):332–343, March 2003.

[59] D. Sakkos, H. Liu, J. Han, and L. Shao. End-to-end video background subtraction with 3d convolutional neural networks. *Multimedia Tools and Applications*, 77:23023–23041, 2017.

[60] Y. Sheikh and M. Shah. Bayesian modeling of dynamic scenes for object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(11):1778–1792, November 2005.

[61] Y. Shen and H. Foroosh. Methods for recognizing pose and action of articulated objects with collection of planes in motion. In *Patent No.: US 8,755,569 B2*, July 2014.

[62] A. Sobral and A. Vacavant. A comprehensive review of background subtraction algorithms evaluated with synthetic and real videos. *Computer Vision and Image Understanding*, 122:4–21, 05 2014.

[63] P.-L. St-Charles, G.-A. Bilodeau, and R. Bergevin. Subsense: A universal change detection method with local adaptive sensitivity. *IEEE Transactions on Image Processing*, 24:359–373, 2015.

[64] C. Stauffer and W. E. L. Grimson. Learning patterns of activity using real-time tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22:747–757, 2000.

[65] C. Sun, I. Junejo, M. Tappen, and H. Foroosh. Exploring sparseness and self-similarity for action recognition. *IEEE transactions on image processing : a publication of the IEEE Signal Processing Society*, 24, 04 2015.

[66] C. Sun, M. Tappen, and H. Foroosh. Feature-independent action spotting without human localization, segmentation, or frame-wise tracking. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2689–2696, 2014.

[67] A. Tariq and H. Foroosh. Feature-independent context estimation for automatic image annotation. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1958–1965, 2015.

[68] A. Tariq and H. Foroosh. A context-driven extractive framework for generating realistic image descriptions. *IEEE Transactions on Image Processing*, 26(2):619–632, 2017.

[69] Y. Tian, M. Lu, and H. A. Robust and efficient foreground analysis for real-time video surveillance. In *Proc. CVPR*, 2005.

[70] K. Toyama, J. Krumm, B. Brumitt, and B. Meyers. Wallflower: Principles and practices of background maintenance. In *Proc. ICCV*, 1999.

[71] O. Tuzel, F. Porikli, and P. Meer. A bayesian approach to background modeling. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Workshops*, pages 58–58, Sep. 2005.

[72] M. Wang, B. Liu, and H. Foroosh. Factorized convolutional neural networks. In *2017 IEEE International Conference on Computer Vision Workshops (ICCVW)*, pages 545–553, 2017.

[73] M. Wang, B. Liu, and H. Foroosh. Look-up table unit activation function for deep convolutional neural networks. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1225–1233, 2018.

[74] R. Wang, F. Bunyak, G. Seetharaman, and K. Palaniappan. Static and moving object detection using flux tensor with split gaussian models. *2014 IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 420–424, 2014.

[75] C. Wren, A. Azarbayejani, T. Darell, and A. Pentland. Pfinder: Real-time tracking of human body. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19:780–785, 1997.

[76] L. Wu and X. Cao. Geo-location estimation from two shadow trajectories. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 585–590, 2010.

[77] L. Yang, J. Li, Y. Luo, Y. Zhao, H. Cheng, and J. Li. Deep background modeling using fully convolutional network. *IEEE Transactions on Intelligent Transportation Systems*, 19:254–262, 2018.

[78] H. Yu. Single-class classification with mapping convergence. *Machine Learning*, 61(1-3):49–69, 2005.

[79] Yuping Shen, N. Ashraf, and H. Foroosh. Action recognition based on homography constraints. In *2008 19th International Conference on Pattern Recognition*, pages 1–4, 2008.

[80] Yuping Shen and Hassan Foroosh. View-invariant action recognition using fundamental ratios. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–6, 2008.

[81] S. Zhang, H. Yao, S. Liu, X. Chen, and W. Gao. A covariance-based method for dynamic background subtraction. In *Proc. ICPR*, pages 1–4, 2008.

[82] T. Zhao, M. Aggarwal, R. Kumar, and H. Sawhney. Real-time wide area multi-camera stereo tracking. In *Proc. CVPR*, 2005.

[83] J. Zhong and S. Sclaroff. Segmenting foreground objects from a dynamic textured background via a robust kalman filter. In *Proc. ICCV*, 2003.

[84] Q. Zhu, L. Shao, Q. Li, and Y. Xie. Recursive kernel density estimation for modeling the background and segmenting moving objects. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 1769–1772, May 2013.

[85] Z. Zivkovic. Improved adaptive gaussian mixture model for background subtraction. In *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.*, volume 2, pages 28–31 Vol.2, Aug 2004.

[86] Z. Zivkovic and F. van der Heijden. Efficient adaptive density estimation per image pixel for the task of background subtraction. *Pattern Recogn. Lett.*, 27(7):773–780, May 2006.