



University of Kentucky
UKnowledge

Theses and Dissertations--Electrical and
Computer Engineering

Electrical and Computer Engineering

2016

ROBUST BACKGROUND SUBTRACTION FOR MOVING CAMERAS AND THEIR APPLICATIONS IN EGO-VISION SYSTEMS

Hasan Sajid

University of Kentucky, hasan.sajid@gmail.com

Digital Object Identifier: <http://dx.doi.org/10.13023/ETD.2016.389>

[Right click to open a feedback form in a new tab to let us know how this document benefits you.](#)

Recommended Citation

Sajid, Hasan, "ROBUST BACKGROUND SUBTRACTION FOR MOVING CAMERAS AND THEIR APPLICATIONS IN EGO-VISION SYSTEMS" (2016). *Theses and Dissertations--Electrical and Computer Engineering*. 92.

https://uknowledge.uky.edu/ece_etds/92

This Doctoral Dissertation is brought to you for free and open access by the Electrical and Computer Engineering at UKnowledge. It has been accepted for inclusion in Theses and Dissertations--Electrical and Computer Engineering by an authorized administrator of UKnowledge. For more information, please contact UKnowledge@lsv.uky.edu.

STUDENT AGREEMENT:

I represent that my thesis or dissertation and abstract are my original work. Proper attribution has been given to all outside sources. I understand that I am solely responsible for obtaining any needed copyright permissions. I have obtained needed written permission statement(s) from the owner(s) of each third-party copyrighted matter to be included in my work, allowing electronic distribution (if such use is not permitted by the fair use doctrine) which will be submitted to UKnowledge as Additional File.

I hereby grant to The University of Kentucky and its agents the irrevocable, non-exclusive, and royalty-free license to archive and make accessible my work in whole or in part in all forms of media, now or hereafter known. I agree that the document mentioned above may be made available immediately for worldwide access unless an embargo applies.

I retain all other ownership rights to the copyright of my work. I also retain the right to use in future works (such as articles or books) all or part of my work. I understand that I am free to register the copyright to my work.

REVIEW, APPROVAL AND ACCEPTANCE

The document mentioned above has been reviewed and accepted by the student's advisor, on behalf of the advisory committee, and by the Director of Graduate Studies (DGS), on behalf of the program; we verify that this is the final, approved version of the student's thesis including all changes required by the advisory committee. The undersigned agree to abide by the statements above.

Hasan Sajid, Student

Dr. Sen-Ching Samson Cheung, Major Professor

Dr. Cai-Cheng Lu, Director of Graduate Studies

ROBUST BACKGROUND SUBTRACTION FOR MOVING CAMERAS AND THEIR
APPLICATIONS IN EGO-VISION SYSTEMS

DISSERTATION

A dissertation submitted in partial fulfillment of the
requirements for the degree of Doctor of Philosophy in the
College of Engineering
at the University of Kentucky

By
Hasan Sajid

Lexington, Kentucky

Director: Dr. Sen-Ching Samson Cheung, Professor of Electrical and Computer
Engineering
Lexington, Kentucky

2016

Copyright © Hasan Sajid 2016

ABSTRACT OF DISSERTATION

ROBUST BACKGROUND SUBTRACTION FOR MOVING CAMERAS AND THEIR APPLICATIONS IN EGO-VISION SYSTEMS

Background subtraction is the algorithmic process that segments out the region of interest often known as foreground from the background. Extensive literature and numerous algorithms exist in this domain, but most research have focused on videos captured by static cameras. The proliferation of portable platforms equipped with cameras has resulted in a large amount of video data being generated from moving cameras. This motivates the need for foundational algorithms for foreground/background segmentation in videos from moving cameras. In this dissertation, I propose three new types of background subtraction algorithms for moving cameras based on appearance, motion, and a combination of them. Comprehensive evaluation of the proposed approaches on publicly available test sequences show superiority of our system over state-of-the-art algorithms.

The first method is an appearance-based global modeling of foreground and background. Features are extracted by sliding a fixed size window over the entire image without any spatial constraint to accommodate arbitrary camera movements. Supervised learning method is then used to build foreground and background models. This method is suitable for limited scene scenarios such as Pan-Tilt-Zoom surveillance cameras. The second method relies on motion. It comprises of an innovative background motion approximation mechanism followed by spatial regulation through a Mega-Pixel denoising process. This work does not need to maintain any costly appearance models and is therefore appropriate for resource constraint ego-vision systems. The proposed segmentation combined with skin cues is validated by a novel application on authenticating hand-gestured signature captured by wearable cameras. The third method combines both motion and appearance. Foreground probabilities are jointly estimated by motion and appearance. After the mega-pixel denoising process, the probability estimates and gradient image are combined by Graph-Cut to produce the segmentation mask. This method is universal as it can handle all types of moving cameras.

KEYWORDS: Background Subtraction, Foreground Segmentation, Freely Moving Cameras, Pan-Tilt-Zoom, Ego-motion Compensation, Mega-Pixels.

Author's signature: Hasan Sajid

Date: September 7, 2016

ROBUST BACKGROUND SUBTRACTION FOR MOVING CAMERAS AND THEIR
APPLICATIONS IN EGO-VISION SYSTEMS

By

Hasan Sajid

Director of Dissertation: Sen-Ching S. Cheung

Director of Graduate Studies: Cai-Cheng Lu

Date: September 7, 2016

This work is dedicated to my family, especially my mother, father, youngest sister, wife
and daughters

ACKNOWLEDGEMENTS

First I would like to thank my advisor, Dr. Sen-Ching Samson Cheung, who has guided my research at the Multimedia Information Analysis lab. I had a wonderful learning experience in my dissertation research. Next, I would like to thank to my dissertation committee members, Dr. YuMing Zhang, Dr. Kevin Donohue and Dr. Nathan Jacobs, for their time and valuable suggestions.

I would also like to thank my friends and all the MIA lab members who have supported me and helped me during my dissertation research. They have always been willing to help me when I encountered difficulties.

Finally, I would like to thank my family for their unconditional support and encouragement. Without them, I would never have been able to finish my dissertation.

TABLE OF CONTENTS

Acknowledgments.....	iii
Table of Contents	iv
List of Tables	vii
List of Figures	viii
Chapter 1 Introduction	1
1.1. Broader Impact and Applications.....	5
1.2. Contribution of Dissertation.....	7
1.3. Organization	8
Chapter 2 Related Works	9
2.1. Background Subtraction Algorithms.....	9
2.1.1. Model-based algorithms	9
2.1.2. Motion-based algorithms	12
2.1.3. Hybrid algorithms.....	13
2.2. Authentication mechanisms	14
Chapter 3 Background Subtraction for Videos with Camera Jitter	16
3.1. Algorithm Overview	16
3.2. Multiple Color Spaces for Background Subtraction	19
3.3. Background Modelling.....	21
3.4. Binary Classification	23
3.4.1. Color-channels Activation/Deactivation	23
3.4.2. Pixel-level Probability Estimation.....	23
3.4.3. Mega-Pixel De-noising.....	24
3.5. Experiments and Results	28
3.5.1. Dataset and Evaluation Metrics	28
3.5.2. Parameter Selection	29
3.5.3. Quantitative Comparison.....	30
Chapter 4 Appearance-based Background Subtraction	33
4.1. Algorithm Overview	33
4.2. Feature Extraction	35

4.3. Model Formation.....	36
4.4. Classification.....	37
4.5. Evaluation on CDnet 2014 Dataset	38
4.5.1. Dataset and Evaluation Metrics	38
4.5.2. Parameter Selection	38
4.5.3. Quantitative Comparison.....	39
4.6. Evaluation on Hopkins155 Dataset	45
4.6.1. Dataset and Evaluation Metrics.....	45
4.6.2. Parameter Selection	46
4.6.3. Quantitative Comparison.....	46
Chapter 5 Motion-based Background Subtraction.....	49
5.1. Algorithm Overview	49
5.2. Motion Segmentation Module.....	50
5.2.1. Motion Feature Extraction.....	51
5.2.2. Iterative Polynomial Fitting.....	51
5.3. Mega-Pixel De-noising	54
5.3.1. Mega-Pixel Formation.....	55
5.3.2. Probability De-noising.....	55
5.4. Graph-Cut Optimization.....	57
5.5. Experiments and Results	57
Chapter 6 Hybrid Background Subtraction.....	60
6.1. Algorithm Overview	60
6.2. Motion Segmentation Module.....	62
6.3. Appearance Module	62
6.3.1. Model Initialization and Formation.....	62
6.3.2. Pixel-wise Probability Estimation	66
6.3.3. Model Update	66
6.4. Mega-Pixel Denoising and Probability Fusion	68
6.5. Graph-Cut Optimization.....	69
6.6. Experiments and Results	70
Chapter 7 Application: In-Air Signature Recognition and Authentication	74
7.1. Introduction	74
7.2. SIGAIR Dataset.....	75

7.3. Proposed System	78
7.3.1. Signature Extraction Module.....	78
7.3.2. Signature Verification Module	82
Chapter 8 Conclusion.....	85
Bibliography	88
Vita.....	100

List of Tables

Table 3.1 CDnet 2014 Camera Jitter test sequence details.	28
Table 3.2 Results for badminton test sequence.....	30
Table 3.3 Results for boulevard test sequence.....	30
Table 3.4 Results for sidewalk test sequence.....	31
Table 3.5 Results for traffic test sequence.	31
Table 3.6 Overall comparison on CDnet 2014 Camera Jitter Category.	31
Table 4.1 CDnet 2014 PTZ and BL test sequence details.	39
Table 4.2 Results for continuousPan test sequence.	40
Table 4.3 Results for intermittentPan test sequence.	40
Table 4.4 Results for twoPositionPTZCam test sequence.	41
Table 4.5 Results for zoomInZoomOut test sequence.	41
Table 4.6 Overall comparison on CDnet 2014 PTZ Category.....	41
Table 4.7 Overall comparison on CDnet 2014 Baseline Category.....	42
Table 4.8 Overall Results with different features on Cdnet 2014 PTZ category.....	44
Table 4.9 Processing time comparison.	45
Table 4.10 Results for test sequences of Hopkins155 dataset.	47
Table 4.11 Overall results on Hopkins155 dataset.	47
Table 4.12 Overall Results with different features on Hopkins155 dataset.	48
Table 5.1 Comparison of Proposed method with other methods. Red font is for best, whereas blue font represents second best method.	59
Table 6.1 Comparison of Proposed method with other methods on short test sequences. Red font is for best, whereas blue font represents second best method.....	72
Table 6.2 Comparison of Proposed method with other methods on long test sequences. Red font is for best, whereas blue font represents second best method.	73
Table 7.1 SIGAIR Dataset Variation and Scenarios.....	77

List of Figures

Figure 1.1 Left: Signing with Google-Glass. Middle: Image captured from Google-Glass. Left: SuBSENSE Segmentation of the middle image.....	4
Figure 3.1 System Overview.....	17
Figure 3.2 Mega-Pixel Formation and Probability Denoising.....	25
Figure 3.3 Comparison of segmentation with probability measure of each pixel individually (left), SP based average motion probability estimation (middle), and MP based average motion probability estimation (right).....	27
Figure 3.4 Input image (Row 1), Ground truth (Row 2) and Proposed method (Row3). Results on CDnet 2014 CJ category.	32
Figure 4.1 BoFs-SVM.....	34
Figure 4.2 Input Image (row 1), BoFs-SVM output (row 2), EFIC output(row 3), and subSENSE output(row 4). CDnet 2014 dataset: continuousPan test sequence(columns 1-4) and zoomInZoomOut(columns 5-7) test sequences.....	43
Figure 4.3 Input Image (row 1), BoFs-SVM output (row 2), EFIC output (row 3), and subSENSE output (row 4). CDnet 2014 dataset – twoPositionPTZCam (columns 1-4) and intermittentPan (columns 5-7) test sequences.....	43
Figure 4.4 Input Image (row 1) and BoFs-SVM output(row 2). Hopkins155 dataset: Cars1(column 1-2), people1(column 3-4) and people2(column5-6).	47
Figure 5.1 System Overview.....	50
Figure 5.2 Motion Segmentation Module.....	51
Figure 5.3 Motion segmentation accuracy and decreasing residual error with increasing number of iterations.	54
Figure 5.4 Mega-Pixel formation, Motion Correction and Graph-Cut optimization.....	54
Figure 5.5 Comparison of segmentation with motion probability measure only (column 1), SP based average motion probability measure (column 2), and MP based average motion probability measure (column 3).	55
Figure 5.6 Input image (row 1), Ground truth (row 2), and proposed system output (row 3). Cars2 (column 1), people1 (column 2), tennis (column 3), people2 (column 4), drive (column 5).	59

Figure 6.1 Algorithm Overview.....	61
Figure 6.2 GMM appearance model initialization and formation.	63
Figure 6.3 Mega-Pixel formation, Denoising and Graph-Cut optimization.	68
Figure 7.1 Google-Glass design and display (courtesy of Martin Missfeldt at http://www.brille-kaufen.org/en/googleglass)	76
Figure 7.2 Left: Signing with Google-Glass. Right: Image captured from Google-Glass.	77
Figure 7.3 Hand Segmentation and Fingertip tracking.	80
Figure 7.4 Signatures on tablet vs Signatures extracted from space by proposed system.	81
Figure 7.5 Normalized Intra(orange color) and Inter(blue color) Person DTW distance histogram.....	83

Chapter 1 Introduction

Background subtraction (BS) is one of the most widely used pre-processing steps in computer vision applications. The goal is to segment out the foreground (FG) from background (BG) in any given scene. It is a well-researched area in computer vision with significant amount of literature and numerous state of the art algorithms [1]. The focus of most research in background subtraction has been on stationary cameras. On the other hand, most videos captured in real life are from moving cameras, ranging from traditional Pan-Tilt-Zoom (PTZ) cameras and hand-held camcorders to the latest smart phones, wearables and dashboard cameras. BS is far more challenging in the case of moving camera as neither FG nor BG pixels are stationary. As large percentage of video content is produced by moving camera, the need for foundational algorithms that can isolate interesting areas in such videos is becoming increasingly pressing.

There are three general approaches for BS: model-based, motion-based and hybrid methods. Model-based approaches construct a model of the background and then compares the pixels of an input image with the model to label them as FG or BG. The main assumption of these algorithms is that the camera remains static and therefore are unsuitable to handle moving camera problem. To overcome this limitation, ego-motion compensation is first applied followed by application of conventional BS algorithm for FG detection [2, 3, 4, 5]. Despite motion compensation, these approaches fail when the assumption of homographic camera motion does not hold, i.e. when the camera center is moving or the complex BG cannot be approximated as a planar surface.

Motion-based algorithms exploit different motion patterns of FG objects and BG. They are more commonly referred to as motion segmentation. In motion segmentation, the moving objects are continuously present in the scene, and the background may also change due to camera motion. In general, motion-based approaches [6, 7, 8, 9] use motion vectors or track feature points followed by a clustering step. The use of these methods is limited by requirement of prior information such as number of FG objects and post-processing to obtain dense segmentation mask. Apart from the aforementioned limitations, such methods fail entirely when both FG and BG are at rest or FG has the same motion as the BG.

Hybrid methods such as [10] and [11] combine both appearance and motion information in an online framework. Motion information in the initial frames is used to initialize FG and BG appearance models, which are then continuously maintained and updated overtime. Classification is done using the appearance model. Although these methods are more powerful than motion-based and appearance-based algorithms, they are prone to view geometric degeneracies such as small frame-to-frame motion, planar scene, and zero camera translation. The need for special initialization procedures and computationally expensive nature limits their applications in real world scenarios.

An ideal BS algorithm should be able handle the aforementioned limitations of model-based, motion-based and hybrid methods. It should have five key traits. First, the algorithm should rely on both motion and appearance. It must be able to continuously update as well as maintain the BG model for constantly changing BG. This offers numerous advantages. First, it allows the algorithm to deal with scenarios, when there is no FG motion or FG and BG exhibit similar motion. Second, the use of BG model in conjunction with motion allows to cope up with dynamic background by exploiting appearance.

The second important trait is the ability to perform online segmentation rather than offline. The offline algorithm requires all frames for segmentation, which is not feasible in many real world applications and can easily become intractable for long videos sequences. Therefore, a universal BS algorithm must be able to do online segmentation for live feed or arbitrarily long video sequences.

The third key trait is the independence from special initialization procedures. It should not rely on any explicit camera motion models nor should it make any assumptions about the scene. This is one of the inherent limitation of existing model-based and hybrid methods.

The fourth trait is non-requirement of any prior information such as number of FG objects, contours etc. The algorithm should be able to automatically identify the correct number of FG objects.

The fifth trait is that such algorithms in addition to being online are computationally inexpensive and efficient. This is very critical taking into account the emerging market and potential growth of resource constrained wearable devices. Head Mounted Wearable Computer (HMWC) such as Google-Glass and Microsoft's HoloLens are particularly popular due to their ability in capturing the viewing perspective of the user and hence open up multiple avenues for research and applications ranging from personal use to law enforcement and healthcare to name a few.

Traditional segmentation algorithms fail on these devices because of the unique challenges associated with wearable cameras. First, wearable camera is likely to be constantly moving and very little assumption can be made about the scene in the video. In

Figure 1.1, the middle and rightmost images show the captured frame and the segmentation mask using the SuBSENSE [12], one of the best background subtraction algorithms as evaluated at the CDnet website [13]. The white region in the mask is supposed to represent the foreground. One can see that the background segmentation is unable to identify the hand at all. Second, the existing segmentation algorithms either for static or moving camera are computationally expensive and cannot cater for real time applications.

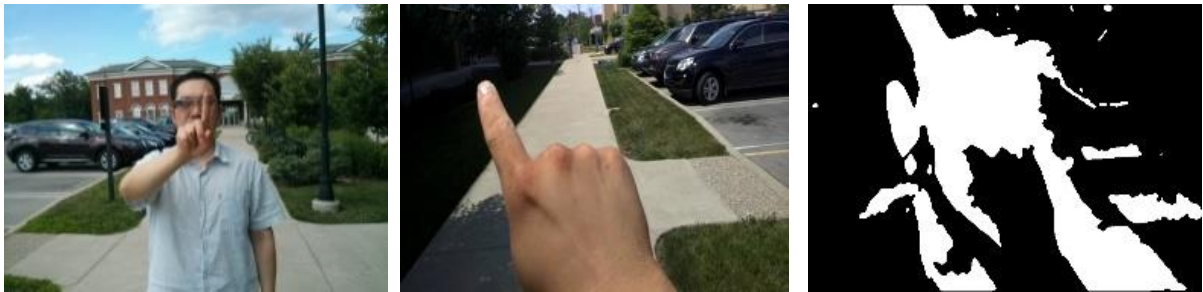


Figure 1.1 Left: Signing with Google-Glass. Middle: Image captured from Google-Glass. Left: SuBSENSE Segmentation of the middle image.

In context of HMWCs, we aim to test the motion based segmentation on well-known Google-Glass platform. The pervasiveness, size and portability of such devices make them prone to theft and hence purport the need of a robust authentication mechanism. The lack of physical interfaces such as keyboards or touch pads limits the choice of authentication mechanisms. The most natural way to introduce a robust authentication mechanism in wearable devices is to exploit the built-in hardware among which color camera is the most common sensor and is found in almost every wearable device. Therefore, we propose Virtual-Signature (VSig) [14], a hand-gestured signature performed by an individual and recognized via the wearable camera. This approach combines the

strength of familiar knowledge-based authentication mechanism [15] based on a person's own signature and the ultra-portability of a HMWC without the need of a writing surface.

Apart from moving camera segmentation, this application poses additional challenges including localization of the fingertip, robust algorithms to handle the variability of hand signing, and adequate visual feedback to user to stay within the field of view of the camera. However, the most important component for success is the underlying segmentation. A picture of a user signing his name with our VSig system while walking outdoor is shown in the leftmost image of Figure 1.1.

1.1. Broader Impact and Applications

Background subtraction allows to identify the important parts/objects in an image or a video. The availability of compact yet powerful computing platforms has made camera an integral part of virtually every device surrounding us. The types of camera ranges from PTZ cameras, hand-held camcorders, dashboard cameras, smart phones to head mounted cameras that support prolonged and high-quality recording. The pervasiveness of cameras and amount of video data generated on daily basis has on one side opened up multiple research opportunities and applications but at the same time poses unique challenges in terms of privacy invasion and computational requirements associated with large amounts of multimedia data.

This dissertation provides a complete range of background subtraction algorithms for platforms with very low to high computational capabilities. The wearable technology is expected to have significant growth in the coming years. The head mounted wearable devices are limited in terms of computational capability and lack physical interface. The

low complexity motion-based algorithm can be used to accurately segment out an individual's hands to recognize gestures for hands-free control and operation of such devices. In the wake of fatal encounters between police and citizens, the officers are now equipped with wearable cameras. The use of wearable camera by law enforcement raises privacy concerns. Such segmentation algorithms can not only ensure privacy but also assist in focusing on potential threats. One can imagine endless possibilities in healthcare, automotive, entertainment industries.

The hybrid method offers a much powerful algorithm for research into high level problems. For example combined with radio frequency technology it can be used for privacy protected video surveillance. Such a system can be employed in clinic as well natural settings for behavioral studies, which is not possible otherwise. Likewise, it can serve an important tool for security and surveillance while ensuring an individual's right to privacy.

The proposed technology with capability to handle camera movements allows for security and surveillance, behavior monitoring, anomaly detection with additional perspectives and information unavailable before. It can be used to segment different types of actions and fed to a learning engine. The automatic segmentation of huge amounts of data make many of previously intractable problems because of labor intensive labeling now possible. It offers limitless possibilities for scene understanding, in robotics, visual surveillance (e.g. anomaly detection, people counting), smart environments (e.g. fall detection, parking occupancy) and video retrieval (tracking, localization).

1.2. Contribution of Dissertation

The main contribution of this dissertation is a complete set of Background Subtraction algorithms, which can robustly handle videos generated from Hand-Held, PTZ as well as freely moving cameras. The first BS algorithm is purely an appearance-based approach that can robustly identify FG objects from PTZ camera sequences. The proposed method extracts multiple features (color, intensity, texture and gradient) by sliding a fixed size window over the entire image and learns a global FG/BG model without any spatial constraint. Foregoing spatial constraint is advantageous for moving camera scenarios where BG is continuously changing. The proposed algorithm is in contrast to existing methods which impose spatial constraint and maintain individual models for each pixel.

The second BS algorithm primarily relies on motion to differentiate FG/BG and uses color information only to denoise motion vectors at pixel level. This method involves two key innovations. The first innovation is an iterative low rank approximation of BG motion, which is compared with original motion vectors to yield initial FG probability estimates. The second innovation is the Mega-Pixel denoising process, which performs spatial regulation over the initial FG probability estimates to produce accurate FG probability estimates. Unlike other methods, the algorithm does not require any special initialization procedure nor does it need to maintain an appearance model, making it computationally efficient.

The third and most powerful algorithm combines both motion and appearance based algorithms in an online framework. Using a set of initial frames, a set of highly reliable FG and BG candidates are obtained from the motion module. Corresponding color

features are extracted for these candidates and separate FG and BG appearance models are learnt. For FG/BG labelling both motion-based and appearance-based probability estimates are first denoised and then average of two probability estimates is computed. Unlike existing methods, it does not require any prior information nor does it restrict camera motion or scene geometry. The proposed method builds global models for FG and BG, which makes it computationally more efficient than existing hybrid and model-based methods that build pixel-wise models.

The second major contribution is an In-Air signature Recognition and Authentication mechanism for HMWCs. It is a direct application of proposed motion-based segmentation on wearable devices. Additionally, we introduce a new dataset named SIGAIR, which comprises of signatures captured from Head Mounted Wearable Computer (HMWC).

1.3. Organization

The dissertation is organized into the following chapters: Chapter 2 details the related work on motion-based, model-based and hybrid BS algorithms as well as authentication mechanisms. Chapter 3 presents the background subtraction algorithm for camera jitter. In Chapter 4 the appearance-based BS algorithm is detailed, which is followed by motion-based BS algorithm in chapter 5. The hybrid method is detailed in chapter 6. In chapter 7, the In-Air Signature Recognition and Authentication mechanism for Google-Glass is presented. The dissertation is concluded in chapter 8.

Chapter 2 Related Works

In this chapter, we provide an in depth review of existing state-of-the-art background subtraction algorithms as well as authentication mechanisms.

2.1. Background Subtraction Algorithms

The existing background subtraction methods can be broadly divided into three categories: model-based, motion-based and hybrid algorithms.

2.1.1. Model-based algorithms

Model-based methods construct a statistical model of the background scene. The statistics can range from simple mean to complex multi-modal distributions. Pixel-based algorithms form a statistical model for each pixel in an image by considering its color only. The most popular algorithms in this category are Gaussian Mixture Model (GMM) [16, 17, 18] and Kernel Density Estimates (KDE) [19, 20]. GMM models each pixel distribution using a mixture of Gaussians. In [21] authors introduce shareable GMM models. Each pixel dynamically searches for the best matched model in its neighborhood, which is then used for classification. Many variants of GMM based methods have been proposed and they are summarized in [1]. KDE accumulates pixel's recent history and estimates a non-parametric probability distribution for each pixel. This approach overcomes the problem of determining the appropriate number of components used in GMM.

Sample consensus is another non-parametric method that relies on recently observed pixels to determine if the incoming pixel is a FG or BG. PAWCS is an example

of sample consensus methods that introduces word based model capable of capturing and retaining color and Local Binary Similarity Pattern (LBSP) features over long periods of time. A novel pixel-level feedback loop mechanism is an integral part of the system, which allows to continuously update and maintain the pixel's model [22]. The spatiotemporal LBSP feature descriptor increases the segmentation accuracy but entails high computational costs.

Codebook is another class of techniques that has been reported in [23] [24]. It comprises of a codebook for each pixel which is a compressed form of background. Each codebook has multiple codewords that are based on a sequence of training images using a color distortion metric. Incoming pixels are matched against all background codewords for classification.

Another class of algorithms take into account inter-pixel spatial dependencies and assume that a pixel undergoes the same change as its neighbor. In [25], the authors incorporate spatial information by using statistical circular shift moments (SCSM) in image regions. In [26], the authors present a block based approach that compares a block in current frame to its reconstruction from PCA coefficients, and labels it as BG if the reconstruction is close. In [12], the authors consider a 5 x 5 grid to compute local binary pattern features and combine them with color. Another approach is presented in [27] in which the authors not only consider the history of intensity values of pixel itself but also its neighbors. Although region based methods take into account the inter-pixel spatial dependencies, they typically assume static camera and fail to handle videos from moving camera as there is no explicit mechanism to account for the movement of the BG regions.

Machine learning based FG/BG classification methods have also been reported in literature [28, 29, 30, 31, 32]. In [29] a one-class SVM is trained for each pixel over a number of frames. [28] proposes a single class SVM for handling dynamic background by extracting features from the neighborhood around each pixel. [31] divides an image into a predefined number of equally sized blocks and trains one class SVM for each block. [32] is another block-based method that builds and maintains model for each block. These methods train models for each pixel or block thereby imposing spatial constraint and therefore prone to failure in case of a moving camera. In [30], the authors tackle the moving camera problem by introducing two one-class SVMs to separately model FG and BG color distributions for each pixel. Unlike our frame-based approach, their classifier is trained on local neighborhood, making it both computationally expensive and brittle for large camera movements. Additionally, the need for manual labeling limits its application for real world scenarios.

To cope with the moving camera problem, another set of algorithms estimate BG through ego-motion compensation [2, 33, 3, 34, 35, 4, 36, 5]. These methods estimate camera motion and then apply conventional background subtraction algorithms to detect FG [2, 33, 3, 34, 35]. More recent work estimates a homography between successive image frames, followed by a registration process to compensate motion [4, 36, 5]. Residual pixels can be further registered using parallax estimation [4]. Most of these methods work well when the scene can be approximated as a planar surface or the camera center is fixed, such as a PTZ surveillance camera.

2.1.2. Motion-based algorithms

The motion-based algorithms rely on the assumption that for any given number of images, the foreground moves differently from the background [37, 38]. These algorithms exploit the difference in motion patterns to segregate FG from the BG. These algorithms are able to handle large camera motions but assume rigid or smooth motion in all BG regions. This is however not true since dynamic background can comprise of non-rigid motions such as waving trees. Another challenge is when the foreground object itself is at rest and follows the same camera motion as the rest of the static background. Other motion-based algorithm estimates background by finding regions that do not change in the sequence [39]. While it alleviates the static foreground problem, such an approach would fail in the case of dynamic background.

Motion-based algorithms can be divided into two categories: layer-based and point trajectory based. Layer-based methods [6, 40, 7, 41, 42, 43, 44, 8, 45] compute dense or sparse optical flows and then cluster them based on some measure of motion consistency. The problem of these approaches is that motion analysis can be quite erroneous in the presence of real world problems such as occlusion and video noise [46]. Methods based on sparse flows are further restricted by the need of an initialization step to establish prior information such as contours, number of objects, etc.

The point trajectory based methods [47, 9, 48] segment images based on point trajectory analysis. First, sparse feature points are detected and tracked in a sequence followed by clustering via spectral [49] or subspace [50] methods. Although these methods are robust in handling large camera motion, they only produce a segmentation of sparse

points, which need to be post-processed for dense segmentation [49]. Hence, the results of these algorithms rely heavily on point tracking throughout the video sequence and post-processing for dense segmentation, making them feasible only for offline processing.

2.1.3. Hybrid algorithms

Hybrid methods use the combination of motion, color and appearance. Lim et al. [51] proposed a block-based iterative appearance modeling method that combines temporal model propagation and spatial model decomposition. Fundamental matrix is used for the initial FG/BG labeling, which is iteratively refined by spatial and temporal smoothness. The proposed method fails to detect small FG objects and is prone to degeneracies in estimating fundamental matrix. Kwak et al. [10] improved the initialization procedure using belief propagation. They introduced a Bayesian filtering framework that combined block-based color appearance models with separate motion models for the BG and FG to estimate labels at each pixel. Although it resulted in better post-processing procedure but the proposed method is still prone to view geometric degeneracies such as small frame-to-frame motion, planar scene, and zero camera translation. The need for special initialization procedures limits their applications in real world scenarios. In order to overcome the limitations associated with fundamental matrix and homography transform, Zamalieva et al. [52] adopted a complementary approach and combined both of them. The proposed method performed a Bayesian selection of appropriate geometric relation between two consecutive frames. Based on selected transformation, the appearance models were propagated and maintained. Despite the improvement, the proposed method is sensitive to the presence of FG objects during initialization phase and requires FG free frames.

Elqursh et al. [11] proposed an online method that maintains pixel based models for both FG and BG. It relies on long term trajectories in low dimensional space, which are modelled as a mixture of Gaussians. The trajectories that do not lie in the space belong to the BG. The long term trajectories, motion and appearance models are combined in a Bayesian filtering framework to obtain the final labels. The shortcoming of proposed method lies in the number of components of parametric Gaussian mixture model, which can vary from scene to scene. In [53], Narayana et al. proposed another hybrid method that primarily uses optical flow orientations to group pixels based on their orientations. A probabilistic framework is employed to automatically identify the correct number of FG objects. The orientation based segmentation is further refined by the use of FG and BG color appearance models for each pixel from previous frames. The proposed method is robust against pure camera translation but fails in case of rotation. Explicit modeling of camera rotation is required to handle such cases.

2.2. Authentication mechanisms

Numerous authentication mechanisms have been reported in literature. We focus our attention on authentication mechanisms which are closely related or applicable to wearable devices. Two dominant methods exist: the first approach is based on vision sensors using either color or depth. The second approach is based on sensors such as accelerometers.

The first type of authentication mechanisms such as [15, 54, 55] rely on image sensors and employ color and depth information to track or segment out hand or fingertip of a person. This is followed by post processing to extract trajectories and features such as position, velocity and acceleration. Finally, signature is matched against pre-stored

signatures for authentication. Although these systems produce accurate results, their usages are limited to a well-controlled indoor environment. Also, they are computationally expensive with requirement of depth in addition to color information. It is important to note that most existing wearable devices do not have any built-in depth sensor and would therefore incur additional hardware cost, making this approach unsuitable for resource constraint wearable devices.

The second type of authentication mechanisms such as [56, 57] are based on readily available accelerometers. This approach requires additional hardware and circuit to capture the hand movement and extract trajectory, which is then transmitted to main device. More recent smart phones have built in accelerometers and can perform gesture recognition without additional hardware. However, accelerometer is a much coarser device compared to camera and is capable only to differentiate simple gestures. For authentication, the user is required to remember lengthy gesture sequences, which are not as straightforward and natural as gestures that are based on the hand-written signature.

The most natural way to introduce a robust authentication mechanism in wearable devices is to exploit the built-in hardware among which color camera is the most common sensor and is found in almost every wearable device. The proposed approach is based on this theme.

Chapter 3 Background Subtraction for Videos with Camera Jitter

In this chapter, we present a novel BS algorithm that can robustly handle videos affected by camera jitter or shaking. We also compare our algorithm with several other state-of-the-art algorithms on camera jitter category of the CDnet-2014 dataset, which is a comprehensive dataset for BS evaluation [13].

3.1. Algorithm Overview

The proposed approach comprises of multiple innovative mechanisms in background modeling, pixel classification and the use of multiple color spaces. The system first creates multiple background models of the scene and stores them in a Background Model Bank (BMB). This is followed by coarse FG probability estimation for each pixel. Next, the image pixels are merged together to form mega-pixels, which are used to spatially denoise the coarse probability estimates to generate binary masks for each of the color channels of both RGB and YCbCr color spaces. The masks generated after processing these input images are then combined to separate foreground pixels from the background. The proposed system consists of five steps as shown in Figure 3.1. Each step is described below.

Step 1: BG Model Selection

The first step is to select an appropriate BG Model for the incoming frame. The selection criterion is based on identifying the BG model in Background Model Bank (BMB) that maximizes the correlation with input image $I(X)$. BMB simply comprises of multiple background models of the scene. The BMB formation process is detailed in section 3.3.

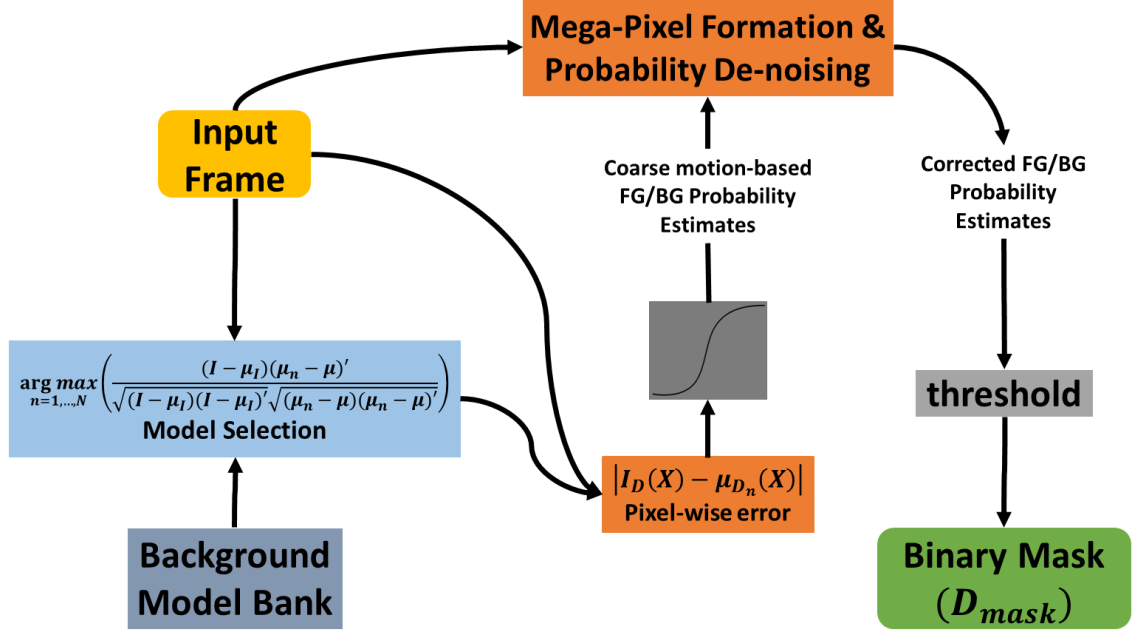


Figure 3.1 System Overview.

$$Corr = \arg \max_{n=1, \dots, N} \left(\frac{(I - \mu_I)(\mu_n - \mu)'}{\sqrt{(I - \mu_I)(I - \mu_I)' \sqrt{(\mu_n - \mu)(\mu_n - \mu)'}}} \right)$$

where, I and μ_n are vector forms of $I(X)$ and $\mu_n(X)$ respectively. μ_I and μ are defined as:

$$\mu_I = \frac{1}{|X|} \sum_j I_j \quad \text{and} \quad \mu = \frac{1}{|X|} \sum_j \mu_{n_j}$$

Step 2: Binary Mask (BM) Generation

The input image and the selected BG model are first used to generate coarse FG probability estimate for each pixel. The probability estimates and input image are then passed to the Mega-Pixel formation and probability denoising module. It segments the image into arbitrary number of MPs and uses the coarse probability estimates to calculate average probability estimate for each MP. The more accurate probability estimates are then

thresholded to generate Binary Mask (BM) for each color channel. We denote the BM for color channel D as $D_{mask}(X)$. The BM generation is discussed in detail in section 3.4.

Step 3: Binary Masks Aggregation/Fusion

The BMs are then used to form Foreground Detection (FGD) masks for RGB and YCbCr color spaces:

$$FGD_{mask}^{colorspace}(X) = \left[\sum_D (D_{mask}(X)) \right] > 1$$

For YCbCr color space, if Cb and Cr channels are deactivated then FGD_{mask}^{YCbCr} will be reduced to the Y channel BM alone. Finally the two FGD masks are combined by taking logical AND between dilated versions of the two to obtain the actual FGD mask:

$$FGD_{mask}(X) = \text{Dilate}(FGD_{mask}^{RGB}(X)) \& \text{Dilate}(FGD_{mask}^{YCbCr}(X))$$

The dilated versions are to ensure that all true foreground pixels are captured in the FGD mask.

Step 4: Binary Masks Purging

The FGD mask is then applied to each of the BMs obtained in step 3. This removes all of the falsely detected foreground regions and increases our confidence in classifying FG and BG pixels in the final step. The resulting component masks are defined as follows:

$$D_{mask}^{new}(X) = D_{mask}(X) \cdot \text{Dilate}(FGD_{mask}(X))$$

Step 5: Foreground Mask

In the final step of the process, FG mask is obtained by the logical OR of all the $D_{mask}^{new}(X)$ masks.

Our innovations primarily fall in the use of multiple color spaces, background model bank for modelling the background and Mega-Pixel denoising for accurate foreground detection. In the following sections, we detail each of these innovations

3.2. Multiple Color Spaces for Background Subtraction

The choice of color space is critical to the accuracy of foreground segmentation. Many different color spaces including RGB, YCbCr, HSV, HSI, lab2000, normalized-RGB (rgb) have been used for background subtraction. Among these color spaces, we focus on the four most widely-used color spaces: RGB, YCbCr, HSV and HSI [58] [59].

RGB is a popular choice for a number of reasons: (a) the brightness and color information are equally distributed in all three color channels; (b) it is robust against both environmental and camera noise [58]; (c) it is the output format of most cameras and its direct usage in BS avoids the computation cost of color conversion [59].

The use of the three other color spaces: YCbCr, HSV and HSI are motivated by human visual system (HVS). The defining color perception in HVS is that it tends to assign a constant color to an object even under changing illumination over time or space [58] [60]. These color spaces segregate the brightness and color information, with YCbCr on Cartesian coordinates whereas HSV and HSI on polar coordinates. While the color constancy makes the BS process more robust against shadow, highlights and illumination

changes, the foreground detection is less discriminatory if brightness information is not used [58][60][61][62].

In comparative studies on color spaces [58][59][63][61], YCbCr has been shown to outperform RGB, HSI and HSV color spaces and is considered to be the most suitable color space for foreground segmentation [58][61][59]. Due to its independent color channels, YCbCr is the least sensitive to noise, shadow and illumination changes. RGB is ranked second with HSI and HSV at the bottom as their polar coordinate descriptions are quite prone to noise [58]. The conversion from RGB to YCbCr is also computationally less expensive than to HSI or HSV.

Based on the above comparison, YCbCr is a natural choice for segmentation. However, [60] and [61] also identify potential problems with the YCbCr color space: when current image contains very dark pixels, the chance of misclassification increases since dark pixels are close to the origin in RGB space. The fact that all chromaticity lines in RGB space meet at the origin makes dark pixels close or similar to any chromaticity line. Such scenario does not occur only when illumination levels are low globally, but also happens when portion of the image becomes darker. This is common especially in indoor scenes with complex illumination sources and scene geometry. Shadows casted by objects is one such example. The exclusive use of YCbCr color space in such situations will result in a decrease in foreground segmentation accuracy.

To address this issue, we propose a solution motivated by our own color vision. In human visual system, color vision is provided through two types of cells; rods and cones. Rods are used for vision under low light known as *scotopic*, in which color vision is not

possible. At intermediate light levels ($0.01 - 1 \text{ cd/m}^2$), our vision is *mesopic*, in which both rods and cones are active. Under *mesopic* light conditions color discrimination is poor. At high levels ($>1 \text{ cd/m}^2$), our vision becomes *photopic*, where cone activity is best and allows for good color discrimination [64]. There are two key observations that motivates our design: (1) the use of two different types of cells, and (2) selection of appropriate cells for different lighting conditions.

Like the human visual system, we propose to use two color spaces: RGB and YCbCr to emulate the two types of cells. We then choose the appropriate channels for the scene in question. This is different from all existing techniques that employ all channels and only one color space. Analogous to the rod cells, RGB and Y channels are used under poor lighting conditions since chromatic information is uniformly distributed across RGB channels and Y represents intensity only. Similar to the cone cells under sufficient lighting condition, we additionally employ the color channels (Cb and Cr) of YCbCr color space to increase foreground segmentation accuracy. During intermediate lighting conditions, both RGB and YCbCr color spaces complement each other in providing a robust FG/BG classification.

3.3. Background Modelling

BG modelling is one of most important steps in a BS process and the accuracy of the model used directly impacts the segmentation results. Most BG models use a variant of multi-modal pixel-wise statistical background model. Such an approach has two problems: first, it is difficult to determine the correct number of modes for modelling the pixel probability distribution function. Second, and more importantly, inter-pixel dependencies are overlooked, which leads to poor segmentation results.

In order to model the BG, we propose Background Model Bank (BMB), which comprises of multiple BG models instead of a single BG model. To form BMB, each background training image is treated as a BG model with selected color channels stacked together as a vector. This initial set of BG models are then merged together into a number of average BG models using an iterative sequential clustering procedure. Two BG mean models (p and q in vector form) with correlation measure greater than the pre-defined parameter $corr_th$ are merged and replaced by their average. The correlation measure is defined as

$$Corr(p, q) = \left(\frac{(p - \mu_p)(q - \mu_q)'}{\sqrt{(p - \mu_p)(p - \mu_p)'} \sqrt{(q - \mu_q)(q - \mu_q)'}} \right)$$

where μ_p and μ_q are defined as:

$$\mu_p = \frac{1}{|X|} \sum_j p_j \text{ and } \mu_q = \frac{1}{|X|} \sum_j q_j$$

This process continues in an iterative fashion unless there are no more average BG models with $Corr > corr_th$.

The use of frame-level clustering is motivated by physical laws that govern scene geometry. Typically real-life scenes comprise of different types of objects. The variety in configurations and interactions between different types of matter and objects generate very intricate and infinite scene geometry. Examples include variations caused by illumination changes, dynamic changes, camera shaking, camera movement etc. This diversity makes

it difficult to accurately capture and model the scene. The use of multiple BG models allows us to capture scene more accurately while keeping spatial dependencies intact.

Another advantage of BMB is that it is computationally simpler than other multi-mode approaches since we choose a model at frame level and ignore the rest of the BG models in the BMB. While there is an additional cost on choosing the model at frame level, it incurs minimal cost because of simple comparison with average BG models than those that rely on pixel-based multi-mode distributions. The experimental results in section 3.5 demonstrate the effectiveness of multiple BG models in capturing scene diversity and camera variations accurately.

3.4. Binary Classification

In this section, we discuss the binary mask generation for each of the selected color channels. It involves color channel activation/deactivation, pixel-level probability estimation, MP formation and probability denoising.

3.4.1. Color-channels Activation/Deactivation

This step is responsible to activate/deactivate the color channels Cb and Cr. Both color channels are used if the mean intensity of input image is greater than empirically determined parameter *channel_th*, which otherwise are not employed.

3.4.2. Pixel-level Probability Estimation

Pixel-wise error, $e(X)$ is calculated between each color channel from both RGB and YCbCr spaces and the chosen BG model as follows.

$$e(X) = |I_D(X) - \mu_{D_n}(X)|$$

where D denotes the color channel in question, $I_D(X)$ is the input image, and $\mu_{D_n}(X)$ is the chosen average BG model.

Once we have calculated the error for each individual pixel, we estimate coarse probability pr for each pixel by passing them through the activation function.

$$pr(X) = \frac{2}{(1 + e^{-(2 * e(X))})} - 1$$

The rationale behind this conversion is that the higher the error the more likely that the pixel belongs to the FG.

3.4.3. Mega-Pixel De-noising

The Mega-Pixel de-noising process performs spatial regulation over raw probability estimates to produce more accurate probability estimates. The final probability estimates are calculated by taking mean of denoised probability estimates over a MP. It consists of two main steps: Mega-Pixel (MP) formation and probability de-noising. Figure 3.2 depicts the proposed MP de-noising process.

Step 1: Mega-Pixel Formation

The notion of Super-Pixel (SP) segmentation is increasingly popular due to its capability in capturing local context and significant reduction in computational complexity. These algorithms combine neighboring pixels into one pixel based on similarity measure such as color, texture, size etc.

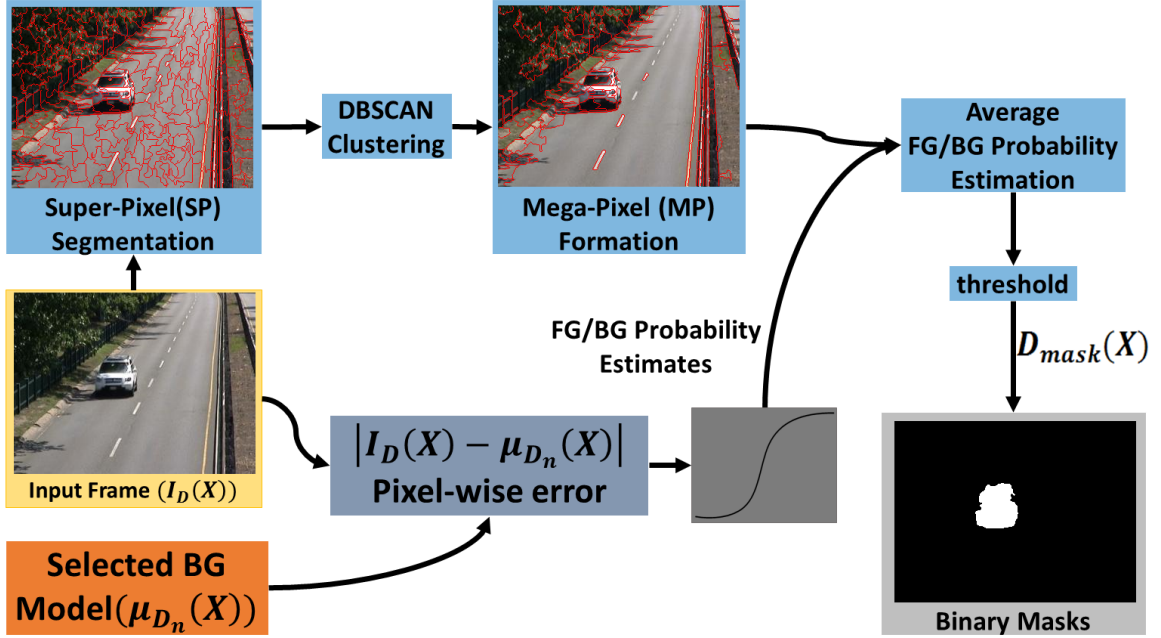


Figure 3.2 Mega-Pixel Formation and Probability Denoising.

In proposed method, we use ERS algorithm [65] to segment the input frame into R Super-Pixels (SP). The SP segmentation is formulated as a graph partitioning problem. For a graph $G = (V, E)$ and R number of SPs, the goal is to find a subset of edges $A \subseteq E$ to approximate a graph $\bar{G} = (V, A)$ with R connected sub-graphs. The vertex corresponds to a pixel in an image and an edge is formed by 4-connected neighborhood with weights computed based on similarity between connected vertices. The clustering objective function comprises of two terms: the entropy rate H of random walk and a balancing term B .

$$\max_A H(A) + \lambda B(A),$$

$$s. t. A \subseteq E \text{ and } N_A \geq R$$

where N_A , is the number of connected components in \bar{G} . The entropy term encourages compact and homogeneous clusters, whereas the balancing term encourages clusters with similar size. Finally, to overcome exact optimization difficulty, a greedy algorithm is used to solve the problem that always provides $\frac{1}{2}$ approximation bound. For more details, we refer readers to [65].

Once SPs are formed, these are combined together to form much bigger Mega-Pixels (MPs) using DBSCAN [66] clustering. DBSCAN is a density based clustering algorithms in which clusters are defined as high density areas, whereas the sparse regions are treated as outliers or borders to separate clusters. For DBSCAN clustering, we use implementation in [67]. For any 2 adjacent SPs p and q , distance function is based on mean Lab color difference and is defined as:

$$d(p, q) = |\mu_p^L - \mu_q^L| + |\mu_p^a - \mu_q^a| + |\mu_p^b - \mu_q^b|$$

$$\mu_p^c = \frac{1}{|p|} \sum_{x \in p} c(x)$$

where, μ_p^c represents the mean value of one of the color channels ($c = \{L, a, b\}$) of SP p . $|p|$ is the total number of pixels in SP p . Two SPs p and q are merged together into a MP if they are adjacent and $d(p, q) \leq \tau_{color}$ where τ_{color} is an empirically determined constant. Figure 3.2 depicts the overall MP formation process. Notice the road SPs correctly merged as a single MP.

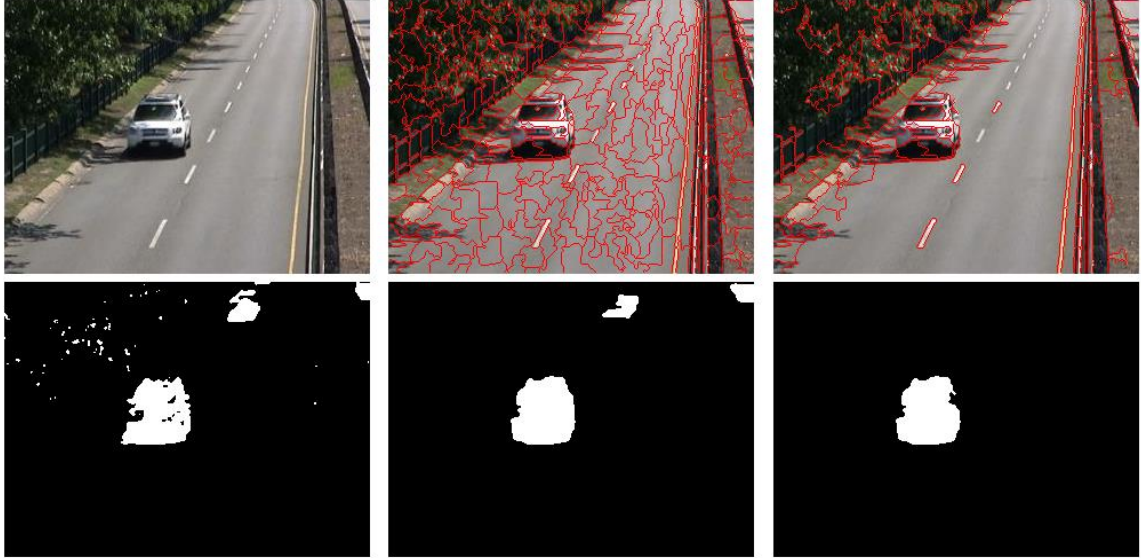


Figure 3.3 Comparison of segmentation with probability measure of each pixel individually (left), SP based average motion probability estimation (middle), and MP based average motion probability estimation (right).

Step 2: Probability De-noising

MP formation is followed by averaging probability estimation for each MP. Average probability (\overline{pr}) of a MP q is defined as:

$$\overline{pr} = \frac{1}{|q|} \sum_{x \in q} pr(x)$$

where, pr represents the coarse FG probability estimate for each pixel. The average probability (\overline{pr}) is then assigned to each pixel belonging to that MP. Finally, to obtain Binary Mask $D_{mask}(X)$ for each color channel D , the average probability measure is thresholded using an empirically determined parameter $prob_th$.

The use of MP and its respective AP allow us to assign the same probability to each pixel belonging to the same object and therefore increases the segmentation accuracy. For example, all the pixels belonging to the road in Figure 3.3 should be BG. Clearly, in Figure 3.3, as we move from left to right, road pixels with erroneous probability estimates would be averaged out using neighboring pixels via SPs or MP, thereby improving the segmentation accuracy. As MPs respect edge integrity, the average probability of a MP represents the same object or part rather than using FG/BG probability estimates for each individual pixel or SPs.

3.5. Experiments and Results

In this section, we evaluate the proposed method with state of the art algorithms on CDnet 2014 dataset camera jitter category. The dataset, parameter setting and quantitative evaluation are detailed below:

3.5.1. Dataset and Evaluation Metrics

The CDnet 2014 dataset [13] is one of the most comprehensive datasets available for evaluating BS algorithms.

Table 3.1 CDnet 2014 Camera Jitter test sequence details.

Test Sequence	Image Resolution	Training Data (Frame #s)	Testing Data (Frame #s)
CJ-badminton	720 x 480	1-799	800-1150
CJ-boulevard	352 x 240	1-789	790-2500
CJ-sidewalk	352 x 240	1-799	800-1200
CJ-traffic	320 x 240	1- 899	900-1570

Table 3.1 details the test sequences for Camera Jitter (CJ) category. The dataset specifies training and testing data to ensure consistency when comparing different algorithms. For evaluation purposes, CDnet recommends seven evaluation metrics. Let $TP = True\ Positive$, $FP = False\ Positive$, $TN = True\ Negative$ and $TP = True\ Positive$. The metrics are defined as:

1. Recall: $Re = \frac{TP}{TP+FN}$
2. Specificity: $Sp = \frac{TN}{TP+FN}$
3. False Positive Rate: $FPR = \frac{FP}{FP+TN}$
4. False Negative Rate: $FNR = \frac{FN}{FP+TN}$
5. Percentage of Wrong Classifications: $PWC = 100 \cdot \frac{FN+FP}{FP+FN+TN+TP}$
6. Precision: $Pr = \frac{TP}{TP+FP}$
7. F-Measure: $FM = \frac{2 \cdot Pr \cdot Re}{Pr+Re}$

An additional metric called average rank R is also defined to aggregate all seven metrics together, which is simply the average of each metric from all 4 test sequences in one category.

3.5.2. Parameter Selection

One set of parameters are used for the entire dataset: $corr_th=0.99$, $prob_th=0.75$, $R=300$, $\tau_{color}=3$ and $channel_th=100$. The parameter setting is based on the set that yields overall best results across all test sequences.

3.5.3. Quantitative Comparison

For quantitative evaluation we consider the top 3 methods reported on the CDnet website in CJ category: EFIC [68], PAWCS [22] and SharedModel [21]. Table 3.2 to Table 3.5 contain the results of proposed method and the top 3 algorithms on four test sequences, whereas the overall results are presented in Table 3.6. It is important to note that the results reported in this paper are official results computed by the CDnet administrator based on our submission of the binary masks. The ground-truth used in the evaluation are withheld by the administrator and unavailable to us.

Table 3.2 Results for badminton test sequence.

Method	Re	Sp	FPR	FNR	PWC	FM	Pr	Rank (R)
MBS [69] [70]	0.8972	0.9967	0.0032	0.1027	0.6676	0.9021	0.9070	1.57
EFIC [68]	0.9340	0.9930	0.0069	0.0659	0.9007	0.8767	0.8259	3.14
PAWCS [22]	0.9107	0.9953	0.0046	0.0892	0.7561	0.8920	0.8740	2.71
SharedModel [21]	0.8922	0.9966	0.0033	0.1077	0.6887	0.8988	0.9054	2.57

Table 3.3 Results for boulevard test sequence.

Method	Re	Sp	FPR	FNR	PWC	FM	Pr	Rank (R)
MBS [69] [70]	0.8731	0.9930	0.0069	0.1268	1.2550	0.8672	0.8613	2.14
EFIC [68]	0.8592	0.9985	0.0014	0.1407	0.7958	0.9101	0.9676	1.28
PAWCS [22]	0.8025	0.9951	0.0048	0.1974	1.3879	0.8444	0.8909	2.57
SharedModel [21]	0.7168	0.9921	0.0078	0.2831	2.0810	0.7638	0.8172	4

Table 3.4 Results for sidewalk test sequence.

Method	Re	Sp	FPR	FNR	PWC	FM	Pr	Rank (R)
MBS [69] [70]	0.9324	0.9962	0.0037	0.0675	0.5440	0.8993	0.8685	1.85
EFIC [68]	0.7375	0.9958	0.0041	0.2624	1.0854	0.7798	0.8273	3.14
PAWCS [22]	0.5580	0.9984	0.0015	0.4419	1.3049	0.6904	0.9050	2.71
SharedModel [21]	0.7366	0.9970	0.0029	0.2633	0.9698	0.7984	0.8715	2.28

Table 3.5 Results for traffic test sequence.

Method	Re	Sp	FPR	FNR	PWC	FM	Pr	Rank (R)
MBS [69] [70]	0.6255	0.9854	0.0145	0.3744	3.6963	0.6781	0.7404	2.85
EFIC [68]	0.8524	0.9684	0.0315	0.1475	3.8789	0.7323	0.6418	3.28
PAWCS [22]	0.8645	0.9851	0.0148	0.1354	2.2390	0.8278	0.7940	1.28
SharedModel [21]	0.8383	0.9821	0.0178	0.1616	2.6848	0.7953	0.7566	2.57

Our algorithm ranks first for both badminton and sidewalk test sequences, second for boulevard test sequence, and third for traffic test sequence. The multiple BG model approach and Mega-Pixel denoising innovation allows our algorithm to minimize false positives as a result of unwanted camera movements caused by jitter and shake.

Table 3.6 Overall comparison on CDnet 2014 Camera Jitter Category.

Method	Re	Sp	FPR	FNR	PWC	FM	Pr	Rank (R)
MBS [69] [70]	0.8321	0.9928	0.0071	0.1678	1.5407	0.8367	0.8443	1.85
EFIC [68]	0.8458	0.9889	0.0110	0.1541	1.6652	0.8247	0.8157	2.85
PAWCS [22]	0.7839	0.9935	0.0064	0.2160	1.4220	0.8136	0.8660	2.28
SharedModel [21]	0.7960	0.9919	0.0080	0.2039	1.6061	0.8141	0.8377	3

The overall comparison in Table 3.6 clearly indicates the superiority of our proposed algorithm: it ranks second in six out of seven metrics and achieves highest F-Measure of 83.67%. Red font in all tables represent the top, whereas blue font represents the second best. Lastly, Figure 3.4 shows sample qualitative results on the CDnet 2014 CJ test sequences.

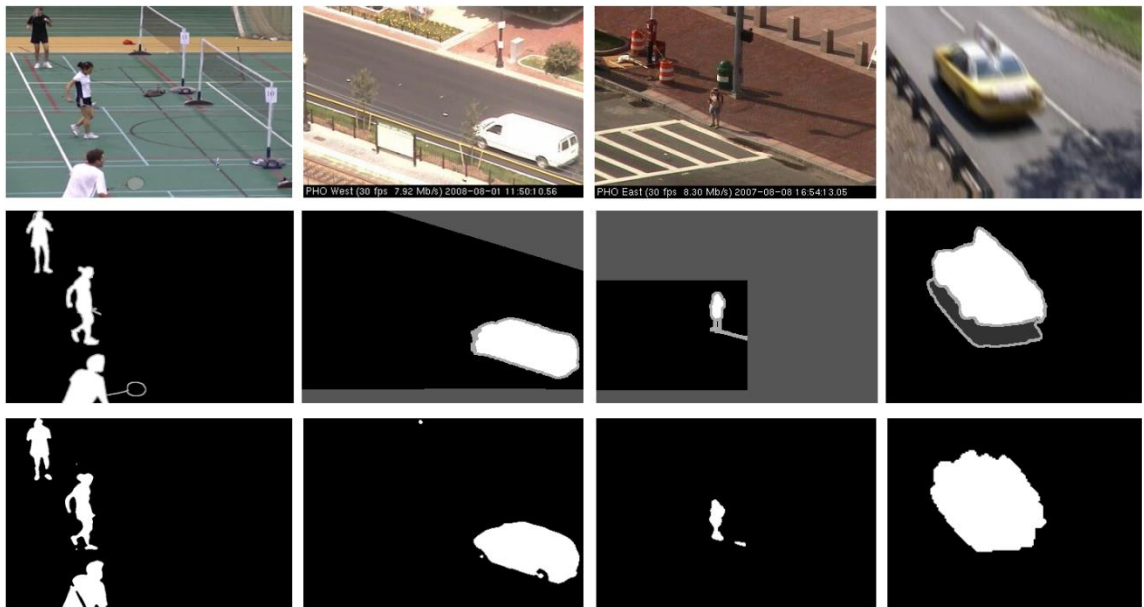


Figure 3.4 Input image (Row 1), Ground truth (Row 2) and Proposed method (Row3). Results on CDnet 2014 CJ category.

Chapter 4 Appearance-based Background Subtraction

In this chapter, we detail the appearance-based BS algorithm named BoFs-SVM that can robustly identify FG objects from PTZ camera sequences. We also compare our algorithm with several other state-of-the-art algorithms on pan-tilt-zoom and baseline categories of the CDnet-2014 dataset, which is a comprehensive dataset for BS evaluation [13], and Hopkins155 dataset, which is a specialized dataset for 3D motion segmentation [47].

4.1. Algorithm Overview

The proposed algorithm learns the background entirely based on the appearance features and their attributes extracted by a sliding window over each pixel to encode the BG into a Bag-of-Features (BoFs). The sliding window captures spatial dependencies at the local level. The extracted features then undergo Principal Component Analysis (PCA). The selected features are then concatenated into feature vectors to train a global FG/BG SVM model for classification. Our algorithm has a number of unique traits to handle moving camera scenes. The extraction of multiple features from image patches instead of individual pixels is the first key trait that makes our algorithm robust to a moving camera. The second key trait is the absence of any global spatial constraint on the features. This is advantageous for moving camera scenarios where BG is continuously changing and spatial constraints do not hold. This is in contrast to existing methods which impose spatial constraint and maintain individual models for each pixel. The selection of the most informative, scene-specific feature set is the third key trait of our algorithm. To avoid the ambiguity of

dynamic background and static foreground, our work focuses exclusively on appearance features. Our algorithm assumes the following:

1. The dominant motion from the scene is due to the camera's movement.
2. A small number of training frames, which can be externally provided or obtained through motion segmentation, are needed.

As a consequence of assumption 2, the proposed algorithm works best when the moving BG scene is predictable, such as those from a pan-tilt-zoom camera. Other moving cameras such as ego-vision cameras would require frequent retraining and adaptation, which is beyond the scope of this method. The proposed algorithm has three main components: Feature extraction, Model formation and Classification. Each component and its functional role are detailed in following sections. Figure 4.1 provides the system overview.

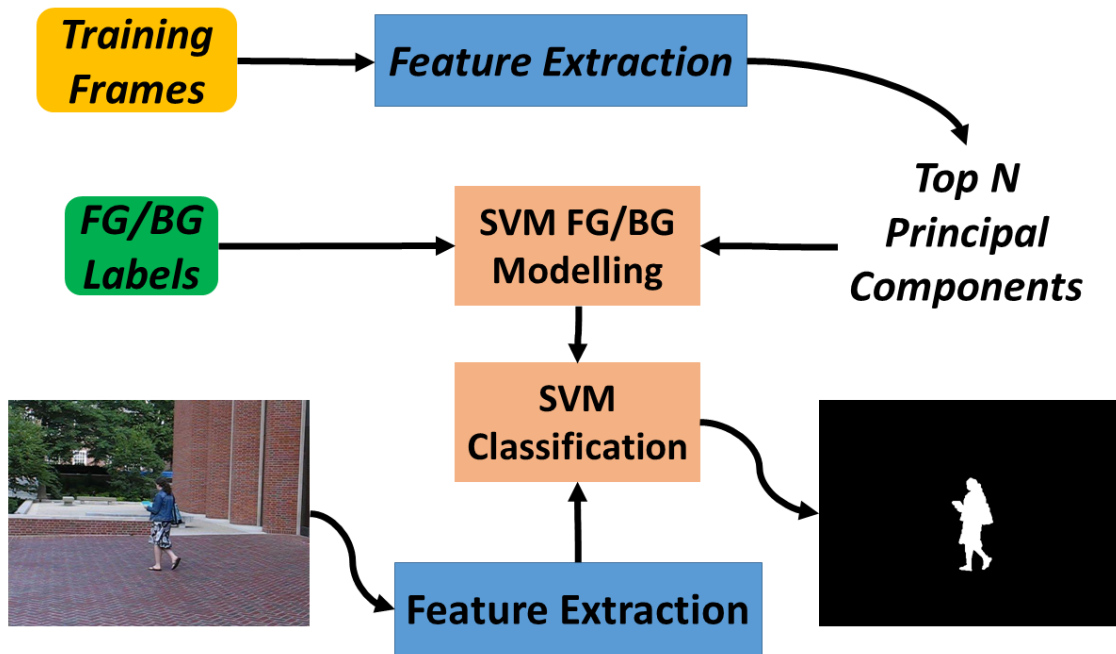


Figure 4.1 BoFs-SVM.

4.2. Feature Extraction

This component is responsible to extract features and perform PCA to select the most significant principal components. For each pixel, a number of image features are extracted from the neighborhood of size $v \times v$ pixels. We use four types of appearance-based features: color, intensity, gradient and texture.

A color feature vector of size $v \times v \times 3$ is formed by concatenating three color channels: R, G and B. Likewise, intensity is the grayscale values of $v \times v$ neighborhood resulting in a feature vector of size $v \times v$.

Texture features are based on Local Binary Patterns (LBP). In LBP, the center pixel is compared with its eight neighbors and a label of '1' is assigned if the center pixel is greater than the neighboring pixel and '0' otherwise. This results in an 8-bit binary pattern. Histogram of these patterns is then calculated for all the pixels in the neighborhood, which is then used as a texture descriptor. Multiple improvements and variations such as reduction in feature vector length, invariance to change of scale and rotation as well as robustness against noise and illumination changes have been reported in literature [71, 72]. We use the scale and rotation invariant uniform LBPs introduced in [73] and its implementation in [74]. This particular choice allows us to meet real time requirements while incorporating texture information into our algorithm. Based on a recent comparison of different LBP schemes in [72], the chosen LBP implementation has significant computational advantage over its counterparts.

A gradient feature vector is formed by first calculating spatial gradients in both x and y directions using the Sobel operator. The magnitudes and directions of the gradient vectors are then used to form feature vectors of size $v \times v \times 2$ at each pixel location.

The choice of window parameter v is critical since a smaller neighborhood can generalize well to different background scenes but can easily lead to frivolous matches. A larger neighborhood can capture unique features for specific background regions but become useless when those regions are no longer in the field of view. We select a 7×7 neighborhood that represents an empirically-optimal compromise for the sequences we have tested.

To find the optimal set of features, the extracted features undergo PCA and we retain the top components. The number of components is based on the percentage of variance explained by the selected subset of principal components, which must exceed a pre-defined *var_threshold* parameter. This rich ensemble of selected features are then combined together to form a Bag-of-Features (BoFs). Our use of BoFs capture only the local contexts which, unlike global spatial contexts, are invariant to small to medium camera and object movement. Nonetheless, local context would fail in the case of drastic scene changes such as going from outdoor to indoor.

4.3. Model Formation

In this step, feature vectors encoded into Bag-of-Features (BOFS) along with their FG/BG labels are used to train a single SVM classifier with 5-fold cross validation. The choice of SVM is deliberate as the high dimensional feature vectors from different image attributes

and large patch size can easily lead to over-fitting problems. SVM classifiers provide automatic safeguard against over-fitting.

The training data comprises of feature vectors of input images and corresponding FG/BG label for each pixel. To automatically generate FG/BG labels, the motion-based BS algorithm proposed in chapter 4 is employed. Apart from the aforementioned unsupervised mode, the proposed method can also take advantage of supervised mode i.e. manually label images yielding more accurate results.

These feature vectors and labels are then used to train a single SVM. Using LIBSVM [75], the problem is formulated as a two-class soft-margin Support Vector Classification with regularization parameter C . The kernel is set to be the radial basis function (RBF) defined as follows:

$$K(x_i, x_j) = e^{-\gamma \|x_i - x_j\|^2}$$

The setting of parameters C and shape parameter γ is based on the combination that yields the best overall performance with a 5-fold cross-validation over training data. During each iteration of the cross validation process, 10% of data is randomly retained for validation purpose, whereas remaining data is used to train the SVM model. For further details of SVM we refer readers to [75].

4.4. Classification

For any given image, we extract feature vectors as described in section 4.2 and pass onto the SVM model. The model returns FG and BG probability estimates for each pixel, which are then classified into FG/BG as follows:

$$M(x, y) = \begin{cases} BG & \text{if } Pr(x, y) \geq th \\ FG & \text{if } Pr(x, y) < th \end{cases}$$

where $M(x, y)$ represents the binary FG/BG decision of the pixel at location (x, y) of the input image, $Pr(x, y)$ is the BG probability estimate returned by the SVM classifier, and th is an empirically-determined threshold parameter. A 7×7 median filter is applied to the binary mask to remove isolated FG pixels.

4.5. Evaluation on CDnet 2014 Dataset

In this section, we compare BoFs-SVM with state of the art algorithms on CDnet 2014 dataset pan-tilt-zoom (PTZ) and baseline (BL) categories. Our goal is to demonstrate the advantages of our algorithm on PTZ sequences over the state-of-the-art from the CDnet comparison website. As static cameras are special case of moving cameras, we use the BL sequences to show that our algorithm is comparable to these algorithms as well. The dataset, parameter setting and quantitative evaluation are detailed below:

4.5.1. Dataset and Evaluation Metrics

The CDnet 2014 dataset [13] is one of the most comprehensive datasets available for evaluating BS algorithms. Table 4.1 details the test sequences for both PTZ and BL categories. The dataset specifies training and testing data to ensure consistency when comparing different algorithms.

4.5.2. Parameter Selection

In order to systematically find the optimal set of parameters, exhaustive search is conducted during the offline training phase to identify the optimal C , th , $var_threshold$ and γ . The

set of parameters that yields the best result in terms of F-Measure over the validation data is chosen. This process has resulted in a single set of parameters to be used for all PTZ sequences in CDnet2014 dataset: $C = 1, th = 0.97, var_threshold = 90\%$ and $\gamma = 0.25$.

Table 4.1 CDnet 2014 PTZ and BL test sequence details.

Test Sequence	Image Resolution	Training Data (Frame #s)	Testing Data (Frame #s)
PTZ-continuousPan	704x480	1-599	600-1700
PTZ-intermittentPan	560x368	1-1199	1200-3500
PTZ-twoPositionPTZ	570x340	1-799	800-2300
PTZ-zoomInZoomOut	320x240	1- 499	500-1130
BL-highway	320x240	1-469	470-1700
BL-office	360x240	1-569	570-2050
BL-pedestrians	360x240	1-299	300-1099
BL-PETS2006	720x576	1-299	300-1200

4.5.3. Quantitative Comparison

For quantitative evaluation we consider the top 4 methods reported on the CDnet website in PTZ category: EFIC [68], PAWCS [22], MBS [69] and SharedModel [21]. Table 4.2 to Table 4.5 contain the results of BoFs-SVM and the top 4 algorithms on four test sequences, whereas the overall results are presented in Table 4.6. It is important to note that the results reported in this paper are official results computed by the CDnet administrator based on our submission of the binary masks. The ground-truth used in the evaluation are withheld by the administrator and unavailable to us. For evaluation purposes, CDnet recommends

seven evaluation metrics: Recall (Re), Specificity (Sp), False Positive Rate (FPR), False Negative Rate (FNR), Percentage of Wrong Classifications (PWC), Precision (Pr) and F-Measure (FM), which are defined in section 3.5. An additional metric called average rank R is also defined to aggregate all seven metrics together, which is simply the average of each metric from all 4 test sequences in one category.

Our algorithm ranks first for both ZoomInZoomOut and twoPositionPTZCam test sequences, second for intermittentPan test sequence, and third for continuousPan test sequence. The results in intermittentPan and continuousPan test sequences is affected by the presence of parked cars in the scene. The lack of spatial constraint and the similarity in features of the parked cars and moving FG cars decreases the segmentation accuracy.

Table 4.2 Results for continuousPan test sequence.

Method	Re	Sp	FPR	FNR	PWC	FM	Pr	Rank (R)
BoFs-SVM [76]	0.4551	0.9992	0.0007	0.5448	0.3762	0.5756	0.7829	2.71
EFIC [68]	0.6880	0.9984	0.0015	0.3119	0.3298	0.7005	0.7134	2.14
PAWCS [22]	0.7664	0.9811	0.0188	0.2335	2.0014	0.3004	0.1868	3.14
MBS [69]	0.5168	0.9990	0.0009	0.4831	0.3661	0.6128	0.7525	2.57
SharedModel [21]	0.6814	0.9674	0.0325	0.3185	3.4128	0.1829	0.1056	4.42

Table 4.3 Results for intermittentPan test sequence.

Method	Re	Sp	FPR	FNR	PWC	FM	Pr	Rank (R)
BoFs-SVM [76]	0.5633	0.9988	0.0011	0.4366	0.5275	0.6669	0.8172	2.57
EFIC [68]	0.9070	0.9998	0.0001	0.0929	0.1039	0.9424	0.9806	1
PAWCS [22]	0.4504	0.9980	0.0019	0.5495	0.7044	0.5452	0.6907	3.57
MBS [69]	0.7072	0.9914	0.0085	0.2927	1.1258	0.5409	0.4379	3.71
SharedModel [21]	0.7649	0.9840	0.0159	0.2350	1.7961	0.4440	0.3128	4.14

Table 4.4 Results for twoPositionPTZCam test sequence.

Method	Re	Sp	FPR	FNR	PWC	FM	Pr	Rank (R)
BoFs-SVM [76]	0.8334	0.9985	0.0014	0.1665	0.3350	0.8532	0.8740	2.28
EFIC [68]	0.9191	0.9965	0.0034	0.0808	0.4353	0.8315	0.7581	3.14
PAWCS [22]	0.7414	0.9991	0.0008	0.2585	0.3887	0.8167	0.9091	2.71
MBS [69]	0.8425	0.9967	0.0032	0.1574	0.5050	0.7959	0.7541	4.14
SharedModel [21]	0.8770	0.9971	0.0028	0.1229	0.4288	0.8270	0.7823	2.71

Table 4.5 Results for zoomInZoomOut test sequence.

Method	Re	Sp	FPR	FNR	PWC	FM	Pr	Rank (R)
BoFs-SVM [76]	0.9362	0.9998	0.0001	0.0637	0.0298	0.9207	0.9058	1.28
EFIC [68]	0.9601	0.5841	0.4158	0.0398	41.520	0.0084	0.0042	3.85
PAWCS [22]	0.8322	0.9865	0.0134	0.1677	1.3700	0.1835	0.1031	3.28
MBS [69]	0.3226	0.9978	0.0021	0.6773	0.3427	0.2583	0.2153	2.85
SharedModel [21]	0.8644	0.9679	0.0320	0.1355	3.2286	0.0901	0.0475	3.71

Table 4.6 Overall comparison on CDnet 2014 PTZ Category.

Method	Re	Sp	FPR	FNR	PWC	FM	Pr	Rank (R)
BoFs-SVM [76]	0.6970	0.9991	0.0008	0.3029	0.3171	0.7541	0.8450	1.85
EFIC [68]	0.8686	0.8947	0.1052	0.1313	10.597	0.6207	0.6143	3
PAWCS [22]	0.6976	0.9912	0.0087	0.3023	1.1161	0.4615	0.4724	3.28
MBS [69]	0.5973	0.9962	0.0037	0.4026	0.5849	0.5519	0.5400	3.14
SharedModel [21]	0.7969	0.9791	0.0208	0.2030	2.2166	0.3860	0.3121	3.71

The overall comparison in Table 4.6 clearly indicates the superiority of our proposed algorithm: it ranks first in five out of seven metrics and yields comparable results

in the Re and FNR metrics. In terms of the F-Measure, we have achieved 13.34% improvement over previous best score of 62.07%. Most of the aforementioned methods fail because of underlying static camera assumption and spatial constraint, whereas our algorithm does not impose any spatial constraint and therefore produces significantly more accurate results. Figure 4.2 and Figure 4.3 and show qualitative results of BoFs-SVM for PTZ test sequences. Note the challenging nature and large camera motion in these test sequences.

Table 4.7 Overall comparison on CDnet 2014 Baseline Category.

Method	Re	Sp	FPR	FNR	PWC	FM	Pr	Rank (R)
BoFs-SVM [76]	0.9115	0.9978	0.0020	0.0884	0.4606	0.6210	0.9308	4.14
EFIC [68]	0.9455	0.9970	0.0030	0.0545	0.5201	0.9309	0.9170	3.85
PAWCS [22]	0.9408	0.9980	0.0020	0.0592	0.4491	0.9397	0.9394	2.71
MBS [69]	0.9158	0.9979	0.0021	0.0842	0.4361	0.9287	0.9431	3.28
SharedModel [21]	0.9545	0.9982	0.0018	0.0455	0.3344	0.9522	0.9502	1

Table 4.7 details the overall results on baseline category of CDnet dataset. Although our algorithm does not outperform other methods. It produces comparable results across all seven metrics. Specifically, our algorithm produces a FM of 0.92 in comparison to top performing method with FM of 0.95.

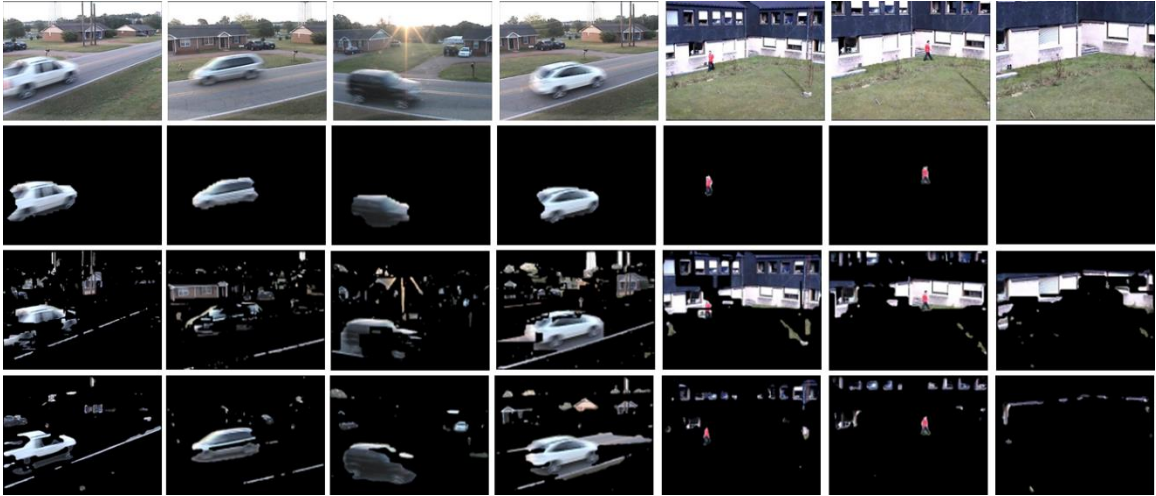


Figure 4.2 Input Image (row 1), BoFs-SVM output (row 2), EFIC output(row 3), and subSENSE output(row 4). CDnet 2014 dataset: continuousPan test sequence(columns 1-4) and zoomInZoomOut(columns 5-7) test sequences.

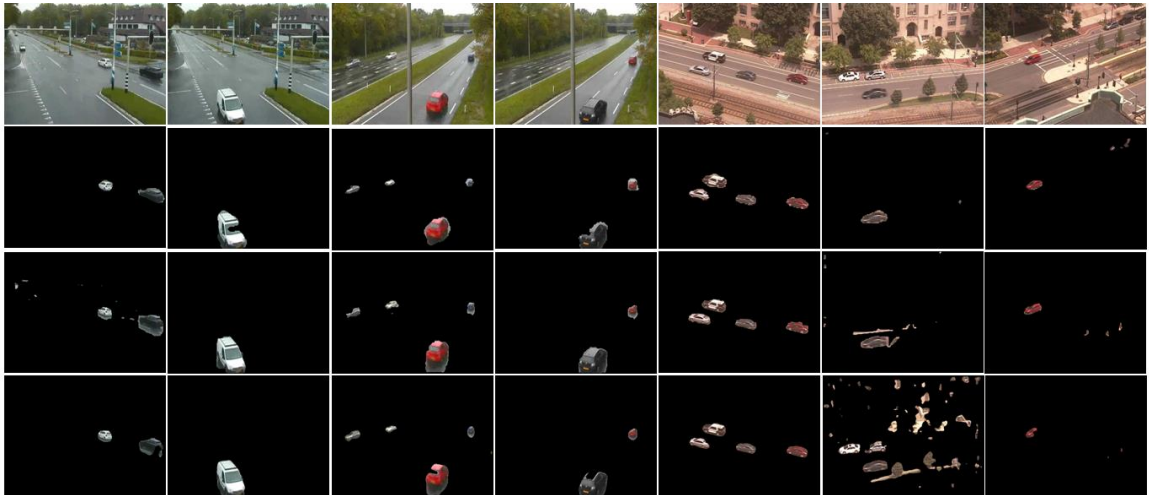


Figure 4.3 Input Image (row 1), BoFs-SVM output (row 2), EFIC output (row 3), and subSENSE output (row 4). CDnet 2014 dataset – twoPositionPTZCam (columns 1-4) and intermittentPan (columns 5-7) test sequences.

In order to highlight the importance of appropriate feature selection, we also tested our algorithm with color, LBP and gradient features individually. The results are detailed in Table 4.8.

Table 4.8 Overall Results with different features on Cdnet 2014 PTZ category.

Method	Pr	Re	FM
BoFs-SVM-Gradient	0.215	0.356	0.268
BoFs-SVM-LBP	0.642	0.421	0.507
BoFs-SVM-Color	0.839	0.623	0.69

Clearly, color features offer significantly more robust solution than LBP and gradient feature vectors. The results for texture (LBP) and color features are affected by two test sequences in PTZ category: continuousPan and intermittentPan. Due to lack of spatial constraint, the proposed method is unable to distinguish moving cars in the foreground and parked cars in the background. The same texture of parked and moving cars results in poor performance when LBP features are chosen, however color features produce more accurate results since the color of moving and parked cars are not necessarily the same. This stresses the need to choose correct type of features depending on the type of the scene.

Lastly, we compare the processing time of our algorithm with other methods in Table 4.9. The processing time for other methods are reported from official CDnet dataset website [7]. Our MATLAB implementation of the proposed method is able to achieve 15 frames per second (fps). With code optimization and C++ implementation, the proposed method is expected to achieve 30 fps.

Table 4.9 Processing time comparison.

Method	Implementation	Resolution	fps
BoFs-SVM [76]	Matlab	320×240	15
EFIC [68]	C++	320×240	16
PAWCS [22]	C++	320×240	27
MBS [69]	Matlab	320×240	9
SharedModel [21]	C++	320×240	35

4.6. Evaluation on Hopkins155 Dataset

In this section, we compare BoFs-SVM with four state of the art algorithms on Hopkins155 dataset. The dataset, parameter setting and quantitative comparison are detailed below.

4.6.1. Dataset and Evaluation Metrics

The reason to include this particular dataset is that these test sequences are taken from hand-held cameras and often used in motion segmentation literature. Although, Hopkins155 dataset comprises of 26 video sequences, most literature focuses on three test sequences: cars1, people1 and people2, since majority of test sequences have large number of frames with zero motion and despite multiple moving objects, only one or few are labeled for evaluation.

Hopkins155 dataset has no FG-free or static frames for training purposes. Therefore similar to other algorithms, we employ first two frames of the sequence to obtain initial labels and train the SVM classifier. Precision (Pr), Recall (Re) and F-Measure (FM) are used for evaluation purposes.

4.6.2. Parameter Selection

Exhaustive search is conducted during the offline training phase to identify the optimal $C, th, var_threshold$ and γ . The set of parameters that yields the best F-Measure over the validation data is chosen.

This process has resulted in a single set of parameters to be used for all of the three test sequences in Hopkins155 dataset: $C = 1, th = 0.97, var_threshold = 90\%$ and $\gamma = 0.25$.

4.6.3. Quantitative Comparison

For quantitative evaluation, we consider 4 state of the art algorithms: Brox and Malik [48], Kwak et al. [10], Sheikh et al. [9], as well as Elqursh and Elgammal [11]. We tabulate individual as well as overall results in Table 4.10. Note that in each column of tables, red font represents the best result and blue font represents the second best.

For cars1 test sequence, our algorithm produces a FM of 0.86, which is comparable to FM of 0.88 produced by the second best method Kwak et al. [10]. The slightly poorer result is due to the limitation of the training data, which are obtained in our experiments based on motion segmentation of first two frames. For cars1, part of the FG has the same motion as the BG and this contributes to the slight loss in performance.

In people1 test sequences, our algorithm has second best recall, whereas for people2 test sequence, it achieves highest recall and second best F-Measure of 0.88. It is important to mention that our current system uses only one set of parameters to train SVM model for consistency. Table 4.11 shows that the proposed method achieves the highest overall recall

and second best F-Measure of 0.87, which is comparable to top F-Measure of 0.89. Lastly, Figure 4.4 depicts results of BoFs-SVM for the three test sequences of Hopkins155 dataset.

Table 4.10 Results for test sequences of Hopkins155 dataset.

Method	Cars1			People1			People2		
	Pr	Re	FM	Pr	Re	FM	Pr	Re	FM
BoFs-SVM [76]	0.81	0.92	0.86	0.87	0.88	0.87	0.83	0.93	0.88
Brox and malik [48]	-	-	-	0.89	0.77	0.83	0.92	0.89	0.90
Kwak et al [10]	0.92	0.84	0.88	0.95	0.93	0.94	0.85	0.89	0.87
Sheikh et al. [9]	0.63	0.99	0.77	0.78	0.63	0.70	0.73	0.83	0.77
Elqursh and Elgammal [11]	0.85	0.97	0.91	0.97	0.88	0.92	0.87	0.88	0.87

Table 4.11 Overall results on Hopkins155 dataset.

Method	Pr	Re	FM
BoFs-SVM [76]	0.837	0.910	0.872
Kwak et al - with NBP [10]	0.906	0.886	0.896
Sheikh et al. [9]	0.713	0.816	0.748
Elqursh and Elgammal [11]	0.896	0.910	0.896

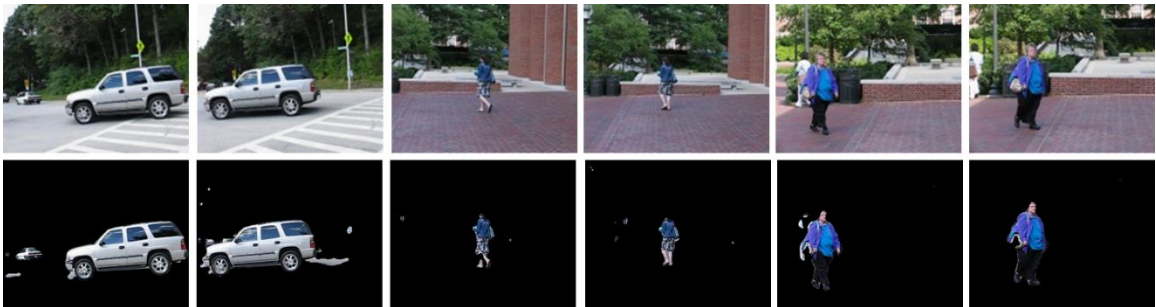


Figure 4.4 Input Image (row 1) and BoFs-SVM output(row 2). Hopkins155 dataset: Cars1(column 1-2), people1(column 3-4) and people2(column5-6).

Table 4.12 Overall Results with different features on Hopkins155 dataset.

Method	Pr	Re	FM
BoFs-SVM-Gradient	0.215	0.356	0.268
BoFs-SVM-LBP	0.642	0.421	0.507
BoFs-SVM-Color	0.839	0.623	0.69

We also tested our algorithm with color, LBP and gradient features individually. Table 4.12 details the overall performance on Hopkins155 dataset. Like CDnet 2014, color features offer the most robust segmentation. LBP and gradient feature vectors yield better results on Hopkins155 in comparison to their performance on CDnet 2014 dataset since Hopkins155 does not involve large camera movements as that of CDnet 2014 and FG objects similar to BG do not appear in the scene.

Chapter 5 Motion-based Background Subtraction

In this chapter, we present an online algorithm for foreground/background segmentation of videos captured from moving cameras. It provides algorithm overview and details the two major innovations: iterative low rank approximation of background motion and Mega-Pixel denoising. The last section provides the comparison of proposed method against state-of-the-art motion-based methods on publicly available test sequences.

5.1. Algorithm Overview

The proposed algorithm primarily relies on motion to differentiate FG/BG and uses color information only to denoise motion vectors at pixel level. Color and other appearance attributes are not used for BS to accommodate fast moving scene. It has two main modules: Motion Segmentation (MS) module and Mega-Pixel Motion Correction module (MP-MC). The MS module first performs an iterative low rank approximation of background motion, which is then compared with original motion vectors to yield initial FG/BG probability estimates. These probability estimates and the input image are then passed onto the MP-MC module. Using an innovative mega-pixel motion correction (MP-MC) process, the image is decomposed into megapixels (MP) and average FG probability for each MP is computed based on the coarse motion-based probability from MS module. These probability measures (data term) and image intensity gradient (smoothness term) are then combined together in Graph-Cut energy minimization framework to obtain the final segmentation mask. Compared with other motion-based approaches, which are feasible for offline processing and require prior information such as number of FG objects, the

proposed algorithm is online and requires no initialization or training. Figure 5.1 provides an overview of the proposed system. Detailed descriptions of each module are provided in the following sections.

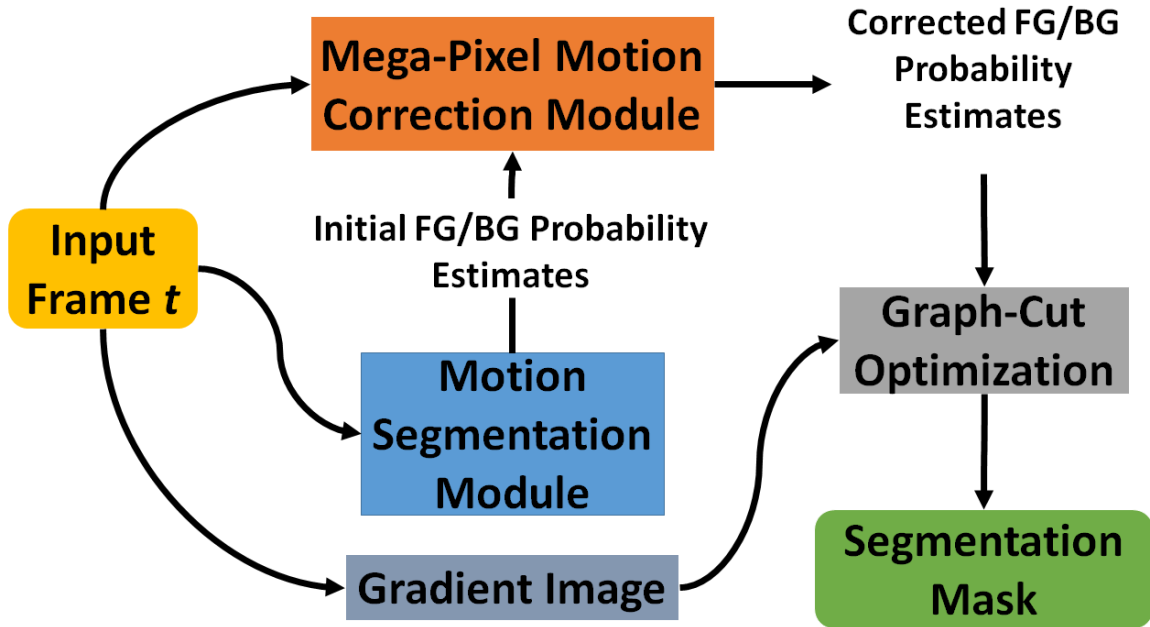


Figure 5.1 System Overview.

5.2. Motion Segmentation Module

The main task of Motion Segmentation (MS) module is to produce an initial coarse probability estimates of FG label for each pixel. It comprises of three main steps: Motion feature extraction, Iterative Polynomial Fitting and FG probability estimation. Figure 5.2 provides a detailed insight into MS module. The technical details of all the components are described in Section 5.2.1 to 5.2.2.

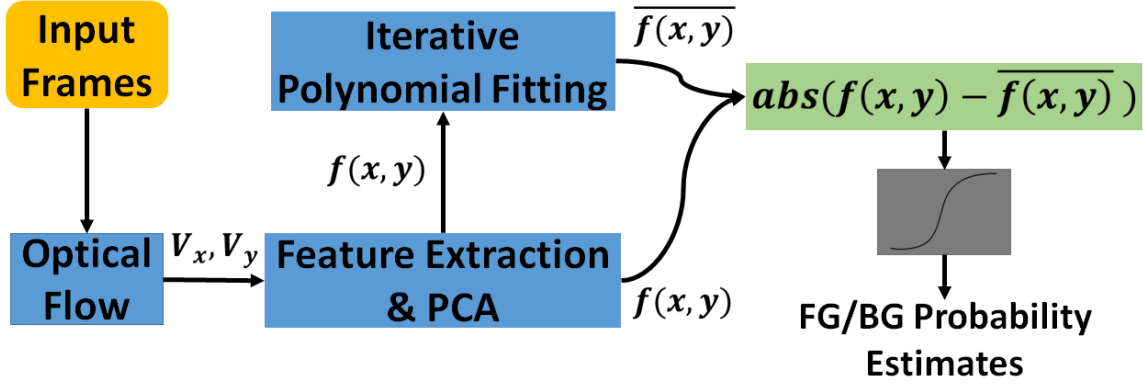


Figure 5.2 Motion Segmentation Module.

5.2.1. Motion Feature Extraction

First, optical flow [77] is used to compute a dense motion vector field between successive frames. The horizontal (V_x) and vertical (V_y) motion vectors are then used to calculate magnitude (V_{mag}) and direction (V_{ang}) as follows:

$$V_{mag} = \sqrt{V_x^2 + V_y^2} \text{ and } V_{ang} = \tan^{-1} \frac{V_y}{V_x}$$

The motion feature ($V_x, V_y, V_{mag}, V_{ang}$) extraction process is followed by Principal Component Analysis (PCA) and the top principal component is chosen for further processing. The choice of using the most significant principal component is twofold: first, empirical testing indicates that additional components do not improve performance. Second, there exist very efficient algorithm for finding the top principal component.

5.2.2. Iterative Polynomial Fitting

In this step, we perform an iterative low rank approximation of the motion features, motivated by the observations that BG pixels exhibits a smooth and spatially varying

motion, whereas FG pixels are represented by outliers. Specifically, we use the following second-order to fit the motion features after PCA:

$$\overline{f(x, y)} = ax^2 + by^2 + cxy + dx + ey + f$$

where $\overline{f(x, y)}$ represents the estimated BG motion. Polynomial coefficients (a, b, c, d, e, f) are estimated by minimizing the sum of the absolute residual E :

$$\operatorname{argmin}_{a,b,c,d,e,f} E = \sum_{\forall x,y} |f_{BG}^{(t)}(x, y) - \overline{f(x, y)}|$$

where $f_{BG}^{(t)}(x, y)$ represents the t^{th} iterated pixel motion at location (x, y) with outliers removed in each iteration.

During the first iteration, $f_{BG}^{(1)}(x, y) = f(x, y)$ where $f(x, y)$ is the original motion feature after PCA. The presence of outliers, i.e. FG, does not guarantee the best fit of the actual BG motion, and therefore we propose an iterative fit rather than a simple polynomial fitting. This is achieved by removing the set of outlier pixels from $f(x, y)$ in each iteration until the change of residual error between consecutive iterations becomes less than a small threshold ϵ . The outlier removal criteria is based on mapping the residue to probability measure and pixels with probability higher than an empirically determined constant τ_{motion} are considered as outliers.

Once converged, pixel wise motion error denoted as $e(x, y)$ is calculated between actual and estimated motion as follows:

$$e(x, y) = |f(x, y) - \overline{f(x, y)}|$$

The pixels with small error represent BG, whereas pixels with large error belong to FG. Finally, this pixel-wise motion error is passed through sigmoid activation function to estimate motion based FG probability measure denoted as p_m . The same criteria is used to calculate FG probability of each pixel during iterative $\overline{f(x,y)}$ approximation.

$$p_m(x, y) = \frac{2}{(1 + e^{-(2 * e(x,y))})} - 1$$

Figure 5.3 shows the effectiveness and increase in segmentation accuracy with increasing iteration. The red colored pixels correspond to those that are retained for the next iteration. During iteration one, clearly many pixels belonging to car, which is considered FG are still part of BG and only few pixels on car are removed as outliers. The reason is that the polynomial fit is corrupted by outliers, however when the outliers are removed in each subsequent iteration, the polynomial fit gets better and the car pixels are correctly identified as FG. Lastly, the plot indicates the decreasing error with increasing number of iterations. The algorithm converges in seven iterations on this cars1 test sequence sample frame. On average, the algorithm is able to converge within ~5 iterations.

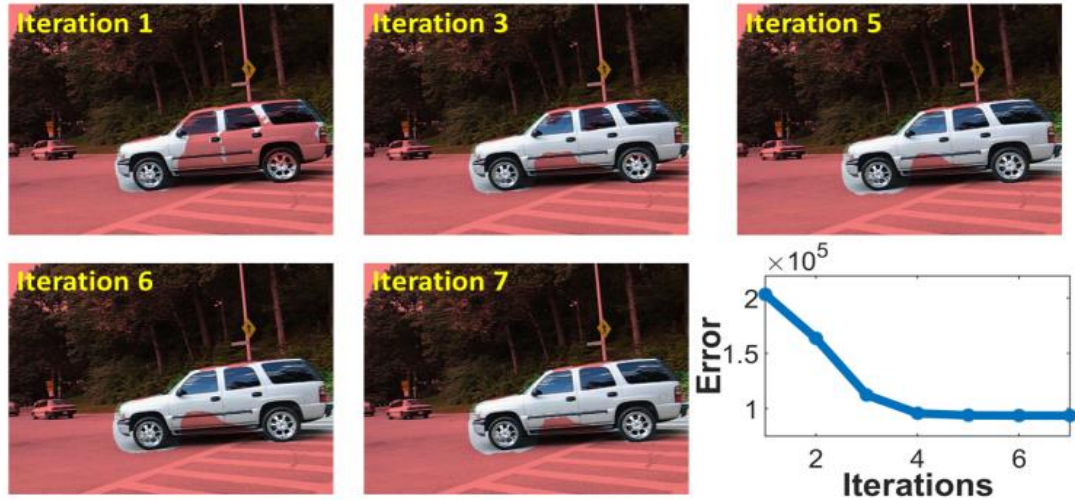


Figure 5.3 Motion segmentation accuracy and decreasing residual error with increasing number of iterations.

5.3. Mega-Pixel De-noising

The Mega-Pixel de-noising process performs spatial regulation over raw probability estimates to produce more accurate probability estimates. The final probability estimates are then calculated by taking mean of denoised probability estimates over a MP. It consists of two main steps: Mega-Pixel (MP) formation and probability de-noising. Figure 5.4 depicts the proposed MP de-noising process.

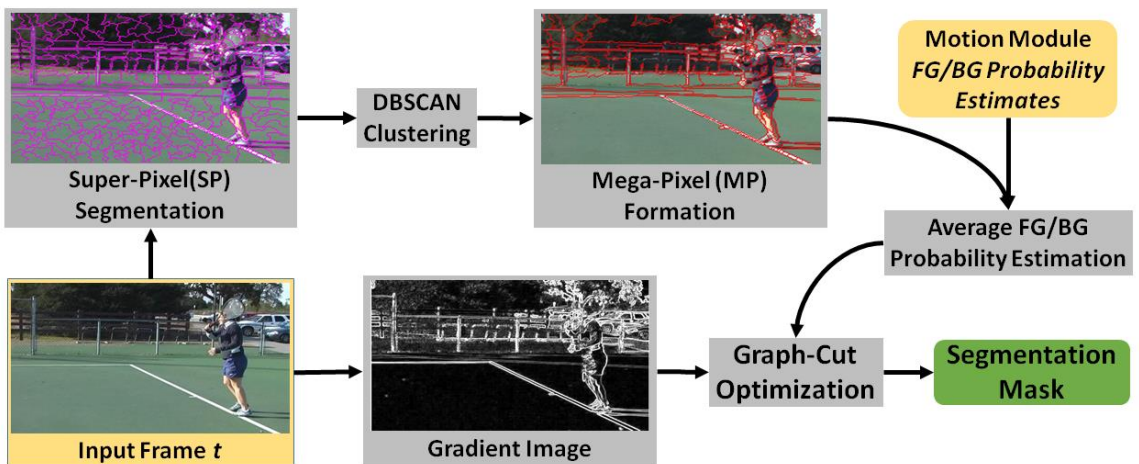


Figure 5.4 Mega-Pixel formation, Motion Correction and Graph-Cut optimization.

5.3.1. Mega-Pixel Formation

The Mega-Pixel formation process is detailed in chapter 3, section 3.4.3. Figure 5.5 depicts the resulting Super-Pixels, Mega-Pixels formed and corresponding masks. Notice the ground SPs correctly merged as a single MP.

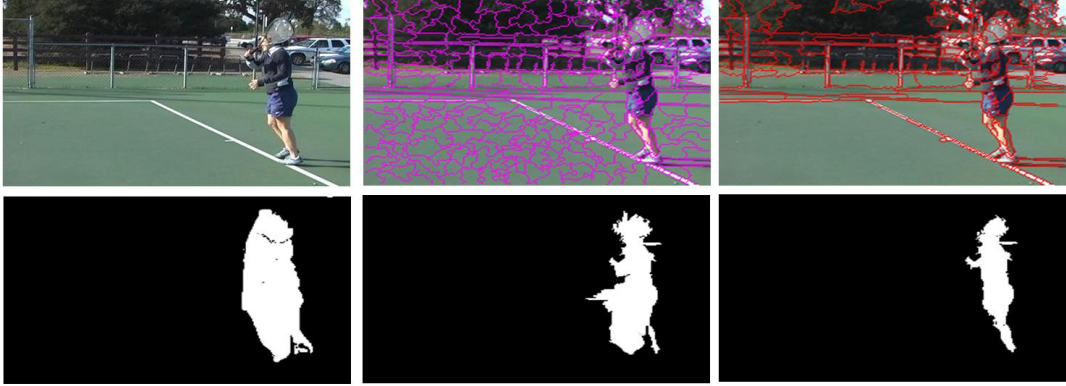


Figure 5.5 Comparison of segmentation with motion probability measure only (column 1), SP based average motion probability measure (column 2), and MP based average motion probability measure (column 3).

5.3.2. Probability De-noising

MP formation is followed by averaging motion probability estimation for each MP.

Average motion probability (\bar{p}_m) of a MP q is defined as:

$$\bar{p}_m = \frac{1}{|q|} \sum_{x \in q} p_m(x)$$

where, p_m represents the initial motion-based FG probability estimate from MS module.

The average motion (\bar{p}_m) probability is then assigned to each pixel belonging to that MP.

The use of MP and the respective \bar{p}_m can be motivated by 2 reasons. First, the pixels belonging to the same object should have same foreground/background classification, however due to limitations of the algorithms and real world non-idealities such as non-rigidity and illumination variation, the probability measure varies and results in decreased segmentation accuracy. For example, as depicted in Figure 5.5, the tennis player hands might have larger motion vectors associated with them due to swinging action, whereas torso region may have different motion vectors. Hence, the nature of motion in real world can result in high misclassification and decreased segmentation accuracy. If we consider appearance, the change in the intensity due to illumination variations results in many of the ground pixels falsely labelled as FG. To reduce such effects, we estimate average probability of a MP that represents the same object or part rather than using FG probability estimates for each individual pixel or SPs. Figure 5.5 shows segmentation masks, using probabilities from individual pixels, SP, and MP respectively. As expected, the pixel based segmentation results in too many false positives. In SP based segmentation, many falsely positive ground pixels and around the tennis player are averaged out by true negative pixels thus increasing segmentation accuracy. Lastly, when all ground SPs are merged into a MP, the dominant true positive SPs probability averages out false positive SPs probability measures, thus reducing misclassification and significantly increasing segmentation accuracy along with low computational cost.

Secondly, the SP-MP formation largely respects/preserves edges and thus noisy motion pixel probability measures along edges due to motion blur are also averaged out, thus preserving edge integrity and increasing segmentation accuracy.

5.4. Graph-Cut Optimization

In this final step, the Mega-Pixel based pixel-wise probability estimates and image intensity are formulated as an energy minimization problem. Graph-Cut implementation is based on [78, 79, 80]. The overall goal is to seek label $l \in \{BG, FG\}$ that minimizes energy:

$$E(l) = \sum_{p \in P} D_p(l_p) + \sum_{\{p,q\} \in O} V_{p,q}(l_p, l_q)$$

The first term ($D_p(l_p)$) is the data term i.e. T-link that connects each pixel to FG and BG nodes. The weight of the T-link between a pixel $p \in P$ and FG and BG nodes are Pr_p and $1 - Pr_p$ respectively. The smoothness term, $V_{p,q}(l_p, l_q)$, often known as N-link represents the relationship between adjacent pixels. For any two adjacent pixels p and q , it is defined as

$$V_{p,q}(l_p, l_q) = |l_p - l_q| \cdot e^{-(|p_R - q_R| + |p_G - q_G| + |p_B - q_B|)}$$

The data term provides an initial estimate of a pixel's tendency towards FG and BG node, whereas the smoothness term encourages same labelling to similar colored pixels. This is achieved by high penalty in case of label switch for similar color pixels, whereas low penalty for label switch in case of different colored pixels. This allows the segmentation to preserve edge integrity and further refines the segmentation results at pixel level.

5.5. Experiments and Results

We evaluate our algorithm on six challenging test sequences, comparing performance with three state-of-the-art algorithms. Five test sequences: Cars1, Cars2, People1, People2 and

Tennis are taken from Hopkins155 dataset [47] and drive test sequence from [11]. The choice of test sequences is based on the popularity of these sequences for evaluation purposes and the availability of results from previous methods. Cars1, Cars2, People1, People2 are short test sequences with a maximum of 40 frames, whereas tennis and drive test sequences are longer with 466 and 456 frames respectively. These sequences include a variety of challenges such as fast camera motion, clutter, zooming, multiple moving objects and large FG areas. Although, Hopkins155 dataset contains 26 video sequences, we believe most of other algorithms avoid using all test sequences for evaluation purposes because of following reasons:

- majority of test sequences have large number of frames with no motion,
- there are multiple moving objects but only a subset is labelled for evaluation and
- the majority of the image (>70%) is occupied by FG, making initialization difficult.

Our system has only four parameters: τ_{motion} to remove outliers during iterative BG motion approximation, ϵ to control convergence during iterative BG approximation, τ_{color} to merge SPs into MP and R to segment image into arbitrary number of SPs. We used only one set of parameters for all test sequences: $\tau_{motion} = 0.8, \tau_{color} = 7, \epsilon = 0.1$ and $R = 300$. The relatively few number of parameters indicate strength of our method and its scalability in terms of real world deployment and applicability.

Table 5.1 provides a quantitative comparison of the proposed method with three state-of-the-art motion-based algorithms: Brox and Malik [48], Sheikh et al. [9] and Narayana et al. [53]. F-Measure (FM) is used for evaluation purposes. In cars2, people1, people2, tennis and drive test sequences, our algorithm outperforms all other algorithms.

Our worst performance is in cars1 test sequence, which is primarily due to same motion of FG and BG and optical flow fails to identify any FG motion.

Table 5.1 Comparison of Proposed method with other methods. Red font is for best, whereas blue font represents second best method.

F-Measure						
Method	Drive	Cars2	People1	People2	Tennis	Cars1
Ours	0.72	0.85	0.94	0.91	0.76	0.70
Brox and malik [48]	-	-	0.83	0.90	-	-
Sheikh et al. [9]	0.04	-	0.70	0.77	0.40	0.77
Narayana et al [53]	-	0.56	0.69	0.88	0.67	0.50

Overall, unlike other methods, our algorithm is able to produce superior or comparable results across different types of challenges. Figure 5.6 depicts sample qualitative results of proposed method on different test sequences.

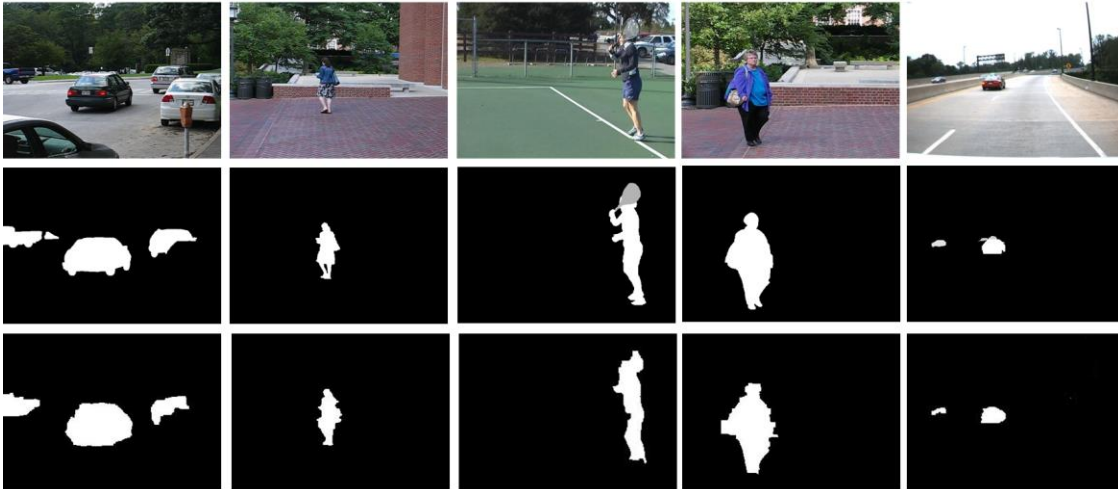


Figure 5.6 Input image (row 1), Ground truth (row 2), and proposed system output (row 3). Cars2 (column 1), people1 (column 2), tennis (column 3), people2 (column 4), drive (column 5).

Chapter 6 Hybrid Background Subtraction

In this chapter, we present a powerful hybrid algorithm for foreground/background segmentation of videos captured from PTZ, hand-held and freely moving cameras. It provides algorithm overview, details the motion and appearance modules and their fusion. The last section provides the comparison of proposed method against six state-of-the-art algorithms on publicly available test sequences.

6.1. Algorithm Overview

The proposed algorithm comprises of two main modules: Motion Segmentation Module (MS) and Appearance Module (AM). The MSM module takes the current and previous frames as input and performs an iterative low rank approximation of background motion, which is then compared with original motion vectors to yield coarse motion-based FG probability estimate for each pixel. The AM module takes the input image and two separate FG and BG Gaussian mixture models as input. Color features are extracted by sliding a window of fixed size over the entire image. The appearance models and color features are then used to compute log-likelihood ratio to generate coarse appearance-based FG probability estimate for each pixel. The coarse motion and appearance based probability estimates then undergo Mega-Pixel (MP) denoising process. The current frame is decomposed into MPs, and the probability at each pixel within a MP is replaced by the average over the whole MP. The denoised motion and appearance probability estimates are then combined together at pixel level by taking mean of both probability measures. The final FG/BG probability measures are then combined with the gradient image in the Graph-Cut energy minimization to produce the final segmentation mask.

The proposed algorithm automatically trains separate appearance-based Gaussian mixture model for FG and BG. For model initialization, motion-based probability estimates and Mega-Pixel denoising process is applied on first few frames to extract highly probable FG and BG pixels. Model parameters are estimated through an iterative expectation-maximization (EM) algorithm. During the online phase, the appearance models are continuously updated based on highly probable FG and BG pixel candidates from previous frame. In contrast to existing methods, the proposed method does not rely on explicit camera motion models nor does it make any assumptions about the scene, it is online, does not require any prior information and computationally efficient since it needs to maintain only two global models instead of pixel-wise models. Figure 6.1 provides an overview of the proposed algorithm. Each component of algorithm is detailed in following sections.

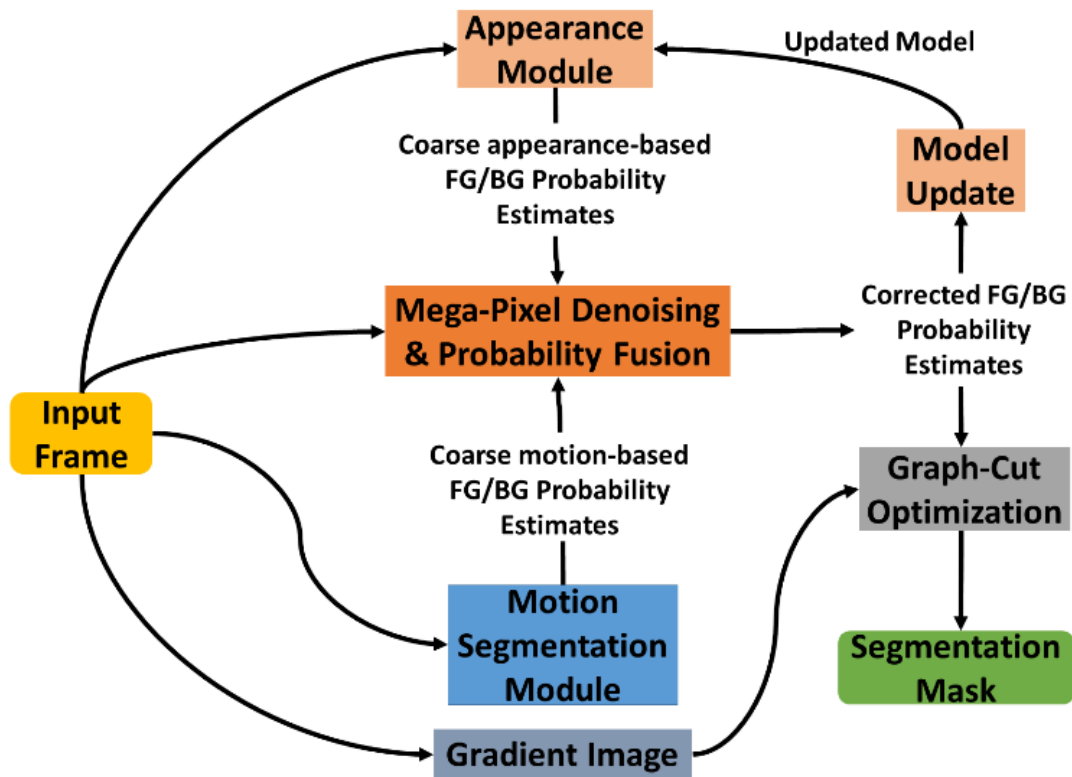


Figure 6.1 Algorithm Overview.

6.2. Motion Segmentation Module

The Motion Segmentation (MS) module takes the current and previous images as input and generates coarse motion-based probability estimates p_m . This module is same as the motion segmentation module for motion-based algorithm. We refer readers to chapter 5, section 5.2 for details.

6.3. Appearance Module

The appearance module takes the current frame as input and produces an initial FG probability estimate for each pixel. We discuss model initialization, pixel-wise probability estimation and appearance model update in following sub-sections.

6.3.1. Model Initialization and Formation

The first step is to form global Gaussian mixture models for both FG and BG. For this purpose, we use the first M frames to obtain a set of highly probable FG and BG pixels based on motion vectors. We assume that during these initial frames:

- FG and BG differ in terms of motion.
- The major motion in the scene belongs to BG and outliers are considered as FG.

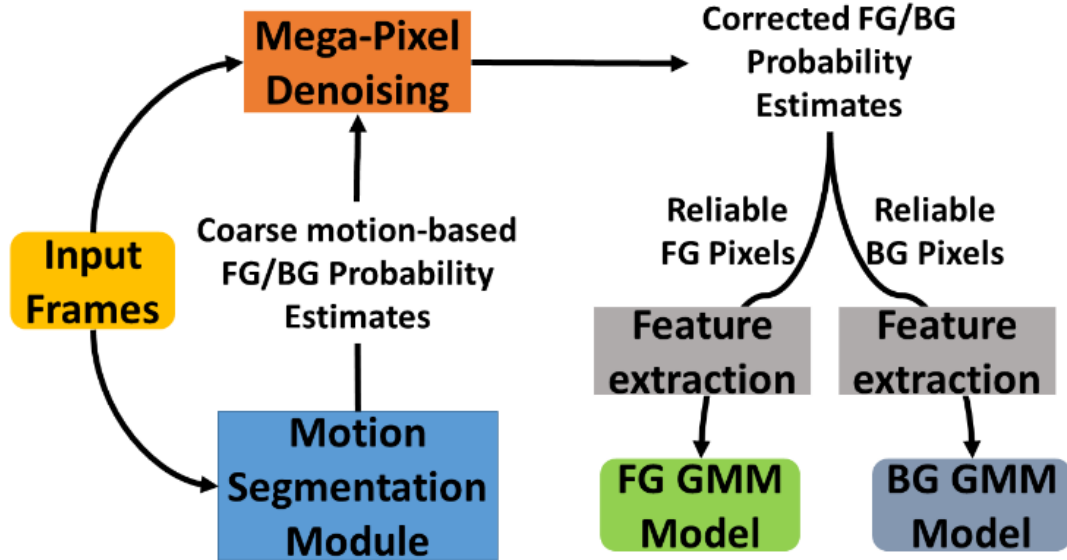


Figure 6.2 GMM appearance model initialization and formation.

Figure 6.2 depicts the overall model formation process. The current and previous frames are passed onto MSM module, which generates pixel-wise but coarse FG probability estimates. These probability estimates then undergo MP denoising (detailed in chapter 5, section 5.3), which yields more accurate FG probability estimates for each pixel. In parallel, for each pixel, a neighborhood of size $v \times v$ is considered and the three color channel: R, G and B are concatenated to form a feature vector of size $v \times v \times 3$. The feature vector of the pixels with probability measure higher than parameter p_h form the subset of reliable FG features, whereas feature vector of pixels less than parameter p_l form the subset of reliable BG features. These subset of reliable FG and BG features from all M frames are then used to build GMM for both FG and BG.

For a set of N feature vectors (x) with dimensionality $D = v \times v \times 3$, the feature vectors are modelled by a mixture of K components, defined by the probability density function:

$$p(x_j) = \sum_{k=1}^K p(k) p(x_j|k)$$

where, $p(k)$ is the prior and $p(x_j|k)$ is the conditional probability density function. The parameters are defined as:

$$p(k) = w_k$$

$$p(x_j|k) = \frac{1}{\sqrt{(2\pi)^D |\Sigma_k|}} e^{-\frac{1}{2}((x_j - \mu_k)^T \Sigma_k^{-1} (x_j - \mu_k))}$$

The parameters of GMM are prior probability (w_k), mean vector (μ_k), covariance matrix (Σ_k) and cumulated posterior probability (E_k), defined as:

$$E_k = \sum_{j=1}^N p(k|x_j)$$

using Bayes theorem,

$$p(k|x_j) = \frac{p(k)p(x_j|k)}{\sum_{i=1}^K p(i)p(x_j|i)}$$

The GMM parameters are learnt using the iterative Expectation-Maximization (EM) algorithm. As a starting point, we use k-means algorithm to estimate parameters: w_k , μ_k , Σ_k and E_k .

E-step:

$$p_{k,j}^{(t+1)} = \frac{w_k^{(t)} N(x_j; \mu_k^{(t)}, \Sigma_k^{(t)})}{\sum_{i=1}^K w_i^{(t)} N(x_j; \mu_i^{(t)}, \Sigma_i^{(t)})}$$

$$E_k^{(t+1)} = \sum_{j=1}^N p_{k,j}^{(t+1)}$$

M-step:

$$w_k^{(t+1)} = \frac{E_k^{(t+1)}}{N}$$

$$\mu_k^{(t+1)} = \frac{\sum_{j=1}^N p_{k,j}^{(t+1)} x_j}{E_k^{(t+1)}}$$

$$\Sigma_k^{(t+1)} = \frac{\sum_{j=1}^N p_{k,j}^{(t+1)} (x_j - \mu_k^{(t+1)})(x_j - \mu_k^{(t+1)})^T}{E_k^{(t+1)}}$$

The iteration stops when $\frac{L^{(t+1)}}{L^{(t)}} \leq C$, with log-likelihood defined as:

$$L = \frac{1}{N} \sum_{j=1}^N \log(p(x_j))$$

It is important to note that the FG and BG models are learnt without any spatial constraint on features. The global nature not only handles moving camera problem but also makes it computationally inexpensive in comparison to conventional methods, which maintain separate models for each pixel. The inclusion of neighborhood pixels allows us to capture local context and further increases the strength of appearance models. The choice of window parameter v is critical. A smaller neighborhood increases the chances of false positives, whereas a larger neighborhood can easily over fit by learning unique features. Based on extensive experimentation, we chose an empirically optimal neighborhood of size 3×3 .

The effectiveness of our automatic initialization procedure is demonstrated by results presented in the experiment section. Another important advantage associated with its unsupervised nature is its capability for out of the box real world deployment.

6.3.2. Pixel-wise Probability Estimation

The first step is to extract color features for each pixel in the current frame. Using FG and BG GMM models, Log-Likelihood Ratio (LLR) is calculated for each feature vector:

$$L_{FG} = \frac{1}{N} \sum_{j=1}^N \log(p_{FG}(x_j))$$

$$L_{BG} = \frac{1}{N} \sum_{j=1}^N \log(p_{BG}(x_j))$$

$$LLR = \frac{L_{FG}}{L_{BG}}$$

Finally, the LLR undergoes activation function to yield appearance-based probability p_a estimate for each pixel.

$$p_a(x, y) = \frac{2}{(1 + e^{-(2*LLR(x,y))})} - 1$$

6.3.3. Model Update

The constantly changing FG and BG pixels makes model update an integral part of the proposed algorithm. During model initialization phase, all of the training data is used and model parameters ($w_k^{(T)}, \mu_k^{(T)}, \Sigma_k^{(T)}, E_k^{(T)}$) are estimated in T EM steps until convergence.

However, for online model update, it is computationally infeasible to retain all of the previous data and then update model parameters.

Inspired by [81], we follow the incremental learning process, where we consider the previously learnt parameters and current data to update model parameters. When new data is available, another \tilde{T} EM steps are performed to update model parameters starting from previously estimated parameters $(\tilde{W}_k^{(0)}, \tilde{\mu}_k^{(0)}, \tilde{\Sigma}_k^{(0)}, \tilde{E}_k^{(0)}) = (W_k^{(T)}, \mu_k^{(T)}, \Sigma_k^{(T)}, E_k^{(T)})$. The iterative EM update stops when $\frac{L^{(t+1)}}{L^{(t)}} \leq C$.

E-step:

$$\tilde{p}_{k,j}^{(t+1)} = \frac{\tilde{w}_k^{(t)} N(\tilde{x}_j; \tilde{\mu}_k^{(t)}, \tilde{\Sigma}_k^{(t)})}{\sum_{i=1}^K \tilde{w}_i^{(t)} N(\tilde{x}_j; \tilde{\mu}_i^{(t)}, \tilde{\Sigma}_i^{(t)})}$$

$$\tilde{E}_k^{(t+1)} = \sum_{j=1}^{\tilde{N}} \tilde{p}_{k,j}^{(t+1)}$$

M-step:

$$\tilde{w}_k^{(t+1)} = \frac{\tilde{E}_k^{(0)} + \tilde{E}_k^{(t+1)}}{N + \tilde{N}}$$

$$\tilde{\mu}_k^{(t+1)} = \frac{\tilde{E}_k^{(0)} \tilde{\mu}_k^{(0)} + \sum_{j=1}^{\tilde{N}} \tilde{p}_{k,j}^{(t+1)} x_j}{\tilde{E}_k^{(0)} + \tilde{E}_k^{(t+1)}}$$

$$\begin{aligned} \tilde{\Sigma}_k^{(t+1)} = & \frac{\tilde{E}_k^{(0)} \left(\tilde{\Sigma}_k^{(0)} + (\tilde{\mu}_k^{(0)} - \tilde{\mu}_k^{(t+1)}) (\tilde{\mu}_k^{(0)} - \tilde{\mu}_k^{(t+1)})^T \right)}{\tilde{E}_k^{(0)} + \tilde{E}_k^{(t+1)}} \\ & + \frac{\sum_{j=1}^{\tilde{N}} \tilde{p}_{k,j}^{(t+1)} (\tilde{x}_j - \tilde{\mu}_k^{(t+1)}) (\tilde{x}_j - \tilde{\mu}_k^{(t+1)})^T}{\tilde{E}_k^{(0)} + \tilde{E}_k^{(t+1)}} \end{aligned}$$

6.4. Mega-Pixel Denoising and Probability Fusion

The coarse motion-based p_m and appearance-based p_a probability estimates are very noisy due to the imprecision of the motion estimation process and appearance modeling. To obtain accurate segmentation boundary, these probability estimates need to be denoised first and therefore undergo Mega-pixel denoising process. The denoising process essentially performs spatial regulation over coarse probability measures to produce more accurate probability estimates. The Mega-Pixel formation and probability denoising processes are detailed in chapter 3, section 3.4.3 and chapter 5, section 5.3. The final probability estimates are then calculated by taking mean of denoised probability estimates over a MP. Figure 6.3 depicts the proposed MP formation and denoising process.

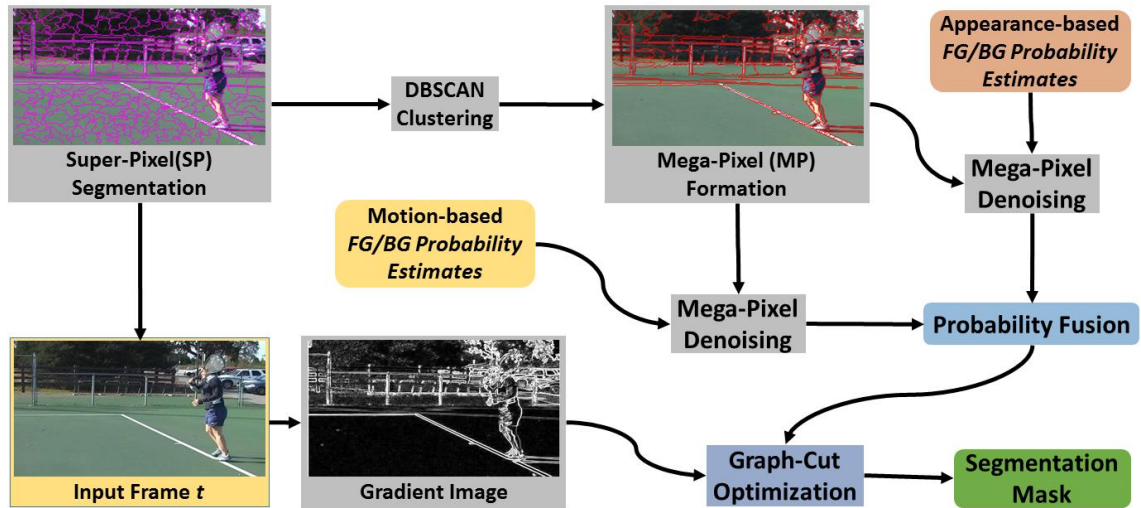


Figure 6.3 Mega-Pixel formation, Denoising and Graph-Cut optimization.

MP denoising process results in more accurate average motion and appearance probability estimates for each MP. Average motion probability (\bar{p}_m) and average appearance probability (\bar{p}_a) of a MP q are defined as:

$$\bar{p}_m = \frac{1}{|q|} \sum_{x \in q} p_m(x)$$

$$\bar{p}_a = \frac{1}{|q|} \sum_{x \in q} p_a(x)$$

where, p_m represents the coarse motion-based FG probability estimate from MS module and p_a represents the coarse appearance-based FG probability estimate from AM module. The average motion (\bar{p}_m) and appearance (\bar{p}_a) probabilities are then assigned to each pixel belonging to that MP. For each pixel, the denoised motion and appearance probabilities are fused together as follows:

$$Pr_p = 0.5 \times \bar{p}_m + 0.5 \times \bar{p}_a$$

6.5. Graph-Cut Optimization

In this final step, the Mega-Pixel based pixel-wise probability estimates and image intensity are formulated as an energy minimization problem. Graph-Cut implementation is based on [78, 79, 80]. The overall goal is to seek label $l \in \{BG, FG\}$ that minimizes energy:

$$E(l) = \sum_{p \in P} D_p(l_p) + \sum_{\{p,q\} \in O} V_{p,q}(l_p, l_q)$$

The first term ($D_p(l_p)$) is the data term i.e. T-link that connects each pixel to FG and BG nodes. The weight of the T-link between a pixel $p \in P$ and FG and BG nodes are Pr_p and $1 - Pr_p$ respectively. The smoothness term, $V_{p,q}(l_p, l_q)$, often known as N-link represents the relationship between adjacent pixels. For any two adjacent pixels p and q , it is defined as

$$V_{p,q}(l_p, l_q) = |l_p - l_q| \cdot e^{-(|p_R - q_R| + |p_G - q_G| + |p_B - q_B|)}$$

The data term provides an initial estimate of a pixel's tendency towards FG and BG node, whereas the smoothness term encourages same labelling to similar colored pixels. This is achieved by high penalty in case of label switch for similar color pixels, whereas low penalty for label switch in case of different colored pixels. This allows the segmentation to preserve edge integrity and further refines the segmentation results at pixel level.

6.6. Experiments and Results

We evaluate our algorithm on twelve challenging test sequences against six state-of-the-art algorithms. Eleven test sequences: Cars1, Cars2, Cars3, Cars4, Cars5, Cars6, Cars7, Cars8, People1, People2 and Tennis are taken from Hopkins155 dataset [47] and drive test sequence from [11].

The choice of test sequences is based on the popularity of these sequences for evaluation purposes and the availability of results from previous methods. All of the test sequences from Hopkins155 dataset except tennis test sequence are short test sequences with a maximum of 50 frames, whereas tennis and drive test sequences are longer with 466 and 456 frames respectively. These sequences include a variety of challenges such as fast camera motion, clutter, zooming, multiple moving objects and large FG areas. Although, Hopkins155 dataset contains 26 video sequences, we believe most of other algorithms avoid using all test sequences for evaluation purposes because of following reasons:

- majority of test sequences have large number of frames with no motion,

- there are multiple moving objects but only a subset is labelled for evaluation and
- the majority of the image (>70%) is occupied by FG, making initialization difficult

The proposed method comprises of a number of parameters. For fair evaluation of our method, we use only one set of parameters:

- $\tau_{motion} = 0.8$, the outlier removal threshold for iterative BG motion approximation,
- $\epsilon = 0.1$, minimum error to stop iterative BG approximation process,
- $\tau_{color} = 3$, color similarity to merge SPs into a MP,
- $R = 300$, the number of SPs an image is segmented into,
- $C = 0.01$, termination threshold for iterative EM algorithm,
- $v = 3$, size of color feature extraction window,
- $p_l = 0.5$, threshold for reliable BG pixel candidates,
- $p_h = 0.9$, threshold for reliable FG pixel candidates,
- $nbg = 9$, number of components for BG GMM,
- $nfg = 7$, number of components for FG GMM and
- $M = 5$, number of frames for appearance model initialization.

The same set of parameters for all test sequences indicate the strength of our method and its scalability in terms of real world deployment and applicability. Table 6.1 and Table 6.2 shows quantitative comparison of proposed method with six state-of-the-art algorithms: Kwak et al. [10], Sheikh et al. [9], Elqursh et al. [11], Narayana et al. [53], Zamalieva et al. [52] and Lim et al. [51]. F-measure (FM) is used for evaluation purposes. The results for [9], [51], [10], and [52] are reported from [52], whereas the results for remaining methods are obtained from original papers.

Table 6.1 Comparison of Proposed method with other methods on short test sequences. Red font is for best, whereas blue font represents second best method.

	F-Measure									
	Cars1	Cars2	Cars3	Cars4	Cars5	Cars6	Cars7	Cars8	People1	People2
Ours	0.92	0.90	0.92	0.85	0.79	0.91	0.93	0.87	0.93	0.93
Sheikh et al.[9]	0.68	0.63	0.76	0.76	0.66	0.80	0.87	0.82	0.52	0.78
Narayana et al.[53]	0.51	0.57	0.73	0.48	0.71	0.84	0.43	0.87	0.70	0.88
Zamalieva et al.[52]	0.82	0.79	0.88	0.89	0.87	0.90	0.87	0.83	0.86	0.91
Kwak et al.[10]	0.78	0.70	0.80	0.62	0.64	0.73	0.69	0.76	0.56	0.80
Lim et al.[51]	0.72	0.85	0.78	0.75	0.75	0.68	0.86	0.75	0.50	0.80
Elqursh et al.[11]	0.91	-	-	-	-	-	-	-	0.89	0.77

We first compare the results of short test sequences: cars1, cars2, cars3, cars4, cars5, cars6, cars7, cars8, people1 and people2. A characteristics of these short sequences is that the FG objects are always in motion. In these test sequences, our algorithm outperforms all other algorithms except cars4 and cars5 test sequences. In cars4 test sequence, the proposed method produces second best result of 0.85, which is comparable to top result of 0.89. For cars5 test sequence, our algorithm is placed at second position, however we observed that the result is effected by the false positives of a similar colored car parked in the scene as that of the moving FG car.

In long test sequences: tennis and drive, the proposed method outperforms other algorithms. We would like to highlight the drive test sequence which is captured by a camera mounted on a car travelling on actual highway. This test sequence is particularly challenging since cars keep entering and exiting the field of view at different points in the video and due to forward motion. The motion-based Sheikh et al. [9] entirely fails due to

underlying orthographic projection assumption failure, whereas Elqursh et al. [11] performs better since it exploits long term trajectories.

Table 6.2 Comparison of Proposed method with other methods on long test sequences. Red font is for best, whereas blue font represents second best method.

	F-Measure	
	Tennis	Drive
Ours	0.92	0.80
Sheikh et al.[9]	0.40	0.04
Narayana et al.[53]	0.68	-
Zamalieva et al.[52]	-	-
Kwak et al.[10]	-	-
Lim et al.[51]	-	-
Elqursh et al.[11]	0.89	0.70

The tennis test sequence allows us to test our algorithm for cases when there is no FG and/or BG motion. The tennis player stops to wait for the ball and sometimes moves fast to intercept the ball. Both methods that heavily rely on motion i.e. Sheikh et al. [9] and Narayana et al. [53] fail since the tennis player stops and there is no motion. Again, Elqursh et al. [11] due to their long term trajectory analysis are able to cope such cases. The continuous update and maintenance of appearance model allows us to handle such type of cases as evident from outperforming results. Overall unlike other methods our algorithm is able to produce superior results across different types of challenges.

Chapter 7 Application: In-Air Signature Recognition and Authentication

In this chapter, we detail one of the applications of the low complexity motion-based algorithm for development of a novel authentication mechanism aimed at Head Mounted Wearable Devices (HMWCs). The authentication mechanism, dataset and results are presented in following sections.

7.1. Introduction

User authentication is key to security and access control for any computer system. Broadly, user authentication can be classified into three categories based on their authentication mechanisms [15]. The first category is the knowledge-based methods that rely on passwords, passcode or gesture. The second category is token-based. As the name suggests, this category relies on a pre-assigned token such as a RFID tag or a smart card. Lastly, we have biometric-based systems which exploit physiological characteristics such as fingerprints, face and iris patterns for authentication [15]. Each of these mechanisms has its advantages and disadvantages. For example, knowledge-based methods are simple but require users to memorize password. Token-based authentication is prone to token theft. Biometric-based authentication is not prone to identity theft but are less preferred by users due to privacy concerns of being tracked.

The recent push towards wearable technology has resulted in proliferation of wearable cameras that support prolonged and high-quality recording. Head mounted cameras are particularly popular due to their ability in capturing the viewing perspective of the user. Many wearable cameras are now equipped with networking and computing

capabilities. Google Glass and Microsoft's HoloLens are perfect examples of Head Mounted Wearable Computer (HMWC) that neatly combine wearable camera, computing platform and display in creating augmented reality experience. The wearable technology is expected to have significant growth in the coming years, with applications ranging from personal use to law enforcement and healthcare to name a few.

In the context of HMWCs, the pervasiveness, size and portability of such devices make them prone to theft and hence purport the need of a robust authentication mechanism. The lack of physical interfaces such as keyboards or touch pads limits the choice of authentication mechanisms. To overcome this problem, we propose Virtual-Signature (VSig), a hand-gestured signature performed by an individual and recognized via the wearable camera. This approach combines the strength of familiar knowledge-based authentication mechanism based on a person's own signature and the ultra-portability of a HMWC without the need of a writing surface.

7.2. SIGAIR Dataset

This section details the SIGAIR dataset. Google-Glass is used as the wearable device for recording hand-gestured signatures from ten individuals and building the SIGAIR dataset. Google-Glass is a head mounted wearable device empowered with a processor, color camera, microphone, display and a touchpad as depicted in Figure 7.1.

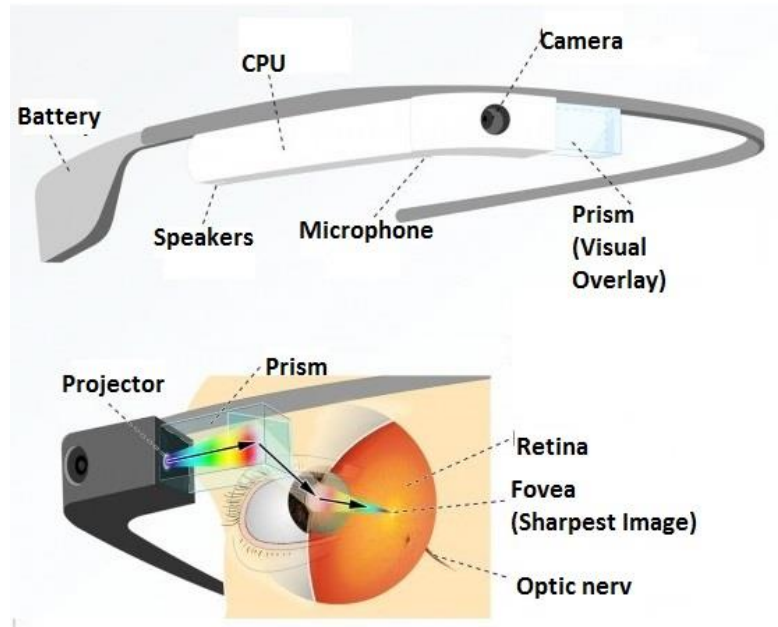


Figure 7.1 Google-Glass design and display (courtesy of Martin Missfeldt at <http://www.brille-kaufen.org/en/googleglass>)

We have collected a total of 96 hand-gestured signatures from 10 different individuals. Out of 96 hand-gestured signatures, 38 are stored for matching purposes during authentication process, whereas remaining 58 hand-gestured signatures are used for testing the proposed system. Each individual is instructed to use his/her index finger to sign in the air while wearing Google-Glass. The camera preview is displayed simultaneously on the prism display and the user is asked to ensure that the tip of the index finger is always visible in the preview. Figure 7.2 depicts example frames of an individual while doing signature in the space as captured by the color camera on Google-Glass. The reason to use fingertip instead of hand's center or any other point is that it offers a more natural analogy to using a pen. This has also been suggested in previous work [15].



Figure 7.2 Left: Signing with Google-Glass. Right: Image captured from Google-Glass.

To ensure variance in dataset and to test the effectiveness of the proposed method, virtual signatures are captured in different environmental settings and scenarios based on whether the hand signing is done in indoor or outdoor environment, background is static or dynamic, and individual himself is stationary or moving. Typical scenarios are tabulated in Table 7.1.

Table 7.1 SIGAIR Dataset Variation and Scenarios.

	Environment	Person	Background
1	Indoor	Stationary	Static
2	Indoor	Stationary	Dynamic
3	Indoor	Moving	Static
4	Indoor	Moving	Dynamic
5	Outdoor	Stationary	Static
6	Outdoor	Stationary	Dynamic
7	Outdoor	Moving	Static
8	Outdoor	Moving	Dynamic

7.3. Proposed System

In the proposed VSig system, an individual uses the index finger to perform signature in the space, which is captured through the color camera of HMWC and compared with the stored signatures for authentication. The reliance on HMWC poses a number of unique challenges in the design. First, unlike stationary camera, wearable camera is likely to be constantly moving and very little assumption can be made about the scene in the video. Traditional background segmentation algorithms, which are mostly designed for stationary cameras, cannot be used to accurately segment the hand. Other challenges include localization of the fingertip, robust algorithms to handle the variability of hand signing, and adequate visual feedback to user to stay within the field of view of the camera. The proposed system comprises of two main modules: Signature Extraction Module (SEM) and Signature Verification Module (SVM). They are described below:

7.3.1. Signature Extraction Module

As the name suggests, the SEM is responsible for extracting signatures from the video. This is achieved in a two-step process: Video Segmentation and hand & fingertip detection.

Step 1: Video Segmentation

The application of proposed method violates the static camera assumption since it is a wearable device and thus makes the segmentation a harder problem. Furthermore, a person with wearable device can be anywhere and for each scenario getting foreground free frames and constructing a background model is impractical and computationally expensive for resources constrained wearable devices. Therefore, we use the motion-based algorithm

presented in chapter 5. It neither relies on static camera assumption i.e the camera can be moving, nor does it require a construction of a background model. For details of segmentation algorithm we refer readers to chapter 5.

Step 2: Hand and Fingertip Detection

In this step, we exploit skin color as a cue for hand segmentation. Skin color has been exploited for many purposes including image segmentation, face and gesture recognition to name a few. In general, YCbCr color space is considered to be the most appropriate choice and yields accurate results for detecting pixels belonging to skin [82, 83]. These studies have shown that the typical range of Y , Cb and Cr for skin color detection are as follows:

$$Y > 80, 77 \leq Cb \leq 127, 133 \leq Cr \leq 173$$

These aforementioned ranges are then used to label each pixel as skin or non-skin. However, to ensure minimum number of False Negative (FN) skin pixels, we use a more relaxed range for Cb and Cr components as follows:

$$75 \leq Cb \leq 135, 130 \leq Cr \leq 180$$

For implementation purposes, publicly available code¹ is used. The result is a binary mask with white pixels representing skin, whereas black pixels represents non-skin pixels. Since our goal is to detect skin pixels belonging to hand only and there is possibility that there may be skin pixels because of presence of a person in a scene or any material such as wooden that falls in the same range as skin color, we combine motion and skin

¹ <http://www.mathworks.com/matlabcentral/fileexchange/28565-skin-detection>

color information together to segment out hand by taking logical AND of motion-based binary mask BM_{motion} and skin color based binary mask BM_{skin} . Figure 7.3 shows example motion based BM_{motion} and skin color based BM_{skin} , segmented hand and fingertip tracking.

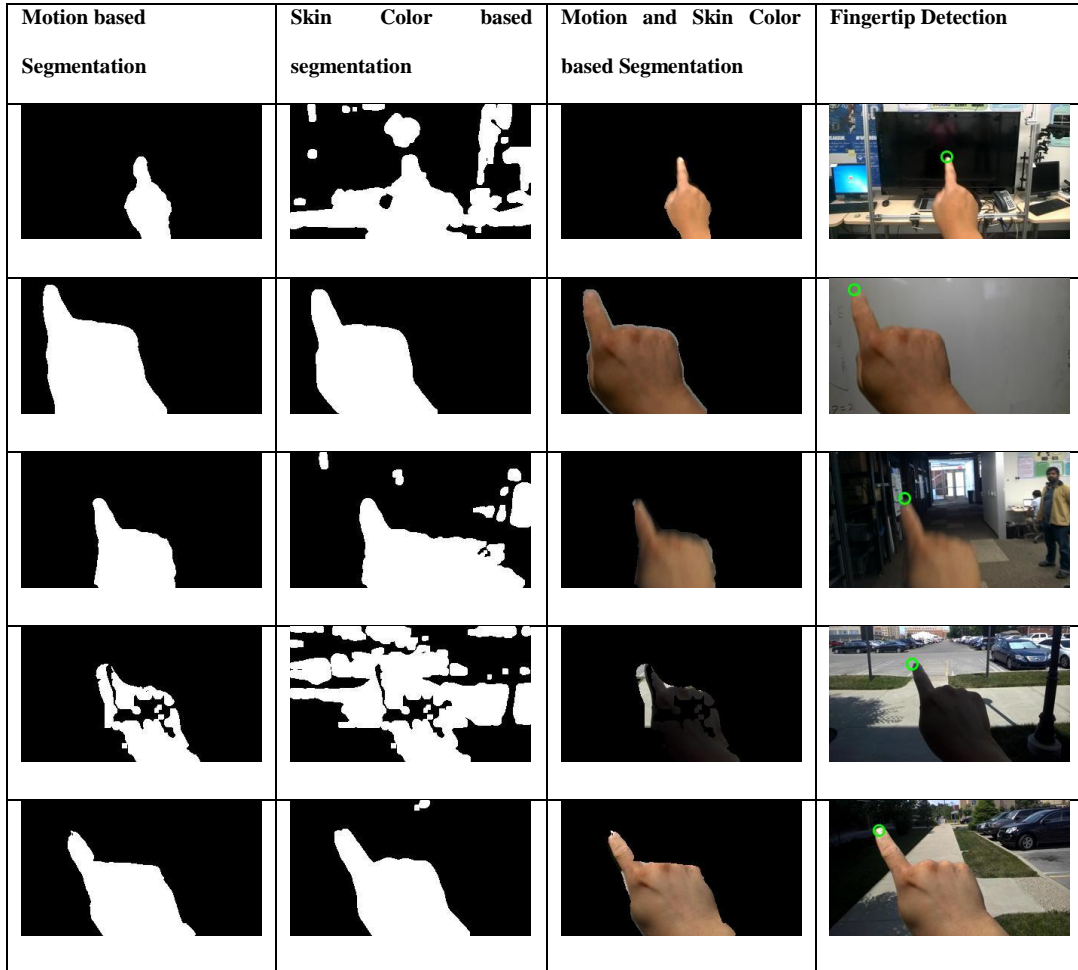


Figure 7.3 Hand Segmentation and Fingertip tracking.

Once the hand object is segmented out, the top 2D coordinates are extracted from the entire sequence as the fingertip location sequence. The 2D coordinates are post-processed by removing outliers and smoothing. A 2D coordinate is labelled as an outlier if Euclidean distance between current position and next position is greater than 50 pixels, otherwise it is a reliable fingertip detection. This step is followed by filtering using a moving average filter with a window size of 7 frames for temporal smoothing. Finally, the

2D spatial coordinates are normalized such that we have a normal Gaussian distribution $N(0, \sigma^2)$ over all 2D positions. The normalization helps to reduce the possible variations in signatures of the same person. For example, a person might sometimes sign very compactly and at other times produce an elongated or stretched out signatures.

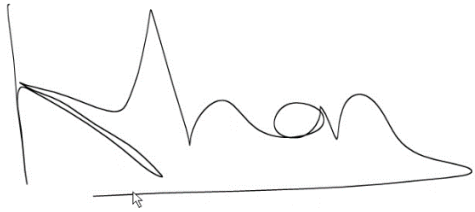







Signatures captured on a Tablet	Signatures extracted by proposed system from Air
	
	
	
	

Figure 7.4 Signatures on tablet vs Signatures extracted from space by proposed system.

Figure 7.4 shows samples of hand-gestured signatures extracted by SEM module side by side and compares them with signatures of the same individual captured on a tablet.

It is observed that there are strong resemblance between the two types of signatures and our proposed algorithm can recognize the signature trace with little distortion.

7.3.2. Signature Verification Module

There are two key requirements for signature matching. First, to enroll or register oneself with the device it should require minimal number of samples. Second, it should be able to take into account the variation in signatures of the same person both spatially and temporally. The spatial variation to a large extent is countered by normalization of spatial coordinates, whereas we propose to use Dynamic Time Warping (DTW) to overcome temporal variations. Another benefit of DTW is that there is no need to collect a large number of signatures from each person to build an exclusive model for each individual. DTW provides similarity measure between two temporal signals varying over time in terms of distance. In our experiments, we choose Euclidean distance as similarity measure. Given two 2D signatures represented by their features as $S_a(t_a)$, where $t_a = 1, 2, \dots, n_1$ and $S_b(t_b)$, where $t_b = 1, 2, \dots, n_2$, we construct a distance matrix D of size $n_1 \times n_2$ such that each of its element $d_{t_a t_b}$ is calculated as:

$$d_{t_a t_b} = \|S_a(t_a) - S_b(t_b)\|$$

The DTW algorithm finds a path between d_{11} and $d_{n_1 n_2}$ in a non-decreasing fashion such that the total sum of elements along this path is minimal. This minimum distance is the DTW distance between two 2D signatures and denoted as $d(S_a, S_b)$.

For signature matching and recognition, $d(S_a, S_b)$ is calculated between input signature against all of the existing signatures in the database or on the wearable device

itself. The signature is matched to the one with minimum distance. If distance is beyond a certain threshold the user is asked to sign again because of poor quality.

SIGAIR Dataset comprising of a total of 96 signatures was used to test the proposed method. The dataset had a total of 58 signatures from 10 individuals for testing purposes. The average accuracy for all 10 individuals achieved by our system is 97.5%, which is comparable to any existing approaches. Apart from reporting the accuracy, we also analyzed inter and intra person signatures DTW distances to demonstrate the feasibility of proposed method for large scale deployment and use.

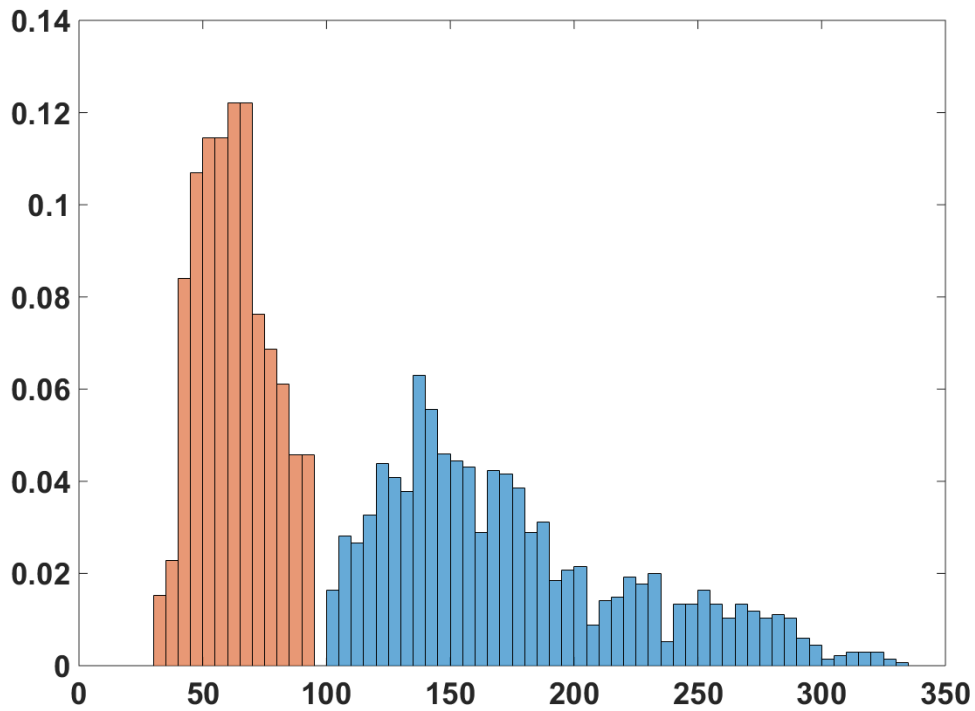


Figure 7.5 Normalized Intra(orange color) and Inter(blue color) Person DTW distance histogram.

Intra person signature is the DTW distances between signatures of the same person. Figure 6 depicts the histogram of intra person signature for all 10 individuals and they are below DTW distance of 85 with peak at 60. On the other hand, if we analyze the histogram of inter person signature DTW distances for all of 10 individuals, the histogram peaks at 140 and has no overlap with intra DTW distance histogram. The inter and intra DTW distance segregation suggests that the proposed method could be scalable to a large dataset.

Chapter 8 Conclusion

The increasing computational platforms equipped with powerful processors and cameras has resulted in exponential increase in videos from moving cameras. The isolation of interesting objects in a scene is one of the pre-processing requirements for many vision applications. The focus of existing algorithms on static camera has created void for processing videos from moving cameras. To fill this void, we have presented three background subtraction algorithms: model-based, motion-based and hybrid.

The model-based algorithm extracts multiple appearance features by sliding a fixed size window over the entire image. A global FG/BG SVM model is then learnt without any spatial constraint. The choice of features and lack of spatial constraint makes our algorithm robust against moving BG. This is demonstrated by results on CDnet 2014 dataset with 13.04% improvement in terms of F-Measure over second best method. The model-based algorithm for scenarios with finite set of scenes such as PTZ.

The motion-based algorithm introduces an innovative motion segmentation scheme based on low rank BG motion approximation and MP based motion correction. Unlike other methods, it does not need to maintain/update the BG model, is computationally inexpensive, has few parameters and operates in an online fashion. The effectiveness of proposed method is demonstrated by evaluation on Hopkins155 dataset and more importantly its application and accuracy in hand segmentation for the hand-gestured signature recognition and authentication system developed for Google-Glass device. This

method is low complexity and therefore ideal for wearable devices. However it would fail if FG and BG have same motion or both are stationary

The hybrid approach combines the innovations of motion and appearance algorithms. The motion module comprises of an innovative motion segmentation scheme based on low rank BG motion approximation. The appearance module models the FG and BG appearance as two separate Gaussian mixture models, which are then used for incoming frame to generate appearance-based probability measure. Inspired by model-based method, color features are extracted by considering neighborhood instead of individual pixel values. The motion and appearance based probability estimates undergo Mega-Pixel based spatial denoising process and are fused together. The combined probability estimate and gradient image under graph-cut optimization to produce segmentation mask. Unlike other methods, the proposed method can automatically identify correct number of FG objects, it is online, does not require special initialization procedure and it is computationally inexpensive since it maintains only global models for FG and BG. Evaluation on challenging test sequences and comparison with six state-of-the-art algorithms demonstrates its superiority and real world applicability. The hybrid method is universal in nature since it can handle all type of scenarios but computationally more expensive than motion-based method. Currently, the proposed method gives equal weightage to both motion and appearance based probability measures, however dynamic weight assignment is under investigation as a part of future work

The second major contribution aimed at application of low complexity motion-based algorithm to wearable devices. It has resulted in the development of a novel virtual signature based authentication mechanism. Unlike other approaches, the proposed method

does not rely on additional hardware or sensors and depends only on the built-in color camera. The novel motion and skin based segmentation algorithm is successfully applied for hand segmentation and fingertip tracking to reconstruct signatures from space. The extracted signatures are then compared with pre-stored signatures using DTW. The proposed method offers convenient enrollment and achieves 97.5% accuracy.

As a part of future work, we will provide the person with real time visual feedback of the signature in the space. In addition, the SIGAIR Dataset will be expanded in size with increased complexity and our algorithms will be tested using fake or forged signature attacks.

Bibliography

- [1] T. Bouwmans, “Recent advanced statistical background modeling for foreground detection-a systematic survey,” *Recent Patents on Computer Science*, vol. 4, no. 3, pp. 147–176, 2011.
- [2] E. Hayman and J.-O. Eklundh, “Statistical background subtraction for a mobile observer,” in *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*. IEEE, 2003, pp. 67–74.
- [3] M. Irani, B. Rousso, and S. Peleg, “Computing occluding and transparent motions,” *International Journal of Computer Vision*, vol. 12, no. 1, pp. 5–16, 1994.
- [4] C. Yuan, G. Medioni, J. Kang, and I. Cohen, “Detecting motion regions in the presence of a strong parallax from a moving camera by multiview geometric constraints,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 29, no. 9, pp. 1627–1641, 2007.
- [5] H. S. Sawhney, Y. Guo, J. Asmuth, and R. Kumar, “Independent motion detection in 3d scenes,” in *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, vol. 1. IEEE, 1999, pp. 612–619.
- [6] J. Y. Wang and E. H. Adelson, “Representing moving images with layers,” *Image Processing, IEEE Transactions on*, vol. 3, no. 5, pp. 625–638, 1994.
- [7] M. J. Black and P. Anandan, “The robust estimation of multiple motions: Parametric and piecewise-smooth flow fields,” *Computer vision and image understanding*, vol. 63, no. 1, pp. 75–104, 1996.

- [8] Y. Weiss, “Smoothness in layers: Motion segmentation using nonparametric mixture estimation,” in *Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on*. IEEE, 1997, pp. 520–526.
- [9] Y. Sheikh, O. Javed, and T. Kanade, “Background subtraction for freely moving cameras,” in *Computer Vision, 2009 IEEE 12th International Conference on*. IEEE, 2009, pp. 1219–1225.
- [10] S. Kwak, T. Lim, W. Nam, B. Han, and J. H. Han, “Generalized background subtraction based on hybrid inference by belief propagation and bayesian filtering,” in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 2174–2181.
- [11] A. Elqursh and A. Elgammal, “Online moving camera background subtraction,” in *Computer Vision–ECCV 2012*. Springer, 2012, pp. 228–241.
- [12] P.-L. St-Charles, G.-A. Bilodeau, and R. Bergevin, “Subsense: A universal change detection method with local adaptive sensitivity,” *Image Processing, IEEE Transactions on*, vol. 24, no. 1, pp. 359–373, 2014.
- [13] Y. Wang, P.-M. Jodoin, F. Porikli, J. Konrad, Y. Benezeth, and P. Ishwar, “Cdnet 2014: An expanded change detection benchmark dataset,” in *Computer Vision and Pattern Recognition Workshops (CVPRW), 2014 IEEE Conference on*. IEEE, 2014, pp. 393–400.
- [14] H. Sajid and S.-c. S. Cheung, “Vsig: Hand-gestured signature recognition and authentication with wearable camera,” in *Information Forensics and Security (WIFS), 2015 IEEE International Workshop on*. IEEE, 2015, pp. 1–6.

- [15] J. Tian, C. Qu, W. Xu, and S. Wang, “Kinwrite: Handwriting-based authentication using kinect.” in *NDSS*, 2013.
- [16] C. Stauffer and W. E. L. Grimson, “Adaptive background mixture models for real-time tracking,” in *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.*, vol. 2. IEEE, 1999.
- [17] Z. Zivkovic and F. van der Heijden, “Efficient adaptive density estimation per image pixel for the task of background subtraction,” *Pattern recognition letters*, vol. 27, no. 7, pp. 773–780, 2006.
- [18] P. D. Z. Varcheie, M. Sills-Lavoie, and G.-A. Bilodeau, “A multiscale region-based motion detection and background subtraction algorithm,” *Sensors*, vol. 10, no. 2, pp. 1041–1061, 2010.
- [19] A. Elgammal, D. Harwood, and L. Davis, “Non-parametric model for background subtraction,” in *Computer Vision—ECCV 2000*. Springer, 2000, pp. 751–767.
- [20] Y. Sheikh and M. Shah, “Bayesian modeling of dynamic scenes for object detection,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 27, no. 11, pp. 1778–1792, 2005.
- [21] Y. Chen, J. Wang, and H. Lu, “Learning sharable models for robust background subtraction,” in *Multimedia and Expo (ICME), 2015 IEEE International Conference on*. IEEE, 2015, pp. 1–6.

- [22] P.-L. St-Charles, G.-A. Bilodeau, and R. Bergevin, “A self-adjusting approach to change detection based on background word consensus,” in *Applications of Computer Vision (WACV), 2015 IEEE Winter Conference on*. IEEE, 2015, pp. 990–997.
- [23] K. Kim, T. H. Chalidabhongse, D. Harwood, and L. Davis, “Background modeling and subtraction by codebook construction,” in *Image Processing, 2004. ICIP’04. 2004 International Conference on*, vol. 5. IEEE, 2004, pp. 3061–3064.
- [24] K. Kim, T. H. Chalidabhongse, D. Harwood, and L. Davis, “Real-time foreground–background segmentation using codebook model,” *Real-time imaging*, vol. 11, no. 3, pp. 172–185, 2005.
- [25] S.-C. Liu, C.-W. Fu, and S. Chang, “Statistical change detection with moments under time-varying illumination,” *Image Processing, IEEE Transactions on*, vol. 7, no. 9, pp. 1258–1268, 1998.
- [26] M. Seki, T. Wada, H. Fujiwara, and K. Sumi, “Background subtraction based on cooccurrence of image variations,” in *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, vol. 2. IEEE, 2003, pp. II–65.
- [27] O. Barnich and M. Van Droogenbroeck, “Vibe: A universal background subtraction algorithm for video sequences,” *Image Processing, IEEE Transactions on*, vol. 20, no. 6, pp. 1709–1724, 2011.
- [28] I. Junejo, A. Bhutta, and H. Foroosh, “Single class support vector machine (svm) for scene modeling,” *Journal of Signal, Image and Video Processing, Springer-Verlag*, May 2011.

- [29] L. Cheng, S. Wang, D. Schuurmans, T. Caelli, and S. Vishwanathan, "An online discriminative approach to background subtraction," in *Video and Signal Based Surveillance, 2006. AVSS'06. IEEE International Conference on*. IEEE, 2006, pp. 2–2.
- [30] M. Gong and L. Cheng, "Foreground segmentation of live videos using locally competing 1svm," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE, 2011, pp. 2105–2112.
- [31] A. Glazer, M. Lindenbaum, and S. Markovitch, "One-class background model," in *Computer Vision-ACCV 2012 Workshops*. Springer, 2012, pp. 301–307.
- [32] H.-H. Lin, T.-L. Liu, and J.-H. Chuang, "Learning a scene background model via classification," *Signal Processing, IEEE Transactions on*, vol. 57, no. 5, pp. 1641–1654, 2009.
- [33] A. Mittal and D. Huttenlocher, "Scene modeling for wide area surveillance and image synthesis," in *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*, vol. 2. IEEE, 2000, pp. 160–167.
- [34] Y. Ren, C.-S. Chua, and Y.-K. Ho, "Statistical background modeling for non-stationary camera," *Pattern Recognition Letters*, vol. 24, no. 1, pp. 183–196, 2003.
- [35] S. Rowe and A. Blake, "Statistical mosaics for tracking," *Image and Vision Computing*, vol. 14, no. 8, pp. 549–564, 1996.
- [36] M. Irani and P. Anandan, "A unified approach to moving object detection in 2d and 3d scenes," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 20, no. 6, pp. 577–589, 1998.

- [37] T. B. Moeslund, A. Hilton, and V. Krüger, “A survey of advances in vision-based human motion capture and analysis,” *Computer vision and image understanding*, vol. 104, no. 2, pp. 90–126, 2006.
- [38] A. Yilmaz, O. Javed, and M. Shah, “Object tracking: A survey,” *Acm computing surveys (CSUR)*, vol. 38, no. 4, p. 13, 2006.
- [39] V. Nair and J. J. Clark, “An unsupervised, online learning framework for moving object detection,” in *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, vol. 2. IEEE, 2004, pp. II–317.
- [40] D. Cremers and S. Soatto, “Motion competition: A variational approach to piecewise parametric motion segmentation,” *International Journal of Computer Vision*, vol. 62, no. 3, pp. 249–265, 2005.
- [41] T. Amiaz and N. Kiryati, “Piecewise-smooth dense optical flow via level sets,” *International Journal of Computer Vision*, vol. 68, no. 2, pp. 111–124, 2006.
- [42] T. Brox, A. Bruhn, and J. Weickert, “Variational motion segmentation with level sets,” in *Computer Vision—ECCV 2006*. Springer, 2006, pp. 471–483.
- [43] M. P. Kumar, P. H. Torr, and A. Zisserman, “Learning layered motion segmentations of video,” *International Journal of Computer Vision*, vol. 76, no. 3, pp. 301–319, 2008.

- [44] H. Tao, H. S. Sawhney, and R. Kumar, “Object tracking with bayesian estimation of dynamic layer representations,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 24, no. 1, pp. 75–89, 2002.
- [45] J. Xiao and M. Shah, “Accurate motion layer segmentation and matting,” in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 2. IEEE, 2005, pp. 698–703.
- [46] S. S. Beauchemin and J. L. Barron, “The computation of optical flow,” *ACM Computing Surveys (CSUR)*, vol. 27, no. 3, pp. 433–466, 1995.
- [47] R. Tron and R. Vidal, “A benchmark for the comparison of 3-d motion segmentation algorithms,” in *Computer Vision and Pattern Recognition, 2007. CVPR’07. IEEE Conference on*. IEEE, 2007, pp. 1–8.
- [48] T. Brox and J. Malik, “Object segmentation by long term analysis of point trajectories,” in *Computer Vision–ECCV 2010*. Springer, 2010, pp. 282–295.
- [49] P. Ochs and T. Brox, “Object segmentation in video: a hierarchical variational approach for turning point trajectories into dense regions,” in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 1583–1590.
- [50] R. Vidal, “A tutorial on subspace clustering,” *IEEE Signal Processing Magazine*, vol. 28, no. 2, pp. 52–68, 2010.
- [51] T. Lim, B. Han, and J. H. Han, “Modeling and segmentation of floating foreground and background in videos,” *Pattern Recognition*, vol. 45, no. 4, pp. 1696–1706, 2012.

- [52] D. Zamalieva, A. Yilmaz, and J. W. Davis, “A multi-transformational model for background subtraction with moving cameras,” in *Computer Vision–ECCV 2014*. Springer, 2014, pp. 803–817.
- [53] M. Narayana, A. Hanson, and E. Learned-Miller, “Coherent motion segmentation in moving camera videos using optical flow orientations,” in *Computer Vision (ICCV), 2013 IEEE International Conference on*. IEEE, 2013, pp. 1577–1584.
- [54] J.-H. Jeon, B.-S. Oh, and K.-A. Toh, “A system for hand gesture based signature recognition,” in *Control Automation Robotics & Vision (ICARCV), 2012 12th International Conference on*. IEEE, 2012, pp. 171–175.
- [55] C. Patlolla, S. Mahotra, and N. Kehtarnavaz, “Real-time hand-pair gesture recognition using a stereo webcam,” in *Emerging Signal Processing Applications (ESPA), 2012 IEEE International Conference on*. IEEE, 2012, pp. 135–138.
- [56] P. Keir, J. Payne, J. Elgoyhen, M. Horner, M. Naef, and P. Anderson, “Gesture-recognition with non-referenced tracking,” in *3D User Interfaces, 2006. 3DUI 2006. IEEE Symposium on*. IEEE, 2006, pp. 151–158.
- [57] E. Farella, S. O’Modhrain, L. Benini, and B. Riccó, “Gesture signature for ambient intelligence applications: a feasibility study,” in *Pervasive Computing*. Springer, 2006, pp. 288–304.
- [58] F. Kristensen, P. Nilsson, and V. Öwall, “Background segmentation beyond rgb,” in *Computer Vision–ACCV 2006*. Springer, 2006, pp. 602–612.

- [59] M. Balcilar, F. Karabiber, and A. Sonmez, "Performance analysis of lab2000h1 color space for background subtraction," in *Innovations in Intelligent Systems and Applications (INISTA), 2013 IEEE International Symposium on*. IEEE, 2013, pp. 1–6.
- [60] T. Horprasert, D. Harwood, and L. S. Davis, "A statistical approach for real-time robust background subtraction and shadow detection," in *IEEE ICCV*, vol. 99, 1999, pp. 1–19.
- [61] Z. Chen, N. Pears, M. Freeman, and J. Austin, "Background subtraction in video using recursive mixture models, spatio-temporal filtering and shadow removal," in *Advances in Visual Computing*. Springer, 2009, pp. 1141–1150.
- [62] R. Cucchiara, C. Grana, M. Piccardi, and A. Prati, "Detecting moving objects, ghosts, and shadows in video streams," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 25, no. 10, pp. 1337–1342, 2003.
- [63] J. Chengjun, C. Guiran, C. Wei, and J. Huiyan, "Background extraction and update method based on histogram in ycbcr color space," in *E-Business and E-Government (ICEE), 2011 International Conference on*. IEEE, 2011, pp. 1–4.
- [64] T. Gevers, A. Gijsenij, J. Van de Weijer, and J.-M. Geusebroek, *Color in computer vision: fundamentals and applications*. John Wiley & Sons, 2012, vol. 23.
- [65] M.-Y. Liu, O. Tuzel, S. Ramalingam, and R. Chellappa, "Entropy rate superpixel segmentation," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE, 2011, pp. 2097–2104.

- [66] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, “A density-based algorithm for discovering clusters in large spatial databases with noise.” in *Kdd*, vol. 96, no. 34, 1996, pp. 226–231.
- [67] P. D. Kovesi, “MATLAB and Octave functions for computer vision and image processing,” available from: <<http://www.peterkovesi.com/matlabfns/>>.
- [68] F. D. P. V. G. Allebosch, D. Van Hamme and W. Philips, “Edge based foreground background segmentation with interior/exterior classification,” in *proceedings of VISAPP*, 2015.
- [69] H. Sajid and S.-C. S. Cheung, “Background subtraction for static & moving camera,” in *IEEE International Conference on Image Processing(ICIP), 2015*. IEEE, 2015.
- [70] H. Sajid and S.-C. S. Cheung, “Background subtraction under sudden illumination change,” in *Multimedia Signal Processing (MMSP), 2014 IEEE 16th International Workshop on*. IEEE, 2014, pp. 1–6.
- [71] D. Huang, C. Shan, M. Ardabilian, Y. Wang, and L. Chen, “Local binary patterns and its application to facial image analysis: a survey,” *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, vol. 41, no. 6, pp. 765–781, 2011.
- [72] C. Silva, T. Bouwmans, and C. Frélicot, “An extended center-symmetric local binary pattern for background modeling and subtraction in videos,” in *International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications, VISAPP 2015*, 2015.

- [73] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 24, no. 7, pp. 971–987, 2002.
- [74] A. Vedaldi and B. Fulkerson, "Vlfeat: An open and portable library of computer vision algorithms," in *Proceedings of the international conference on Multimedia*. ACM, 2010, pp. 1469–1472.
- [75] C.-C. Chang and C.-J. Lin, "Libsvm: a library for support vector machines," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 2, no. 3, p. 27, 2011.
- [76] H. Sajid, S. C. Sen-ching, and N. Jacobs, "Appearance based background subtraction for ptz cameras," *Signal Processing: Image Communication*, vol. 47, pp. 417–425, 2016.
- [77] C. Liu, "Beyond pixels: exploring new representations and applications for motion analysis," Ph.D. dissertation, Citeseer, 2009.
- [78] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 23, no. 11, pp. 1222–1239, 2001.
- [79] V. Kolmogorov and R. Zabih, "What energy functions can be minimized via graph cuts?" *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 26, no. 2, pp. 147–159, 2004.

- [80] Y. Boykov and V. Kolmogorov, “An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 26, no. 9, pp. 1124–1137, 2004.
- [81] S. Calinon and A. Billard, “Incremental learning of gestures by imitation in a humanoid robot,” in *Proceedings of the ACM/IEEE international conference on Human-robot interaction*. ACM, 2007, pp. 255–262.
- [82] J. A. M. Basilio, G. A. Torres, G. S. Pérez, L. K. T. Medina, and H. M. P. Meana, “Explicit image detection using ycbcr space color model as skin detection,” *Applications of Mathematics and Computer Engineering*, pp. 123–128, 2011.
- [83] S. K. Singh, D. Chauhan, M. Vatsa, and R. Singh, “A robust skin color based face detection algorithm,” *Tamkang Journal of Science and Engineering*, vol. 6, no. 4, pp. 227–234, 2003.

Vita

Hasan Sajid was born in Lahore, Punjab, Pakistan.

EDUCATION:

- Master of Science in Electrical Engineering, University of Kentucky, USA, 2014
- B.S. in Mechatronics Engineering, National University of Sciences and Technology, Pakistan, 2007

PROFESSIONAL POSITIONS:

- Research Assistant, 05/2014 – 08/2016, Center for Visualization & Virtual Environments, University of Kentucky, KY, USA
- Team Lead (Mechatronics), 06/2011 – 08/2012, National University of Sciences and Technology, Pakistan
- Research Officer, 10/2007 - 05/2011, Public sector R&D Organization, Pakistan

HONORS AND AWARDS:

- Best Paper Finalist in IEEE Multimedia Signal Processing (MMSP), 2014
- US State Dept. Fulbright scholarship for MS program, University of Kentucky, 2012
- Power and Energy Institute of Kentucky tuition fee waiver scholarship, USA, 2012.

- Won *1st Prize* in National Engineering Robotics Contest (NERC) 2006, organized by NUST, FESTO Germany and Higher Education Commission (HEC) Pakistan. A total of **54 teams** participated in the nationwide contest.

PUBLICATIONS:

- Hasan Sajid, Sen-Ching Samson Cheung and Nathan Jacobs, "Appearance based Background Subtraction for Moving Cameras", Journal of Signal Processing: Image Communication, 2016.
- Hasan Sajid, Sen-Ching Samson Cheung, "Universal Multi-Mode Background Subtraction", IEEE Transactions on Image Processing (TIP), 2015. (First Revision)
- Hasan Sajid and Sen-Ching Samson Cheung, "V_{Sig}: Hand-Gestured Signature Recognition and Authentication with Wearable Camera", in IEEE International Workshop on Information Forensics and Security (WIFS), 2015.
- Hasan Sajid and Sen-Ching Samson Cheung, "Background subtraction for Static and Moving Camera", in IEEE International Conference on Image Processing (ICIP), 2015.
- Hasan Sajid, Sen-Ching Samson Cheung, "Background subtraction under sudden illumination change", IEEE workshop on Multimedia Signal Processing (MMSP), pp. 1-6, 2014.
- O. Hasan, S. Mansoor, H. Sajid, Z. Amjad, S. Nisar and T. Hasan, "Al Zahrawi - A Training Robot and Simulator for Minimal Invasive Surgery". INMIC 2011.
- Nassar Ikram, Shakeel Durranii, Hasan Sajid, Husnain Saeed, "A Wireless Multimedia Sensor Network Based Intelligent Safety and Security System (IS₃),"

sensorcomm, pp.388-392, 2009 Third International Conference on Sensor Technologies and Applications, 2009.

- Dr. Nassar Ikram, Shakeel Durrani, Hasan Sajid and Husnain Saeed, ‘Wireless Sensor Networks: Fundamentals, Technology and Applications’, In: Junaid Ahmed Zubairi (Ed), Applications of Modern High performance Networks, Bentham Publishers, NY USA. (eISBN: 978-1-60805-077-2, 2009)
- Hasan Sajid, "A Universal Background Subtraction System". MS Thesis, University of Kentucky, 2014.
- Hasan Sajid, Sen-Ching Samson Cheung and Nathan Jacobs, "Motion and Appearance based Background Subtraction for Freely Moving Cameras", In preparation for IEEE Transactions on Circuit and Systems for Video Technology(TCSVT), 2016.