HUMAN ACTIVITY DETECTION USING TEMPORAL INFORMATION FROM A VIDEO

A thesis submitted in partial fulfillment of the requirement for the

Degree of Master of Computer Application

of

Jadavpur University

By

RIA PAUL

Registration Number: 137320 of 2016-2017

Examination Roll Number: MCA196010

Under the Guidance of

Dr. Debotosh Bhattacharjee

Department of Computer Science and Engineering

Jadavpur University, Kolkata-700032

India

May 2018

FACULTY OF ENGINEERING AND TECHNOLOGY

JADAVPUR UNIVERSITY

CERTIFICATE OF RECOMMENDATION

This is to certify that the thesis entitled "HUMAN ACTIVITY DETECTION USING TEMPORAL INFORMATION FROM A VIDEO" has been satisfactorily completed by Ria Paul (University Registration No.: 137320 of 2016-17, Examination Roll No.: MCA196010). It is a bonafide piece of work carried out under my guidance and supervision and be accepted in partial fulfillment of the requirement for the Degree of Master of Computer Application, Department of Computer Science and Engineering, Faculty of Engineering and Technology, Jadavpur University, Kolkata.

_____

Dr. Debotosh Bhattacharjee. (Thesis Supervisor)

Department of Computer Science and Engineering

Jadavpur University, Kolkata-700032

Countersigned

_____

Prof. Mahan Tapas Kundu.

Head, Department of Computer Science and Engineering,

Jadavpur University, Kolkata-700032.

_____

Prof. Chiranjib Bhattacharjee

Dean, Faculty of Engineering and Technology,

Jadavpur University, Kolkata-700032.

FACULTY OF ENGINEERING AND TECHNOLOGY
JADAVPUR UNIVERSITY

CERTIFICATE OF APPROVAL

This is to certify that the thesis entitled "HUMAN ACTIVITY DETECTION USING TEMPORAL INFORMATION FROM A VIDEO" is a bonafide record of work carried out by Ria Paul in partial fulfillment of the requirements for the award of the degree of Master of Computer Application in the Department of Computer Science and Engineering, Jadavpur University during the period of January 2018 to May 2018. It is understood that by this approval the undersigned do not necessarily endorse or approve any statement made, opinion expressed or conclusion drawn there in but approve the thesis only for the purpose for which it has been submitted.

_____

Signature of Examiner

Date:

_____

Signature of Superviser

Date:

FACULTY OF ENGINEERING AND TECHNOLOGY
JADAVPUR UNIVERSITY

DECLARATION OF ORIGINALITY AND COMPLIANCE OF ACADEMIC ETHICS

I hereby declare that this thesis entitled "HUMAN ACTIVITY DETECTION USING TEMPORAL INFORMATION FROM A VIDEO" contains literature survey and original research work by the undersigned candidate, as part of his Degree of Master of Computer Application.

All information in this document has been obtained and presented in accordance with academic rules and ethical conduct.

I also declare that, as required by these rules and conduct, I have fully cited and referenced all materials and results that are not original to this work.

Name: Ria Paul

University Registration No. : 137320 of 2016-17

Examination Roll No. : MCA196010

Thesis Title: HUMAN ACTION DETECTION FROM A VIDEO

_____

Signature

Date:

ACKNOWLEDGEMENT

I am pleased to express my deepest gratitude to my thesis guide, **Dr. Debotosh Bhattacharjee,** Department of Computer Science and Engineering, Jadavpur University, Kolkata for his invaluable guidance, constant encouragement and inspiration during the period of my dissertation.

I am highly indebted to **Jadavpur University** for providing me the opportunity and the required infrastructure to carry on my thesis.

I am thankful to all the teaching and non-teaching staff whose helping hands have smoothed my journey through the period of my research.

Last but not the least; I would like to thank my family members, classmates, seniors and friends for giving me constant encouragement and mental support throughout my work.

Ria Paul

University Registration No. : 137320 of 2016-17

Examination Roll No. : MCA196010

Master of Computer Application

Department of Computer Science and Engineering

Jadavpur University

ABSTRACT

With the wide range of applications in vision-based intelligent systems, the attention of researchers in the computer vision field have attracted by image and video analysis technologies. Despite the diversity of computer vision researches, few literature reviews have been proposed to monitor people and recognize their activities.

This thesis represents a study on human activity recognition from video. In this work, I proposed an effective way for human activity recognition by detecting moving parts of the human body by using a Gaussian Mixture Model and people detection technique. Video file is segmented as frames in the form of RGB images, and these images are used for feature extraction.

Publicly available Weizmann and KTH datasets are used for both training and test sample. From each frame moving human part is identified first, and then the identified part is used for feature extraction. For a moving activity like running, walking velocity and width is calculated from the centroid. For still activity like hand clapping and hand waving frame subtraction technique is used to find the centroid. The experimental results datasets validate the efficiency of the proposed technique .

Keywords: Foreground detection, Edge detection, Blob analysis, Centroid, Gaussian Mixture Model, HOG features, People detection

CONTENTS

CHAPTER 1

INTRODUCTION

Nowadays, it's a very hot topic on video-based human action detection, which has recently been demonstrated to be very useful in a wide range of applications including video surveillance, telemonitoring of patients and senior people, medical diagnosis and training, video content analysis and search, and intelligent human-computer interaction. This is mainly driven by the need to find innovative ways to encourage physical activity. An example of a health application of HAR is SELFBACK 1 , an EU funded project that is developing a self-management system for patients with Lower Back Pain.

According to Mannini and Sabatini, human activities can be categorized into static postures, such as sitting, standing, lying; or dynamic motions, such as running, walking, stair climbing, and so forth. There are different ways to represent actions and extract features for action recognition. Most of the approaches in the field can be divided into five categories:

1. Spatio-Temporal
2. Frequency Based
3. Local Descriptors
4. Shape-Based
5. Appearance-Based

My basic idea in my work is that detection of the foreground object and use it for feature extraction. There are lots of methods by which we can detect foreground object Cluster-Based Background Modelling Algorithm, Temporal average filter, Gaussian mixture model, Teknomo- Fernandez algorithm, Scaling Coefficients, but we use Gaussian Mixture Model, HOG features and SVM classifier for foreground object detection.

Now for moving activity (e. g. running, walking, jumping, jogging, etc.) velocity and body stretchiness is calculated with respect to its position in the frame and for still activity (e. g. hand clapping, hand waving) frame subtraction method is used to calculate the centroid of each frame depending on the key pose of human action.

(a) *Foreground human object detection*: My project depends on human detection. It depends on the accuracy of the detection of the foreground object from a frame. Gaussian Mixture Model is used for foreground detection for moving activity (e.g., running, walking, jumping) and HOG feature and SVM classifier is used for still activity(e.g., hand waving, hand clapping, etc.)

(b) *Selection Of Action Region*: There are different kind of regions and more than one regions in a frame. Any human action is a combination of the human body. Identification of these important regions or active regions and their features are the basis for defining any action. Higher accuracy in the selection of active regions ensures better efficiency in activity recognition.

(c) *The Classifiers*: It recognizes the actual active regions. It ensures the efficiency to recognize the actual region. So, it is one of the crucial tasks in activity recognition.

ACTIVITY RECOGNITION CHALLENGES

There are still many issues and challenges that motivate the development of new activity recognition techniques to improve accuracy under more realistic conditions. Challenges corresponding with activity recognition have been discussed in researches . A number of these challenges are:

•*Human behavior*: performing multiple tasks at the same time makes the recognition process more difficult .

•*The definition of physical activities:* develop a clear understanding of the definition of the activities under investigation and their specific characteristics .

•*Intraclass variability:* the same activity may be performed differently by different individuals .

•*Intraclass similarity:* classes that are fundamentally different, but that show very similar characteristics in the sensor data .

•*Selection of attributes and sensors:* the selection of the attributes to be measured and the sensors that measure it plays an important role in recognition performance .

•*Sensor inaccuracy:* the sensor data play an essential role in the overall recognition results.

•*Sensor placement:* the wrong installation or orientation of sensors could be causing a problem or affect the recognition performance .

•*Resource constraints:* power consumption is the main factor affecting the size of the battery and sensor nodes (if using inertial sensor) .

•*Usability*: the systems should be easier to learn and more efficient to use.

•*Privacy*: sensitive user information should be not invading users' private life.

•*Multiple residents*: More than one resident can be present in the same environment. And of course, another challenge is corresponding to the application domain itself, but we present the common and the most popular.

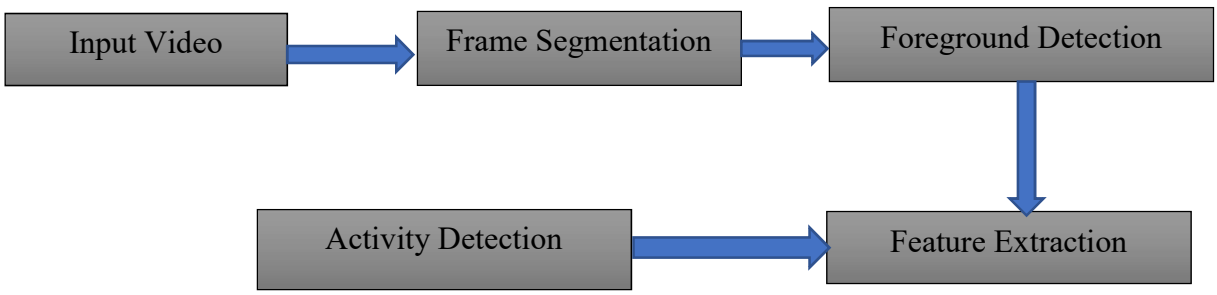
MOTIVATION

Within this field, we observed unsolved issues in recognition of actions that had a degree of fuzziness in belonging to a particular class. These are a group of actions which possess some similarities and can be confused among themselves during recognition. Examples of such action pair include running-walking, hopping-walking, running-hopping. Our method attempts at recognizing these actions accurately. In addition to this, we wanted to come up with a method that can be easily scaled from processing over from 2-D datasets to 3-D datasets.

THE OBJECTIVE OF THE THESIS

Our main objective of the project is to recognize human activity in a simple mathematical way from video databases using some simple and effective feature extraction techniques. The process includes :

i. At first, the video is divided into several frames.
ii. Segmenting and resizing the video frames.
iii. Processing of the video frames to extract features.
iv. Different foreground detection technique to separate moving and still activity.
v. Extract features to calculate velocity and body stretchiness to separate moving activities.
vi. Extract features to calculate the change of upper body mass to separate still activities.
vii. Compute the accuracy of the proposed method.



APPLICATION OF HUMAN ACTIVITY RECOGNITION

Activity recognition is a core building block behind many new applications. The applications of mobile activity recognition can be classified according to their targeted beneficial subjects:

1) Applicable for the end users such as fitness tracking, health monitoring, fall detection, behavior-based context- awareness, home and work automation, and self-managing system;

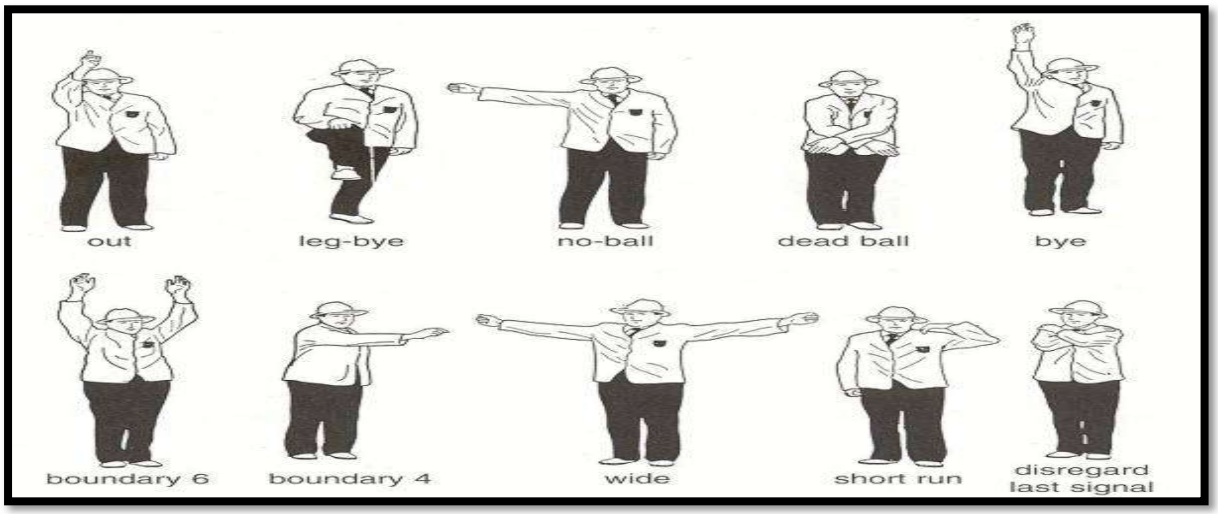
2) Applications for the third parties such as targeted advertising, research platforms for the data collection, corporate management[18], and accounting;

3) Applications for the crowds and groups such as social networking and activity-based crowdsourcing. In this section, review some representative applications.

4) *Daily Life Monitoring*: Applications in regular life monitoring usually aim to provide a convenient reference for the activity logging, or assisting with exercise and healthy lifestyles. These devices are equipped with the embedded sensors such as accelerometer, gyroscope, GPS; and they track people's steps taken, stairs climbed, calorie burned, hours slept, distance traveled, quality of sleep, etc. An online service is provided for users to review data tracking and visualization in reports. Compared with smartphone sensors, these devices are more sophisticated since their sensors are explicitly designed for activity detection and monitor. The drawback is that they are much more expensive.

5) *Surveillance*: Cameras installed in areas that may need something such as banks, airports, military installations, and convenience stores. Currently, surveillance systems are mainly for recordings. Activity recognition using CCTV's aims to monitor suspicious activities for real-time reactions like fighting and stealing.

5) Sports play analysis: analyzing the play and deducing the actions in the sports given below



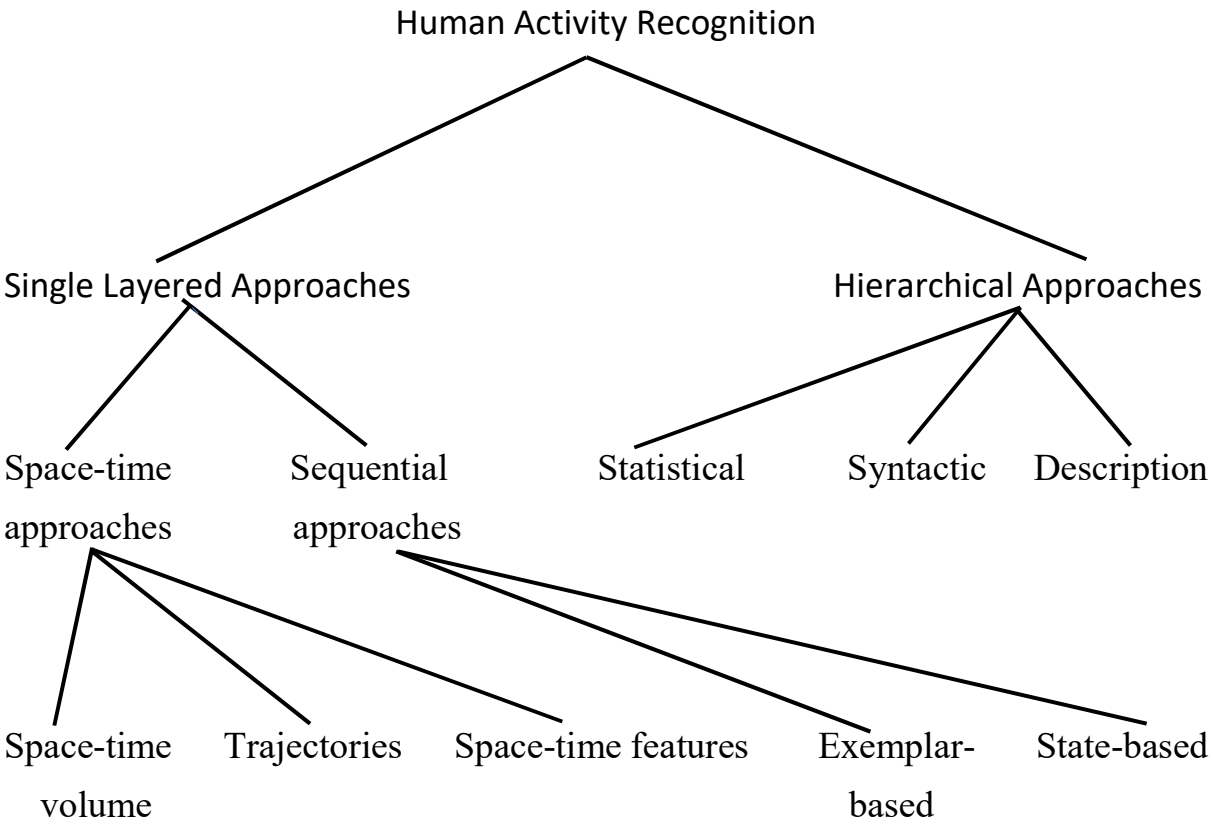
6) *Personal Biometric Signature*: A subject's motion pattern is usually exclusive and unique. For example, when people raise their hands, it is almost impossible for two people's hands to share the same motion patterns. Even in a successful imitation, the differences still exist because of the difference in the motion related bones and muscles on human bodies. Sensors such as accelerometers can capture those differences. The activity recognition techniques provide a possible solution for human biometric signature with patterns in motion/gestures.

7) *Elderly and Youth Care*: There is a growing need in elderly care (both physically and mentally), partially because of the retirement of the baby Boomer generation. A primary goal of the current research in human activity monitoring is to develop new technologies and applications for elderly care. Those applications could help prevent harm, e.g., to detect older people's dangerous situations. Architecture on the smartphone is developed with the purpose of users' fall detection

HUMAN ACTIVITY CATEGORIZATION

The human activity categorization problem has remained a challenging task in computer vision for more than two decades. Previous works on characterizing human behavior have shown great potential in this area. First, we categorize the human activity recognition methods into two main categories: (i) *unimodal* and

(ii) *multimodal* activity recognition methods according to the nature of sensor data they employ. Then, each of these two categories is further analyzed into sub-categories depending on how they model human activities. Thus, we propose a hierarchical classification of the social activity recognition methods



RELATED WORKS

Many different feature extraction approaches have been proposed for accelerometer data for activity recognition [3]. We broadly classify these into handcrafted, frequency-transform and deep features.

2.1 *Hand-crafted Features*: This is the most common approach to HAR and involves the computation of several defined measures on either the raw accelerometer data (time-domain) or the frequency transformation of the data (frequency domain).These measures are designed to capture the characteristics of the signal that are useful for distinguishing different classes of activities. In the case of both time and frequency domains, the input is a vector of real values $\vec{v} = v_1, v_2, \ldots v_n$ for each axis x, y, and z. A function θi is then applied to each

vector to compute a single feature value. Standard time domain features include mean, standard deviation, and percentiles; while typical frequency domain features include energy, spectral entropy, and dominant frequency. The time-domain and frequency domain features used in this work are presented in the following Table.

Time Domain Features	Frequency Domain Features
Mean	Dominant frequency
Standard deviation	Spectral centroid
Inter-quartile range	Maximum
Lag-one-autocorrelation	Mean
Peak-to-peak amplitude	Median
Power	Standard Deviation
Skewness	
Kurtosis	
Log-energy	
Zero crossings	
Root squared mean	

While hand-crafted features have worked well for HAR, a significant disadvantage is that they are domain specific. A different set of features need to be defined for each different type of input data, i.e. accelerometer, gyroscope, etc. Hence, some understanding of the characteristics of the data is required. Also, it is not always clear which features are likely to work best. Choice of elements is usually made through empirical evaluation of different combinations of features or with the aid of feature selection algorithms

2.2 *Frequency Transform Features*: Frequency transform features extraction involves applying a single function ϕ on the raw accelerometer data to transform this into the frequency domain, where it is expected that distinctions between different activities are more emphasized. The main difference between frequency transform and hand-crafted features is that the coefficients of the

transformation are directly used for feature representation without taking further measurements. Common transformations that have been applied include Fast Fourier Transforms (FFTs) and Discrete Cosine Transforms (DCTs). FFT is an efficient algorithm optimized for computing the discrete Fourier transform of digital input. Fourier transforms decompose an input signal into its constituent sine waves. In contrast, DCT, a similar algorithm to FFT, decomposes a given signal into its constituent cosine waves. Also, DCT returns an ordered sequence of coefficients such that the most significant information is concentrated at the lower indices of the sequence. This means that higher DCT coefficients can be discarded without losing information, making DCT better for compression.

2.3 *CNN Feature Extraction*: Convolutional Neural Networks (CNNs) have been applied for feature extraction in HAR, due to their ability to model local dependencies that may exist between adjacent data points in the accelerometer data [8]. CNNs are a type of Deep Neural Network that is able to extract increasingly more abstract feature representations by passing the input data through a stack of multiple convolutional operators [4], where each layer in the stack takes as input, the output of the previous layer of convolutional operators.

CHAPTER 2

CONCEPT OF PROJECT-BASED TOPICS

Image Processing:

Image processing is a method to convert an image into digital form and perform some operations on it, to get an enhanced image or to extract some useful information from it. It is a type of signal dispensation in which input is an image, like video frame or photograph and output, may be image or characteristics associated with that image. Usually, the **Image Processing** system includes treating images as two-dimensional signals while applying already set signal processing methods to them.

Digital Processing techniques help in the manipulation of digital images by using computers. To get over such flaws and to get originality of information, it has to undergo various phases of processing. The three general phases that all types of data have to undergo while using the digital technique are Preprocessing, enhancement and display, information extraction.

Video Processing:

Video Processing Video processing covers most of the image processing methods but also includes ways where the temporal nature of video data is exploited. Image Analysis Here, the goal is to analyze the image first to find objects of interest and then extract some parameters of these objects. For example, finding an object's position and size.

Analog video is a video signal transferred by an analog signal. When combined into one channel, it is called composite video as is the case, among others with NTSC, PAL, and SECAM.

Analog video may be carried in separate channels, as in two-channel S - Video (YC) and multi-channel component video formats.

Interlacing was invented because, when standards were being defined, it was difficult to transmit the amount of information in a full frame quickly enough to avoid flicker. The double number of fields presented to the eye reduces perceived flicker.

Because of interlacing, the odd and even lines are displaced in time from each other. This is generally not noticeable except when fast action is taking place

onscreen, when blurring may occur. For example, in the video in the following figure, the moving helicopter is blurred more than the still background.

The progressive and interlaced scan pattern

Progrssive scan captures, transmits, and displays an image in a path similar to text on a page—line by line, top to bottom. The interlaced scan pattern in a CRT display also completes such a scan, but in two passes (two fields). The first pass displays the first and all odd numbered lines, from the top left corner to the bottom right corner. The second pass displays the second and all even numbered lines, filling in the gaps in the first scan.

This scan of alternate lines is called *interlacing*. A *field* is an image that contains only half of the lines needed to make a complete picture. Persistence of vision makes the eye perceive the two fields as a continuous image. In the days of CRT displays, the afterglow of the display's phosphor aided this effect.

Interlacing provides full vertical detail with the same bandwidth that would be required for a full progressive scan, but with twice the perceived frame rate and refresh rate. To prevent flicker, all analog broadcast television system used interlacing.

.

Spatial Sampling

Spatial sampling is the process of collecting observations in a two-dimensional framework. Careful attention is paid to (1) the quantity of the samples, dictated by the budget at hand, and (2) the location of the samples. A sampling scheme is generally designed to maximize the probability of capturing the spatial variation of the variable under study. Once initial samples have been collected and its variation documented, additional measurements can be taken at other locations. This approach is known as second-phase sampling, and various optimization criteria have recently been proposed to determine the optimal location of these new observations. In this chapter, we review the fundamentals of spatial sampling and second-phase designs. Their characteristics and merits under different situations are discussed, while a numerical example illustrates a modeling strategy to use covariate information in guiding the location of new samples. The chapter ends with a discussion on heuristic methods to accelerate the search procedure.

Temporal Sampling

In digital video, the temporal sampling rate is defined as the frame rate– or rather the field rate – rather than the notional pixel clock. The image sampling

frequency is the repetition rate of the sensor integration period. Since the integration period may be significantly shorter than the time between repetitions, the sampling frequency can be different from the inverse of the sample time:

- 50 Hz – PAL video
- 60 / 1.001 Hz ~= 59.94 Hz – NTSC video

Video digital to analog converter operate in the megahertz range (from ~3 MHz for low-quality composite video scalers in early games consoles to 250 MHz or more for the highest-resolution VGA output).

When the analog video is converted to digital video, a different sampling process occurs, this time at the pixel frequency, corresponding to a spatial sampling rate along scan lines. A common pixels sampling rate is:

- 13.5 MHz – CCIR 601, D1 video

Video Format

A **video file format** is a type of file format for storing digital video data on a computer system. Video is almost always stored using lossy compression to reduce the file size.

A video file normally consists of a container (e.g., in the Matroska format) containing video data in a video coding format (e.g.,VP9) alongside audio data in an audio coding format. The container can also contain synchronization information, subtitles, and metadata such as the title. A standardized video file type such as .webm is a profile specified by a restriction on which container format and which video and audio compression formats are allowed.

The good design dictates typically that a file extensions enables the user to derive which program will open the file from the file extension. In contrast to that, some very general-purpose container types like AVI (.avi) and QuickTime (.mov) can contain video and audio in almost any format, and have file extensions named after the container type, making it very hard for the end user to use the file extension to derive which codec or program to use to play the files.

CHAPTER 3

METHODS AND MATERIALS

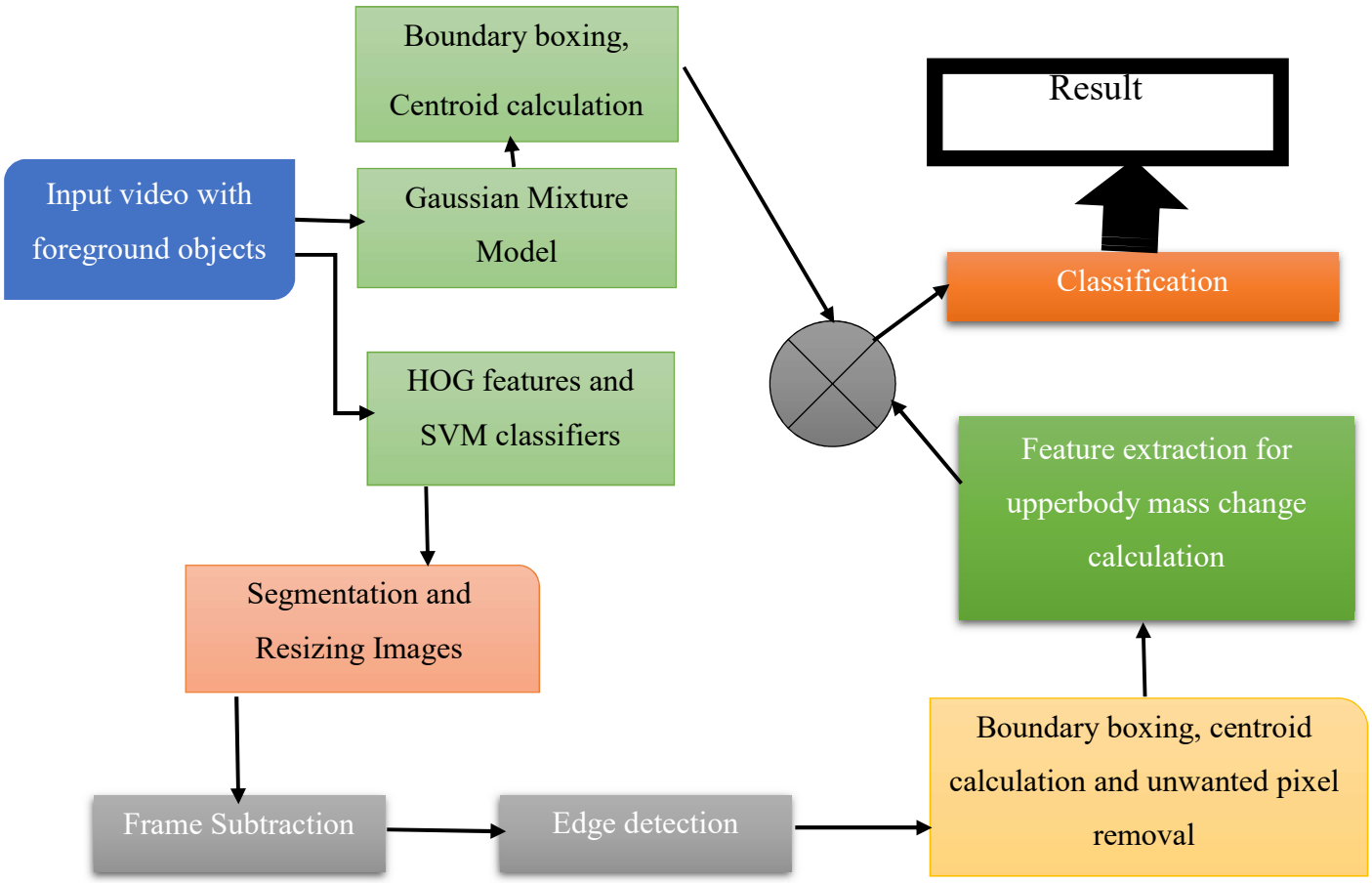
In this section, a detailed process for single human activity detection using video frames. The proposed method comprises mainly four steps:

1. Segmentation
2. Foreground detection
3. Processing
4. Classification.

Methods used in this project are written step by step :

a. Frames are extracted from videos.
b. In this project, gray images which are extracted from the video are used.
   Total around 100 videos are used for each activity detection from KTH, Weizzman data set.
c. Most of these videos are taken using still camera with a still background as well. From each video, first frames are extracted and saved them as 256*256 pixels format for still activity detection.
d. For moving activity detection at first foreground, human detection is done by Gaussian Mixture Model.
e. For still activity detection at first foreground, human detection is done using Histogram of Oriented Gradient (HOG Features) features and a trained Support Vector Machine (SVM) classifier. The object detects unoccluded people in an upright position.
f. After object detection features are calculated by using boundary boxing and centroid of the detected object of almost 100 images of a video.
g. After feature extraction, classification is done for moving activity from the predefined threshold, which is set from the previous experience of the sample set.

h. For still activity, after human object detection, those are saves as a gray image which is converted to a binary image for frame subtraction.
i. After frame subtraction features are extracted and predefined threshold, which is set from the previous experience of the sample set is used to identify the activity.

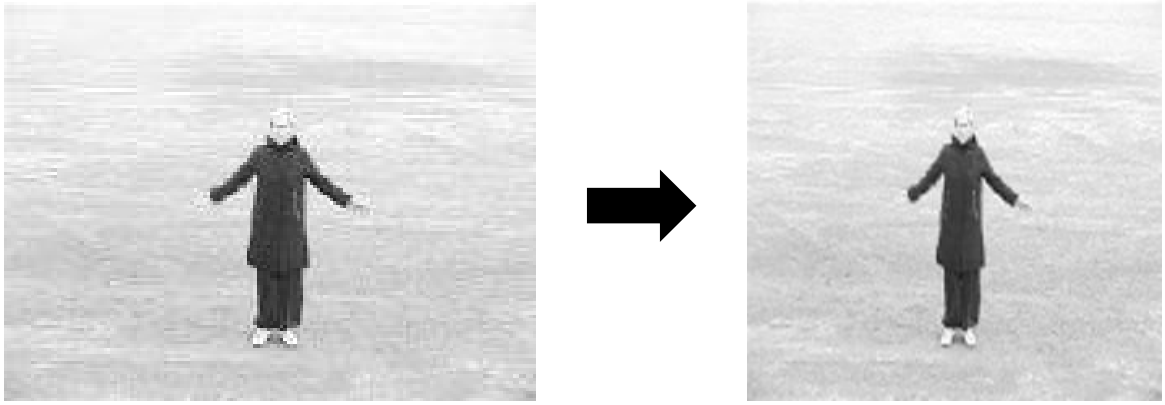


Matlab R2016a does the entire task in this thesis.

SEGMENTING AND RESIZING IMAGES

The mostly static camera is used to capture the video both in KTH and Weizmann dataset. There are 30 different persons' videos from KTH dataset, and ten different person's video from Weizmann Dataset is recorded for each activity. In this project, we have segmented each video according to its frames

rate. If in any video, a number of frames is greater than 200, then only 100 frames are stored in a file size 256*256 pixels for further work.



Original frame Resized frame

FOREGROUND DETECTION

Foreground detection is one of the important tasks in the field of computer vision and image processing, whose aim is to detect changes in image sequences. **Background subtraction** is any technique which allows an image's foreground to be extracted for further processing (object recognition, etc.).

Many applications do not need to know everything about the evolution of movement in a video sequence, but only require the information of changes in the scene, because an image's regions of interest are objects (humans, cars, text, etc.) in its foreground. After the stage of image preprocessing (which may include image denoising, post-processing like morphology, etc.) object localization is required, which may make use of this technique.

It is detecting foreground to separate these changes taking place in the foreground of the background. It is a set of techniques that typically analyze the video sequences in real time and are recorded with a stationary camera.

Here two different techniques are used for foreground detection. For moving activity detection Gaussian Mixture Model is used. Since in Matlab Gaussian Mixture Model is unable to detect a still object, HOG feature extraction and SVM are combined for still object detection.

Gaussian Mixture Model

In statistics, a **mixture model** is a probabilistic model for representing the presence of subpopulations within an overall population, without requiring that an observed data set should identify the sub-population to which an individual observation belongs. Formally a mixture model corresponds to the mixture distribution that represents the probability distribution of observations in the overall population. However, while problems associated with "mixture distributions" relate to deriving the properties of the overall population from those of the sub-populations, "mixture models" are used to make statistical inference about the properties of the sub-populations given only observations on the pooled population, without sub-population identity information.

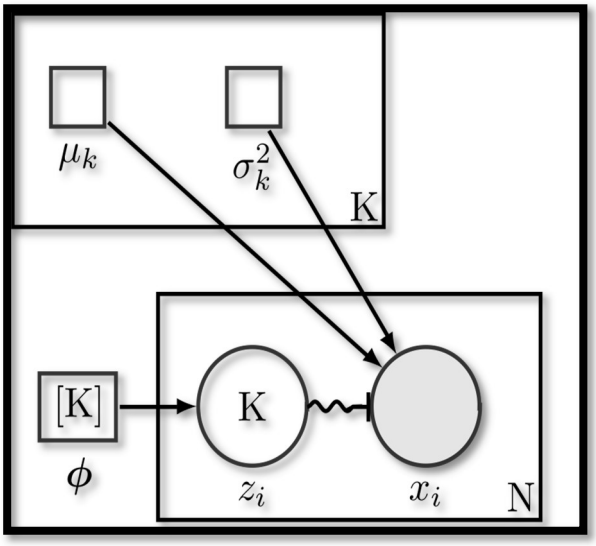
Mixture models should not be confused with models for compositional data, i.e., data whose components are constrained to sum to a constant value (1, 100%, etc.). However, compositional models can be thought of as mixture models, where members of the population are sampled at random. Conversely, mixture models can be thought of as compositional models, where the total size reading population has been normalized to 1.

A typical finite-dimensional mixture model is a hierarchical model consisting of the following components:

- *N* random variables that are observed, each distributed according to a mixture of *K* components, with the components belonging to the same parametric family of distributions (e.g., all normal, all Zipfian, etc.) but with different parameters
- *N* random latent variables specifying the identity of the mixture component of each observation, each distributed according to a *K*-dimensional categorical distribution.
- A set of *K* mixture weights, which are probabilities that sum to 1.
- A set of *K* parameters, each specifying the parameter of the corresponding mixture component. In many cases, each "parameter" is a set of parameters. For example, if the mixture components are Gaussian Distributions, there will be a mean and variance for each component. If the mixture components are categorical distributions (e.g., when each observation is a token from a finite alphabet of size *V*), there will be a vector of *V* probabilities summing to 1.

Mathematically, a basic parametric mixture model can be described as follows:

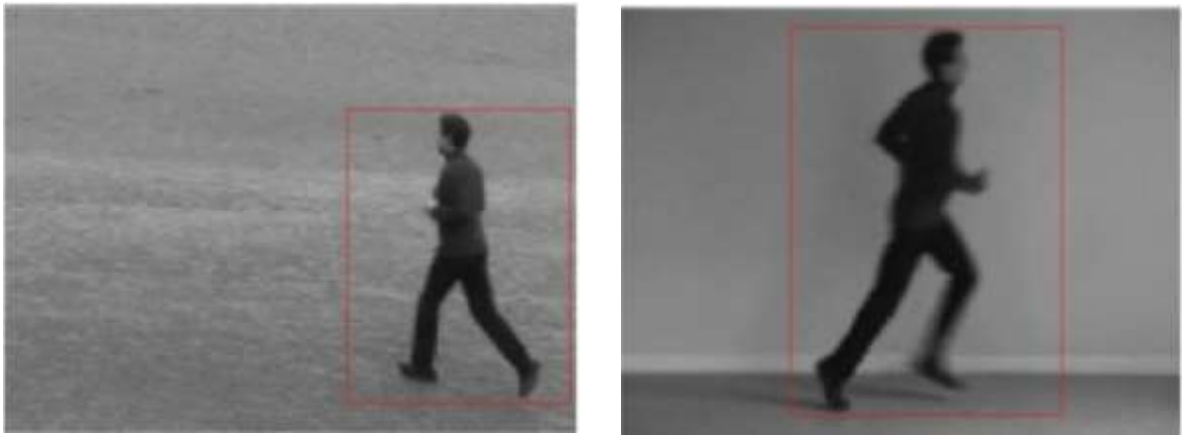
$$
\begin{aligned}
K &= \text{number of mixture components} \\
N &= \text{number of observations} \\
\theta_{i=1\ldots K} &= \text{parameter of distribution of observation associated with component } i \\
\phi_{i=1\ldots K} &= \text{mixture weight, i.e., prior probability of a particular component } i \\
\boldsymbol{\phi} &= K\text{-dimensional vector composed of all the individual } \phi_{1\ldots K}\text{; must sum to 1} \\
z_{i=1\ldots N} &= \text{component of observation } i \\
x_{i=1\ldots N} &= \text{observation } i \\
F(x|\theta) &= \text{probability distribution of an observation, parametrized on } \theta \\
z_{i=1\ldots N} &\sim \text{Categorical}(\boldsymbol{\phi}) \\
x_{i=1\ldots N}|z_{i=1\ldots N} &\sim F(\theta_{z_i})
\end{aligned}
$$



Non-bayesian categorical mixture model using plate notation

Smaller squares indicate fixed parameters; larger circles indicate random variables. Filled-in shapes indicate known values. The indication [K] means a vector of size *K*.

In this project, I use 50 video frames for training background model with 3 Gaussian modes in the mixture model, and the initial mixture model variance for “uint8” image data type is 30*30. The threshold to determine the background model is 0.07



Detected people using GMM

HOG FEATURES AND SVM CLASSIFIERS

The **histogram of oriented gradients (HOG)** is a feature descriptor used in computer vision and image processing for object detection. The technique counts occurrences of gradient orientation in localized portions of an image. This method is similar to that of edge orientation histograms,scale invariant feature transform descriptors, and shape context, but differs in that it is computed on a dense grid of uniformly spaced cells and uses overlapping local contrast normalization for improved accuracy.

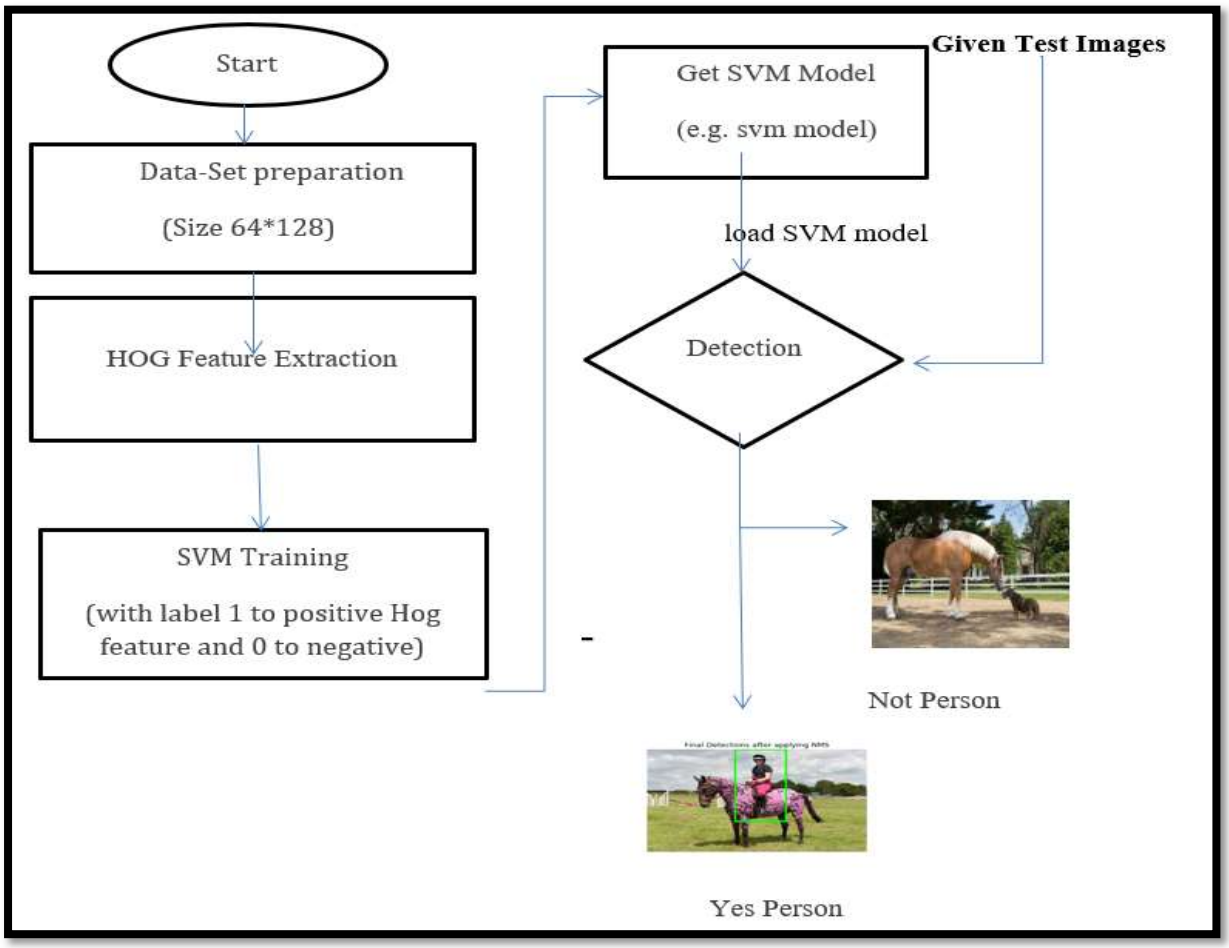
The histogram of oriented gradients (HOG) was first proposed by Dalal and Triggs [23] as an image descriptor for localizing pedestrians. In this work, we use weighted HOGs as implemented in [1], where the subregions are obtained by dividing each image cell into 5×6 equal non-overlapping regions.3 In each subregion, the orientation and magnitude of each pixel are calculated. The absolute orientations are discretized over nine equally sized bins in the 0°- 180° range and the resulting 9-bin histogram is calculated weighting each pixel by the magnitude of its orientation according to the histogram bin.

Robert K. McConnell of Wayland Research Inc. first described the concepts behind HOG without using the term HOG in a patent application in 1986. However, usage only became widespread in 2005 when Navneet Dalal and Bill Triggers, researchers for the French National Institute for Research in Computer Science and Automation (INRIA), presented their supplementary work on HOG descriptors at the Conference on Computer Vision and Pattern Recognition (CVPR). In this work, they focused on pedestrian detection in static images, although since then they expanded their tests to include human detection in videos, as well as to a variety of common animals and vehicles in static imagery.

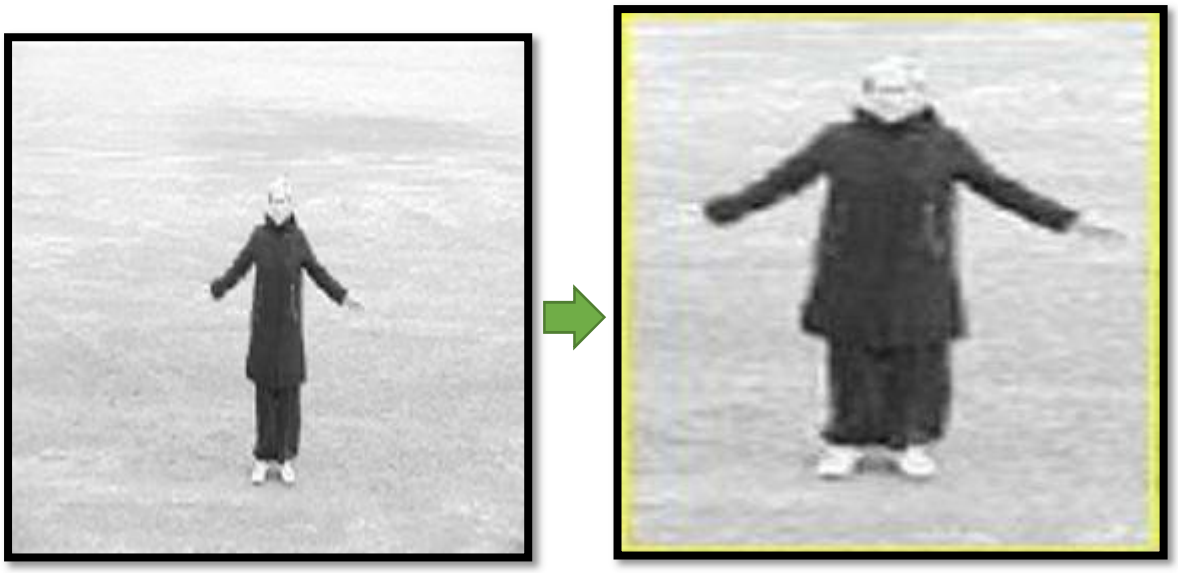
The intent of a feature descriptor is to generalize the object in such a way that the same object produces as close as possible to the classification task easier.

The creators of this approach trained a Support Vector Machine to recognize HOG descriptors of people.

The HOG person detector is fairly simple to understand. One of the main reason for this is that it uses a “global” feature to describe a person rather than a collection of local features. The HOG person detector uses a sliding detection window which is moved around the image. At each position of the detector window, a HOG descriptor is computed for the detection window. This description is then shown to the trained SVM, which classifies it as either a “person” or “ not a person”.to recognize persons at different scales the image is sub-sampled to multiple sizes. Each of the sub-sampled images is searched.in the project pre-defined “peopleDetector” method is used for people detection, which is created using HOG feature and SVM classifiers. Human is detected from the frames and discovered part is stored for feature extraction.



Flowchart Person identification using HOG features



Original image and corresponding detected humans

FEATURE EXTRACTION

Two types of features are extracted for this project. One is for moving activity (e.g., running, walking, etc)and another is for still activity (e.g., hand waving, hand clapping).

Terms and methods

Frame Subtraction:

Frame subtraction is a widely used approach for detecting moving objects in videos from static cameras. The rationale in the approach is that of detecting the moving objects from the difference between the current frame and a reference frame, often called "background image," or "background model." Frame subtraction is mostly done if the image in question is a part of a video stream. Background subtraction provides important cues for numerous applications in computer vision, for example, surveillance tracking or human poses estimation.

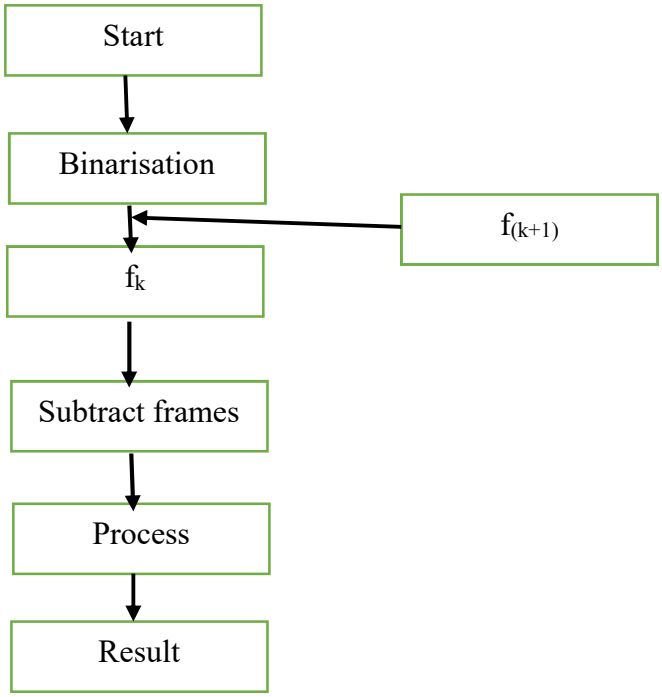
Background subtraction is generally based on a static background hypothesis which is often not applicable in real environments. With indoor scenes, reflections, or animated images on screens lead to background changes. Similarly, due to wind, rain, or illumination changes brought by weather, static backgrounds methods have difficulties with outdoor scenes.

Frame difference:

|frame(i) – frame(i+1)| <Th

The estimated background is just the previous frame. It works on the particular condition of object speed and frame rate. It is very sensitive to the threshold.

It is used to visualize the moving object in a sequence of frames. For still activities frame subtraction method is used. After detecting the human body from frames, we need to remove the still background. To do that any RGB frame is converted to a binary image, then any frame is subtracted from its previous frame to get the foreground moving part. And the centroid is calculated.



Frame1

Frame2

Frame subtraction=frame1-frame2

Removal of unwanted small objects:

After frame subtraction, the subtracted binary image contains some unwanted small objects in the background. These small objects create a problem to calculate the centroid of the foreground object. In this project, we have used the following procedure.

- Binary area open: It removes all unwanted blobs. It removes all connected components that have less than a predefined number of pixels, producing another binary image. The default connectivity is 8 for 2-D and 26 for 3D.
- Algorithm: the basic steps are
  1) Determine the connected components
     L=bwlabel(BW,CONN);
  2) Compute the area of each component
     S=regionprops(L,'Area');
  3) Remove all small objects
     BW2= ismember(L,find([S.Area]>=P));

***Edge Detection*:**

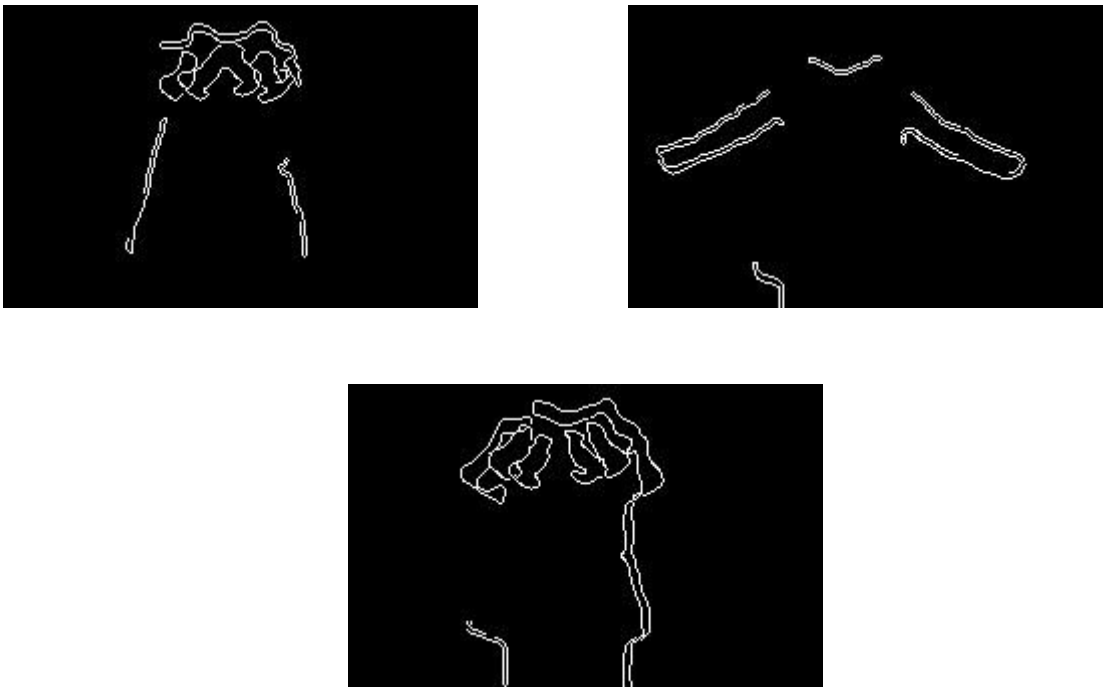
Edge detection is an image processing technique for finding the boundaries of objects within images. It works by detecting discontinuities in brightness. Edge detection is used for image segmentation and data extraction in areas such as image processing, computer vision, and machine vision. Common edge detection algorithms include Sobel, Canny, Prewitt, Roberts, and fuzzy logic methods.

• Definition of edges - Edges are significant local changes of intensity in an image. - Edges typically occur on the boundary between two different regions in an image

• Goal of edge detection - Produce a line drawing of a scene from an image of that scene. - Important features can be extracted from the edges of an image (e.g., corners, lines, curves). - These features are used by higher-level computer vision algorithms (e.g., recognition).

• Edge descriptors Edge normal: unit vector in the direction of maximum intensity change. Edge direction: unit vector to perpendicular to the edge normal. Edge position or center: the image position at which the edge is located. Edge strength: related to the local image contrast along the normal

• The four steps of edge detection (1) Smoothing: suppress as much noise as possible, without destroying the true edges. (2) Enhancement: apply a filter to enhance the quality of the edges in the image (sharpening). (3) Detection: determine which edge pixels should be discarded as noise and which should be retained (usually, thresholding provides the criterion used for detection). (4) Localization: determine the exact location of an edge (sub-pixel resolution might be required for some applications, that is, estimate the location of an edge to better than the spacing between pixels). Edge thinning and linking are usually required in this step



Detected edges of the people

Centroids:

In mathematics and physics, the **centroid** or **geometric center** of a plane figure is the arithmetic mean position of all the points in the figure. Informally, it is the point at which a cutout of the shape could be perfectly balanced on the tip of a pin.centroid of an object in the labeled image: take the mean of all the coordinates of this tagged object both x and y coordinates, call this centroid (this is how regionprops in Matlab does)

The centroid of a shape is the arithmetic mean (i.e., the average) of all the points in a shape. Suppose a shape consists of n distinct points $\mathbf{x_1} \cdots \mathbf{x_n}$, then the centroid is given by

$$\mathbf{c} = \frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i$$

In the context of image processing and computer vision, each shape is made of pixels, and the centroid is simply the weighted average of all the pixels constituting the shape.

To calculate the centroid of an object aq rectangular box is created around the box which is known as boundary boxing. Calculation of centroid from the image as follows:

- Label connected component in a binary image: it is used to compute a matrix of the same size as the input image. The matrix contains labels for the connected objects in the image. The number of connected objects is by default 8. The elements of the matrix are integer values greater than or equal to 0. The pixels labeled 0 are the background. The pixels labeled 1 make up one object; the pixels labeled 2 make up the second object, and so on.

Algorithm:

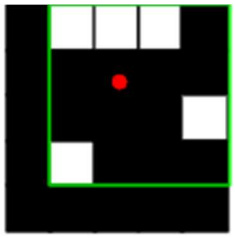
1) Run length encodes the input image.
2) Scan the runs assigning preliminary labels and recording label equivalences in a local equivalence table.
3) Resolve the equivalence classes.
4) Relabel the runs based on the resolved equivalenced classes.

Measure properties of image regions:

Image region property analysis: Step 1: Read the image file. Step 2: Convert the RGB image into Grayscale image. Step 3: Calculate the region property using regionprops function. STATS = regionprops(L,properties) Step 4 : It is also done for the complement image. The regionprops command measures object or region properties in an image and returns those in a structure array. When applied to an image with labeled components, it creates one structure element for each component.

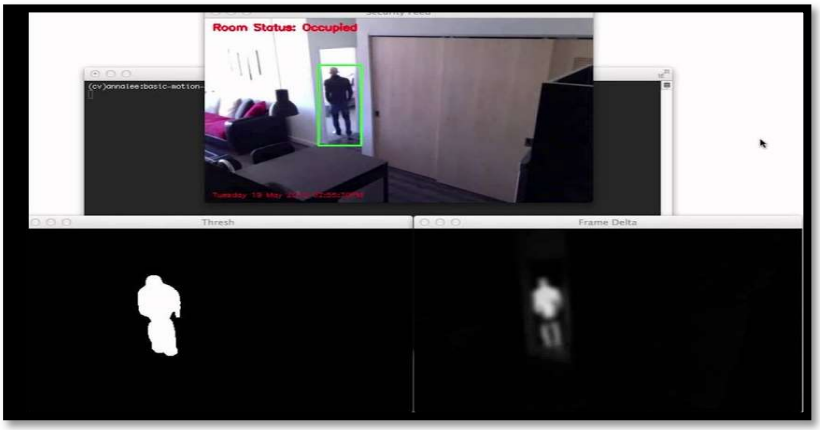
This example uses regionprops to create a structure array containing some basic properties for labeled. When you set the properties parameter to 'basic', the regionprops function returns three commonly used measurements: area, centroid (or center of mass), and bounding box.

It measures a set of properties for each labeled region in the label matrix calculated above for each 8-connected component in the binary image. Position integer elements of the label matrix correspond to the different areas. Centre of the mass of the region is returned as a 1-by-Q vector. The first element of the centroid is the horizontal coordinate of the center of mass. The second element of the centroid is the vertical coordinate of the center of mass. All other elements of the centroid are in order of dimension. This figure illustrates the centroid and bounding box. The region consists of the white pixels; the green box is the bounding box, and the red dot is the centroid.



Boundary Boxing:

BBOX is the rectangle surrounds the foreground silhouette, which is represented by(X,Y, DX, DY) Where (X.Y), DX, DY are the top left point, the width, and height of the silhouette respectively.



Bounding box around the detected people

Velocity Calculation:

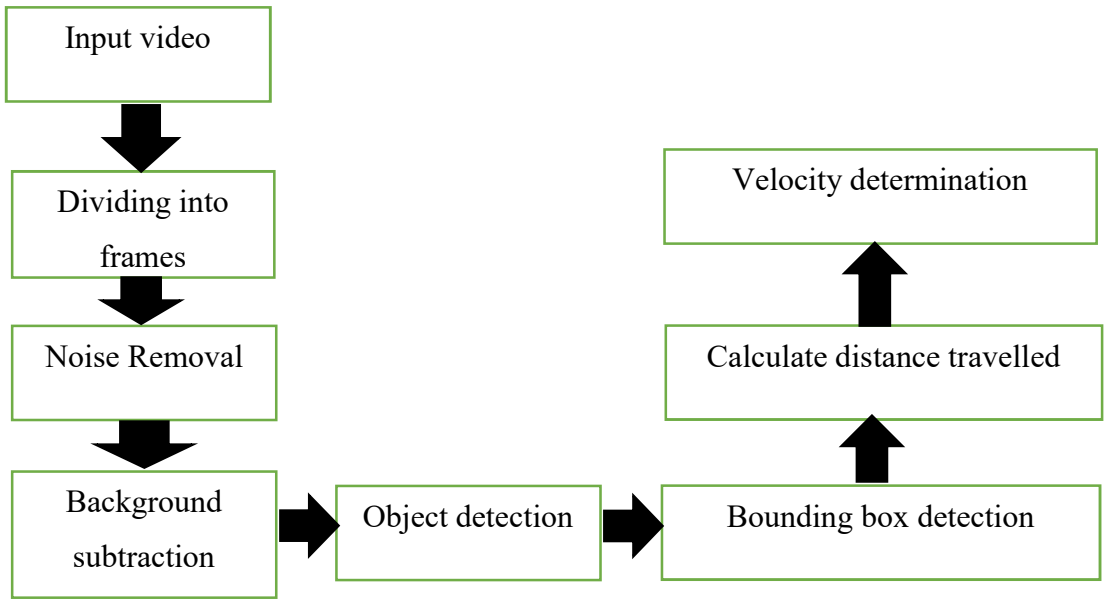
To separate “running” and “walking” activity velocity is calculated. Generally in running activity velocity of a person is higher than walking of that person. To calculate the velocity two things are calculated first,

1)Distance covered by a person from one side of the frame to another side.

2)In how many frames the above distance is covered.

By using the values of distance with respect to frame rate, the velocity of the object is defined. The velocity of moving object is determined using the distance traveled by the centroid to the frame rate of the video. The speed of moving object in the sequence frames is defined in pixels/second.

The following figure shows the velocity calculation process:



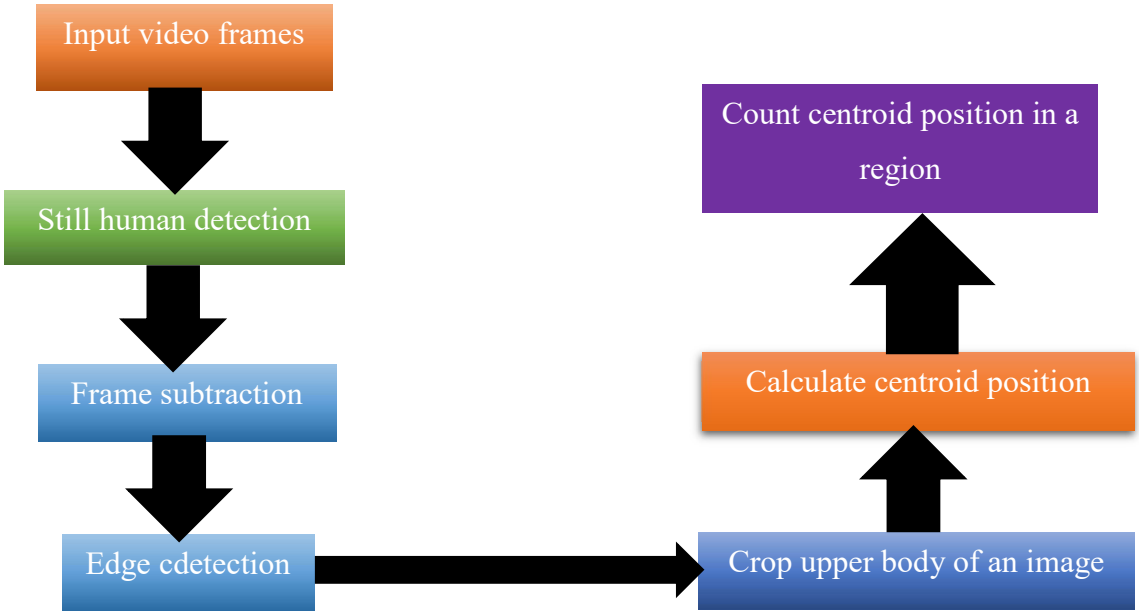
Flowchart of object determination

Body Stretchiness:

To differentiate another moving activity “ jumping” from “running and walking activities body stretchiness feature is calculated. This feature is calculated from previously saved boundary boxing of an object from the co-ordinate of the top left corner of the rectangular boundary and its width towards the axis body stretchiness is calculated.

Upper body mass change

To separate two activities “hand waving” and “hand clapping” this feature is extracted. From the detection of the whole human body, only the upper portion of the body is identified and from that how many times the centroid is placed within that particular region is calculated and used for classification.



Chapter 4

HUMAN ACTIVITY CLASSIFICATIONS

Hear a total of five types of activity is classified. The events are running walking, jumping, hand clapping.

SEPARATE MOVING AND STILL ACTIVITIES

Since the Gaussian mixture, model to identify the still person so that first moving and still activities are separated by calculating several frames identified for human detection by a Gaussian mixture model. If the number is less then ten the activity as in any video number of frames is more then 30.

Separate "running" from "walking."

Since the velocity for the running of a person is comparatively higher than walking of that person. So the velocity is calculated from the sample video set, and a threshold is determined from that to classify "running" and " walking'' activity.

dataset	Name	No of frames to cross one side to another	Velocity= distance/num
Weizmann	Moshe	20	7.5198
Weizmann	Sahar	22	6.6918
KTH	Person 1	14	10.6225
KTH	Person 2	19	7.646
KTH	Person 3	12	11.8199
KTH	Person 4	14	10.3255
KTH	Person 5	18	8.2296
KTH	Person 6	33	4.0282
KTH	Person 7	26	5.7572
KTH	Person 8	32	4.6281

Result of sample data of "running."

dataset	Name	No of frames to cross one side to another	Velocity= distance/num
Weizmann	Moshe	37	4.05
Weizmann	Sahar	40	3.8173
KTH	Person 1	33	4.4574
KTH	Person 2	48	3.17
KTH	Person 3	33	4.5782
KTH	Person 4	44	3.5211
KTH	Person 5	28	5.3168
KTH	Person 6	49	1.5059
KTH	Person 7	36	4.1217
KTH	Person 8	43	3.5118

Result of sample data of "walking."

separate "Handclapping" from "hand waving.''

Depending on the centroid position on upper body mass, these two still activities are recognized. Generally, in "hand waving" number of time occurred of the centroid in the upper portion of the human body is more than "Hand Clapping."From the above sample set threshold is calculated, and the threshold is 30 to 68 for hand waving otherwise hand clapping. Thus all human activities are classified.

dataset	Name	Handwaving	Handclapping
KTH	Person-activity 1	33.3333	15
KTH	Person-activity 2	36.48	9.901
KTH	Person-activity 3	39.53	4.8387
KTH	Person-activity 4	9.18	14.85
KTH	Person-activity 5	47.94	17.82
KTH	Personv-activity 6	37.62	8.91
KTH	Person-activity 7	47.52	5.94
KTH	Person-activity 8	47.36	10.89

Still actions on KTH dataset

Chapter 5

RESULT

Both KTH and Weizmann datasets are used for this project. There are 100 to 110 videos for each activity. Out of 110 videos, 8 to 10 videos are used to trained and remaining are used for testing.

The class of actions includes walking, running, hand waving, hand clapping, jumping according to our consideration.

Datasets

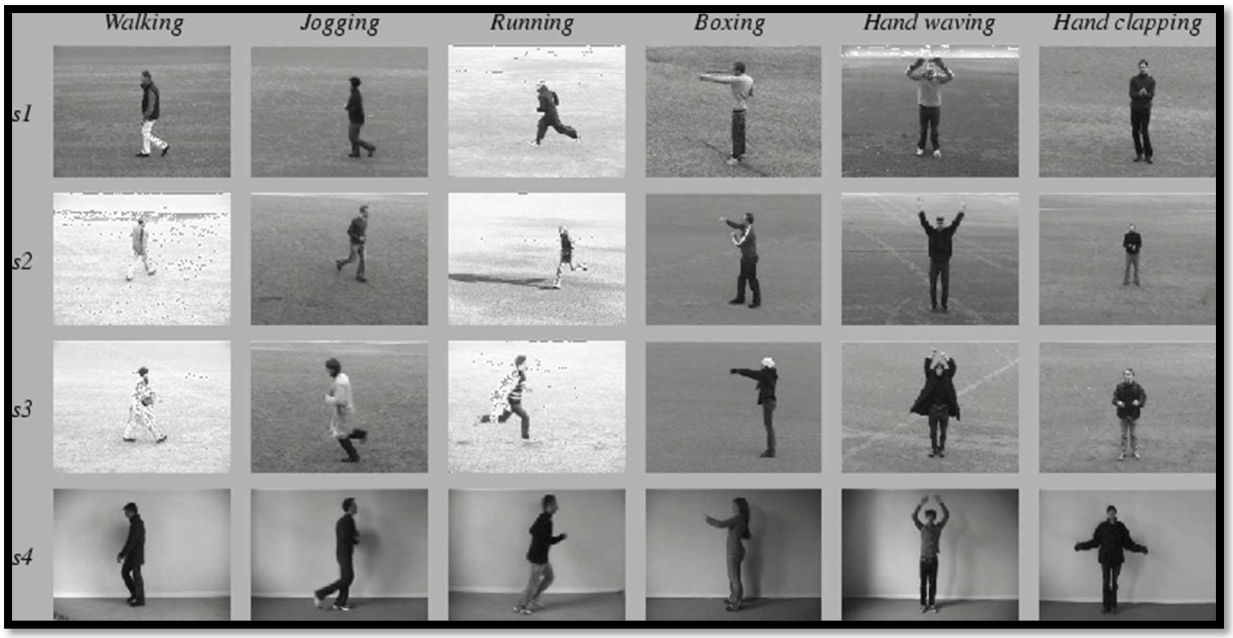
Weizmann[14] and KTH[15] datasets are used for this project

Weizzman dataset

This is very common dataset; many state-of-the-art approaches use the dataset for their purpose and evaluate performance on it; which allows easy comparison. The database contains 90 low-resolution(180*144, deinterlaced 5fps) video sequences showing 9 different people each performing 10 natural actions as “run”, “walk”, “skip”, “jumping back”,”gallop sideways”, “wave two hands”, “wave one hand”, “ bend”.

KTH dataset

This database is also widely used for Human Activity Recognition. The current video database video contains six types of human actions(walking, running, hand waving, hand clapping, jumping.) performed several times by 25 subjects in four different scenarios. Currently, the database contains 2391 sequences. All sequences are taken over the homogeneous background with a static camera with 25fps frame rate. The sequences are downsampled to the spatial resolution 160*120 pixels and have a length of 4 seconds on average; all sequences are stored in AVI file format.



Different types of activities of KTH dataset

Evaluation Strategy

For evaluation of the performance of the proposed methodology, confusion matrix, and Misclassification rate are used.

Confusion matrix:

In the field of machine learning and precisely the problem of statistical classification, a confusion matrix, also known as an error matrix. A confusion matrix is a table that is often used to describe the performance of a classification model (or "classifier") on a set of test data for which the true values are known. It allows the visualization of the performance of an algorithm. It allows easy identification of confusion between classes, e.g., one class is commonly mislabeled as the other. Most performance measures are computed from the confusion matrix.

It is a particular type of contingency table, with two dimensions (" actual" and "predicted") and identical sets of "classes" in both dimensions. For example in table 2 out of 10: hand waving " vides of Weimann dataset, our method recognize eight videos as "hand waving," and other two videos are "hand clapping." In the case of Weizmann dataset, we have confused matrix only for four activities 'running', 'walking,' 'hand waving,' 'hand clapping.' Here 'hand

waving' is confused with 'hand clapping.' On the other side, for KTH dataset 'hand waving' is confused with 'hand clapping' and 'running' confused with 'walking.'

Our main objective was to classify human actions in KTH [1] data set. Actions involve boxing, clapping, walking, running, jogging, and hand waving. Some of the actions in this database are very closely related actions like jogging, running, waiving, and clapping. This data set is a very popular benchmark used in many action recognition papers. The confusion matrix for this experiment is shown in the following tables.

ACTION	RUN	WALK	HAND WAVING	HAND CLAPPING
RUN	0.92	0.03	0.0	0.0
WALK	0.08	0.97	0.0	0.0
HAND WAVING	0.0	0.0	0.78	0.21
HAND CLAPPING	0.0	0.0	0.22	0.79

Confusion matrix on KTH dataset

ACTION	RUN	WALK	HAND WAVING	HAND CLAPPING
RUN	1.0	0.0	0.0	0.0
WALK	0.0	1.0	0.0	0.0
HAND WAVING	0.0	0.0	1.0	0.0
HAND CLAPPING	0.0	0.0	0.0	1.0

Confusion matrix on Weizmann dataset

Misclassification rate:

The misclassification error refers to the number of an individual that we know that bellow to a category that is classified by the method in a different category.

We usually do not try to model misclassification we try to minimize it, and the best method depends on the problem and data type that allow you to minimize the misclassification error.

Let X be a feature space with a finite number of elements. Moreover, let C be a set of classes. Let y:X→ C be a classifier, and let c be the target concept to be learned. Then the true misclassification rate denoted as Err * (y) and defined as:

Err * (y) =|{x € X : y(x) ≠ c(x)}|/|X|

Our proposed approach is entirely dependent on the correctness of segmentation and foreground detection. If GMM is unable to detect moving human, then the misclassification rate increases. On the other hand, if people detector is unable to detect the boundary boxing around the human body, then the misclassification rate will increase for activity 'hand waving' and 'hand clapping.' In Weizmann dataset misclassification rate between 'hand waving' and 'hand clapping' is 5.56% and in KTH dataset misclassification rate is 13.5%.

Analysis

Confusion matrix assures the effective accuracy rate of our proposed technique. The misclassification has taken place in case of both 'running'. 'walking,' and 'hand waving' and 'hand clapping' for KTH dataset. And Weizmann dataset is only occurred for 'hand clapping' and 'hand waving.' Overall performance is 94.44% for Weizmann dataset and 90.6 for KTH dataset.

Chapter 6

Conclusion and future work

In this chapter, we look back at the problem defined in the introduction and summarize the work done in this project towards solving it. Conclusions are drawn from the work; mainly the result presented in the last chapter.

Conclusion

This thesis has shown a combination of feature extraction, evaluation, and selection can work together to provide a high-quality dataset for use with an inference engine. The proposed method is based on two significant steps. In the first step, multiple humans are detected in the given video sequences. Then, we extracted the texture and shape features from given sequences and fused them based on vector dimensions.

Feature extraction is not necessarily a difficult problem . in fact, in feature evaluation step has shown many of the best features turn out to be model data or primary functions of it. For some activities, more complex features are needed. To summarize we may draw the following conclusions—

- Feature extraction is effective and relatively simple.
- Feature evaluation can be done automatically by statistical means.
- Feature subset selection is sensible and useful.
- Extraction, Evaluation, Selection can work together.

Especially notable is how well the combination of these works. With careful selection, even simple features may contribute significantly to recognize results.

Future work

Finally, human activity recognition tasks constitute the foundation of human behavior understanding, which requires additional contextual information such as W5+ (who, where, what, when, why, and how) [144]. The same activity may have different behavior interpretations depending on the context in which it is performed. More specifically, the “where” (place) context can provide the location information to be used to detect abnormal behaviors. For example, lying down on the bed or a sofa is interpreted as taking a rest or sleeping, but in inappropriate places such as the floor of the bathroom or kitchen, it can be interpreted as a fall or a sign of a stroke.

Moreover, the “when” (time) context also plays another critical contextual role for behavior understanding. For example, a person usually watching TV after midnight can be regarded as an insomniac.

Another example is that a person will be detected as picking something up if he/she squats and stands up quickly. However, if he/she squats for a more extended period, there might be a motion difficulty due to osteoarthritis or senility.

Furthermore, the number of repetitions of action can also be informative. For example, eating too many times or too little a day can be an early symptom of depression. The interaction between people or between person and objects is also a good indicator to identify the meaning of the activity. For example, if a person is punching a punch-bag, he might be doing exercise. But if he is punching the wall, it can indicate anger or a mental disorder.

In conclusion, this review provides an extensive survey of existing research efforts on video-based human activity recognition systems, covering all critical modules of these systems such as object segmentation, feature extraction and representation, and activity detection and classification.

Moreover, three application domains of video-based human activity recognition are reviewed, including surveillance, entertainment, and healthcare. In spite of the great progress made on the subject, many challenges are raised herein together with the related technical issues that need to be resolved for real-world practical deployment.

Furthermore, generating descriptive sentences from images [145] or videos is a further challenge, wherein objects, actions, activities, environment (scene) and context information are considered and integrated to generate descriptive sentences conveying key

The feature extraction step can be expanded almost without limits. It is indeed possible to apply more advanced statistical means to extract more complex features. However, it should be kept in mind that the evaluation tends to suggest that complex features are not generally excellent. The single point of improvement upon the recognition results could well be using an inference engine with intrinsic time modeling, for instance, a state shape model such as the Hidden Markov Model.

REFERENCES

[1] H. Ye, T. GU, X. Zhu, J. Xu, X. Tao, J. Lu, and N. Jin, Track: Infrastructure- free floor localization via mobile phone sensing, in Proc. Int. Pervasive Computing and CommunicationsConf.,Lugano, Switzerland, 2012, pp.

[2] A. Ofstad, E. Nicholas, R. Szcodronski, and R. R. Choudhury, Aampl: Accelerometer augmented mobile phone localization, in Proc. 1st Int. Mobile Entity Localization and Tracking in GPS-Less Environments Workshop, San Francisco, USA, 2008, pp. 13-18.

[3] S. Kozina, H. Gjoreski, M. Games, and M. Luˇstrek, Efficient activity recognition and fall detection using accelerometers, in Evaluating AAL Systems Through Competitive Benchmarking, Springer, 2013, pp. 13-23.

[4] Fitbit, Sensors Overview, http://www.fitbit.com/one, 2014, Mar. 17.

[5] A. Avci, S. Bosch, M. Marin-Perianu, R. Marin- Perianu, and P. Havinga, Activity recognition using inertial sensing for healthcare, wellbeing and sports applications: A survey, in Proc. 23rd Int. Architecture of Computing Systems Conf., Hannover, Germany, 2010, pp. 1-10.

[6] J. W. Lockhart, T. Pulickal, and G. M.Weiss, Applications of mobile activity recognition, in Proc. 14th Int. Ubiquitous Computing Conf., Seattle, USA, 2012, pp. 1054-1058.

[7] O. D. Incel, M. Kose, and C. Ersoy, A review and taxonomy of activity recognition on mobile phones, BioNanoScience, vol. 3, no. 2, pp. 145-171, 2013.

[8] O. D. Lara and M. A. Labrador, A survey on human activity recognition using wearable sensors, Communications Surveys & Tutorials, IEEE, vol. 15, no. p. 1192-1209

[9] N. D. Lane, E. Miluzzo, H. Lu, D. Peebles, T. Choudhury, and A. T. Campbell, A survey of mobile phone sensing, Communications Magazine, IEEE, vol. 48, no. 9, pp. 140-150, 2010.
[10] H. F. Rashvand and K. -F. Hsiao, Smartphone intelligent applications: A brief review, Multimedia Systems, pp. 1- 17, 2013.

[11] H. Becker, M. Borazio, and K. Van Laerhoven, How to log sleeping trends? A case study on the long-term capturing of user data, in Smart Sensing and Context, Springer, 2010, pp. 15-27.

[12] Google Android, Sensors Overview, http://developer. android.com/guide/topics/sensors/sensors overview.html, 2014, Mar. 01.

[13] L. Bao and S. S. Intille, Activity recognition from user annotated acceleration data, in Pervasive Computing, Springer, 2004, pp. 1-17.

[14] Y. E. Ustev, O. Durmaz Incel, and C. Ersoy, User, device and orientation independent human activity recognition on mobile phones: Challenges and a proposal, in Proc. 13th Int.

Pervasive and Ubiquitous Computing Adjunct Publication Conf., Zurich, Switzerland, 2013, pp. 1427- 1436.

[15] Y.-S. Lee and S.-B. Chou, Activity recognition using hierarchical hidden Markov models on a smartphone with 3D accelerometer, in Hybrid Artificial Intelligent Systems, Springer, 2011, pp. 460-467.

[16] N. Ravi, N. Dandekar, P. Mysore, and M. L. Littman, Activity recognition from accelerometer data, AAAI, vol. 5, pp. 1541-1546, 2005.

[17] J. R. Kwapisz, G. M. Weiss, and S. A. Moore, Cell phone based biometric identification, in Proc. 4th Int. BiometricTheory, Applications and Systems Conf., Washington, DC, USA, 2010, pp. 1-7.

[18] J. G. Casanova, C. S. ´A Vila, A. De Santos Sierra, G.B. Del Pozo, and V. J. Vera, A real-time in- air signature biometric technique using a mobile

[19] A. Bulling, U. Blanke and B. Schiele, "A tutorial on human activity recognition using body-worn inertial sensors", ACM Computing Surveys (CSUR), vol. 46, no. 3, pp. 1-33, 2014.
[20] 2. O. D. Lara and M. A. Labrador, “A survey on human activity recognition using wearable sensors,” IEEE Commun. Surveys Tuts., vol. 15, no.

https://www.geeksforgeeks.org/confusion-matrix-machine-learning/

http://www.rroij.com/open-access/human-activity-recognition-challenges-and-process-stages-.pdf

https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_imgproc/py_contours/py_contour_features/py_contour_features.html

https://stackoverflow.com/questions/25587900/determine-the-minimum-bounding-box-to-capture-an-object-in-matlab

https://in.mathworks.com/help/vision/ref/peopledetectoracf.html

https://in.mathworks.com/matlabcentral/answers/115406-human-detection-in-a-frame

https://in.mathworks.com/help/vision/ref/vision.foregrounddetector-system-object.html

https://in.mathworks.com/help/vision/examples/tracking-cars-using-foreground-detection.html

https://in.mathworks.com/matlabcentral/answers/169092-how-to-choose-the-parameters-of-vision-foregrounddetector

https://in.mathworks.com/help/vision/ref/vision.blobanalysis-system-object.html

https://in.mathworks.com/matlabcentral/answers/232199-how-to-use-vision-blobanalysis-to-represent-regionprops

https://in.mathworks.com/help/vision/ref/blobanalysis.html

www.wikipedia.com