Title: HUMAN MOTION DETECTION
Author: Ashok. Koujalagi
THE OXFORD COLLEGE OF SCIENCE
(Affiliated to Bangalore University)
HSR Layout
BANGALORE, INDIA – 560102

DEPARTMENT OF COMPUTER SCIENCE AND APPLICATIONS
JULY 2013

THESIS REPORT
ON
“HUMAN MOTION DETECTION”
(SUBMITTED FOR PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF MASTER OF SCIENCE, IN COMPUTER SCIENCE
DEPARTMENT OF COMPUTER SCIENCE & APPLICATION)
Under the Guidance of

Dr. JAYAPRAKASH
(Assistant professor)

By
ASHOK. KOUJALAGI (Author)
E-mail: ashok.koujalagi@yahoo.com
ACKNOWLEDGEMENT

I am faithful towards the Almighty who give me courage and help me to take right decision and work towards the ultimate aim.

It is my pleasure to express with deep sense of gratitude to Dr. Jayaprakash, project Guide, Department of Computer Science for her constant guidance, continual encouragement, understanding; more than all, continuous support.

I would like to express my gratitude to Dr. SEBASTIAN NIXON HOD, Department of Computer Science, The Oxford College Of Science, for providing an environment to work in and for his inspiration during the tenure of the course.

In jubilant mood I express ingeniously my whole-hearted thanks to all teaching staff and members working as limbs of our computer science department for their timely encouragements, which prompted the acquirement of the requisite knowledge to finalize my course successfully.

I would like to thank my parents for their endless support and encourage throughout my life. It is indeed a pleasure to thank my friends who persuaded and encouraged me to take up and complete this task.

At last but not least, I express my gratitude and appreciation to all those who have helped me directly or indirectly towards the successful completion of this thesis work.

ABSTRACT

Human motion recognition is currently one of the most interesting and active research topics. Motion segmentation is an essential process in Human motion detection. In this thesis the motion segmentation problem is studied and methods that can be used to solve the problem is identified and the advantages and disadvantages are extracted, which helps to obtain the better method for motion segmentation. The objective of this paper is to find out the best method for motion segmentation and to design the solution in order to fulfill the drawbacks of background subtraction method. This paper helps the researchers to easily find out the best method for motion segmentation, in which the time can be saved in identification of the suitable method as well as searching the solution for the problem. This paper also provides the applications of Human motion detection. Finally, further research directions are suggested.

Keywords: Human motion detection, Motion segmentation, Background subtraction method
Glossary

HMD- Human motion detection
Bgs- Background subtraction
Gmm- Gaussian mixture model

List of Figures

1. Classification of motion detection 19
2. Approaches of Motion Segmentation 20
3. The Optical Flow vector of a moving object in video sequence 32

List of Tables

1. Advantages and Disadvantages of Background Subtraction method 42
2. Advantages and Disadvantages of Optical Flow Method 43
3. Advantages and Disadvantages of Temporal Differencing 44
4. Advantages and Disadvantages of Statistical Methods 45
# Table of Content

<table>
<thead>
<tr>
<th>Name</th>
<th>Page No</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Introduction</td>
<td></td>
</tr>
<tr>
<td>1.1 General Introduction</td>
<td>1</td>
</tr>
<tr>
<td>1.2 State of the problem</td>
<td>4</td>
</tr>
<tr>
<td>1.3 Objective of the problem</td>
<td>4</td>
</tr>
<tr>
<td>1.4 Scope of the paper</td>
<td>5</td>
</tr>
<tr>
<td>1.5 Outline of the thesis</td>
<td>5</td>
</tr>
<tr>
<td>2. Literature survey</td>
<td>6</td>
</tr>
<tr>
<td>2.1 Motion Segmentation</td>
<td></td>
</tr>
<tr>
<td>2.1.1 Approaches of Motion Segmentation</td>
<td>7</td>
</tr>
<tr>
<td>2.2 Object classification</td>
<td></td>
</tr>
<tr>
<td>2.2.1 Shape-based classification</td>
<td>11</td>
</tr>
<tr>
<td>2.2.2 Motion-based classification</td>
<td>12</td>
</tr>
<tr>
<td>2.3 Tracking</td>
<td>12</td>
</tr>
<tr>
<td>2.4 Behavior Understanding</td>
<td></td>
</tr>
<tr>
<td>2.4.1 General Techniques</td>
<td>15</td>
</tr>
<tr>
<td>2.4.2 Action Recognition</td>
<td>16</td>
</tr>
<tr>
<td>2.5 Application of Human Motion Detection</td>
<td>17</td>
</tr>
<tr>
<td>3. Motion Segmentation</td>
<td>18</td>
</tr>
<tr>
<td>3.1 Methods for Motion Segmentation</td>
<td>20</td>
</tr>
<tr>
<td>4. Background Subtraction Method</td>
<td>22</td>
</tr>
<tr>
<td>4.1 Introduction</td>
<td></td>
</tr>
<tr>
<td>4.2 Background Subtraction Method Blocks</td>
<td>24</td>
</tr>
<tr>
<td>4.3 Method Overview</td>
<td>25</td>
</tr>
<tr>
<td>4.4 Applications of Background Subtraction Method</td>
<td>26</td>
</tr>
<tr>
<td>4.5 Requirements</td>
<td>27</td>
</tr>
<tr>
<td>4.6 Advantages and Disadvantages</td>
<td>27</td>
</tr>
<tr>
<td>5. Optical Flow</td>
<td>28</td>
</tr>
<tr>
<td>5.1 Introduction</td>
<td></td>
</tr>
<tr>
<td>5.2 Optical Flow in Motion analysis</td>
<td>29</td>
</tr>
<tr>
<td>5.3 Optical Flow computation</td>
<td>30</td>
</tr>
<tr>
<td>5.4 Methods for determining optical Flow</td>
<td>31</td>
</tr>
<tr>
<td>5.5 Uses of Optical Flow</td>
<td>31</td>
</tr>
<tr>
<td>5.6 Applications of Optical Flow</td>
<td>34</td>
</tr>
<tr>
<td>6. Temporal Differencing</td>
<td>36</td>
</tr>
<tr>
<td>6.1 Introduction</td>
<td></td>
</tr>
<tr>
<td>6.2 Advantages of Temporal Differencing Method</td>
<td>37</td>
</tr>
<tr>
<td>6.3 Disadvantages of Temporal Differencing Method</td>
<td>37</td>
</tr>
<tr>
<td>7. Statistical Method</td>
<td>38</td>
</tr>
<tr>
<td>7.1 Introduction</td>
<td>38</td>
</tr>
<tr>
<td>7.2 Basic steps of a Statistical Method</td>
<td>40</td>
</tr>
<tr>
<td>7.3 Applications</td>
<td>41</td>
</tr>
<tr>
<td>8. Advantages and Disadvantages of Background Subtraction Method</td>
<td>42</td>
</tr>
<tr>
<td>9. Advantages and Disadvantages of Optical Flow Method</td>
<td>43</td>
</tr>
<tr>
<td>10. Advantages and Disadvantages of Temporal Differencing</td>
<td>44</td>
</tr>
<tr>
<td>11. Advantages and Disadvantages of Statistical Method</td>
<td>45</td>
</tr>
<tr>
<td>12. The Basic Background Subtraction Method</td>
<td>46</td>
</tr>
<tr>
<td>13. Proposed Method</td>
<td>48</td>
</tr>
<tr>
<td>14. Conclusion and Future enhancement</td>
<td>50</td>
</tr>
<tr>
<td>Bibliography</td>
<td></td>
</tr>
<tr>
<td>Appendix</td>
<td></td>
</tr>
</tbody>
</table>
1. INTRODUCTION

1.1 GENERAL INTRODUCTION

Human motion detection is the topic under object recognition, in computer vision, this is the task of finding a given object in an image or video sequence. Humans recognize a multitude of objects in images with little effort, despite the fact that the image of the objects may vary somewhat in different view points, in many different sizes / scale or even when they are translated or rotated. Objects can even be recognized when they are partially obstructed from view. This task is still a challenge for computer vision systems in general. One of the simplest type of motion identification is to detect image points that refer to moving points in the scene. Human motion detection involves motion segmentation, object classification, tracking and behavior understanding. This paper describes the motion segmentation, where it involves in the identification of the best method for motion segmentation. The field of computer vision is concerned with problems that involve interfacing computer with their surrounding environment through visual means. One such problem, object recognition, involves detecting the presence of a known object in an image, given some knowledge about what that object should look like [4]. As humans, we take this ability for granted, as our brains are extraordinarily proficient at both learning new objects and recognizing them later. However, in computer vision, this same problem has proven to be one of the most difficult and computationally intensive of the field. Given the current state of the art, successful algorithm for object recognition requires one to define the problem with a more specific focus. In this thesis, we consider a sub-problem of object recognition: human motion detection, in which we are interested in recognizing humans based solely on the characteristic patterns of motion that they exhibit [24]. Its characteristics include real-time performance, insensitivity to background clutter and movement, and a modular design that can be generalized to other types of motion. The motion analysis processing can in the simplest case be to detect motion, i.e., find the points in the image where something is moving. More complex types of processing can be to track a specific object in the image and over time, to group points that belong to the same rigid object that is moving in the scene, or to determine the magnitude and direction of the motion of every point in the image. The information that is produced is often related to a specific image in the sequence, corresponding to a specific time-point, but then depends also on the neighboring images. This means that motion analysis can produce time time-dependent information about motion. Applications of motion analysis can be found in rather diverse areas, such as surveillance, medicine, film industry, and navigation of autonomous vehicles. HMD detection includes the following sections namely motion detection, tracking and understanding as shown in the figure below:

HMD consists of following sections: Motion segmentation, Object classification, Tracking and behavior understanding. This paper presents the motion segmentation with its methods. Motion Segmentation is a means of separating moving objects in an image from a static background. The background image is accumulated over multiple images, and the invariant values of each pixel are taken to be the background either after a certain number of frames or as a median value for the pixel over say 20 frames. An object which moves dynamically from frame to frame will change the value of pixels in the image and obscure the background which it moves across and an outline of the changed values can be used to segment out moving objects from the non moving background.

Motion segmentation has some methods namely background subtraction, optical flow, temporal differencing and statistical method. The main aim of this paper is to find out the best method among the mentioned above by considering its advantages and disadvantages.

Background subtraction method: The idea is to subtract the current image from a reference image, which is acquired from a static background during a period of time. The subtraction leaves only the non-stationary or new objects [1]. The technique has been used for years in many vision systems as a preprocessing step for object detection and tracking.

Optical flow: Optic flow is the perceived visual motion of objects as the observer moves relative to them. This allows to judge how close are to certain objects, and how quickly approaching them. It is also useful for avoiding obstacles: if an object in front is expanding but not moving, probably headed straight for it, but if it is expanding but moving slowly to the side, it will probably pass by it [24].

Temporal differencing: One direct method of motion detection is to use temporal difference where the absolute difference at each pixel between two or three consecutive Frames are calculated and a threshold is applied to get the
motion object [17]. Although this method is simple to implement it is not so effective to get the whole region of the moving object, especially the inner part of the moving object [18].

Statistical method: Statistic theory is widely used in the motion segmentation field. In fact, motion segmentation can be seen as a classification problem where each pixel has to be classified as background or foreground [15]. Statistical approaches can be further divided depending on the framework used. Statistical approaches provide a general tool that can be used in a very different way depending on the specific technique [16].

This thesis helps to easily recognize the best method for motion segmentation. Visual surveillance in dynamic scenes, especially for humans and vehicles, is one of the current challenging research topics in computer vision. It is a key technology to fight against terrorism, crime, public safety and for efficient management of traffic. The work involves designing of efficient visual surveillance system in complex environments. In video surveillance, detection of moving objects from a video is important for object classification, target tracking, activity recognition, and behavior understanding. Detection of moving objects in video streams is the first relevant step of information and background subtraction is a very popular approach for foreground segmentation. In this thesis, we have simulated different background subtraction methods to overcome the problem of illumination variation, background clutter, shadows, and camouflage. Object classification is done using silhouette template based classification to categorize objects into human, group of human and vehicle. Detecting and tracking of human body parts is important in understanding human activities, occlusion and. In order to extend the surveillance area and overcome occlusion, fusion of data from multiple cameras is employed in our project. We have tracked objects across multiple cameras with non-overlapping based on object appearances. A brightness transfer function is determined from the cumulative histograms of the images.

1.2 STATEMENT OF THE PROBLEM
The field of computer vision is concerned with problems that involve interfacing computers with their surrounding environment through visual means. One such problem, object recognition, involves detecting the presence of a known object in an image. However, in computer vision, this same problem has proven to be one of the most difficult and computationally intensive of the field. Given the current state of the art, a successful algorithm for object recognition requires one to define the problem with a more specific focus. In this thesis, we consider a sub-problem of object recognition: human motion detection, in which we are interested in recognizing humans based solely on the characteristic patterns of motion that they exhibit. The techniques for human motion detection, such as those that recognize motion segmentation, tracking, object classification, where each one has its own methods to identify. Among those methods particularly in motion segmentation identifying the suitable and best method is the problem of this thesis.

1.3 Objective of the paper
The main objective of this thesis paper is to identify the best method for motion segmentation among some of the methods by considering all its advantages and disadvantages.

1.4 Scope of the paper
This thesis paper presents the best method for motion segmentation, which helps the future developers to easily choose the most suitable method for their thesis work so that their work will be less regarding this issue.

1.5 Outline of Thesis
The Thesis is organized as follows: Chapter Two is Literature Survey which deals with different motion segmentation methodologies and applications. Chapter Three deals with Motion segmentation and its approaches. Chapter four deals with Background Subtraction method and its applications. Chapter five deals with Optical Flow method, its applications and its methods. Chapter six deals with Temporal Differencing and its advantages and drawbacks. Chapter seven deals with Statistical Method and its applications. Chapter eight deals with advantages and disadvantages of Background Subtraction method. Chapter nine deals with advantages and disadvantages of Optical Flow method. Chapter ten deals with advantages and disadvantages of Temporal Differencing. Chapter eleven deals with advantages and disadvantages of Statistical Method. Chapter twelve deals with proposed method to overcome the drawbacks of Background Subtraction Method. Chapter thirteen deals with conclusion and future enhancement.

2. Literature Survey
Human motion detection is one of most interesting research topic focused on building systems for observing humans and understanding their appearance, movements and activities. Many research papers can be noticed in this area. Human motion detection has a wide range of potential applications such as smart surveillance, advanced user interface, motion based diagnosis, to name a few. Nearly every system of vision-based human motion analysis starts with human detection. Human detection aims at segmenting regions corresponding to people from the rest of an image. It is a significant issue in a human motion detection system since the subsequent processes such as tracking and action recognition are greatly dependent on it. This process usually involves motion segmentation and object classification. The initial stage of the extraction of visual information is the detection of moving objects from a video sequence. At the video surveillance researches, background subtraction techniques are mostly used for detection motion in many real-time video surveillance applications. Recently,
there has been a large amount of work addressing the background subtraction, adaptation and background model initialization. Often the assumption is made that an initial background model can be obtained by using a short training sequence in which no foreground objects are present. However, in some monitoring areas, such as public area, crowded corridors, it is difficult or impossible to control the area being monitored. In such cases there may be needed to train the model using a sequence which contains foreground objects. An ideal background subtraction could produce good results while foreground regions in motions during training sequence. There is also needed to maintenance background model to adapt all possible changes in the monitoring area. A large number of people detection and tracking algorithms rely on the process of background subtracting, a technique which detects changes from a model of the background scene. They mainly focused on model representation and techniques for adaptation on background modeling. The ability for representing multiple models to have the background values some techniques to model motion which is part of the background. Unimodal representations, which store a single mean intensity value per pixel, are more prone to have false detections in the case of moving any background regions. The methods in some studies estimate background intensity values using temporal smoothing and choose a single value from the set of past observations. In minimum and maximum intensity values, and maximum temporal derivative for each pixel are stored for initialization background model. The study in proposed an edge-based background representation called the background primal sketch. PFinder uses a unimodal background model to locate interesting objects.

At present, most segmentation methods use either temporal or spatial information of the images. Several conventional approaches to Motion Segmentation, Object Classification, Tracking, Behavior Understanding are outlined in the following.

2.1 Motion Segmentation

In [29] this paper, described that when dealing with motion segmentation often also the object tracking problem (i.e. follow object elements which move across a sequence of images) needs to be solved. Segmentation and tracking are double linked: solving one can help to solve the other and vice versa. Segmentation can be used as initialization for tracking objects, on the other hand tracking single elements of an object (features) can give information on the whole object structure and helps to solve the segmentation problem. Hence, it is not always easy to a clear border between motion segmentation and object tracking literature. For this reason the following motion segmentation review will also include some regions in an image by differencing between current image and a reference background image in a pixel-by-pixel fashion. However, it is extremely sensitive to changes of dynamic scenes due to lighting and extraneous events.

The numerous approaches to this problem differ in the type of a background model and the procedure used to update the background model. The simplest background model is the temporally averaged image, a background approximation that is similar to the current static scene. Most researchers show more interests in building different adaptive background models in order to reduce the influence of dynamic scene changes on motion segmentation. The basic scheme of background subtraction is to subtract the image from a reference image that models the background scene. Typically, the basic steps of the algorithm are as follows:

Background modeling constructs a reference image representing the background.
Threshold selection determines appropriate threshold values used in the subtraction operation to obtain a desired detection rate.
Subtraction operation or pixel classification classifies the types of a given pixel belongs, i.e., the pixel is the part of background (including ordinary background and shaded background), or it is a moving object.[30]

The initial background model is obtained even if there are moving foreground objects in the field of view, such as walking people, moving cars. The basic idea in the background model initialization presented in this study is depending on that stationary pixel intensity value is brightness value which has the highest redundancy ratio on intensity values taken from a training sequence[24]. The difficult part of background subtraction is not the differencing itself, but the maintenance of a background model is some representation of the background and its associated statistics is Background subtraction called to this modeling process. Also, real-time background maintenance is important for real surveillance applications. Because, the monitoring area may not be stay the same for long periods of time. There could be illumination changes, such as the sun being blocked by clouds causing changes in brightness, or physical changes, such as a car that comes into the scene and parks should not be considered as a part of the scene background, however the stationary pixels should play the role of background for detecting motion of a person getting out or passing in front of the car. What an ideal background maintenance system should have and the problems of background scene maintenance for surveillance system are good explained in [29]. In these cases, the notion in this study is to use an adaptive background model to accommodate changes to the background while maintaining the ability to detect independently moving objects. The algorithm presented is based on two processes to update the background model is a pixel based maintenance and a object based maintenance.
Statistical methods
Recently, some statistical methods to extract change regions from the background are inspired by the basic background subtraction methods described above. The statistical approaches use the characteristics of individual pixels or groups of pixels to construct more advanced background models, and the statistics of the backgrounds can be updated dynamically during processing. Each pixel in the current image can be classified into foreground or background by comparing the statistics of the current background model. This approach is becoming increasingly popular due to its robustness to noise, shadow, change of lighting conditions, etc. A recent study built a statistical model by representing each pixel with three values: its minimum and maximum intensity values, and the maximum intensity difference between consecutive frames observed during the training period. Static theory is widely used in the motion segmentation field. In fact, motion segmentation can be seen as a classification problem where each pixel has to be classified as background or foreground. Statistical approaches can be further divided depending on the framework used. Common frameworks are Maximum A posteriori Probability (MAP), Particle Filter (PF) and Expectation Maximization (EM) [29]. Statistical approaches provide a general tool that can be used in a very different way depending on the specific technique.

Temporal differencing
The approach of temporal differencing makes use of pixel-wise difference between two or three consecutive frames in an image sequence to extract moving regions. Temporal differencing is very adaptive to dynamic environments, but generally does a poor job of extracting the entire relevant feature pixels, e.g., possibly generating holes inside moving entities. As an example of this method, the paper detected moving targets in real video streams using temporal differencing. After the absolute difference between the current and the previous frame was obtained, a threshold function was used to determine change. By using a connected component analysis, the extracted moving sections were clustered into motion regions. These regions were classified into predefined categories according to image-based properties for later tracking. An improved version is to use three-frame differencing instead of two-frame differencing. For instance, has successfully developed a hybrid algorithm for motion segmentation by combining an adaptive background subtraction algorithm with a three-frame differencing technique. This hybrid algorithm is very fast and surprisingly effective for detecting moving objects in image sequences.

Optical flow
Flow is generally used to describe coherent motion of points or features between image frames. Motion segmentation based on optical flow uses characteristics of flow vectors of moving objects over time to detect change regions in an image sequence. For example, one of the paper performed monofonic operation which computed the displacement vector field to initialize a contour-based tracking algorithm, called active rays, for the extraction of articulated objects which would be used for gait analysis. The work by another paper also focused on the segmentation of optical flow fields of articulated objects. Its major contributions were to add kinematic motion constraints to each pixel, and to combine motion segmentation with estimation in Expectation Maximization computation. Also, in other paper work, each pixel was represented by its optical flow. These flow vectors were grouped into blobs having coherent motion and characterized by a mixture of multivariate Gaussians. Optical flow methods can be used to detect independently moving objects even in the presence of camera motion. However, most flow computation methods are computationally complex and very sensitive to noise, and cannot be applied to video streams in real-time without specialized hardware. In addition to the basic methods described above, there are some other approaches to motion segmentation. Using the extended EM algorithm, Friedman and Russell implemented a mixture of Gaussian classification model for each pixel. This model attempted to explicitly classify the pixel values into three separate predetermined distributions corresponding to background, foreground and shadow.

Optical Flow is the distribution of apparent velocities of movement of brightness patterns in an image, Optical Flow is an old concept greatly exploited in computer vision. It was first formalized and computed for image sequences by Horn and Schunck in the [29]. However, the idea of using discontinuities in the optical flow in order to segment moving objects is even older, in [28] there is a list of older methods based on this idea but they all assume the optical flow is already known. Since the work of Horn and Schunck, many other approaches have been proposed. In the past the main limitation of such methods was the high sensitivity to noise and the high computational cost. Nowadays, thanks to the high process speed of computers and to improvements made by research, OF is widely used. Optical flow algorithm can work also when the moving object has a homogeneous surface, provided that the object edges can be identified and used as straight lines. The limitation of this method is that is only able to deal with rigid motion because it requires straight lines in order to compute the optical flow. This approach uses K-means in order to build the final clusters hence it assumes the number of moving objects (i.e. the number K of clusters) is known a priori.

2.2 Object Classification
Moving regions detected in video may correspond to different objects in real-world such as pedestrians, vehicles, clutter, etc. It is very important to recognize the type of a detected object in order to track it reliably and analyze its activities correctly. Currently, there are two major approaches towards moving object classification which are shape-based and motion-based methods. Shape-based methods make use of the objects’ 2D spatial information whereas motion-based methods use
temporally tracked features of objects for the classification solution.

2.2.1 Shape-based Classification

Common features used in shape-based classification schemes are the bounding rectangle, area, silhouette and gradient of detected object regions. The approach presented in makes use of the objects’ silhouette contour length and area information to classify detected objects into three groups: human, vehicle and other. The method depends on the assumption that humans are, in general, smaller than vehicles and have complex shapes. Dispersedness is used as the classification metric and it is defined in terms of object’s area and contour. Classification is performed at each frame and tracking results are used to improve temporal classification consistency. Saptharishi et al. propose a classification scheme which uses a logistic linear neural network trained with differential learning to recognize two classes: vehicle and people [25]. Papageorgiou presents a method that makes use of the Support Vector Machine classification trained by wavelet transformed object features (edges) in video images from a sample pedestrian database [26]. This method is used to recognize moving regions that correspond to humans. Another classification method proposed by Brodsky et al. [27] uses a Radial Basis Function (RBF) classifier which has a similar architecture like a three-layer back-propagation network. The input to the classifier is the normalized gradient image of the detected object regions.

2.2.2 Motion-based Classification

Some of the methods in the literature use only temporal motion features of objects in order to recognize their classes [28]. In general, they are used to distinguish non-rigid objects (e.g. human) from rigid objects (e.g. vehicles). The method proposed in [8] is based on the temporal self-similarity of a moving object. As an object that exhibits periodic motion evolves, its self-similarity measure also shows a periodic motion. The method exploits this clue to categorize moving objects using periodicity. Optical flow analysis is also useful to distinguish rigid and non-rigid objects. A. J. Lipton proposed a method that makes use of the local optical flow analysis of the detected object regions [28]. It is expected that non-rigid objects such as humans will present high average residual flow whereas rigid objects such as vehicles will present little residual flow. Also, the residual flow generated by human motion will have a periodicity. By using this cue, human motion, thus humans, can be distinguished from other objects such as vehicles.

2.3 Tracking

The goal of object tracking is to establish correspondence between objects across frames. It is assumed in this study that the regions can enter and exit the scene and they can also get occluded by other regions. Regions carry information like shape and size of the silhouette, and colors data on a bounding box location estimated for each person[29]. Object tracking in video streams has been a popular topic in the field of computer vision. Tracking is a particularly important issue in human motion analysis since it serves as a means to prepare data for pose estimation and action recognition. In contrast to human detection, human tracking belongs to a higher-level computer vision problem. However the tracking algorithms within human motion analysis usually have considerable intersection with motion segmentation during processing. Tracking over time typically involves matching objects in consecutive frames using features such as points, lines or blobs. That is to say, tracking may be considered to be equivalent to establishing coherent relations of image features between frames with respect to position, velocity, shape, texture, color, etc[33].

Useful mathematical tools for tracking include Kalman filter, the Condensation algorithm, Dynamic Bayesian Network, etc. Kalman filtering is a state estimation method based on Gaussian distribution[8]. Unfortunately, it is restricted to situations where the probability distribution of the state parameters is unimodal. That is, it is inadequate in dealing with simultaneous multi-modal distributions with the presence of occlusion, cluttered background resembling the tracked objects, etc. The Condensation algorithm has shown to be a powerful alternative. It is a kind of conditional density propagation method for visual tracking. Based upon sampling the posterior distribution estimated in the previous frame, it is extended to propagate these samples iteratively to successive images. By combining a tractable dynamic model with visual observations, it can accomplish highly robust tracking of object motion. However, it usually requires a relatively large number of samples to ensure a fair maximum likelihood estimation of the current state.

Model-based tracking: Traditionally, the geometric structure of human body can be represented as stick figure, 2-D contour or volumetric model [33,4]. So body segments can be approximated as lines, 2-D ribbons and 3-D volumes accordingly.

The essence of human motion is typically addressed by the movements of the torso, head and four limbs, so the stick-figure representation can be used to approximate a human body as a combination of line segments linked by joints [34].

2-D contour: This kind of representation of human body is directly relevant to the human body projection in the image plane. In such description, human body segments are analogous to 2-D ribbons or blobs. For instance, The parameterized image motion of the patches was constrained to enforce the articulated movement, and was used to deal with the analysis of articulated motion of human limbs. In the work by Leung and Yang [32], the subject’s outline was estimated as edge regions represented by 2-D ribbons which were U-shaped edge segments. The 2-D ribbon model was used to guide the labeling of the image data. A silhouette or contour is relatively easy to be extracted from both the model and image. Based upon 2-D contour representation.
Volumetric models: The disadvantage of 2-D models is its restriction to the camera’s angle, so many researchers are trying to depict the geometric structure of human body in more detail using some 3-D models such as elliptical cylinders, cones, spheres, etc. The more complex 3-D volumetric models, the better results may be expected but they require more parameters and lead to more expensive computation during the matching process.

An important advantage of 3-D human model is the ability to handle occlusion and obtain more significant data for action analysis. However, it is restricted to impractical assumptions of simplicity regardless of the body kinematics constraints, and has high computational complexity as well[33].

Region-based tracking: The idea here is to identify a connected region associated with each moving object in an image, and then track it over time using a cross-correlation measure. Region-based tracking approach [34] has been widely used today.

The region-based tracking approach works reasonably well. However, difficulties arise in two important situations. The first is that of long shadows, and it may result in connecting up blobs that should have been associated with separate people. This problem may be resolved to some extent by making use of color or exploiting the fact that shadow regions tend to be devoid of texture. The more serious, and so far intractable, problem for video tracking has been that of congested situations. Under these conditions, people partially occlude one another instead of being spatially isolated. This makes the task of segmenting individual humans very difficult. The resolution to this problem may require tracking systems using multiple cameras.

Active contour based tracking: Active contour based tracking has been intensively studied over the past few years. A statistical framework, for which the observed inter-frame difference density function was approximated using a mixture model, was used to provide the initial motion detection boundary. Then the detection and tracking problems were addressed in a common framework that employed a geodesic active contour objective function. Using the level set formulation scheme, complex curves could be detected and tracked while topological changes for the evolving curves were naturally managed.

Feature-based tracking: Abandoning the idea of tracking objects as a whole, this tracking method uses sub-features such as distinguishable points or lines on the object to realize the tracking task. Its benefit is that even in the presence of partial occlusion, some of the sub-features of the tracked objects remain visible. Feature-based tracking includes feature extraction and feature matching. Low-level features such as points are easier to extract. It is relatively more difficult to track higher-level features such as lines and blobs. So, there is usually a trade-off between feature complexity and tracking efficiency.

2.4 Behavior understanding

After successfully tracking the moving humans from one frame to another in an image sequence, the problem of understanding human behaviors from image sequences follows naturally. Behavior understanding involves action recognition and description. As a final or long-time goal, human behavior understanding can guide the development of many human motion analysis systems. In our opinion, it will be the most important area of future research in human motion analysis.

Behavior understanding is to analyze and recognize human motion patterns, and to produce high-level description of actions and interactions. It may be simply considered as a classification problem of time varying feature data, i.e., matching an unknown test sequence with a group of labeled reference sequences representing typical human actions. It is obvious that the basic problem of human behavior understanding is how to learn the reference action sequences from training samples, how to enable both training and matching methods effectively to cope with small variations at spatial and time scales within similar classes of motion patterns, and how to effectively interpret actions using natural language. All these are hard problems and have received increasing attentions from researchers.

2.4.1 General techniques

Action recognition involved in behavior understanding may be thought as a time-varying data matching problem. The general analytical methods for matching time-varying data are outlined in the following.

- Dynamic time warping: Dynamic time warping (used widely for speech recognition in the early days), is a template-based dynamic programming matching technique. It has the advantage of conceptual simplicity and robust performance, and has been used in the matching of human movement patterns recently [33,34]. As far as Dynamic time warping is concerned, even if the time scale between a test pattern and a reference pattern may be inconsistent, it can still successfully establish matching as long as time ordering constraints hold.

- Hidden Markov models: Hidden Markov models (HMMs) [33], a kind of stochastic state machine, is a more sophisticated technique for analyzing time-varying data with spatio-temporal variability. Its model structure can be summarized as a hidden neural network: Neural network [4,34] is also an interesting approach for analyzing time-varying data. As larger data sets become available, more emphasis is being placed on neural networks for representing temporal
2.4.2 Action recognition
Template matching: This approach based on template matching, first converts an image sequence into a static shape pattern, and then compares it to pre-stored action prototypes during recognition.
State-space approaches: The approach based on the state space models defines each static posture as a state, and uses certain probabilities to generate mutual connections between these states. Any motion sequence can be considered as a tour through various states of these static postures. Through these tours, joint probability with the maximum value is selected as the criterion for action classification. Nowadays, the state space models have been widely applied to prediction, estimation, and detection of temporal series. HMM is the most representative method used to study discrete time series.
Semantic description: Applying concepts of natural languages to vision systems is becoming popular, so the semantic description of human behaviors has recently received considerable attention [33]. Its purpose is to reasonably choose a group of motion words or short expressions to report the behaviors of the moving objects in natural scenes.

2.5 Applications of human motion detection
Smart Surveillance
- From Passive “after the fact” tool to Active media
- Able to detect a burglary in progress and alert security
Perceptual Interface
- Human-Machine Interface based on gestures, body poses and facial expressions
- Complement to speech recognition and natural language understanding
- Active video games: the player has to physically simulate the game actions.
Virtual Reality
- Better virtual world interaction
- Modify virtual reality based on body movements, human poses and actions
Much more...
- Analysis of trainings and athletic performance
- Personal identification based on gait (biometric feature)
- Model based image coding
- Character animations[5]

3. MOTION SEGMENTATION
One of the simplest types of motion analysis is to detect image points that refer to moving points in the scene. The typical result of this processing is a binary image where all image points (pixels) that relate to moving points in the scene are set to 1 and all other points are set to 0. This binary image is then further processed, e.g., to remove noise, group neighboring pixels, and label objects. Motion detection can be done using several methods; the two main groups are differential methods and methods based on background segmentation, temporal differencing and optical flow.
Moving to human detection is the first steps processes for nearly every system of vision-based human analysis. The aim on moving human detection is to segment the regions corresponding to people from the rest of an image sequence. It is known to be a significant and difficult research problem. The four conventional approaches to moving object detection: temporal differencing (two-frame or three-frame), optical flow, and background estimation and statistical method. Temporal differencing is very adaptive to dynamic environments, but generally does a poor job of extracting all relevant feature pixels. Optical flow can be used to detect independently moving targets in the presence of camera motion, however most optical flow computation methods are very complex and are inapplicable to real-time algorithms without specialized hardware. Background subtraction is a particularly popular method for motion segmentation especially under those situations with a relatively static background. It attempts to detect moving regions in an image by differencing between current image and a reference background image in a pixel-by-pixel fashion. However, it is extremely sensitive to changes of dynamic scenes due to lighting and extraneous events. To overcome that problem a new statistical background model initialize and maintenance of the background model approach is presented.

Fig 3.1 Classification of human motion detection
In computer vision, segmentation is the process of partitioning a digital image into multiple segments (sets of pixels, also known as super pixels). The goal of segmentation is to simplify and/or change the representation of an image into something that is more meaningful and easier to analyze. Image segmentation is typically used to locate objects and boundaries (lines, curves, etc.) in images. More precisely, image segmentation is the process of assigning a label to every pixel in an image such that pixels with the same label share certain visual characteristics. The result of image segmentation is a set of segments that collectively cover the entire image, or a set of contours extracted from the image. Human motion analysis has three levels as shown in the figure above. This thesis paper describes the lower level, Motion segmentation.

3.1 Methods for motion segmentation
Motion segmentation has four methods namely
- Background subtraction method
- Optical flow method
- Temporal differencing method
- Statistical method

![Motion Segmentation Diagram](image)

The main attributes of Motion Segmentation methods are as follows.

- Feature-based or Dense-based: In feature-based methods, the objects are represented by a limited number of points like corners or salient points, whereas dense methods compute a pixel-wise motion.
- Occlusions: it is the ability to deal with occlusions.
- Multiple objects: it is the ability to deal with more than one object in the scene.
- Spatial continuity: it is the ability to exploit spatial continuity.
- Temporary stopping: it is the ability to deal with temporary stop of the objects.
- Robustness: it is the ability to deal with noisy images (in case of feature based methods it is the position of the point to be affected by noise but not the data association).
- Sequentiality: it is the ability to work incrementally, this means for example that the algorithm is able to exploit information that was not present at the beginning of the sequence.

Missing data: it is the ability to deal with missing data.
Non-rigid object: it is the ability to deal with non-rigid objects.
Camera model: if it is required, which camera model is used.

4. Background Subtraction Method

4.1 Introduction
Background subtraction is commonly used technique for achieving the segmentation. It is the Classical and popular approach to detect foreground regions by subtracting the background image. The basic scheme of background subtraction is to subtract the image from a reference image that models the background scene. The popularity of background subtraction largely comes from its computational efficiency and is widely used in real-time methods for identifying foreground objects in a video stream. The most common paradigm for performing background subtraction is to build an explicit model of the background foreground objects are then detected by calculating the difference between the current frame and the background model. It compares an observed image with an estimate of the image if it contained no object of interest. The areas of the image plane where there is a significant difference between the observed and estimated image indicates the location of the objects of interest. The principle behind bgs is to subtract and threshold a background model image from the current frame. The result gives the differences between the two subtracted images, and it is hypothesized that these differences correspond to moving objects. In practice, this is not always the case, as differences may correspond to shadows, changes of lighting, or camera noise. Furthermore, some of them may correspond to changes in an image, like waving leaves or waves on a lake, which are irrelevant to the application. The challenge, then, is to propose a background model that allows filtering of these unavoidable perturbations, while still correctly detecting the moving objects of interest. The name background subtraction comes from the simple technique of subtracting the observed image from the estimated image and thresholding the result to generate the objects of interest. A natural improvement is to build a moving average estimate of the background, to keep track of illumination changes. To analyze the behavior, the first step is human detection and tracking. Tracking involves detection of regions of interest in a frame and then finding frame-to-frame correspondence of each region’s location and shape. Nearly, every system in the HMD starts with segmentation, current motion segmentation methods mainly based on background subtraction or temporal differencing or optical flow or statistical methods. Development of a reliable background models adaptive to dynamic changes in complex environments is still a challenge.

The basic approach for background subtraction is to store the background image as the reference image, in which there is no movement and then in every other frame subtract the reference image to extract the alien objects in the scene. The alien
Background modeling

The defining characteristic of a BGS algorithm is how it defines and updates its background model.

Recursive techniques: Recursive techniques maintain a single background model that is updated with each new video frame. These techniques are generally computationally efficient and have minimal memory requirements. Running Gaussian average, If we consider the background to be nearly static, then the main source of variation in a pixel’s color will be due to camera noise. Since it is common to model camera noise as being Gaussian, it is natural to model each pixel in the background model as a Gaussian distribution. Gaussian mixture model GMM with adaptive number of Gaussians, the major strengths of this approach are its computational efficiency, robustness to noise, and simplicity. A notable limitation is that it does not model the variance of a pixel.

Non-recursive techniques: Non-recursive techniques maintain a buffer L of n previous video frames and estimate a background model based solely on the statistical properties of these frames. This causes non-recursive techniques to have higher memory requirements than recursive techniques. However, since they have explicit access to the most recent n video frames they can model aspects of the data not possible with recursive techniques.

Median filtering: Median filtering sets each channel of a pixel in the background model to be the median value as determined from the buffer of video frames are in ascending order. Extending the buffer to include the last background model value makes the algorithm more robust to noise when small buffer sizes are used.

Foreground detection:

Pixels in a new video frame that cannot be adequately explained by the background model are considered to be from a foreground object. The major distinction that defines how this comparison is performed is whether or not the background model is statistical in nature. Algorithms that lack a statistical framework classify a new pixel as being from the foreground.

The major limitation of this approach is that it uses a single threshold for all pixel models even though some pixels may exhibit more variation than others. As such, methods that provide a measure of variance for each pixel are preferable. Algorithms which model pixels as a probability density function classify a new pixel as being from the foreground.

Post processing Techniques:

A number of post-processing techniques that can be used to improve upon the foreground masks that result from foreground detection. Noise removal: Due to camera noise and limitations of the background model, the foreground mask typically contains numerous small noise blobs. These erroneous blobs can be removed by applying a noise filtering algorithm to the foreground mask. Removing these erroneous blobs early is desirable since they can interfere with later post-processing stages. Blob processing: Most applications using BGS are interested in identifying foreground objects. As such, connected-component labeling is almost always performed in order to identify object-level blobs.

4.3 Method overview

We estimate foreground/background appearance models in the videos captured by a moving camera, where a video frame is divided into a regular grid of blocks. The method is based on the following assumptions: At the beginning of a sequence,
the major motion in the scene belongs to background and the outliers are considered as foreground. Spatially adjacent background (or foreground) areas have similar motions. The temporal variations of foreground and background appearances are smooth subject to the proper image registration by motion estimation. The most commonly used Background subtraction is ViBe called Visual Background extractor. The number of cameras available worldwide has increased dramatically over the last decade. But this growth has resulted in a huge augmentation of data, meaning that the data are impossible either to store or to handle manually. In order to detect, segment, and track objects automatically in videos, several approaches are possible. Simple motion detection algorithms compare a static background frame with the current frame of a video scene, pixel by pixel. This is the basic principle of background subtraction, which can be formulated as a technique that builds a model of a background and compares this model with the current frame in order to detect zones where a significant difference occurs. The purpose of a background subtraction algorithm is therefore to distinguish moving objects (hereafter referred to as the foreground) from static, or slow moving, parts of the scene (called background). Note that when a static object starts moving, a background subtraction algorithm detects the object in motion as well as a hole left behind in the background (referred to as a ghost). Clearly a ghost is irrelevant for motion interpretation and has to be discarded. An alternative definition for the background is that it corresponds to a reference frame with values visible most of the time, that is with the highest appearance probability, but this kind of framework is not straightforward to use in practice. While a static background model might be appropriate for analyzing short video sequences in a constrained indoor environment, the model is ineffective for most practical situations; a more sophisticated model is therefore required. Moreover, the detection of motion is often only a first step in the process of understanding the scene. The first assumption is to obtain reliable foreground and background models through motion segmentation in the first few frames. The second and third assumptions allow us to take spatial and temporal evidences for estimating block-wise motion and appearance models.

4.4 Applications of Background Subtraction method

- Traffic monitoring (counting vehicles, detecting & tracking vehicles)
- Human action recognition (run, walk, jump, squat . . .)
- Human-computer interaction (“human interface”)
- Object tracking (watched tennis lately?!?)
- And in many other cool applications of computer vision such as digital forensics.

4.5 Requirements

A reliable and robust background subtraction algorithm should handle:

- Sudden or gradual illumination changes.
- High frequency, repetitive motion in the background (such as tree leaves, flags, waves . . .), and Long-term scene changes (a car is parked for a month).

4.6 Advantages and Disadvantages

**Advantages**
- Extremely easy to implement and use!
- All pretty fast.
- Corresponding background models are not constant, they change over time.

**Disadvantages**
- Accuracy of frame differencing depends on object speed and frame rate!
- Mean and median background models have relatively high memory requirements.

5. Optical flow

5.1 Introduction

Optic flow is the perceived visual motion of objects as the observer moves relative to them. For example, say you are driving a car. A sign on the side of the road would move from the center of your vision to the side, growing as you approached. If you had 360 degree vision, this sign would proceed to move quickly past your side to your back, where it would shrink. This motion of the sign is its optic flow. Optical flow is a concept for considering the motion of objects within a visual representation. Typically the motion is represented as vectors originating or terminating at pixels in a digital image sequence. Optical flow is useful in pattern recognition, computer vision, and other image processing applications. It is closely related to motion estimation and motion compensation. Often the term optical flow is used to describe a dense motion field with vectors at each pixel, as opposed to motion estimation/compensation which uses vectors for blocks of pixels as in video compression methods such as MPEG. Optical flow or optic flow is the pattern of apparent of objects, surfaces, and edges in a visual scene caused by the relative motion between an observer and the scene. Optical flow techniques such as motion detection, object segmentation, time-to-collision and focus of expansion calculations, motion compensated encoding, and stereo disparity measurement utilize this motion of the objects' surfaces and edge. Optical flow is the distribution of apparent velocities of movement of brightness patterns in an image. Optical flow can arise from relative motion of objects and the viewer. Consequently, optical flow can give important information about the spatial arrangement of the objects viewed and the rate of change of this arrangement. Discontinuities in the optical flow can help in segmenting images into regions that correspond to different objects. Optical flow method involves two stages. Firstly, the features are found in two or more consecutive images. The act
of feature extraction, if done well, will both reduce the amount of information to be processed (and so reduce the workload), and also go some way towards obtaining a higher level of understanding of the scene, by its very nature of eliminating the unimportant parts. Secondly, these features are matched between the frames. In the simplest and commonest case, two frames are used and two sets of features are matched to give a single set of motion vectors. Additionally, finding optic flow using edges has the advantage (over using two dimensional features) that edge detection theory is well advanced. It has the advantage over approaches which attempt to find flow everywhere in the image. The features are found according to the below algorithm:

Feature selection algorithm:
1. Compute the spatial gradient matrix and its minimum eigen value at every pixel in the image.
2. Call the maximum value of eigen values over the whole image.
3. Retain the image pixels that have an eigen value larger than a percentage of maximum eigen values. This percentage can be 10% or 5%.
4. From those pixels, retain the local max. Pixels (a pixel is kept if its eigen value is larger than that of any other pixel in its 3 x 3 neighborhood).
5. Keep the subset of those pixels so that the minimum distance between any pair of pixels is larger than a given threshold distance (e.g. 10 or 5 pixels).

5.2 OPTICAL FLOW IN MOTION ANALYSIS
Optical flow gives a description of motion and can be a valuable contribution to image interpretation even if no quantitative parameters are obtained from motion analysis. Optical flow can be used to study a large variety of motions moving observer and static objects, static observer and moving objects, or both moving. Optical flow analysis does not result in motion trajectories instead, more general motion properties are detected that can significantly increase the reliability of complex dynamic image analysis. Optical-flow based motion analysis can recognize these basic elements by applying a few relatively simple operators to the flow. Optical Flow: Describes coherent motion of points or features between image frames Segmentation done by grouping motion vectors into groups having coherent motion, Can be used with ego-motion. Optical flow is Computationally intensive and sensitive to noise. Not real-time without hardware.

5.3 OPTICAL FLOW COMPUTATION
Optical flow computation is based on two assumptions:
1. The observed brightness of any object point is constant over time.
2. Nearby points in the image plane move in a similar manner (the velocity smoothness constraint). Suppose we have a continuous image; \( f(x,y,t) \) refers to the gray-level of \((x,y)\) at time \( t \). Representing a dynamic image as a function of position and time permits it to be expressed.

Types of Detection used in the Software, Two frames difference motion detector
This type of motion detector is the simplest one and the quickest one. The idea of this detector is based on finding amount of difference in two consequent frames of video stream. The greater is difference, the greater is motion level. As it can be seen from the picture below, it does not suite very well those tasks, where it is required to precisely highlight moving object. However it has recommended itself very well for those tasks, which just require motion detection.

Motion detectors based on background modeling: In contrast to the above motion detector, these motion detectors are based on finding difference between current video frame and a frame representing background. These motion detectors try to use simple techniques of modeling scene's background and updating it through time to get into account scene's changes. The background modeling feature of these motion detectors gives the ability of more precise highlighting of motion regions. Below are demonstrated outputs of two versions of motion detectors based on background modeling. One does more precise highlight of moving objects' borders, but consumes more computational resources. Another one does less precise objects' highlight in the cost of requiring much less computational resources.

i) Low-Precision Background Modeling
This software has capability to detect motion detection based on optical flow and gives certain level of motion detection which can be used as a threshold. Analyzing motion level and comparing it with predefined threshold allows raising alarm, when detected motion level is greater then the level which is considered to be safe. In addition to motion level detection, there are four types of motion detectors and all of them support highlighting of detected motion regions (can be turned on/off).

ii) High-Precision Background Modeling
Optical flow computation is based on two assumptions:
1) The observed brightness of any object point is constant over time.
2) Nearby points in the image plane move in a similar manner (the velocity smoothness constraint). Optical flow computation will be in error if the constant brightness and velocity smoothness assumptions are violated. In real imagery, their violation is quite common. Typically, the optical flow changes dramatically in highly textured regions, around moving boundaries, at depth discontinuities, etc. Resulting errors propagate across the entire optical flow solution.

5.4 Methods for determining Optical Flow
Phase correlation – inverse of normalized cross-power spectrum.
Block-based methods – minimizing sum of squared differences or sum of absolute differences, or maximizing normalized cross-correlation.
Differential methods of estimating optical flow, based on partial derivatives of the image signal and/or the sought flow
field and higher-order partial derivatives. Discrete optimization methods – the search space is quantized, and then image matching is addressed through label assignment at every pixel, such that the corresponding deformation minimizes the distance between the source and the target image. The optimal solution is often recovered through min-cut max-flow algorithms, linear programming or belief propagation methods.

5.5 Uses of Optical Flow

Motion estimation and video compression have developed as a major aspect of optical flow research. While the optical flow field is superficially similar to a dense motion field derived from the techniques of motion estimation, optical flow is the study of not only the determination of the optical flow field itself, but also of its use in estimating the three-dimensional nature and structure of the scene, as well as the 3D motion of objects and the observer relative to the scene, most of them using the Image Jacobian.

Optical flow was used by robotics researchers in many areas such as: object detection and tracking, image dominant plane extraction, movement detection, robot navigation and visual odometry. Optical flow information has been recognized as being useful for controlling micro air vehicles. The application of optical flow includes the problem of inferring not only the motion of the observer and objects in the scene, but also the structure of objects and the environment. Since awareness of motion and the generation of mental maps of the structure of our environment are critical components of animal (and human) vision, the conversion of this innate ability to a computer capability is similarly crucial in the field of machine vision.

![Fig 5.1](image) The optical flow vector of a moving object in a video sequence.

Consider a five-frame clip of a ball moving from the bottom left of a field of vision, to the top right. Motion estimation techniques can determine that on a two dimensional plane the ball is moving up and to the right and vectors describing this motion can be extracted from the sequence of frames. For the purposes of video compression the sequence is now described as well as it needs to be. However, in the field of machine vision, the question of whether the ball is moving to the right or if the observer is moving to the left is unknowable yet critical information. Not even if a static, patterned background were present in the five frames, could we confidently state that the ball was moving to the right, because the pattern might have an infinite distance to the observer.

Motion detection is very important in image processing. One way of detecting motion is using optical flow. Optical flow cannot be computed locally, since only one independent measurement is available from the image sequence at a point, while the flow velocity has two components. A second constraint is needed. The method used for finding the optical flow in this project is assuming that the apparent velocity of the brightness pattern varies smoothly almost everywhere in the image. Optical flow is the distribution of apparent velocities of movement of brightness patterns in an image. Optical flow can arise from relative motion of objects and the viewer. Consequently, optical flow can give important information about the spatial arrangement of the objects viewed and the rate of change of this arrangement. Discontinuities in the optical flow can help in segmenting images into regions that correspond to different objects. Optical flow gives a description of motion and can be a valuable contribution to image interpretation even if no quantitative parameters are obtained from motion analysis. Optical flow can be used to study a large variety of motions moving observer and static objects, static observer and moving objects, or both moving. Optical flow analysis does not result in motion trajectories instead, more general motion properties are detected that can significantly increase the reliability of complex dynamic image analysis [34]. Motion, as it appears in dynamic images, is usually some combination of four basic elements:

i) Translation at constant distance from the observer.
ii) Translation in depth relative to the observer.
iii) Rotation at constant distance about the view axis.
vi) Rotation of a planar object perpendicular to the view axis.

Optical flow based motion analysis can recognize these basic elements by applying a few relatively simple operators to the flow.

Motion form recognition is based on the following facts:

a) Translation at constant distance is represented as a set of parallel motion vectors.
b) Translation in depth forms a set of vectors having a common focus of expansion.
c) Rotation at constant distance results in a set of concentric motion vectors.
d) Rotation perpendicular to the view axis forms one or more sets of vectors starting from straight line segments. Exact determination of rotation axes and translation trajectories can be computed, but with a significant increase in difficulty of analysis [34].

5.6 Application of Optical Flow

Altitude measurement

Ever notice when traveling by plane, the higher you are the slower the ground below you seems to move? For aerial robots that have a known constant speed, by analyzing pixel velocity from a downward facing camera the altitude can be calculated.
The slower the pixels travel, the higher the robot. A potential problem however is when your robot rotates in the air, but this can be accounted for by adding additional sensors like gyroscopes and accelerometers.

Velocity measurements for constant altitude

For a robot that is traveling at some known altitude, by analyzing pixel velocity, the robot velocity can be calculated. This is the converse of the altitude measurement method. It is impossible to gather both altitude and velocity data simultaneously using only optical flow, so a second sensor (such as GPS or an altimeter) needs to be used. If however your robot was an RC car, the altitude is already known (probably an inch above the ground). Velocity can then be calculated using optical flow with no other sensors. Optical flow can be used to directly compute time to impact for missiles. Optical flow also is a technique often used by insects to gauge flight speed and direction.

Tracking

Please see tracking above, and background subtraction below. The optical flow method of tracking combines both of those methods together. By removing the background, all that needs to be done is analyze the motion of the moving pixels.

3D scene analysis

By analyzing motion of all pixels, it is possible to generate rough 3D measurements of the observed scene. For example, the below image of the subway train: the pixels on the far left are moving fast, and they are both converging and slowing down towards the center of the image. With this information, 3D information of the train can be calculated (including velocity of train and angle of the track).

6. Temporal differencing

6.1 Introduction

Temporal differencing is very adaptive to dynamic environment, but generally does a poor job of extracting all relevant feature pixels that is segmented regions contain holes in motion object. To fill the holes of blob, we use spatial clustering (the creation of spatial distribution between pixels). The Temporal differencing is very adaptive to changes in dynamic environment and another advantage is that it does not make assumptions about the scene. Temporal differencing change detection based on frame difference attempts to detect moving region by making use of the difference of consecutive frames in video sequence. The approach of temporal differencing makes use of pixel-wise difference between two or three consecutive frames in an image sequence to extract moving regions. As an example of this method, detected moving targets in real video streams using temporal differencing. After the absolute difference between the current and the previous frame was obtained, a threshold function was used to determine change. By using a connected component analysis, the extracted moving sections were clustered into motion regions. These regions were classified into predefined categories according to image-based properties for later tracking. An improved version is to use three-frame differencing instead of two-frame differencing. This hybrid algorithm is very fast and surprisingly effective for detecting moving objects in image sequences. Pixel-wise difference between 2 or 3 consecutive frames in a sequence Adaptive to dynamic environments, but does often miss entire feature pixels (ex. holes),improved version use three-frame differencing. Temporal difference is a prediction method. It has been mostly used for solving the reinforcement learning problem. Temporal differencing learning is a combination of Monte Carlo ideas and dynamic programming ideas. Temporal differencing resembles a Monte Carlo method because it learns by sampling the environment according to some policy. Temporal differencing is related to dynamic programming techniques because it approximates its current estimate based on previously learned estimates (a process known as bootstrapping). The Temporal differencing learning algorithm is related to the temporal difference model of animal learning. Temporal differencing is an efficient model-free method that saves on storage space since no model of the environment is explicitly stored. Temporal differencing attempts to detect moving regions by making use of the pixel-by-pixel difference of consecutive frames (two or three) in a video sequence. This method is highly adaptive to dynamic scene changes; however, it generally fails in detecting whole relevant pixels of some types of moving objects. A sample object for inaccurate motion detection. The mono colored region of the human on the left hand side makes the temporal differencing algorithm to fail in extracting all pixels of the human’s moving region. Also, this method fails to detect stopped objects in the scene. Additional methods need to be adopted in order to detect stopped objects for the success of higher level processing. temporal differencing, which is used to find the difference between two continuous image data, obtains the changes in the volume of a moving object. If the background variation is not great, this method works well. The temporal differencing method uses two continuous images for subtraction to detect moving objects. It quickly adapts to changes in lighting or background. Although this method may detect moving objects which are already broken up, we learn if any moving object is detected.

6.2 Advantages of Temporal Differencing method

Temporal differencing is highly adaptive to dynamic scene changes.

It uses pixel-wise difference between two or three consecutive frames.
6.3 Disadvantages of Temporal Differencing Method

Temporal differencing method fails in detecting whole relevant pixels of same type of moving objects.

It fails to detect a change between consecutive frames and loses the object.

7. Statistical Method

7.1 Introduction

Statistical studies allow analysts to estimate key parameters of cost or production models. Econometric analyses require a large data set to ensure reliable results. Obtaining the number of observations needed to derive an efficient and unbiased estimate of cost (or production) structures can often prove to be a difficult task. Regression results are sensitive to model specification (for example, a linear vs. a non-linear functional form). Statistics is the study of the collection, organization, analysis, and interpretation of data.\textsuperscript{[1],[2]} It deals with all aspects of this, including the planning of data collection in terms of the design of surveys and experiments. A statistician is someone who is particularly well versed in the ways of thinking necessary for the successful application of statistical analysis. Such people have often gained this experience through working in any of a wide number of fields. There is also a discipline called mathematical statistics that studies statistics mathematically. The statistical methods are proposed to extract change regions from the background and there methods are inspired by the background subtraction methods. The statistical methods use the characteristics of the individual pixels or groups of pixels to construct more advanced background models and the statistics of the background can be updated dynamically during processing. Each pixel in the current image can be classified into foreground or background by comparing the statistics of the current background model. The majority of the statistical methods proposed so in the literature for background subtraction are Gaussian or kernel distribution to model the background. A statistical approach provides a general tool that can be used in a very different way depending on the specific technique. Statistical motion segmentation can be seen as a classification problem, where each pixel has to be classified as background or foreground. Statistics is the study of the collection, organization, analysis, and interpretation of data. It deals with all aspects of this, including the planning of data collection in terms of the design of surveys and experiments. A statistician is someone who is particularly well versed in the ways of thinking necessary for the successful application of statistical analysis. Such people have often gained this experience through working in any of a wide number of fields. There is also a discipline called mathematical statistics that studies statistics mathematically. Statistical methods can be used for summarizing or describing a collection of data; this is called descriptive statistics. This is useful in research, when communicating the results of experiments. In addition, patterns in the data may be modeled in a way that accounts for randomness and uncertainty in the observations, and are then used for drawing inferences about the process or population being studied; this is called inferential statistics. Inference is a vital element of scientific advance, since it provides a means for drawing conclusions from data that are subject to random variation. Descriptive statistics and analysis of the new data tend to provide more information as to the truth of the proposition. Descriptive statistics and the application of inferential statistics (predictive statistics) together comprise applied statistics. Theoretical statistics concerns both the logical arguments underlying justification of approaches to statistical inference, as well encompassing mathematical statistics. Mathematical statistics includes not only the manipulation of probability distributions necessary for deriving results related to methods of estimation and inference, but also various aspects of computational statistics and the design of experiments. Statistics is closely related to probability theory, with which it is often grouped; the difference is roughly that in probability theory, one starts from the given parameters of a total population to deduce probabilities pertaining to samples, but statistical inference moves in the opposite direction, inductive inference from samples to the parameters of a larger or total population. A common goal for a statistical research project is to investigate causality, and in particular to draw a conclusion on the effect of changes in the values of predictors or independent variables on dependent variables or response. There are two major types of causal statistical studies: experimental studies and observational studies. In both types of studies, the effect of differences of an independent variable (or variables) on the behavior of the dependent variable are observed. The difference between the two types lies in how the study is actually conducted. Each can be very effective. An experimental study involves taking measurements of the system under study, manipulating the system, and then taking additional measurements using the same procedure to determine if the manipulation has modified the values of the measurements. In contrast, an observational study does not involve experimental manipulation. Instead, data are gathered and correlations between predictors and response are investigated.

7.2 Basic steps of a statistical experiment

1. Planning the research, including finding the number of replicates of the study, using the following information: preliminary estimates regarding the size of the solution, alternative hypotheses, and the estimated experimental variability. Consideration of the selection of experimental subjects and the ethics of research is necessary. Statisticians recommend that experiments compare (at least) one new treatment with a standard treatment or control, to allow an unbiased estimate of the difference in treatment effects.

2. Design of experiments, using blocking to reduce the influence of confounding variables, and randomized
assignment of solution to subjects to allow unbiased estimates of treatment effects and experimental error. At this stage, the experimenters and statisticians write the experimental protocol that shall guide the performance of the experiment and that specifies the primary analysis of the experimental data.

3. Performing the experiment following the experimental protocol and analyzing the data following the experimental protocol.
4. Further examining the data set in secondary analyses, to suggest new hypotheses for future study.
5. Documenting and presenting the results of the study.

7.3 Applications

Statistical techniques are used in a wide range of types of scientific and social research, including: biostatistics, computational biology, computational sociology, network biology, social science, sociology and social research. Some fields of inquiry use applied statistics so extensively that they have specialized terminology. These disciplines include: Actuarial science, Applied information economics, Biostatistics, Business statistics, Chemometrics (for analysis of data from chemistry), Data mining (applying statistics and pattern recognition to discover knowledge from data), Demography, Econometrics, Energy statistics, Engineering statistics, Epidemiology, Image processing, Psychological statistics. Statistics form a key basis tool in business and manufacturing as well. It is used to understand measurement systems variability, control processes (as in statistical process control or SPC), for summarizing data, and to make data-driven decisions. In these roles, it is a key tool, and perhaps the only reliable tool.

8. Advantages and Disadvantages of Background Subtraction Method

<table>
<thead>
<tr>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>It is mostly suitable for static scenes</td>
<td>It is extremely sensitive to changes of dynamic scenes due to lightening and extraneous events.</td>
</tr>
<tr>
<td>It subtracts the image pixel by pixel from a referenced background image</td>
<td>Accuracy of frame differencing depends on object speed and frame rate</td>
</tr>
<tr>
<td>It is widely used in real-time methods</td>
<td>Requirement of memory is high.</td>
</tr>
<tr>
<td>It is easy to use and implement</td>
<td>It is fast and effective</td>
</tr>
<tr>
<td>It is fast and effective</td>
<td></td>
</tr>
</tbody>
</table>

9. Advantages and Disadvantages of Optical Flow Method

<table>
<thead>
<tr>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>It makes use of the flow vectors of moving object over time to detect moving regions in an image</td>
<td>Optical flow method is computationally complex</td>
</tr>
<tr>
<td>It can detect motion in video sequence even when frame in a moving camera</td>
<td>It is cannot be used in real-time methods</td>
</tr>
<tr>
<td>It allows the user to judge how close the object is</td>
<td>It need special hardware to perform operations</td>
</tr>
<tr>
<td>It increases the reliability of complex dynamic image analysis</td>
<td>It results in error if the constant brightness and velocity smoothen assumption are violated</td>
</tr>
<tr>
<td>It is widely used by robotic researchers</td>
<td></td>
</tr>
</tbody>
</table>

10. Advantages and Disadvantages of Temporal Differencing Method

<table>
<thead>
<tr>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>It attempts to detect moving region by making use of pixel by pixel differencing of consecutive frames(two or three)</td>
<td>It fails in detecting whole relevant pixels of the same type of moving objects</td>
</tr>
<tr>
<td>It is highly adaptive to dynamic scene change</td>
<td>It fails to detect stopped objects in the scene</td>
</tr>
<tr>
<td>It does not make assumptions about the scene</td>
<td>It is not effective to get the whole region of the moving object</td>
</tr>
<tr>
<td>It is an efficient model-free method that saves on storage space since no model of the environment is explicitly stored.</td>
<td>It generates holes inside the moving objects</td>
</tr>
<tr>
<td>Additional methods need to be adopted to detect stopped object for the success of higher level processing</td>
<td>It fails to detect a change between consecutive frames and loses the object</td>
</tr>
</tbody>
</table>
11. Advantages and Disadvantages of Statistical Method

<table>
<thead>
<tr>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Most advanced method uses statistical methods</td>
<td>It require a large data set to ensure reliable results</td>
</tr>
<tr>
<td>it is popular because of its reliability in scenes</td>
<td>Well versed ways of thinking is necessary for the successful application of statistical method</td>
</tr>
<tr>
<td>Uses the characteristics of individual pixels or group of pixels to construct advanced background model</td>
<td></td>
</tr>
<tr>
<td>It can be adapted dynamically during the processing</td>
<td></td>
</tr>
</tbody>
</table>

12. The Basic Background Subtraction Method

The estimated background is just the previous frame. And the frame difference: \( |frame_i-frame_{i-1}| > Th \). It’s very sensitive to the threshold \( Th \). If we make background as the average or the median of the previous \( n \) frames, it would be rather fast but very memory consuming. Background as the running average:

\[
B_{t+1} = \alpha F_t + (1-\alpha)B_t
\]

\( \alpha \), the learning rate, is typically 0.05 No more memory requirements

The background model is computed as a chronological average from the pixel’s history. History means just the previous \( n \) frames. A weighted average where recent frames have higher weight.

At each new frame, each pixel is classified as either foreground or background. if the pixel is classified as foreground, it is ignored in the background model.

In this way, we prevent the background model to be polluted by pixel logically not belonging to the background scene. The running average:

\[
B_{t+1}(x,y) = \alpha F_t(x,y) + (1-\alpha)B_t(x,y) \quad \text{if } F_t(x,y) \text{ is background}
\]
\[
B_{t+1}(x,y) = B_t(x,y) \quad \text{if } F_t(x,y) \text{ is foreground}
\]

But they do not provide an explicit method to choose the threshold. For this, Wren have proposed to model background independently at each \((i,j)\) pixel location. This gives the fitting one Gaussian distribution \((\mu,\sigma)\) over the histogram which provides the background PDF. Background pdf is updated by running average given below:

\[
\mu_{t+1} = \alpha F_t + (1-\alpha)\mu_t
\]
\[
\sigma_{t+1}^2 = \alpha(F_t - \mu_t)^2 + (1-\alpha)\sigma_t^2
\]

\( F_t \) is the current pixel value, \( \bar{b} \) previous average and \( \sigma \) is the covariance. If \( |F_t - \mu| > Th \), This can be chosen as \( k\sigma \) and classified as foreground. Otherwise \( F_t \) will be considered as background. By looking at the above mentioned advantages and disadvantages with its complete information described earlier, the paper concludes that background subtraction method is the better method for motion segmentation because Temporal differencing does a poor job of extracting all relevant feature pixels. Optical flow computation methods are very complex and are inapplicable to real-time algorithms without specialized hardware. Background subtraction is a particularly popular method for motion segmentation especially under those situations with a relatively static background. It attempts to detect moving regions in an image by differencing between current image and a reference background image in a pixel-by-pixel fashion. However, it is extremely sensitive to changes of dynamic scenes due to lighting and extraneous events. So, in order to overcome that problem, the paper presents an solution where background subtraction method can be adapted for dynamic scenes also.

Generally, bgs method make use of Gaussian mixture model for separating foreground images from background scene but it is suitable for static images but fails in case of dynamic images.

The proposed solution is also based on Gaussian elimination method which can also handle dynamic images.

13. Proposed Method

In practice, the illumination in the scene could change gradually (daytime or weather conditions in an outdoor scene) or suddenly (switching light in an indoor scene). A new object could be brought into the scene or a present object removed from it. In order to adapt to changes, the can update the training set by adding new samples and discarding the old ones.

A reasonable time period \( T \) can be choosed and

At time \( t \), \( X_T = \{x^{(t)}, \ldots, x^{(t-T)}\} \)

For each new sample the training data set \( X_T \) is updated and re-estimate \( \hat{p}(\hat{x}|X_T, BG) \). However, among the samples from the recent history there could be some values that belong to the foreground objects and we should denote this estimate as
\[ p(\mathbf{x}^{(t)}|X_T, \text{BG} + \text{FG}) = \sum_{m=1}^{M} \hat{p}_m \mathcal{N}(\mathbf{x}; \hat{\mu}_m, \hat{\sigma}^2_m I) \] (1)

Where \(\hat{\mu}_1, \ldots, \hat{\mu}_m\) are the estimates of the means and \(\hat{\sigma}_1, \ldots, \hat{\sigma}_m\) are the estimates of the variances that describe the Gaussian components. The covariance matrices are assumed to be diagonal and the identity matrix I has proper dimensions. The mixing weights denoted by \(\hat{\pi}_m\) are non-negative and add up to one. Given a new data sample \(\mathbf{x}(t)\) at time t the recursive update equations are:

\[ \hat{\pi}_m - \hat{\pi}_m + \alpha (p^{(t)}_m - \hat{\pi}_m) \] (2)

\[ \hat{\mu}_m - \hat{\mu}_m + o^{(t)}_m (a/\hat{\pi}_m) \hat{\delta}_m \] (3)

\[ \hat{\sigma}^2_m - \hat{\sigma}^2_m + o^{(t)}_m (a/\hat{\pi}_m) (\hat{\delta}_m^T \hat{\delta}_m - \hat{\delta}_m^2) \] (4)

Where \(\hat{\delta}_m = \mathbf{x}^{(t)} - \hat{\mu}_m\). Instead of the time interval T that was mentioned above, constant \(\alpha\) describes an exponentially decaying envelope that is used to limit the influence of the old data. We keep the same notation having in mind that approximately \(\alpha = 1/T\). For a new sample the ownership \(o^{(t)}_m\) is set to 1 for the close component with largest \(\hat{\pi}_m\) and the others are set to zero. We define that a sample is close to a component. The squared distance from the m-th component is calculated as:

\[ D^2_m(\mathbf{x}^{(t)}) = \hat{\delta}_m^T \hat{\delta}_m / \hat{\delta}_m^2 \] (5)

If there are no close components a new component is generated with \(\hat{\pi}_{M+1} = \alpha, \hat{\mu}_{M+1} = \mathbf{x}^{(t)} \) and \(\hat{\delta}_{M+1} = \sigma_0\), where \(\sigma_0\) is some appropriate initial variance. If the maximum number of components is reached, the component can be discarded with smallest \(\hat{\pi}_m\).

The presented method presents an on-line clustering method. Usually, the intruding foreground objects will be represented by some additional clusters with small weights \(\hat{\pi}_m\). Therefore, we can approximate the background model by the first B largest clusters:

\[ \hat{p}(\mathbf{x}|X_T, \text{BG}) = -\sum_{n=1}^B \hat{\pi}_n \mathcal{N}(\mathbf{x}; \hat{\mu}_n, \hat{\sigma}^2_n I) \] (6)

If the components are sorted to have descending weights \(\hat{\pi}_m\) then,

\[ B = \arg \min_b \left( \sum_{n=1}^B \hat{\pi}_n > (1 - cf) \right) \]

where cf is a measure of the maximum portion of the data that can belong to foreground objects without influencing the background model. For example, if a new object comes into a scene and remains static for some time it will probably generate an additional stable cluster. Since the old background is occluded the weight \(\pi_{B+1}\) of the new cluster will be constantly increasing. If the object remains static long enough, its weight becomes larger than cf and it can be considered to be part of the background. Equation 2 helps to conclude that the object should be static for approximately \(\log(1-cf) / \log(1-\alpha)\) frames.

14. Conclusion and future enhancement

Conclusion

This paper has discussed an best possible approach, includes possible main steps in real-time human motion detection. The parts of the human motion detection, motion segmentation with its methods were described. By reading and referring many research papers, this paper concludes that the background subtraction method is best for motion segmentation. This paper proposed the solution for the problem of background subtraction method. An ideal background subtraction could produce good results while foreground regions in motions. This paper successfully provided with advantages and disadvantages of every method and detailed information, so that it helps to choose the best method for motion segmentation.

Future enhancement

There are a variety of enhancements that could be made to this system to achieve greater detection accuracy and increased robustness:

The current thesis paper can further refine by including how actually each method will work by using the respective software.

Various test cases can be included by taking any image as an input, so that better picture can be obtained.

This paper can be further enhanced by including the best methods for remaining human motion detection sections.

Bibliography


[3] www.ensta-paristech.fr/~manzaner/.../Tutorial_HumanMotion_LACN...

[4] En.wikipedia.org/wiki/Motion_analysis
Computer vision

Computer vision is a field that includes methods for acquiring, processing, analysing, and understanding images and, in general, high-dimensional data from the real world in order to produce numerical or symbolic information, e.g., in the forms of decisions. A theme in the development of this field has been to duplicate the abilities of human vision by electronically perceiving and understanding an image. This image understanding can be seen as the disentangling of symbolic information from image data using models constructed with the aid of geometry, physics, statistics, and learning theory. Computer vision has also been described as the enterprise of automating and integrating a wide range of processes and representations for vision perception.

Applications range from tasks such as industrial machine vision systems which, say, inspect bottles speeding by on a production line, to research into artificial intelligence and computers or robots that can comprehend the world around them. The computer vision and machine vision fields have significant overlap. Computer vision covers the core technology of automated image analysis which is used in many fields. Machine vision usually refers to a process of combining automated image analysis with other methods and

APPENDIX

Computer vision

Computer vision is a field that includes methods for acquiring, processing, analysing, and understanding images and, in general, high-dimensional data from the real world in order to produce numerical or symbolic information, e.g., in the forms of decisions. A theme in the development of this field has been to duplicate the abilities of human vision by electronically perceiving and understanding an image. This image understanding can be seen as the disentangling of symbolic information from image data using models constructed with the aid of geometry, physics, statistics, and learning theory. Computer vision has also been described as the enterprise of automating and integrating a wide range of processes and representations for vision perception.

Applications range from tasks such as industrial machine vision systems which, say, inspect bottles speeding by on a production line, to research into artificial intelligence and computers or robots that can comprehend the world around them. The computer vision and machine vision fields have significant overlap. Computer vision covers the core technology of automated image analysis which is used in many fields. Machine vision usually refers to a process of combining automated image analysis with other methods and
technologies to provide automated inspection and robot
guidance in industrial applications.

As a scientific discipline, computer vision is concerned with
the theory behind artificial systems that extract information
from images. The image data can take many forms, such as
video sequences, views from multiple cameras, or multi-
dimensional data from a medical scanner.

As a technological discipline, computer vision seeks to apply
its theories and models to the construction of computer vision
systems. Examples of applications of computer vision include
systems for:

- Controlling processes, e.g., an industrial robot;
- Navigation, e.g., by an autonomous vehicle or mobile
  robot;
- Detecting events, e.g., for visual surveillance or
  people counting;
- Organizing information, e.g., for indexing databases
  of images and image sequences;
- Modeling objects or environments, e.g., medical
  image analysis or topographical modeling;
- Interaction, e.g., as the input to a device for
  computer-human interaction, and
- Automatic inspection, e.g., in manufacturing
  applications.

Sub-domains of computer vision include scene reconstruction,
event detection, video tracking, object recognition, learning,
indexing, motion estimation, and image restoration. In most
practical computer vision applications, the computers are pre-
programmed to solve a particular task, but methods based on
learning are now becoming increasingly common. Object
recognition — one or several pre-specified or learned objects
or object classes can be recognized, usually together with their
2D positions in the image or 3D poses in the scene. Google
Goggles provides a stand-alone program illustration of this
function. Identification an individual instance of an object is
recognized. Examples include identification of a specific
person's face or fingerprint, or identification of a specific
vehicle. Detection the image data are scanned for a specific
condition. Examples include detection of possible abnormal
cells or tissues in medical images or detection of a vehicle in
an automatic road toll system. Detection based on relatively
simple and fast computations is sometimes used for finding
smaller regions of interesting image data which can be further
analyzed by more computationally demanding techniques to
produce a correct interpretation. There are, typical functions
which are found in many computer vision systems.

Image acquisition: A digital image is produced by one or
several image sensors, which, besides various types of light-
sensitive cameras, include range sensors, tomography devices,
radar, ultra-sonic cameras, etc. Depending on the type of
sensor, the resulting image data is an ordinary 2D image, a 3D
volume, or an image sequence. The pixel values typically
correspond to light intensity in one or several spectral bands
(gray images or colour images), but can also be related to
various physical measures, such as depth, absorption or
reflectance of sonic or electromagnetic waves, or nuclear
magnetic resonance.

Pre-processing: Before a computer vision method can be
applied to image data in order to extract some specific piece of
information, it is usually necessary to process the data in order
to assure that it satisfies certain assumptions implied by the
method. Scale-space representation to enhance image
structures at locally appropriate scales.

Feature extraction: Image features at various levels of
complexity are extracted from the image data. Typical
examples of such features are Lines, edges and ridges,
Localized interest points such as corners, blobs or points.
More complex features may be related to texture, shape or
motion.

Detection/segmentation: At some point in the processing a
decision is made about which image points or regions of the
image are relevant for further processing.

High-level processing: At this step the input is typically a
small set of data, for example a set of points or an image
region which is assumed to contain a specific object.

Decision making: Making the final decision required for the
application.

Applications of computer vision

One of the most prominent application fields is medical
computer vision or medical image processing. This area is
characterized by the extraction of information from image data
for the purpose of making a medical diagnosis of a patient.
Generally, image data is in the form of microscopy images, X-
ray images, angiography images, ultrasonic images, and
tomography images. An example of information which can be
extracted from such image data is detection of tumours,
artherosclerosis or other malign changes. It can also be
measurements of organ dimensions, blood flow, etc. This
application area also supports medical research by providing
new information, e.g., about the structure of the brain, or about
the quality of medical treatments.

A second application area in computer vision is in industry,
sometimes called machine vision, where information is
extracted for the purpose of supporting a manufacturing
process. One example is quality control where details or final
products are being automatically inspected in order to find
defects. Another example is measurement of position and orientation of details to be picked up by a robot arm. Machine vision is also heavily used in agricultural process to remove undesirable food stuff from bulk material, a process called optical sorting.

Military applications are probably one of the largest areas for computer vision. The obvious examples are detection of enemy soldiers or vehicles and missile guidance. More advanced systems for missile guidance send the missile to an area rather than a specific target, and target selection is made when the missile reaches the area based on locally acquired image data. Modern military concepts, such as "battlefield awareness", imply that various sensors, including image sensors, provide a rich set of information about a combat scene which can be used to support strategic decisions. In this case, automatic processing of the data is used to reduce complexity and to fuse information from multiple sensors to increase reliability.

Typical tasks of computer vision

Each of the application areas described above employ a range of computer vision tasks; more or less well-defined measurement problems or processing problems, which can be solved using a variety of methods. Some examples of typical computer vision tasks are presented below.

Recognition

The classical problem in computer vision, image processing, and machine vision is that of determining whether or not the image data contains some specific object, feature, or activity. This task can normally be solved robustly and without effort by a human, but is still not satisfactorily solved in computer vision for the general case — arbitrary objects in arbitrary situations. The existing methods for dealing with this problem can at best solve it only for specific objects, such as simple geometric objects (e.g., polyhedra), human faces, printed or hand-written characters, or vehicles, and in specific situations, typically described in terms of well-defined illumination, background, and pose of the object relative to the camera.

Different varieties of the recognition problem are described in the literature:

- Object recognition — one or several pre-specified or learned objects or object classes can be recognized, usually together with their 2D positions in the image or 3D poses in the scene. Google Goggles provides a stand-alone program illustration of this function.
- Identification — an individual instance of an object is recognized. Examples include identification of a specific person’s face or fingerprint, or identification of a specific vehicle.
- Detection — the image data are scanned for a specific condition. Examples include detection of possible abnormal cells or tissues in medical images or detection of a vehicle in an automatic road toll system. Detection based on relatively simple and fast computations is sometimes used for finding smaller regions of interesting image data which can be further analyzed by more computationally demanding techniques to produce a correct interpretation.

Several specialized tasks based on recognition exist, such as:

- Content-based image retrieval — finding all images in a larger set of images which have a specific content. The content can be specified in different ways, for example in terms of similarity relative a target image (give me all images similar to image X), or in terms of high-level search criteria given as text input (give me all images which contains many houses, are taken during winter, and have no cars in them).
- Pose estimation — estimating the position or orientation of a specific object relative to the camera. An example application for this technique would be assisting a robot arm in retrieving objects from a conveyor belt in an assembly line situation or picking parts from a bin.
- Optical character recognition (OCR) — identifying characters in images of printed or handwritten text, usually with a view to encoding the text in a format more amenable to editing or indexing (e.g. ASCII).
- 2D Code reading Reading of 2D codes such as data matrix and QR codes.
- Facial recognition

Motion analysis

Several tasks relate to motion estimation where an image sequence is processed to produce an estimate of the velocity either at each points in the image or in the 3D scene, or even of the camera that produces the images. Examples of such tasks are:

- Ego motion — determining the 3D rigid motion (rotation and translation) of the camera from an image sequence produced by the camera.
- Tracking — following the movements of a (usually) smaller set of interest points or objects (e.g., vehicles or humans) in the image sequence.
- Optical flow — to determine, for each point in the image, how that point is moving relative to the image
Scene reconstruction

Given one or (typically) more images of a scene, or a video, scene reconstruction aims at computing a 3D model of the scene. In the simplest case the model can be a set of 3D points. More sophisticated methods produce a complete 3D surface model.

Image restoration

The aim of image restoration is the removal of noise (sensor noise, motion blur, etc.) from images. The simplest possible approach for noise removal is various types of filters such as low-pass filters or median filters. More sophisticated methods assume a model of how the local image structures look like, a model which distinguishes them from the noise. By first analysing the image data in terms of the local image structures, such as lines or edges, and then controlling the filtering based on local information from the analysis step, a better level of noise removal is usually obtained compared to the simpler approaches. An example in this field is the inpainting.

Object Recognition

Object Recognition: Object Recognition in computer vision, this is the task of finding a given object in an image or video sequence. Humans recognize a multitude of objects in images with little effort, despite the fact that the image of the objects may vary somewhat in different view points, in many different sizes / scale or even when they are translated or rotated. Objects can even be recognized when they are partially obstructed from view. This task is still a challenge for computer vision systems in general.

Object recognition can be described as all of the following:

 Artificial Intelligence
 Applications of Artificial Intelligence
 Pattern recognition
 Subfield of computer vision research

Artificial Intelligence

Artificial intelligence (AI) is the intelligence of machines and the branch of computer science that aims to create it. AI textbooks define the field as the study and design of intelligent agents where an intelligent agent is a system that perceives its environment and takes actions that maximize its chances of success. AI research is highly technical and specialized, deeply divided into subfields that often fail to communicate with each other.[5] Some of the division is due to social and cultural factors: subfields have grown up around particular institutions and the work of individual researchers. AI research is also divided by several technical issues. There are subfields which are focussed on the solution of specific problems, on one of several possible approaches, on the use of widely differing tools and towards the accomplishment of particular applications. The central problems of AI include such traits as reasoning, knowledge, planning, learning, communication, perception and the ability to move and manipulate objects. General intelligence (or "strong AI") is still among the field's long term goals. Currently popular approaches include statistical methods, computational intelligence and traditional symbolic AI. There are an enormous number of tools used in AI, including versions of search and mathematical optimization, logic, methods based on probability and economics, and many others.

Applications of Artificial Intelligence

Artificial Intelligence has been used in a wide range of fields including medical diagnosis, stock trading, robot control, law, scientific discovery and toys. However, many AI applications are not perceived as AI: "A lot of cutting edge AI has filtered into general applications, often without being called AI because once something becomes useful enough and common enough it's not labeled AI anymore," Nick Bostrom reports.[1] "Many thousands of AI applications are deeply embedded in the infrastructure of every industry."[2] The late 90s and early 21st century, AI technology became widely used as elements of larger systems[2][3] but the field is rarely credited for these successes.

In machine learning, pattern recognition is the assignment of a label to a given input value. An example of pattern recognition is classification, which attempts to assign each input value to one of a given set of classes (for example, determine whether a given email is "spam" or "non-spam"). However, pattern recognition is a more general problem that encompasses other types of output as well. Other examples are regression, which assigns a real-valued output to each input; sequence labeling, which assigns a class to each member of a sequence of values (for example, part of speech tagging, which assigns a part of speech to each word in an input sentence); and parsing, which assigns a parse tree to an input sentence, describing the syntactic structure of the sentence.
Pattern recognition

Pattern recognition algorithms generally aim to provide a reasonable answer for all possible inputs and to do “fuzzy” matching of inputs. This is opposed to pattern matching algorithms, which look for exact matches in the input with pre-existing patterns. A common example of a pattern-matching algorithm is regular expression matching, which looks for patterns of a given sort in textual data and is included in the search capabilities of many text editors and word processors. In contrast to pattern recognition, pattern matching is generally not considered a type of machine learning, although pattern-matching algorithms (especially with fairly general, carefully tailored patterns) can sometimes succeed in providing similar-quality output to the sort provided by pattern-recognition algorithms.

Pattern recognition is studied in many fields, including psychology, psychiatry, ethology, cognitive science, traffic flow and computer science.

Subfield of computer vision research

Computer vision is a field that includes methods for acquiring, processing, analysing, and understanding images and, in general, high-dimensional data from the real world in order to produce numerical or symbolic information, e.g., in the forms of decisions. A theme in the development of this field has been to duplicate the abilities of human vision by electronically perceiving and understanding an image. This image understanding can be seen as the disentangling of symbolic information from image data using models constructed with the aid of geometry, physics, statistics, and learning theory. Computer vision has also been described as the enterprise of automating and integrating a wide range of processes and representations for vision perception.

Applications range from tasks such as industrial machine vision systems which, say, inspect bottles speeding by on a production line, to research into artificial intelligence and computers or robots that can comprehend the world around them. The computer vision and machine vision fields have significant overlap. Computer vision covers the core technology of automated image analysis which is used in many fields. Machine vision usually refers to a process of combining automated image analysis with other methods and technologies to provide automated inspection and robot guidance in industrial applications.

As a scientific discipline, computer vision is concerned with the theory behind artificial systems that extract information from images. The image data can take many forms, such as video sequences, views from multiple cameras, or multi-dimensional data from a medical scanner.

As a technological discipline, computer vision seeks to apply its theories and models to the construction of computer vision systems. Examples of applications of computer vision include:

- Controlling processes, e.g., an industrial robot;
- Navigation, e.g., by an autonomous vehicle or mobile robot;
- Detecting events, e.g., for visual surveillance or people counting;
- Organizing information, e.g., for indexing databases of images and image sequences;
- Modeling objects or environments, e.g., medical image analysis or topographical modeling;
- Interaction, e.g., as the input to a device for computer-human interaction, and
- Automatic inspection, e.g., in manufacturing applications.

Sub-domains of computer vision include scene reconstruction, event detection, video tracking, object recognition, learning, indexing, motion estimation, and image restoration.

In most practical computer vision applications, the computers are pre-programmed to solve a particular task, but methods based on learning are now becoming increasingly common.

Applications

Object recognition methods has the following applications:

- Image panoramas
- Image watermaking
- Global Robot Localization
- Face Detection
- Optical Character Recognition
- Manufacturing Quality Control
- Content based image indexing
- Object counting and monitoring
- Automated vehicle parking system
- Visual positioning and tracking

Gaussian Mixture Model

In statistics, a mixture model is a probabilistic model for representing the presence of sub-populations 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 inferences about the properties of the sub-populations given only observations on the pooled population, without sub-population-identity information. Mixture Models are a type of density model which comprise a number of component functions, usually Gaussian. These component functions are combined to provide a multimodal density. They can be employed to model the colours of an object in order to perform tasks such as real-time colour-based tracking and segmentation. These tasks may be made more robust by generating a mixture model corresponding to background colours in addition to a foreground model, and employing Bayes’ theorem to perform pixel classification. Mixture models are also amenable to effective methods for on-line adaptation of models to cope with slowly-varying lighting conditions.

General mixture model

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

- N random variables corresponding to observations, each assumed to be distributed according to a mixture of K components, with each component belonging to the same parametric family of distributions but with different parameters.
- N corresponding 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, each of which is a probability (a real number between 0 and 1), all of which sum to 1
- A set of K parameters, each specifying the parameter of the corresponding mixture component. In many cases, each "parameter" is actually a set of parameters. For example, observations distributed according to a mixture of one-dimensional Gaussian distributions will have a mean and variance for each component. Observations distributed according to a mixture of V-dimensional categorical distributions will have a vector of V probabilities, collectively summing to 1.

In addition, in a Bayesian setting, the mixture weights and parameters will themselves be random variables, and prior distributions will be placed over the variables. In such a case, the weights are typically viewed as a K-dimensional random vector drawn from a Dirichlet distribution (the conjugate prior of the categorical distribution), and the parameters will be distributed according to their respective conjugate priors.

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

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

In a Bayesian setting, all parameters are associated with random variables, as follows:

\[ K, N = \text{as above} \]
\[ \theta_{i=1..K}, \phi_{i=1..K} = \text{as above} \]
\[ z_{i=1..N}, x_{i=1..N}, F(x|\theta) = \text{as above} \]
\[ \alpha = \text{shared hyperparameter for component parameters} \]
\[ \beta = \text{shared hyperparameter for mixture weights} \]
\[ H(\theta|\alpha) = \text{prior probability distribution of component parameters, parametrized on } \alpha \]
\[ \theta_{i=1..K} \sim H(\alpha) \]
\[ \phi \sim \text{Symmetric-Dirichlet}_K(\beta) \]
\[ z_{i=1..N} \sim \text{Categorical}(\phi) \]
\[ x_{i=1..N} \sim F(\theta) \]

This characterization uses F and H to describe arbitrary distributions over observations and parameters, respectively. Typically H will be the conjugate prior of F. The two most common choices of F are Gaussian aka "normal" (for real-valued observations) and categorical (for discrete observations). Other common possibilities for the distribution of the mixture components are:

- Binomial distribution, for the number of "positive occurrences" (e.g., successes, yes votes, etc.) given a fixed number of total occurrences
- Multinomial distribution, similar to the binomial distribution, but for counts of multi-way occurrences (e.g., yes/no/may be)
- Negative binomial distribution, for binomial-type observations but where the quantity of interest is the number of failures before a given number of successes occurs
• Poisson distribution, for the number of occurrences of an event in a given period of time, for an event that is characterized by a fixed rate of occurrence
• Exponential distribution, for the time before the next event occurs, for an event that is characterized by a fixed rate of occurrence
• Log-normal distribution, for positive real numbers that are assumed to grow exponentially, such as incomes or prices
• Multivariate normal distribution, for vectors of correlated outcomes that are individually Gaussian-distributed
• A vector of Bernoulli-distributed values, corresponding, e.g., to a black-and-white image, with each value representing a pixel.
• Non-Bayesian Gaussian 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.

\[ K, N \quad \text{as above} \]
\[ \phi_{i=1...K}, \phi \quad \text{as above} \]
\[ z_{i=1...N}, x_{i=1...N} \quad \text{as above} \]
\[ \mu_{i=1...K} \quad \text{mean of component } i \]
\[ \sigma^2_{i=1...K} \quad \text{variance of component } i \]
\[ \nu, \sigma_0^2 \quad \text{shared hyperparameters} \]
\[ \mu_{i=1...K} \quad \text{mean of component } i \]
\[ \sigma^2_{i=1...K} \quad \text{variance of component } i \]
\[ \nu, \sigma_0^2 \quad \text{shared hyperparameters} \]
\[ \phi \quad \text{as above} \]
\[ \beta \quad \text{concentration hyper parameter of } \phi \]

A Bayesian version of a Gaussian mixture model is as follows:

\[ K, N \quad \text{as above} \]
\[ \phi_{i=1...K}, \phi \quad \text{as above} \]
\[ z_{i=1...N}, x_{i=1...N} \quad \text{as above} \]
\[ \mu_{i=1...K} \quad \text{mean of component } i \]
\[ \sigma^2_{i=1...K} \quad \text{variance of component } i \]
\[ \nu, \sigma_0^2 \quad \text{shared hyperparameters} \]
\[ \phi \quad \text{as above} \]
\[ \beta \quad \text{concentration hyper parameter of } \phi \]

The random variables:

\[ z_{i=1...N} \sim \text{Categorical}(\phi) \]
\[ x_{i=1...N} \sim \text{Categorical}(\theta_{z_{i}}) \]

A typical non-Bayesian mixture model with categorical observations looks like this:

\[ K, N \quad \text{as above} \]
\[ \phi_{i=1...K}, \phi \quad \text{as above} \]
\[ z_{i=1...N}, x_{i=1...N} \quad \text{as above} \]
\[ V \quad \text{dimension of categorical observations, e.g., size of word vocabulary} \]
\[ \theta_{i=1...K, j=1...V} \quad \text{probability for component } i \text{ of observing item } j \]
\[ \theta_{i=1...K} \quad \text{vector of dimension } V \text{ composed of } \theta_{i,1...V} \text{ must sum to 1} \]

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; likewise for [V].

A typical Bayesian mixture model with categorical observations looks like this:

\[ K, N \quad \text{as above} \]
\[ \phi_{i=1...K}, \phi \quad \text{as above} \]
\[ z_{i=1...N}, x_{i=1...N} \quad \text{as above} \]
\[ V \quad \text{dimension of categorical observations, e.g., size of word vocabulary} \]
\[ \theta_{i=1...K, j=1...V} \quad \text{probability for component } i \text{ of observing item } j \]
\[ \theta_{i=1...K} \quad \text{vector of dimension } V \text{ composed of } \theta_{i,1...V} \text{ must sum to 1} \]
\[ \alpha \quad \text{shared concentration hyper parameter of } \theta \text{ for each component} \]
\[ \beta \quad \text{concentration hyper parameter of } \phi \]

Non-Bayesian categorical mixture model using plate notation:

\[ K, N \quad \text{as above} \]
\[ \phi_{i=1...K}, \phi \quad \text{as above} \]
\[ z_{i=1...N}, x_{i=1...N} \quad \text{as above} \]
\[ V \quad \text{dimension of categorical observations, e.g., size of word vocabulary} \]
\[ \theta_{i=1...K, j=1...V} \quad \text{probability for component } i \text{ of observing item } j \]
\[ \theta_{i=1...K} \quad \text{vector of dimension } V \text{ composed of } \theta_{i,1...V} \text{ must sum to 1} \]

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; likewise for [V].
The random variables:

\[ \phi \sim \text{Symmetric-Dirichlet}_{K}(\xi) \]
\[ \theta_{i=1...K} \sim \text{Symmetric-Dirichlet}_{V}(c) \]
\[ z_{i=1...N} \sim \text{Categorical}(\phi) \]
\[ x_{i=1...N} \sim \text{Categorical}(\theta_{z_{i}}) \]

A Gaussian Mixture Model (GMM) is a parametric probability density function represented as a weighted sum of Gaussian component densities. GMMs are commonly used as a parametric model of the probability distribution of continuous measurements or features in a biometric system, such as vocal-tract related spectral features in a speaker recognition system. GMM parameters are estimated from training data using the iterative Expectation-Maximization (EM) algorithm or Maximum A Posteriori (MAP) estimation from a well-trained prior model.