Thesis

Performance evaluation of feature sets for carried object detection in still images

Submitted by

Hrushikesh N. Kulkarni

Department of Electrical and Computer Engineering

In partial fulfillment of the requirements

For the Degree of Master of Science

Colorado State University

Fort Collins, Colorado

Summer 2014

Master's Committee:

Advisor: J. Ross Beveridge
Co-Advisor: Bruce A. Draper

Sudeep Pasricha
David G. Alciatore

Abstract

Performance evaluation of feature sets for carried object detection in still images

Human activity recognition has gathered a lot of interest. The ability to accurately detect carried objects on human beings will directly help activity recognition. This thesis performs evaluation of four different features for carried object detection. To detect carried objects, image chips in a video are extracted by tracking moving objects using an off the shelf tracker. Pixels with similar colors are grouped together by using a superpixel segmentation algorithm. Features are calculated with respect to every superpixel, encoding information regarding their location in the track chip, shape of the superpixel, pose of the person in the track chip, and appearance of the superpixel. ROC curves are used for analyzing the detection of a superpixel as a carried object using these features individually or in a combination. These ROC curves show that the detection using Shape features as they are calculated have very less information. The location features, though simple to calculate, have a significant usable information. Detection using pose of a person in the track chip and appearance of the superpixel depend largely on the data used for their calculation. Pose detections are more likely to be correct if there are no occlusions, while appearance work better if we have high resolution of input images.

Acknowledgements

I have been fortunate to work with people who are bright, smart, and hard working. I thank Dr. Draper for being patient, and giving me the time to explore. Dr. Beveridge for reminding me to look at the broader picture. Dr. Maciejewski, Dr. Sega, and Dr. Young for employing me and giving me the timely guidance throughout my graduate career. Dr. Rajopadhye and Dr. Alciatore for being very supportive throughout the journey. Dr. Pasricha for his timely help. ECE and CS staff for making special administrative accommodations for me. ENS and SNA staff for maintaining all the computing resources in perfect shape.

I thank my Aai-Baba (parents) for never giving up on me and cheering me whenever required. Anna-Ajji (grandparents) who have always been the guiding light for me. Raju-kaka for being a corner stone of all my education advances. Sandeep-mama for giving me the timely guidance throughout my career. Sarmukaddam-kaka and Shailu-dada for their prompt help when it was required the most.

My labmates, Som-raju, Wimroy-Anna, Pradyumna-gari, Jatin-bhai, Dutta, Jindal, have been instrumental throughout my work. Mo and Maggie have always been open to all my questions and gave me valuable suggestions when I was doing experiments. Vamshi, Bhavesh, and Karthik, without whom my coursework would not have been enjoyable.

I found a home away from home within my near and dear friends i.e. “Manor-mates”. I spent the best days of my life with them. Thank you Manish, Mihir, Ishan, Kishor, Aditya, Mayank and many others. Special thanks to Kaustubh, Saket, Harshad, Ketki, Tejasi and Poonam, for being such wonderful friends, and for always being there when needed. I had a special association with the ISA, and I thank them for making my stay at CSU memorable. My friends and teachers from far back in school deserve thanks as well.

DEDICATION

To my parents, and to these beautiful mountains.

Table of Contents

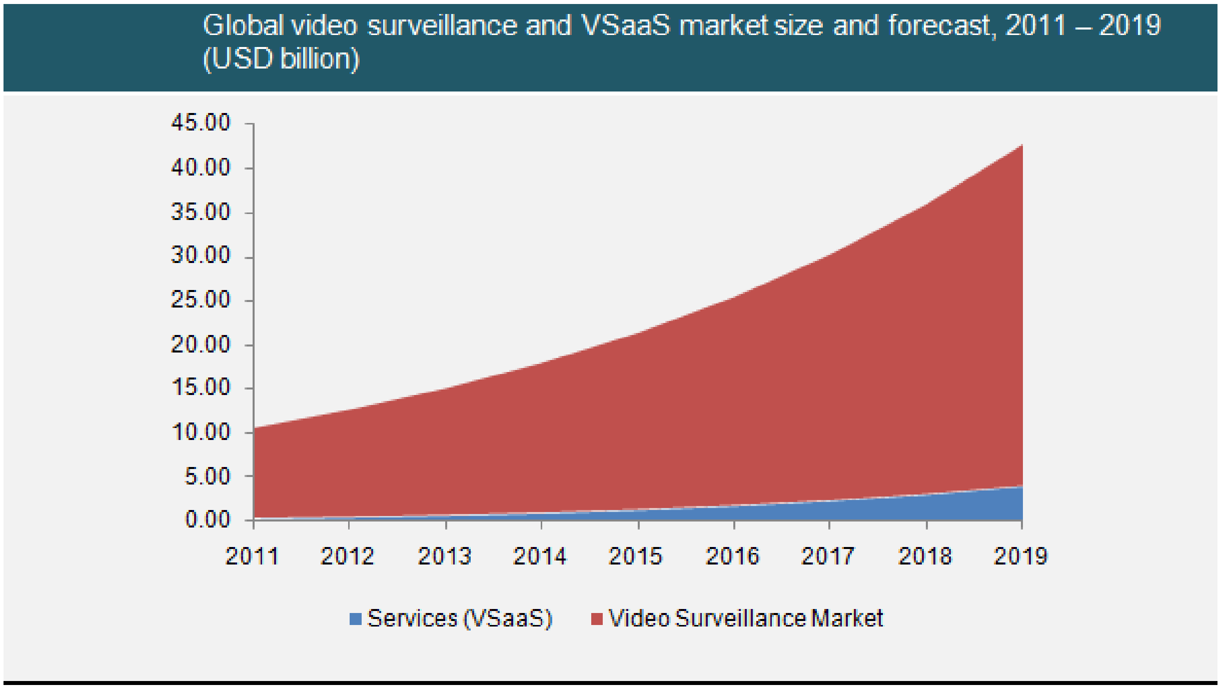
List of Figures

CHAPTER 1

Introduction

In recent years, surveillance cameras have gained phenomenal world wide use. For example, the Domain Awareness System developed by a joint effort with the City of New York and Microsoft deploys close to 3000 cameras [14], allowing security personnel a better view of the city. Another article from the Telegraph [2], reports figures, from a survey carried out in Great Britain, stating that it has close to 5.9 million CCTV cameras.

A survey carried out by Transparency Market Research indicates an exponential growth rate in the global video surveillance market, as shown in Figure 1.1. We can see that the Video Surveillance as a Service (VSaaS) is growing at a linear rate, while the video surveillance market is growing exponentially. This linear rate is due to the difficulty in automatically



Source: Transparency Market Research

Figure 1.1. Global Video Surveillance and VSaaS market size and forecast, 2011 - 2019, [16]

analyzing video data. The gap between the VSaaS and total video surveillance market indicates the potential scope of development of video analytics. This indicates that there is a huge demand on automatic and standalone operations in the video market. The above examples indicate that there is a huge data overload from a network of cameras, and the need to automatically detect and analyze human activity is unprecedented. Actions in the scene involving people carrying objects attracts more attention for obvious reasons, and automatically detecting carried objects is the heart of this thesis.

Detecting objects carried by people is a difficult and an interesting task. Carried objects are in different shapes, sizes and blend well with the person's clothing. For example, a person carrying a suitcase can be easily confused with a long coat. For the scope of this thesis, only large objects are considered as carried objects. A coffee mug in our hands, or, a person walking his dog are not considered as examples of carried objects.

An important breakthrough in the field of carried object detection was published in an approach commonly known as "Backpack" by Haritaoglu et al. [9] in 1999. The algorithm presented in this [9] approach detects backpacks by monitoring the periodicity of protrusions in the human silhouette. Following this there have been many approaches contributing incremental developments in the detection accuracy. These related approaches for detecting carried objects are discussed in detail in the Literature Review chapter along with component techniques used in this thesis. All published approaches use video information, and there is a requirement to explore the information available in still video frames for detecting carried objects. Filling this gap in the literature of detecting carried objects by using only still frames in a video is the core contribution of this thesis. This thesis also provides a baseline performance metric for gauging the performance of other complex techniques. Hence any complex technique should exceed the performance of detecting carried objects using still images.

To detect carried objects in still images, all moving objects from a video are tracked using an off the shelf tracker, and image crops are extracted in every frame around the moving object. These image crops are known as image tracks, and are used as an input for further processing. Pixels with similar color are grouped together, such a group is called as superpixel. Image features relative to these super pixels are calculated encoding information regarding the location of this super pixel in the image track, shape of this superpixel, appearance of the superpixel, and pose of a person in the image track. Location features encode likelihood of finding a carried object at a given location in the image track. Pose features encode different pose configurations for carrying an object. Appearance and shape features encode appearance and shapes properties of carried objects.

Chapter 2 illustrates different key approaches for detecting carried objects on people, along with important concepts used in this thesis. The process of extracting image chips from videos, calculation of image features, and the methodology of detection is discussed in Chapter 3. Performance of features encoding information about location, pose, shape, and appearance for detecting carried objects is evaluated in Chapter 4 using ROC plots. Conclusion from these results, and possible improvements are discussed in Chapter 5.

CHAPTER 2

Literature Review

There are two major schools of thought in the field of carried object detection. One school of thought detects *people carrying objects* by analyzing differences in human gait and motion patterns. The other school of thought detects *carried objects on people* by segmenting protrusions in the human silhouette and analyzes these protrusions according to periodicity and its temporal association with the foreground silhouette. The Figure 2.1 illustrates a layout of the literature, and overlapping concepts.

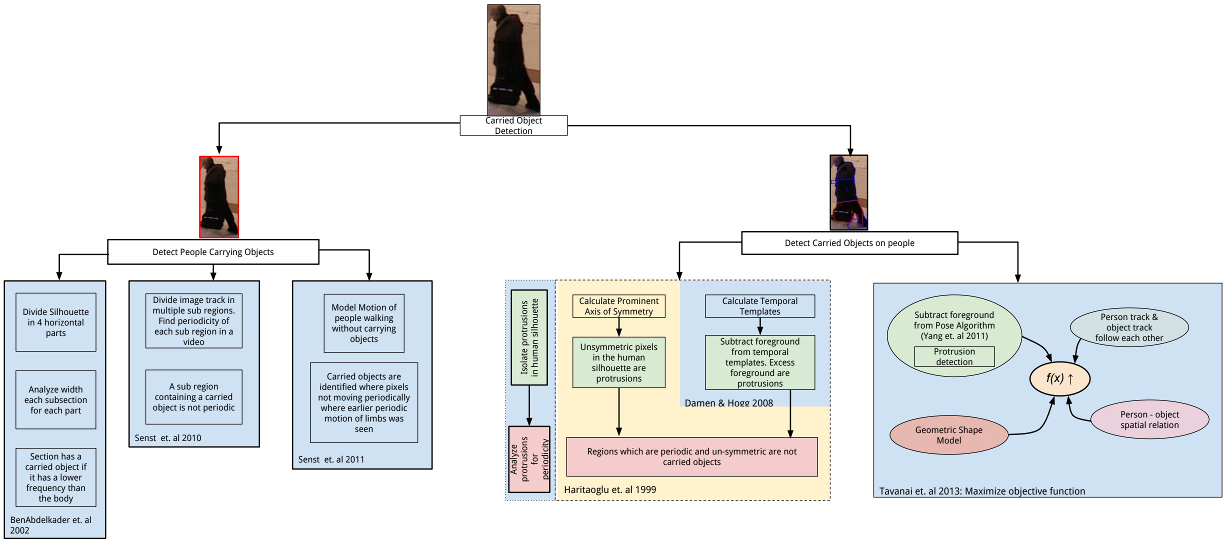

Carried Object Detection
Detect People Carrying Objects
Divide Silhouette in 4 horizontal parts
Analyze width each subsection for each part
Section has a carried object if it has a lower frequency than the body
BenAbdelkader et. al 2002
Divide image track in multiple sub regions. Find periodicity of each sub region in a video
A sub region containing a carried object is not periodic
Senst et. al 2010
Model Motion of people walking without carrying objects
Carried objects are identified where pixels not moving periodically where earlier periodic motion of limbs was seen
Senst et. al 2011
Detect Carried Objects on people
Isolate protrusions in human silhouette
Analyze protrusions for periodicity
Calculate Prominent Axis of Symmetry
Unsymmetric pixels in the human silhouette are protrusions
Calculate Temporal Templates
Subtract foreground from temporal templates. Excess foreground are protrusions
Damen & Hogg 2008
Regions which are periodic and un-symmetric are not carried objects
Haritaoglu et. al 1999
Subtract foreground from Pose Algorithm (Yang et. al 2011)
Protrusion detection
Person track & object track follow each other
f(x) ↑
Geometric Shape Model
Person - object spatial relation
Tavanai et. al 2013: Maximize objective function

Figure 2.1. Schools of Thought for Carried Object Detection

The first school of thought is expressed in the work by BenAbdelkader and Davis [3], Senst et al. [18] uses differences in human gait when carrying an object and when not carrying an object. In another approach Senst et al. [17] observe that humans, when walking, exhibit two different types of motion. Head and torso have a uniform direction of motion, while arms and legs show periodic changes, they also argue that people carrying objects do not follow this motion pattern.

In the second school of though there are two major steps for detecting carried objects. The first step is to segment different foreground regions, and the second step is to check if these segments show any periodic behavior over time or if they have any spacial relationship with other foreground regions. Haritaoglu et al. [9] and Damen et al. [6] assume that carried objects are protrusions in the human silhouette, and classify protrusions which do not show periodic properties as carried objects. Tavanai et al. [20] segment convex and elongated foreground regions, and classify them as carried objects if they follow a spatial relationship with the foreground.

Both schools of thought as summarized above use information from multiple frames in a video, either for gait analysis or to check if a particular region is periodic. In contrast, the algorithm discussed in this thesis aims at analyzing different features calculated using only single image frames, with an emphasis on performance gain or lost. Detection performance is compared with other published works. The following sections discuss previously published approaches and other important component techniques.

2.1. Backpack: Detection of People Carrying Objects Using Silhouettes

A system commonly referred to as "Backpack" [9] was one the first works published for detecting carried objects on human beings. Authors of this work assume that the human body is roughly symmetric in shape and shows periodic changes. The sections below discuss key contributions and shortcomings of this approach.

2.1.1. Symmetry Analysis.

Haritaoglu et al. [9] assume that the human silhouette is symmetric about a body axis when unencumbered. To calculate this body axis principal component analysis is performed on silhouette pixel locations to find out the axis corresponding to the maximum eigen vector. An axis parallel to the axis just calculated and passing through the median location of all silhouette pixel locations is used as the body axis. Figure 2.2 shows a silhouette of a person extracted from PETS2006 dataset and walking perpendicular to the camera.

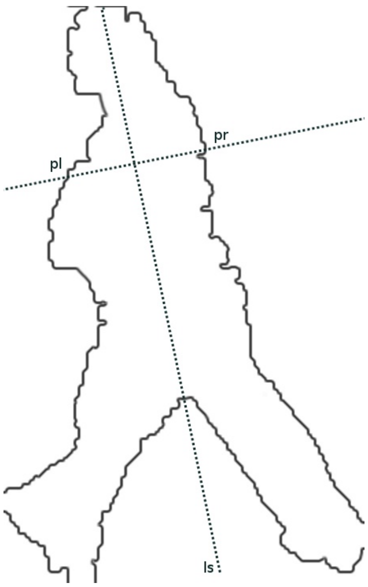


FIGURE 2.2. Symmetric and Antisymmetric Analysis.

In Figure 2.2, l^s is the body axis, and p_l, p_r are a pair of points on the silhouette such that $p_l p_r$ is $\perp$ to l^s. q_i^l and q_i^r denote the length of the line segment $[p_l, p_s]$ and $[p_r, p_s]$ respectively. The length of the line segment from p_s to x is denoted as q_s^x. Hairtaoglu et al. [9] classify each pixel x on the silhouette as symmetric or non-symmetric according to the following equation.

$$x = \begin{cases} \text{Non symmetric} & \text{if } q_s^r > min(q_i^l, q_i^r) + \epsilon \\ \text{Symmetric} & \text{otherwise} \end{cases} \tag{2.1}$$

Regions classified as non-symmetric are further used in periodicity analysis for carried object detection.

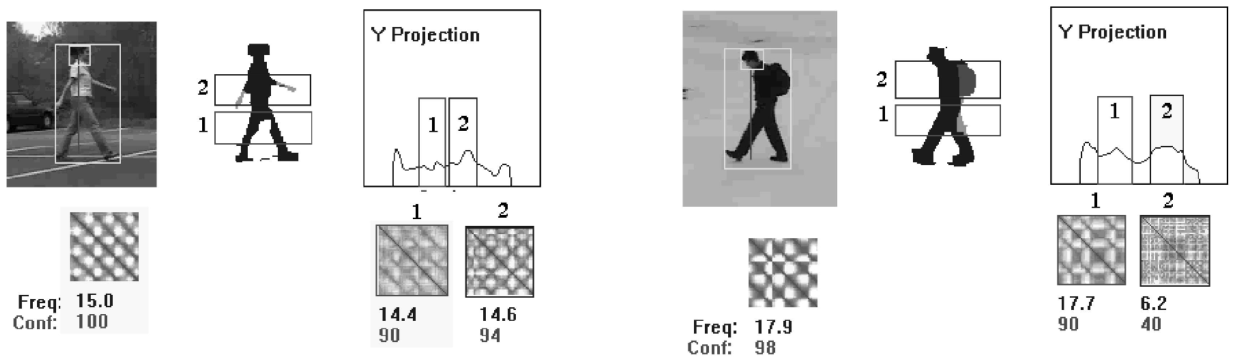


FIGURE 2.3. Periodicity Analysis Source [10]

2.1.2. PERIODICITY ANALYSIS. All non symmetric regions are grouped together and are used for further shape periodicity analysis. Non symmetric regions which do not show significant periodicity properties are classified as carried objects, as shown in Figure 2.3.

The Y projection is divided into two subsections (see 1 and 2 in Figure 2.3). Projected histograms for both subsections are calculated for each frame, and compared with successive frames to find the repetition period by correlation. The repetition of each subsection is now compared with the repetition for the entire body. If it is comparable with the entire body, then the given subsection does not have a carried object. Conversely, repetition for a subsection which is not comparable with the entire body is classified as Carried Objects.

2.1.3. SHORTCOMINGS. Damen and Hogg [6] point out the main sources of errors in detecting carried objects by Haritaoglu et al. [9]. Major source of error is because the position of the body axis as discussed in section 2.1.1 is displaced by the presence of carried objects. Also a number of walking cycles are required to calculate the gait frequency, which is a limitation when objects of interest are not seen for longer durations.

2.2. CARRIED OBJECT DETECTION TEMPORAL TEMPLATE MATCHING

Damen and Hogg [7] extend the work of Haritaoglu et al. [9] and present a very interesting approach for detecting carried objects. Moving objects in the scene are tracked using an off the shelf tracker by Magee et al. [13]. Temporal templates representing motion and shape are calculated by aligning and averaging foreground regions of tracked objects. A library of exemplar temporal templates is also calculated by observing people walk without a carried object. The difference between the tracked temporal template and the library exemplars give likely locations of the carried object. These protrusions are then analyzed for periodicity, with an idea that periodic protrusions correspond to limbs and are not carried objects.

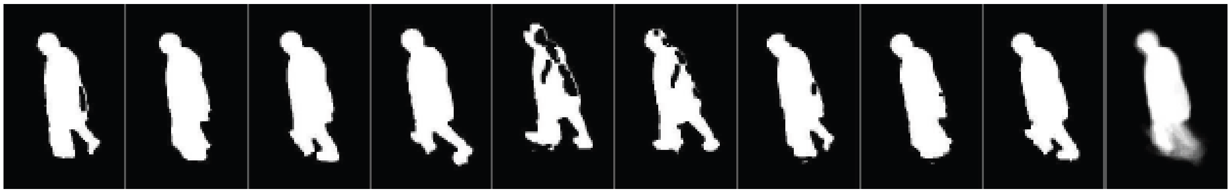


Figure 2.4. Foreground segmentations along with created temporal template in the last frame [7]

.

2.2.1. Building Temporal Templates. A temporal template is created by aligning temporal templates using Iterative Closest Point (ICP) [23] and then averaging the foreground segmentation. Damen and Hogg [6] slice the trajectory in sections 2 seconds long, and assume that people do not change their walking directions during these slices, otherwise the averaged temporal template will not be correctly formed. A temporal template is calculated for every trajectory slice.

Damen and Hogg [6] use the EPFL data set [8] for building exemplar temporal templates. Fig. 2.5 shows the EPFL setup for capturing images of a person walking without a carried object, and being observed from 8 camera view points. The exemplar temporal templates are created by aligning and averaging the silhouettes for each camera view. An example of a exemplar temporal template is shown in Figure 2.6. These exemplars are used for finding out protrusions in the tracked temporal template.

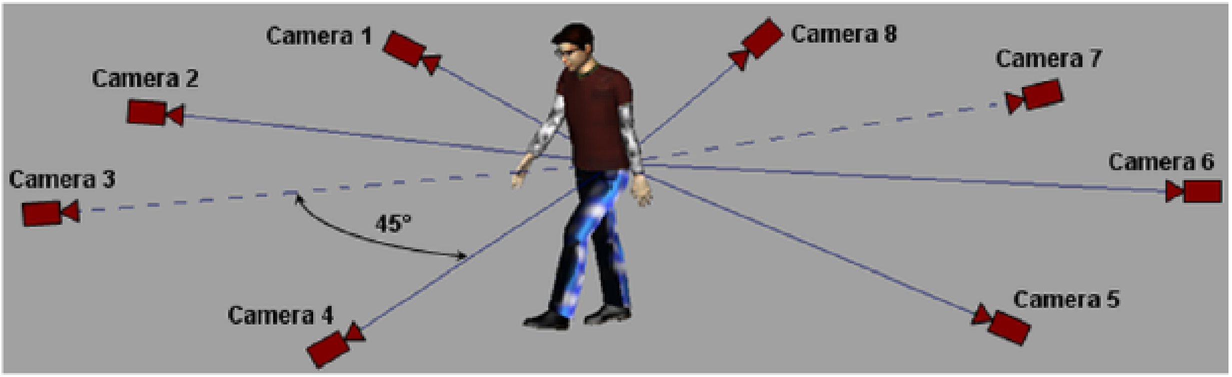


FIGURE 2.5. Camera Setup for capturing spatio temporal templates. [8]

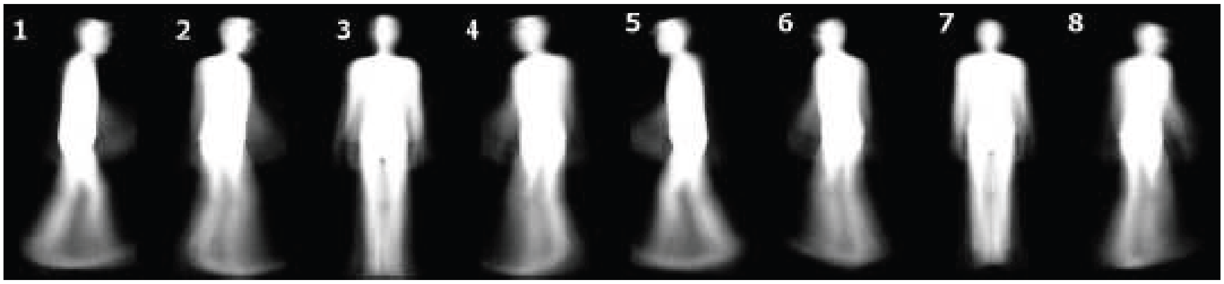


FIGURE 2.6. Exemplars created using EPFL dataset [8, 7]

An exemplar for the trajectory chip is selected from the library which is closest to the track chip using L_1 distance. A higher matching weight is added to the head and shoulder region and lower weight is used for the areas near the feet.

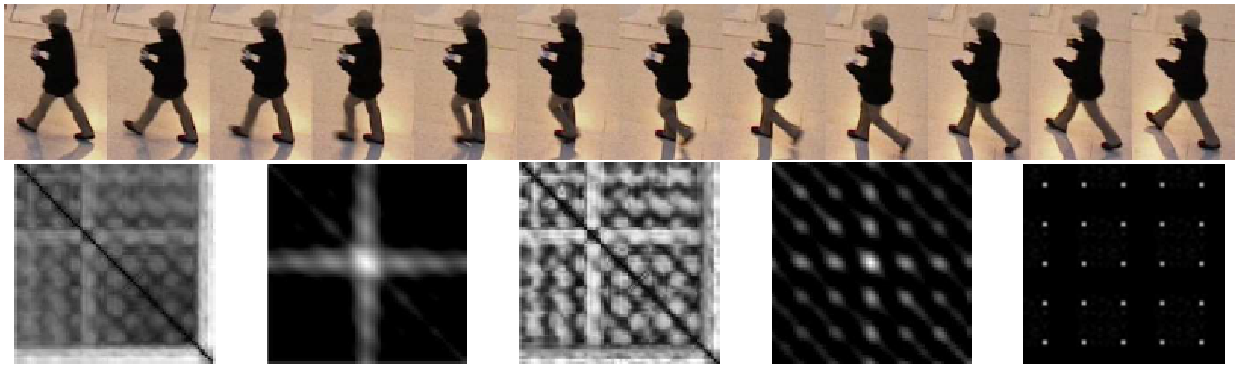


FIGURE 2.7. A walking sequence and a lattice representing the *Similarity Matrix* [5]

2.2.2. Periodicity Analysis. Tracked foreground regions are aligned using Iterative Close Point algorithm [22] and the L_1 distance between two alighted foreground regions is calculated. Periodicity in a sequence of n frames is calculated by creating a $n \times n$ matrix with the L_1 distance between the respective aligned foreground regions. Frequency across a similarity matrix with minimum L_1 scores is the periodicity of the sequence. Two such matrices are created for calculating the periodicity of aligned foreground regions and protrusions. L_1 scores of aligned protrusion regions are calculated using the technique described in Section 2.2.1. Damen and Hogg [6] label a protrusion as a carried object if the frequency of the protrusion is smaller than the frequency of the complete body.

2.3. Motion Based Carried Object Detection

Carrying objects on the body affects the way in which we walk. We walk with higher cadence and shorter strides, and the duration for which our feet tend to be on the ground depend on the weight of the object being carried. BenAbdelkader and Davis [3] look at using these clues for detecting if the person is carrying an object. Shape and periodicity clues are obtained by subdividing the human silhouette into 5 horizontal segments, and observing the temporal behavior of the bounding box width over each segment. Periodicity and amplitudes of the bounding boxes are compared with those of a person walking without a carried object and deviations from this provide evidence that a person may be carrying an object.

Fig.2.8 shows the different divisions on the human silhouette. The width of individual subdivisions is calculated and compared with other frames in the video. Fig.2.9 shows the auto-correlation results.

This method has the following shortcomings

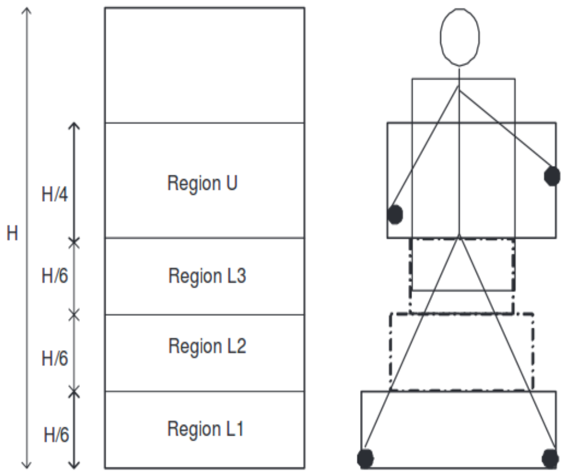
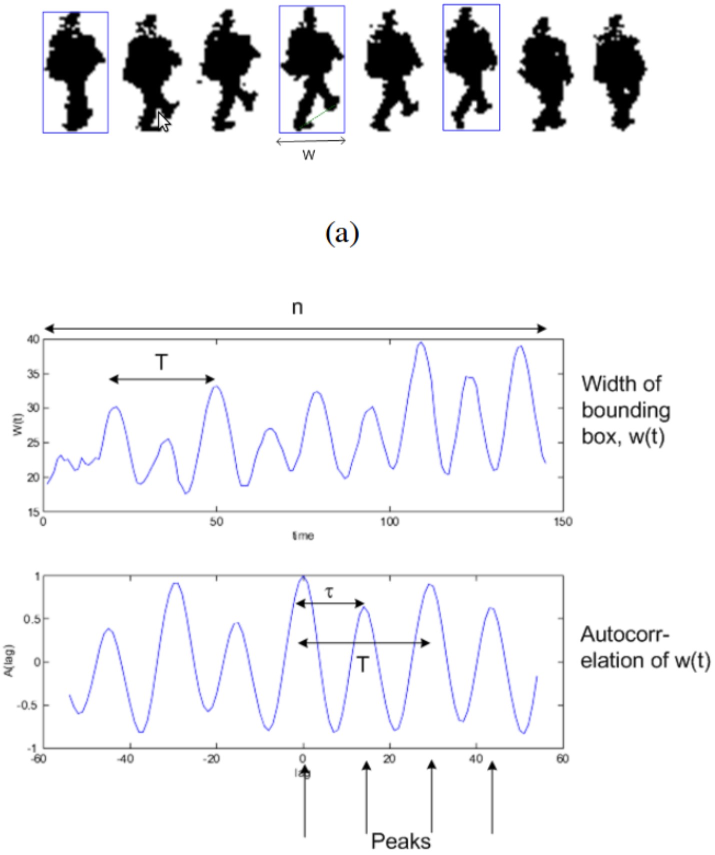


FIGURE 2.8. Subdivisions of the body silhouette [3]



(a)

FIGURE 2.9. Autocorrelation of the width for previous frames [3]

- As the algorithm depends on accurately detecting the human silhouette, it is susceptible to errors if background subtraction does not perform correctly.

- The algorithm does not segment the carried objects, it only detects if the person is carrying an object.
- Since the 4 horizontal segments do not cover the area about the shoulders, it is difficult to identify the cases in which the carried object is carried on the shoulders.

2.4. Carried Object Detection using Geometric Shape Models

In this section we will look at another approach by Tavanai et al. [20] which integrates the geometric shape of the carried object and its association with the person carrying it. A carried object is assumed to have tracks which have constant spatial relationship with the person carrying it. For example a dragged suitcase is at a constant distance from the person, thus maintaining a constant spatial relationship. An objective function which takes into account these parameters is a major contribution of this approach. Figure 2.10 give a block diagram description of approach presented by Tavanai et al. [20]

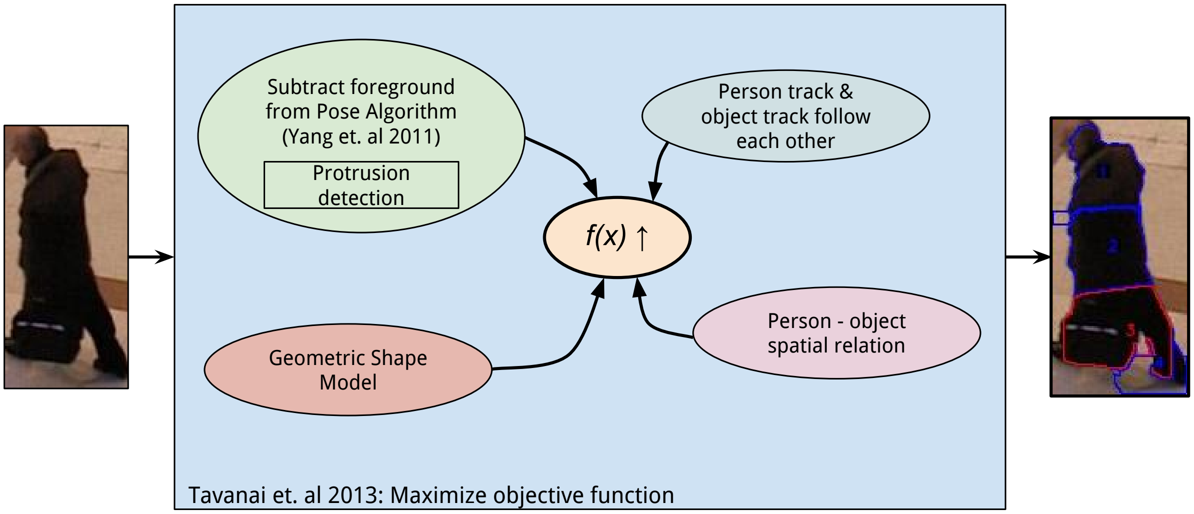

Subtract foreground from Pose Algorithm (Yang et. al 2011)
Protrusion detection
Person track & object track follow each other
f(x) ↑
Geometric Shape Model
Person - object spatial relation
Tavanai et. al 2013: Maximize objective function

Figure 2.10. Carried object detection using Geometric shapes

2.4.1. Geometric Shape Models.

Objects carried by humans fall into two categories: convex shapes for objects like backpacks, suitcases etc., and, elongated shapes for shovels, brooms etc. Convexity measure is calculated by the technique proposed by Zunic and Rosin [24], and denotes the probability of randomly choosing two points inside the polygon and all the points on the line segment between these two points are inside the polygon. To learn the shape of parallel objects a degree of parallelism is calculated. To calculate the degree of parallelism only those contour segments are considered which can be partitioned into line segments which are non-overlapping, collinear, roughly parallel and are close to each other.

2.4.2. Person-Carried Object Relation. A candidate track is most likely to carry an object if it follows the trajectory of the person with a spatio-temporal consistency, and it overlaps with the protrusion. Protrusions are found by subtracting the foreground from the person region estimates predicted by the Articulated Pose Estimation algorithm [21].

2.4.3. Person-Object Spatial Relation. To learn the spatial relation of the person with respect to the object a map of votes is created. This map gives a count of the number of times a given pixel is a carried object relative to the centroid of the person region. Fig.2.11 gives an example of this spatial relation. The first image shows the carried object marked by red, and the following four images show the development of this spatial association when tracked across multiple frames.

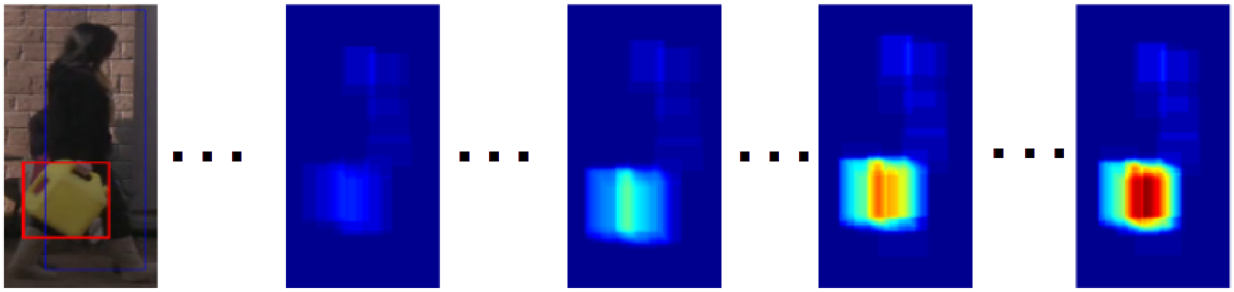


Figure 2.11. Spatial distribution of the object relative to the person centroid
Source: [20]

This approach gives an interesting view of detecting carried objects by optimizing an objective function taking into account the spatial relation between the person and object track, and the geometric shape of protrusion detected by subtracting foreground region from the pose estimates by Yang and Ramanan [21].

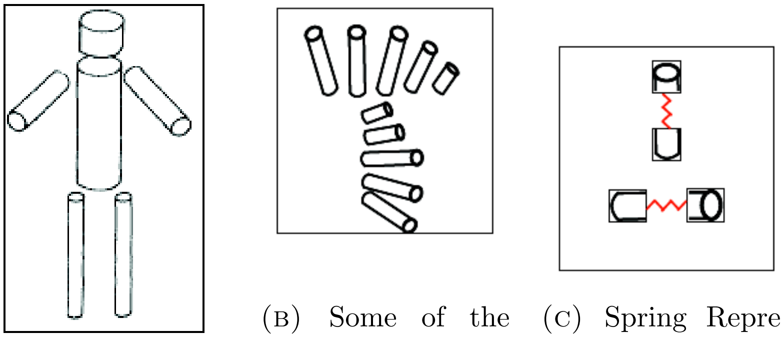
2.5. SLIC Segmentation

The segmentation method used in this thesis is “Simple Linear Iterative Clustering”, commonly referred as *SLIC* by Achanta et al. [1]. It builds on the K-Means clustering approach to form super pixels or segmentation regions. SLIC segmentation has significant advantages over the other segmentation methods, making it an easy choice for superpixel segmentation. Some key advantages are given below.

- There are only two parameters to tune. One defines the number of super pixels on the image, other controls the compactness of the cluster. A smaller value of compactness will force the superpixels to adhere more tightly to the image boundaries. Conversely a larger values values of compactness will force more regular superpixel shapes.
- SLIC has linear $O(N)$ complexity in terms of number of pixels in the image.

2.6. Pose Estimation

There are specific poses which relate to the action of carrying an object. As stated earlier, if we carry a suitcase, our arms will align to support the weight and guide the suitcase. The intuition is to learn the association of carrying objects and the corresponding pose of the human body. For the purpose of learning poses, we use the approach commonly known as articulated pose estimation with flexible mixtures-of-parts, by Yang and Ramanan [21].



(A) A Classicial Articulated Model

(B) Some of the Possible Orientations for Articulated Model

(C) Spring Representations for Near-Vertical and Near-Horizontal Limbs

FIGURE 2.12. Representation of Articulated Pose [21]

Yang and Ramanan [21] use spring models to link different limbs as shown in Figure 2.12, which are constrained with a co-occurrence relationship. For example, a forearm is always connected to the shoulder via the upper arm. A relational tree graph is built to encode these relationships, and detect human pose. The authors of this algorithm also make their source code available, making it an ideal choice for our approach.

CHAPTER 3

Methods

In this chapter, the methodology used for training and testing features based on location, shape and appearance of a segmentation region, and the pose of a person are discussed. Section 3.1 will discuss the method used for extracting image chips from a video. These image chips are used for training and testing purposes. Following the section dealing with data preparation, every section individually deals with training and testing location, shape, pose and appearance features.

3.1. Data Preparation

In this section we will discuss the process of preparing segmented regions on track chips used for calculating image features. As shown in fig. 3.1, every video is passed through a Low Level Vision System (LLVS) which gives two outputs, a track chip, and a foreground-background mask. A track chip is a crop from the video frame containing an object in motion. Segmentation is carried out on this track chip as discussed below.

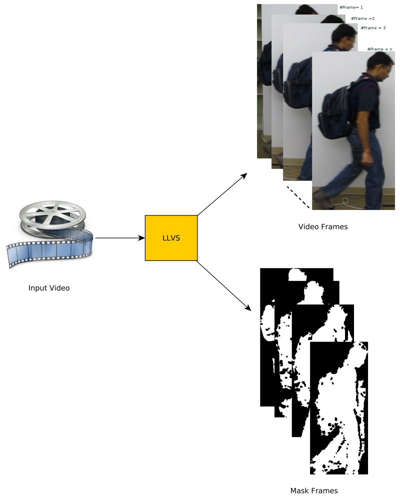


Figure 3.1. LLVS input and output

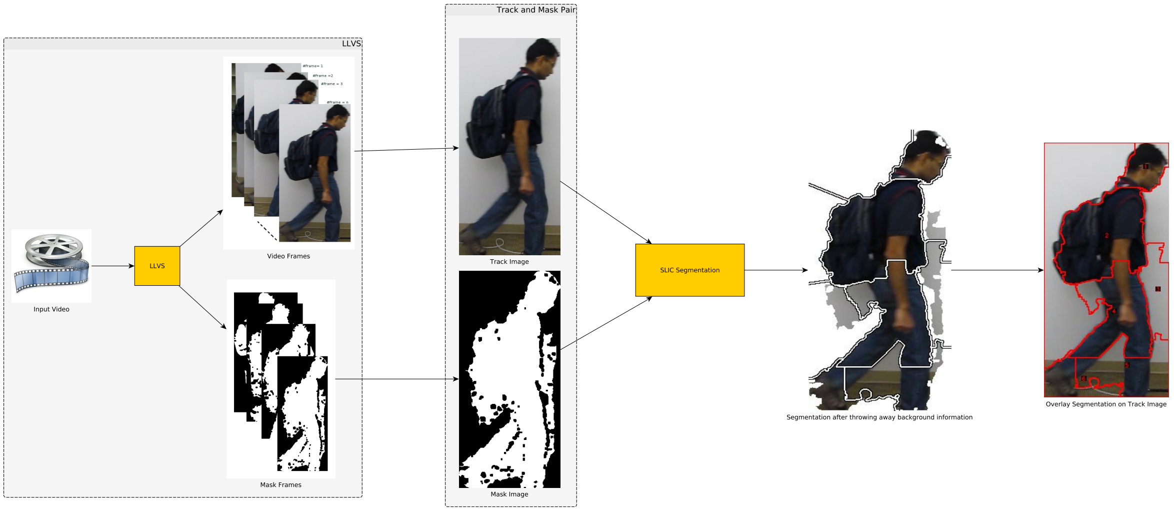


Figure 3.2. Conditioning the input through LLVS and SLIC

As shown in fig. 3.2 the foreground image is passed through the Simple Linear Iterative Clustering algorithm by Achanta et al. [1]. Superpixels from this algorithm serve as a input to the remaining stages of the algorithm.

For every superpixel we find out its likelihood of being a carried object depending on location, shape, pose of the person carrying the object, and appearance of the superpixel. Section 3.3 will discuss how the inputs are used for detecting the carried object depending on the location information, section 3.5 will talk more about how different poses are associated with the location of the carried object, section 3.6 will discuss how the appearance of the carried object is learned, and section 3.4 will tell us about how the shape of the carried object is learned.

3.2. Labeling

Labeling is a process of marking individual segmentation region with one of the three tags: *a*) Person Regions *b*) Worn Carry Regions *c*) Drag Carry Regions .

The ground truth is obtained along with the database, over here PETS2006 [19], which contains a bounding box over the carried object in every frame in the video. The process of labeling is carried in two steps. The first step reads from the ground-truth database and marks the segmented regions which have more than 70% overlap with the ground truth bounding box. The second step qualifies these labels by asking the user if these regions contain an object which was "worn" or "dragged" by the person. These regions get labels as "Worn Carry" or "Dragged Carry".

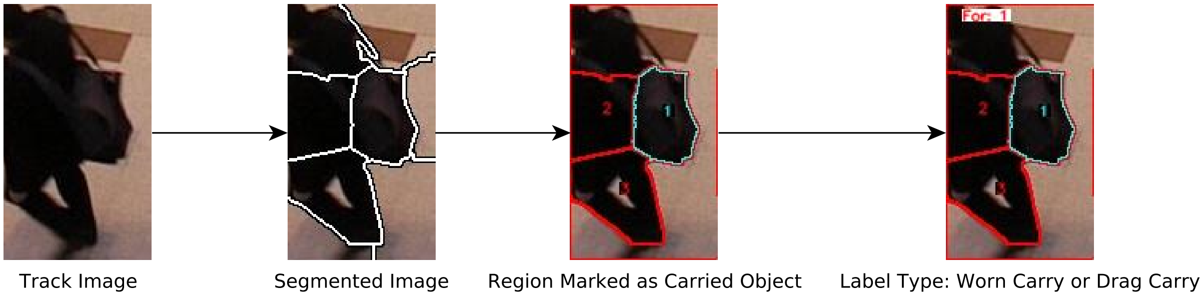

Track Image
Segmented Image
Region Marked as Carried Object
Label Type: Worn Carry or Drag Carry

Figure 3.3. Labeling Steps

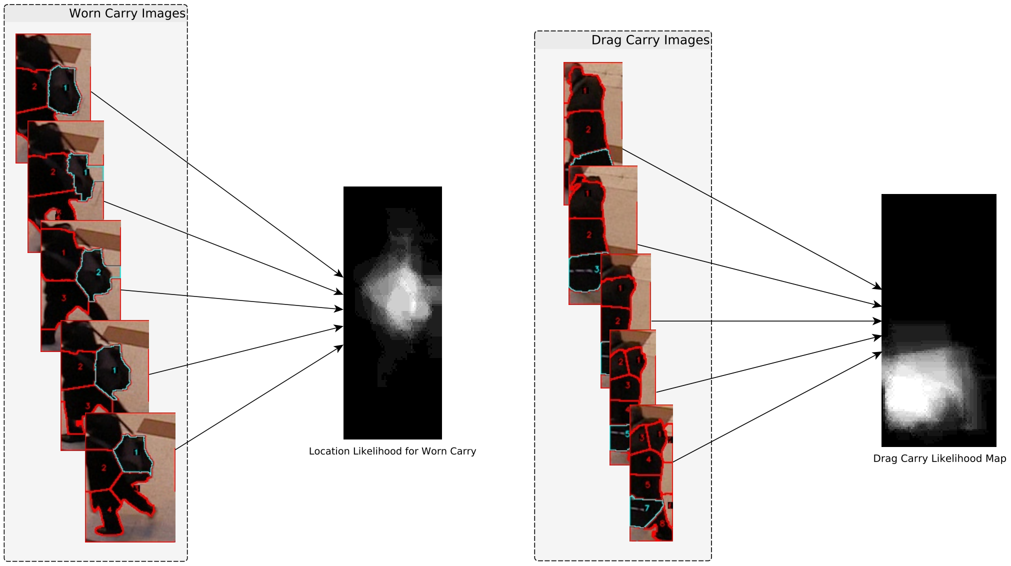
The Figure 3.3 above shows the process of labeling track chips. First, the track chips are segmented as given in section 3.1 and in this case, region #1 is marked as a carry region. Later the region is hand labeled as "Worn Carry Object" by asking the label specification to the user. The following sections make use of these labels for calculating features and building models for detecting carried objects.

3.3. Location based Detection

There are certain locations on the human body where a person is more likely to carry an object. For example a suitcase will be near our feet, and a backpack will be on your back. In this section, a process for learning these favorable locations is described in detail. All track chips are resized to 140 × 70 for convenience.

3.3.1. Learning Favorable Locations.

Figure 3.4 shows a pictorial representation for calculating likelihood of a particular location being a Carry Region. Let a location



(A) Location Likelihood for Worn Carry Object

(B) Location Likelihood for Drag Carry Object

FIGURE 3.4. Location Likelihood of Worn and Drag Carry Objects after resizing the images.

likelihood image be defined as $L(x, y)$. Hence, $L_{worn}(x, y)$, $L_{drag}(x, y)$ and $L_{person}(x, y)$ are the likelihoods encoding a particular location being a “Worn Carry Region”, a “Drag Carry Region” and a “Person Region”. Every pixel (x, y) in this image encodes the likelihood of the pixel belonging to a particular region based on the training labels. Let N be the total number of labeled images defined as follows.

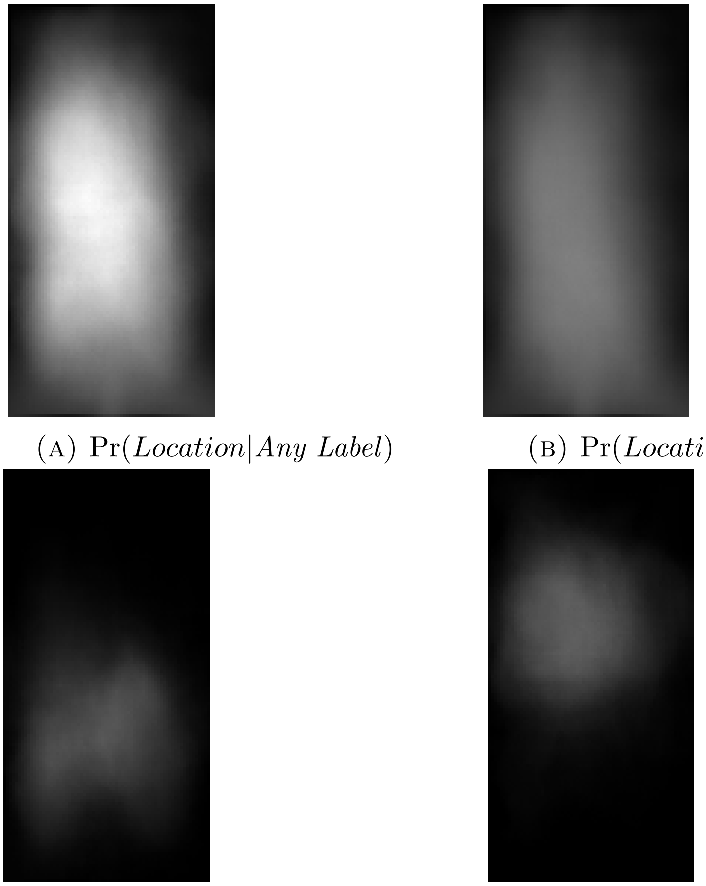
$$N = |\text{Images labelled Worn}| + |\text{Images labelled Drag}| \tag{3.1}$$

$$L_{Worn}(x, y) = \frac{\sum_{label=\text{Worn}} I(x, y) \in Superpixel_{label} = Worn}{N} \tag{3.2}$$

$$L_{Drag}(x, y) = \frac{\sum_{label=\text{Drag}} I(x, y) \in Superpixel_{label} = Drag}{N} \tag{3.3}$$

$$L_{Person}(x,y) = \frac{\sum_{label=\text{Person}} I(x,y) \in Superpixel_{label} = Person}{N} \tag{3.4}$$

Equations (3.3) to (3.3) calculates the likelihood of a particular pixel (x, y) being a part of a super pixel of that label divided by the total number of images.



(A) $\Pr(Location|Any\ Label)$ (B) $\Pr(Location|Person)$

(C) $\Pr(Location|Drag\ Carry)$ (D) $\Pr(Location|Worn\ Carry)$

FIGURE 3.5. Location Likelihood for different labels

Figure 3.5 shows different likelihood maps calculated for different labels. The white areas show higher likelihoods of a particular location in the image being a part of that label, while the darker regions show lower likelihood probabilities.

3.3.2. Detection using Learned Location Probabilities.

During the testing phase, a video is processed by the LLVS system to extract segmentation regions as explained in section 3.1. The algorithm for detecting if a given region is a carried object or not is described below.

```
Data: Likelihood maps for Worn Carry, Drag Carry and Person Regions; segmented
      regions
Result: Carry labels for every segmented region
foreach superpixel in the track chip do
    DragScore = Calculate Drag Score for the given super-pixel
    WornScore = Calculate Worn Score for the given super-pixel
    PersonScore = Calculate Person Score for the given super-pixel
    [Type, Score] = Maximum( DragScore, WornScore, PersonScore)
    SuperPixelLabel = Type
    SuperPixelScore = Score
end
```

Algorithm 1: Predicting labels of regions based on Location Information

Hence, according to this algorithm, we will be able to assign labels to the super pixels using the location likelihood.

3.4. Detection using Shape

In this section a method for classifying superpixels into three types which are: *a*) Person Region *b*) Drag Carry Region *c*) Worn Carry Region , based on its shape is discussed. The motivation behind classification using shape features derives from the fact that carried objects are shaped differently from person regions. Superpixel regions containing carried objects tend to have sharp corners and are broad and wide. On the other hand, regions not containing carried objects are long and rectangular.

To learn the shapes of different super pixel regions, we calculate the image moments across the centroid of a super pixel using a technique mentioned by Hu [11] and commonly known as Hu Moments. Hu moments are useful because they encode shapes and are invariant to scale, rotation, and translation changes. The Hu moments are calculate as follows.

For a image I(x,y), the raw moments are calculated according to the following equation.

$$M_{ij} = \sum_x \sum_y x^i y^j I(x, y) \tag{3.5}$$

$$Centroids(\bar{x}, \bar{y}) = \left(\frac{M_{10}}{M_{00}}, \frac{M_{01}}{M_{00}} \right) \tag{3.6}$$

Central moments can be calculated as

$$\mu_{pq} = \sum_x \sum_y (x - \bar{x})^p (y - \bar{y})^q I(x, y) \tag{3.7}$$

Since the moments are always taken relative to centroid, central moments are translation invariant. Scale and translation invariant moments can be found dividing the corresponding central moments by a scaled (00)th moment.

$$\eta_{ij} = \frac{\mu_{ij}}{\mu_{00}^{1+\frac{i+j}{2}}} \quad i + j \geq 2 \tag{3.8}$$

These scale and translation invariant moments are used to define Hu [11] moments as follows which are also rotational invariant.

$$
\begin{aligned}
&I_1 = \eta_{20} + \eta_{02} \\
&I_2 = (\eta_{20} - \eta_{02})^2 + \eta_{11}^2 \\
&I_3 = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2 \\
&I_4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2 \\
&I_5 = (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] + \\
&(3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \\
&I_6 = (\eta_{20} - \eta_{02})[(\eta_{30} + \eta_{12})^2 - (\eta_{21} - \eta_{03})^2] + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03}) \\
&I_7 = (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] \\
&-(\eta_{30} - 3\eta_{12})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]
\end{aligned}
\tag{3.9}
$$

All these seven moments corresponding to every super pixel are calculated and are used for training an expectation maximization algorithm as given in the following section.

3.4.1. Training on Shape Features.

The process of training different features using expectation maximization is shown in the Figure 3.6. Hu features for super pixels in the training set are collected independently in three sets. For every set a Expectation Maximization (EM) algorithm is trained using OpenCV [15] to fit 3 Gaussians. Hence we have three EMs in this case, each EM models the data for Worn, Drag and Person regions. These 3 EMs will be used in the testing processes.

3.4.2. Testing on Shape Features.

During the testing phase we want to predict the label of the super pixel based on the Hu moments. This Hu moment is passed through each EM model, over here EM models belonging to worn carry regions, drag carry regions and person regions. Every EM gives a log-likelihood of the point in high-dimensional space defined by Hu moments being generated by the gaussians in the EM model, as shown in Figure 3.7. A point defined in this Hu space is given a label which has the maximum likelihood

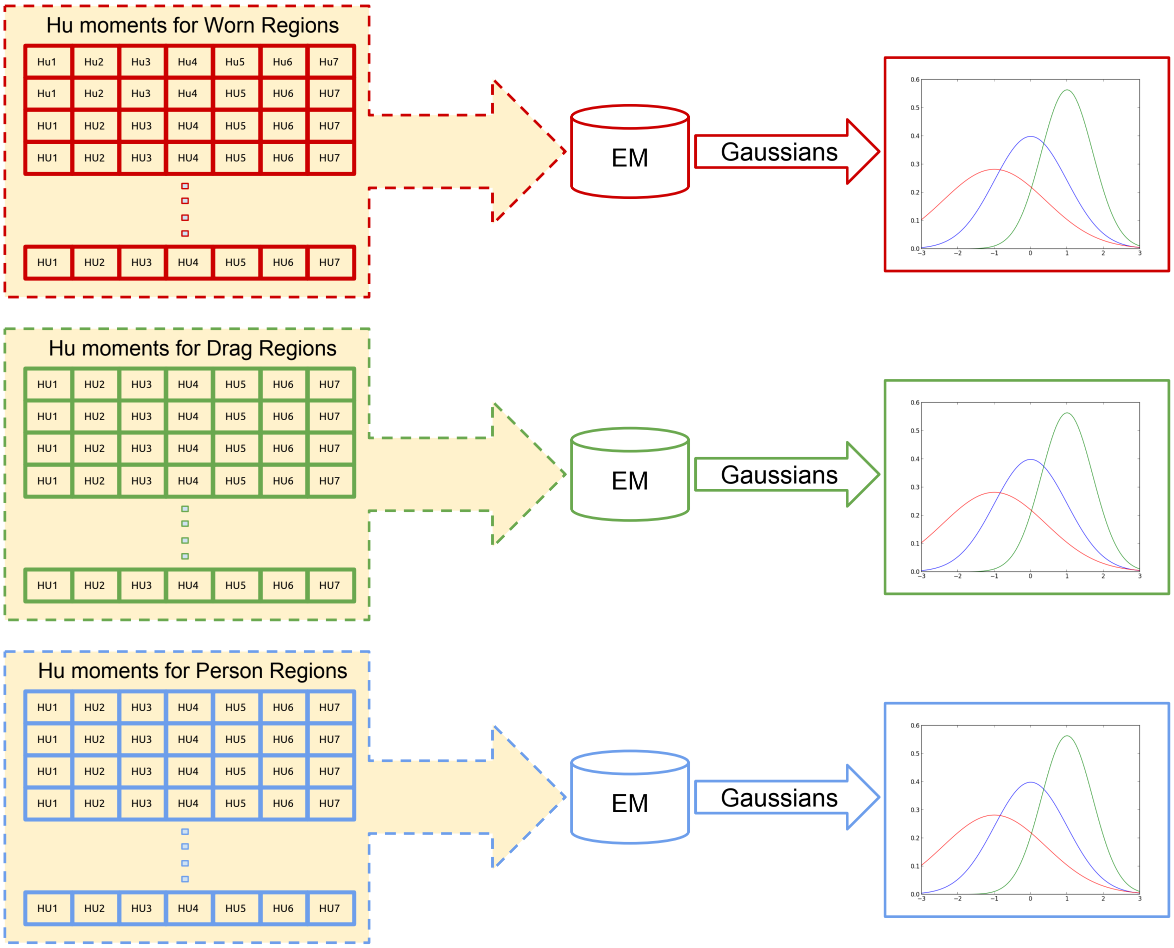

Hu moments for Worn Regions
Hu1 Hu2 Hu3 Hu4 Hu5 Hu6 Hu7
Hu1 Hu2 Hu3 Hu4 HU5 HU6 HU7
HU1 HU2 HU3 HU4 HU5 HU6 HU7
HU1 HU2 HU3 HU4 HU5 HU6 HU7
HU1 HU2 HU3 HU4 HU5 HU6 HU7
EM
Gaussians
Hu moments for Drag Regions
HU1 HU2 HU3 HU4 HU5 HU6 HU7
HU1 HU2 HU3 HU4 HU5 HU6 HU7
HU1 HU2 HU3 HU4 HU5 HU6 HU7
HU1 HU2 HU3 HU4 HU5 HU6 HU7
HU1 HU2 HU3 HU4 HU5 HU6 HU7
EM
Gaussians
Hu moments for Person Regions
HU1 HU2 HU3 HU4 HU5 HU6 HU7
HU1 HU2 HU3 HU4 HU5 HU6 HU7
HU1 HU2 HU3 HU4 HU5 HU6 HU7
HU1 HU2 HU3 HU4 HU5 HU6 HU7
HU1 HU2 HU3 HU4 HU5 HU6 HU7
EM
Gaussians
0.0 0.1 0.2 0.3 0.4 0.5 0.6
−3 −2 −1 0 1 2 3

FIGURE 3.6. Training process for shape features

between the 3 EMs. For example, if the EM for Worn Objects has the maximum likelihood, then the point is selected as a Worn Object.

3.5. DETECTION USING POSE INFORMATION

In this section the process of learning a pose of a person is related to the action of carrying objects. Two pose features, which are, Cosine Carry Angle and Carry Distance are used. Cosine Carry Angle is the cosine of the angle made by the arm with the line joining the arm

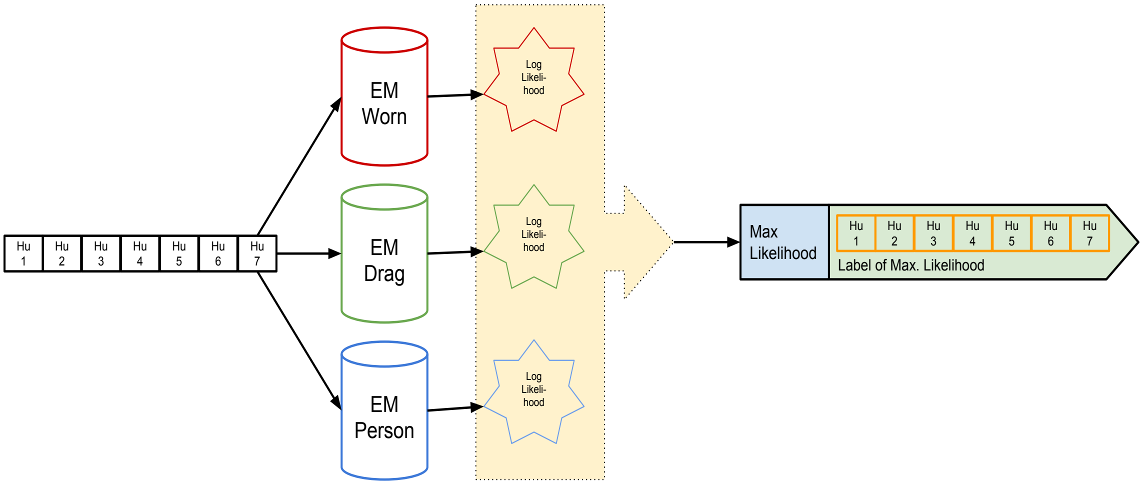
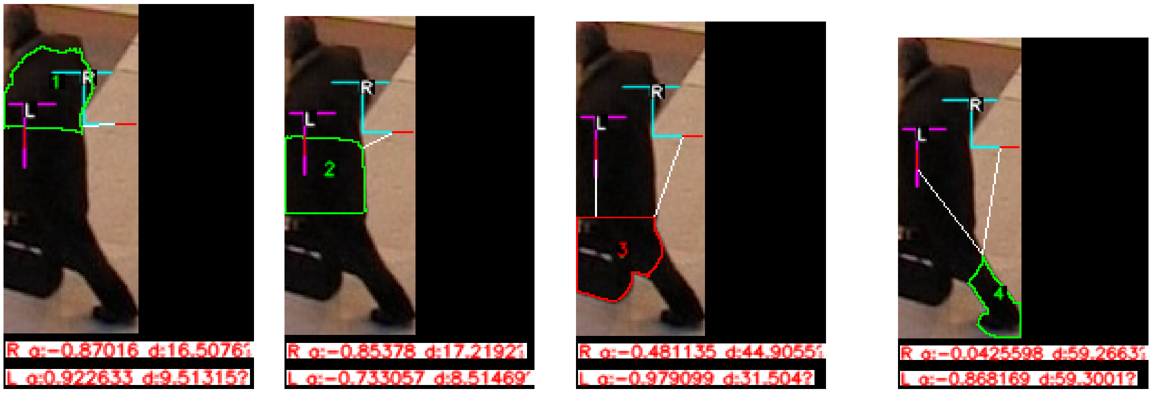


FIGURE 3.7. Predicting label of a super pixel using Hu moments

and the centroid of the superpixel region. Carry Distance is the distance from the end of the arm to the superpixel region.



(A) Pose feature when corresponding to contour 1

(B) Pose feature when corresponding to contour 2

(C) Pose feature when corresponding to contour 3

(D) Pose feature when corresponding to contour 4

FIGURE 3.8. Sample run showing the Cosine angle representation of data

The pose algorithm as described by Yang and Ramanan [21] gives the location of different arm joints. Using these joint locations different pose properties are calculated. Figure 3.8 shows a sample relation of different pose features for different region segmentation. Region

#3 is a super pixel over a suitcase carried by a person. We can see the Left Arm is pointing straight at the carried object, and since the Right Arm is occluded its predicted pose is incorrect.

3.5.1. Learning Carry Pose. Figure 3.9 charts the process of training an EM using the pose properties. Three different EMs are calculated for left arm and right arm. As discussed in the previous sections, each EM model a Gaussian over a set of points in a high dimensional space. These EM models are used for detection as described in the following section.

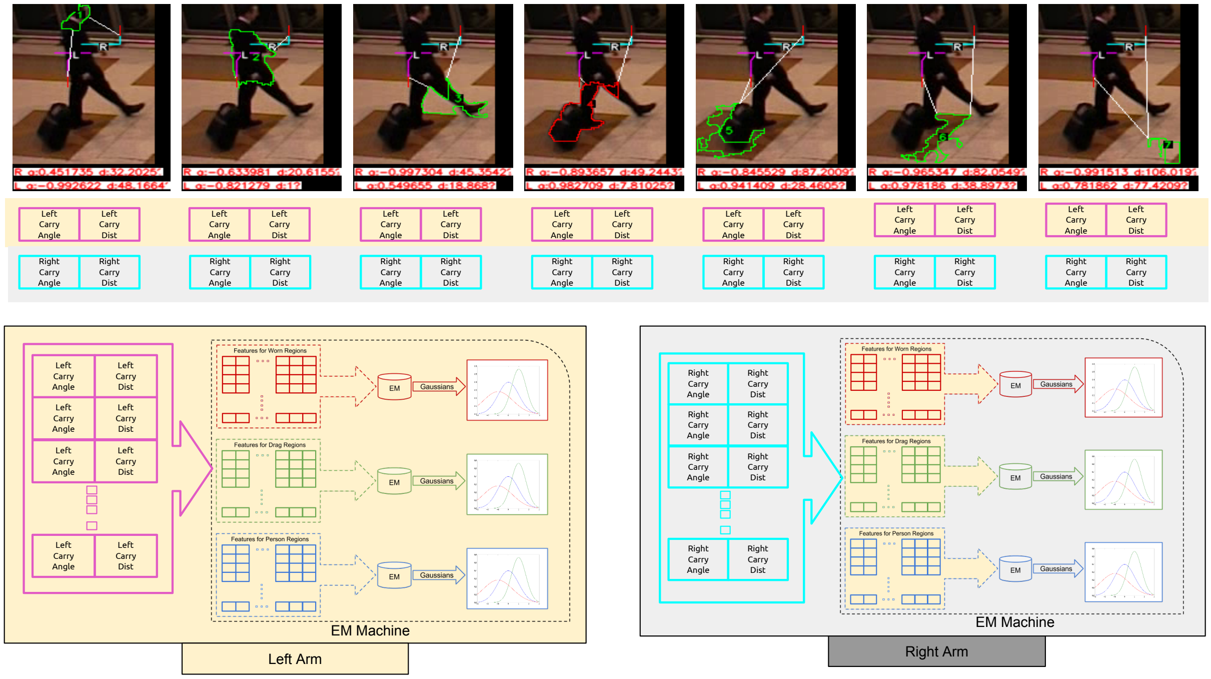


Figure 3.9. Pose Training

3.5.2. Detecting if the given pose is a carry pose. As mentioned in the previous section, previously trained EM models are used for detecting if the given pose in relation to the specific segmented region is a carry pose. Figure 3.10 describes the process of predicting if a given region is a carry region based on the pose of a person. Now we have two final

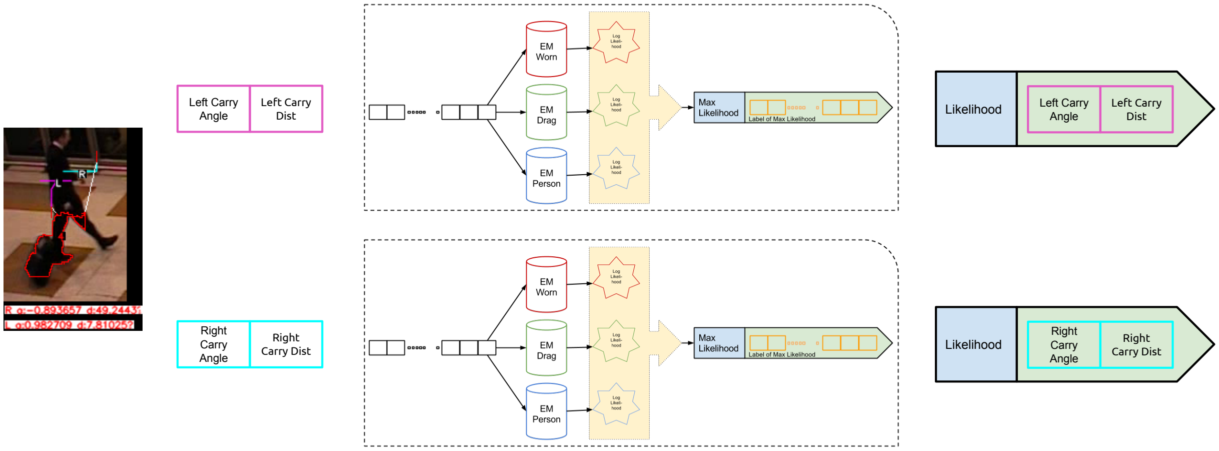


FIGURE 3.10. Predicting label of a region based on the pose of the person

scores, one for the left hand and the other for the right hand. The label with the maximum likelihood between both the arms is assigned to the segmented region.

3.6. Detection using Appearance

Carried objects look different from clothes, jackets, trousers etc. In this section we will look at ways to take advantage of this distinction. SIFT descriptors are calculated around SIFT key points [12] and a Bag of Features type of detection algorithm is trained. Following sections elaborate this process in more detail.

3.6.1. Learning the Appearance of carry objects.

For the purpose of learning the appearance of carried objects, we use the Bag of Features [4] approach. Figure 3.11 describes the process of generating a vocabulary. All SIFT features are accumulated together in a bin for further processing. These feature points are then clustered using K-Means. These K clusters now form K words in the vocabulary, which is then used as a representation for the accumulated features.

To model the distribution of feature points across the generated vocabulary, a histogram is calculated. Every bin in this histogram represents a K-means cluster. Hence there are K

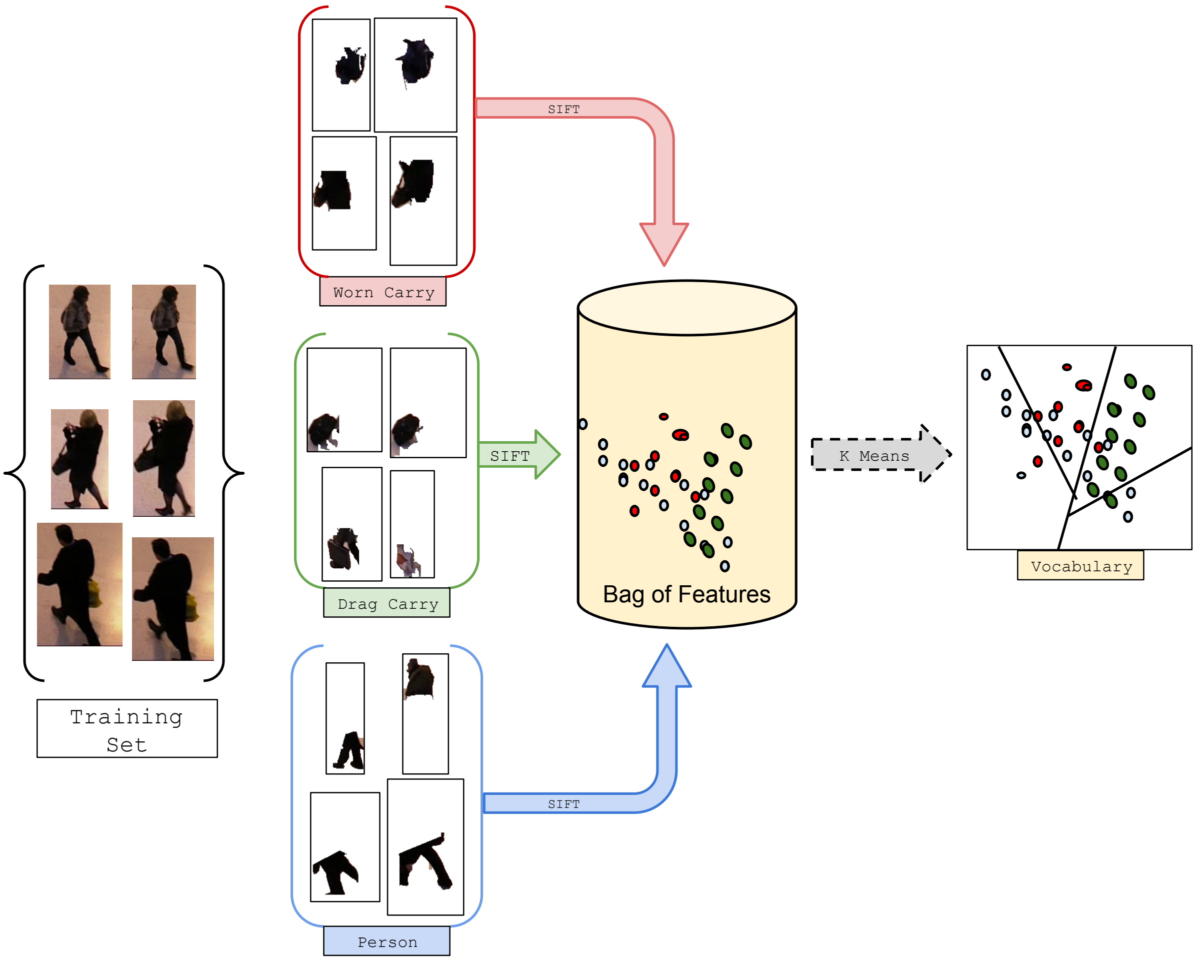


FIGURE 3.11. Generating the Bag of Features Vocabulary

bins in the histogram. Histograms belonging to every group label is calculated and used for training an EM as shown in the Figure 3.12.

3.6.2. TESTING IF THE GIVEN APPEARANCE IS A CARRIED OBJECT. In this section we will discuss on how to test if the given feature belongs to a Carried Object. As shown in the Figure 3.13, a histogram of the feature distribution in the vocabulary is calculated. This histogram is then passed through the EM models to find the model with maximum likelihood. The label belonging to the maximum likelihood is given to the region.

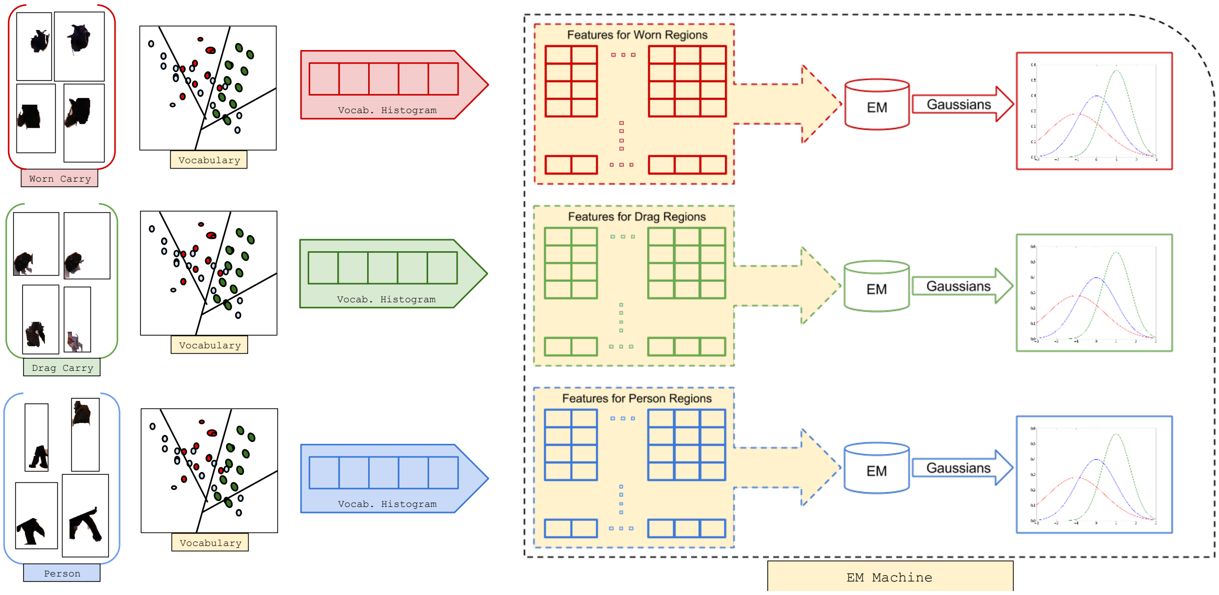


FIGURE 3.12. Training EMs for Bag of Features

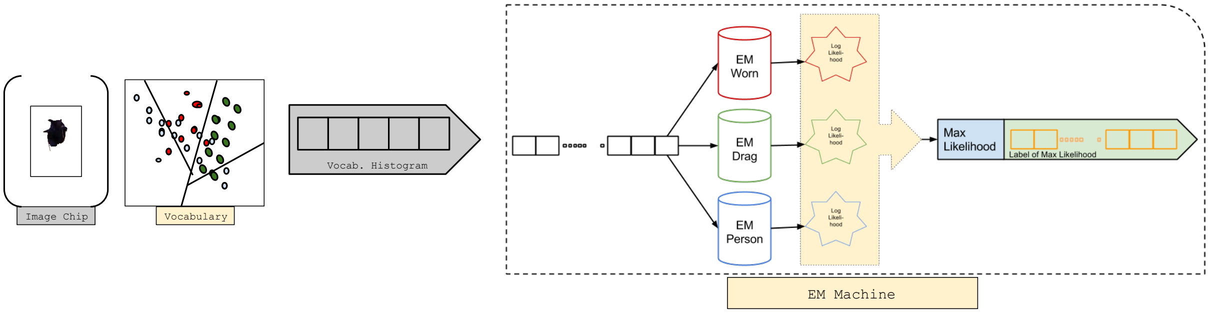


FIGURE 3.13. Testing using Bag of Words

CHAPTER 4

Results

This chapter evaluates different features for detecting a Carried Object by plotting the ROC curves using the PETS2006 dataset [19]. Section 4.1 discusses the testing protocol for the features. Section 4.2 evaluates the effectiveness of each feature for detecting Carried Objects, and comments on the results.

4.1. Testing Protocol

4.1.1. Work Dataset. PETS2006 dataset [19] is a video surveillance dataset captured at a railway station, is used in this thesis for training and testing purposes. It has people carrying common objects like briefcases, suitcases, and rucksacks. For an easier performance comparison with other published work on the same dataset, a testing protocol as described in Damen and Hogg [6] is used. The main keypoints for this testing protocol is as follows.

- Tracks of people walking together are manually removed.
- Tracks shorter than 10 frames are dropped.
- Out of the 7 sequences in the PETS2006 dataset, the sequence from camera number 3 is used, as there are many people seen from the side in this view.

4.2. Comparison of Performance

To evaluate the performance of detection by different features, an ROC plot is drawn. An ROC curve is a plot of false positive rate vs. true positive rate for different threshold values. ROC curves give an indication on how much information is available in the feature, and area under the curve (AUC) more than 0.5 is desirable. The diagonal line in the ROC plot indicates random classification with $AUC = 0.5$.

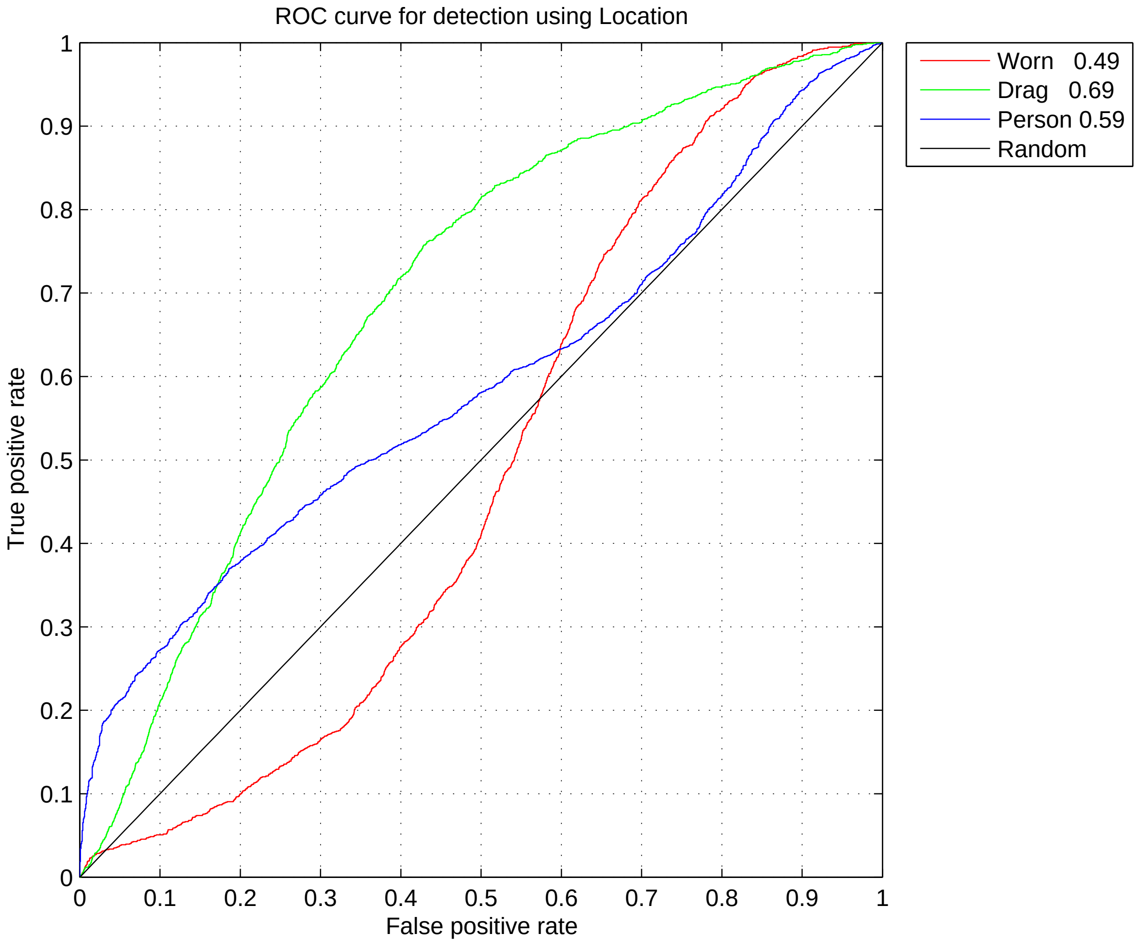


FIGURE 4.1. ROC plot using Location Data

Figure 4.1 charts the ROC plot for detection using location of a superpixel in the track chip. We can see that the Drag Carry detection beats the Worn Carry detection. It can be also seen that for the range where Person detection is above random, the Worn Carry detection is below random, and vice versa. This is because the region in the image chip which is most likely to be a person region, is equally likely to be a worn carry object. The shape of the curve for detecting Dragged Carried Objects, indicate that the Dragged Carry location likelihood map can be used as a filter, example, if operating at the false positive rate of 0.7 we can correctly identify 90% of the Dragged Carry Objects.

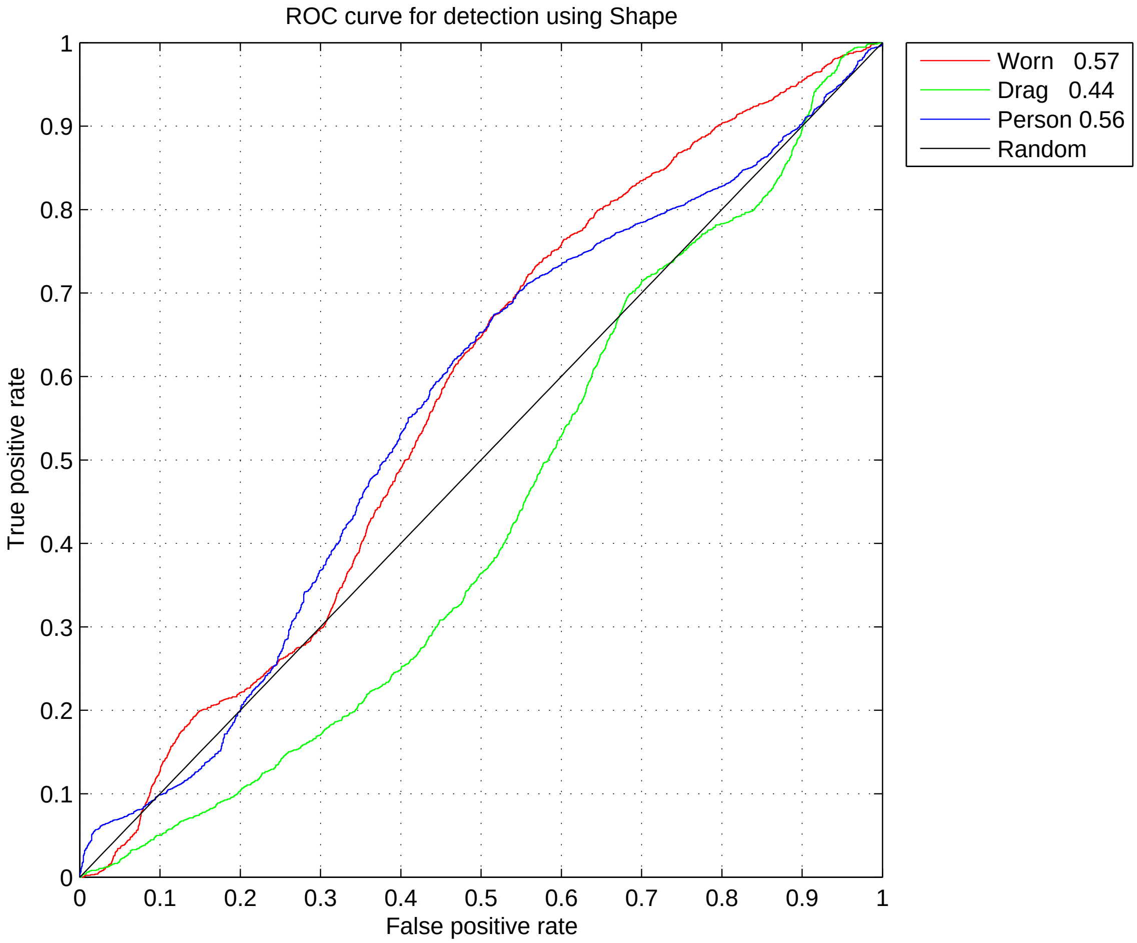


Figure 4.2. ROC plot using Shape Data

Figure 4.2 shows a plot of false positive rate vs true positive rate using Shape Features. We can see that, the curves for detecting Worn Carry object and for detecting Drag carry regions take opposite shape. For the region where Worn Carry detection is above random, the Drag Carry detection is close to random and vice versa. This is because, shape features are calculated using SLIC segmentation which generates uniform superpixel segments. Such uniformity between the superpixels does not help distinguish Carry Regions from non Carry Regions. The shape of the curve indicates that there is some information available in the signal, but not particularly useful.

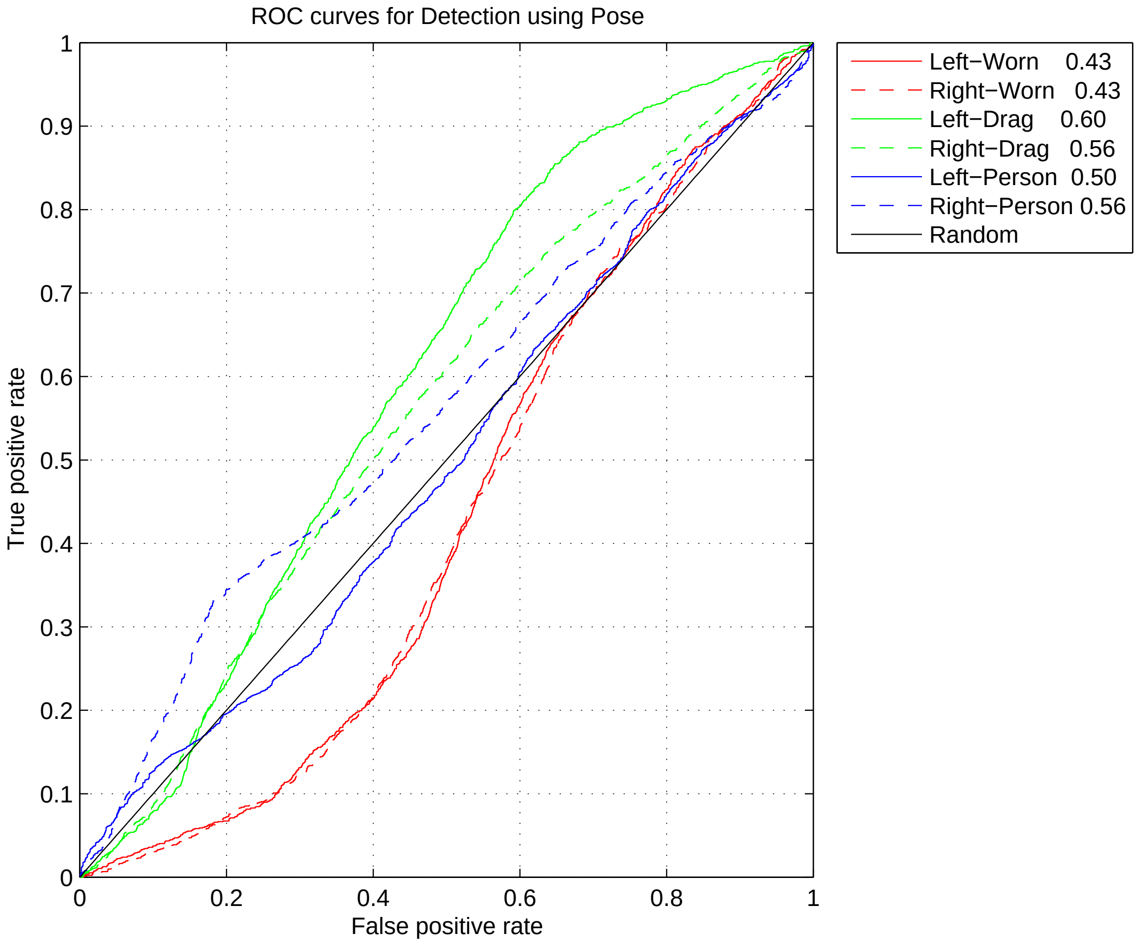


FIGURE 4.3. ROC plot for Pose Data

Figure 4.3 shows ROC plots for detecting carried objects using pose information of the left and right arm. It can be seen that since the act of dragging an object is correlated to the pose of the person, there is useful information available for detecting Dragged Carried Objects. The AUC is above random for detecting Dragged Carried Objects because the pose detection is incorrect when the person is occluded. The act of carrying objects like backpacks, have no correlation with the pose of a person. Hence the ROC curve for detecting Worn Carry Objects is below random.

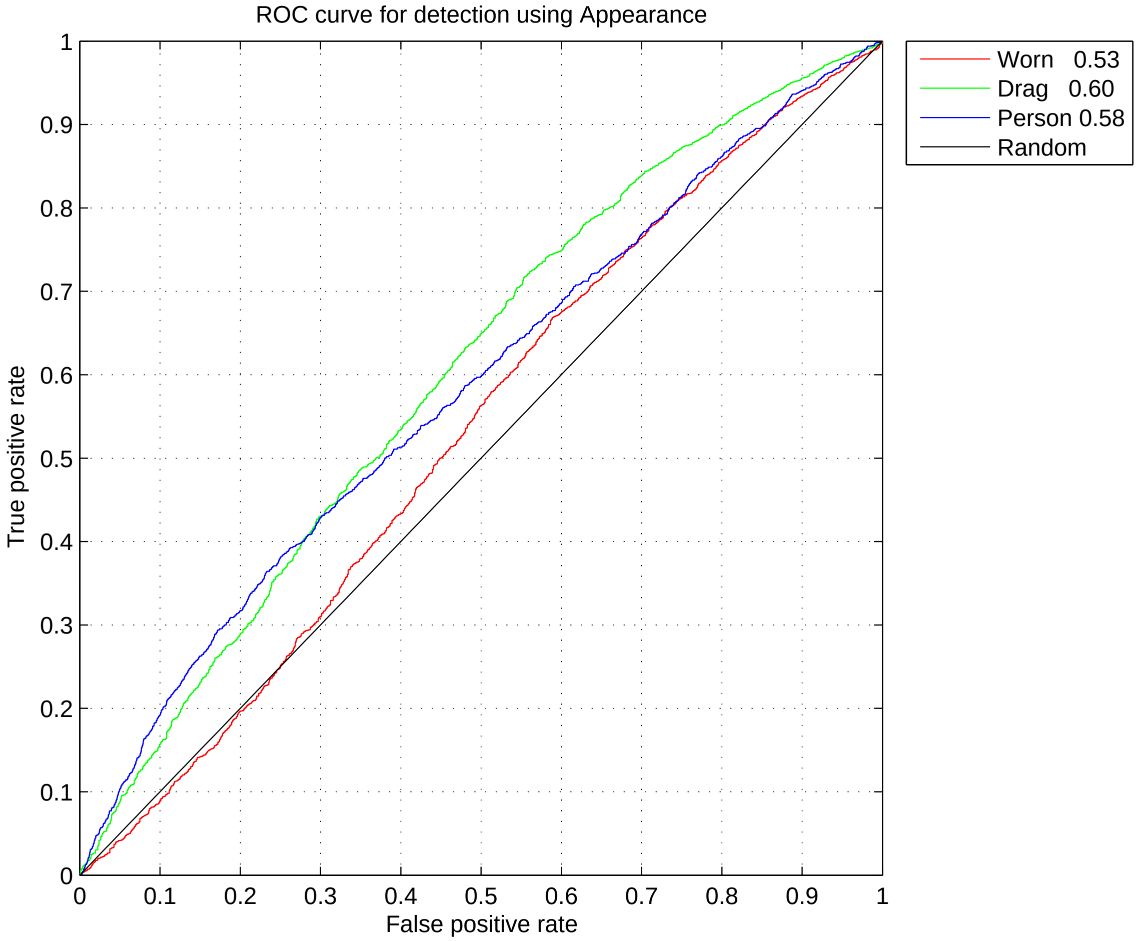


FIGURE 4.4. ROC plot for using Appearance Data

Figure 4.4 shows ROC plots for detection using appearance information. We can see that it has a marginal improvement over random detection. The improvement is not much since the resolution of the PETS2006 dataset is poor causing very few SIFT points (calculated using OpenCV) to be detected.

4.2.1. USING COMBINED DETECTION. Location, Appearance, Pose, and shape features are independent of each other, hence we can combine these features in different ways. The simplest way is to multiply the detection probability for all features. Figure 4.5 shows the ROC plots for detection using a combination of all the features. It can be seen that detection of Worn Carried Objects is random for a significant portion in the range, while there is a useful signal present for detecting Dragged Carried Objects.

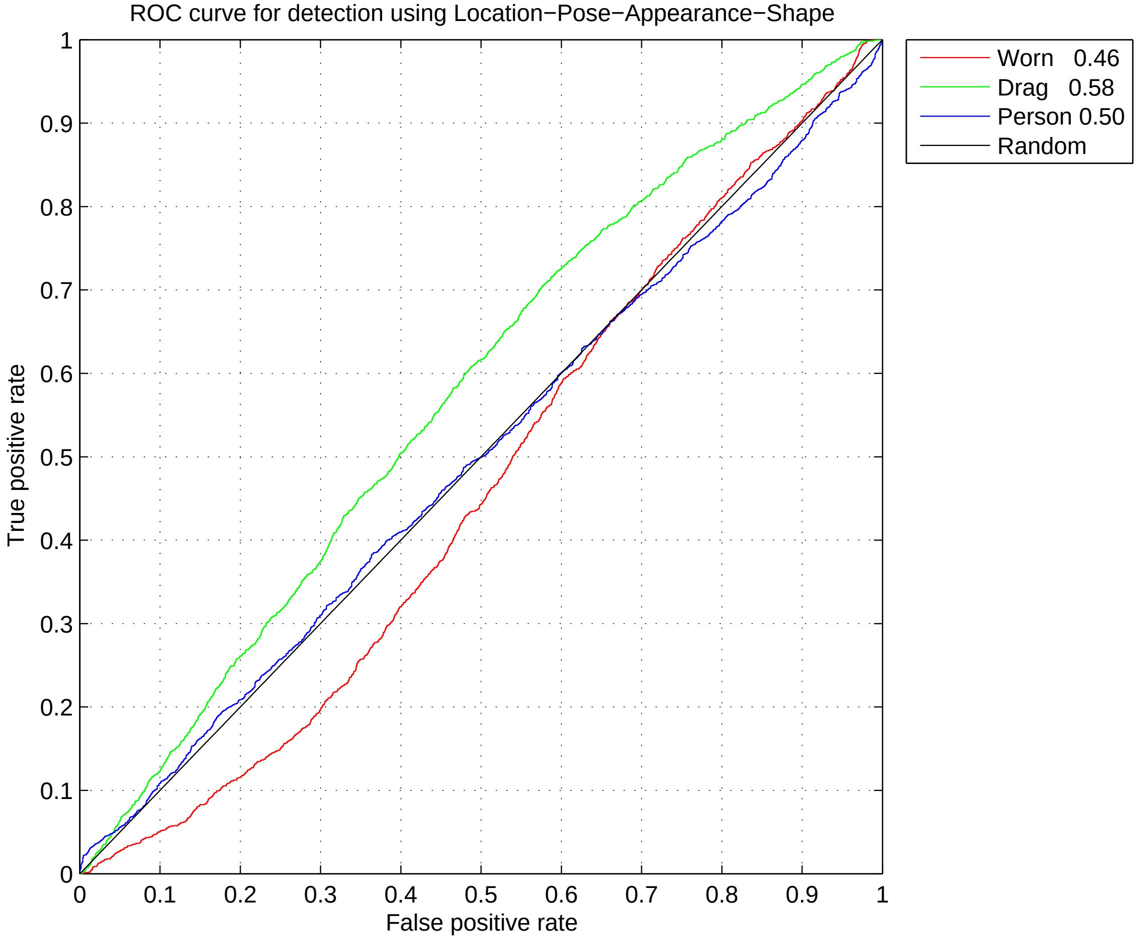


FIGURE 4.5. ROC plot for using Location, Shape, Pose , and Appearance Data

Figure 4.6 shows an ROC plot using Location, Pose, and Appearance features. We can see that the detection rates are very similar to those shown in Figure 4.5. This indicates that even after removing the shape features from the detection combination, there is no significant loss in the detection rate. This is another indication that shape features do not have much valuable information.

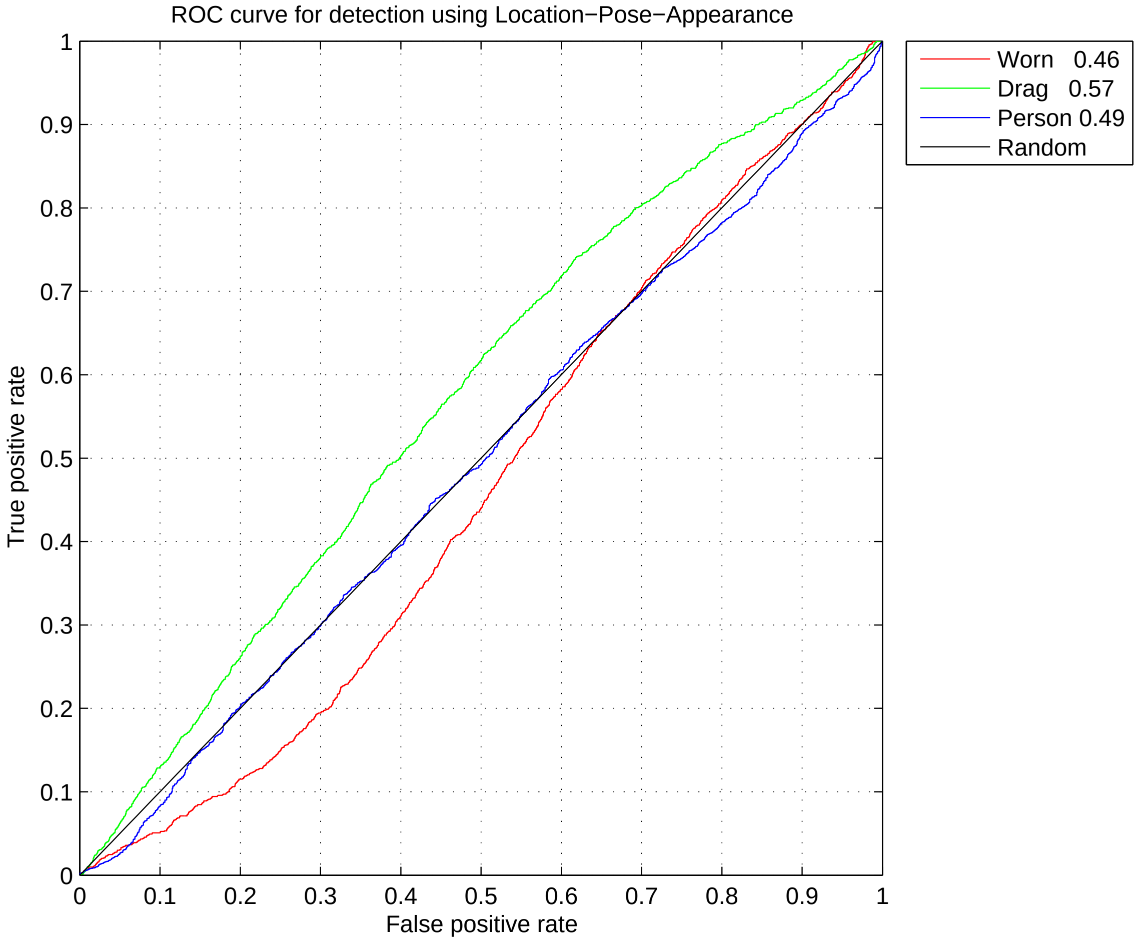


Figure 4.6. ROC plot for using Location, Pose , and Appearance

Figure 4.7 shows an ROC plot using only Location and Pose information. It can be seen that since location and pose features are strong indicators of Dragged Carried Objects, the ROC curve for Dragged Objects is significantly above random. Similarly Location and Pose information being weak indicators of Worn Carried Objects, the detection of Worn Carried Objects is less than random.

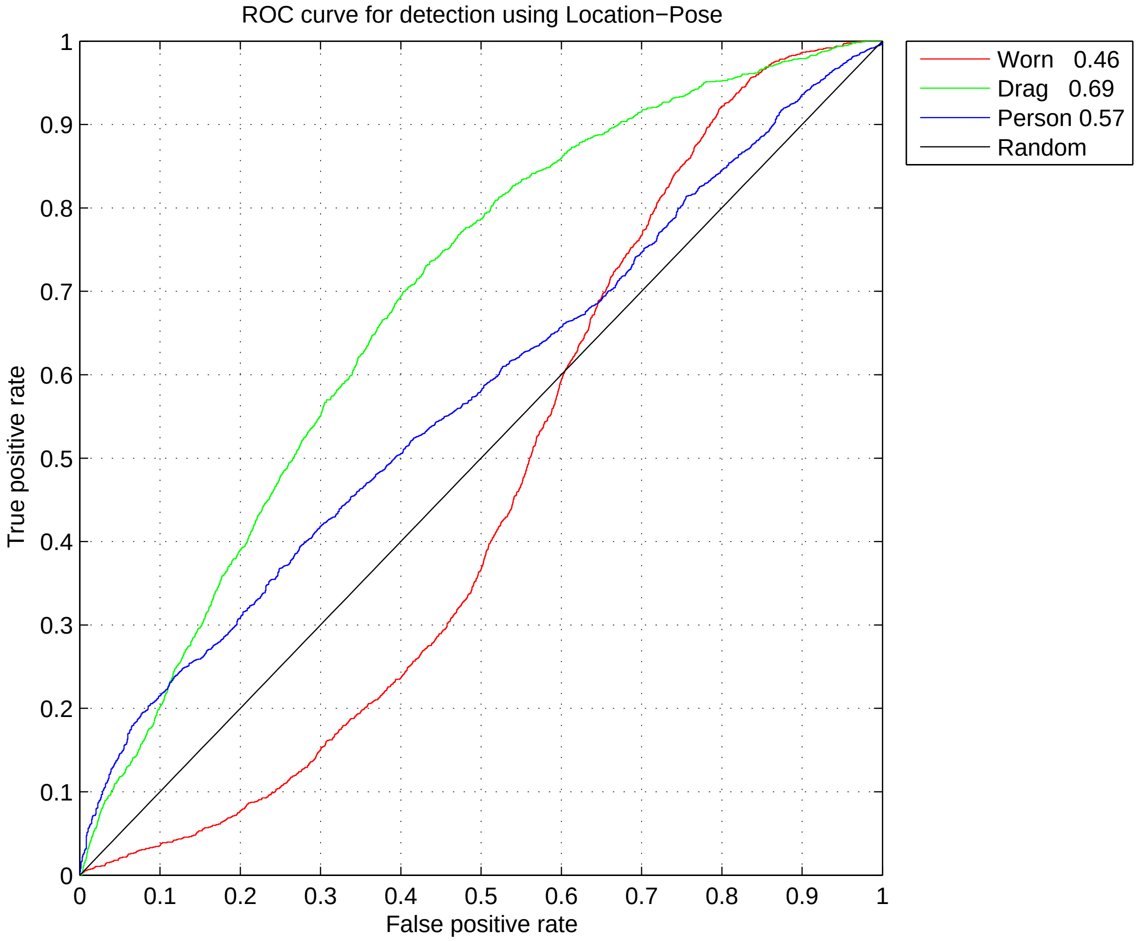


FIGURE 4.7. ROC plot for using Location, Pose

CHAPTER 5

Conclusion

This thesis studied different features, encoding information regarding Location, Shape, Pose and Appearance, and evaluated their performance for detecting carried objects in still images. Location features are calculated by building a location likelihood map of superpixels labeled as Dragged, Worn and Person. Shape features are calculated using Hu Moments over the contour of every superpixel generated using the SLIC superpixel segmentation by Achanta et al. [1]. Pose features are calculated using the pose algorithm proposed by Yang and Ramanan [21], and carry distance and carry angle is calculated using the pose of the arm and the superpixel. Appearance features are calculated using the appearance of superpixels labeled as Dragged, Worn and Person. These four features are evaluated on the PETS2006 dataset using the ROC curves.

ROC curves discussed in Chapter 4, show that location features can be used as filters for detecting carried objects, and location features are better at detecting Dragged Carried objects than Worn Carried Objects. Since, SLIC superpixel segmentation generates uniformly shaped superpixels, the Shape features do not have enough information to encode the difference between shapes of superpixels contours over Carried Objects and Person regions. The pose algorithm used cannot predict accurate pose in the event of an occlusion. To encode the appearance of superpixels, higher resolution images are required. Hence, the detection accuracy using Pose features and the appearance of the segmented superpixel regions depends on the quality of images.

In conclusion, this thesis provides a baseline for detecting carried objects in still images using four different features. It also shows the performance that can be achieved using individual features using only still images from a video. This evaluation provides a firm performance metric for detecting carried objects, and any complex work should match or exceed this metric.

5.1. Future Work

This thorough evaluation of features lead us to an interesting alternative approach for detecting carried objects. The features encoding information about shape, appearance, pose and location can be used as priors for segmenting a carried object. Figure 5.1 shows different features used as priors for segmentation.

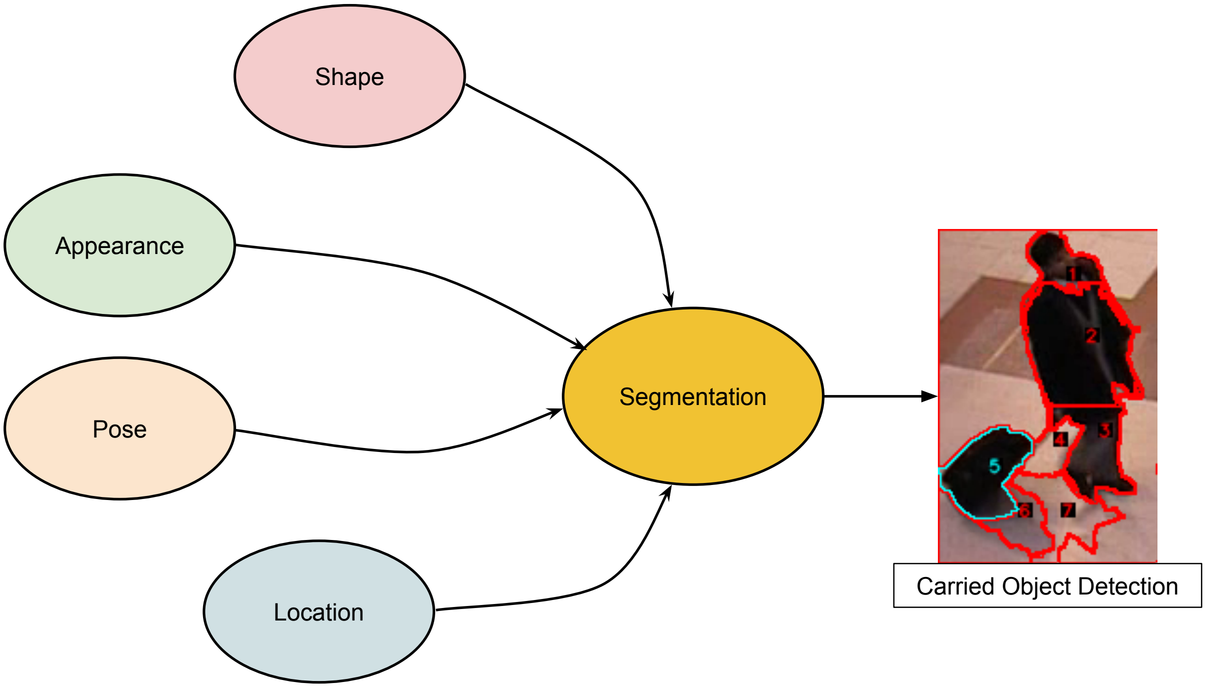


Figure 5.1. Features as priors for segmentation

In order to detect carried objects in a video, a Markov model based smoothing detection over consecutive frames can be used. This would allow us to directly compare the effectiveness of features to other published approaches which use video as an input.

Bibliography

[1] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Susstrunk. Slic superpixels compared to state-of-the-art superpixel methods. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 34(11):2274–2282, 2012. ISSN 0162-8828. doi: 10.1109/TPAMI.2012.120.

[2] David Barrett. One surveillance camera for every 11 people in britain, says cctv survey. 2013. URL `http://www.telegraph.co.uk/technology/10172298/One-surveillance-camera-for-every-11-people-in-Britain-says-CCTV-survey.html`.

[3] C. BenAbdelkader and L. Davis. Detection of people carrying objects : a motion-based recognition approach. In *Automatic Face and Gesture Recognition, 2002. Proceedings. Fifth IEEE International Conference on*, pages 378–383, May 2002. doi: 10.1109/AFGR.2002.1004183.

[4] Gabriella Csurka, Christopher R. Dance, Lixin Fan, Jutta Willamowski, and Cdric Bray. Visual categorization with bags of keypoints. In *In Workshop on Statistical Learning in Computer Vision, ECCV*, pages 1–22, 2004.

[5] Dima Damen. *Activity Analysis: Finding Explanations for Sets of Events.* PhD thesis, University of Leeds, 2009.

[6] Dima Damen and David Hogg. Detecting carried objects in short video sequences. In *ECCV (3)*, pages 154–167, 2008.

[7] Dima Damen and David Hogg. Detecting carried objects from sequences of walking pedestrians. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 99 (PrePrints), 2011. ISSN 0162-8828. doi: http://doi.ieeecomputersociety.org/10.1109/TPAMI.2011.205.

[8] M. Dimitrijevic, V. Lepetit, and P. Fua. Human body pose detection using bayesian spatio-temporal templates. *Computer Vision and Image Understanding*, pages 127–139, November 2006. URL http://www.sciencedirect.com/science/article/pii/S1077314206001135.

[9] I. Haritaoglu, R. Cutler, D. Harwood, and L.S. Davis. Backpack: detection of people carrying objects using silhouettes. In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, volume 1, pages 102 –107 vol.1, 1999. doi: 10.1109/ICCV.1999.791204.

[10] I. Haritaoglu, D. Harwood, and L.S. Davis. W4: real-time surveillance of people and their activities. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22 (8):809–830, Aug 2000. ISSN 0162-8828. doi: 10.1109/34.868683.

[11] Ming-Kuei Hu. Visual pattern recognition by moment invariants. *Information Theory, IRE Transactions on*, 8(2):179–187, 1962. ISSN 0096-1000. doi: 10.1109/TIT.1962.1057692.

[12] D.G. Lowe. Object recognition from local scale-invariant features. In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, volume 2, pages 1150–1157 vol.2, 1999. doi: 10.1109/ICCV.1999.790410.

[13] D.R. Magee. Tracking multiple vehicles using foreground, background and motion models. *Image and Vision Computing*, 22:143–155, 2001.

[14] NYPD. Domain awareness system. *NYPD*, 2012. URL http://www.nyc.gov/portal/site/nycgov/menuitem.c0935b9a57bb4ef3daf2f1c701c789a0/index.jsp?pageID=mayor_press_release&catID=1194&doc_name=http%3A%2F%2Fwww.nyc.gov%2Fhtml%2Fom%2Fhtml%2F2012b%2Fpr291-12.html&cc=unused1978&rc=1194&ndi=1.

[15] OpenCV. `http://docs.opencv.org/modules/ml/doc/expectation_maximization.html`, 2014. URL `http://docs.opencv.org/modules/ml/doc/expectation_maximization.html`.

[16] Transparency Market Research. Video surveillance and vsaas market - global industry analysis, size, share, growth, trends and forecast, 2013-2019. 2013. URL `http://www.transparencymarketresearch.com/video-surveillance-vsaas-market.html`.

[17] T. Senst, R.H. Evangelio, and T. Sikora. Detecting people carrying objects based on an optical flow motion model. In *Applications of Computer Vision (WACV), 2011 IEEE Workshop on*, pages 301–306, Jan 2011. doi: 10.1109/WACV.2011.5711518.

[18] Tobias Senst, Ruben Heras Evangelio, Volker Eiselein, Michael Ptzold, and Thomas Sikora. Towards detecting people carrying objects - a periodicity dependency pattern approach. In Paul Richard and Jos Braz, editors, *VISAPP (2)*, pages 524–529. INSTICC Press, 2010. ISBN 978-989-674-029-0.

[19] Kevin C. Smith, Pedro Quelhas, and Daniel Gatica-Perez. Detecting abandoned luggage items in a public space. In *IEEE Performance Evaluation of Tracking and Surveillance Workshop (PETS)*, 0 2006. IDIAP-RR 06-39.

[20] Aryana Tavanai, Muralikrishna Sridhar, Feng Gu, Anthony G Cohn, and David C Hogg. Carried object detection and tracking using geometric shape models and spatio-temporal consistency. In Mei Chen, Bastian Leibe, and Bernd Neumann, editors, *"Computer Vision Systems - 9th International Conference, ICVS 2013, St Petersburg, Russia, July 16-18, 2013, Proceedings"*, volume 7963 of *Lecture Notes in Computer Science*, pages 223–233. Springer, 2013.

[21] Yi Yang and Deva Ramanan. Articulated pose estimation with flexible mixtures-of-parts. In *CVPR*, pages 1385–1392, 2011.

[22] Zhengyou Zhang. Iterative point matching for registration of free-form curves and surfaces, 1994.

[23] Zhengyou Zhang. Iterative point matching for registration of free-form curves and surfaces. *Int. J. Comput. Vision*, 13(2):119–152, October 1994. ISSN 0920-5691. doi: 10.1007/BF01427149. URL `http://dx.doi.org/10.1007/BF01427149`.

[24] Jovia Zunic and Paul L. Rosin. A convexity measurement for polygons. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26:173–182, 2002.