**Candidate’s Declaration**

I hereby declare that the work entitled "**Helmet Recognition using Image Descriptors and Classifier**”, carried out in partial fulfillment of the requirement for the award of the degree of "**Integrated Master of Science in Applied Mathematics**", submitted in the Department of Mathematics, Indian Institute of Technology, Roorkee, is an authentic record of my own work carried out during a period of January 2016 to May 2016 under the supervision of Dr. N. Sukavanam, Department of Mathematics, Indian Institute of Technology, Roorkee.

The matter presented in this dissertation has not been submitted by me for the award of any other degree. I also confirm that:

Date: Swapnil Gupta

This is to certify that the above statement made by the candidate is correct to the best of my knowledge and belief.

Date:

Place: Roorkee

Dr. N. Sukavanam

Head of Department

Department of Mathematics

Indian Institute of Technology

Roorkee-247667, India

**Acknowledgment**

With great pleasure, I express my deepest gratitude to my supervisor, Dr. N. Sukavanam, Head of Department, Department of Mathematics, IIT Roorkee who gave endless support and encouragement right from the beginning till conclusion of this work and also for providing me infrastructural facilities to complete this dissertation successfully.

I owe my special thanks to my parents and friends for their unconditional love, encouragement and support.

Last but not the least, I am thankful to God for giving me the potential to complete this work.

Date: Swapnil Gupta

**Abstract**

Motorcycle accidents have been rapidly growing throughout the years in many countries.”So, to protect the head of bike rider during accidents, helmets are suggested. The main aim of this dissertation is to propose an approach for detection of bike-riders without helmet. For this, we first detect bike riders from surveillance video using background subtraction and object segmentation. Then we determine whether bike-rider is using a helmet or not using visual features and binary classifier. In order to evaluate the approach, a performance comparison of three widely used kernels in SVM classifier namely linear, sigmoid (MLP) and radial basis function (RBF) for classification is provided. Indeed, the algorithm step regarding the helmet detection accomplished an accuracy rate of 96.67%.”

**Contents**

**1 Introduction**  1

1.1 Objective  2

**2 Preliminaries** 3

2.1 Background Subtraction 3

2.1.1 Conventional Approaches 3

2.2 Gaussian Filter 8

2.2.1 Properties of Gaussian Filter 9

2.3 Thresholding 9

2.3.1 Otsu’s Binarization 10

2.3.2 Otsu’s Method (Mathematics) 10

2.4 Morphological Operations 11

2.4.1 Basic Concepts 12

2.4.2 Fundamental Operations 13

2.5 Feature Extraction 15

2.5.1 Histogram of Oriented Gradients (HOG) 15

2.5.2 Example of Hog 15

2.6 Support Vector Machine (SVM) 19

2.6.1 Hard Margin Classifier 19

2.6.2 Soft Margin Classifier 20

**3 Methodology and Experiments** 22

3.1 Improved Gaussian Mixture Model 22

3.2 Noise Removal – Gaussian Filter 24

3.3 Thresholding 26

3.4 Morphological Operations 26

3.5 Object Segmentation 27

3.6 Histogram of Oriented Gradients (HOG) 29

3.7 Classification 32

**4 Results and Conclusion**  33

4.1 Results 33

4.2 Conclusion and Future Work 34

**Bibliography**

**Appendices**

**CHAPTER 1**

**INTRODUCTION**

In almost every country, one of the most or popular mode of transportation are motorcycles because of their less prices and low operation and maintenance cost in comparison with another vehicles. However, there is aahigh risk involved because of aless protection and this increases the number of “motorcycle accidents in the last few decade.

“To reduce the involved risk, it is highly desirable for bike-riders to use helmet as it protects the motorcyclist head against accidents. Governments have made it mandatory by charging a punishable offense to ride a bike without helmet. However, the existing video surveillance based methods are passive and require signiﬁcant human assistance. In general, such systems are infeasible due to involvement of humans, whose efficiency decreases over long duration.”

Over the past decades, some artificial intelligent techniques like computer vision and machine learning with growing progress has been widely applied in this area to detect helmet. Here, we implemented computer vision algorithms, such as: background and foreground image detection to segment moving objects in scene and image descriptors to extract features for intelligent traffic system.

Computational intelligence algorithms like machine learning algorithms to classify the objects are also used. The algorithm used here detects any changes caused by moving objects. The problem is that movement in objects such as trees and bushes exist, which means the background isn't static all the time. So our system has to differentiate which movements are caused by actually moving objects we are interested in and which movements are caused by background objects. All the static areas and the areas of dynamic background calculated using probabilistic model, form the background mask. All the rest is defined as foreground. Performing this kind of parting makes it possible to have actual moving objects and the background separately in different layouts.

Our system is dealing with several important challenges such as dynamic background separation, shadow removal, denoising and moving object detection. The proposed system will work in real-time scenarios, hence there is a challenge of optimizing the algorithms so that they will be able to run in real-time.

Lighting changes can be solved by applying techniques such as contrast stretching and singular value equalization. Noise caused by weather changes (rain or wind) can be removed using blurring followed by morphological operations. Removal of shadow can be achieved after modelling the background mask followed by morphological operations using thresholding.

* 1. **Objective**

Our main aim is to come up with an”approach for detection of bike-riders without helmet using surveillance videos. For this approach, we created a strategy divided into two parts: motorcycle and helmet detection. So it first detects the bike riders from surveillance video using background subtraction and object segmentation techniques. Then it predicts whether bike-rider is wearing a helmet or not using visual features from feature extraction techniques and binary”classifier.

**CHAPTER 2**

**PRELIMINARIES**

* 1. **Background Subtraction**

Detecting the foreground objects in an image is an essential step in analyzing a real world scene. A statistical model can be used to describe images of the scene which doesn’t contain intruding things having some regular attributes. These statistical models can then be used to detect intruding things by recognizing the image parts which don’t satisfy the model. This whole method is known as “Background Subtraction” which then helps in detecting foreground objects which can be used in further image processing tasks such as image classification tasks etc.

* + 1. **Conventional Approaches**

1. **Frame differencing**

Segmenting moving objects in an image from the background is the initial step in all motion detection algorithms. There are two main steps in these algorithms, first being taking an image as a background , the second being spotting the frames obtained at time t, represented by to compare with the background image represented as . With the help of Image subtraction algorithms, we can deal with the problem of segmenting moving objects in an image.

The method is subtracting pixel value from the corresponding pixel value at the same position on the background image denoted as for each pixel value of .The mathematical expression of this method can be written as:



**Fig. 2.1** Background at time t:

The required assumption of this technique is to choose the background at time . Nevertheless, this method performs well in removing the background but fails when background pixels are moving. To improve the subtraction, we often put a threshold to the subtraction represented by .

Higher threshold is required for dealing with faster movements to improve the accuracy of the model.

**Fig.2.2** Video frames at consecutive times

** (a) (b)**

**(c) (d)**

**Fig. 2.3** (a) =25, (b) =50, (c) =100, (d) =200

**Mean Filter**

For”calculating the image containing only the background, a series of preceding images are averaged. For calculating the background image at the instant t,

where n is the number of preceding images taken for averaging and H(x,y,t) is a function used as a video sequence where t is the time dimension, x and y are the pixel location variables. This averaging refers to averaging corresponding pixels in the given images. ‘n’ would depend on the video speed (number of images per second in the video) and”the amount”of movement in the video. Threshold the difference between the image H(x,y,t) at time t = t and the background calculated. Thus the foreground can be given by is:

Where is threshold. Instead of mean, median can also be used when calculating

1. **Gaussian Mixture Model**

Background subtraction methods have been proven to perform better on separating the moving objects such as humans, vehicles etc but they fail to perform when they deal on images with a single fixed cameras. Change in environmental conditions such as variance in illumination , to and fro motion of tree branches or any other changes makes it harder to get the background from a continous set of frames. Hence, Gaussian Mixture Model technique was introduced which uses mixture of Gaussians to detect objects in motion from fixed cameras.

Due to the complexity and variability of the situation, only single Gaussian cannot model all the variations which imparts a need for a variable number of Gaussian models for each pixel. This number is represented by which is defined empirically and is generally in the range of 3 and 5. Variability in the number of components helps the model to adapt to the situation. Though it may seem quite good but it is not able to solve the proble of merged shadows and highly occluded objects. . Let I1, I2..... It denotes the intensity of a pixel for past t, consecutive”frames. Then, the probability of intensity value for a pixel at time t can be given by:

“

Where is weight and is jth Gaussian probability density function with mean and as variance at time t. For each pixel each pixel, the Gaussian components with low variance and high weight correspond to background class and others with high variance correspond to foreground class. At time t, the pixel intensity It is checked against”all Gaussian components.”

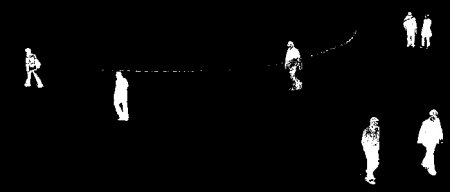
If jth component satisfies the condition:

“Then jth component is considered to be a match. Also, the current pixel is classified as background or foreground according to the class of jth Gaussian model. The weight update rule is given”by :

Where”β is learning rate which determines how frequently parameters are adjusted. Here, is a threshold which has significant impact when different regions have different lightning. Generally the value of is kept around 3, as µt-1 ± 3 accounts for approximately 99% of data. Also, other parameters of matched models are”updated as:

Here, “If no components are matched, we create a new Gaussian model having mean the current pixel value, high variance and low prior weight. This new model either takes the place of the component which is least probable when maximum number of components is reached or added as a extra component if the maximum is not reached.

**(a) (b)**



**Fig. 2.4** (a) Image, (b) Results of GMM

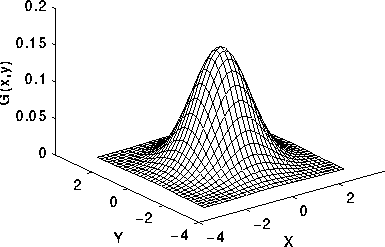
In real life scenarios, segmenting the foreground objects might be hard due to the presence of shadows casted by the moving objects. These moving objects creates a dynamic shadow effect which in turn results in significant illumination changes fooling the foreground segmentation algorithm to falsely classify them as foreground. These shadows often be regarded as noise. Imporved Adaptive Gaussian Mixture model is used to detect shadows in these type of situations which gives the mask consisting of three layers: foreground, shadows and background. Since , shadow is denoted by gray layer in the mask, it is very easy to discard the shadows by manipulating the GMM parameters. Thus, eventually it would give us a binary mask containing only foreground and background masks.

* 1. **Gaussian Filter**

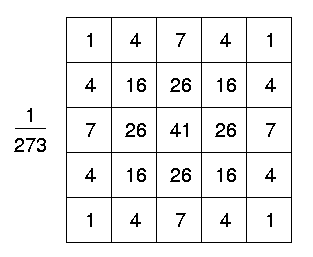
Noise signals need not be necessary of these shadows only but can also be changes happened due to weather conditions which can also be considered as background. The noise examples can be undesirable events such as snow, shaking trees due to wind, rain, illumination changes during the day. These changes must be removed for accurate recognition or segmentation.

Gaussian Filter is a type of filter which is used to remove noise and blur images. It does weighted averaging and its weight are assigned according to the Bell curve. Central pixel will get higher weight and if you go away from the center, you will get lower and lower weights. The Gaussian function in one dimension is represented as follows:

When working with images we need to use the two dimensional Gaussian function which is basically the product of two 1D functions which are Gaussian (one for each direction) and is given by:



**Fig.2.5** 2D Standard Gaussian distribution ( mean = (0,0) and σ = 1 )

The 2D Gaussian function given by the above equation is used for sampling Gaussian kernel coeffecients. The Gaussian filter uses this 2D distribution as initial point spread function which it does so by convolving the 2D Gaussian distribution function with the image.

**Fig.2.6** Approximation of a Gaussian with a σ of 1 by a integer valued 5 by 5 convolution kernel

* + 1. **Properties of Gaussian Filter**
* Low pass filter which is non-uniform
* Diminishing of kernel coefficients as it gets far from the centre.
* Higher weighting of the central pixels than those on the boundary.
* Wider peak caused by Larger values of σ (greater blurring).
* To maintain the Gaussian nature of the filter, Increase of Kernel Size with increasing σ.
* Gaussian kernel coefficients depend on the value of σ. At the edge of the mask, coefficients must be close to 0.
* No directional bias and rotational symmetric property of the kernel.
* Seperability of Gaussian Kernel
* Not preserve brightness of Image.
  1. **Thresholding**

One of the simplest methods of image segmentation is Threshoding. Binary Images can be created from a grayscale image with the help of thresholding. In the simplest thresholding methods each pixel in an image is replaced by a white pixel if the image intensity Si,j is more than some fixed constant T (i.e. Si,j >T), or a black pixel if the image intensity is lesser than that constant.

* + 1. **Otsu’s Binarization**

**Otsu's method**, is basically performing clustering based thresholding automatically. This method is named after Nobuyuki Otsu. An arbitrary value for threshold value is used in global thresholding. But a image which is bimodal containing two classes of pixels (foreground pixels and background pixels) or whose histogram is having two peaks, threshold value can be chosen as a value in the middle of those peaks. Otsu binarization calculates the efficient threshold which separates the two classes such that their inter-class variance is maxima and their combined spread (intra-class variance) is minimal.

* + 1. **Otsu’s Method (Mathematics)**

In Otsu's method Exhaustive searching of the threshold is done which will minimize the intra-class variance (the variance within the class), represented as a weighted sum of variances of the two classes:

Where weights and denotes probabilities of the two classes separated by a threshold h, and and are variances of these two classes.

The class probability is computed from the M bins of the histogram:

Otsu showed that maximizing inter-class variance is the same as minimizing the intra-class variance:

which can be expressed in terms of class means µ and class probabilities w and.

While the class mean µ1,2, H (h) is:

The relations given below can be easily shown correct:

The class means and probabilities can be calculated iteratively. Hence, this idea yields a efficient and effective algorithm.



**(a) (b)**

**Fig. 2.7** (a) Original image, (b) Image having thresholding effect

* 1. **Morphological Operations**

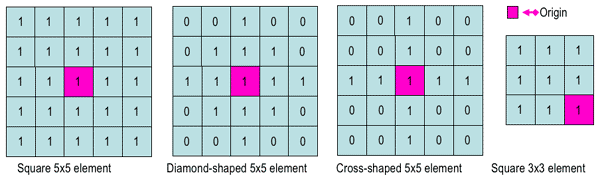
These binary images constitutes of various imperfections. Especially, the binary regions resulting from simple thresholding gets disturbed by texture and noise. Morphological image processing helps in deleting these imperfections along with maintain the structure of the image.

**2.4.1 Basic concepts**

Morphological image processing comprises of non-linear operations which relates to the morphology of features or shape of an image. The Morphological operators combine a binary image and a structuring element using a set operator (intersection, union, inclusion, complement). The objects in the input image is processed by them in accordance with characteristics$of$its$shape, $encoded$in$the$structuring$element.

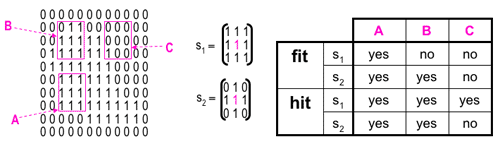
The structuring element is a small matrix of pixels, either with a value of zero or one:

1. The size of the structuring element is specified by matrix dimensions.
2. The shape of the structuring element is specified by pattern of ones and zeros.
3. Although generally the origin is outside the structuring element but it usually must be one of its pixels.



**Fig. 2.8** Examples of structuring elements which are Simple

Placing an structuring element in binary image results in association of each pixels with the corresponding pixels of the neighbourhood. The structuring element fits the image if both the pixels in the element and the corres. Pixel in the image is 1. Similarly, a structuring element hits the image if either one of the pixels in the element and the corres. Pixel in the image is set to 1.



**Fig. 2.9** Fitting and hitting with structuring elements s1 and s2 of binary element

We ignore the zero valued pixels of the structuring element i.e the irrelevant image parts.

**2.4.2 Fundamental operations**

****

**Fig. 2.10** Original Image

**1) Erosion**

The concept of erosion emanates from the idea of soil erosion. It $ erodes the $boundaries of foreground object (Always try to keep foreground in white) as the kernel or the structuring element slides through the image (as in 2D convolution). A pixel in the original image (either 1 or 0) will be considered 1 only if all the pixels under the kernel are 1, otherwise it is eroded (made to zero).

**Fig. 2.11** Eroded Image



All the pixels near boundary will be discarded depending upon the size of kernel. So the thickness or size of the foreground object decreases or simply white region decreases in the image. It is useful for removing small white noises, detach two connected objects etc.

**2) Dilation**

“It is just opposite of erosion. Here, a pixel element is ‘1’ if atleast one pixel under the kernel is ‘1’. So it increases the white region in the image or size of foreground object increases. Normally, in cases like noise removal, erosion is followed by dilation Erosion not only removes white noises, but also shrinks the object in an image. So after erosion, we generally apply dilation, since noise is gone and won’t come back. But finally the object area increases. It is also useful in joining broken parts of an object.”



**Fig. 2.12** Dilated Image

1. **Opening**

Opening is basically another name of erosion followed by dilation which also removes noise.



**Fig. 2.13** Opening of an original image

1. **Closing**

Closing is vice-versa of opening i.e. dilation followed by erosion. It is helpful in closing small holes in the foreground objects, or small black points on the object.



**Fig.2.14** Closing of an original image

**“Morphological filtering** of a binary image is conducted by considering compound operations like opening and closing as filters. They may act as filters of shape. For example, opening with a disc structuring element smooth corners from the inside, and closing with disc smooth corners from the outside. But also these operations can filter out from an image any details that are smaller in size than the structuring element, e.g. opening is filtering the binary image at a scale defined by the size of the structuring element. Only those portions of the image that fit the structuring element are passed by the filter; smaller structures are blocked and excluded from the output image. The size of the structuring element is most important to eliminate noisy details but not to damage objects of interest.”

* 1. **Feature Extraction**

“Feature extraction is a technique in which if there is a large set of data and that data is redundant, then it is transformed into a reduced set of features (named as feature vector). Now these selected features are expected to contain the relevant information from input data, so the desired task can be performed by using the reduced representation instead of the complete initial data. “

* + 1. **Histogram of Oriented Gradients (HOG)**

A feature descriptor is a representation of an image or an image patch that simplifies the image by extracting useful information and throwing away extraneous information.

The histogram of oriented gradients (HOG) is a feature descriptor which describes local object appearance and shape within an image by the distribution of intensity gradients or edge directions. The image is divided into small connected regions called cells, and for the pixels within each cell, a histogram of gradient directions is compiled. The descriptor is the concatenation of these histograms. For improved accuracy, the local histograms can be contrast-normalized by calculating a measure of the intensity across a larger region of the image, called a block, and then using this value to normalize all cells within the block. This normalization results in better invariance to changes in illumination and shadowing.

For extracting features, classification of features is important. So, in the HOG feature descriptor, the distribution (histogram) of directions of gradients ( oriented gradients ) are used as features. Gradients ( x and y derivatives ) of an image are useful because the magnitude of gradients is large around edges and corners ( regions of abrupt intensity changes ) and we know that edges and corners pack in a lot more information about object shape than flat regions.

**2.5.2 Example of HOG**

Let us take an example for better understanding of Hog feature descriptor. Consider an image patch of size 64 x 128 x 3 (channels) as shown in figure.

**Step 1:** Division of Image

Image is divided into 8 x 8 pixels per cell.



**Fig. 2.18** 64x128 pixel image divided into 8x8 pixel per cell

**Step 2:** Calculating gradient images

1.) Compute the gradient vector at each pixel which comprises of horizontal and vertical gradients.

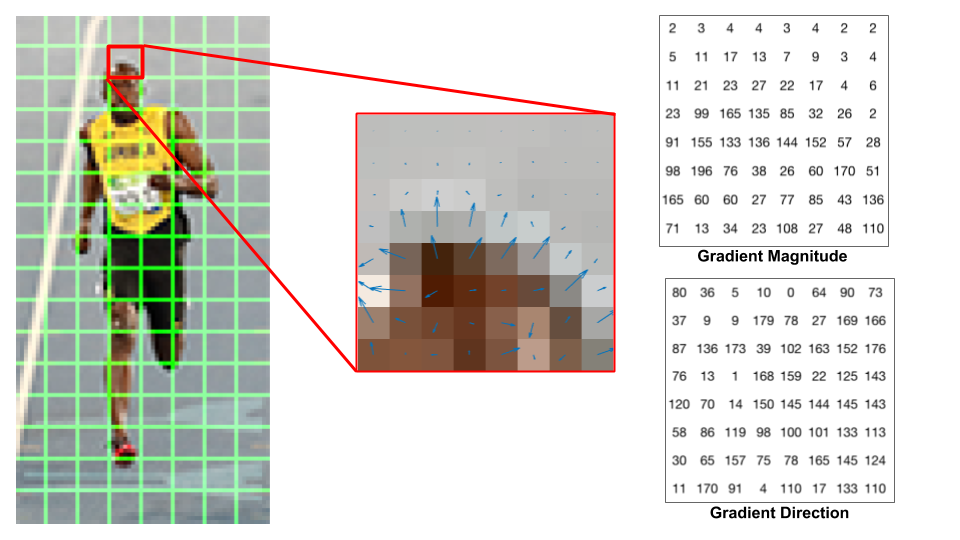
|  |
| --- |
| -1 |
| 0 |
| 1 |

2.) This can be achieved by filtering the image with the following kernels:

|  |  |  |
| --- | --- | --- |
| -1 | 0 | 1 |

1. Next, we can find the magnitude and direction of gradient using the following formula:





**Fig. 2.19** Center: The RGB patch and gradients represented using arrows.

Right: The gradients in the same patch represented as numbers

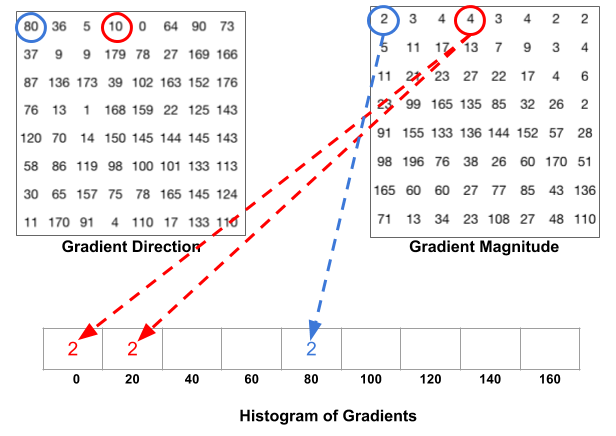
**Step 3:** Creating Histogram of Gradients

1.) Histogram is a vector (or an array) of 9 bins (numbers) corresponding to angles 0, 20, 40, 60 … 160.

2.) Our task is to put 64 gradient vectors (in 8x8 pixel cell) into a 9-bin histogram.

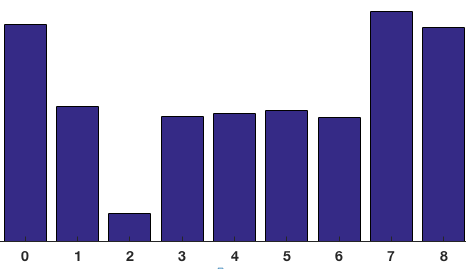
3.) Selection of bin is based on the direction and selection of vote (the value that goes into the bin) is based on the magnitude.

Let’s first focus on the pixel encircled in blue. It has an angle (direction) of 80 degrees and magnitude of 2. So it adds 2 to the 5th bin. The gradient at the pixel encircled using red has an angle of 10 degrees and magnitude of 4. Since 10 degrees is half way between 0 and 20, the vote by the pixel splits evenly into the two bins. If the angle is greater than 160 degrees, it is between 160 and 180, and we know the angle wraps around making 0 and 180 equivalent. So in the example below, the pixel with angle 165 degrees contributes proportionally to the 0 degree bin and the 160 degree bin.



**Fig. 2.20** Histogram of gradients

5.) The contributions of all the pixels in a cell are added up to create the 9-bin histogram. For the above patch it looks like this:



**Step 4:** Normalization

In the previous step, we created a histogram based on the gradient of the image. Gradients of an image are sensitive to overall lighting. If you make the image darker by dividing all pixel values by 2, the gradient magnitude will change by half, and therefore the histogram values will change by half. Ideally, we want our descriptor to be independent of lighting variations. So we “**normalize**” our histogram to make it unaffected by lighting variations. .)

Normal Normalization

Consider RGB color vector [128, 64, 32].



Magnitude of vector

Normalized vector [0.87, 0.43, 0.22]

1.) A bigger block of size 2x2 cells (16x16 pixels per block) is used instead of 1 cell with 8x8 pixel.

1 cell= 8x8 pixels gives 1 histogram

1 bigger block= 2x2 cells= 16x16 pixels gives 4 histograms

16 x 16 block Normalization

1.) A bigger block of size 2x2 cells (16x16 pixels per block) is used instead of 1 cell with 8x8 pixels.

1 cell= 8x8 pixels gives 1 histogram

1 bigger block= 2x2 cells= 16x16 pixels gives 4 histograms

2.) Concatenation of 4 histograms is done to form a 36 x 1 element vector.

3.) This vector is normalized in the same way as a 3×1 vector is normalized above.

4.) The window is then moved by 8 pixels and the process is repeated.



**Fig. 2.21** Block Movement

**Step 5:** Calculating HOG Feature vector

* Final feature vector is calculated by concatenating 36×1 vectors into one giant vector.
* There are 7 horizontal and 15 vertical positions of a block in an image patch.
* Therefore, total of 7 x 15 = 105 positions of a block.
* Each 16×16 pixel block is represented by a 36×1 vector.
* So concatenating them all into one giant vector, we obtain 36×105 = **3780** dimensional vector.

**2.6 Support Vector Machine (SVM)**

Support Vector Machine” (SVM) is a supervised machine learning algorithm which can be used for both classification and regression challenges. However, it is mostly used in classification problems. In this algorithm, we plot each data item as a point in n-dimensional space (where n is number of features) with the value of each feature being the value of a particular coordinate. Then, we perform classification by finding the hyper-plane that differentiates the two classes very well.



**2.6.1 Hard Margin Classifier**

If the training data is linearly separable, we can select two parallel hyperplanes that separate the two classes of data, so that the distance between them is as large as possible. The region bounded by these two hyperplanes is called the "margin", and the maximum-margin hyperplane is the hyperplane that lies halfway between them.



For the linearly separable case, consider a data set of (,), i=1, 2……., m where ε and the labels ε {-1, +1}. The boundaries separating the 2 classes would be:

‹w,› + b >=1 for =+1 (22)

‹w,› + b <=-1 for =-1 (23)

Combining both we get,

(‹w,› + b) >=1 (24)

Here, w is a vector normal to the hyperplane. Support vector Machines find an hyperplane which separates the two classes by maximizing the margin giving the following quadratic optimization problem:

min .w subject to (‹w,› + b) >=1 (25)

We find the Lagrange function:

= .w - -1) (26)

where >=0 and i=1 to m are the Lagrange multipliers. Transforming the above to its dual we get

max = - (27)

subject to >=0 i=1, 2,…. , m (28)

= 0 (29)

The Optimal Hyperplane can be obtained by maximizing this Dual problem. The solution determines the variables and . Hence, an Optimal separation Hyperplane is achieved as given by

f(x,) = + (30)

**2.6.2 Soft Margin Classifier**

Hard margin SVM can work only when data is completely linearly separable without any errors (noise or outliers). In case of errors either the margin is smaller or hard margin SVM fails. So, soft margin SVM was proposed by Vapnik to solve the problem of non-linear separable data by introducing slack variables. For non linearly separable data, we want to find a better choice of features than the high order polynomial in non-linear equations because using high order polynomial becomes computationally expensive. So we define **kernels** which define new features. These are functions which takes low dimensional input space and transform it to a higher dimensional space i.e. it converts non separable problem to a separable problem.

Following are different kind of kernels:

Polynomial Kernel:

K (, =

RBF Kernel:

K (,= exp (

, where above two kernels are the most used kernels.

**CHAPTER 3**

**METHODOLOGY AND**

**EXPERIMENTS**

In order to accomplish our final objective, mixtures of various techniques of image processing are used.

Flowchart in Fig. 3.1 well defined the implemented algorithm. Right off the bat it is required to get a continuous video grouping streams gained by the camera. Then each frame of the video sequence is converted into a gray scale image. The details of the proposed work are given underneath.

**3.1 Improved Gaussian Mixture Model**

One of the issues, while isolating the moving objects from background, is ascended to the way that objects in genuine have shadows. Due to movement of shadow, background in the scene may be wrongly considered as foreground as movement causes noteworthy changes of lightning. In this way, here "Improved Adaptive Mixture of Gaussian" proposed by Zivcovic is utilized which recognize shadow (shadow) to expel them in further preparing of an image.

GMM strategy is a pixel-wise operation, which implies the probabilistic estimation for deciding, whether the pixel is associated to the foreground mask with pixel value 1 or to the background mask with pixel value 0, is designed for each pixel. It enables the algorithm to identify the shadows and returns mask involving 3 layers: foreground, Identified shadows and background. Finally only binary mask containing foreground and background is acquired by changing GMM parameters so as to neglect all the identified shadows.

**Fig. 3.1** Flowchart for detection of bike-riders without helmet

(C) Background

Subtraction (GMM)

(D) Noise Removal

(B) Convert Image to Grayscale

(I) Object

Classification

(H) Feature Extraction

(G) Object Segmentation

(A) Video Frames

(F) Morphological

Operations

(E) Thresholding

Reject

Class:

Everything Else

Class:

Bike-Rider

(J) Feature Extraction for Upper 25% part (HOG)

(K) SVM Classifier

Class: With Helmet

Reject

Class: Without Helmet

Bike Riders Without

Helmet Detected

**Fig. 3.2** (A) A Sample RGB image, (B) Conversion to a gray scale image, (C) Binary mask is displayed with shadow after applying GMM

**(A)** **(B)**



**(C)**

**3.2 Noise Removal – Gaussian Filtering**

Not just shadows can cause unwanted changes in the scene yet climate conditions have critical effect which can add noise in the scene. So, they must be considered as a component of the background mask. Cases of noise can be undesired condition, for example, rain, snow, trees and shrubs movements caused by blowing wind and illumination changes throughout the day. All of these changes must be removed for more exact detection.

Gaussian filter is applied after applying GMM strategy to blur the images, with the goal that little changes will be eliminated and object borders end up smooth or even get lost, based on the window estimate applied for obscuring. For our case window size of 5x5 with standard deviation 0.8 is chosen by hit and trial from different pair of parameters. Subsequently, just objects with huge changes will at present be perceived as moving parts. Figure 3.3 demonstrates the effect of the blurring on the foreground mask after manipulating GMM parameters for shadow removal.

**(A)**



**(B)**

**Fig 3.3** (A) Removal of shadow from above foreground mask, (B) Output of Gaussian filter with kernel size 5x5 and sigma value 0.8

**3.3 Thresholding**

After applying Gaussian filter, foreground mask is transformed into binary image using clustering based thresholding where the grayscale images are bunched in two sections as foreground and background, or on the other hand are demonstrated as a combination of two Gaussians.

A discretionary incentive for threshold is used in global thresholding. Yet, for bimodal image whose histogram has two peaks, can around take a value in the middle of those peaks as threshold incentive. So for our case we use Otsu binarization which is a type of clustering based thresholding and consequently ascertain a threshold value from histogram of image. Figure 3.4 shows the result of thresholding on filtered image (from figure 3.3).

**Fig. 3.4** Output of Otsu binarization after applying Gaussian filter to foreground mask

**3.4 Morphological Operations**

Blurring just lessens the noise. For better detection some morphological tasks, for example, ‘closing’ should be implemented to our foreground mask. An operation of opening takes place when dilation is applied directly after erosion whereas closing is dilation followed by erosion.

To further process the foreground mask, morphological operations involving closing is applied because foreground objects contain little gaps inside the objects (mainly due to presence of helmet black screen). Figure 3.5 demonstrates the effect of closing operations on the foreground mask with structuring element (kernel) of square-shaped 2x2 in size.



**Fig. 3.5** Result of closing operation applied on thresholded image from above figure(3.4)

**3.5 Object Segmentation**

“To segment objects from processed frame into parts, we use contours which segment objects based on object boundaries. Contours are curves joining all the continuous points (along the boundary), having same color or intensity. The contours are a useful tool for shape analysis and object detection and recognition.”

Background subtraction strategy just takes moving objects and disregards non-valuable subtle elements, for example, static objects. Still there might be numerous moving objects which are not of our advantage, for example, people, four wheeler and so forth. Filtering of bigger objects than bike riders is done on the basis of their areas. Consider jth object be Oj and an area of Aj then criteria for selecting Oj will be T1 < Aj < T2, where T1 and T2 are threshold for least and most area, respectively. The technique expect that for a settled camera, boundary comprising of only bikes is all around separated from objects with extensive area, for example, truck or little territory of noise. The target behind this is to just consider objects which will probably fall in bike riders classification. It helps in decreasing theacomplexitybof furthergsteps. Results of object segmentation are shown in figure 3.6.

**(A)**



count_2.png

**(B)**

**Fig. 3.6 (A)** Bounding box around foreground objects**, (B)** Processed frame segmented into objects

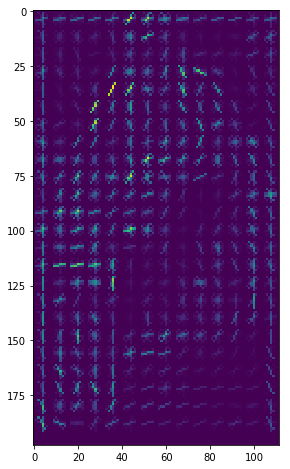
**3.6 Histogram of Gradients (HOG)**

“Object classification requires some suitable representation of visual features. Here, HOG descriptors are proven to be very efficient in object detection. These descriptors capture local shapes through gradients. We used 64 as a bounding box size, 9 bins, 8 x 8 pixels per cell and 2 x 2 cells per block. The resulting feature vector is h, where h Rn and n (dimensions) is 3780.”

To visualize the HOG feature of an object in an image, we plot the 9×1 normalized histograms in the 8×8 cells. We can observe from below image that shape of the person is given by dominantadirection ofatheahistogram.

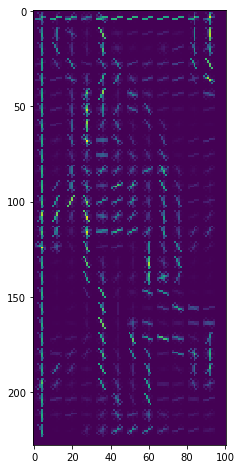
**Phase 1 – Detection of Bike riders vs. others (non bike riders)**

**(A)** **(B)**



**Fig. 3.7** (A) Segmented image of Bike Rider, (B) Gradient representation of bike-rider using arrows

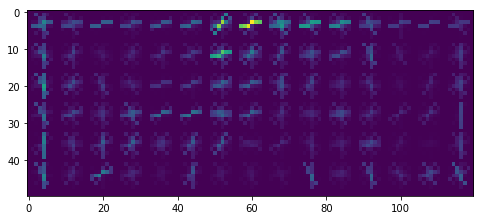
1. **(B)**



**Fig. 3.7** (A) Segmented image of a human, (B) Gradient representation of a human using arrows

**Phase – 2: Detection of bike rider wearing helmet vs. Bike rider not wearing helmet**

So for detecting bike riders without helmet, we find the head of bike rider and afterward utilize the way that fitting area of helmet will most likely be in upper territories of bike rider. Let's take U1/4 as upper one fourth piece of the objectatakenafromabackgroundamodelling step. For a moving bike, pixels in head region will have intensity of 1 i.e. white in U1/4. This step is very efficient which is reflected in our classification results for phase-2.

**6767_3.pngFig. 3.8** Left: Upper 25% part of bike-rider, Right: Gradient representation of upper 25% part

**3.7 Classification**

**Phase 1: Detection of Bike riders vs. others (non bike riders)**

Afterafeatureaextraction, subsequent stage is to classifyathem as ‘bike-riders’ vs ‘non bike riders', which is achieved with the help of binary classifier. We pick SVM because of its robustnessainaclassificationaperformanceaevenawhenatrainedafrom few number of featureavectors. Also, weause differentakernels such as linear,asigmoid (MLP),aradial basis functionk(RBF) to arrivebat bestahyper-plane.

**Phase 2: Detection of bike rider wearing helmet vs. Bike rider not wearing helmet**

“The method needs to determine if biker is violating the law i.e. not using helmet. For this purpose, we consider two classes: i) Bike-rider not using helmet (Positive Result), and ii) Biker using helmet (Negative Result). The support vector machine (SVM) is used to classify using extracted features from previous step (HOG). To analyze the classification results and identify the best solution, different combinations of kernels are used.”

**CHAPTER 4**

**RESULTS AND CONCLUSION**

“In this section, we present experimental results and discuss the suitability of the best performing representation and model over the others. For purpose of results and related experiments, standalone Linux machine with specifications Intel Xeon(R) CPU E5620@ 2.40GHz x 8 was used. In our experiments, we used OpenCV 2.7 and scikit-learn 0.16.”

**4.1 Results**

**Phase 1: Detectionaof Bikeariders vs. others (non bike riders)**

Table 1 presents results for bike-rider detection usingadifferentakernels in SVM such as linear,asigmoid (MLP),aradial basis functionk(RBF). The experimental results in Table I show that average performance of classification using HOG with linear kernel is better than combinations, because feature vector for this representation is sparse in nature which is suitable for linear kernel. Table I displays the accuracy of detecting a bike-rider in a frame.

**Phase 2: Detection of bike rider wearing helmet vs. Bike rider not wearing helmet**

The method needs to determine if biker is violating the law i.e. not using helmet. For this purpose, we consider two classes: i) Bike-rider not using helmet (Positive Result), and ii) Biker using helmet (Negative Result). The support vector machine (SVM) is used to classify using extracted features from previous step. To analyze the classification results and identify the best solution, different combination of features and kernels are used. However, HOG with linear kernel performs better than other combinations.

From the results presented in Table I & Table II, it can be observed that using HOG descriptors with linear kernel helps in achieving best performance. To test the performance, a surveillance video of around ten minutes at 30 fps i.e. 17,000 frames was used which give accuracy of 96.67%.

**Table 2.1** PERFORMANCEaOF PHASE-IaCLASSIFICATION (%)aOF DETECTIONaOFaBIKE-RIDER

|  |  |  |
| --- | --- | --- |
| **SVM (Kernel)** | **Accuracy** | **Time Taken (seconds) to train the model** |
| **Linear** | **96.83** | **136.69** |
| **MLP** | **95.45** | **303.16** |
| **RBF** | **90.33** | **280.45** |

**Table 2.2** PERFORMANCEaOF PHASE-||aCLASSIFICATION (%)aOF

‘BIKE-RIDER WITH HELMET’ VS ‘BIKE-RIDER WITHOUT HELMET’

|  |  |  |
| --- | --- | --- |
| **SVM (Kernel)** | **Accuracy** | **Time Taken (seconds) to train the model** |
| **Linear** | **96.67** | **138.98** |
| **MLP** | **94.45** | **299.16** |
| **RBF** | **93.37** | **250.56** |

**4.2 Conclusion and Future work**

In this thesis, framework for detection of traffic rule violators who ride bike without using helmet is proposed. This framework will also assist the traffic police for detecting such violators in odd environmental conditions such as hot sun, etc. A computer vision system is used that is divided as follow: moving objects segmentation, moving objects classification and helmet use detection. The results were satisfactory and SVM classifier with linear kernel using the HOG descriptor achieved the best result, with an accuracy rate of 0.9667.

The presented results are promising, but can be improved. An important step for this is the image capture, that needs a better quality. One of the future works is the licence plate recognize which need an image with better quality to recognize the characters.

The used descriptors return a lot of features and this can difficult the classification. So, the attribute selection is another future work. As, it is very important to increase the success rates, an analyst for this kind of algorithm is required.

Another future work is the detection of motorcycle passenger. A motorcycle can carry the motorcyclist and a passenger, but the proposed systems do not detect more than one helmet in an image. Since a passenger without helmet is a traffic violation.