HAND DETECTION IN STILL IMAGES

M.Tech. Thesis

By
BHUMI SHAH

DISCIPLINE OF ELECTRICAL ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY INDORE
JULY 2018

HAND DETECTION IN STILL IMAGES

A THESIS

Submitted in partial fulfillment of the
requirements for the award of the degree
of
Master of Technology

by
BHUMI SHAH

DISCIPLINE OF ELECTRICAL ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY INDORE
JULY 2018

INDIAN INSTITUTE OF TECHNOLOGY INDORE

CANDIDATE'S DECLARATION

I hereby certify that the work which is being presented in the thesis entitled **HAND DETECTION IN STILL IMAGES** in the partial fulfillment of the requirements for the award of the degree of **MASTER OF TECHNOLOGY** and submitted in the **DISCIPLINE OF ELECTRICAL ENGINEERING, Indian Institute of Technology Indore**, is an authentic record of my own work carried out during the time period from July 2016 to June 2018 under the supervision of Dr. Vivek Kanhangad, Associate Professor, IIT Indore.

The matter presented in this thesis has not been submitted by me for the award of any other degree of this or any other institute.

Signature of the student with date
(BHUMI SHAH)

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

Signature of the Supervisor of
M.Tech. thesis (with date)
(Dr. VIVEK KANHANGAD)

BHUMI SHAH has successfully given her M.Tech. Oral Examination held on 6th July 2018.

Signature of Supervisor of M.Tech. thesis
Date:

Convener, DPGC
Date:

Signature of PSPC Member #1
Date:

Signature of PSPC Member #2
Date:

Acknowledgements

I express my sincere gratitude to my guide Dr. Vivek Kanhangad for the continuous support, for his patience, motivation, and immense knowledge. His guidance helped me in conducting the research and writing of this thesis.

I would like to thank the members of my thesis committee: Dr. Trapti Jain and Dr. Aruna Tiwari for their insightful comments and encouragement. Their questions and suggestions helped me to widen my knowledge from various perspectives. Again, I am thankful to Dr. Trapti Jain, HOD, Dept. of Electrical Engineering for her support and cooperation. I am thankful to all the faculty members of the Department of Electrical Engineering for their guidance and support.

I must also thank my fellow batch mates Kalyani, Pratishtha, Garima and all other friends for making my stay at IIT Indore delightful.

Last but not the least, I also thank my family for their unceasing encouragement and support throughout this journey.

Bhumi Shah

Abstract

Hand detection has been an active area of research in the past few years due to its potential use in gesture recognition and human-computer interaction. In spite of the advancements made in this area, hand detection in static images still remains a challenge due to the high flexibility and shape variations of the articulated hand. In this work, we propose a method based on region proposals generated by skin segmentation and classification on the basis of extracted features. Specifically, the approach involves converting the RGB image into hybrid HCgCr colour space and then segmenting it into skin and non-skin regions using K-means clustering. The binary image obtained is smoothed by using morphological operations. After removing facial regions, we get region proposals for further processing. Various shape, colour and texture based features are extracted from these region proposals. These features include histogram of oriented gradients (HOG), dense colour histogram (DCH), gist and four-patch local binary pattern (FPLBP). Finally, features extracted from each region proposal are fed into a trained SVM classifier which gives a hand or a non-hand label. The dataset used is a combination of Oxford hand dataset, NUS hand posture I and NUS hand posture II datasets.

Contents

List of Figures

List of Tables

Abbreviations

Abbreviations	Description
CNN	Convolutional Neural Network
FRCNN	Faster Region-based Convolutional Neural Network
HOG	Histogram of Oriented Gradients
LPQ	Local Phase Quantisation
GMM	Gaussian Mixture Models
HSV	Hue-Saturation-Value colour space
RBF	Radial Basis Function
FPLBP	Four-Patch Local Binary Pattern
SVM	Support Vector Machine
LDA	Linear Discriminant Analysis
STFT	Short-Time Fourier Transform
DCH	Dense Colour Histogram
AP	Average Precision
PASCAL	Pattern Analysis, Statistical Modelling and Computational Learning
VOC	Visual Object Classes
PR	Precision Recall
AUC	Area Under Curve

Chapter 1

Introduction

1.1 Overview

Locating human hands is extremely useful in human-computer interaction (HCI) and robotics. Hands are present everywhere in our view and we use hands as our primary channel of interaction with the physical world, for manipulating objects, and expressing ourselves to other people. This is important in HCI as it provides a way for the computer to parse the raw visual inputs into a set of semantic information, making it possible for computers to react to human beings. In the HCI scenarios, the hand motion can serve for both communicative and manipulative purposes, e.g. sign language recognition and object manipulation. This not only provides natural and touch-less interaction experiences, but also makes it more convenient for the disabled to use and control the electronic devices. As virtual reality and augmented reality gears and applications are gaining momentum, researches which can improve hand gesture-based interaction have high importance, for which hand detection is a very important step. One of the first and crucial step in vision-based gesture recognition system is the identification of hands in images, called hand segmentation [5]. Image segmentation is the process of identifying clusters of pixels in images with common characteristics. The hand segmentation process categorises pixels into hand region

and background. Thus, focusing efforts on the analysis of hands in images and video data is a worthwhile endeavour as it could benefit many studies that aim to use hand recognition for gesture understanding, automation, machine-human interface, etc. Yet surprisingly, relatively little attention has been paid to developing methods that can robustly extract hands in images or videos. Even the small corpus of existing work that directly addresses hand detection does so only in relatively constrained environments. Despite the previous work in this field, this problem remains challenging due to the high flexibility and shape variations of the articulated hand. While generic object detection benchmarks have been very successful over the last decade, hand detection from a single image is still a challenging task due to the fact that hand shapes have great appearance variation under different wrist rotations and articulations of fingers [2, 6]. However, a more thorough look at hands and how they relate to human cognition reveals that there is even more virtue to analysing hands than one might assume.

Since the hand is highly flexible, it is quite a challenging task to recognise that in unconstrained environment. There have been many sensor-based methods to fulfil this task, in which specialised hardware is used to measure hand motion, e.g. data-gloves and optical sensors [7]. Although they provide quite accurate measurements, such systems are expensive. In contrast, vision-based approaches are much cheaper and can provide non-intrusive and natural interaction experiences. However, the vision-based approaches have their own challenges. First, the degree of freedom (DOF) of hand is high, which results in a large space of feasible hand postures. Therefore, recognising hand postures in such a large space is hard. Also, in contrast to other articulated objects such as the human body, the hand can rotate freely in 3D space, and thus suffers from severe self-occlusion in many viewpoints in monocular inputs, e.g. the fingers occlude each other or they are occluded by the palm. Moreover, the environment for the HCI applications is usually uncontrolled, e.g. illumination

variation, cluttered background, etc.

1.2 Related works

Hand detection has been studied as part of human layout parsing and gesture recognition for many years. Although several researchers have been trying to address the problem of detection of hands in an image, a robust algorithm is still elusive. This is primarily due to the fact that hands are highly deformable and articulated in nature. For several vision tasks such as parsing hand poses, gesture understanding for robotics, HCI, human layout detection [8], action recognition [9], sign language recognition [10] and human activity analysis [8], hand detection is inevitable.
Hand detection methods can be categorised into three main approaches, viz.

- colour-based methods,
- model-based methods,
- motion-based methods.

Methods based on skin colour build a skin model in a colour space for detecting hand regions. Mixture of Gaussian [11] is commonly used to model colours of skin and non-skin regions for hand localisation [12] and hand tracking [13, 14]. Colour-based methods often require prior knowledge of skin colour, extracted either from training data or from face detection, to build the skin model. Recently, ego-centric cameras have become popular. Images captured by such cameras often have a dynamic background, which makes hand detection even more difficult. A pixel labelling approach recently proposed by Li and Kitani [6] has shown to be quite successful in hand detection in ego-centric videos.
Model-based methods model the appearance of hands using a hand template. They can be implemented as a Viola-Jones like detector [15], or as a histogram of oriented

gradients detector [16] built from a large number of images, or learned as an ensemble of edges [17, 18] from a set of 2D projections of a 3D synthetic model. It is also possible to detect hands from human pictorial structure [19], which may bring more context information and allow inferring hand position. This is a common practice for static images, but usually it requires at least the upper body being visible for the inference of human structure.

Motion-based methods are used for ad-hoc applications, e.g. activity analysis and gesture recognition. They segment hands from background by motion and appearance cues [20, 21]. Hands can usually be tracked easily and it doesn't require a strong appearance model in most of the cases. However, motion-based methods are not suitable for moving cameras which produce images with lots of background motion.

Some of the works on hand detection used external hardware such as depth sensors [22]. But the use of depth sensors might not be feasible in all environments. Pisharady et al. [23] used a saliency map to detect hands in the image. Mittal et al. [2] proposed a state-of-the-art hand detection algorithm by fusing three techniques, namely, hand shape detector, context detector and skin-based detector. Their approach achieves 42.30% average precision on Oxford hand dataset [2]. Ong et al. [24] proposed a tree classifier to detect and recognise hand pose. Kolsch et al. [25] proposed cascaded hand detector using Haar features. Buehler et al. [10] proposed hand detection technique using multiple cues. Do et al. [26] proposed a hand detection and tracking method for fine grained action recognition in videos. The work done in [26] uses multiple cues for hand detection like the work of Mittal et al. [2] and includes upper body detection and flow information.

Other hand detection algorithms [27, 28] proposed in past used CNN based approach. Hoang Ngan Le T. et al. [29] proposed to use a multi-scale Faster Region-based Convolutional Neural Network (FRCNN) along with other cues such as face, steering wheel and cell-phone to detect hands inside a car for studying driver cell-phone usage.

Deng et al. [30], proposed a Convolutional Neural Network (CNN) based approach to detect hands and estimate rotation. This work proposes a context aware region proposal algorithm that uses a multi-component Support Vector Machine (SVM).
In this work, a hand detection method based on region proposal generation using skin segmentation is proposed. Hand detector model is trained using SVM classifier on shape, colour and texture based features with positive hand images and negative non-hands images which includes human faces, animals, flowers, etc. The dataset used is a combination of Oxford hand dataset [2] and NUS hand posture datasets [3, 4].

1.3 Organisation of the report

This chapter has introduced the background, motivation of the thesis and related work done by researchers in past. The remaining contents of this thesis are organised as follows:

- **Chapter 2**: This chapter provides details about proposed approach. Section 2.1 covers the steps involved in training phase. In Section 2.3, detailed description of testing phase is covered.

- **Chapter 3**: This chapter includes a detailed description of the databases used and experimental results. Section 3.2 covers experimental results for both validation and test experiments. This also includes a brief discussion of the performance results.

- **Chapter 4**: In this chapter, conclusions are made and a discussion on the possibility of future work is presented.

Chapter 2

Proposed Methodology For Hand Detection

In this chapter, a detailed description of proposed hand detection methodology has been discussed. Figure 2.1 shows the block diagram of the approach used for training phase, which consists of training SVM classifier on features extracted from hand and non-hand images of training set. Section 2.1 covers details of the same.

Figure 2.2 shows the block diagram of the approach used in testing phase, which mainly consists of four processing steps namely, skin segmentation, region proposal, feature extraction and classification. Section 2.3 elaborates each step involved in testing phase.

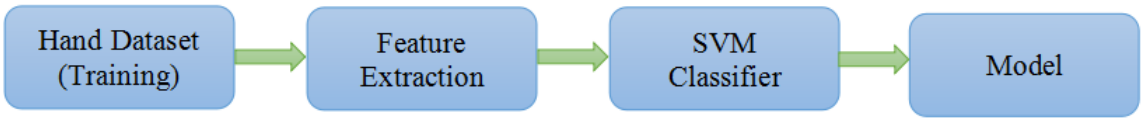

Figure 2.1: Block diagram of training phase

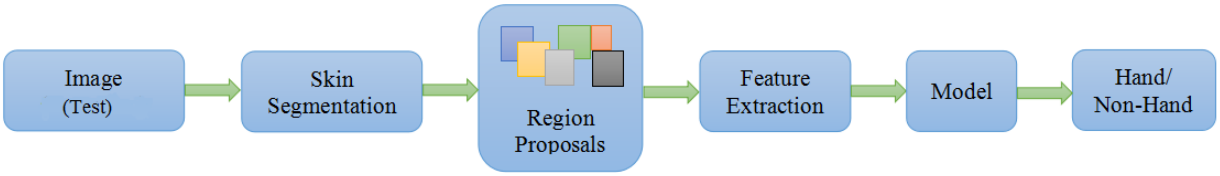

Figure 2.2: Block diagram of testing phase

In the following sections, detailed description of block diagrams is discussed. Section 2.1 details the training phase which includes feature extraction and training the classifier.

2.1 Training phase

For training, around 9000 hand samples, which are considered as positive training images, are cropped from the annotated training dataset [2] as shown in Figure 2.3. Around 20000 non-hand images, which constitute negative training images, are created by taking random patches from the images which does not include hand region. Also, some random images like human faces, flowers, animals, buildings, etc. are added to non-hand image dataset, samples of which are shown in Figure 2.4. So, in totality, training dataset comprises around 30000 images. To reduce computation time for feature extraction, these images are down-sampled to 50 x 50 pixels.

Figure 2.3: Samples of positive training images

Figure 2.4: Samples of negative training images

2.1.1 Feature extraction

After getting positive and negative training images, shape, texture and colour based features are extracted that carry discriminatory information useful for classification between hand and non-hand classes. For this purpose, following popular feature descriptors are explored namely, HOG and GIST (shape descriptors), FPLBP and LPQ (texture descriptors) and DCH (colour descriptor).

2.1.1.1 Histogram of Oriented Gradients

In the Histogram of Oriented Gradients (HOG) feature descriptor [16], the histograms 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, which pack in a lot more information about object shape than flat regions. Implementation of the HOG descriptor algorithm is as follows:

1. Divide the image into small connected regions called cells, and for each cell compute a histogram of gradient directions or edge orientations for the pixels within the cell.

2. Discretise each cell into angular bins according to the gradient orientation.

3. Each cell's pixel contributes weighted gradient to its corresponding angular bin.

4. Groups of adjacent cells are considered as spatial regions called blocks. The grouping of cells into a block is the basis for grouping and normalisation of

histograms.

5. Normalised group of histograms represents the block histogram. The set of these block histograms represents the descriptor.

Figure 2.5 demonstrates the algorithm implementation scheme:

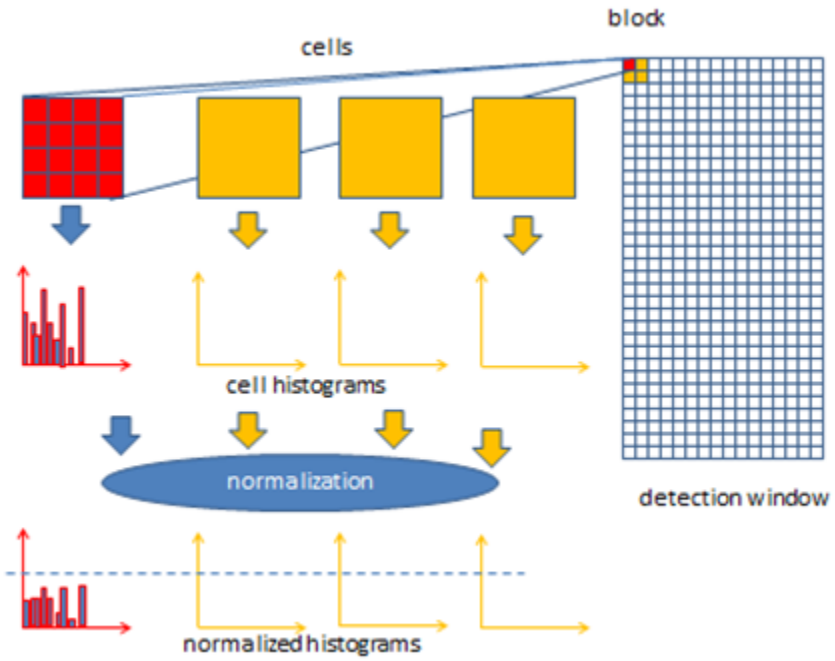

Figure 2.5: Implementation scheme of HOG descriptor
Image Source: [1]

Computation of the HOG descriptor requires the following basic configuration parameters:

- Masks to compute derivatives and gradients.
- Geometry of splitting an image into cells and grouping cells into a block.
- Block overlapping.
- Normalisation parameters.

In this work, cell size of 7 x 7 and block size of 2 x 2 cells is used while all other parameters are kept to default values. This gives us a feature vector of length 1296.

2.1.1.2 Local Phase Quantization

The Local Phase Quantization (LPQ) [31] is a blur insensitive texture descriptor, which uses local phase information extracted using short-time Fourier transform (STFT) computed in local neighbourhood at each pixel position of the image. The local frequency could be computed using a short-time Fourier transform on local M x M neighbourhoods N_x at each pixel position x of the image defined by

$$F(u,x) = \sum_{y \in N_x} f(x-y) e^{-j2\pi u^T y} \tag{2.1}$$

The transform is evaluated for all image positions using simply 1-D convolutions for rows and columns successively. In LPQ, only four complex coefficients are considered, corresponding to 2-D frequencies $u_1 = [a, 0]^T, u_2 = [0, a]^T, u_3 = [a, a]^T, u_4 = [a, -a]^T$. Let

$$F_x^c = [F(u_1, x), F(u_2, x), F(u_3, x), F(u_4, x)], \tag{2.2}$$

$$F_x = [Re\{F_x^c\}, Im\{F_x^c\}]^T \tag{2.3}$$

The corresponding 8-by-M^2 transformation matrix is

$$W = [Re\{w_{u1}, w_{u2}, w_{u3}, w_{u4}\}, Im\{w_{u1}, w_{u2}, w_{u3}, w_{u4}\}]^T \tag{2.4}$$

such that

$$F_x = W f_x \tag{2.5}$$

The phase information in the Fourier coefficients is recorded by observing the signs of the real and imaginary parts of each component in F_x. This is done by using a

simple scalar quantiser

$$q_j = \begin{cases} 1, & g_j \geq 0 \\ 0, & g_j < 0 \end{cases} \tag{2.6}$$

The resulting eight binary coefficients $q_j(x)$ are represented as integer values between 0-255 using binary coding

$$f_{LPQ}(x) = \sum_{j=1}^{8} q_j 2^{j-1} \tag{2.7}$$

Figure 2.6 illustrates implementation scheme of LPQ descriptor. In this work, window size (M) of 3 x 3 is chosen which gives feature vector of length 256.

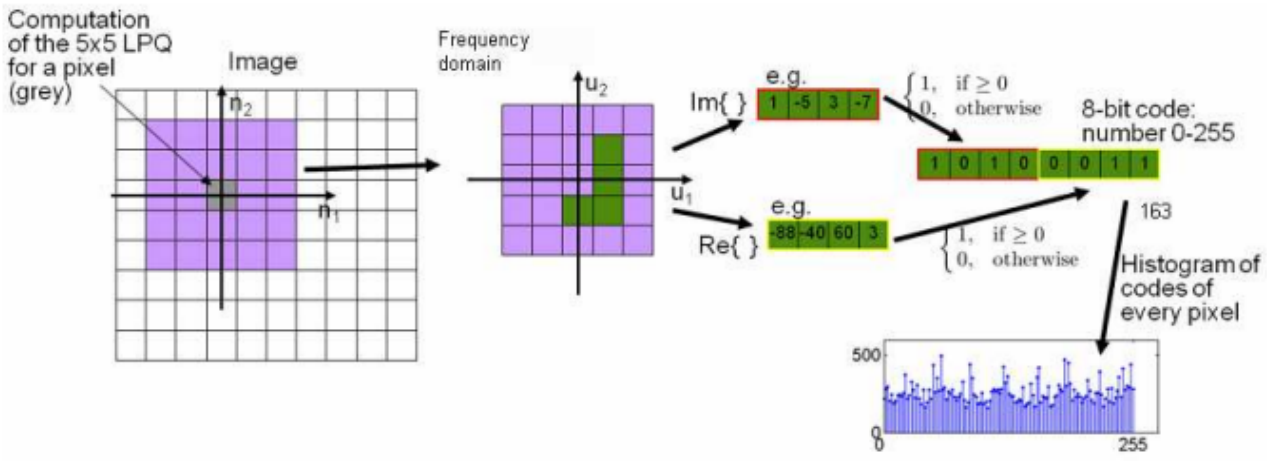

Figure 2.6: Implementation scheme of LPQ descriptor
Image Source: [32]

2.1.1.3 Dense Colour Histogram

For calculating Dense Colour Histogram (DCH), image is densely divided into a grid of overlapping local patches, and each patch is represented by a feature vector concatenating colour histograms computed around its local region. A colour histogram in LAB colour space is extracted from each patch. LAB colour histograms are com-

puted on multiple downsampled scales and L2 normalised.

In this work, 16-bin colour histograms are computed in each LAB channels, and in each channel, 3 levels of downsampling are used with scaling factors 0.5, 0.75 and 1, grid step of 10 and patch size of 11 x 11 pixels. This gives us a feature vector of length 2304.

2.1.1.4 GIST

The Gist descriptor was proposed in [33]. Gist is a model for recognition of real world scenes. It is based on a low dimensional representation of the scene, termed by the authors as the spatial envelope. It provides an overview of a scene, thus the name Gist. In this work, the Gist descriptor is computed by convolving the image with 32 Gabor filters at 4 scales and 8 orientations. This produces 32 feature maps each with the same size as the input image. Each feature map is then divided into 16 regions by a 4 x 4 grid and the average value for each region is computed. Finally, a 512 element Gist descriptor is obtained by concatenating the 16 averaged values for each of the 32 feature maps. Thus, Gist provides a rough description of the scene by summarising the gradient information (scales and orientations) for different parts of the image. Intuitively, visually similar images will have similar Gist feature vectors.

2.1.1.5 Four-Patch Local Binary Pattern

Wolf et al. [34] proposed a novel patch method based on local binary patterns (LBPs), comparing four patches to produce a single bit value in the code assigned to the central pixel. Four-patch local binary pattern (FPLBP) captures more local information than the typical LBP descriptor. The following equations represent the FPLBP descriptor:

$$FPLBP_{R_1,R_2,\omega,Q,\alpha}(C_p) = \sum_{i=0}^{\frac{Q}{2}-1} F\left(d_i - d_{i+\frac{Q}{2}}\right) 2^i \tag{2.8}$$

$$d_i = D(P_{c1,i}P_{c2,(i+\alpha)}\text{mod}Q) \tag{2.9}$$

$$d_{i+\frac{Q}{2}} = D\left(P_{c1,i+\frac{Q}{2}}P_{c2,(i+\frac{Q}{2}+\alpha)\text{mod}Q}\right) \tag{2.10}$$

where C_p is the central pixel. $C_{p1,i}$ and $C_{p2,i}$ represent the central pixel of a $w \times w$ patch located in the inner and outer circles, respectively. $D(:,:)$ and $F(x)$ is distance and similarity measure functions, respectively.

In this work, the FPLBP descriptor is computed with parameters $Q = 8$, $w = 3$ and $\alpha = 1$ giving a feature vector of length 192.

2.1.2 Training the classifier

Support vector machine (SVM) [35] is one of the most powerful classification methods that is based on statistical learning theory. This supervised learning algorithm in many cases gives high prediction accuracy than well established classification algorithms such as neural networks [36].

2.1.2.1 Support vector machine

Support vector machine is a supervised learning method. The core idea is to map the nonlinear input vector to a high dimensional feature space and construct the optimal hyperplane, as shown in Figure 2.7. In the case of linearly separable data, the hyperplane is given by Eq. 2.11

$$w \cdot x + b = 0 \tag{2.11}$$

where "." means the dot product of vectors, w is a vector and b is a scalar.

In the case of the linearly non-separable data, hyperplane (2.11) is unable to separate two classes. It can be mapped to a higher dimensional space by using nonlinear mapping function, as shown in Figure 2.8. Mapping function ϕ(x) is reflected in the form of kernel function $K(x_i, x_j)$. The following hyperplane is obtained after

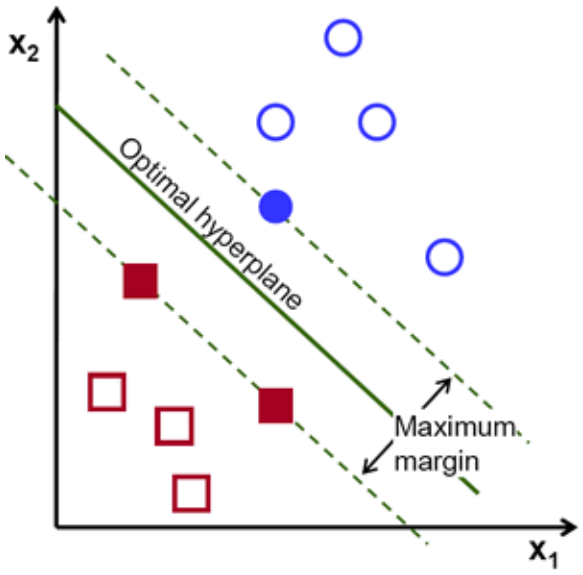

Figure 2.7: Maximum margin optimal hyperplane separating two classes.
Image Source: [37]

transformation:

$$\mathrm{W}^{*}\phi(\mathrm{x}) + \mathrm{b} = 0 \tag{2.12}$$

Figure 2.9 shows the effect of kernel transformation i.e the data which is not linearly separable in input space becomes separable in feature space. By using kernel function, the computation of separating hyperplane can be done without carrying out the mapping into feature space explicitly.

In addition to the linear case, there are three basic forms of the kernel function:

$$\text{polynomial}: \ \mathrm{K}(x_i,\ x_j) = (\gamma x_i^T x_j + \mathrm{r})^d, \gamma > 0. \tag{2.13}$$

$$\text{RBF}: \qquad \mathrm{K}(x_i,\ x_j) = \exp(-\gamma \|x_i - x_j\|^2), \gamma > 0. \tag{2.14}$$

$$\text{sigmoid}: \mathrm{K}(x_i,\ x_j) = \tanh(\gamma x_i^T x_j + \mathrm{r}). \tag{2.15}$$

where γ, r and d are all parameters.

In this work, binary SVM classifier is used for classification in hand/non-hand classes.

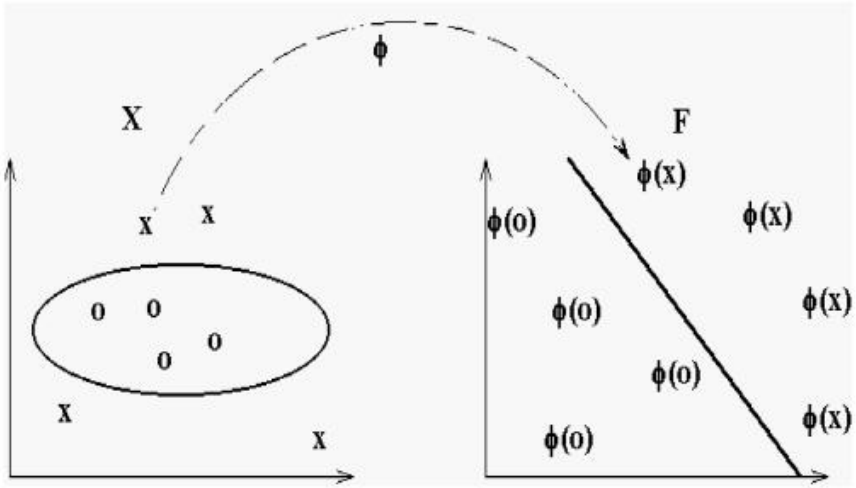

Figure 2.8: Illustration of the mapping ϕ
Image Source: [38]

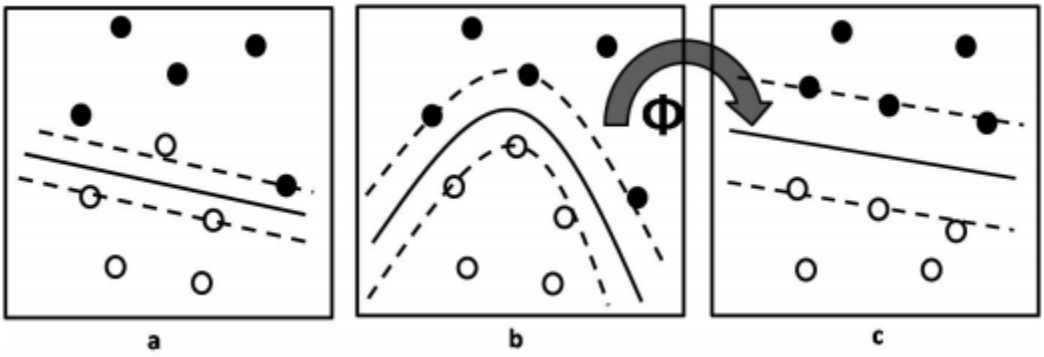

Figure 2.9: Kernel transformation. (a) data not linearly separable in input space. (b) data only separable by non linear surface. (c) data linear separable in kernel-mapped feature space.
Image Source: [39]

SVM model of polynomial kernel of order 3 is trained after features are extracted on training image dataset.

2.2 Validation phase

For validation phase, images have been collected in the same way as the training images but from the validation dataset [2]. In this step, various features and classifier models are explored and best among them is deployed for testing phase. Details of validation phase is covered in section 3.2.

2.3 Testing phase

There can be lots of hands in a single image and not all of them would be of the same size. Also, different images could have different number of hands at different scales. If we searched for a hand using a sliding window approach [2] at multiple scales, it would give us more than 100000 proposals and we would have to check each of these proposal separately and identify it as a hand or non-hand. This greatly increases the computation time. So instead of using sliding window approach, a region proposal based method is used in this work, which gives significantly less number of proposals. After getting the region proposals, features (HOG, GIST, FPLBP and DCH) of these proposals are extracted and fed into trained classifier model for a hand or non-hand label. Following sections give detailed description of all steps involved in testing phase, as shown in block diagram Figure 2.10. Figure 2.11 shows a sample from test dataset [2] for illustration purpose.

2.3.1 Gray world algorithm

The Gray world algorithm was proposed by Buchsbaum [40]. It estimates the illuminant by assuming that a certain spatial spectral average exists for total visual field.

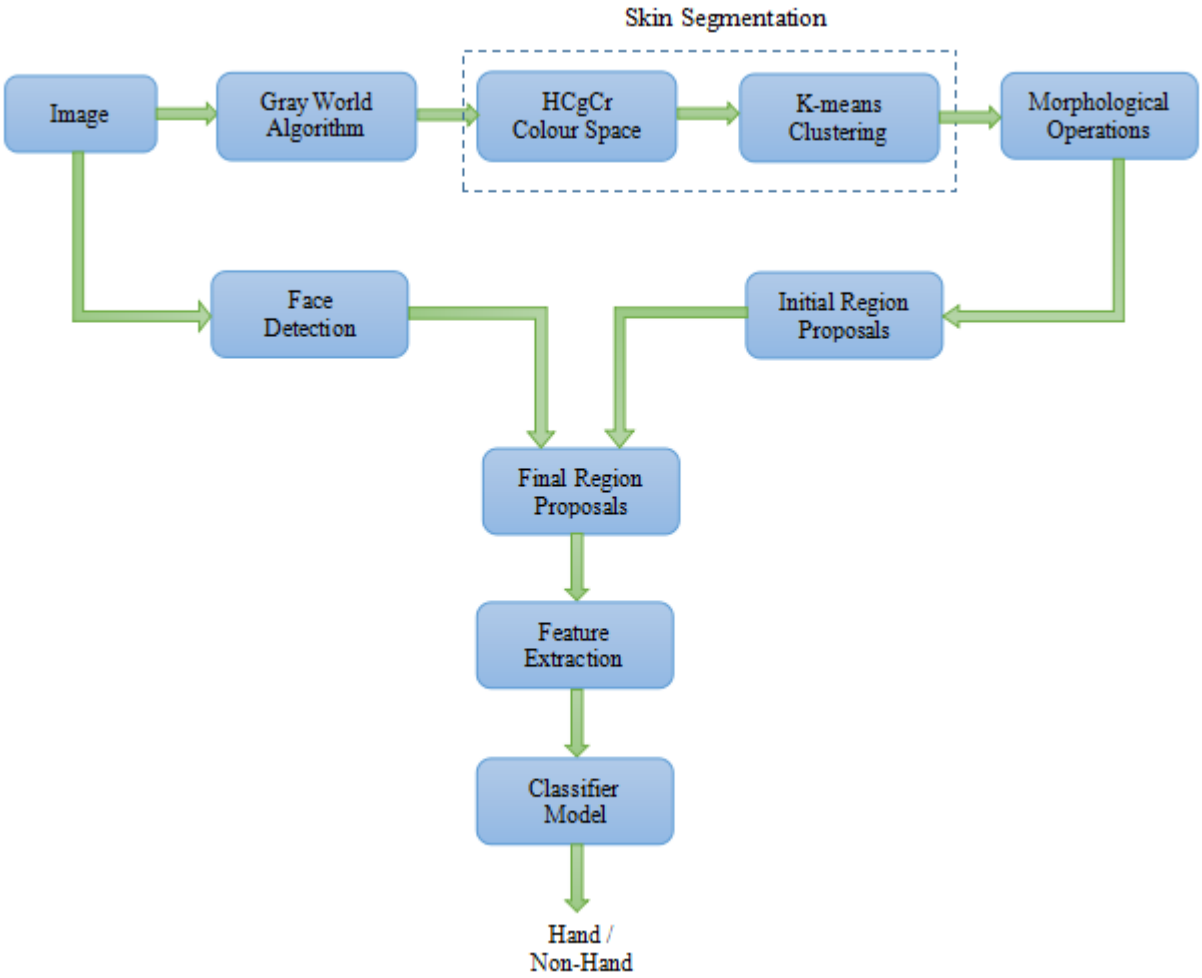

Figure 2.10: Detailed block diagram of testing phase

Figure 2.11: Sample image from test dataset

Mathematically, the gray world assumption can be expressed as a scaling of the three colour channels with reference to the gray value in order to achieve an average of gray across the entire image. Initially the gray value, y, was calculated using Equation 2.16 . Using this gray value, the scaling factors of the red, green and blue channels were calculated using Equation 2.17 and 2.18 respectively.

$$y = \frac{r_m + g_m + b_m}{3}, \tag{2.16}$$

where

$$r_m = \frac{\Sigma_{i=1}^{n} r_i}{n},\ g_m = \frac{\Sigma_{i=1}^{n} g_i}{n},\ b_m = \frac{\Sigma_{i=1}^{n} b_i}{n} \tag{2.17}$$

and n is the total number of pixels in the image. The scaling factors are defined as

$$\alpha_R = \frac{y}{r_m},\ \alpha_G = \frac{y}{g_m},\ \alpha_B = \frac{y}{b_m} \tag{2.18}$$

2.3.2 Skin segmentation

The major goal of skin segmentation is to discriminate between skin and non-skin pixels. This is usually accomplished by introducing a metric, which measures distances of pixel colour to skin tone. Skin colour has been proven to be a useful and robust cue for hand detection [2, 41]. Numerous techniques for skin colour modelling and recognition have been proposed in the past years [42–46]. Colour allows fast processing and is highly robust to geometric variations of the skin pattern. The studies suggest that human skin has a characteristic colour, which is easily recognised by humans. So employing skin segmentation proved to be an important step for hand detection. One of the major questions in using skin colour in skin detection is how to choose a suitable colour space. A rapid survey of common colour spaces is given in following sections.

2.3.2.1 RGB colour space

RGB is a colour space originated from CRT display applications (or similar applications), when it is convenient to describe colour as a combination of three coloured rays (red, green and blue). It is one of the most widely used colour spaces for processing and storing of digital image data. However, high correlation between channels, significant perceptual non-uniformity, mixing of chrominance and luminance data make RGB not a very favourable choice for colour analysis and colour based recognition algorithms [47].

2.3.2.2 Normalised RGB colour space

Normalised RGB is a representation that is easily obtained from the RGB values by a simple normalisation procedure:

$$r = \frac{R}{R+G+B} \tag{2.19}$$

$$g = \frac{G}{R+G+B} \tag{2.20}$$

$$b = \frac{B}{R+G+B} \tag{2.21}$$

As the sum of the three normalised components is known (r + g + b = 1), the third component does not hold any significant information and can be omitted, reducing the space dimensionality. The remaining components are often called "pure colours", for the dependence of r and g on the brightness of the source RGB colour is diminished by the normalisation. A remarkable property of this representation is for matte surfaces: while ignoring ambient light, normalised RGB is invariant (under certain assumptions) to changes of surface orientation relatively to the light source. This, together with the simplicity of the transformation, helped this colour space to gain popularity among researchers [48, 49].

2.3.2.3 HSV colour space

Hue-saturation based colour spaces are introduced when there is a need for the user to specify colour properties numerically. They describe colour with intuitive values, based on the artists idea of tint, saturation and tone. Hue defines the dominant colour (such as red, green, purple and yellow) of an area, saturation measures the colourfulness of an area in proportion to its brightness [8]. The intensity, lightness or value is related to the colour luminance as shown in Figure 2.12. The intuitiveness of the colour space components and explicit discrimination between luminance and chrominance properties made this colour space popular in the works on skin segmentation. However, there are several undesirable features of this colour space, including hue discontinuities and the computation of brightness, which conflicts badly with the properties of colour vision.

$$H = arccos\frac{1/2[(R-G)+(R-B)]}{\sqrt{(R-G)^2+(R-B)(G-B)}} \tag{2.22}$$

$$S = 1 - 3\frac{(R,G,B)}{R+G+B} \tag{2.23}$$

$$V = \frac{R+G+B}{3} \tag{2.24}$$

An alternative way of Hue-Saturation computation using log opponent values was introducing additional logarithmic transformation of RGB values aimed to reduce the dependence of chrominance on the illumination level. The polar coordinate system of Hue-Saturation space, resulting in cyclic nature of the colour space makes it inconvenient for parametric skin colour models that need tight cluster of skin colours for best performance. Here, different representations of Hue-Saturation using Cartesian coordinates can be used [47]:

$$X = ScosH; Y = SsinH \tag{2.25}$$

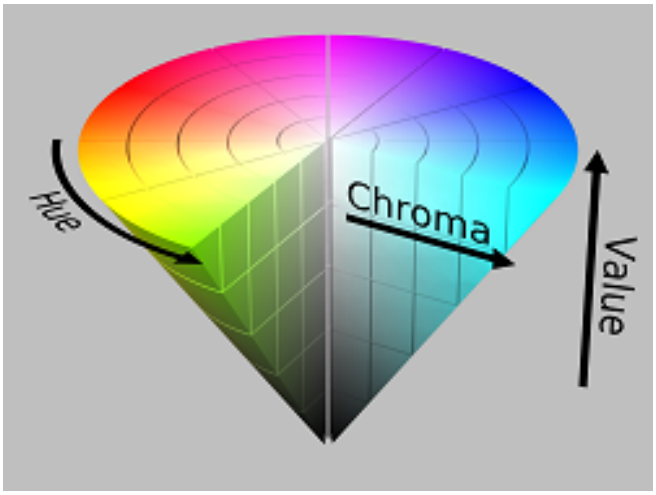

Figure 2.12: HSV colour space
Image Source: [50]

2.3.2.4 YCbCr colour space

YCbCr is an encoded nonlinear RGB signal, commonly used by European television studios and for image compression work. Colour is represented by luma (which is luminance, computed from nonlinear RGB [51]), constructed as a weighted sum of the RGB values, and two colour difference values Cr and Cb that are formed by subtracting luma from the red and blue components in RGB [52] as shown in Figure 2.13. The simplicity of the transformation and explicit separation of luminance and chrominance components makes this colour space attractive for skin colour modelling.

$$Y = 0.299R + 0.587G + 0.114B \tag{2.26}$$

$$Cr = R - Y \tag{2.27}$$

$$Cb = B - Y \tag{2.28}$$

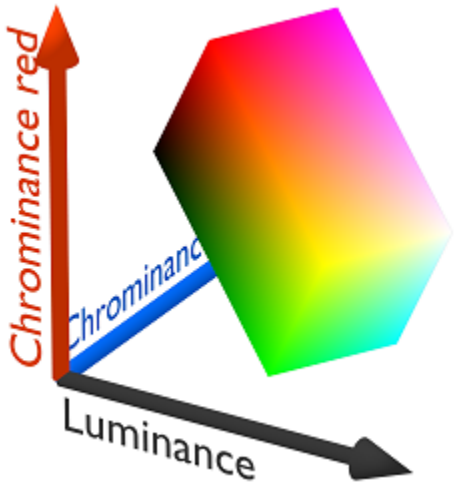

Figure 2.13: YCbCr colour space
Image Source: [53]

2.3.2.5 YCgCr colour space

This new colour space, proposed by JJ De Dios et al. [54], was developed for face detection. This novel colour space, YCgCr, uses the smallest colour difference (G-Y) instead of (B-Y). It is defined exclusively for skin analysis applications, mainly for face segmentation. Expressed in matrix form, RGB components can be easily transformed to YCgCr components:

$$\begin{bmatrix} Y \\ Cg \\ Cr \end{bmatrix} = \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix} + \begin{bmatrix} 65.481 & 128.553 & 24.966 \\ -81.085 & 112 & -30.915 \\ 112 & -93.768 & -18.214 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{2.29}$$

In this work, a hybrid colour space is created by combining two colour spaces HSV (Hue, Saturation, Value) and YCgCr (luminance, chrominance in green, chrominance in red). The H, Cg and Cr components are used to form a hybrid HCgCr colour space as shown in Figure 2.14. Exclusion of the luminance components (V and Y) makes this skin detection method less susceptible to illumination variation. Figure 2.15 shows the sample image in HCgCr colour space.

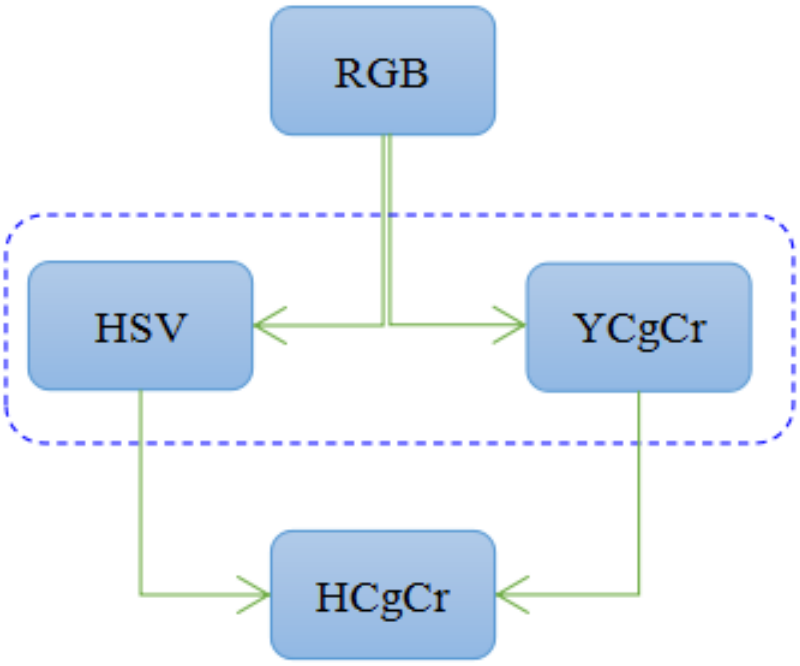

Figure 2.14: Creation of HCgCr colour space

Figure 2.15: Image in HCgCr colour space

After image is converted in HCgCr space, next step is to segment the image into skin and non-skin regions. The segmentation step uses K-means clustering algorithm [55].

2.3.2.6 K-means clustering

K-means clustering is an unsupervised learning algorithm. The flow path of this clustering algorithm is as follows [56]:

1. Determine the number of clusters K.

2. Determine the value of the centroids.

In determining the centroid value for the initial iteration, the initial value of the centroid is randomised or heuristically chosen. Meanwhile, in determining the value of the centroid, which is at the stage of iteration, the formula used is as follows:

$$\overline{V_{ij}} = \frac{1}{N_i} \sum_{k=0}^{N_i} X_{kj}, \tag{2.30}$$

where:

- i, k is the index of the cluster.
- j is the index of the variable.
- v_{ij} is the centroid of the i^{th} cluster for the j variable.
- N_i is the amount of data that belongs to the i^{th} cluster.
- X_{kj} is the value of the k^{th} data present in the cluster for the j^{th} variable.

1. Initialise k clusters centroids.

2. Calculate the distance between the centroid point c_i with each sample point x_i. In this work, cityblock distance has been used which is given by Equation 2.31.

$$d_{cityblock}(u, v) = \sum_{i=1}^{N} |x_i - c_i| \tag{2.31}$$

3. Sample point is assigned to that cluster which has shortest distance between sample point and cluster centroid.

4. Iterate through steps 2 and 3 until the centroid value is fixed and the cluster members do not move to another cluster.

Figure 2.16 shows the output of skin segmentation process. Since the output image has some noise, morphological operations (dilation and erosion) are applied to obtain a smooth, closed, and clean image as shown in Figure 2.17. These are referred to as initial region proposals, as shown in block diagram 2.10. These proposals also include faces which should be removed to reduce unnecessary proposals. This is done with the help of face detection algorithm.

2.3.3 Region proposals

Region proposal methodology identifies prospective objects in an image using segmentation. After getting initial region proposals, these are refined by removing face regions using Viola-Jones face detection algorithm.

2.3.3.1 Viola-Jones face detection algorithm

The Viola-Jones object detection framework [15] can be used for detecting objects in real time but it is mainly applied to face detection application. The detection rate of this framework is quiet high which makes the algorithm robust and it also processes the images quickly. Four main steps followed in this algorithm are:

Figure 2.16: Skin Segmentation

Figure 2.17: Morphologically smoothed image

1. Haar-like features [57] are used for the feature extraction.
2. An integral image is formed by computing the rectangles adjacent to the rectangle present at (x,y) into a single image representation which helps in speeding the algorithm.
3. Adaboost learning algorithm is used to build the classifier to be trained.
4. The final method is cascade classifier which can efficiently combine many features.

Once the strong classifiers are cascaded, the face regions can be detected. These face regions are removed from initial region proposals to get final region proposals as shown in Figure 2.18.

Figure 2.18: Final region proposals

Figure 2.19 shows region proposals on sample RGB image. As can be seen from figure, region proposals mostly include skin regions. Some skin-like regions are also considered as proposals.

After we get final region proposals, HOG, GIST, FPLBP, and DCH features are extracted from each proposals and fed into trained classifier model for a hand/non-hand label. Figure 2.20 shows output of the proposed approach on sample image.

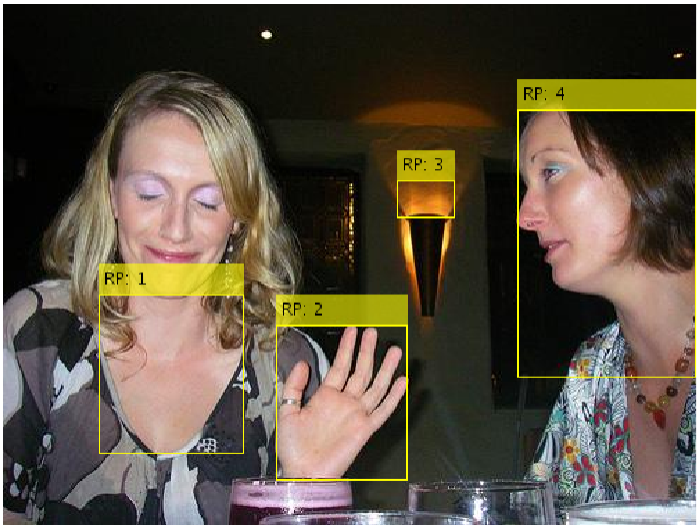

Figure 2.19: Region proposals shown in sample image

Figure 2.20: Detected hand in sample image

Chapter 3

Dataset, Experiments And Discussion

3.1 Dataset and evaluation measure

In order to evaluate the performance of the proposed method, Oxford hand dataset and NUS hand posture datasets are used in this work.

3.1.1 Oxford hand dataset

Oxford hand dataset [2] is a comprehensive dataset of hand images collected from various public image data set sources as shown in Figure 3.1.

The images have no restriction imposed on the pose, visibility of people, or on the environment. In each image, all the hands that can be perceived clearly by humans are annotated. The annotations consist of a bounding rectangle, which does not have to be axis aligned and oriented with respect to the wrist. There are total of 13050 annotated hand instances. Examples are shown in Figure 3.2.

Training Dataset			Validation Dataset		
Source	#Instances	#Big Instances	Source	#Instances	#Big Instances
Buffy Stickman	887	438	Movie Dataset	1856	649
INRIA pedestrian	1343	137	*Total*	1856	649
Poselet (H3D)	1355	580	Test Dataset		
Skin Dataset [2]	703	139	Source	#Instances	#Big Instances
PASCAL VOC 2007 train and val set	1867	507	PASCAL VOC 2007 test set	1626	562
PASCAL VOC 2010 train and val set (except human layout set)	3008	1060	PASCAL VOC 2010 human layout val set	405	98
Total	9163	2861	*Total*	2031	660

Figure 3.1: Statistics of the hand dataset
Image Source: [2].

Figure 3.2: Sample images from the hand dataset with bounding box annotations overlaid
Image Source: [2].

3.1.2 NUS hand posture dataset I

The NUS hand posture dataset I [3] is a very small database of colour images (gray scale images are also provided) where the images are captured using uniform background colour. There are ten hand pose classes and 24 samples per class. The image size is 160 x 120. The hand poses in the NUS hand posture dataset I are illustrated in Figure 3.3.

Figure 3.3: Sample images from NUS I dataset
Image Source: [3].

3.1.3 NUS hand posture dataset II

The NUS hand posture dataset II [4] contains ten hand pose classes. These classes are similar but not the same as in the NUS hand posture dataset I mentioned above. The images are captured in varying environments by 40 subjects of different ethnic backgrounds and ages. In total, there are 2000 hand images and 2000 background images. The image size is 160 x 120. The hand poses in the NUS hand posture dataset II are illustrated in Figure 3.4.

Evaluation measure The performance is evaluated using average precision (AP) (the area under the Precision Recall curve). As used in PASCAL VOC [58], a hand detection is considered true or false according to its overlap with the ground-truth bounding box. A box is positive if the overlap score is more than 0.5, where the

Figure 3.4: Sample images from NUS II dataset
Image Source: [4].

overlap score (O) between two boxes is defined by Equation 3.1:

$$O = \frac{area(B_g \cap B_d)}{area(B_g \cup B_d)}, \tag{3.1}$$

where B_g is the ground-truth bounding box and B_d is the detected bounding box.

3.2 Experiments

The performance of the proposed approach is evaluated on the publicly available database, namely Oxford hand dataset [2] and is detailed in the following sections.

3.2.1 Validation experiments

Validation experiments have been carried out to select best features and classifier model suitable for hand detection task. For this purpose, validation images have been collected the same way as the training images but from the validation dataset. Validation dataset comprises total 5411 images, out of which 3207 are hand images and 2204 are non-hand images. The performance is evaluated using average precision (AP) (the area under the Precision Recall curve).

As can be seen from Table 3.1, SVM classifier trained on DCH, HOG, GIST and

FPLBP features yields higher average precision on validation dataset and hence is used in this thesis.

Table 3.1: Performance of SVM and LDA classifier for various features on validation dataset

Features	SVM				LDA			
	Recall (%)	Precision (%)	AP (%)	Acc(%)	Recall (%)	Precision (%)	AP (%)	Acc(%)
DCH	91.80	99.53	**98.57**	94.46	87.28	86.68	92.37	84.62
HOG	91.95	98.78	**98.14**	81.09	94.16	89.74	90.60	81.51
GIST	98.97	93.38	**98.40**	95.23	86.81	88.71	93.86	85.32
FPLBP	83.48	93.17	**90.27**	85.03	72.71	83.10	81.37	71.50
LPQ	70.41	47.27	70.66	56.97	77.88	70.37	**84.38**	70.59

Figures 3.5 and 3.6 shows Precision Recall curve comparing various features used in training SVM and LDA classifiers respectively on validation dataset.

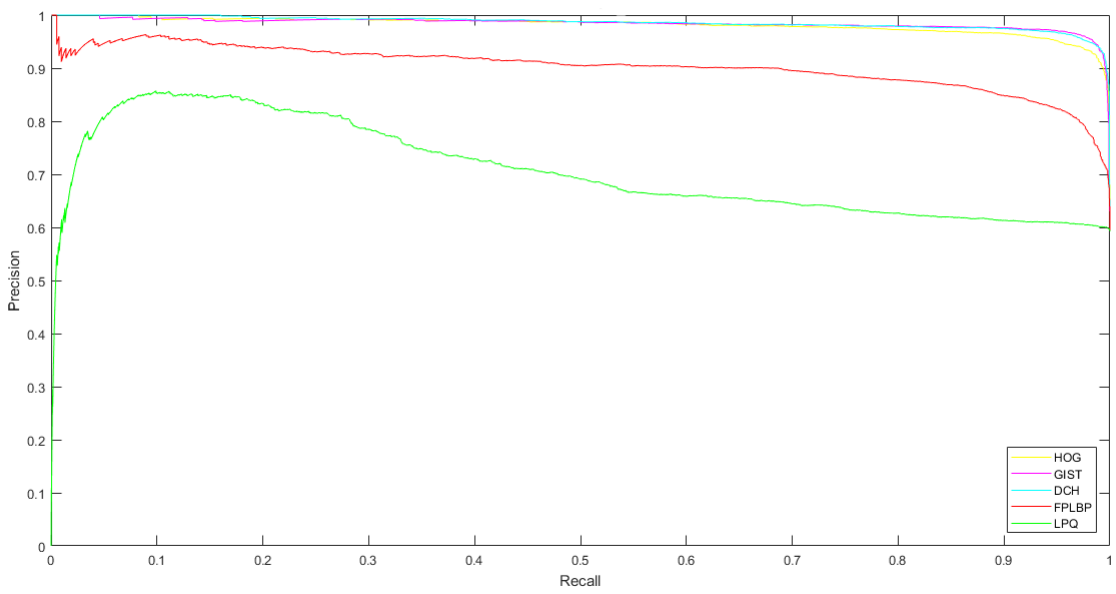

Figure 3.5: PR curve comparing various features using SVM classifier on validation dataset

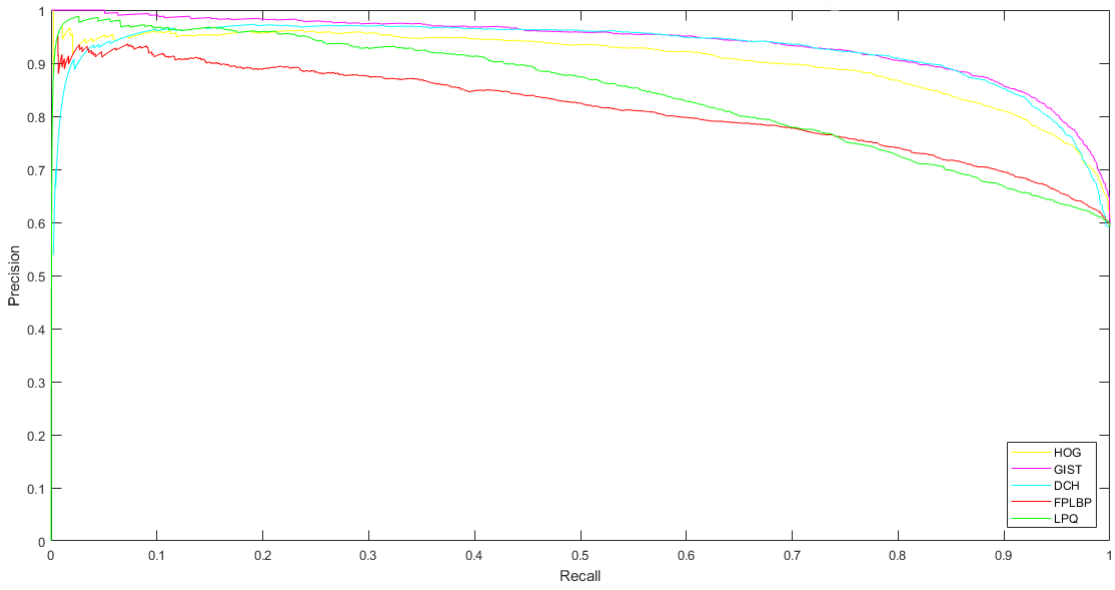

Figure 3.6: PR curve comparing various features using LDA classifier on validation dataset

3.2.2 Test experiments

Evaluation of the proposed work is done on widely used Oxford hand dataset [2] consisting of 436 images. The images are processed as mentioned in Section 2.3. The performance is evaluated using average precision (AP) as used in PASCAL VOC challenge.

3.2.3 Comparative analysis

For a comparison of performance, we have compared the performance of proposed approach with the one proposed by Mittal et al. [2], Roy K. et al. [41], Deng et al. [30] along with RCNN [59] and FRCNN [60] based hand detection method on Oxford hand dataset. Table 3.2 shows this comparison with the results reported in their published works.

Table 3.2: Performance comparison with existing approaches

Approach	Avg. Precision (%)	Avg. Execution Time (in secs)
Mittal *et al.* [2]	42.30	120.0
Denge *et al.* [30]	58.10	8.0
Roy K *et al.* [41]	49.51	-
RCNN [59]	31.23	9.0
FRCNN [60]	14.22	0.08
Proposed work	**30.58**	**7.0**

3.3 Discussion

Hand detection is a formidably challenging problem due to cluttered backdrops. The performance of the proposed hand detection algorithm is demonstrated on the publicly available images of Oxford hand dataset. In this thesis, we have explored the performance of region proposal and feature-based method for hand detection.

We have compared performance of two classifiers namely, SVM and LDA, for various features and subsequently selected only those features which gave high average precision on validation dataset. Hence, HOG, DCH, GIST and FPLBP features were used for training the SVM classifier model.

We have compared the performance (in terms of average precision) of the proposed approach with some of the existing works. The execution time in processing a single image in testing phase is also reported using MATLAB R2017a, on a system with Intel Core i3 CPU clocking at 1700 MHz with 4 GB RAM. As demonstrated in Table 3.2, we observed a substantial improvement over FRCNN method [60] since performance of this method is degraded when objects to be detected are small in size. The proposed work has comparable performance with RCNN method [59] in terms of average precision but performs better in terms of execution time. Our method have average execution time of 7 seconds per image, as compared to 120 seconds per image needed by Mittal et al. [2] method. This reduction in execution time is due

to the use of region proposal method based on skin segmentation rather than sliding window approach used in [2]. The proposed approach is weaker than state-of-the-art methods which uses CNN-based architecture [30, 41]. The reason for this is we chose to explore feature-based technique rather than using deep networks which are computationally intensive and require large amount of data to prevent overfitting. Furthermore, analysing as well as visualising features in an intuitive fashion is still a topic of research in deep networks. Hyper-parameters tuning and network architecture design are also quite challenging. Because of all these reasons, we attempted to use region proposal and feature-based method which gave reasonable performance as compared to works done by [2, 59, 60].

The proposed algorithm failed to detect hands in the presence of shadows, severe occlusions and illumination conditions. Also, in some cases, other skin-like regions are confused with hands giving false positives due to which average precision is reduced.

Chapter 4

Conclusion And Future Work

4.1 Conclusion

This work is an attempt to contribute towards the solution of hand detection problem. The proposed method is based on region proposals generated by skin segmentation. This is followed by the extraction of local texture, shape and colour based descriptors, which are then fed into trained classifier model for hand detection.

The performance of the proposed approach is evaluated on publicly available hand dataset, namely Oxford hand dataset. This dataset contains 13050 images annotated with bounding box around hand region that are collected from various public image datasets. The dataset is considered to be diverse as there is no restriction imposed on the pose or visibility of people and background environment. This dataset has much cluttered background, more viewpoint variations and articulated shape changes. Both validation and test experiments are performed and results are presented using average precision (AP) and intersection over union metric. It is found that performance of the proposed approach on Oxford hand dataset achieves an average precision of 30.58% with average execution time of 7 seconds per image.

4.2 Future work

The proposed method is only based on single pixel colour information without using any texture information, hence other features such as texture features can be explored in the future, which in turn will be helpful to build more accurate region proposals. When the background colour is very close to the hand colour, it is still a challenging situation to classify even using the best solution with multiple colour spaces. Also, the colour information can be extracted at super-pixel level rather than from single pixel. The robustness of the algorithm to poor illumination, shadows, blur etc. can also be improved.

In addition to this, performance of the detector can be improved by retraining the SVM model using the hard negatives generated after the testing phase instead of just the training phase. Execution time can be further reduced such that hand detection can be done in real-time.

References

[1] "Histogram of oriented gradients descriptor," https://software.intel.com/en-us/ipp-dev-reference-histogram-of-oriented-gradients-hog-descriptor.

[2] A. Mittal, A. Zisserman, and P. H. Torr, "Hand detection using multiple proposals," in *Proceedings of the British Machine Vision Conference*, 2011, pp. 1–11.

[3] P. P. Kumar, P. Vadakkepat, and A. P. Loh, "Hand posture and face recognition using a fuzzy-rough approach," *International Journal of Humanoid Robotics*, vol. 7, no. 3, pp. 331–356, 2010.

[4] P. K. Pisharady, P. Vadakkepat, and L. A. Poh, "Hand posture and face recognition using fuzzy-rough approach," in *Computational Intelligence in Multi-Feature Visual Pattern Recognition*, 2014, pp. 63–80.

[5] S. S. Rautaray and A. Agrawal, "Vision based hand gesture recognition for human computer interaction: a survey," *Artificial Intelligence Review*, vol. 43, no. 1, pp. 1–54, 2015.

[6] C. Li and K. M. Kitani, "Pixel-level hand detection in ego-centric videos," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 3570–3577.

[7] A. Aristidou and J. Lasenby, "Motion capture with constrained inverse kinematics for real-time hand tracking," in *International Symposium on Communications, Control and Signal Processing*, 2010, pp. 1–5.

[8] M. Andriluka, S. Roth, and B. Schiele, "Pictorial structures revisited: People detection and articulated pose estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 1014–1021.

[9] Y. Yang, C. Fermuller, Y. Li, and Y. Aloimonos, "Grasp type revisited: A modern perspective on a classical feature for vision," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 400–408.

[10] P. Buehler, M. Everingham, D. P. Huttenlocher, and A. Zisserman, "Long term arm and hand tracking for continuous sign language tv broadcasts," in *Proceedings of the British Machine Vision Conference*, 2008, pp. 1105–1114.

[11] M. J. Jones and J. M. Rehg, "Statistical color models with application to skin detection," *International Journal of Computer Vision*, vol. 46, no. 1, pp. 81–96, 2002.

[12] L. Sigal, S. Sclaroff, and V. Athitsos, "Skin color-based video segmentation under time-varying illumination," *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

[13] M. Kolsch and M. Turk, "Hand tracking with flocks of features," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2005, pp. 1187–vol.

[14] P. Pérez, C. Hue, J. Vermaak, and M. Gangnet, "Color-based probabilistic tracking," in *Proceedings of the European Conference on Computer Vision*, 2002, pp. 661–675.

[15] P. Viola and M. J. Jones, "Robust real-time face detection," *International Journal of Computer Vision*, vol. 57, no. 2, pp. 137–154, 2004.

[16] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2005, pp. 886–893.

[17] B. Stenger, A. Thayananthan, P. H. Torr, and R. Cipolla, "Model-based hand tracking using a hierarchical bayesian filter," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 9, pp. 1372–1384, 2006.

[18] I. Oikonomidis, N. Kyriazis, and A. A. Argyros, "Markerless and efficient 26-dof hand pose recovery," in *Proceedings of the Asian Conference on Computer Vision*, 2010, pp. 744–757.

[19] P. Buehler, M. Everingham, D. P. Huttenlocher, and A. Zisserman, "Upper body detection and tracking in extended signing sequences," *International Journal of Computer Vision*, vol. 95, no. 2, p. 180, 2011.

[20] Y. Sheikh, O. Javed, and T. Kanade, "Background subtraction for freely moving cameras," in *Proceedings of the International Conference on Computer Vision*, 2009, pp. 1219–1225.

[21] H. Trinh, Q. Fan, P. Gabbur, and S. Pankanti, "Hand tracking by binary quadratic programming and its application to retail activity recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 1902–1909.

[22] I. Oikonomidis, N. Kyriazis, and A. A. Argyros, "Efficient model-based 3d tracking of hand articulations using kinect," in *Proceedings of the British Machine Vision Conference*, 2011, p. 3.

[23] P. K. Pisharady, P. Vadakkepat, and A. P. Loh, "Attention based detection and recognition of hand postures against complex backgrounds," *International Journal of Computer Vision*, vol. 101, no. 3, pp. 403–419, 2013.

[24] E.-J. Ong and R. Bowden, "A boosted classifier tree for hand shape detection," in *Proceedings of the IEEE Conference on Automatic Face and Gesture Recognition*, 2004, pp. 889–894.

[25] M. Kölsch and M. Turk, "Robust hand detection," in *Proceedings of the IEEE Conference on Automatic Face and Gesture Recognition*, 2004, pp. 614–619.

[26] N. H. Do and K. Yanai, "Hand detection and tracking in videos for fine-grained action recognition," in *Proceedings of the Asian Conference on Computer Vision*, 2014, pp. 19–34.

[27] J. Tompson, M. Stein, Y. Lecun, and K. Perlin, "Real-time continuous pose recovery of human hands using convolutional networks," *ACM Transactions on Graphics*, vol. 33, no. 5, p. 169, 2014.

[28] S. Sridhar, F. Mueller, A. Oulasvirta, and C. Theobalt, "Fast and robust hand tracking using detection-guided optimization," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3213–3221.

[29] T. H. N. Le, Y. Zheng, C. Zhu, K. Luu, and M. Savvides, "Multiple scale faster-rcnn approach to drivers cell-phone usage and hands on steering wheel detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2016, pp. 46–53.

[30] X. Deng, Y. Zhang, S. Yang, P. Tan, L. Chang, Y. Yuan, and H. Wang, "Joint hand detection and rotation estimation using cnn," *IEEE Transactions on Image Processing*, vol. 27, no. 4, pp. 1888–1900, 2018.

[31] V. Ojansivu and J. Heikkilä, "Blur insensitive texture classification using local phase quantization," in *Proceedings of the International Conference on Image and Signal Processing*, 2008, pp. 236–243.

[32] "Cvpr 2011 tutorial," http://www.ee.oulu.fi/research/imag/mvg/files/pdf/CVPR-tutorial-final.pdf.

[33] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *International Journal of Computer Vision*, vol. 42, no. 3, pp. 145–175, 2001.

[34] L. Wolf, T. Hassner, and Y. Taigman, "Descriptor based methods in the wild," in *Workshop on Faces in 'Real-Life' Images: Detection, Alignment, and Recognition*, 2008, pp. 1–5.

[35] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995.

[36] C. Kyrkou and T. Theocharides, "A parallel hardware architecture for real-time object detection with support vector machines," *IEEE Transactions on Computers*, vol. 61, no. 6, pp. 831–842, 2012.

[37] "Introduction to support vector machines," https://docs.opencv.org/2.4.13.4/doc/tutorials/ml/introduction-to-svm/introduction-to-svm.html.

[38] "Polynomial kernel," https://en.wikipedia.org/wiki/Polynomial-kernel.

[39] Y. KK, "Texture detection using svm for mobility assistant for the visually impaired system," Ph.D. dissertation, Indian Institute of Technology Delhi, 2016.

[40] G. Buchsbaum, "A spatial processor model for object colour perception," *Journal of the Franklin Institute*, vol. 310, no. 1, pp. 1–26, 1980.

[41] K. Roy, A. Mohanty, and R. R. Sahay, "Deep learning based hand detection in cluttered environment using skin segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 640–649.

[42] S. L. Phung, A. Bouzerdoum, and D. Chai, "A novel skin color model in ycbcr color space and its application to human face detection," in *Proceedings of the International Conference on Image Processing*, 2002, pp. I–I.

[43] V. Vezhnevets, V. Sazonov, and A. Andreeva, "A survey on pixel-based skin color detection techniques," in *Proceedings of the Graphicon*, 2003, pp. 85–92.

[44] J. Y. Lee and S. I. Yoo, "An elliptical boundary model for skin color detection," in *Proceedings of the International Conference on Imaging Science, Systems, and Technology*, 2002, pp. 400–407.

[45] A. Albiol, L. Torres, and E. J. Delp, "Optimum color spaces for skin detection," in *Proceedings of the International Conference on Image Processing*, 2001, pp. 122–124.

[46] Y. Wu and X. Ai, "Face detection in color images using adaboost algorithm based on skin color information," in *Proceedings of the International Workshop on Knowledge Discovery and Data Mining*, 2008, pp. 339–342.

[47] J. I. B. Monge, *Hand Gesture Recognition for Human-Robot Interaction*. Skolan för datavetenskap och kommunikation, Kungliga Tekniska högskolan, 2006.

[48] C. A. Doukim, J. A. Dargham, and A. Chekima, "Comparison of three colour spaces in skin detection," *Borneo Science*, vol. 24, no. 3, pp. 75–81, 2009.

[49] J. A. Dargham and A. Chekima, "Lips detection in the normalised rgb colour scheme," in *Proceedings of the Information and Communication Technologies*, 2006, pp. 1546–1551.

[50] "Hsl and hsv," https://en.wikipedia.org/wiki/HSL-and-HSV.

[51] R. A. Redner and H. F. Walker, "Mixture densities, maximum likelihood and the em algorithm," *Society for Industrial and Applied Mathematics review*, vol. 26, no. 2, pp. 195–239, 1984.

[52] M. Störring, "Computer vision and human skin colour," Ph.D. dissertation, Citeseer, 2004.

[53] "Ycbcr," https://en.wikipedia.org/wiki/Talk:YCbCr.

[54] J. J. De Dios and N. García, "Face detection based on a new color space ycgcr," in *Proceedings of the International Conference on Image Processing*, 2003, pp. III–909.

[55] J. MacQueen *et al.*, "Some methods for classification and analysis of multivariate observations," in *Proceedings of the Berkeley Symposium on Mathematical Statistics and Probability*, 1967, pp. 281–297.

[56] N. Wakhidah, "Clustering menggunakan k-means algorithm (k-means algorithm clustering)," *Universitas Semarang*, vol. 8, no. 1, pp. 33–39, 2010.

[57] J. Whitehill and C. W. Omlin, "Haar features for facs au recognition," in *Proceedings of the International Conference on Automatic Face and Gesture Recognition*, 2006, pp. 5–pp.

[58] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, 2010.

[59] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the*

IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 580–587.

[60] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Proceedings of the Advances in Neural Information Processing Systems*, 2015, pp. 91–99.