Computer-Vision Based Nepali Sign Language Interpretation

Project Members:

Priti Oli (068Bex432)

Rabin Banjade (068Bex434)

Rupak Khatiwada (068Bex437)

Sandeep Neupane (068Bex440)

Project Supervisor

Mr. Dinesh Baniya Kshatri

Outline

- **Motivation**
- **Introduction**
- **Methodology**
- **Result**
- **Tools used**
- **Conclusion**
- **Limitations**

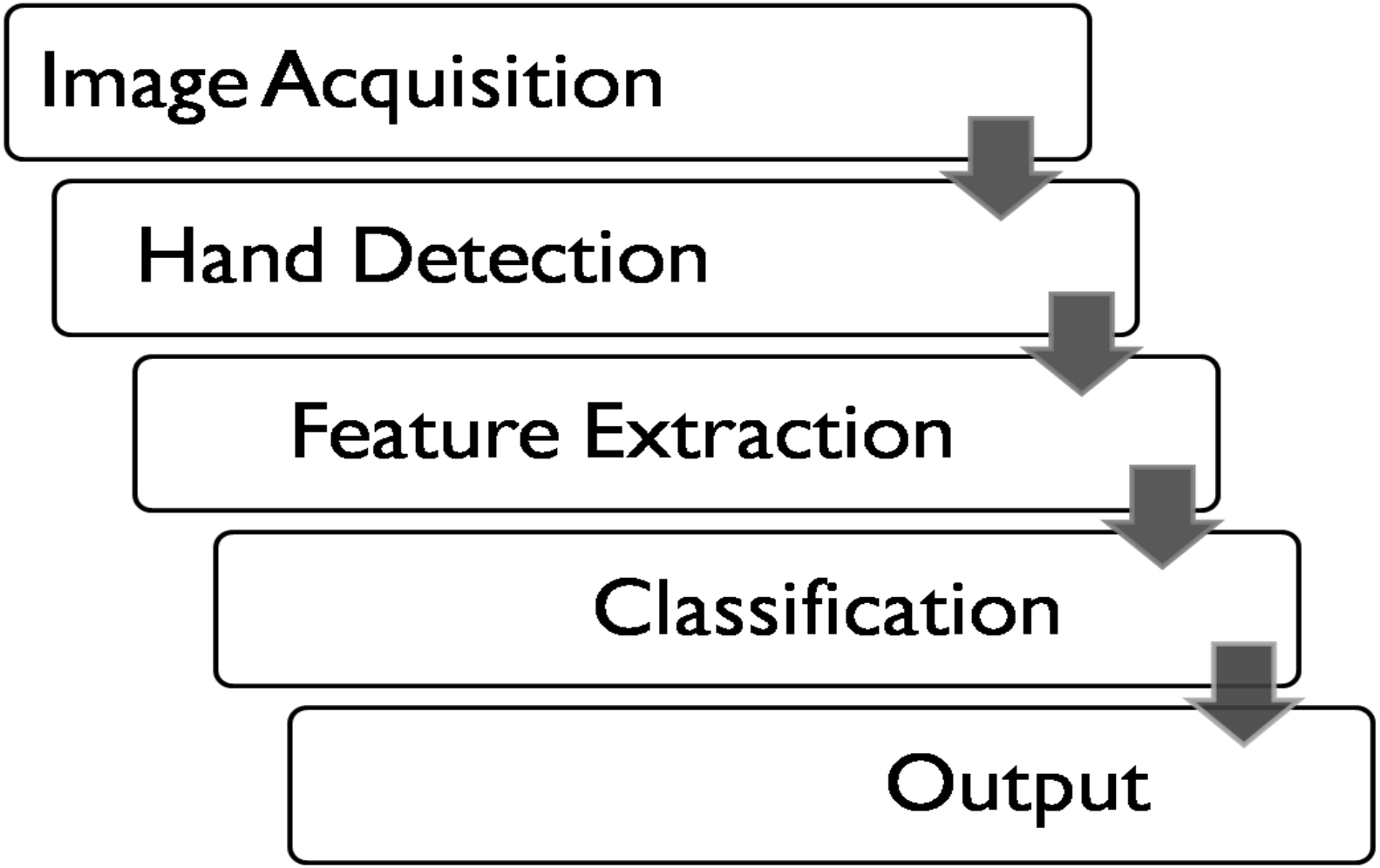
Motivation

- **How many of you know what these signs mean?**

Introduction

- **Convert Nepali Sign Language (NSL) into text and speech**
- **Computer vision and image processing utilized for:**
  - Acquiring and processing signs
- **Machine learning algorithms implemented for:**
  - Classifying signs
  - Converting signs into text and speech

Methodology



Methodology
(Image Acquisition)

- **Single camera for capturing image**

- **Laptop integrated webcam used**

- **Camera specifications:**
  - Frame Rate:10 Frames per second (FPS)
  - Camera Resolution:1.3 Mega Pixels (MP)
  - Resolution: 640 x 480 pixels

Methodology
(Background Subtraction)

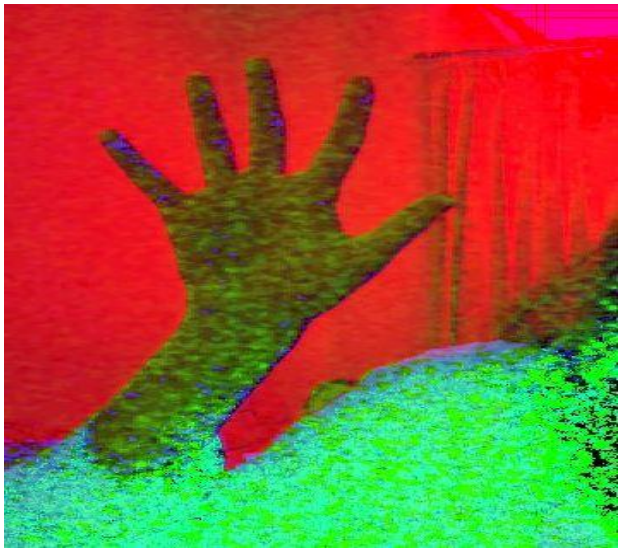
- **Image averaging method**
  - Learns average and standard deviation of pixels
  - Pixel frame is subtracted from the reference frame
  - Value greater than threshold then foreground, else background

Image with Person in Background

Image After Background Subtraction

Methodology
(Color-based Image Segmentation)

- **Histogram based segmentation**
  - YCbCr and HSV color spaces used
  - Thresholded image ORed with background subtraction



HSV Colorspace Image

YCrCb Colorspace Image

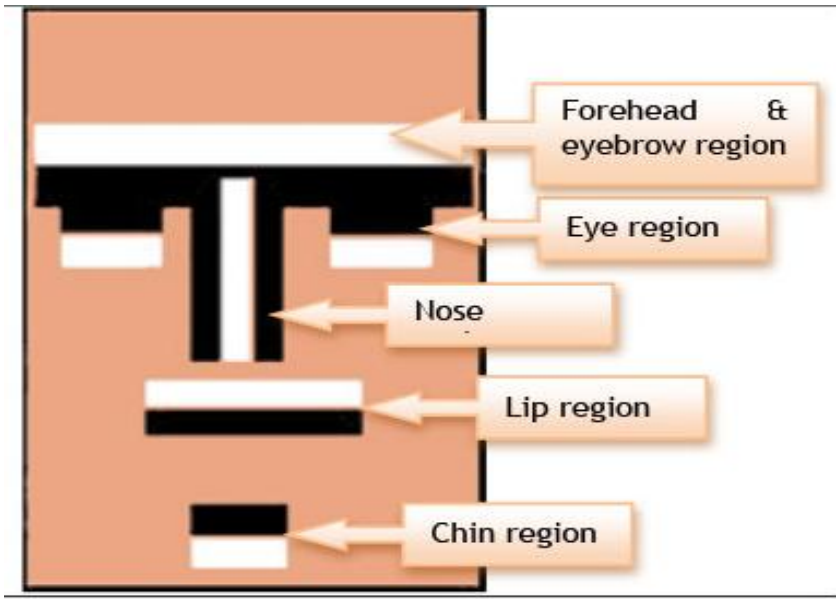
Methodology
(Image Binarization)

- **Adaptive Thresholding**
  - Otsu's thresholding
  - Different threshold values for different image portions
  - Used under varying lighting conditions
  - Automatically calculates the image thresholding value

Otsu's Thresholded Output

Methodology
(Face Detection : Haar Cascade)

- **Uses Haar features**
  - Groups the pixels into rectangular regions
  - Uses change in the contrast values



Haar Features for Face

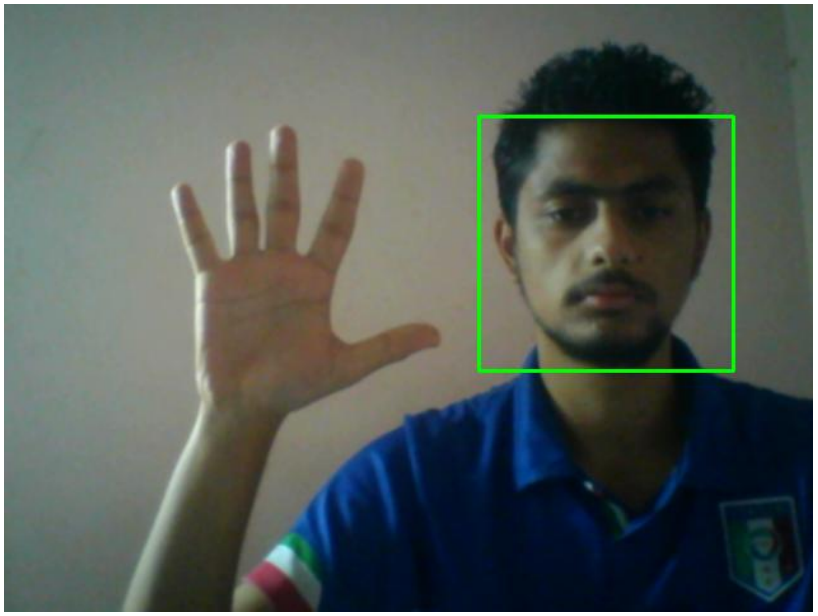
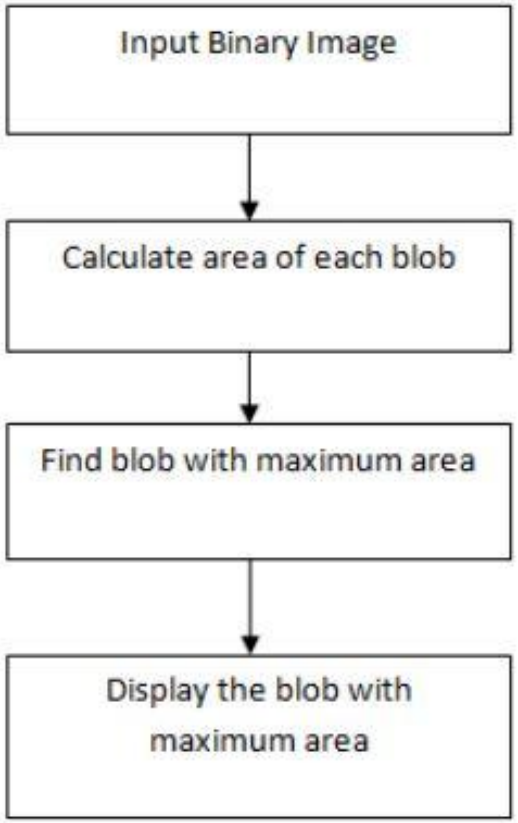


Image showing Detected Face

Methodology
(Noise Elimination:Blob Removal)



Block diagram of blob removal process

Methodology
(Morphological processing)

- **Morphological processing is used for :**
  - removal of noise in image
  - Segmentation of region of interest in image

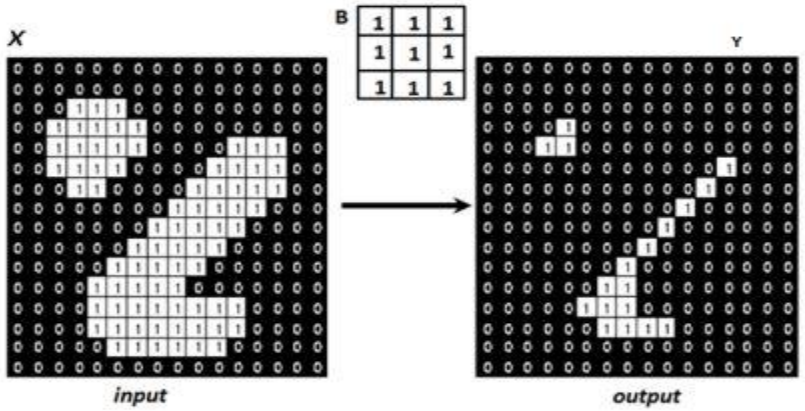


Image showing Erosion Operation by 3×3 kernel function

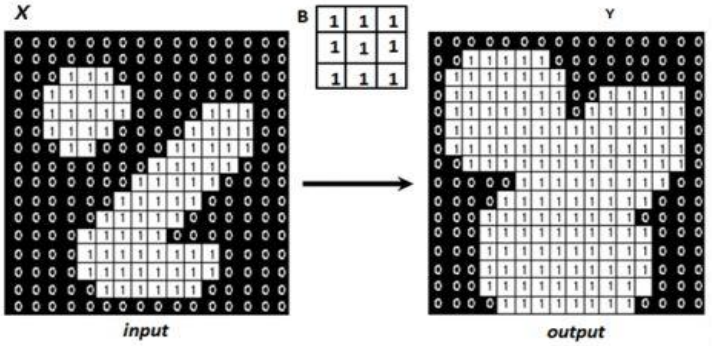


Image showing Dilatation operation by 3×3 kernel function

Methodology

(Palm Detection and Arm Detection)

- **Case 1: Image with Forearm but no Elbow**
  - **Palm Detection Process:**
    - AND maximum contour image with horizontal line image
    - Find line with maximum length (L) . This line corresponds to line through center of palm (approx.)
  - **Arm Removal Process:**
    - Distance (d) = L /2.0
    - Distance (d') = Distance below the center of palm where image will be cut
    - If d < 150:

      $$d' = (1.75 \times d) + 15$$

      Else: d' = d

Methodology
(Palm Detection and Arm Removal)

- **Case 1: Image with Forearm but no Elbow**

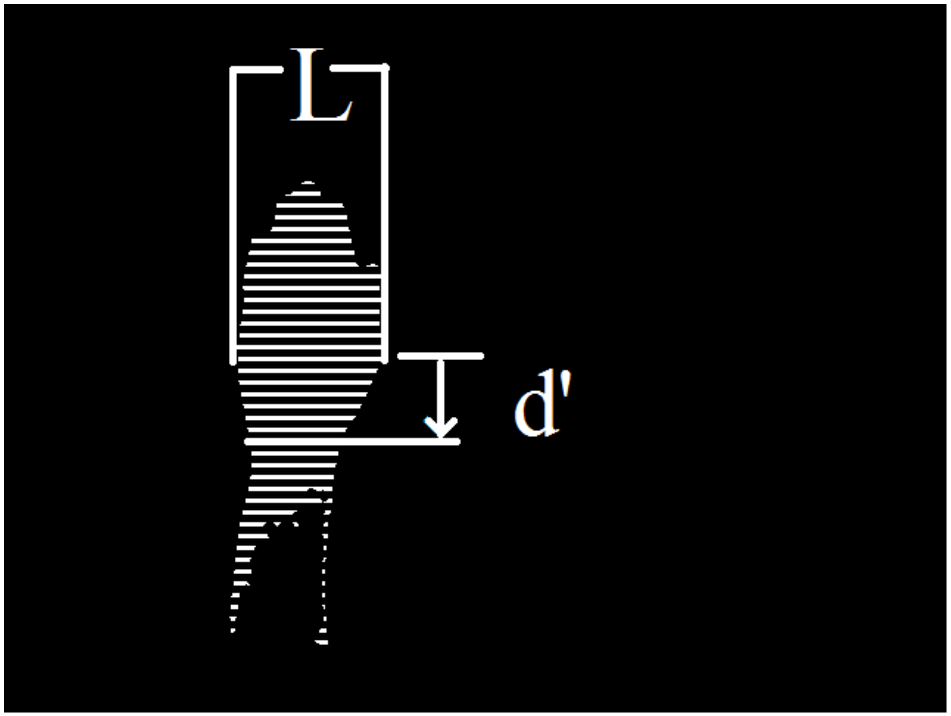
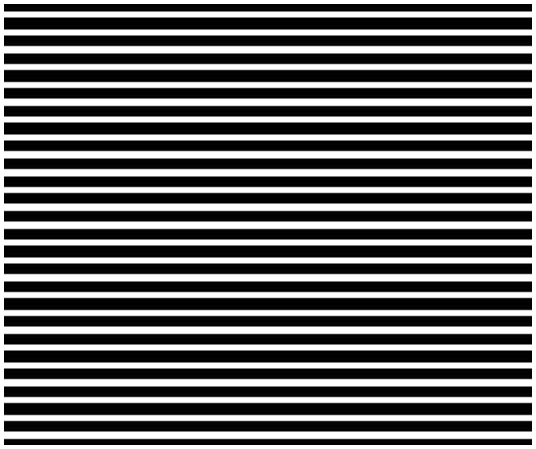


Image Showing Distance (d') to be Cut from Center of Palm

Methodology
(Palm Detection and Arm Removal)

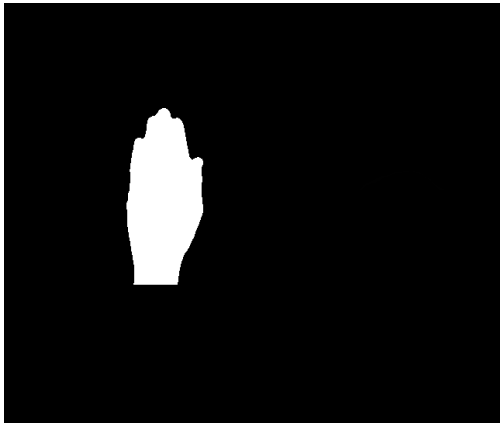


Horizontal line Image

ANDing

Image Containing Hand

Palm Detection



Final Image without Arm

Arm Removal

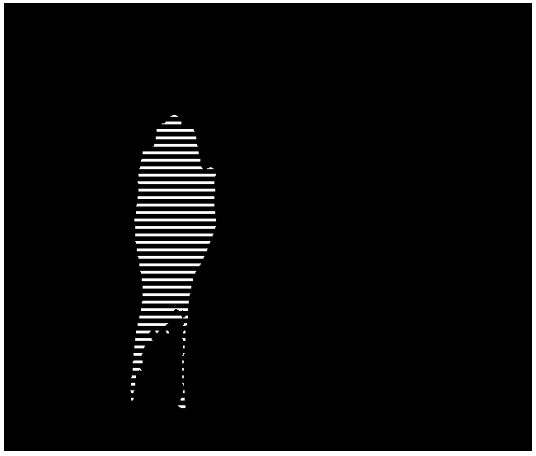


Image after ANDing

Methodology
(Palm Detection and Arm Detection)

- **Case 2: Image with Forearm and Elbow**
  - Find bounding rectangle around maximum contour image
  - Resize image to 30 pixels x 30 pixels
  - Divide image into 3 equal portions vertically (0:10), (11:20), (21:30)
  - Calculate area of second portion, i.e. (11:20)
  - If area below threshold (pixel count = 160 pixels)
    - Crop image from y to y + (0.7 × h) without changing width of bounding rectangle
      - where (y) is top right coordinate of bounding rectangle and (h) is height of bounding rectangle
  - Use Case 1(Image with Forearm but no Elbow) again

Methodology
(Case 2: Image with Forearm and Elbow)

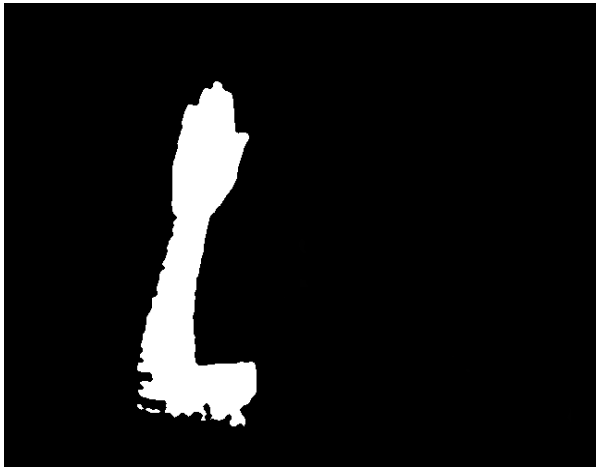
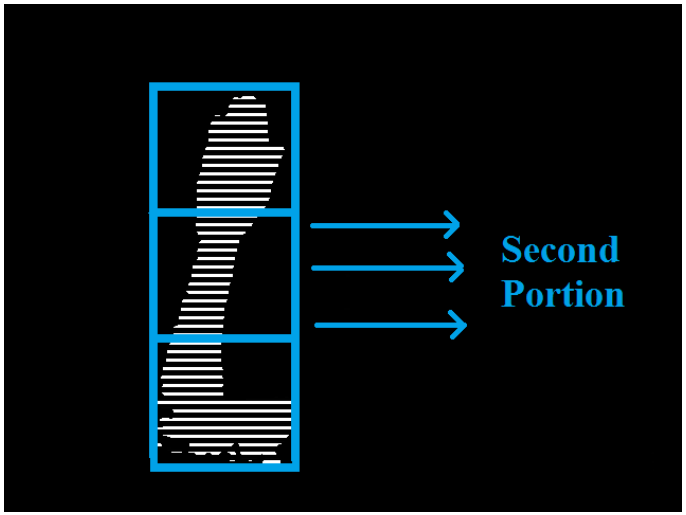


Image with elbow portion



Bounding rectangle in image

Palm Detection

Arm Removal

Final image after using algorithm 2

Methodology
(Feature Extraction)

- **Features are :**
  - measurable property of phenomenon
  - informative, discriminating and independent
  - crucial step for machine learning algorithms
- **Features for classification**
  - Hu moments
  - Raw Binary Image pixels
  - Combination of both the above features

Methodology

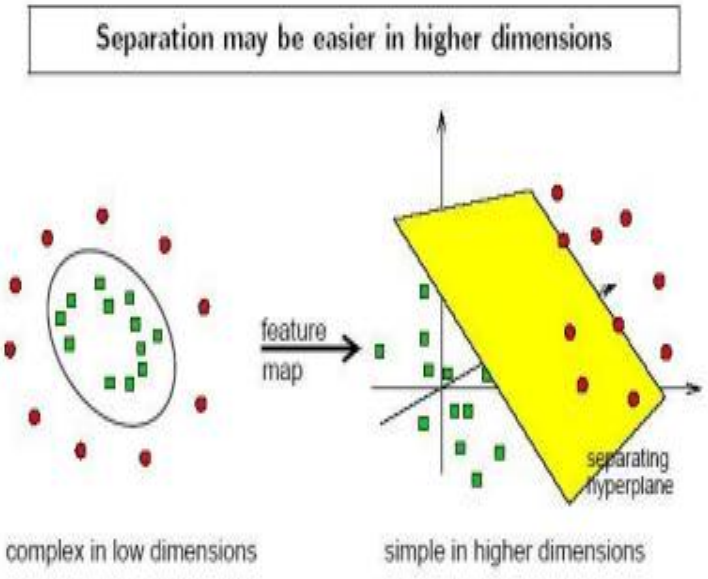
(Feature Extraction)

- **Binary image represent**
  - Image with only two values for a pixel
  - The image pixel represent ‘on’ or ‘off’
  - Foreground represent ‘1’ and background ‘0’
- **Hu moment**
  - Particular weighted average of pixel intensities
  - Moment invariants are used for invariance in translation , scaling and rotation

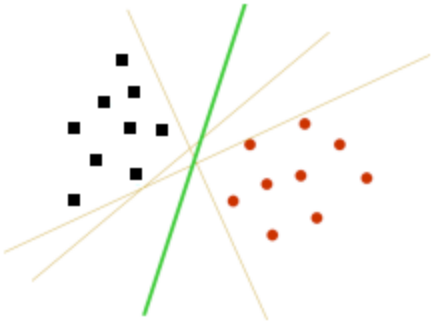
Support Vector Machines
(classification)

- **Supervised learning method**
- **classifier based on**
  - polynomial functions,
  - radial basis functions
  - splines or other functions
- uses a hyper-linear separating plane to create a classifier
- classifies by making a non-linear transformation of the original input space into a high dimensional feature space, where an optimal separating hyper plane can be found

Support Vector Machines
(classification)



Feature Mapping in Higher Dimension



Optimal Separating Hyperplane

Support Vector Machines (classification)

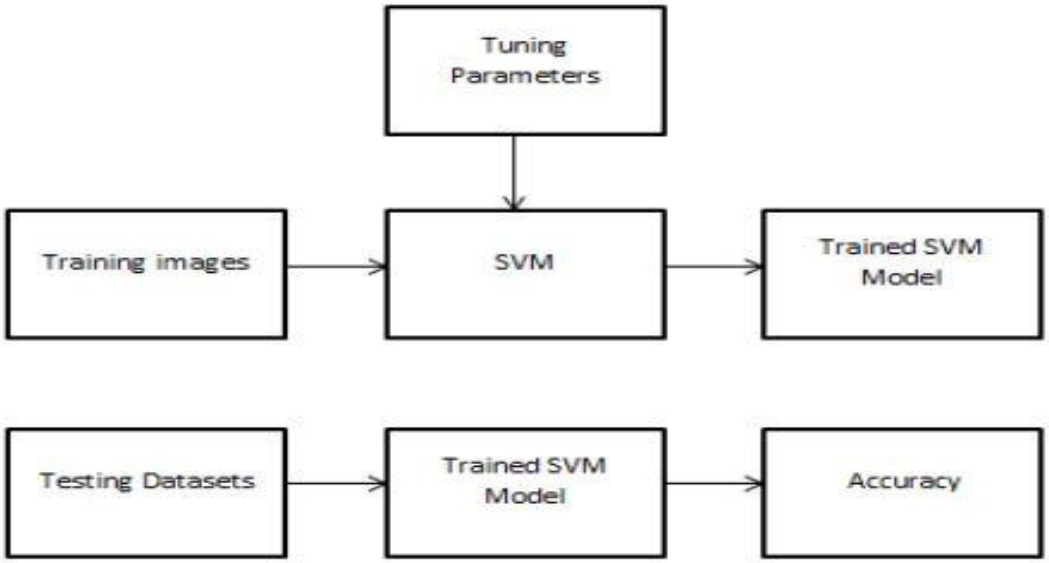
- **separating planes must be optimal,**
- **RBF (Radial Basis Function) kernel used**
- **Tuning parameters for SVM classification**
  - Grid search used

Training and Testing

- **Training**
  - To find the best values of variables in a decision function
  - Trained model is saved to minimize the testing time
- **Testing**
  - The trained classification model is loaded from the disk
  - Files with the vector of features whose classes are to be predicted are fed as input

Training and Testing

- Classifier returns the labels of unknown class of image



Block diagram of training and testing phase

Results

The results obtained can be summarized as below:

Attribute	Total Number of Instances	Correctly Classified	Incorrectly Classified	Accuracy
Raw Binary Pixel Values + Hu moment (Training)	6106	6103	3	99.95%
Hu Raw Binary Pixel Values Only (Testing)	693	628	65	90.62%
Raw Binary Pixel + Hu Moments (Testing)	693	634	59	91.49%

Text to speech

- **Obtained results were assigned corresponding speech file**
- **Both female and male voice were recorded**
- **Each sound is a wav file recorded of about 2-3 seconds**
- **Each sound clip is either Nepali consonants or numerals**

Tools used

- **Python**
  - Programming platform
- **PyQt4**
  - Graphic User Interface
- **Open Cv**
  - Image processing Library
- **Mat plot lib**
  - For plotting image properties
- **Scikit learn**
  - For classification
- **C pickle**

Limitations

- **Color segmentation**
  - Differences in hand color from person to person
- **Background learning**
  - not fully effective and prone to inaccuracies
- **Inability to subtract dynamic background**
  - Our approach fails in case of dynamic setting
- **Problem during varying lightning conditions**
- **Inability for classification of some signs like 'फ'**
  - Our algorithmic approach fails for certain signs

Improvements Over Existing Projects

- **Number of classified gestures**
  - 44 (“0” to “9” and “ka” to “gya”)
- **Accuracy of the classifier**
  - 91 % with both raw and hu moment as attributes
- **New algorithm for finding center of palm region**
- **Arm removal algorithm**
- **Classifier used : SVM**
- **Works under low light conditions**

Conclusion

- **More datasets can improve results for training.**
- **Foundation for effective communication**
- **Not limited to Nepali Sign language recognition**
- **Varying training data we can use our project for classification of other sign languages as well**

Answers..

s n d