Road Sign Detector using Image Processing and Machine Learning

Haidar Khan

Advisor: Dr. Vaziri

New Paltz
STATE UNIVERSITY OF NEW YORK

Overview

- Project Goals
- Problem & Motivation
- Background
- Approach
- Results
- Engineering Constraints
- Conclusion
- Future Work

Project Goal

- Design a system which can:
  - Detect the presence of selected road signs in images

New Paltz
STATE UNIVERSITY OF NEW YORK

Problem & Motivation

- Distracted driving can result in missing or misreading road signs
  - Tickets and fines
  - Accidents
- Challenging problem because input is binary data and output is meaningful information.

Possible Solution

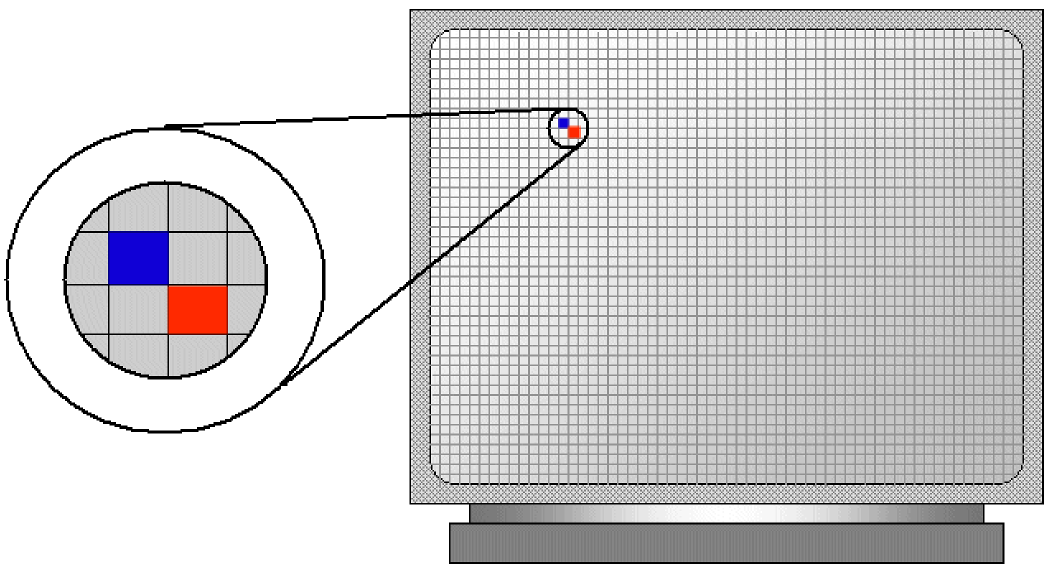
- A road sign detection system in a vehicle would:
  1. Capture the driver's field of view using a camera
  2. Detect selected road signs
  3. Report information to the vehicle operator

Background - Computer Vision

- This project falls under the field of computer vision
  - Extracting real information about the world from low level data (images)
  - Teaching computers to "see" as humans do

Background - Pixels

- Fundamental data structure - pixel
- Each pixel has a color property:
  - value(s) representing the color we see.
    - Typically [Red, Green, Blue]



Background – Color-spaces

- Multiple ways to represent the color of a pixel
  - Binary – 0 or 1
  - Grayscale – 1 value, typically 8 bit
  - Red-Green-Blue (RGB) – 3 values
  - Hue-Saturation-Value (HSV) – 3 values
  - Many more…

Background - Images

- Taken together pixels form an image matrix.
  - Each element is a pixel.
  - Same color-space

$$\begin{bmatrix} p_{1,1} & \cdots & p_{1,m} \\ \vdots & \ddots & \vdots \\ p_{n,1} & \cdots & p_{n,m} \end{bmatrix}$$

New Paltz
STATE UNIVERSITY OF NEW YORK

Background - Videos

- Videos are represented by a collection of images
  - Showing a number of images, or frames, one after another forms a video
  - The speed of the video is FPS or frames per second
    - Typically 25-30 FPS

Background - Image Processing

- Color-space Conversions
- Image Filtering
- Image Segmentation
- Feature Extraction

- Processing will be done using MATLAB

Background - Machine Learning

- Learn from training data, then classify new data.
- Two main types:
  - Supervised - training data is labeled
  - Unsupervised – training data is not labeled
- Supervised learning algorithm - Support Vector Machines (SVM)

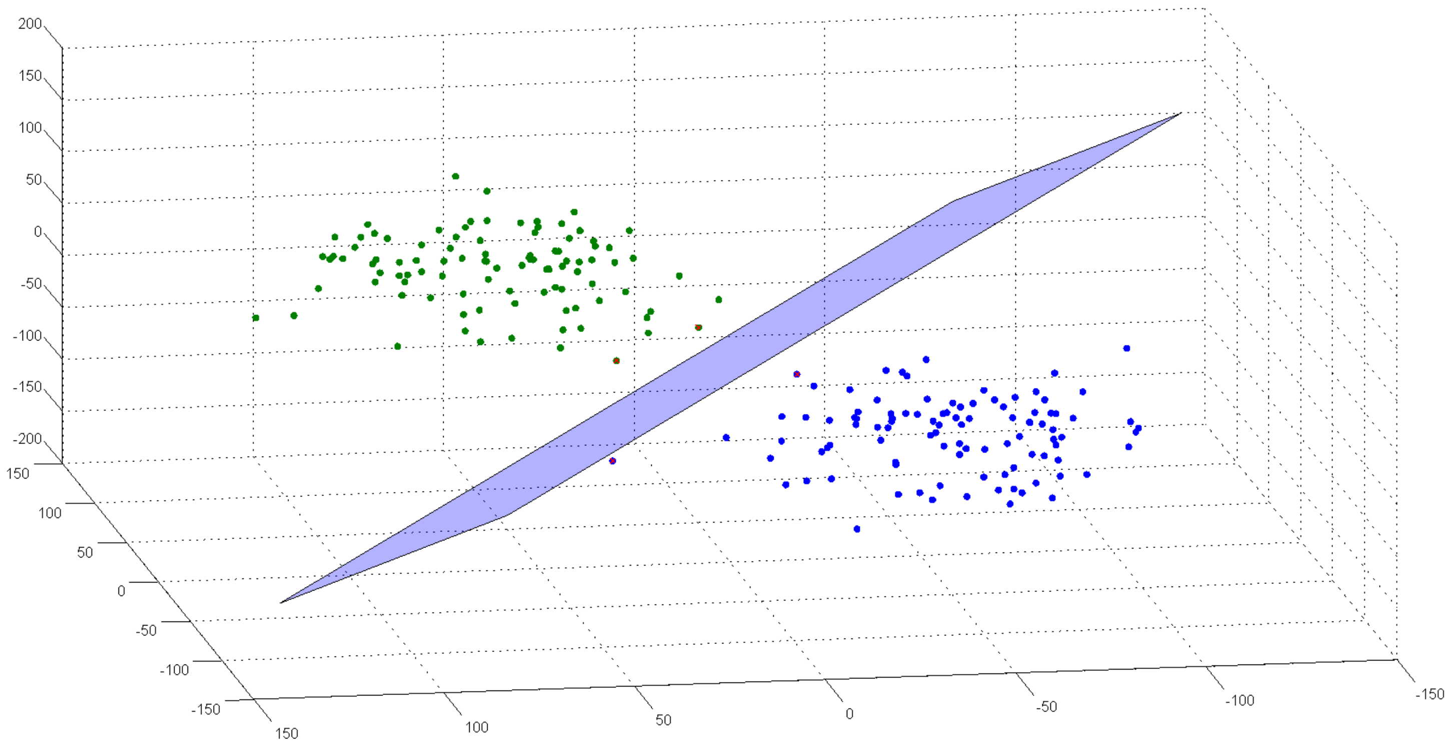
New Paltz
STATE UNIVERSITY OF NEW YORK

Background - Support Vector Machines

- Invented by Vladmir Vapnik, Bell Labs in 1995 [2]
- Binary classifier
  - Training data is labeled as belonging to one of two classes (+, -)
- Finds optimal separating hyper-plane between the two sets of training data.
- The hyper-plane is then used to classify new data.

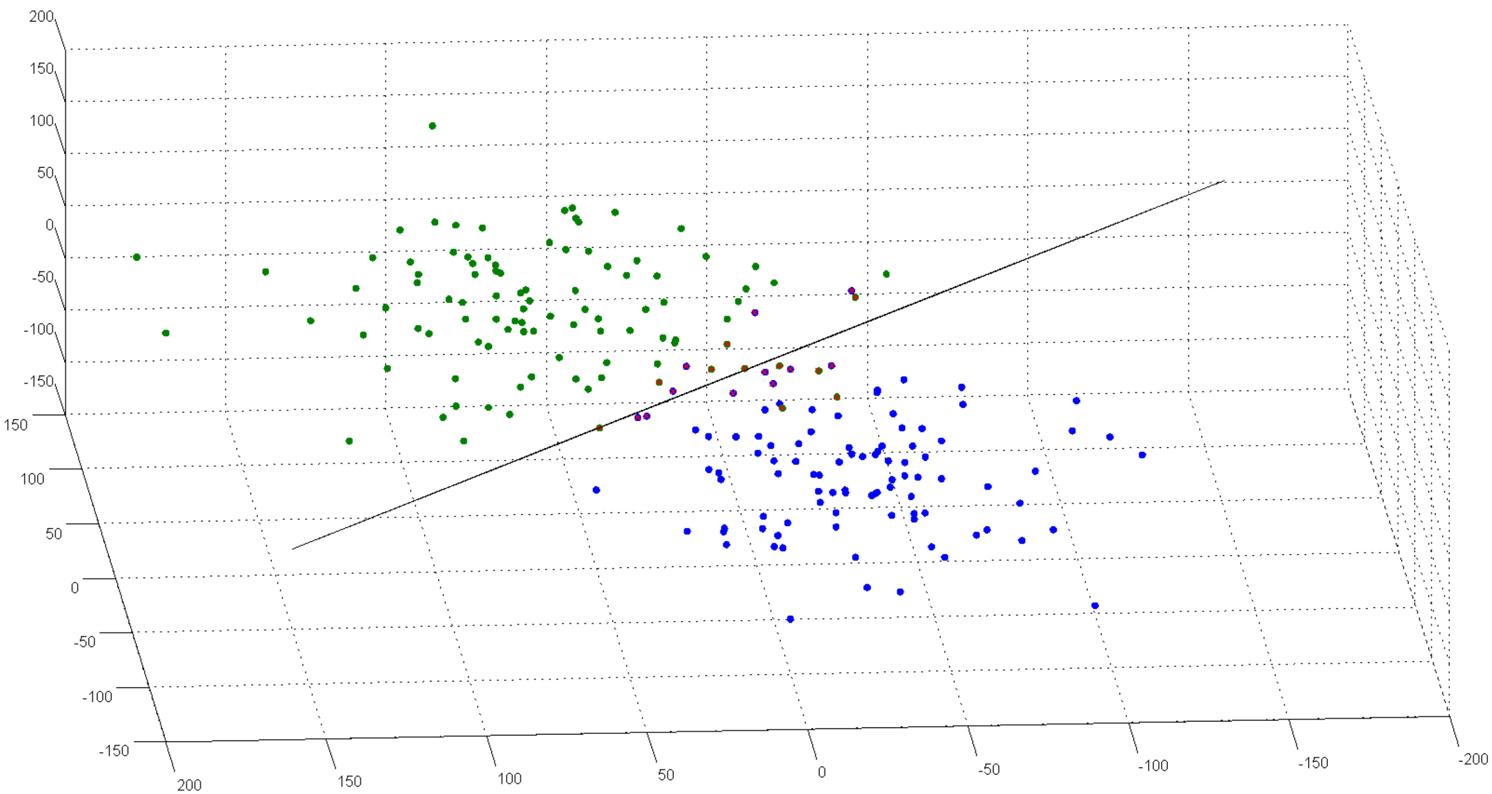
New Paltz
STATE UNIVERSITY OF NEW YORK

SVM – Separable Data



New Paltz
STATE UNIVERSITY OF NEW YORK

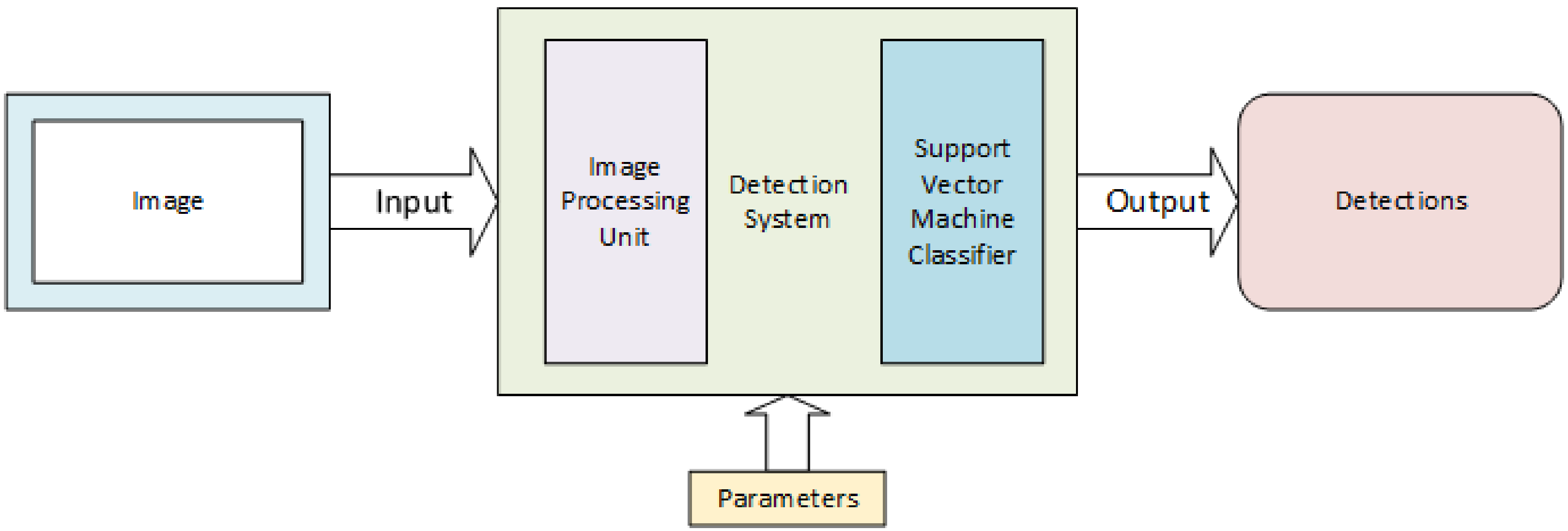
SVM – Non-separable Data



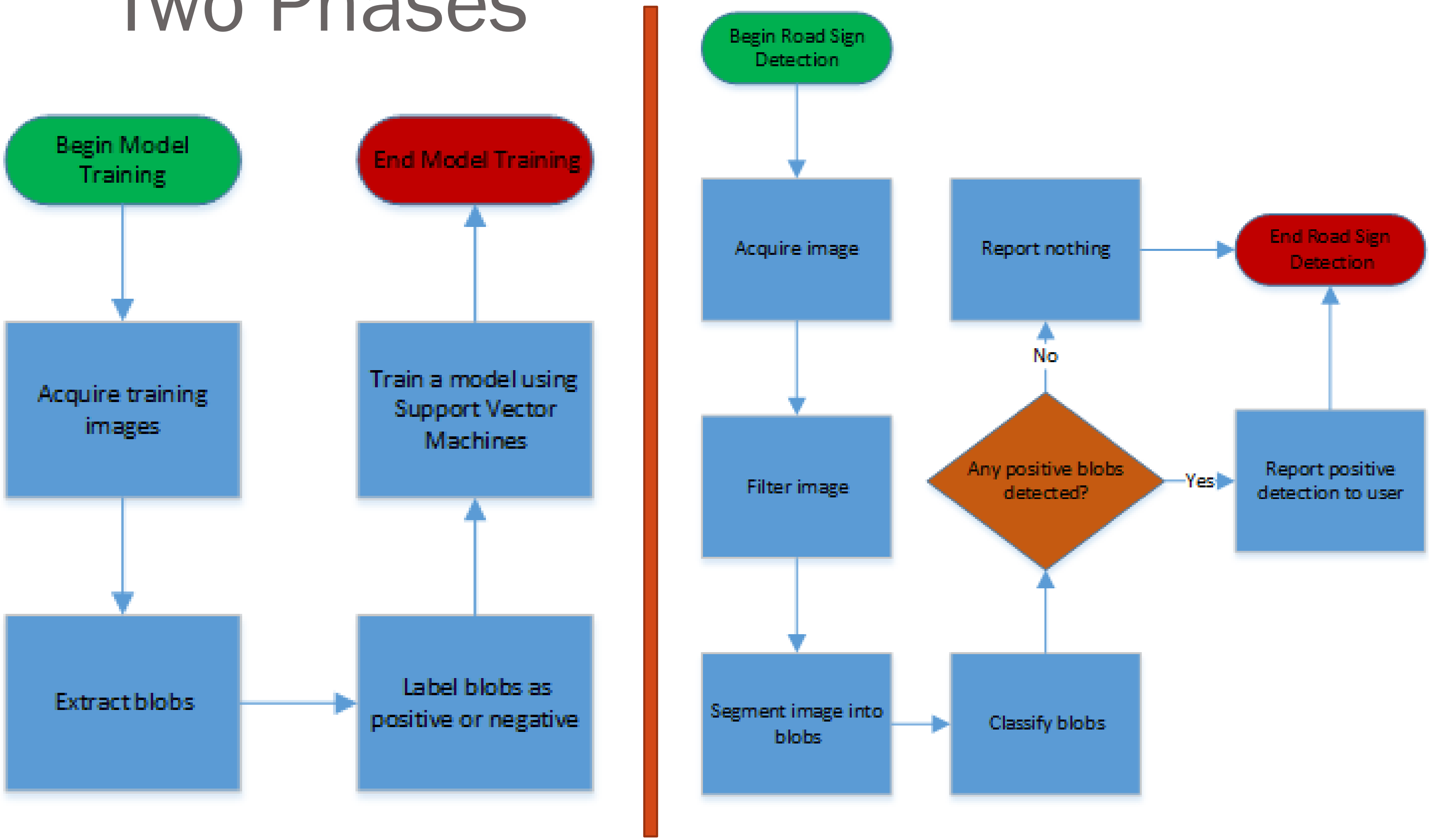
New Paltz
STATE UNIVERSITY OF NEW YORK

Approach

- Input: image and parameters
- Output: Any detections



Two Phases


Begin Model Training
Acquire training images
Extract blobs
Label blobs as positive or negative
Train a model using Support Vector Machines
End Model Training
Begin Road Sign Detection
Acquire image
Filter image
Segment image into blobs
Classify blobs
Any positive blobs detected?
No
Yes
Report nothing
Report positive detection to user
End Road Sign Detection

Phase 1

Extracting possible road signs from images.

RGB to HSV

$$[R', G', B'] = \frac{[R, G, B]}{255}$$

$$\Delta = \max(R', G', B') - \min(R', G', B')$$

$$H = \begin{cases} 60° * \left(\frac{G' - B'}{\Delta} mod 6\right), \max = R' \\ 60° * \left(\frac{B' - R'}{\Delta} + 2\right), \max = G' \\ 60° * \left(\frac{R' - G'}{\Delta} + 4\right), \max = B' \end{cases}$$

$$S = \begin{cases} 0, \Delta = 0 \\ \frac{\Delta}{\max(R', G', B')}, o.w. \end{cases}$$

$$V = \max(R', G', B')$$

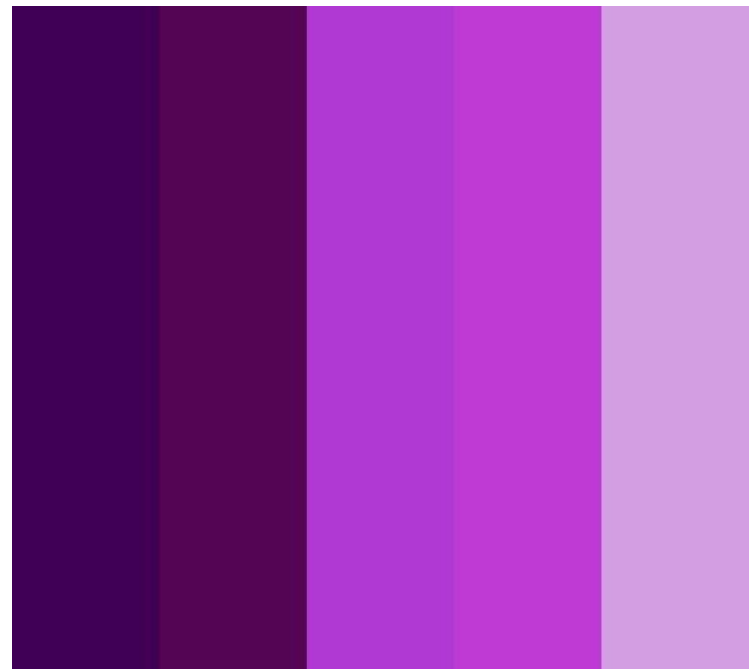


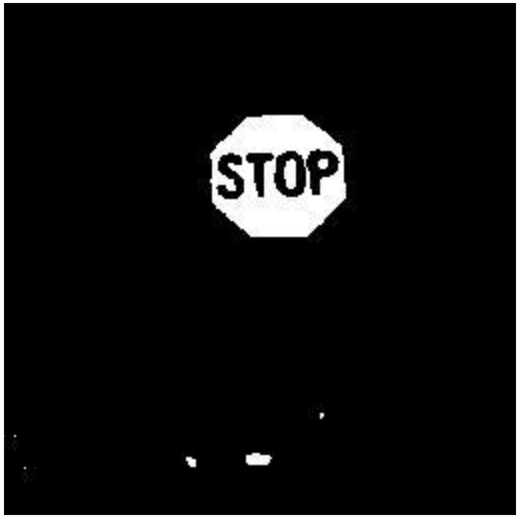
Image Filtering

- Image is filtered by defining ranges for Hue, Saturation, and Value.
- To handle brightness differences, range for Value is 0-100%
- Use Hue and Saturation to select color

New Paltz
STATE UNIVERSITY OF NEW YORK

Image Segmentation

- Filter returns a binary image
- Segmentation separates these pixels into blobs
- Agglomerative segmentation algorithm



Segmentation Algorithm

//Psuedo-code for algorithm (actual code written in MATLAB)

//Input: array of `n` pixels ($\mathbf{p}$ = [x, y]), distance threshold `d`

//Output: array `Class[1…n]` - class label for each pixel

//Initialize: Each pixel is its own class

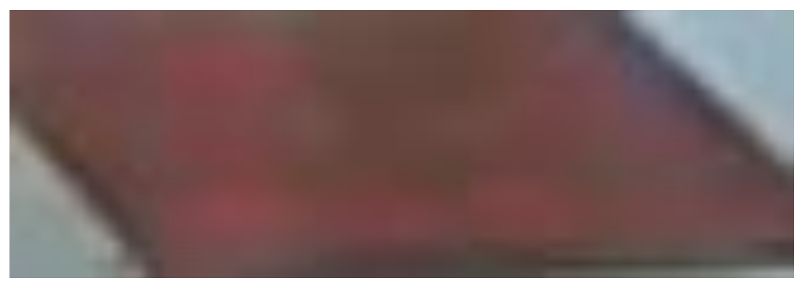
```
Class <= 1:n
For i <= 1 to n
       For j <= i+1 to n
              if( distance(p_i, p_j) < d)
                     Class[j] = Class[i]
```

//Efficiency class of algorithm is $O(n^2)$

Segmentation Visualization

Blob Extraction

- Pixels belonging to the same class form blobs
- Each blob is a candidate for a road sign
- Classify blobs into road signs and not road signs using machine learning algorithm
- Feature Extraction

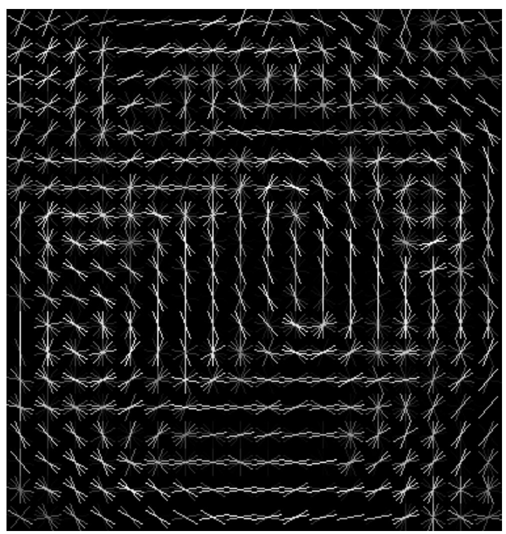
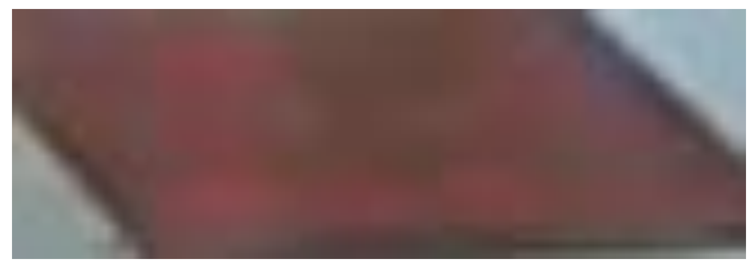
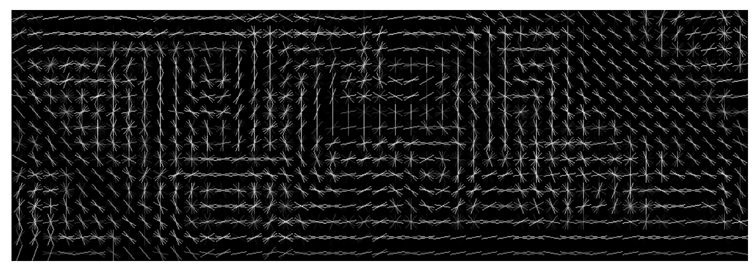


Feature Extraction

- Raw images are of high dimension and contain redundant data.
- Purpose of feature extraction is to extract the most important aspects of an image.

Histogram of Oriented Gradients

- Developed by two researchers, Dalal and Triggs
- Works by breaking an input image into grid of cells
- Computes 31 values for each cell
  - value represent gradients in different directions

Phase 2

Learning a classifiers

New Paltz
STATE UNIVERSITY OF NEW YORK

Training a SVM

- Before classifying blobs, need to train the computer
- Need a set of blobs labeled as road signs and not road signs.
  - Training matrix - $\boldsymbol{X}$
  - Training labels- $\boldsymbol{Y}$

Input to the SVM

$$\boldsymbol{x} = \begin{bmatrix} x_1 & \cdots & x_m \end{bmatrix}$$

$$\boldsymbol{X} = \begin{bmatrix} \boldsymbol{x}_1 \\ \vdots \\ \boldsymbol{x}_n \end{bmatrix}$$

$$\boldsymbol{Y} = \begin{bmatrix} \{1, -1\} \\ \vdots \\ \{1, -1\} \end{bmatrix}$$

Output of the SVM

- Two parameters of the optimal separating hyper-plane.

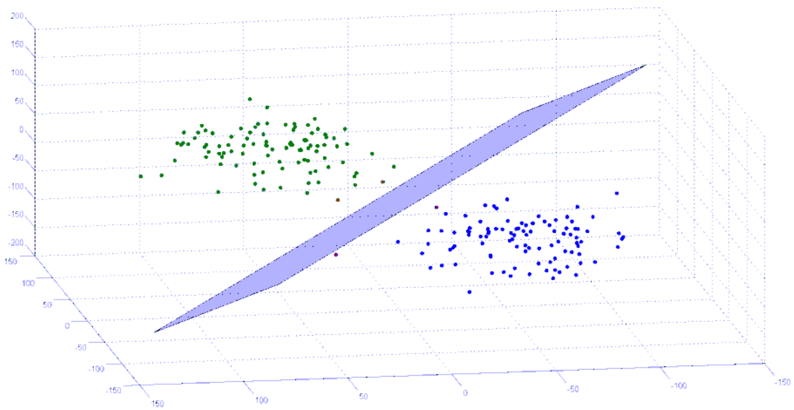
  $\boldsymbol{w}$ – normal vector to the hyperplane

  b – bias value

- Found by solving the following optimization problem:

$$arg\ min_{(w,b)} \frac{1}{2} ||\boldsymbol{w}||^2$$

Subject to: $\boldsymbol{Y}_i(\boldsymbol{w}^T \boldsymbol{X}_i - b) \geq 1, i = 1 \dots n$

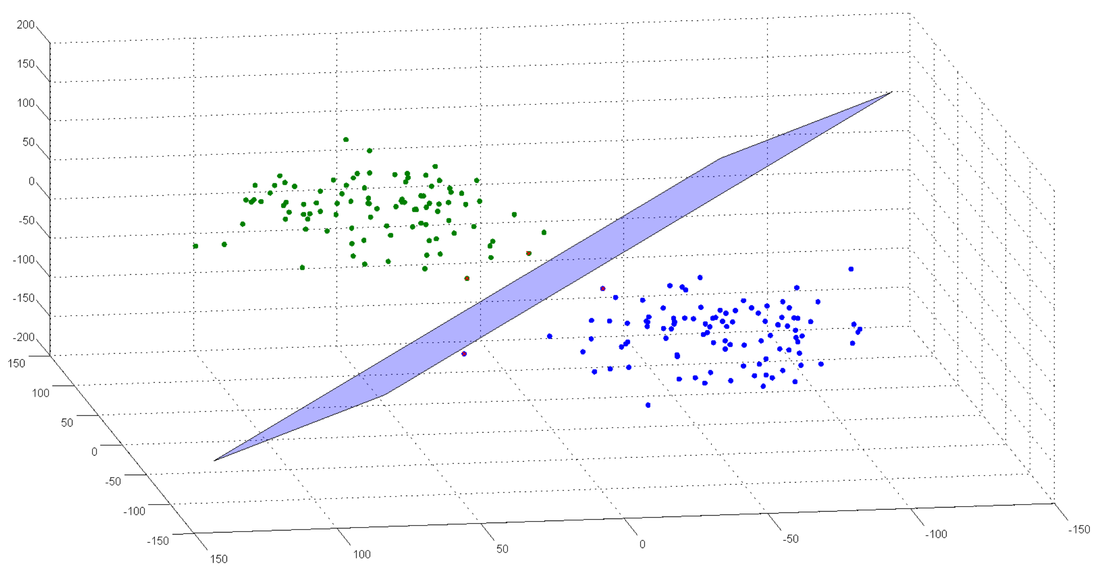


Using the Classifier

- To classify a new blob $\boldsymbol{x}$, the following function is used:

$$g = \boldsymbol{w}^T \boldsymbol{x} + b$$

- If $g > 0$, $\boldsymbol{x}$ is a road sign, otherwise it is not



New Paltz
STATE UNIVERSITY OF NEW YORK

Evaluating the Classifier

- A major problem with machine learning algorithms is the inability to generalize well.
- A way to measure this is Hold-One-Out Cross Validation (HOOCV)

New Paltz
STATE UNIVERSITY OF NEW YORK

HOOCV

- Uses the training data to validate the model.
- Holds out one training vector and trains a temporary classifier on the rest.
- Tests temp. classifier on the held out vector
  - Success if output == true label
- Repeat for every training vector
- Returns the percentage of success.

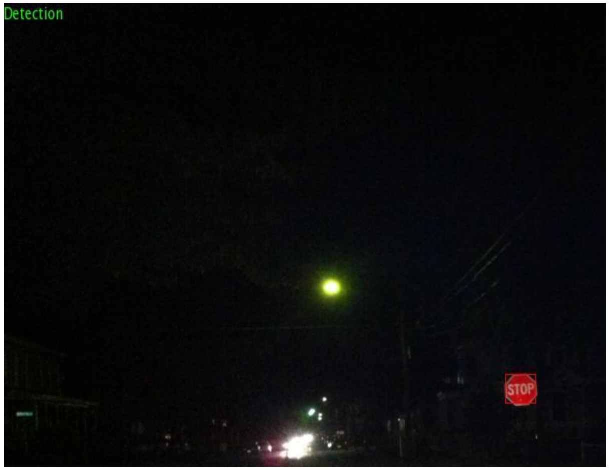
Results

- Trained models for stop sign, speed-limit sign, and intersection-ahead sign.
- HOOCV on each model was >85% success.
- Speed
  - MATLAB implementation
    - average of .35 s per frame
    - About 2.9 FPS

Results


Detection
STOP
ONE WAY


Detection

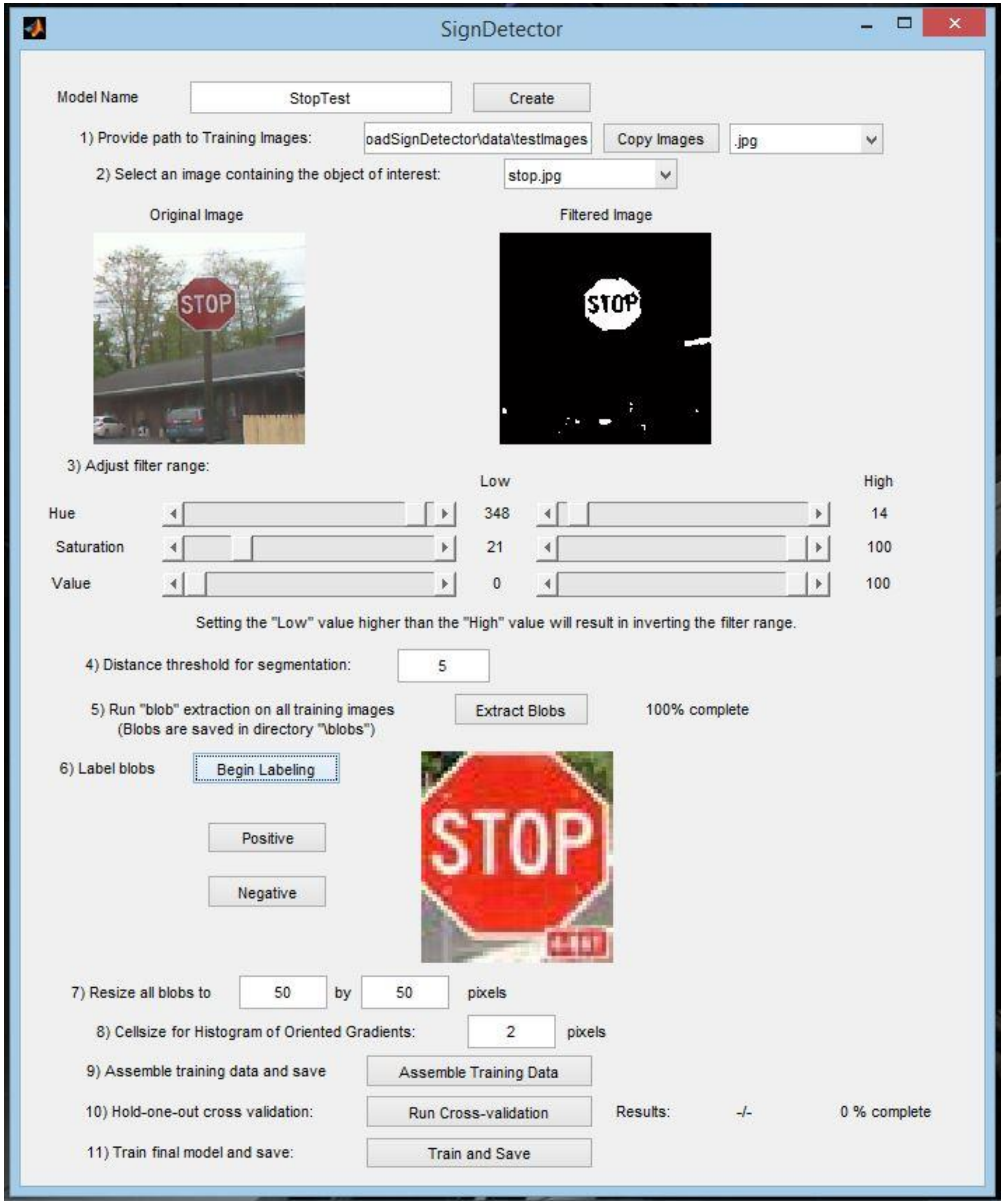

Detection
STOP


Detection

Stop Sign Results

MATLAB GUI

- Graphical User Interface
  - Streamlines the process of training a model
- User inputs a set of training images, selects parameters, and does labeling.
- Software trains a model and tests it using Hold-One-Out Cross Validation (HOOCV)


SignDetector
Model Name
StopTest
Create
1) Provide path to Training Images:
oadSignDetector\data\testImages
Copy Images
.jpg
2) Select an image containing the object of interest:
stop.jpg
Original Image
Filtered Image
STOP
3) Adjust filter range:
Low
High
Hue
348
14
Saturation
21
100
Value
0
100
Setting the "Low" value higher than the "High" value will result in inverting the filter range.
4) Distance threshold for segmentation:
5
5) Run "blob" extraction on all training images
(Blobs are saved in directory "\blobs")
Extract Blobs
100% complete
6) Label blobs
Begin Labeling
Positive
Negative
STOP
7) Resize all blobs to
50
by
50
pixels
8) Cellsize for Histogram of Oriented Gradients:
2
pixels
9) Assemble training data and save
Assemble Training Data
10) Hold-one-out cross validation:
Run Cross-validation
Results:
-/-
0 % complete
11) Train final model and save:
Train and Save

Engineering Constraints

- Economic
  - A reduced number of accidents on the road will have an effect on insurance companies
- Manufacturability
  - Embedded implementation can be run with a raspberry pi ($35) and a webcam (~$20)
- Social & Political
  - New laws dealing with these type of devices, possibly requiring it in vehicles
- Sustainability
  - This technology will not become obsolete because of the extensive road sign infrastructure in place today.
- Health & Safety
  - Goal of the project is to increase safety on the road

Summary

- A framework for detecting road signs in video streams
- Image processing is used to extract road signs from images.
- Uses machine learning to teach the system how various road signs appear.

Future Work

- Implementation in an embedded setting
  - OpenCV C++ libraries
- Improving image filtering methods to handle white signs
- Testing different feature extraction methods
  - Canny Edge Detection

Timeline

January February March April	May	June	July August	September	October November	December
Researched possible topics; chose road sign detector.	Senior design I poster board presentation.	Began research on approach and development	Began design and development. Reported preliminary results to advisor	Continued into second phase of development (Machine Learning algorithms implemented)	Finished MATLAB GUI, began developing OpenCV module.	Senior Design 2 Presentation Write-up.

Questions

References

- [1] Chapelle, O.; Haffner, P.; Vapnik, V.N., "Support vector machines for histogram-based image classification," *Neural Networks, IEEE Transactions on* , vol.10, no.5, pp.1055,1064, Sep 1999
doi: 10.1109/72.788646
URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=788646&isnumber=17091
- [2] Dalal, N.; Triggs, B., "Histograms of oriented gradients for human detection," *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on* , vol.1, no., pp.886,893 vol. 1, 25-25 June 2005
doi: 10.1109/CVPR.2005.177
URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=1467360&isnumber=31472