VISUAL PATTERN RECOGNITION IN ROBOTICS

Dr. Ashwin Patani\textsuperscript{1}, Prof. Miloni Ganatra\textsuperscript{2}

\textsuperscript{1,2}Asst. Professor, Department of Electrical & Electronics Engineering, Indus University, Rancharda, Ahmedabad, Gujarat, India

ABSTRACT

Visual Pattern Recognition in Robotics: Real-time pattern recognition algorithm to detect & recognize the sign-board consists of 3 steps:

Color-based filtering, locating sign(s) in an image and detection of pattern of the sign. Very good results are obtained for the 2nd & 3rd steps and the first one is having tough challenges.

The standard technique for detecting and recognizing road signs consists of three steps (Lalonde and Li, 1995). First, color segmentation or color thresholding is applied to emphasize possible signs in an image. Thus, restricting the search area in the image. Second, template matching is applied for shape detection. Third, specific signs are detected using template matching or neural networks.

In this part, a specialized robust color filter is applied on the image to mark the red circle surrounding the sign board pattern. The algorithm has to use a broad definition of red and white colors since various intensities and saturations can occur in real images (depending on weather, day/nightlight, etc.). On the other hand, only red color on traffic signs should be detected to minimize the number of starting points for the pattern recognition algorithm. Thus, detection of red color outside a traffic sign (like on buildings, cars, etc.) should be minimized.

Detection and recognition are two major steps for determining types of traffic signs. Detection refers to the task of locating the traffic signs in given images. Where recognition requires some knowledge base, or specific criteria to understand what pattern is showing. Based on that pattern, control signal will be sent to robot and robot will move in that specified direction.

Keywords: Segmentation, Shape Detection, Neural Networks.
1. INTRODUCTION

The term Visual Pattern Recognition (ViPR) refers to the task of categorizing some object to a correct class based on the visual measurements about the object. Computer vision has been applied to a wide variety of applications for intelligent transportation systems such as in traffic monitoring, traffic-related parameter estimation, driver monitoring, and intelligent vehicles, etc. Traffic sign recognition (TSR) is an important basic function of intelligent vehicles.

Detection and recognition are two major steps for determining types of traffic Signs. Detection refers to the task of locating the traffic signs in given images. It is common to call the region in a given image that potentially contains the image of a traffic sign—the region of interests (ROI). Taking advantages of the special characteristics of traffic signs, ViPR systems typically rely on the color and geometric information in the images to detect the ROI. After identifying the ROIs, we extract features of the ROIs, and classify the ROIs using the extracted feature values.

Identifying what pixels of the images are red is a special instance of the color segmentation problems. This task is not easy because images captured by cameras are affected by a variety of factors, and the “red” pixels as perceived by human may not be encoded by the same pixel values all the time.

Fig.1: Block Diagram of ViPR

Our ViPR system consists of a USB camera, which captures image, sends it to processing unit (laptop or a PC) as shown in the block diagram. Image is acquired by MATLAB’s video adapters (winvideo in our case), Image is filtered & processed to Locate Sign Board first & then recognize the pattern inside it. This computation is done within seconds in an extremely efficient manner.

Finally the command to control the motion of the machine is sent via PARALLEL PORT from computer. This data is directly fed to the driver circuit which drives the Motors of robot. ROBO recognizing patterns: life galvanized from “dead parts”……!!!

2. TASKS NEED TO BE FOLLOWED

To fulfill the aim, robot is required to complete the following tasks:

- Robot has to capture the images continuously.
- Image enhancement.
- Simultaneously it is required to process the image to find out whether the signboard is present in the captured image or not.
- If signboard is present, it is required to identify & store only that portion of the image which is lying inside the Sign Board.
- Once the portion within sign board is acquired, recognition of the pattern is required.
- After recognizing pattern properly, robot is required to perform the task which is inferred by that pattern.
In short Sign boards will be present on the path; Robo will capture video & search for the sign board, it will read the pattern move into the direction inferred by that pattern.

3. EXPERIMENTAL WORKS

3.1 ViPR system consists of

Main sections / subparts of our project are:
- Image acquisition setup: It consists of a video camera, web camera, or an analogue camera with suitable interface for connecting it to processor.
- Processor: It consists of either a personal computer or a dedicated image processing unit.
- Image analysis: Certain tools are used to analyze the content in the image captured and derive conclusions e.g. locating position of an object.
- Machine control: After making the conclusion, mechanical action is to be taken e.g. serial or parallel port of a PC can be used to control left and right motors of a robot to direct it towards Left, Right, Forward or Backward.

3.1.1 IMAGE Acquisition setup

Image acquisition basically means capturing an image. Image capturing can be done using video camera available in various resolutions e.g. 640 x 480 pixels.

There are two types of cameras generally available: Digital cameras (CCD – charge coupled device and CMOS sensor based) and analogue cameras. Digital cameras generally have a direct interface with computer (USB port), but analogue cameras require suitable grabbing card or TV tuner card for interfacing with PC.

Power requirements: CCD cameras give high quality, low-noise images. It generates an analog signal and uses analog to digital converter (ADC) and thus consumes high power.

CMOS cameras have lesser sensitivity resulting in poor image quality but consume lesser power resulting in more battery life.

3.1.2 Image analysis

It consists of extracting useful information from the captured images. We first decide the characteristics of the object to look for, in the image. This characteristic of the object must be as robust as possible.

Generally, for the purpose of tracking or identifying the object we utilize:

- Color
- Intensity
- Texture or pattern
- Edges – circular, straight, vertical stripes
- Structure – Arrangement of objects in a specific manner

Quantitative/Statistical analysis of image:

- Center of gravity - point where the desired pixels can be balanced
- Pixel count – a high pixel count indicates presence of object
- Blob – an area of connected pixels

3.1.3 Machine control

Machine Control consists of controlling a robot based on the conclusion derived from image analysis and to achieve this using PC, its parallel port or serial port can be used for driving the robot.
For e.g. H-bridge, PWM control can be extensively used for right and left motor movement of a robot. Serial-port can be used for transferring data, which necessitates a micro-controller on the robot to interpret the data.

### 3.2 Support provided by Image Acquisition Toolbox in MATLAB

Now-a-days most of the cameras are available with USB interface. Once the drivers are installed for the camera, the computer detects the device whenever connected. Alternatively, if a Digital Video camcorders or CCD camera is connected with grabber card and interfaced with computer, Windows OS automatically detects the device.

In order to execute out further instructions, camera is needed to be connected to the PC. In MATLAB, we can check if the support is available for your camera. MATLAB has built-in adaptors for accessing these devices. An adaptor is software that MATLAB uses to communicate with an image acquisition device.

```matlab
>> imaqhwinfo
>> Cam=imaqhwinfo;
>> Cam.InstalledAdaptors

To get more information about the device,

```matlab
>> dev_info = imaqhwinfo('winvideo', 1)
```

#### 3.2.1 Previewing video

To preview the video captured by the camera is done by defining an object and associate it with the device.

```matlab
>> vid=videoinput('winvideo',1,'RGB555_640x480')
```
The details of the acquisition parameters are as shown in the following snapshot. Now to see the video being captured by the camera, use preview command:

```
>>preview(vid)
```

Window pop-up displays what the camera is capturing as shown below,

![Video Preview](image)

The camera may support multiple video formats. To see all the supported formats,

```
>>dev_info = imaqhwinfo('winvideo',1);
>>celldisp(dev_info.SupportedFormats); %displays list of supported formats
```

### 3.2.2 Capturing an image
Now to capture an image from the video, we define the object `vid` and use `get data` to capture a frame from the video.

```
>>start(vid); % This command initiates capturing of frames and stores, the frames in memory
>>im=getdata(vid,1);
>>figure,imshow(im);
```

### 3.2.3 Storing the image
You can store the captured image as a .jpg file using `imwrite` function.

```
>>imwrite(im,'testimage.jpg');
```
3.4 Interfacing PC ports
MATLAB provides support to access serial port (also called as COM port) and parallel port (also called as printer port or LPT port) of a PC. Parallel port has 25 pins. Parallel port cables are locally available (commonly referred as printer port cables).

- **Pins 2-9 are bi-directional data pins (pin 9 gives the most significant bit (MSB))**
- **Pins 10-13 and 15 are output pins (status pins)**
- **Pins 1,14,16,17 are input pins (control pins), while pins 18-25 are Ground pins**

MATLAB has an adaptor to access the parallel port (similar to adaptor for image acquisition). To access the parallel port in MATLAB, define an object

```
>>parport = digitalio('parallel','LPT1');
```

Parallel Port address is obtained using,

```
>>get(parport,'PortAddress')
```

To put the output data to the parallel port into a matrix, put the output data to the parallel port into a matrix,

```
>>dataout = logical([1 0 1 0 1 0 1 1]);
```

To output these values, use the putvalue function,

```
>>putvalue(parport,dataout); Or the decimal equivalent of the binary data can also be used as below.
>>data = 23;
>>putvalue(parport,data);
```

The pins of the parallel port is connected to the driver IC for the left and right motors of the robot to control the left, right, forward and backward motion of the vehicle. We have used Quad H-Bridge driver IC L293D for driving the motor in both clockwise and anti-clockwise directions.

3.5 ViPR DESIGN
Visual Pattern Recognizing Robo’s design is divided into 6 main phases:

<table>
<thead>
<tr>
<th></th>
<th>Image Acquisition</th>
<th>USB Camera</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Image Acquisition</td>
<td>USB Camera</td>
</tr>
<tr>
<td>B</td>
<td>Image Transmission on Forward channel</td>
<td>Camera itself</td>
</tr>
<tr>
<td>C</td>
<td>Image Reception AT PROCESSOR</td>
<td>USB cable</td>
</tr>
<tr>
<td>D</td>
<td>Image Enhancement &amp; Processing &amp; Pattern Recognition</td>
<td>Matlab code</td>
</tr>
<tr>
<td>E</td>
<td>Transmission on Reverse channel</td>
<td>Parallel port cable</td>
</tr>
<tr>
<td>F</td>
<td>Motor Driving</td>
<td>H-Bridge driver circuit</td>
</tr>
</tbody>
</table>

3.5.1 Image Acquisition
Hardware Used: USB Camera
Supported Formats: YCbCr -640X480, 352x576, 720x576
We have used USB camera for image acquisition. Image is fed in YCbCr (Luminance(Y), Chrominance – Cb for blue & Cr for red) format.
We have mounted this camera on robot. So that it can capture the Sign-board image properly.

3.5.2 Image Transmission on Forward channel
Image transmission mode: USB cable
Transmitted Image Format : YCbCr
Image captured by camera is transmitted by itself using its USB cable in YCbCr format.
3.5.3 Image Reception AT PROCESSOR

In our case capturing device is USB came, which can be accessed as easily as any capturing device only because the powerful support provided by MATLAB adaptors.

3.5.4 Image Enhancement & Processing & Pattern Recognition

1. Image Enhancement

Assuming that initialization has been done as shown in Experimental work, we will get image at any instant by following code,

```matlab
>>Image = peekdata(video_object);
```

We are getting Image in YCbCr format which is shown below,

We need to convert this image into RGB format for further processing using following command,

```matlab
>>RGB = ycbcr2rgb (image);
```

RGB image:

Image enhancement is necessary as we get somewhat noisy image with sharp transitions as shown below,

Zoomed view of the sign board from above original RGB:
So as to remove the noise (sharp transitions) we have used Smoothing filter.

Applied mask is shown below,

\[
\text{Applied Mask} = \begin{bmatrix}
\frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\
\frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\
\frac{1}{9} & \frac{1}{9} & \frac{1}{9}
\end{bmatrix}
\]

\[
\text{image}=\text{imfilter}(\text{image}, \text{ones(3,3)/9});
\]

Filtered Image is shown below.

Zoomed sign board from previous image

We can clearly see that sharp transitions are reduced by using smoothing filter.

3.6 IMAGE PROCESSING

3.6.1 Detecting Red Color of Sign Board Border

Identifying what pixels of the images are red is a difficult in color segmentation problems. This task is not easy because images captured by cameras are affected by a variety of factors, and the “red” pixels as perceived by human may not be encoded by the same pixel values all the time.
By performing many experiments & trials we have found the following criterion that is most suitable to detect RED color as perceived by human.

RED should be > 100, (R-G) should be >30, (R-B) should be >30

We are replacing every pixel which lies in above mentioned range with, R = 0; G = 0; B = 0;

All other pixels which are outside this range are replaced with,

R = 255; G = 255; B = 255;

Thus all pixels of image will become white & only those pixels which are RED will have black color. Thick border of sign board & some background pixels which are RED will become black as shown in figure below, this is whole task is done in another variable ‘imtemp’.

As we can see RED color is detected somewhere else also.

3.6.2 Avoiding detection of RED color in objects having no clear borders

We are using matlab function ‘imclearborder’ using 8 connected region algorithm to avoid detection of Red color pixels which are not connected & forming any boundary,

\[ BW = \text{imclearborder(imtemp,8);} \]

We get following image,

Black color is flooded, white objects forms clear boundary and are connected will have preserved area inside them.

3.6.3 Taking negative of above image to obtain Board Mask & adding with grayscale image of original RGB image.

\[ BW = -BW; \]
Grayscale image of original RGB image:

>> Gray = rgb2gray (RGB);

Adding both of above images to extract our ROI (Region of Interest)

>> Final_image = BW + Gray;

We get following image by adding BW & Gray,

3.6.4 Binarization of ROI for pattern recognition

Here our region of interest is the sign. We get binary image of ROI by,

>> image = im2bw(final_image,0.35); % reuse of variable ‘image’

0.35 is the threshold value that we obtain by experiments for proper binarization. We get following image,
Extraction of Pattern from Environment is accomplished; now this Image is passed to PATTERN RECOGNITION function that we have designed.

3.6.5 Pattern Recognition

Pattern recognition is "the act of taking in raw data and taking an action based on the category of the pattern". Most research in pattern recognition is about methods for supervised learning and unsupervised learning.

Pattern recognition aims to classify data (patterns) based either on a priori knowledge or on statistical information extracted from the patterns. This is in contrast to pattern matching, where the pattern is rigidly specified.

Whole Image is scanned 2 times,
1st Vertical scanning will be done which will store LEFT or RIGHT arrow co-ordinates in Array - 'distanceY'.
LEFT & RIGHT is having most probability in robo motion so we are doing vertical scanning first.
2nd Horizontal scanning is done which will store UP or DOWN arrow Co-ordinates in Array - 'distanceX'.

3.6.6 Logic to differentiate ARROWS

Logic is application specific; any transforms are not used to avoid the delay produced by them.
Image is scanned vertically from left to right,
When any black pixel is found, 'flagBLACK' will be set the Y coordinate where black pixel found is stored in 'x1'
We call this 1st BLACK PIXEL
flagWHITE' will only be set when 'flagBLACK' is set and WHITE pixel is found.
Now 'flagCOMPLETE' will be set only when black pixel is found & flagWHITE' is set as flagCOMPLETE' is set Y coordinate is stored in 'x2'
We call this 2nd BLACK PIXEL.
Then after x2-x1 difference is stored in 'distance Y' Array. We will get as many distance Y values as many time - 'BLACK-WHITE-BLACK' pixels are found.
Lastly this value are observed, if left to right this values (distance between 1st BLACK & 2nd BLACK) are incrementing the arrow is LEFT arrow & if they are decrementing it is RIGHT arrow
Same way in horizontal scanning, if distance between 2 BLACK pixels are incrementing from top to bottom then arrow is UP arrow & if they decrementing arrow is down arrow
3.6.7 Displaying the result in figure

4. FLOW CHART

![Flow Chart Image]

The flow chart outlines the process from start to end, detailing each step with decision points and actions. This visual representation is crucial for understanding the flow and decision-making process within the described system.
5. RESULT

After these much soft & hard work what we have obtained is: We have developed successfully working Matlab code for, Image Acquisition, Image Enhancement, ROI detection, Pattern Recognition.
H-bridge driver circuit to provide necessary current to DC motors.
We have constructed robot structure having 4 high torque DC geared motors of 60rpm.
Image Acquisition Code is working nicely Pattern recognition algorithm can recognize fuzzy images even hence it is much accurate.
Board detection algorithm needs some improvement for Board detection without any failure.

6. APPLICATIONS & FUTURE SCOPE

After having achieved promising results for detecting sign boards, we can modify our ViPR system for the various applications like Pattern Recognition for Industrial Purpose.
Vision based Intelligence in Automated Robotic System. Pattern Analyzing & Decision making This is very much in need in the case of bottling plants, IC manufacturing units where bulk production is carried out and manual monitoring is not possible Developing Perception, Communication and Visual Image Understanding in machine. Developing Interactive communication medium between Human and a machine. Vision based Car Navigation. This technique can greatly reduce the chances of vehicle accidents by indicating warning sign boards that come across any freeway, highway and alerting it to other direction or stopping it.

7. CONCLUSION

After satisfactorily completed, we have successfully demonstrated our claim of implementing pattern recognition in robotics. Robot is totally autonomous & takes decision based on visual information. It categorizes sign board pattern to a correct class based on the visual measurements about the pattern. It involves low end cam it is a very economic solution and with further modifications it can be implemented in bottling plant & any machinery parts manufacturing industry and has great potential for implementing it in vision based car navigation. It is completely suited for on field applications since image acquisition takes fractions of a second.

REFERENCES


