Automatic Nepali Number Plate Recognition with Support Vector Machines

Ashok Kumar Pant
Institute of Science and Technology
Tribhuvan University
Kathmandu, Nepal
Email: ashokpant87@gmail.com

Prashnna Kumar Gyawali
Institute of Engineering
Tribhuvan University
Kathmandu, Nepal
Email: gyawali.prasanna@gmail.com

Shailesh Acharya
Institute of Engineering
Tribhuvan University
Kathmandu, Nepal
Email: sailes437@gmail.com

***Abstract*—Automatic number plate recognition is the task of extracting vehicle registration plates and labeling it for its underlying identity number. It uses optical character recognition on images to read symbols present on the number plates. Generally, numberplate recognition system includes plate localization, segmentation, character extraction and labeling. This research paper describes machine learning based automated Nepali number plate recognition model. Various image processing algorithms are implemented to detect number plate and to extract individual characters from it. Recognition system then uses Support Vector Machine (SVM) based learning and prediction on calculated Histograms of Oriented Gradients (HOG) features from each character. The system is evaluated on self-created Nepali number plate dataset. Evaluation accuracy of number plate character dataset is obtained as; 6.79% of average system error rate, 87.59% of average precision, 98.66% of average recall and 92.79% of average f-score. The accuracy of the complete number plate labeling experiment is obtained as 75.0%. Accuracy of the automatic number plate recognition is greatly influenced by the segmentation accuracy of the individual characters along with the size, resolution, pose, and illumination of the given image.**

Keywords*—Nepali License Plate Recognition, Number Plate Detection, Feature Extraction, Histograms of Oriented Gradients, Optical Character Recognition, Support Vector Machines, Computer Vision, Machine Learning

I. Introduction

Automatic Number Plate Recognition (ANPR) systems are important systems in transportation management and surveillance. They are capable of identifying vehicles by extracting the number plate and reading the plate identity which is unique identification code given to each vehicle. ANPR systems can be used for automatic traffic control, electronic toll collection, vehicle tracking and monitoring, border crossing, security and many more.

Developing ANPR system requires integration of computer vision algorithms with imaging hardware. Computer vision algorithms include image processing techniques for number plate localization, plate orientating and sizing, normalization and character segmentation. Beside these, it includes pattern recognition techniques for optical character recognition. For better identification accuracy, machine learning techniques are used to learn from input data. There are many difficulties that ANPR system may face; such as, poor resolution, poor illumination conditions, blurry inputs, plate occlusion, different font size and variety of plate structures.

In this research paper, we proposed a machine learning based Nepali number plate recognition system, which is capable of automatically labelling a given number plate to its identity. Automatic number plate recognition is a widely researched problem from many decades and in many countries it is successfully applied to practical domain too. But for Nepali number plates, there are very few researches conducted so far. Most of them are based on simple distance measures for character matching. Plate localization and segmentation are again not researched much for handling all the situations. Nepali number plate character are selected from the pool of 29 characters (Fig. 1) in a specific orders. Order defines various characteristic of the number plates (Fig. 2) such as vehicle type, vehicle load, etc. The number plates used in Nepal are usually of two formats, one containing all the characters in a single row and the other containing two rows of characters. Characters are selected from Devnagari script. Here, we propose a complete number plate recognition pipeline that automatically localizes, normalizes and segments number plates from vehicle images; segments characters from detected number plates and passes them to classification system for labeling. Classification system implements SVM based machine learning algorithms for learning and prediction.

Recent interests of ANPR systems include sophisticated machine learning techniques (like deep learning, neural networks, SVMs) along with good plate localization and character segmentation algorithms. Localization of license plate refers to extracting the region in an image that contains the plate and some of the widely used techniques for localization include scale shape analysis, edge detection, mathematical morphology [1], connected component analysis [2], regional segmentation [3], and statistical classification [4]. Different algorithms have claimed their accuracy for localization from 80% to 96%. The segmentation phase extracts the region of individual characters from the plate. Frequently used algorithms for segmentation include region merging and splitting, edge gradient analysis and region analysis. Coordinate of window enclosing each character is ascertained by segmentation. Template matching and statistical classification were widely used for number plate character recognition in the past. But with the advent of technology and machine learning algorithms, Artificial Neural Networks, Support Vector Machines, Hidden Markov Models are some of the widely used techniques in the current scenario. These algorithms claim to offer accuracy of up to 98% for tasks like character recognition even under different environmental variations. [5] presented quite a good results in different

Fig. 2: License plate identifiers for the Nepali vehicles.

inclination and exposure conditions. The shape and characters placement in the number plates of the vehicle are exclusively distributed around the globe and moreover, Nepalese plate use Devnagari characters which make the problem even more complex. At such the recognition methods and algorithm for the Nepalese plate should be dealt uniquely.

II. System Overview

The system architecture that we used for the overall license plate recognition is shown in Fig. 3. The overall system is divided into four subsystems; preprocessing, feature extraction, number plate character training and prediction. The first part deals with preprocessing of the image to fine tune the important details in the image along with plate localization and character segmentation. Plate localization is carried out to identify the position of license plate in the image frame. Character segmentation extracts the windows in the plate containing each character. After preprocessing stage, extracted number plate characters are labeled for Optical Character Recognition (OCR) and feed into feature extraction subsystem. Feature extraction engine extracts useful characteristics from given character image. Which are then passed into OCR engine for training. Training subsystem creates trained models for each class and saves them for future prediction. Prediction engine takes extracted features and saved classifiers to predict the character present in the given image.

III. Image Preprocessing

The first step of any image processing system is to pre-process the acquired image. Pre-processing includes some of the operations on the image to enhance the region of interest and thus such operations completely rely upon the context. In our case, the region of interest is the license plate of the vehicle and thus we implement following operations that change the image into a form suitable for further processing and also increase the efficiency of the algorithm used for localization, segmentation and feature extraction tasks.

A. HSV Color Space Conversion

Hue Saturation and Value (HSV) color space model describes colors in terms of the Hue, Saturation, and Value. HSV color model is often preferred over the RGB model for the situations when color description plays an integral role. The HSV model describes colors similar to the human eye perceive the colors [6]. RGB defines color in terms of a combination of primary colors, whereas, HSV describes color using more familiar comparisons such as color, vibrancy and brightness. The coordinate system is cylindrical, and the colors are defined inside a hexcone [7]. The hue value H runs from 0 to 360°. The saturation S is the degree of strength or purity and is from 0 to 1. Value V corresponds to the brightness which also ranges from 0 to 1.

B. Color Masking

The image in HSV mode is now masked for the appropriate color. Nepalese license plate have different colored plates depending upon the ownership of the vehicles. For examples, private vehicles have red-colored plates with white characters in it and public vehicles have black plates with white characters. Hence depending upon the color of number plates of vehicle, masking is performed [8]. In this paper, we are primarily focused on recognition of license-plate numbers of private vehicles, thus, we masked the red color from the image. In HSV model, the Hue value from around $0° - 10°$ and $350° - 360°$ can be approximated as the red color. Using these range, we masked the red color regions from the images.

C. License Plate Localization

The localization of the region of license plate from the image of the vehicle has always been one of the challenging tasks because of its unaccountable variations in the shape, size, color, texture as well as spatial orientations. For the localization of the license plate, we initially extract different contours within the image. Contours are the region inside the curve joining all the continuous points (along the boundary), usually having the same color and intensity. Before finding the contours, we threshold the image to convert into binary form. The purpose of converting to the binary form is for the accuracy of finding contours. Then the edges within the image are detected using sobel edge detector [9]. The detected edges are grouped to form the contours with the edges behaving as implicit boundaries between positive and negative regions.

The set of tangents (detected edges) $T = t_1, t_2, \ldots, t_N$ are grouped together to produce contours that are guaranteed to be closed and simple (without self-intersection) [10]. However, it is assumed that only a subset $T_o < T$ of tangents in the image lie on the boundaries of objects of interest.

The resulting contours within the image will be of different shapes (mostly irregular). We approximate those contours by a rectangle. For drawing rectangle around the contours, among the different approximations methods of the contour formation, we opted for a chain approximation which compresses horizontal, vertical and diagonal segments and leaves only their end points [11]. From these end points a rectangle is constructed by using all the position of extreme pixels as one of the points in the side of the rectangle.

Within the image of the vehicle with license plate, it is most likely to get large numbers of rectangular contours. Hence among the different candidates, to select the rectangle with license plate, we perform following two tests.

1) Aspect ratio test.
2) Profile test.

Aspect ratio is calculated by using,

$$Aspect\ Ratio = \frac{Width}{Height} \tag{1}$$

Numbers	Zonal Representation		Vehicles' Category Color (Foreground)	Heavy size	Middle size	Light size
०	मे	Mechi	Private	क	च	प
१	को	Koshi	Government	ग	झ	ब
२	स	Sagarmatha	Public	ख	ज	थ
३	ज	Janakpur	Diplomatic	सि डी	सि डी	सि डी
४	ना	Narayani	Tourist	य	य	Not available
५	बा	Bagmati	Public/National corporation	घ	ञ	Not available
६	ग	Gandaki	Note: The box in the shaded region are the categories that are used in this research!!			
७	लु	Lumbini				
८	ध	Dhaulagiri				
९	भे	Bheri				
	रा	Rapti				
	क	Karnali				
	से	Seti				
	मा	Mahakali				

Fig. 1: Nepali license plate specifications.

Fig. 4: Row profile in the left, localized image in the center and column profile of two blocks of characters in the right.

Nepali vehicles have license numbers encoded in the both rear and front side with two different sized rectangular plates. The front sized plates are usually in $4:1$ ratio and the back sized plates are in $4:3$ ratio.

The rectangles with desired aspect ratio will only qualify for the further processing. The aspect ratio test won't guarantee to output that one specific rectangle which has got the license plate but do make the job easier for further processes by eliminating the large numbers of the candidate rectangles.

Profile tests are performed to accurately identify a license plate. Profile tests are carried out on candidate image regions that are filtered from aspect ratio test. For a profile test, we normally convert image regions into binary form and then calculate both row and column profiles. The row profile of the license plate rectangle will either have one or two peaks above the preset threshold corresponding to the $4:1$ or $4:3$ ratio plates respectively.

Similarly, in the column profile there will be minimum of four and maximum of eight peaks above threshold for the $4:1$ ratio plate and for the $4:3$ ratio plate, which is previously detected from the row profile, we will first divide it into two halves and then the resulting column profile will have, for the upper part, minimum of three and maximum of four peaks and for the lower part the minimum one and maximum four peaks above threshold and all corresponding to the characters associated with the license plate. The rectangle with the best match is selected as the license plate for the segmentation process.

D. Character Segmentation

Character Segmentation algorithm works on the detected number plate region and extracts the exact rectangular boundary enclosing each individual character. We extract the character from the localized plate by analyzing their projections but before, skew correction are made.

1) De-skewing: As the images of the vehicles are taken when the vehicles are moving, it is very likely that there will be skew in the image. The skew also results in the image due to difference in the camera position and the license plate of the vehicle, bumpy roads and the resulting vibrations of the vehicle [12]. The method is usually applied for aligning the scanned documents so that OCRs accuracy and performance can be enhanced. We are trying to use the same concept for enhancing the character segmentation process.

De-skewing is a process whereby skew is removed by rotating an image by the same amount as it is skewed but

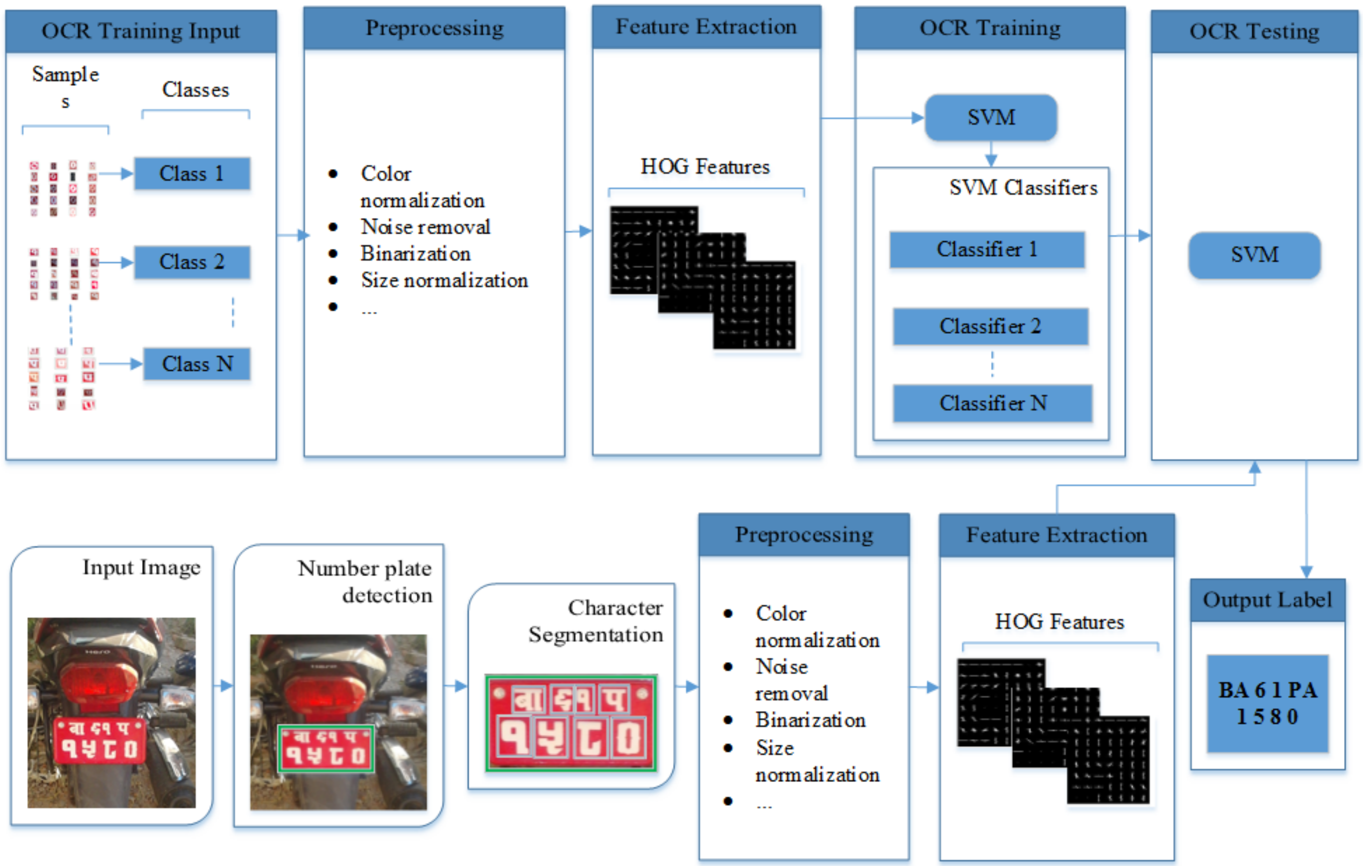


Fig. 3: Automatic Nepali Number Plate Recognition System.

in the opposite direction. This results in a horizontally and vertically aligned image where the text runs horizontally across the localized plate than at an angle. For this, the slope of the localized license plate is calculated using the formula given below.

$$Slope = \frac{average_{left} - average_{right}}{\frac{width}{2}} \tag{2}$$

Where, $average_{left}$ is the average height of the pixel when scanned from left part of the image and $average_{right}$ is the right counterpart.

2) Projection Analysis: Projection analysis is carried out on profile images (Fig. 4) to extract individual characters from localized number plate. As shown in Fig. 4 (right image), the peak of the projection indicates the presence of the characters and the deep valley represents the boundary between the characters. Using a preset threshold based on experimental observations, the regions in the graph can be vertically segmented to get the horizontal width of each character. The first character used in the number plate represents the zone identifier and can contain two sets of symbol to form a single character identifier. So two maxima and minima are obtained for it and the thresholding will extract them separately. To avoid this problem a simple width test is performed and if the width is too small to represent a unique character it is considered as part of the first character that is extracted. However, as the Fig. 4 (right image), represents the plate with aspect ratio $4:3$ i.e. containing the characters in the two rows; the plate needs to be preprocessed unlike the plate with characters in single row. The boundary between two rows should be identified and the plate needs to be divided into two windows containing each row. As shown in the Fig. 4 (left image), analysis of the horizontal projection shows two peaks representing each row and minima representing the boundary between them. The plate is dissected through this point to separate the two rows for further processing by vertical projection.

IV. Character Recognition

After segmenting the individual characters from the localized plate, the characters are passed into the OCR system for the prediction. OCR system implements support vector machine based learning and prediction algorithms. OCR system takes input as a numeric feature vector extracted from given character image. In the learning phase, system is provided with labeled data to learn a specific class behavior. SVM based learning system learns in one-versus-all manner and produces classifiers for each class, which are then used for later prediction of unknown (unlabeled) data. Details of the feature extraction algorithm and support vector machines are described in the following sections.

A. Histograms of Oriented Gradients Features

Histograms of Oriented Gradients features are widely used in computer vision applications for pattern extraction from images. HOG features are robust and easy to compute. HOG features compute the local object appearance and shape within an image using the distribution of intensity gradients or edge directions [13]. HOG method for feature extraction is similar to Scale-Invariant Feature Transform (SIFT), edge oriented histograms, and shape contexts, but it is computed on a dense grid of uniformly spaced cells.

HOG features are computed by dividing the image into small connected regions, called cells, and for each cell compiling a histogram of gradient directions or edge orientations for the pixels within the cell [14]. The combination of these histograms then represents the descriptor. For better accuracy, local histograms can be locally normalized. Normalization results are more robust to illumination changes or shadowing.

B. Support Vector Machines

Support Vector Machine is a machine learning technique widely used in pattern recognition, object classification and regression analysis [15]. Given a set of training data belonging to different classes, SVM attempts to derive a hyper plane or a set of hyper planes that form the boundary between the data belonging to different classes. Any test data can be classified to one of the predefined classes by checking its position relative to the hyper planes. In our context, SVM is used for the classification of segmented characters into one of the twelve trained classes $(\mathit{0\text{-}9}, BA\ and\ PA)$.

The HOG feature vectors of different classes can be visualized in a space forming clusters at different locations such that each cluster contains members belonging to a certain class. The problem is to categorize a new image segment to one of these classes by identifying which cluster is the best fit for the feature vectors of the test image. The boundary between the non-linearly separated clusters of data is identified by SVM. Based on this hyper plane boundary the test image is assigned to a certain region of the multi-dimensional space and hence to the cluster with the best match. Member of each cluster may be spread in a large region in space because of the diversity in images even within a same class. So the sampling image data from the infinite set of population has to be chosen in such a way that the diversity in images belonging to a single class is included. For a cluster, the data which is closest to the neighboring class is called a Support Vector. If we consider two possible classes of data then there are two support vectors one from each class. The plane passing through these two Support Vectors are called Support Planes. They give the boundary of region occupied by their respective classes. If we consider,

$$w^T x + b_0 < -k_1 \tag{3}$$

to be a region to which all the members of class 1 belong to and

$$w^T x + b_0 > +k_1 \tag{4}$$

to be a region to which members of class 1 not belong, where, $w = [w_1, w_2, w_3, \ldots, w_N]$ is a weight vector, $x = [x_1, x_2, x_3, \ldots, x_N]$ is the feature vector and b_0 is the bias, the decision boundary for these two classes can be defined midway between these two planes and is given as

$$w^T x + b_0 = 0 \tag{5}$$

TABLE I: Number plate character dataset.

Character	No. of samples
0	96
1	171
2	176
3	186
4	206
5	194
6	159
7	147
8	135
9	123
BA	206
PA	234
Total	**2033**

Large distance of Support Vectors from the boundary means the classes are very much distinguishable based on the features we have chosen. Using this hyper plane boundary any new test data can be categorized to one of the two sides. The same concept can be elaborated to visualize classification of data to multiple classes by generating a set of hyper planes.

V. Experiments and Results

The aim of this research is to evaluate a complete Automatic License Plate Recognition system for the Nepali number plates. First of all the system is trained against labeled optical character dataset. And then, system is evaluated against new character samples and whole number plate to measure the accuracy of the proposed Automated Nepali Number Plate Recognition System (ANNPRS).

A. Datasets

1) **License Plate Dataset**
There are about 400 images of the license plates captured from different vehicles at different orientation and lighting conditions. Some of them were taken when the vehicles were moving and some of them when the vehicles were parked.
2) **Character Dataset**
Character dataset contains total 2033 samples for 12 classes. Dataset is created by manually cropping license plate characters. Detail of the number of the samples for each class is given in the Table I.

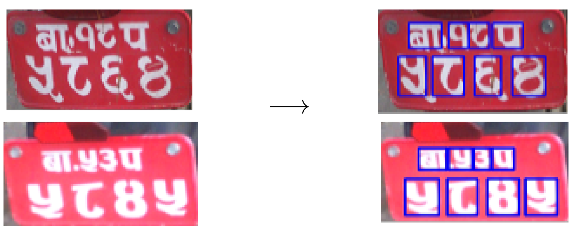
B. Segmentation Results

Proposed system is tested for 12 randomly chosen samples for license plate detection/localization test in which the license-plate of the vehicles was clearly visible. Among those samples, the system correctly localized and segmented the license-plate of 8 samples. The other samples seems to have faded color in their license plate. Some of the number plate segmentation results are shown in Fig. 4.

The localized plates were then used to extract the characters from them. Fig. 5 shows few examples of character segmentation from segmented number plates. In this research, the segmentation accuracy largely depended upon the localization's accuracy and spacing between the characters.

(a) Original image (b) HSV based thresholding for red color (c) Candidate rectangles after approximating from contours (d) Localized license plate

Fig. 4: Number plate segmentation.



(a) Localized license plate (b) Segmented characters

Fig. 5: Character segmentation.

TABLE II: Number of Training and Testing samples.

Character	No. of samples
0	80
1	141
2	134
3	146
4	168
5	157
6	118
7	114
8	108
9	106
BA	166
PA	188
Total	**1626**

(a) Training

Character	No. of samples
0	16
1	30
2	42
3	40
4	38
5	37
6	41
7	33
8	27
9	17
BA	40
PA	46
Total	**407**

(b) Testing

C. Character Recognition Results

This section describes experimentation results in character dataset. Three separate experiments were conducted on character dataset by taking 70%, 80% and 90% of data for training and remaining 30%, 20% and 10% data for testing respectively. For each experiment, data samples were selected randomly and exclusively for training and testing. Final accuracies are averaged measures of all three experiments.

Table II shows training and testing dataset corresponding to one particular experiment (80%-train, 20%-test).

Table III describes the number of test samples given to the classification system and retrieved and un-retrieved samples corresponding to the experiment (80%-train, 20%-test). Un-retrieved samples are the test samples that the system outputs as unknown character class. Retrieved samples are actual retrieved from system with some prediction class. Confusion Matrix (CM) of retrieved samples corresponding to this experiment is shown in Table IV. Confusion matrix describes actual prediction classes for given number of test samples. $CM(i, j)$ represents the count of instances whose known class labels are class i and whose predicted class labels are class j.

TABLE III: Individual Character Recognition Results

Class	Test Samples	Un-retrieved Samples	Retrieved Samples
0	16	3	13
1	30	4	26
2	42	4	38
3	40	6	34
4	38	2	36
5	37	4	33
6	41	7	34
7	33	1	32
8	27	5	22
9	17	0	17
BA	40	6	34
PA	46	5	41
Total	**407**	**47**	**360**

Table V shows experimentation accuracies in character dataset. Accuracies are measured by averaging all the individual accuracies from all the experiments. Average system accuracy evaluates the average per-class effectiveness of a classification system. System error is the average per-class classification error of the system. Precision (also called positive

TABLE IV: Character Recognition Confusion Matrix

Class	0	1	2	3	4	5	6	7	8	9	BA	PA	Total
0	**11**	0	0	1	0	0	0	1	0	0	0	0	13
1	0	**25**	1	0	0	0	0	0	0	0	0	0	26
2	1	1	**34**	2	0	0	0	0	0	0	0	0	38
3	0	2	1	**30**	0	0	0	0	0	1	0	0	34
4	0	0	0	2	**32**	0	0	0	0	2	0	0	36
5	0	2	2	0	2	**25**	0	1	0	0	0	1	33
6	0	0	1	0	0	0	**32**	0	0	1	0	0	34
7	0	0	0	0	2	0	0	**29**	0	1	0	0	32
8	0	1	1	0	0	0	1	0	**18**	1	0	0	22
9	0	0	1	0	1	2	0	0	0	**13**	0	0	17
BA	0	2	0	1	0	2	0	0	0	2	**25**	2	34
PA	0	2	0	0	0	1	0	2	1	0	0	**35**	41
Total	12	35	41	36	37	30	33	33	19	21	25	38	

TABLE V: Character recognition results

Matrics	Score(%)
Average System Accuracy	93.2
System Error	6.79
Precision	87.59
Recall	98.66
F-score	92.79

predictive value) is the number of correctly classified positive examples divided by the number of examples labeled by the system as positive. Recall (also called sensitivity) is the number of correctly classified positive examples divided by the number of positive examples in the test dataset. F-score is the combination of the precision and recall [16].

D. Number Plate Recognition Results

For the experimentation with whole number plate recognition, we collected 8 random number plates that are correctly localized and segmented. Table VI describes the results corresponding to number plate recognition experiment. Table VII shows samples given to the recognition system and corresponding output labels.

VI. Conclusion

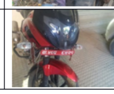
With the rapid development of transportation technology, automatic number plate identification has been essential from different perspectives. It has many real life applications, like traffic monitoring, parking, border control and surveillance. In this research, we have proposed a learning based automatic number plate recognition model for Nepali number plates. Model is experimented with self-collected Nepali number plate dataset and accuracies are measured. Evaluation results shows 87.59% of average Precision, 98.66% of average Recall and 92.79% of average F-Score on segmented character dataset experiment. Also, the accuracy of the complete number plate labeling experiment is obtained as 75.0%. Accuracy of the character recognition is greatly influenced by the segmentation accuracy of the characters. It is also affected by the size, resolution, pose, and illumination of the given images.

This research work can be extended to National wide number plate recognition by extending the character classes. Also, it is recommended to use larger training dataset along with better plate and character segmentation algorithms, for better prediction capability. Due to various standards used around the world, license plate localization cant be generalized and thus, is still a challenging problem around the globe. Moreover, the character's standard is also not properly implemented in Nepal which contributes to disturb the accuracy of any license number recognition system.

TABLE VI: Number plate recognition results

No. of samples	Correctly classified	Miss-classified
8	6	2

TABLE VII: License plate recognition tests.

S.N.	Image	Actual Label	Classified Label	Remarks
1.		बा ६ ३ प ६ २ ० ५	BA 6 3 X 6 2 0 5	Incorrect X:Unclassified character
2.		बा ६ १ प १ ५ ८ ०	BA 6 1 PA 1 5 8 0	Correct
3.		बा ६ प २ ६ ९ ८	BA 6 PA 2 6 9 8	Correct
4.		बा ४ १ प १ ५ १ ७	BA 4 1 PA 1 5 1 7	Correct
5.		बा २ ७ प १ ६ ७ ५	BA 2 7 X 1 6 7 4	Incorrect X:Unclassified characters
6.		बा ५ ० प ५ ६ ७	BA 5 0 PA 5 6 7	Correct
7.		बा ५ ९ प २ ३ ४ ६	BA 5 9 PA 2 3 4 6	Correct
8.		बा ५ ८ प ८ ९ २ १	BA 5 8 PA 8 9 2 1	Correct

References

[1] B. Hongliang and L. Changping, "A hybrid license plate extraction method based on edge statistics and morphology," in *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, vol. 2. IEEE, 2004, pp. 831–834.

[2] S. Saha, S. Basu, M. Nasipuri, and D. K. Basu, "An offline technique for localization of license plates for indian commercial vehicles," *arXiv preprint arXiv:1003.1072*, 2010.

[3] D. Zheng, Y. Zhao, and J. Wang, "An efficient method of license plate location," *Pattern Recognition Letters*, vol. 26, no. 15, pp. 2431–2438, 2005.

[4] T. D. Duan, T. H. Du, T. V. Phuoc, and N. V. Hoang, "Building an automatic vehicle license plate recognition system," in *Proc. Int. Conf. Comput. Sci. RIVF*. Citeseer, 2005, pp. 59–63.

[5] T. Naito, T. Tsukada, K. Yamada, K. Kozuka, and S. Yamamoto, "Robust license-plate recognition method for passing vehicles under outside environment," *Vehicular Technology, IEEE Transactions on*, vol. 49, no. 6, pp. 2309–2319, 2000.

[6] T. Berk, A. Kaufman, and L. Brownston, "A human factors study of color notation systems for computer graphics," *Communications of the ACM*, vol. 25, no. 8, pp. 547–550, 1982.

[7] A. R. Smith, "Color gamut transform pairs," in *ACM Siggraph Computer Graphics*, vol. 12, no. 3. ACM, 1978, pp. 12–19.

[8] S.-L. Chang, L.-S. Chen, Y.-C. Chung, and S.-W. Chen, "Automatic license plate recognition," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 5, no. 1, pp. 42–53, 2004.

[9] D. P. Suri, D. E. Walia, and E. A. Verma, "Vehicle number plate detection using sobel edge detection technique," *International Journal of Computer Science and Technology, ISSN*, pp. 2229–4333, 2010.

[10] J. H. Elder, A. Krupnik, L. Johnston *et al.*, "Contour grouping with prior models," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 25, no. 6, pp. 661–674, 2003.

[11] C.-H. Teh and R. T. Chin, "On the detection of dominant points on digital curves," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 11, no. 8, pp. 859–872, 1989.

[12] L. Jin, H. Xian, J. Bie, Y. Sun, H. Hou, and Q. Niu, "License plate recognition algorithm for passenger cars in chinese residential areas," *Sensors*, vol. 12, no. 6, pp. 8355–8370, 2012.

[13] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1. IEEE, 2005, pp. 886–893.

[14] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

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

[16] M. Sokolova and G. Lapalme, "A systematic analysis of performance measures for classification tasks," *Information Processing & Management*, vol. 45, no. 4, pp. 427–437, 2009.