Detecting Lane Departures Using Weak Visual Features

Ella (Jungsun) Kim, jungsun@stanford.edu
Shane Soh, shanesoh@stanford.edu
Advised by Professor Ashutosh Saxena

***Abstract*—Many current lane departure warning systems rely on strong visual cues, such as distinct lane markings[1][2] or the presence of a leading vehicle[6]. In this project, we propose a system that detects lane departures using only weak visual features derived from dashboard camera footages. In doing so, we aim to build a model that can detect lane departures even on roads with faded or missing lane markings, in inclement weather (e.g. snow-covered roads, fog and heavy rain) and in poor lighting conditions. Our proposed model uses extracted optical flow trajectories[4] of various points in the scene detected using the Shi-Tomasi corner detector[5]. These trajectories are then normalized and featurized using Histogram of Oriented Gradients (HOG). We used SVM with a radial basis function (RBF) kernel to classify these trajectories into one of two categories: *straight* (vehicle is within the lane) and *deviate* (vehicle is leaving the lane). Our SVM model is compared against a logistic regression model that provided a performance baseline. Both models produced reasonable results with our SVM model detecting lane departures up to 88% accuracy.**

I. Introduction

A. Motivation

Lane departure warning (LDW) systems are designed to warn the driver should the vehicle unintentionally leaves its lane on freeways and arterial roads. These systems seek to improve driver safety by addressing the common causes of road incidents and collisions, namely inattention, intoxication, incapacitation and drowsiness, which have resulted in a significant portion of traffic fatalities in the U.S[9]. Many major car manufacturers, such as Ford, Mercedes and Toyota, have started incorporating LDW systems in their vehicles since 2002. However, the biggest limitation of such systems is their inability to track faded, missing or incorrect lane markings.

We seek to address this limitation by developing a LDW system that does not rely on strong visual features such as lane markings or the presence of a leading vehicle. Instead, our model reliably detects lane departures by inferring the relative motion of the vehicle to its surroundings through the tracking of various weak visual features in the scene.

B. Objective

Given optical flow data extracted from road view footages, we aim to classify each time interval, i.e. a certain number of video frames, into two categories: *straight* (vehicle is within the lane) and *deviate* (vehicle is leaving the lane). We implemented and compared two different machine learning algorithms, SVM with radial basis function (RBF) kernel and logistic regression, to identify the better performing model and its corresponding features.

II. Data Collection and Feature Extraction

A. Data Collection

We collected and labeled over 15 hours of driving footage on both highway and local routes using off-the-shelf dashboard video cameras. We chose to focus on highway driving videos for this project and had a total of 142 straight examples and 214 deviate examples. Most of the highway videos were taken on Route 101.

B. Generating Optical Flow Trajectories

Fig. 1. Optical flow tracking of detected corners and its corresponding mask

We used the Shi-Tomasi corner detection[5] algorithm to detect points in the scene that are then tracked using the Lucas-Kanade method of optical flow[3]. This method was chosen over dense optical flow (which is commonly used in similar motion-detection tasks such as human recognition[7][8]) for two reasons: 1) it is computationally more efficient and is thus more suited for real-time applications such as LDW systems, and 2) it is more suitable for night-time driving footages where the trackable features are likely to be sparse.

We also employed a number of heuristics at this stage. The corner detection algorithm was made to run at every fixed interval of 5 frames on a masked area as shown in Fig 1. The mask was chosen to encourage the detection of points around the vanishing point of the scene as tracking these points were found to be most indicative of relative motion. The mask also reduced the number of detected points on relatively irrelevant image features such as the vehicle dashboard and cars on the opposing traffic. The mask, however, did not limit the region in which the points can be tracked. Instead, the points were

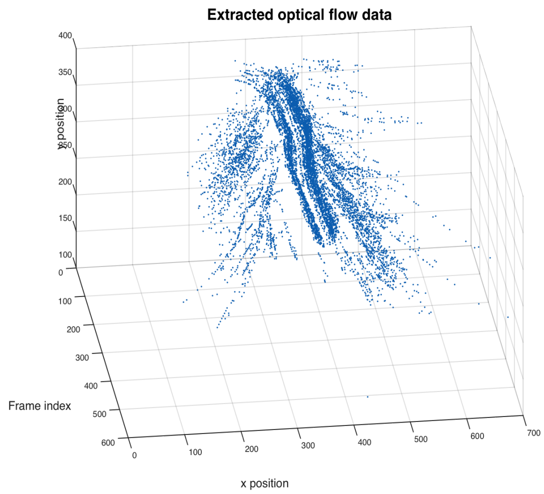

Fig. 2. Typical set of extracted optical flow trajectories (x and y positions plotted against frame indices, i.e. time)

allowed to be track until they vanished from the scene. We additionally imposed an artificial threshold on the speed of the points being tracked, i.e. the displacement of each tracked point across frames, as we found that most of these noisy optical flow trajectories tend to be the ones corresponding to fast-moving points.

C. Extracting HOG Features

We collapsed the generated optical flow trajectories (see Fig 2) into 2D binary image representations of x-positions of detected points against time (see Fig 3). We then normalized each set of trajectories based on the mean x-positions of the detected points and featurized them using Histogram of Oriented Gradients (HOG). HOG essentially counts the occurrences of gradient orientation in localized portions of an image and generates a feature set that corresponds to histogram bins of orientations. We found that HOG features accurately characterize the optical flow trajectories while remaining relatively invariant to the presence of noisy trajectories. We also used feature selection as a means of normalizing the locations of the HOG cells that contain the trajectories (to be discussed further in the next section).

There are three main parameters that we can vary: the interval size, i.e. number of frames used per training example, cell size and the number of orientation histogram bins used for HOG feature extraction. The interval size is perhaps the most important parameter as it represents a direct tradeoff between accuracy and the predictive value of our models. Intuitively, larger intervals give the model more information to train and test on, and thus result in higher accuracies. However, larger intervals correspond to lower predictive value as more frames are required before the model can detect the lane departure with reasonable accuracy, i.e. more time would have elapsed. Our current model can achieve the stated 88% accuracy using an interval of 50 frames (corresponding to about 2 seconds of video). When applied to a LDW system, this means that it

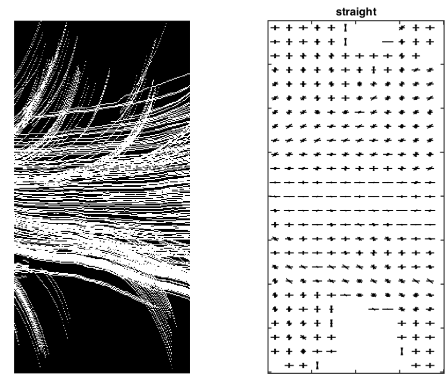

Fig. 3. 2D binary image representation of trajectories and its corresponding HOG features

may take up to 2 seconds into the maneuver before the system can detect lane departures with the aforementioned accuracy.

For this project, we used intervals of 50 frames, 16x16 cells and 9 histogram bins, which gave us a 9504-large sparse feature set that we then reduced to 1100 features using feature selection.

III. Methods and Evaluation

A. Feature Selection

Initial fitting on the entire feature set (of 9504 features) yielded 27% cross validation error with the SVM model and 20% cross validation error with the logistic regression model. In order to determine an optimal set of features efficiently, we performed filter feature selection using the minimum attainable classification error as our ranking criterion, i.e. the criterion used to assess the significance of every feature for binary classification. We plotted various performance metrics of the SVM and LR models as we increase number of features selected as shown in Fig 4 and Fig 5.

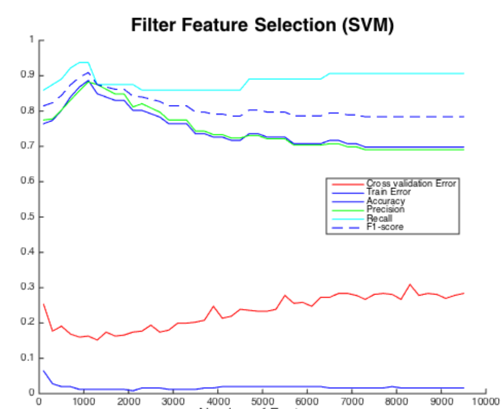

Fig. 4. Plot of various metrics against number of features selected (SVM)

Fig. 5. Plot of various metrics against number of features selected (Logistic Regression)

The optimal number of features for the SVM model was determined to be 1100 features. As shown in Fig 4, the cross validation error decreased with increasing feature size before increasing again after approximately 1000 features. Similarly, accuracy and precision increased initially before decreasing after approximately 1000 features. For the logistic regression model, the training error dropped to zero very quickly as the number of features increased from 1 to 100 (not visible in the plot given the x-axis scale). The cross-validation error (as well as other metrics) seems to remain relatively invariant to the size of the feature set (see Fig 5). This is expected as regression models have the tendency to overfit their training data.

Given this analysis, we can also deduce that logistic regression is unsuitable given the high feature dimensionality. On the other hand, SVM is particularly suited given the sparseness and high dimensionality of our feature set. There have also been many prior works that have successfully utilized SVM for classifying HOG descriptors, particularly in detecting humans or human actions[7][8].

B. Logistic Regression Model

TABLE I. CONFUSION MATRIX FOR LOGISTIC REGRESSION MODEL

	Predicted *Straight*	Predicted *Deviate*
Actual *Straight*	31	11
Actual *Deviate*	12	52

We used a logistic regression model to provide a baseline performance (results shown in Table I). Out of total 356 training examples, 250 were used to train the model and 106 (30% of the total examples) were used to test. Cross-validation error was computed using 10-fold cross-validation. Logistic regression classified the test set with 78% accuracy and a cross-validation error of 19%.

C. SVM Model

We trained a SVM model with RBF kernel, i.e. $K(x,z) = exp(-\gamma\|x-z\|^2)$, with a box constraint of 1 and a kernel scale of 33.67 (heuristically chosen using subsampling). The RBF kernel has an infinite dimensional kernel feature space and is suitable for exploiting non-linear relationships between features.

For the SVM model, we were essentially solving the following optimization problem (with box constraint, C = 1):

$$max_\alpha W(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2}\sum_{i,j=1}^{m} y^{(i)}y^{(j)}\alpha_i\alpha_j\langle x^{(i)}, x^{(j)}\rangle$$
$$s.t. 0 \leq \alpha_i \leq C, i = 1, ..., m$$
$$\sum_{i=1}^{m} \alpha_i y^{(i)} = 0$$

TABLE II. CONFUSION MATRIX FOR SVM MODEL

	Predicted *Straight*	Predicted *Deviate*
Actual *Straight*	34	8
Actual *Deviate*	4	60

As shown in Table II, SVM classified the test set with 88% accuracy and cross validation error (using 10-fold) of 16%. Our model shows high recall (94%) but a comparatively lower precision (88%). The higher number of false negatives seems to be a result of models tendency to classify driving within the lane on curved roads as a departure from the lane. However, this should improve with greater number of training examples that depict *straight* driving on curved roads.

D. Comparison of Models

TABLE III. SVM VS. LOGISTIC REGRESSION (LR)

	CV Error	Accuracy	Precision	Recall	F-score
SVM	0.1573	0.8868	0.8824	0.9375	0.9091
LR	0.1882	0.7830	0.8254	0.8125	0.8189

For the purpose of this project, we chose to evaluate our models primarily on its precision and recall (and hence its F-score). The recall is a particularly important metric for our application as we seek to minimize the number of false warnings. Reducing the number of false negatives to close to zero would be essential in applications that apply corrective actions to the vehicle upon detecting lane departures.

From Table III, it is clear that SVM outperforms logistic regression in every given metric. This is an expected result given that our model has a high-dimensional feature space and is trained on a small sample size.

IV. ADDITIONAL ANALYSIS

Once we identified that SVM is the better model for our purpose, we experimented with various methods to improve the performance of our model. In this section, we will only discuss the methods that yielded meaningful observations.

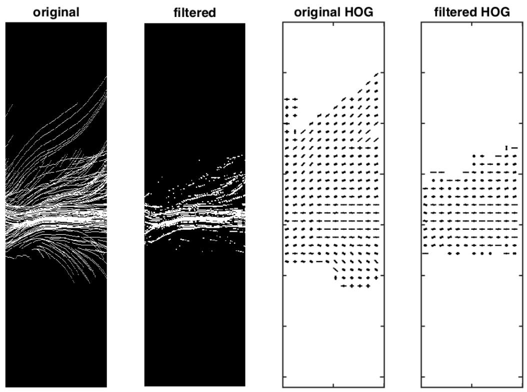

Fig. 6. Trajectories and respective HOG features before and after morphological opening

A. *Pre-processing Optical Flow Trajectories*

We experimented with two methods of cleaning the optical flow trajectories in an attempt to improve the performance of the model: 1) density-based clustering methods (e.g. DBSCAN) to select only the densest clusters of trajectories, and 2) morphological opening on the binary image representations to remove sparse trajectories, i.e. noisy trajectories (see Fig 6).

However, density-based clustering methods were too slow for our purpose (especially given the large number of optical flow data we were dealing with) and morphological opening resulted in deteriorated performance for both SVM and logistic regression models. This is likely because many of what we believed were noisy trajectories might have been structured enough to have sufficient discriminative value.

B. *Error Analysis*

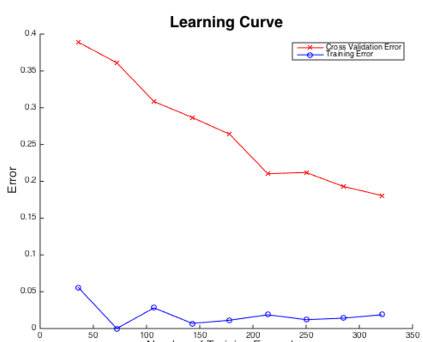

Fig. 7. Learning curve depicting training error and cross-validation error against number of samples

We plotted the learning curve for our model as shown in Fig 7. The low training error but much higher cross-validation error is indicative of a high variance, i.e. over-fitting. The learning curve also indicates that the performance of our model is likely to benefit greatly from having more training examples as more data will help the model generalize better. The high variance is expected given the high dimensionality of our features, especially in comparison to the small number of training examples.

V. Conclusions

We conclude that using SVM with a radial basis function kernel and filter feature selection gives us a reasonable accuracy of 88%, which is notably higher than the logistic regression model (78%). However, the performance of our SVM model is still below that of conventional LDW systems that use strong visual features (typically around 98% and above). Nonetheless, the findings from this project have shown us that such a model could be a viable approach for detecting lane departures. Furthermore, as our approach relies solely on tracking weak visual features present anywhere in the scene, it is expected to be more robust compared to existing methods that are dependent on strong visual cues. In this aspect, we can expect our model to outperform existing models given non-ideal conditions (e.g. inclement weather, low light, etc).

VI. Future Work

We intend to train our model with more data, particularly of driving footages taken in the night and in inclement weather. Our findings indicate that the accuracy of our SVM model will greatly improve as a result. We also intend to extend the model to do multiclass classification, particularly to differentiate between left and right lane deviations as well as other actions such as turns and merges.

We ultimately plan to incorporate our findings in a bigger overall project (under the guidance of Prof. Ashutosh Saxena) to build better predictive models for assistive driving technologies. Specifically, we intend to 1) combine existing features with additional features such as the driver's head pose and GPS data to predict driver's intents, 2) use our system as a method of determining the current lane the vehicle is in (when given road information, such as the number of lanes), and 3) use our system to automatically label drivers actions on large quantities of driving footage which can then be used to train other models.

Acknowledgment

We would like to thank Prof Ashutosh Saxena and Ashesh Jain for their guidance and support. We would also like to thank the few individuals who helped us with the data collection.

References

[1] Hsiao, P., Hung, K., Huang, S., Kao, W., Hsu, C., Yu, Y., (2011). An Embedded Lane Departure Warning System. 2011 IEEE 15th International Symposium on Consumer Electronics.

[2] McCall, J. C., Trivedi, M. M., (2006). Video-based lane estimation and tracking for driver assistance: survey, system, and evaluation. IEEE Conference on Intelligent Transportation Systems.

[3] Lucas, B. D., Kanade, T., (1981). An Iterative Image Registration Technique with an Application to Stereo Vision. International Joint Conference on Artificial Intelligence.

[4] Tomasi, C., Kanade, T, (1991). Detection and Tracking of Point Features. Carnegie Mellon University Technical Report.

[5] J. Shi and C. Tomasi, (1994). Good Features to Track. IEEE Conference on Computer Vision and Pattern Recognition.

[6] Olsen, E. C. B., (2003). Modeling slow lead vehicle lane changing. Doctoral Dissertation, Department of Industrial and Systems Engineering, Virginia Polytechnic Institute.

[7] Dalal, N., Triggs, B., (2005). Histograms of Oriented Gradients for Human Detection. IEEE Conference on Computer Vision and Pattern Recognition.

[8] Zhu, Q., Avidan, S., Yeh, M., Cheng, K., (2006). Fast Human Detection Using a Cascade of Histograms of Oriented Gradients. IEEE Conference on Computer Vision and Pattern Recognition.

[9] Hadden, J.A., Everson, J.H., Pape, D.B., Narendran, V.K., Pomerleau, D.A., (1997). Modeling and analysis of driver/vehicle dynamics with ”run-off-road” crash avoidance systems. In Proceedings of the 30th ISATA Conference.