2007 IEEE International Conference on
Robotics and Automation
Roma, Italy, 10-14 April 2007

Classification-Based Wheel Slip Detection and Detector Fusion for Outdoor Mobile Robots

Chris C. Ward and Karl Iagnemma

Abstract— **This paper introduces a signal-recognition based approach for detecting autonomous mobile robot immobilization on outdoor terrain. The technique utilizes a support vector machine classifier to form class boundaries in a feature space composed of statistics related to inertial and (optional) wheel speed measurements. The proposed algorithm is validated using experimental data collected with an autonomous robot operating in an outdoor environment. Additionally, two detector fusion techniques are proposed to combine the outputs of multiple immobilization detectors. One technique is proposed to minimize false immobilization detections. A second technique is proposed to increase overall detection accuracy while maintaining rapid detector response. The two fusion techniques are demonstrated experimentally using the detection algorithm proposed in this work and a dynamic model-based algorithm. It is shown that the proposed techniques can be used to rapidly and robustly detect mobile robot immobilization in outdoor environments, even in the absence of absolute position information.**

I. Introduction

Mobile robot wheel slip frequently occurs when driving over low-traction terrain, deformable terrain, steep hills, or during collisions with obstacles, and can frequently result in robot immobilization. As position estimation systems typically rely heavily on wheel odometry [1],[2], undetected immobilization can lead to poor localization between low-frequency absolute position updates, resulting in poor map registration. Autonomous robots should quickly detect that they are immobilized in order to take appropriate action, such as planning an alternate route away from the low-traction terrain region or implementing a traction control algorithm [3].

Wheel slip can be accurately estimated through the use of encoders by comparing the speed of driven wheels to that of undriven wheels [4], however this does not apply for all-wheel drive vehicles or those without redundant encoders. Ojeda and Borenstein have proposed comparing redundant wheel encoders against each other and against yaw gyros as a fuzzy indicator of wheel slip, even when all wheels are driven [5], and have also proposed a motor current-based slip estimator [6]; however this technique requires accurate current measurement and terrain-specific parameter tuning. A body of work exists in the automotive community related to traction control and anti-lock braking systems (ABS); however, this work generally applies at significantly higher speeds than is typical for autonomous robots [7],[8].

A potentially simple approach to detecting robot slip and immobilization is to analyze GPS measurements. However, nearby trees and buildings can cause signal loss and multipath errors and changing satellites can cause position and velocity jumps [9],[10]. Additionally, GPS provides low frequency updates (e.g. typically near 1 Hz [11]) making GPS alone undesirably slow for immobilization detection.

Another potentially simple approach could rely on comparison of wheel velocities to a robot body velocity estimate derived from integration of a linear acceleration measurement in the direction of travel. As will be shown in Section IIIb, however, such an approach is not robust at low speeds during travel on rough, outdoor terrain.

Machine learning/classification techniques have been employed in various mobile robotics applications including vibration-based terrain classification [12] and self-supervised vision-based road detection [13], as well as other applications such as speech recognition [14]. The authors are aware of no previous work utilizing these techniques for robot immobilization detection.

Here a method is presented for detecting robot immobilization using a signal-recognition approach. Offline, a support vector machine (SVM) classifier is trained to recognize immobilized conditions within a feature space formed using inertial measurement unit (IMU) and optional wheel speed measurements. The trained SVM can then be used to quickly detect immobilization with little computation. Experimental results show the algorithm to quickly and accurately detect immobilization in various scenarios.

To improve algorithm accuracy and robustness, immobilization detector fusion techniques have been explored. One technique is proposed to minimize false immobilization detections. A second is proposed to increase overall detection accuracy while maintaining rapid detector response. The two fusion techniques are demonstrated with experimental data using the algorithm proposed in this work and a dynamic model-based algorithm described in an accompanying paper [15].

This paper is organized as follows. In Section II the classifier-based immobilization detection algorithm is presented and its performance is experimentally demonstrated in Section III. In Section IV two detector fusion techniques are presented along with a summary of the dynamic model-based slip detector previously developed

Manuscript received September 14, 2006. This work was supported by the DARPA LAGR program. The authors are with the Department of Mechanical Engineering, Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: chriscw@mit.edu, kdi@mit.edu).

1-4244-0602-1/07/$20.00 ©2007 IEEE.

[15]. The effectiveness of the fusion techniques is analyzed through experimental results. In Section V conclusions are drawn from this work and future work is suggested.

II. Mobile Robot Immobilization Classification

A. Classification Algorithm Overview

The algorithm proposed in this work was inspired by the observation that a human in a vehicle with eyes closed can quickly and robustly distinguish whether the vehicle is:

1) completely stopped with wheels stopped,
2) driving normally over outdoor terrain, or
3) immobilized, with the wheels rotating but slipping.

Even in the absence of training for this task and without visual feedback, a human can interpret clues such as vehicle heave/jounce and motor/engine sound signature to discriminate between cases 1-3 with reasonable accuracy.

The proposed algorithm uses a signal-recognition approach to detect mobile robot immobilization (case 3 above) based on inertial and wheel speed measurements. The measurements are used to form n features that can be used to distinguish between the two classes "immobilized" and "normal driving." A support vector machine (SVM) is used to determine class boundaries within the n-dimensional feature space [16].

The SVM is trained using a hand-labeled data set of l instance-label pairs ($\boldsymbol{x}_1$, y_1), … , ($\boldsymbol{x}_i$, y_i), … , ($\boldsymbol{x}_l$, y_l) with $\boldsymbol{x}_i \in \Re^n$ and $y_i \in \{-1,1\}$ [17],[18]. In this work, "normal" is labeled as $y_i = -1$ and "immobilized" as $y_i = 1$. The l training instance feature vectors, $\boldsymbol{x}_i$, are combined to form the l x n feature matrix, $\boldsymbol{X} = [\boldsymbol{x}_1 \ \cdots \ \boldsymbol{x}_l]^T$, and the labels form the l x 1 training label vector, $\boldsymbol{y} = [y_1 \ \cdots \ y_l]^T$.

Classification accuracy is improved by scaling each feature type to have similar magnitudes [18]. To scale each feature to the range [-1, 1], the n x n scale factor matrix, $\boldsymbol{S}$, is formed such that:

$$\boldsymbol{S}_{i,j} = \begin{cases} \dfrac{1}{\max(|\text{column } j \text{ of } \boldsymbol{X}|)} & \text{if } i = j \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

and the scaled training feature matrix, $\tilde{\boldsymbol{X}}$, is then:

$$\tilde{\boldsymbol{X}} = \boldsymbol{X} \cdot \boldsymbol{S} . \tag{2}$$

$\tilde{\boldsymbol{X}}$ and $\boldsymbol{y}$ are used to train a SVM using a radial basis function (RBF) kernel as this kernel performs well with both non-linear and linear class relations and requires few kernel parameters [18]. SVM parameters are found using a grid search to systematically find a parameter set that minimizes the average classification error and error standard deviation of a v-fold cross-validation [18]. The final SVM model is trained using the best SVM parameter set and the entire training data set.

The parameter search and SVM training can be computationally expensive. However training is performed only once, offline, producing an SVM model suitable for computationally inexpensive online classification. Note that during online classification, each measured feature vector, $\boldsymbol{x}$, is first multiplied by the scale factor matrix, $\boldsymbol{S}$, before classification by the trained SVM.

During online classification, the output of the SVM's decision function is a scalar decision value, $\hat{f} \in (-\infty,\infty)$, where the value of $\hat{f}$ is a measure of the distance of the instance from the class boundary in the n-dimensional feature space. Typically an instance is labeled as:

$$label = \begin{cases} \text{immobilized (1)} & \text{if } \hat{f} > 0 \\ \text{normal (-1)} & \text{if } \hat{f} < 0 \\ \text{unknown (0)} & \text{if } \hat{f} = 0 \end{cases} \tag{3}$$

However, increased accuracy can usually be achieved at the expense of lowered labeling completeness (i.e. labeling more instances "unknown") using the following:

$$label = \begin{cases} \text{immobilized (1)} & \text{if } \hat{f} > threshold \\ \text{normal (-1)} & \text{if } \hat{f} < -threshold \\ \text{unknown (0)} & \text{if } \textit{-threshold} \le \hat{f} \ge threshold \end{cases} \tag{4}$$

where $threshold \ge 0$. In this work (3) has been used unless otherwise specified, meaning that all data has been classified.

B. Feature Vector Selection

In this work four features have been chosen to form the feature vector $\boldsymbol{x}_i = [x_{i,1}, x_{i,2}, x_{i,3}, x_{i,4}]$. Each feature is a numerical representation of sensor data that attempts to mimic the sensory cues a human operator would exploit when attempting to detect immobilized conditions. Data is sampled at a rate f_s and a numerical transform is calculated on a set of N data points for each feature instance. Fig. 1 illustrates the coordinate system used in feature definitions.

The first two features were chosen as the variance of the N element groupings i of roll rate, $\dot{\boldsymbol{\theta}}_{i,N}$, and pitch rate, $\dot{\boldsymbol{\varphi}}_{i,N}$, such that:

$$x_{i,1} = \mathrm{var}(\dot{\boldsymbol{\theta}}_{i,N}) = \mathrm{E}\left((\dot{\boldsymbol{\theta}}_{i,N} - \mathrm{E}(\dot{\boldsymbol{\theta}}_{i,N}))^2\right), \tag{5}$$

$$x_{i,2} = \mathrm{var}(\dot{\boldsymbol{\varphi}}_{i,N}) = \mathrm{E}\left((\dot{\boldsymbol{\varphi}}_{i,N} - \mathrm{E}(\dot{\boldsymbol{\varphi}}_{i,N}))^2\right). \tag{6}$$

These two features are a measure of the degree of roll and pitch experienced by a vehicle during travel over uneven outdoor terrain.

The third feature was chosen as a measure of the variation in the z-axis (vertical) acceleration. The variance is a measure of the total variation from the mean over all frequencies; however empirical results have shown that only high frequency z-axis acceleration signal variation effectively distinguishes immobilized conditions. For feature three, $\mathbf{P}_{\mathbf{a}_z,i}$ the p element vector of the power spectrum coefficients of grouping i of z-axis acceleration is calculated using a discrete Fourier transform, where:

$$p = \left\lceil \frac{N+1}{2} \right\rceil . \tag{7}$$

Where $\lceil \ \rceil$ is the ceiling function. Then feature three is calculated as:

$$x_{i,3} = \sum_{k=\lceil p/2 \rceil}^{p} P_{a_z,i,k} \ . \tag{8}$$

For this work, $N = 50$ was chosen and $f_s = 100$ Hz, resulting in a sum of the frequency content from 25 to 50 Hz. This frequency range was empirically determined to perform well for the robot system used in this work.

Feature four was chosen as the mean of the magnitude of the wheel angular accelerations:

$$x_{i,4} = \text{mean}\left(\left|\dot{\boldsymbol{\omega}}_{lft,i,N} + \dot{\boldsymbol{\omega}}_{rt,i,N}\right|\right) = \text{E}\left(\left|\dot{\boldsymbol{\omega}}_{lft,i,N} + \dot{\boldsymbol{\omega}}_{rt,i,N}\right|\right) \tag{9}$$

where $\dot{\boldsymbol{\omega}}_{lft,i,N}$ and $\dot{\boldsymbol{\omega}}_{rt,i,N}$ are the N element groupings i of the left and right wheel angular accelerations, respectively. During outdoor driving, terrain unevenness leads to variations in wheel torque, leading to variations in wheel angular acceleration. This variation is minimized when the robot is immobilized.

III. Experimental Results

A. Robot Description

An autonomous mobile robot developed for the DARPA LAGR (Learning Applied to Ground Robots) program [19] has been used to experimentally validate the algorithm (Fig. 2). The robot is 1.2 m long x 0.7 m wide x 0.5 m, has four rubber pneumatic tires, and is a front-wheel differential-drive configuration. The robot is equipped with 4096 count per revolution front wheel encoders, an Xsens MT9 IMU, and a Garmin GPS 16 differential GPS (not used in this algorithm). IMU and wheel encoders are sampled at 100 Hz. The robot has been used to collect data to process offline using a Matlab implementation of the slip detector.

B. Algorithm Performance

The SVM classifier was trained on data gathered during traversal of mud, loose mulch, and various grasses at speeds ranging from 0 to 1.0 m/s. The training data included 14 instances of the robot coming to a complete stop with the wheels still spinning, which were initiated by retarding robot motion with a spring scale. Using $N = 50$, the classifier was trained with 408 instance-label pairs, 18% of which were labeled as immobilized.

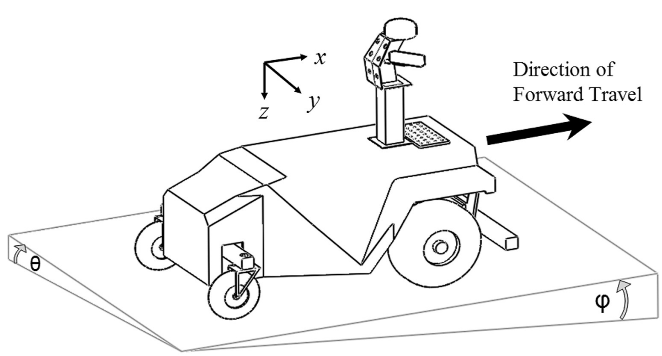

Fig. 1. Body-fixed coordinate system used in this work.

Fig. 2. The LAGR robot.

The classifier was tested using two distinct data sets. In the first set, the robot was driven once again over grass; however immobilization was initiated when the robot experienced significant wheel slip while attempting to surmount a hill. In the second set, the robot was driven over loose gravel mixed with dry, brittle soil, and immobilization was initiated by retarding robot motion with a spring scale. Note that this terrain type was not present in the training data set.

Test results using all four features described in Section IIb are shown in Fig. 3. Total classification accuracy was 94.7%. The figure shows that all incorrectly labeled points were near an actual immobilized period, with 98.1% of normal points correctly classified. The 1.9% of normal points classified as immobilized were all near the start or end of an immobilized period, which could indicate small errors in hand labeling of these extremal points. 75% of immobilized instances were classified correctly; however all immobilized periods were recognized as immobilized in at least some of the data instances comprising that occurrence.

Using only the first three features so that only IMU measurements were required, total classification accuracy was 92.0%, with 97.7% of normal instances and 59.1% of immobilized instances correctly classified. With only three features, classification accuracy was reduced, however false immobilized detections remained low and all immobilized occurrences were again detected.

Fig. 4 shows a receiver operating characteristic (ROC) curve for classification of the test data set using all four features. The vertical axis shows the percentage of total instances that are classified correctly while the horizontal axis shows the percentage classified incorrectly. The curves are generating by progressively increasing *threshold* in (4), causing fewer points to be classified and more points to be "unknown." Thus, increasing *threshold* results in a more conservative classifier. The upper-right endpoint of each line is the classifier accuracy with all instances classified (*threshold* = 0).

It can be seen that as threshold is increased, the percent of incorrect classification initially deceases rapidly, while the percent correct remains near constant, meaning in this region the majority of correctly labeled points were further than *threshold* from the class boundary. This curve shows the

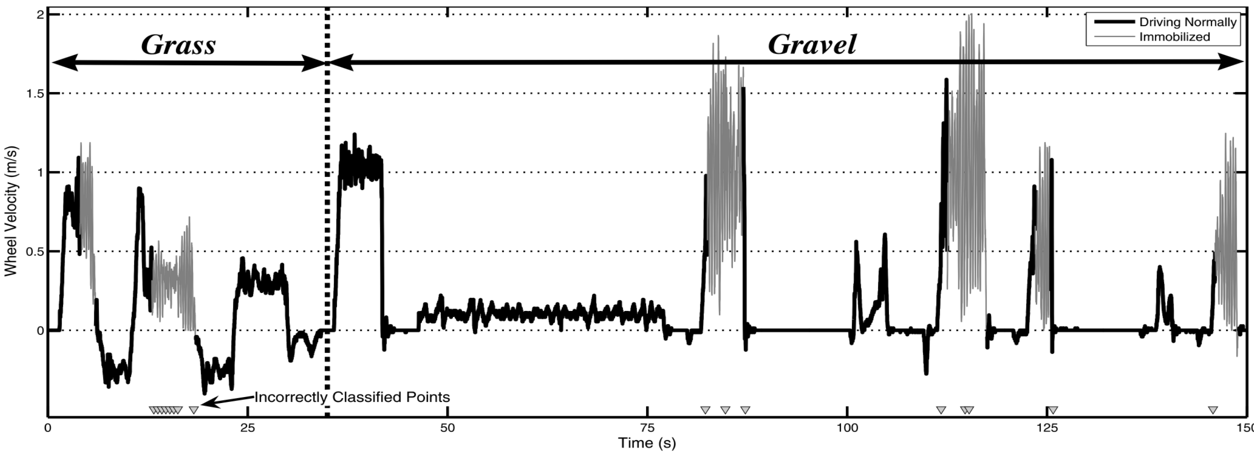

Fig. 3. Experimental results of classifier-based immobilization detection. Each incorrectly classified point is a 0.5 second instance. Wheel velocity is effective linear velocity at wheel radius.

possible tradeoffs between number of instances labeled and labeling accuracy and can be a useful design tool.

One potentially simple slip detection technique is to estimate robot body velocity by integrating acceleration measurements (after subtracting gravitational acceleration due to vehicle pitch) then comparing this estimate against wheel velocity, thereby estimating wheel slip [20]. Fig. 5 compares wheel velocity with estimated body velocity for the grassy hill data set. At low speeds accelerometer errors dominate, causing the velocity estimate to quickly diverge. In this case a detector based on this estimate would detect immobilized for the majority of the data set and be ineffective. Because the velocity estimate error is essentially a random walk, in some cases such a detector would estimate the velocity to always be larger than the wheel velocity, thus never detecting immobilization.

IV. Detector Fusion

A. Fusion Techniques

To increase immobilization detection accuracy two techniques have been explored to fuse multiple detector outputs. The first technique (termed Fusion 1) is designed to minimize false immobilization detections at the expense of increasing the number of immobilized instances incorrectly classified as normal. For d detectors, D_i, each with output 1 for "immobilized" and -1 for "normal":

$$\text{Fusion } 1 = \begin{cases} 1 & \begin{array}{l}\text{if } (D_1 = 1) \text{ AND } (D_2 = 1)\ldots \\ \ldots \text{AND } (D_i = 1)\ldots \text{AND } (D_d = 1)\end{array} \\ -1 & \quad\quad \text{otherwise} \end{cases} . \qquad (10)$$

Thus Fusion 1 detects immobilized only if all detectors agree that the robot is immobilized.

The second technique (termed Fusion 2) is designed to increase total detection accuracy and yield faster immobilization detection than Fusion 1. For Fusion 2, each detector output, D_i, is expressed as a continuous variable on the interval [-1, 1], with an output of 1 meaning the detector is completely confidant that the robot is immobilized, -1

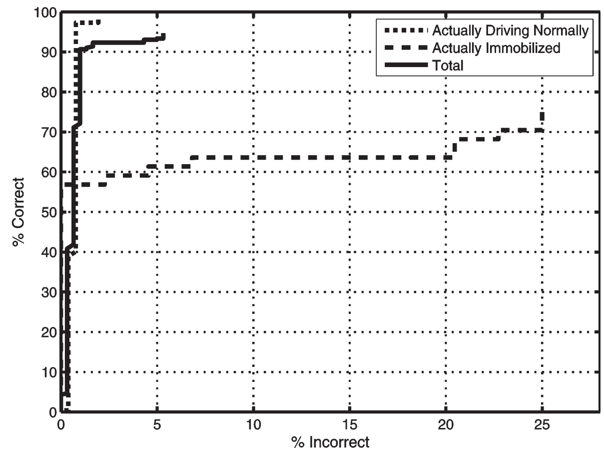

Fig. 4. ROC curve for immobilization detection experimental results.

meaning the detector is completely confidant the robot is driving normally, and 0 meaning there is an equal probability of the robot being immobilized or driving normally. Fusion 2 is a weighted average of the detector outputs:

$$\text{Fusion } 2 = \begin{cases} 1 & \text{if } \sum_{i=1}^{d} w_i D_i > a \\ -1 & \sum_{i=1}^{d} w_i D_i < -a \\ 0 & \text{otherwise} \end{cases} \qquad (11)$$

where a is a threshold value and w_i are weights with:

$$\sum_{i=1}^{d} w_i = 1 . \qquad (12)$$

B. Dynamic Model-Based Wheel Slip Detector Summary

An alternate, dynamic model-based wheel slip detection algorithm was proposed in [15] and has been studied in conjunction with the classification-based algorithm proposed in this work to test the efficacy of the fusion techniques. The proposed approach uses a dynamic vehicle model fused with wheel encoder, IMU, and (optional) GPS measurements in an extended Kalman filter (EKF) to create an estimate of the robot's longitudinal velocity. The proposed algorithm utilizes a novel tire traction/braking model in combination with sensor data to estimate external resistive forces acting upon the robot and calculate the robot's acceleration and velocity. Weak constraints are used to constrain the evolution of the resistive force estimate based upon physical reasoning. The algorithm has been shown to accurately detect immobilized conditions on a variety of terrain types and provide an estimate of the robot's velocity during "normal" driving. In experimental testing the algorithm falsely labeled ~0.2% of data points as immobilized. Near perfect detection is desired; which may be achievable by fusing multiple detection algorithms.

One drawback of the model-based slip detection algorithm is that it requires identification of a small number of physical tire model parameters. The classification-based approach presented here was developed as an alternative, model-free approach to detecting robot immobilization.

C. Fusion Results

The performance of the fusion techniques described in Section IVa was studied using the detector proposed in this work (i.e. the SVM method) and the detector proposed in [15] (i.e. the EKF method). The output of the SVM method was scaled for Fusion 2 by first determining the smallest *threshold* for which all classified training points are classified correctly, $threshold_{100\%}$. Then:

$$D_{SVM} = \mathrm{sat}\left(\frac{\hat{f}}{threshold_{100\%}}, 1\right) \tag{13}$$

where the saturation function, sat(x, y), is defined here as:

$$\mathrm{sat}(x,y) = \begin{cases} x & \text{if } |x| < |y| \\ \mathrm{sign}(x)\cdot|y| & \text{otherwise} \end{cases}. \tag{14}$$

The EKF method outputs a detected class for each of the N data points that make up instance i of the SVM method, but does not output a confidence value. D_{EKF} is therefore taken as the mean of the N data points, providing an estimate of the detector's confidence. If half of the N points are classified as immobilized, then there is approximately a 50% chance the robot was immobilized during those data points and D_{EKF} = 0. This estimate has the drawback of assigning low confidence when immobilization begins near the end of the N points, possibly leading to sub-optimal detection time; however it provides a computationally simple method to test

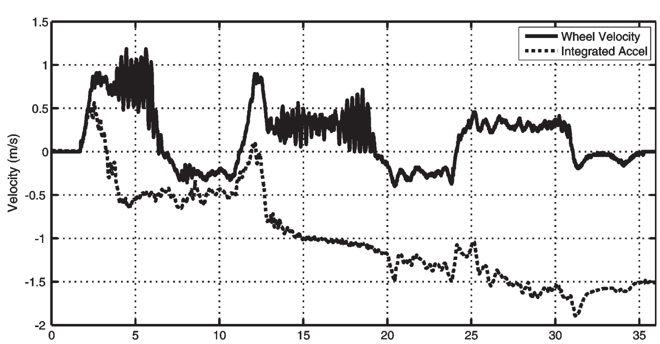

Fig. 5. Inaccuracy of estimating robot velocity by integrating measured accel

the fusion technique performance. For Fusion 2, $w_1 = w_2 = 0.5$ and $a = 0$ were used.

Fig. 6 shows a dataset of the robot driving over loose mulch which demonstrates the relative performance of the fusion techniques. In sections A and C the robot was driven normally under remote control, and in sections B and D the robot was commanded to drive forward at 0.5 m/s but was restrained with a spring scale, causing immobilization. The bottom plot indicates the moments when immobilization was detected by the two detectors and two fusion techniques.

It can be seen that in section A the SVM method falsely detects immobilization, likely due to the rapid wheel speed oscillation. The EKF method, however, correctly labels this instance as normal driving, allowing both fusion techniques to correctly label this section as normal. Similarly, in section C the EKF method misclassifies an instance as immobilized, but the SVM method and both fusion methods correctly classified this section.

In section B, the SVM method detected immobilization very rapidly, while the EKF method's detection time was approximately 1.0 second slower. In this case, the SVM method detects immobilized immediately after the robot begins to decelerate, while the EKF method detects immobilized when the robot comes to a stop. As expected, Fusion 1 only detected immobilization when both detectors agreed. Fusion 2 was able to detect immobilization approximately 0.5 seconds sooner than Fusion 1 because the SVM method expressed high confidence in its output while the EKF method expressed an uncertain output (i.e. an output near 0). In section D, the SVM method expressed a low confidence in its early immobilization detection and neither fusion technique detected immobilization until the EKF method was in agreement. Table I compares detection accuracy of the four methods when run on the Section IIIb test set, which included 301 half second instances. All four techniques detected the 6 immobilized periods. The SVM method detected immobilized the quickest followed by Fusion 2; however in some cases the SVM method detected immobilized before the vehicle was stopped, accounting for the 3 false positives. Both fusion techniques eliminated these false positives, with Fusion 2 demonstrating the highest total accuracy.

Table I compares detection accuracy of the four methods when run on the Section IIIb test set, which included 301 half second instances. All four techniques detected the 6 immobilized periods. The SVM method detected immobilized the quickest followed by Fusion 2; however in some cases the SVM method detected immobilized before the vehicle was stopped, accounting for the 3 false positives. Both fusion techniques eliminated these false positives, with Fusion 2 demonstrating the highest total accuracy.

Although not shown in Fig. 6 or Table I, it is possible that a detector could falsely label an instance with high enough confidence for the point to be mislabeled by Fusion 2 but not Fusion 1. Fusion 1 should therefore be more robust to false

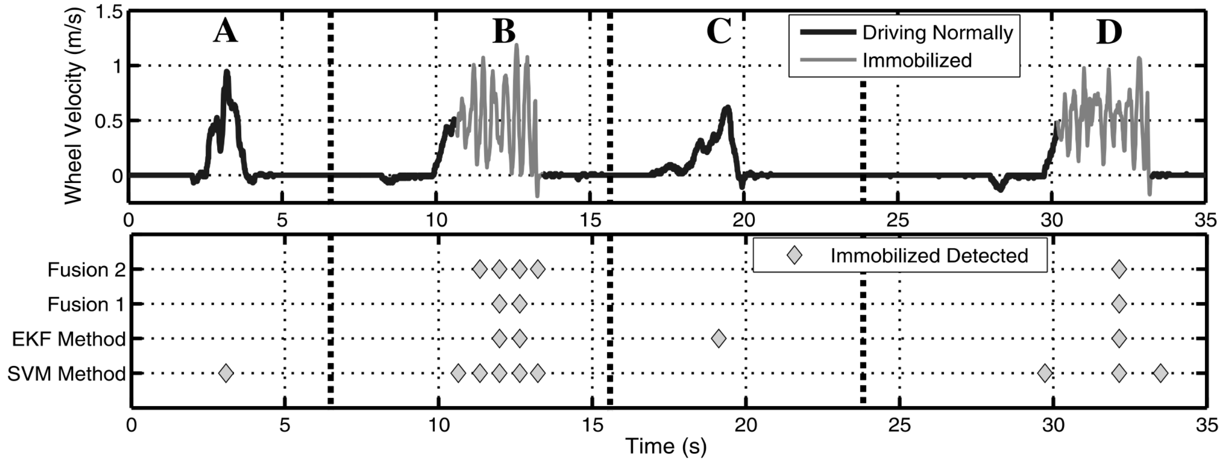

Fig. 6. Detector fusion results. Wheel velocity is effective linear velocity at wheel radius.

positives. If both detectors mislabel an instance, it will be mislabeled under both fusion techniques.

V. Conclusions

A signal recognition based approach to detecting robot immobilization has been proposed and experimentally validated. Four distinguishing features have been proposed for the algorithm requiring an IMU and (optionally) wheel encoders or tachometers, both common sensors on outdoor mobile robots. Future work will explore the effects of SVM kernel selection and robot speed and configuration on algorithm performance and test the algorithm on alternate terrain types and situations.

Two simple detector fusion techniques have been proposed to combine the output of the classifier-based immobilization detector and a dynamic model-based detector. Fusion 1 resulted in a conservative approach to minimize false detections, while Fusion 2 provided faster performance while potentially allowing more false detections. Both fusion techniques were shown to eliminate false immobilization detections on the experimental data set and increase overall accuracy compared to each individual detector. Future work will explore using various fusion techniques to combine more than two detectors and for alternative applications including terrain classification.

Acknowledgment

The authors would like to thank Andrew Lookingbill at Stanford University for his help with the experiments.

References

[1] J. Borenstein, H. R. Everett, L. Feng, *"Where am I?" Sensors and Methods for Mobile Robot Positioning*, Univ. of Michigan, April 1996, http://www-personal.umich.edu/~johannb/shared/pos96rep.pdf.

[2] Y Fuke, E. Krotkov, "Dead reckoning for a lunar rover on uneven terrain," *Proc. of 1996 IEEE Intl. Conf. on Robotics and Automation*.

[3] K. Iagnemma, S. Dubowsky, "Traction Control of Wheeled Robotic Vehicles with Application to Planetary Rovers." *Intl Journal of Robotics Research*, vol. 23, no. 10, pp. 1029-1040, October 2004.

[4] F. Gustafsson, "Slip-based tire-road friction estimation," *Automatica*, vol. 33, no. 6, pp. 1087-1099, 1997.

[5] L. Ojeda, G. Reina, J. Borenstein, "Experimental results from FLEXnav: An expert rule-based dead-reckoning system for Mars rovers," *IEEE Aerospace Conference 2004*, March 2004.

[6] L. Ojeda, D. Cruz, G. Reina, J. Borenstein, "Current-based slippage detection and odometry correction for mobile robots and planetary rovers," IEEE Transactions on Robotics, vol. 22, no. 2, April 2006.

[7] H. Tan, Y. Chin, "Vehicle antilock braking and traction control: a theoretical study," *Intl. Jrnl of Systems Science*, vol. 23, no. 3, 1992.

[8] *Automotive Handbook*, 5th ed., Robert Bosch GmbH, Germany, 2000.

[9] S. Sukkarieh, E. Nebot, H. Durrant-Whyte, "A high integrity IMU/GPS navigation loop for autonomous land vehicle applications," *IEEE Trans. on Robotics and Automation*, vol. 15, no. 3, June 1999.

[10] B. Hofmann-Wellenhof, H. Lichtenegger, J. Collins, *Global Positioning System: Theory and Practice*, 5th ed., Springer-Verlag, Wien, 2001.

[11] *GPS 16/17 Series Technical Specifications*, Garmin Intl., Inc., 2005.

[12] C. Brooks, K. Iagnemma, "Vibration-based terrain classification for planetary exploration rovers," *IEEE Trans. on Robotics*, vol. 21, no. 6, December 2005.

[13] H. Dahlkamp, A. Kaehler, D. Stavens, S. Thrun, G. Bradski, "Self-supervised monocular road detection in desert terrain," *Proc. of the Robotics Science and Systems Conference*, Philadelphia, PA, 2006.

[14] A. Ganapathiraju, J. Hamaker, J. Picone, "Applications of support vector machines to speech recognition," *IEEE Trans. on Signal Processing*, vol. 52, no. 8, August 2004.

[15] C. Ward, K. Iagnemma, "Model-based wheel slip detection for outdoor mobile robots," *2007 IEEE Int. Conf. on Robotics and Automation*.

[16] C.-C. Chang, C.-J. Lin, "LIBSVM: a library for support vector machines," 2001. http://www.csie.ntu.edu.tw/~cjlin/libsvm

[17] T. Hastie, R. Tibshirani, J. Friedman, *The elements of statistical learning: data mining, inference, and prediction*, Springer, NY, 2001.

[18] C.-W. Hsu, C.-C. Chang, C.-J. Lin, "A practical guide to support vector classification," Accessed: August 2006, Available: http://www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf.

[19] "Learning Applied to Ground Robots," accessed July 24, 2006, http://www.darpa.mil/ipto/Programs/lagr/vision.htm.

[20] J. Y. Wong, *Theory of Ground Vehicles*, 3rd ed., Wiley, 2001.

Table I. Comparison of accuracy of detection and fusion techniques on Section IIIb test set.

	SVM Method	EKF Method	Fusion 1	Fusion 2
Total Accuracy:	94.7%	95.7%	91.7%	98.0%
# False Positives:	3	0	0	0
# False Negatives:	13	13	25	6