Monocular Rear-View Obstacle Detection Using Residual Flow

Jose Molineros, Shinko Y. Cheng, Yuri Owechko
HRL Laboratories, LLC
3011 Malibu Canyon Road
Malibu CA 90265
{jmmolineros,sycheng,yowechko}@hrl.com

*Dan Levi, †Wende Zhang
*GM Advanced Technology Center-Israel
†GM Research
{dan.levi, wende.zhang}@gm.com

Abstract— **We present a system for automatically detecting and localizing obstacles from a moving vehicle using a monocular wide angle camera. Our system was developed in the context of finding obstacles and particularly children when backing up. Camera viewpoint is transformed to a virtual bird-eye view. We developed a novel image registration algorithm to obtain ego-motion that in combination with variational dense optical flow outputs a residual motion map with respect to the ground. The residual motion map is used to identify and segment 3D and moving objects. Our main contribution is the feature-based image registration algorithm that is able to separate and obtain ground layer ego-motion accurately even in cases of ground covering only 20% of the image, outperforming RANSAC**.

I. INTRODUCTION

Many automotive accidents involve cars backing up, and hitting all types of objects [1]. Rear collision detection systems attempt to drastically reduce the risk of this type of accidents.

Existing backup warning systems based on sonar or radar sensors provide gross localization information only, not enough to accurately determine if the vehicle will collide with a an obstacle. Ranges go up to 5 [m]. Reaction times for humans are between .25 [s] and .8 [s] before brakes are even applied [2]. A car backing up at 15 [mi./h] will travel 5 [m] in .75 [s], which might be too close for comfort.

Cameras on the other hand offer the capability to extend range and process information to determine the nature of the alarm and obstacles, as opposed to the limited information gathered by radar or ultrasonic devices. Research work in rear obstacle detection using cameras include [3][4][5].

Previous systems for obstacle detection based on cameras sometimes utilize stereo and sometimes monocular input. Many utilize the idea of inverse perspective mapping (IPV) in polar histogram analysis, for example [5] and [6]. The paper in [6] uses stereo, IPV and polar histograms to detect close range obstacles. Work in [5] uses only one camera but coarsely uses polar histogram analysis due to not being able to obtain accurate ego motion. In [3] a system is presented that uses monocular input and ego-motion from odometry sensors. The authors in [4] present a system for backup aid that detects obstacles from three independent methods that are fused at the output stage. It assumes road boundaries are always within the video image and has some difficulty with false positives due to shadows not being eliminated if motion compensation is not accurate.

We introduce a method requiring only monocular camera input from a wide angle camera that is a valuable alternative to polar histogram analysis. A novel ground ego-motion recovery algorithm is developed and obstacle segmentation is based on state-of-the-art variational optical flow algorithms. We assume an urban setting with a ground layer that in a short range around the car is parallel to the car longitudinal axis.

The rest of the paper is organized as follows: section II describes our system. Section III describes its use in a child detection application. Section IV is implementation details and section V presents the conclusions.

II. SYSTEM OVERVIEW

Our system termed VOFOD (variational optical flow obstacle detection) uses a combination of methods that to the best of our knowledge is novel. By registering the ground plane in consecutive image frames and exploiting the obtained ego-motion we generate an optical flow map where areas corresponding to the ground layer are set to zero. This map provides a motion saliency mechanism that directs attention to areas in the image containing obstacles.

We change the video point-of-view to a virtual camera view looking straight down, termed bird-eye view, or BEV (figure 1). This allows us to run fast, accurate registration based on point features, and ultimately to obtain the ground motion vector. Known and fixed camera positioning with respect to the automobile body frame can be used to determine the distance from the vehicle to the base of a detected obstacle.

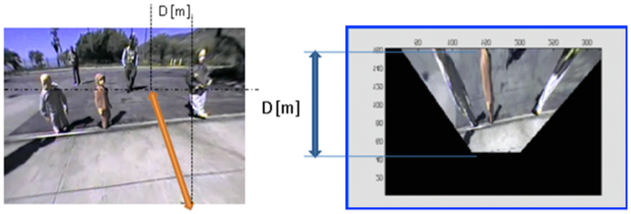

Fig. 1: (Left) An image taken from a camera rigidly mounted on a vehicle. (Right) Result of virtually rotating the camera view to bird-eye view.

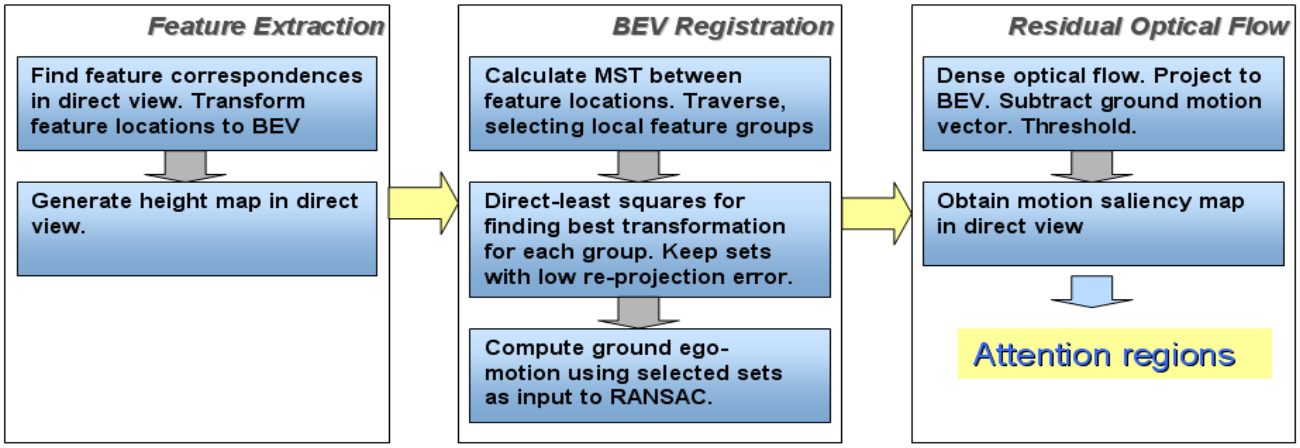
Feature Extraction
Find feature correspondences in direct view. Transform feature locations to BEV
Generate height map in direct view.
BEV Registration
Calculate MST between feature locations. Traverse, selecting local feature groups
Direct-least squares for finding best transformation for each group. Keep sets with low re-projection error.
Compute ground ego-motion using selected sets as input to RANSAC.
Residual Optical Flow
Dense optical flow. Project to BEV. Subtract ground motion vector. Threshold.
Obtain motion saliency map in direct view
Attention regions

Fig. 2. System block diagram. There are three defined stages: feature correspondences, BEV registration and ground ego-motion estimation, and residual optical flow for obstacle detection.

We obtain optical flow information for each point in the image, and use the knowledge about ground motion to select the flow vectors corresponding to objects that are not part of the ground plane. We accomplish this by projecting the optical flow vectors to BEV. The obtained ground motion vector is then subtracted from projected dense optical flow, therefore selecting moving areas not belonging to the ground plane.

Once in BEV, points at a height with respect to the ground plane will have a different difference in deformation than points in the ground plane when looking at consecutive images. Optical flow vectors in BEV are easily classified into ground/non-ground, and by assuming a planar ground, classified into obstacles and non-obstacles.

The algorithm consists of three stages, where each step utilizes the results from the previous stage (figure 2):

--*Stage 1*: Finding feature correspondences in consecutive images. Transforming point feature locations to a downward camera view (BEV). A map is also generated that gives the metric distance from the camera to a particular pixel location in the image.

--*Stage 2:* Classifying feature points into ground and non-ground points by means of our registration algoritm. Obtaining three-parameter homography, or ground ego-motion, in BEV.

--*Stage 3*: Estimating an 8-parameter homography transformation between consecutive frames, in original camera viewpoint. Obtaining residual optical flow and segmenting obstacles. Outputting metric distances from vehicle to obstacles.

A. Stage 1

Stage 1 consists of an algorithm to extract image feature points and virtually transform them to a virtual camera viewpoint looking straight down (BEV). Figure 1 shows an example image after virtual camera rotation. Changing viewpoint simplifies the problem of estimating ego-motion by reducing the number of unknown parameters from eight to three. This simplification allows us to estimate orthographic transformation parameters directly, with corresponding benefits in terms of registration accuracy and speed (example: figure 5).

Suitability to obstacle detection applications of a particular feature detector/descriptor will depend on video quality, feature matching accuracy and computation speed. In our application scenario, images are captured by a fish-eye rear production camera with relatively low video quality. Another important fact to be taken into account is that concrete surfaces tend to have low feature density, making it difficult to extract enough quality features for matching and registration. A top down virtual camera viewpoint exploits the approximation of vehicle motion on the ground as planar motion and offers the advantages of no ambiguity between rotational and translational ego-motion parameters and reliability in estimation of transformation parameters due to the linearity of image motion.

Point feature types found in literature include SIFT, SURF, Harris, KLT, and CenSurE. Perhaps not surprisingly, SIFT produces the best quality registration in terms of average re-projection error in our experiments. However, unless accelerated by hardware, it might be unsuitable for hard real-time applications. A second best is Center Surround features (CenSurE) [11], a feature type designed in the context of visual odometry in rough outdoor terrain over long periods. In our scenario, we found its reliability to be close to SIFT with speed improvements of four to five times in our experiments. It produces the best compromise between match quality and speed. In all of our video test sequences, both CenSurE and SIFT have produced enough quality correspondences in production fish-eye cameras currently installed in automobiles. Other detectors we tested, such as SURF and KLT, have not.

B. Stage 2

Our registration algorithm in this stage estimates orthographic transformation parameters, minimizing re-projection error and eliminating outliers. The main novelty of our obstacle detection approach lies in this registration algorithm.

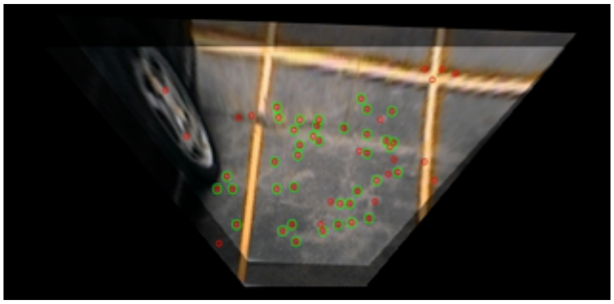

Fig. 3. Result of stage 2. Two bird-eye view images have been registered. The red dots are features in image 1. Green circles indicate features in image 2 that have been classified as ground.

Stage 2 takes as input BEV feature locations (in consecutive images) and their matched correspondences and calculates ground ego-motion. We classify features as 'ground points' vs. 'non-ground', where ground points are found by first finding image features lying on a plane parallel to the virtual camera plane and then refined by eliminating features lying on surfaces ‘at height’ with respect to the ground.

Robust ground plane determination is essential to the goal of obstacle detection based on camera motion. Methods to determine the ground plane include 2D dominant motion estimation and layer extraction. These methods can be top-down or bottom-up. Top-down approaches assume that the ground plane is dominant in the scene, or alternatively that the scene can be modeled with a few planar layers. In car backing up scenarios the ground plane might not be dominant and the rest of the scene is not necessarily well modeled with a few planar layers.

Bottom up approaches, in contrast, divide the image into small patches where a local patch transformation is calculated. These transformation calculations are then grouped together to form layers. In [7] the measurements are combined with a robust weighting scheme for global ego-motion determination and ground plane segmentation. A regularization step is later applied to recover small non-planar motions.

In [7] ground layer segmentation results look promising, but there is no information in terms of computation performance for real-time applications. Our own preliminary implementation of the scheme in [7] yielded performance and speed that convinced us point features are still the way to go.

We developed point feature-based registration scheme that makes use of BEV and does not assume a dominant ground layer. Our approach produces good quality registration even in cases of 80% of the image being background clutter and obstacles.

We utilize a two-tiered algorithm to obtain matches between two clusters of points according to an orthographic projection model. The two tiers are a Minimum-Spanning Tree (MST) [16] point selection algorithm, followed by RANSAC or one of its variants [8][9].

A feature selection strategy outputs a “bag” of candidate ground features, after traversing an MST generated between the point correspondences in the first image. We test each vertex plus its immediate connected neighbors, forming a local set, against the corresponding local set in the second image. A direct least-squares estimation technique [11] is used to compute the transformation parameters between local sets, as well as choose and keep sets with low average re-projection error. These local set of features are added to a global “bag” of candidate ground points.

After we traverse the MST, we use RANSAC to obtain the final global orthographic transformation between ground points. Although the MST stage is enough most of the time, local sets of points in surfaces (at height and parallel to the ground) will pass the least-squares re-projection error test. Therefore RANSAC is applied to eliminate these outlier sets.

MST allows us to exploit naturally occurring clusters of ground feature points and produces more accurate registration in the presence of higher number of outliers than RANSAC. It also avoids having to unnaturally segment the image into coarse blocks for analysis that a two-tiered RANSAC scheme would require. Exploiting the fact that ground features tend to cluster together becomes crucial in cases where it is difficult to localize repeatable feature points, such as concrete surfaces in roads and parking lots. Figure 3 shows the registration result of two bird-eye view

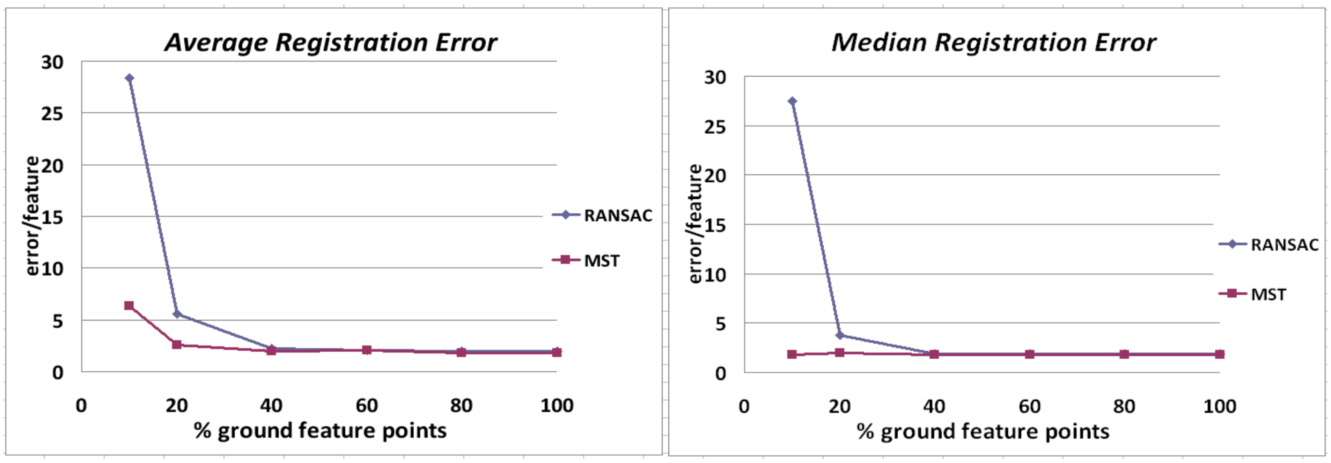

Fig. 4. Average Euclidean error per feature correspondence. RANSAC-based BEV registration (blue) vs. MST-based method (red). The *X*-axis represents the percentage of ‘ground points’.

frames after obtaining 2-D ground motion parameters: translation (Δx, Δy) and rotation Δθ.

In order to compare registration quality between MST and RANSAC, we tested the algorithms in simulated data. We generated a number of random points in one set, some belonging to ground class and others to non-ground class. Points in a second set correspond to the first set, transformed according to known rotation and translation. Uncertainty in position has been added to points in the second set. Ground points have an uncertainty of RND*5 pixels and non-ground points have an uncertainty of RND*50 pixels. The proportion of ground points is varied and re-projection error for both algorithms is obtained. We added some tendency of points of the same class to cluster together spatially.

Figures 4 compares quality of registration attained using MST vs. RANSAC [8]. The X-axis represents the percentage of true ground points in the set. We can see MST efficiently exploits the tendency of points of the same class to cluster together in order to register images with a minority of ground points.

Stage 2 registration algorithm provides a very noticeable improvement over other conventional methods for ground registration [15]. Figure 5 illustrates the quality improvement.

Description of stage 2 is as follows:

1) Generate a *minimum-spanning tree (MST) graph* between bird-eye view (BEV) point locations in one image.
2) Obtain groups of features that are spatially close, by traversing MST structure.
3) Classify clusters as ground/non-ground according to conformance to an orthographic transformation model. Classification is based on their average re-projection error [11].
4) Calculate 2-D ground motion using all selected ground point clusters. Global orthographic transformation is obtained by a RANSAC-style algorithm [9] to eliminate entire clusters that might have low average re-projection error but are not ground.
5) Classifying original feature points (CenSurE), obtained *in direct view*, into ground/non-ground, by their corresponding BEV re-projection error.
6) Obtain ground motion vector.

Once ground has been registered, obstacles can be detected by exploiting the mis-registration that objects at height or moving objects exhibit. We accomplish obstacle segmentation by obtaining a residual dense optical flow map in the stage we describe next.

C. *Stage 3*

Stage 3 produces a residual motion map corresponding to objects that are not part of the ground plane. When the camera is moving, detected obstacles can be stationary or moving. If the camera is stationary, only moving objects will be detected.

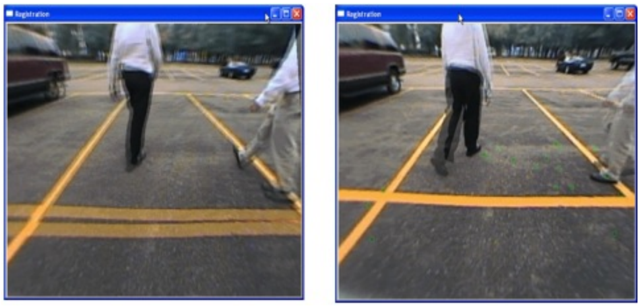

Fig. 5. Comparison of ground registration quality. (Left) Using a top-down method from literature to directly estimate 8-parameter homography. (Right) Using stage 2 algorithm, re-projected to normal camera view.

In order to find obstacles it is not enough to cluster point features that have been classified as non-ground. A type of more exhaustive image analysis is needed. We utilize dense optical flow for that purpose.

Optical flow as a good way to obtain regions of interest in the image, as well to provide an interest score to a particular pixel in the image. That interest score is the magnitude of the flow vector. Thus, we densely classify pixels as ground/non-ground.

Traditional dense optical flow algorithms are slow, sometimes unsuitable for real-time applications. We use a bidirectional multi-grid method for accelerating variational optical flow computations (VOF) [12]. This particular method is fast and accurate and enables the generation of the obstacle likelihood map.

Recently, variational numerical schemes have been used successfully to obtain high accuracy and real-time dense optical flow. The bidirectional multi-grid approach found in [12] can be considered a generalization of the traditional Horn-Schunck algorithm, but allowing for multi-grid schemes with non-dyadic grid hierarchies, as well as fast variational numerical methods for solving systems of equations. Near real-time performance and high accuracy are crucial for obtaining a residual motion map in the context of video obstacle segmentation.

Residual motion maps indicate motion different to ground plane motion. After obtaining optical flow vectors from two consecutive images, we project the vectors to BEV and subtract them from ground motion obtained in stage 2 (figure 6). Resulting BEV residual flow is used to obtain flow in direct view without barrel distortion. We use the mapping between BEV coordinates and direct view coordinates to create a binary mask in direct view that in combination with the optical flow map produces direct view residual motion (fig. 7, left).

As the ground plane has already been obtained, the camera is calibrated, and the relationship between bird-eye views and original views are known, distances (in feet or meters) from the vehicle to the base of obstacles are straightforward to calculate given that distances vary linearly with height (fig. 1, right).

While varying the point-of-view to estimate ground motion is a published idea, and VOF computation is a method found in literature as well, their combination and application to obstacle detection is new to the best of our knowledge. Our registration algorithm in combination with VOF methods provide a powerful, novel alternative (and perhaps complement) to other simpler real-time algorithms in automotive applications.

Calculation of dense optical flow and BEV ground ego-motion estimation are currently implemented as parallel threads. Once the ground motion vector is obtained, we merge the results to produce the residual motion map and obstacle segmentation.

In summary, stage 3 consists of:

1) Calculating dense variational optical flow for the image.
2) Transforming VOF to bird-eye view (BEV).
3) Subtracting BEV-VOF from ground plane motion vector
4) Determining VOF of obstacle pixels in direct view.

Figure 7 (left) shows the color-coded residual motion map obtained from stage 3. The distance in meters from the camera to the corresponding position in the ground plane is known, and varies according to *Y* coordinate in the undistorted image. When looking for obstacles that have a expected height range (in meters for example), this information allows us to estimate the size in pixels such an obstacle (with a base at that *Y* pixel position) would have in the image.

The optical flow map in figure 7 was obtained from consecutive images (left of figure 6 and figure 8). Color intensity represents magnitude of the optical flow vector, whereas the color itself represents vector direction. This image itself can be used as a saliency map, or it can be combined with other static-image saliency algorithms [13] to indicate regions of interest. In our example application, described next, the magnitude of the optical flow vector at a particular pixel, normalized to a maximum value of 1.0, represents a likelihood of it belonging to a obstacle.

III. APPLICATION: CHILD DETECTION

VOFOD is used in combination with a state-of-the-art parts-based person classifier termed FSM (feature synthesis [14]) to detect children in arbitrary poses. A pedestrian classifier will take as input a window of particular proportions centered around a given location in the image and output a probability of it containing a human. Processing times for state-of-the-art parts-based classifiers is still far from real-time unless the number of windows to be considered as input is drastically reduced. VOFOD is used to reduce the typical number of input windows from >50K to less than 500 in our test sequences.

Figure 7 is the result of exhaustively searching for windows that may contain pedestrians of height approximately expected of children. The boxes were generated with a fixed aspect ratio, and their height is adjusted according to *Y*-coordinate to fall within an acceptable height range. We sum the values of the VOFOD saliency map inside each box to obtain a likelihood of it containing a child. Only boxes with a score above a predefined threshold are colored. Lighter colors represent higher box scores.

Our experiments in child detection are detailed in [13]. Although it is making great strides, performance of the state-

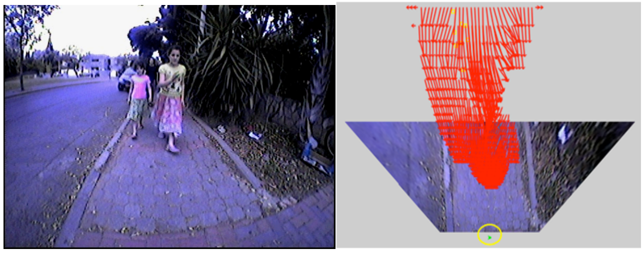

Fig. 6. (Left) Residual optical flow. (Right) Optical flow obtained in BEV, subtracted from ground plane motion. Ground plane motion is shown as a small green arrow enclosed in the yellow circle.

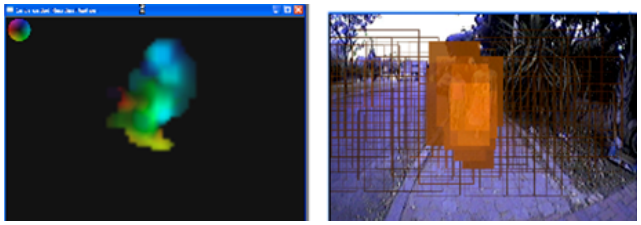

Fig. 7. (Left) Residual optical flow map. Color represents flow vector direction and pixel intensity represents flow magnitude. (Right) Using the residual optical flow map for child detection. Boxes were generated according to expected child height and scored according to average optical flow intensity. Only boxes scoring above a threshold are colored.

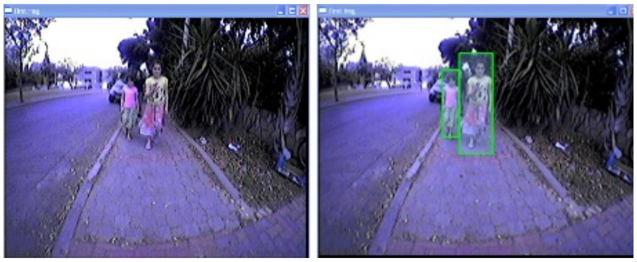

Fig. 8. Children detection using a moving camera. (Left) Input video image. (Right) detected children.

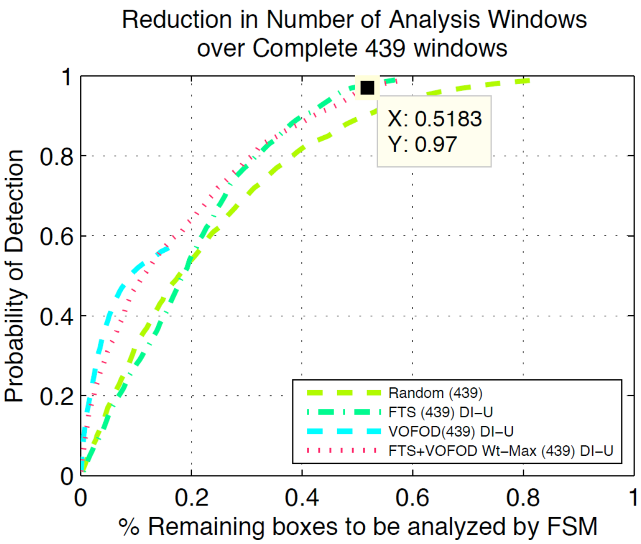

Fig. 9. Probability of detection as a function of the percentage of boxes remaining, for analysis with a pedestrian classifier (FSM). FTS = A static image saliency algorithm. VOFOD excels at low probability of detection, and in combination with FTS it extends range to high probability of detection.

of-the-art in human arbitrary pose detection still needs to improve. At the same time, in a warning application it is important to eliminate false positives as they might desensitize the driver to ignore alerts. Under such conditions, it might be desirable to operate under low probability of detections but very low false positives. VOFOD excels under such conditions (figure 9). Meanwhile, obstacles of every type are still detected.

Our video sequences include children at all distance ranges. Nearly static children at close and mid ranges in consecutive images can be detected by residual optical flow from a moving platform due to parallax, but static children far away will not be detected.

By using VOFOD in combination with an image-based saliency algorithm, such as frequency-tuned saliency (FTS [17]), we can operate in high probability of child detection as well, although with a lower reduction in the number of boxes. We refer the reader to [13] for details.

IV. IMPLEMENTATION

Although we feel the VOFOD implementation can be greatly optimized, perhaps up to an order of magnitude, currently the system has been implemented in separate threads for different modules. Running times, in C++ for the main modules currently are 3.3 FPS for stage 1 + 2 combined, in 640 x 480 resolution, and 2.5 FPS for stage 3 in 320 x 240 resolution. Timings are in an Intel Xeon CPU W3550 @ 3.07 MHz.

Many optimizations are possible in our proof-of-concept code, including avoiding processing in certain parts of the image and making the code more compact and efficient. The slowest components are point feature extraction and dense optical flow. The complete system runs at video rate by processing only two or three pairs of consecutive images per second of video.

V. Conclusion

We have introduced an obstacle detection system based on bird-eye view registration for estimating ground motion and variational dense optical flow for detecting the obstacles. In ego-motion estimation, our approach of using an MST for ground point selection resulted in acceptable separation of ground vs. non-ground regions in cases with very few ground points, which are the instances in which RANSAC by itself tends to fail. This in turn allows for better ground registration and ego-motion estimation. Fast variational optical flow and its projection to bird-eye view allow us to obtain residual motion to segment obstacles by analysis of a residual motion map.

One example application of our algorithm is as an attention mechanism for child detection. Highlighted areas can be later processed with a parts-based classifier to identify children. Utilization of this attention mechanism greatly reduces computation time allowing for real-time performance.

References

[1] http://www.osha.gov/dcsp/success_stories/compliance_assistance/motor_vehicle_case_study.html

[2] Sens, M. J., Cheng, P. H., Wiechel, J. F. and Guenther, D. A. (1989), "Perception/Reaction Tim Values for Accident Reconstruction", SAE 890732, Society of Automotive Engineers, pp 79-94

[3] Vestri C., et al., "Real-Time Monocular 3D Vision System", 16th World Congree on Intelligent Transport Systems, Stockholm, 2009

[4] Ma G., et al., "A Real-Time Rear View Camera Based Obstacle Detection", 12th IEEE International Conference on Intelligent Transportation Systems, pp 1-6, September 2009

[5] Yankun Z., Chunyang H., and Norman W., "A Single Camera Based Rear Obstacle Detection System", Proc. IEEE Intelligent Vehicles Symposium, 2011

[6] Broggi A., Medici P., and Porta P. P., "StereoBox: A Robust and Efficient Solution for Automotive Short-Range Obstacle Detection", EURASIP Journal on Embedded Systems, 2007

[7] Ke Q. and Kanade T., "Transforming Camera Geometry to a Virtual Downward-Looking Camera" Robust Ego Motion Estimation and Ground-Layer Detection", Proceeding of the IEEE Computer Vision and Pattern Recognition (CVPR '03), 2003

[8] Fischer M. A. and Bolles, R. C., "Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated cartography", Comm. Of the ACM, vol. 24, pp 381-396, 1981

[9] Torr, P.H.S. and Zisserman A., "MLESAC: A New Robust Estimator with Application to Estimating Image Geometry", IEEE Conference on Computer Vision and Image Understanding (CVPR '00), 2000

[10] Konolige K., Agrawal M., and Sola J., "Large Scale Visual Odometry for Rough Terrain", In Proc. International Symposium on Robotics Research, 2007

[11] Umeyama S., "Least-squares Estimation of Transformation Parameters Between Two Point Patterns," IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), vol.13, no.4, pp. 376-380, 1991

[12] Bruhn A et al., "Variational Optical Flow Computation in Real Time", IEEE Trans. On Image Processing, vol. 14, no. 5, May 2005

[13] Cheng, S. et al., "Parts-based Object Recognition Seeded by Frequency Tuned Saliency for Child Detection in Active Safety", In Submission

[14] Bar-Hillel A., Levi D., and Krupka E., "Part-based Feature Synthesis for Human Detection", Computer Vision-ECCV, 2010

[15] Zhou, J and Li, B., "Homography-based Ground Detection for a Mobile Robot Platform Using a Single Camera", IEEE Conference on Robotics and Automation (ICRA '06), 2006

[16] Chazelle, B, "A Minimum Spanning Tree Algorithm with Inverse-Ackermann Type Complexity", Journal of the Association for Computing Machinery 47 (6): 1028–1047, 2000

[17] Achanta R. et al., "Frequency-tuned Salient Region Detection", in IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 1597-1604, 2009