Multi-modal Obstacle Detection and Evaluation of Occupancy Grid Mapping in Agriculture

Mikkel Kragh[a,*], Peter Christiansen[a,*], Timo Korthals[b,*], Thorsten Jungeblut[b], Henrik Karstoft[a], Rasmus N. Jørgensen[a]

[a] Department of Engineering, Aarhus University, Finlandsgade 22, DK-8200 Aarhus N, Denmark

[b] Cognitronics & Sensor Systems, Bielefeld University, Inspiration 1, D-33619 Bielefeld, Germany

* Corresponding author. Email: {mkha, pech}@eng.au.dk, tkorthals@cit-ec.uni-bielefeld.de

Abstract

In recent years, mapping and automation has been increasingly investigated and applied in precision agriculture. The ultimate goal of this development is to apply autonomous vehicles operating efficiently without any human intervention. Such autonomous operation imposes severe safety hazards, demanding accurate and robust risk detection, and avoidance systems. It is unlikely that one sensor can single-handedly guarantee this, and therefore multiple sensing modalities are often combined in order to increase detection performance and introduce redundancy. In this paper, we present a global mapping approach utilizing diverse sensor technologies to achieve a uniform obstacle interpretation of the environment. Using occupancy grid maps, we fuse information from a monocular color camera, a RADAR, and a LIDAR in combination with IMU-assisted GPS-positioning. For each sensor, we present detection algorithms, mapping from raw sensor data to a 2D grid-based obstacle interpretation of the environment. These are then fused temporally with the occupancy grid algorithm, and afterwards spatially in a competitive and complementary way to produce a combined global obstacle map. The method is evaluated on an extensive dataset recorded at Research Centre Foulum, Denmark, in June 2015. The dataset comprises sensor data from a tractor-mounted recording system in a grass mowing scenario with various obstacles. A ground truth map has been obtained with a mapping drone. Results show promising obstacle detection capabilities and an increase in performance when fusing information across sensor modalities and layers. The proposed mapping framework is able to fuse a vast amount of information across a diverse sensor set, using an efficient and novel approach for obstacle detection in agriculture.

Keywords: Multi-modal Sensor Fusion, Obstacle Detection, Occupancy Grid Mapping, Precision Farming, Agriculture

1. Introduction

The application of robots or vehicles operating autonomously in agricultural fields demands extreme perception capabilities of the safety system. It is unlikely that a single perception sensor is capable of ensuring this safety alone, and thus multiple sensor technologies must be combined to provide accurate and robust risk detection and avoidance. These sensors might operate in different coordinate systems with different representations. For instance, a LIDAR operates in 3D cartesian coordinates, an automotive RADAR operates in 2D polar coordinates, and cameras operate in projective spaces of 2D pixel coordinates. Sensor fusion can be handled on various abstraction levels such as data-, feature- or decision-level, but all methods require a mapping to a common representation. One such fusion algorithm on feature-level is occupancy grid maps (Elfes 1990). In 2D, they represent a global map of the environment and are generated from inverse sensor models (ISMs). An ISM is associated with a specific sensor and includes a detection algorithm of a certain feature (e.g. “vehicle”, “human”, “field”, “ground”) and a mapping from sensor data to a local 2D grid in the vehicle frame.

In research on automotive vehicles, 2D grid mapping is widely applied for fusing information across sensing modalities, providing a simple yet efficient framework (Winner 2015). In agricultural environments, a few applications with grid mapping have been proposed as well (Reina and Milella 2012; Ahtiainen et al. 2015). However, these only use a single or two sensing modalities, and thus do not provide a full evaluation of the potential of occupancy grid mapping.

In this paper, we present a global mapping approach utilizing simultaneous information from a monocular color camera, a thermal camera, a RADAR, and a LIDAR in combination with IMU-assisted GPS-positioning. For each of the sensors, we present detection algorithms, mapping from raw sensor data to a 2D grid-based obstacle interpretation of the environment. These grids represent multiple obstacle layers (“human”, “object”, “vegetation”, etc.) and are updated temporally using the occupancy grid algorithm. Finally, they are fused spatially across layers and sensor modalities using competitive and complementary fusion.

2. Materials and Methods

2.1. Setup

A variety of sensor modalities and corresponding detection algorithms are used to ensure detection and provide redundancy for all relevant obstacle types. A Velodyne HDL-32E LIDAR (laser range scanner) is used for long range depth estimation and is robust towards changes in illumination and weather. A Delphi ESR automotive RADAR is used for mid and long range depth and velocity estimation, and is even more robust towards changes in illumination and

weather than the LIDAR. A Logitech C920 color camera is used to detect and distinguish between different obstacle types, but is significantly more sensitive towards changes in illumination. Finally, a thermal camera is useful for capturing heat radiation from humans and animals. However, since only static, non-living obstacles are present in the dataset, this sensor is excluded from the paper. Together, the sensors both complement and overlap each other in terms of detection capabilities and robustness. A Vectornav VN-100 Inertial Measurement Unit (IMU) and a Trimble AG GPS361 Real Time Kinematic (RTK) GPS unit are used for pose estimation. Offline calibration is performed by hand by estimating extrinsic parameters of sensor positions. The specific sensor platform used for the experiments is presented and explained in detail in a previous paper (Christiansen et al. 2015).

2.2. Detection Algorithms

In the following sections, the algorithms used to produce classifications and their conversions to ISMs are described.

2.2.1. LIDAR

A single LIDAR scan provides a 3D point cloud consisting of depth measurements distributed 360° horizontally around the vehicle. For each point, we calculate 13 features using statistics from a local neighborhood (Kragh, Jørgensen, and Pedersen 2015). These features describe the height, shape, orientation and reflectance of the structure and help distinguish between points representing three classes: “ground”, “vegetation”, and “object”. A Support Vector Machine (SVM) classifier with probability estimates (Wu, Lin, and Weng 2004) is then trained to classify individual points into these classes. Figure 1 (left) shows an example of pseudo-colored probability estimates of the “object” class.

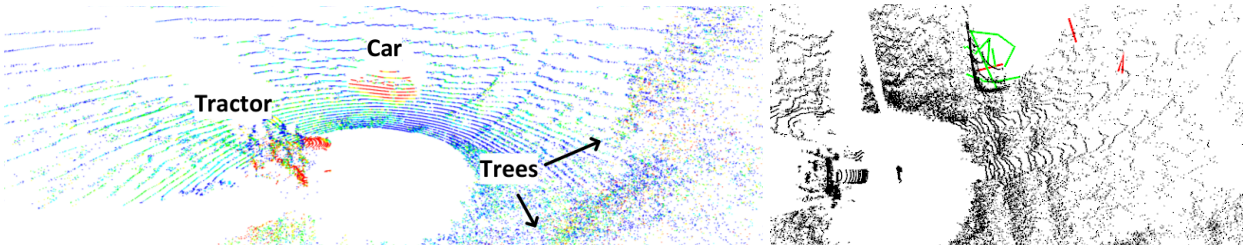

Figure 1. Left: Point cloud with pseudo-colored probability estimates of “object” class illustrating low (blue) and high (red) probabilities. Right: RADAR tracks overlaid on point cloud. Green are confirmed tracks and red are unconfirmed.

2.2.2 RADAR

The automotive RADAR combines mid- and long-range functionality simultaneously, so that it can detect close-distance objects with a horizontal field of view (FOV) of ±45° and far-distance objects with a narrow FOV of ±10°. The RADAR itself provides a processed list of up to 32 tracked objects, each with an angle and a range. However, most of these represent internal noise in the RADAR and therefore need to be processed further. For that, we apply the Kuhn-Munkres assignment algorithm (KMA), tracking detections from subsequent frames (Munkres 1957). Only detections that are less than 2 m apart from one frame to the next are associated. A track i is described by its current position and its track length L_i and is confirmed when $L_i \geq L_{min} = 3$. All confirmed tracks are then converted to detection probabilities:

$$P_{radar,i} = \frac{L_i - L_{min}}{L_i}$$

2.2.3 Color Camera

For the color camera, we apply three detection algorithms; Locally Decorrelated Channel Features for Pedestrian detection (PED) (Nam, Dollár, and Han 2014), You Only Look Once (YOLO) (Redmon et al. 2016), and Fully Convolutional Network for Semantic Segmentation (SS) (Long et al. 2015).

PED is a state-of-the-art pedestrian detector trained on the INRIA dataset (Dalal and Triggs 2005). PED uses three color and seven edge feature channels followed by a local decorrelation step creating 40 decorrelated feature channels. The algorithm uses an AdaBoost (Freund and Schapire 1996) based classifier and detects humans at multiple locations and scales using a speed efficient multiscale sliding window approach.

YOLO is a deep convolutional neural network (CNN) for object detection trained on 20 object classes on the Pascal Visual Object Classes (VOC) dataset (Everingham, Eslami, and Gool 2013). In this work, the 20 objects are mapped to three object classes: “human”, “vehicle”, and “unknown”.

In agriculture, elements such as the field and shelterbelts cannot naturally be delimited by a bounding box as normally provided by object detection algorithms. SS is a semantic segmentation method, meaning that each pixel in the image is classified as an object class. The algorithm is trained to recognize 60 object classes in the PASCAL-Context dataset (Mottaghi et al. 2014). As described in (Christiansen et al. 2016), these element classes can be remapped to a few agricultural classes. In this work, the classes are remapped to “unknown”, “grass”, “ground”, “human”, “shelterbelt”, “vehicle”, and “water”. An example of the outputs from the algorithms described above is presented in two cropped images in Figure 2.

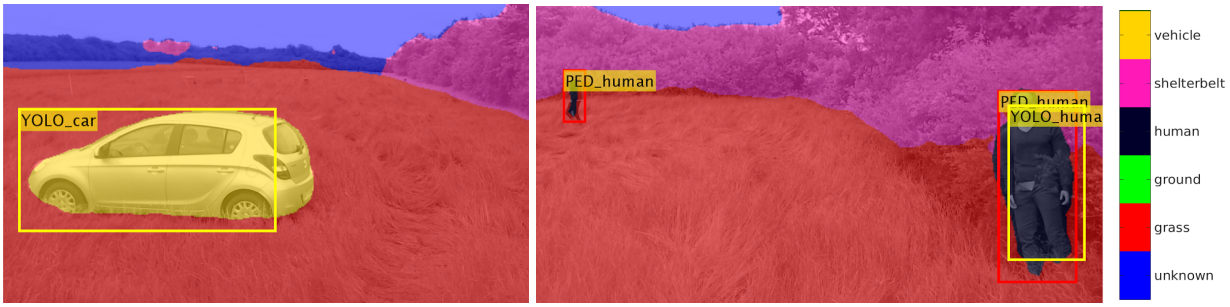

Figure 2. Example output of camera algorithms. PED detects both humans. YOLO is able to detect the vehicle and a human, but fails to detect the more distant human. SS detects both humans, the car, sky, ground and most of the shelterbelt. However, SS fails to detect the shelterbelt at far distance and around the human.

PED and YOLO algorithms output bounding box coordinates that are converted to a new image for each object class with a rectangle filled with a confidence measure of a detection. SS outputs an image for each object class, where each pixel contains a confidence measure of classification.

2.3. Mapping

Within this publication, two challenges are faced by mapping the algorithms' detections into a map representation of the vehicle's environment: First, by locating and mapping the detections into a map, evaluation against a ground truth map is easily applicable. Second, the map representation serves as the common way of fusing detections of a single algorithm temporally, and spatially across different modalities. A technique which suits these requirements is the Occupancy Grid Mapping (OGM).

2.3.1. Occupancy Grid Mapping

Two-dimensional occupancy grids were originally introduced by Elfes (Elfes 1990). In this representation, the environment is subdivided into a regular array or a grid of rectangular cells. The resolution of the environment representation directly depends on the size of the cells. In addition to this discretization of space, a probabilistic measure of occupancy is associated with each cell. This measure takes on any real number in the interval [0, 1] and describes one of the two possible cell states: occupied or unoccupied. An occupancy probability of 0 means definitely unoccupied space, and a probability of 1 means definitely occupied space. A value of 0.5 refers to an unknown state of occupancy.

The occupancy grid is an efficient approach for representing uncertainty, fusing multiple sensor measurements, and to incorporate different sensor models (Winner 2015). To learn an occupancy grid M given sensor information z, different update rules exist (Hähnel 2004). For our approach, we use the Bayesian update rule which is applied to every cell $m \in M$ as follows: Given the positions x_t of the vehicle at each point in time t, suppose $x_{1:t} = x_1, ..., x_t$ are the positions of the vehicle at the individual steps in time, and $z_{1:t} = z_1, ..., z_t$ are the perceptions of the environment. Occupancy probability grids determine for each cell c of the grid the probability that this cell is occupied by an obstacle. Thus, occupancy probability grids seek to estimate

$$P(m|z_{1:T}, x_{1:T}) = \prod_{t=1}^{T} \frac{P(m|z_t, x_t)}{1-P(m|z_t, x_t)} = \prod_{t=1}^{T} Odds(m|z_t, x_t) .$$

This equation already describes the online capable, recursive update rule that populates the current measurement z_t to the grid, where $P(m|z_t, x_t)$ is the so called inverse sensor model (ISM). The ISM is used to update the OGM in a Bayesian framework, which deduces the occupancy probability of a cell, given the sensor information.

2.3.2. Inverse Sensor Modelling

The ISM implements the inverse measurement model, which deduces from the sensor measurement to the occupancy probability at the particular cell. It is commonly used for sensors with a planar sensor lobe oriented parallel to the ground. In that case, a quite simplistic model can be applied, e.g. for a laser range finder. Each cell m that is covered by the beam of the observation z and whose distance to the sensor is shorter than the measured one, is supposed to be unoccupied. The cell in which the beam ends (the measurement point) is supposed to be occupied, and everything behind is unknown (Stachniss 2009). For our implementation, however, the cameras, LIDAR, and RADAR are non-planar, as their sensor lobes are tilted. Every non-planar sensor, compared to planar operating sensors, can only be evaluated at the measurement point, and thus do not provide any information in front of the measurement. Each sensor-algorithm combination requires its own ISM, converting from the algorithm's output to a 2D measurement grid representation. For this, a geometric interpretation is needed in order to transform features from the sensor frame to the vehicle frame.

2.3.2.1 ISM for LIDAR

From the SVM classifier, a 3D point cloud with class probabilities is provided for each class: "ground", "vegetation",

and “object”. A 2D class probability grid is created for each class by projecting all points onto a locally estimated plane and averaging over class probabilities of points lying within a grid cell. From these class probability grids P^*_{class}, two ISM obstacle layers are produced: “object” and “vegetation”. Figure 4 (left) illustrates an example of the “object” layer. The calculation of the log odds ratio of “object” combines the probability of the cell m being an object and the cell not being ground:

$$\begin{aligned} logOdds(P_{object}(m)) &= logOdds(P^*_{object}(m)) + logOdds(1 - P^*_{ground}(m)) \\ &= log(P^*_{object}(m)) - log(1 - P^*_{object}(m)) + log(1 - P^*_{ground}(m)) - log(P^*_{ground}(m)) \end{aligned}$$

2.3.2.2 ISM for RADAR

An ISM obstacle layer “radar” is produced by converting all confirmed detections from polar to cartesian coordinates and averaging over detection probabilities of tracks lying within a grid cell. This provides a probability grid $P^*_{ground}(m)$. The calculation of the log odds ratio of “radar” for cell m is then given by:

$$logOdds(P_{radar}(m)) = logOdds(P^*_{radar}(m)) = log(P^*_{radar}(m)) - log(1 - P^*_{radar}(m))$$

2.3.2.3 ISM for Camera - Inverse Perspective Mapping

Within this chapter, the projection of a camera image onto a planar ground map is described. We assume a pinhole model for the camera, a constant transformation between the camera frame and the vehicle’s footprint, and a flat world. To calculate the pixel-wise transformation from the camera frame into the vehicle frame, the inverse perspective mapping introduced by (Bertozzi and Broggi 1996) is applied.

Because of the flat world assumption, the projection is ill-defined for any detection that does not reside on the ground level. Kohlbrecher bypasses this problem by assuming every detected object to be grounded (Kohlbrecher 2011). In this way, an occupancy grid is generated by traversing through every column of a detection image starting from the bottom. This creates a ray in the occupancy grid, starting at the sensor position towards the horizon. When a detection ($P > 0.5$) occurs along this ray, the given cell is mapped accordingly and all subsequent cells are mapped as unknown ($P = 0.5$).

In this work, a positive detection pixel is extended by the estimated depth of a given obstacle before mapping unknown pixels. Figure 3 illustrates an example of this procedure. In the center image, a positive detection (white blob) of a vehicle seen by the SS algorithm is shown along with the estimated horizon. At the right, the same image converted through inverse perspective mapping to an occupancy grid is visualized, showing how the vehicle is assumed to have a depth of 2 meters.

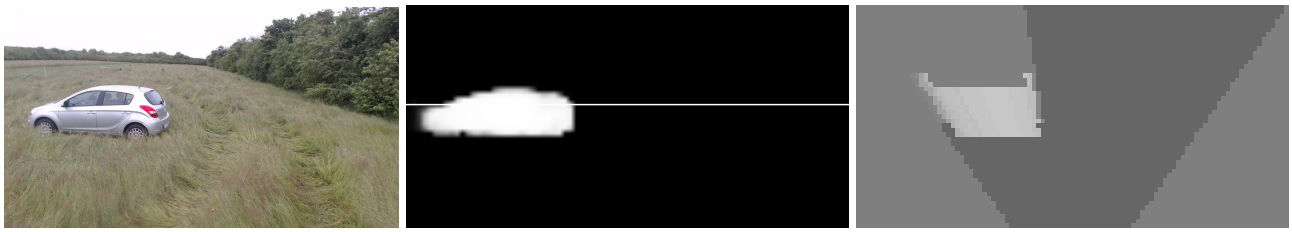

Figure 3. Left: Input image. Center: Horizon and detection of vehicle with semantic segmentation. Right: Inverse perspective mapping showing vehicle, FOV and unknown areas both behind the vehicle and outside the FOV.

2.3.3 Grid Map Representation

Different approaches exist for handling the residency of a map. For spatially limited applications, commonly one global map is used. To reduce the memory consumption, so called topo-metric maps are used as well, where the map size is reduced to e.g. rooms which are interconnected by a graph (Hähnel 2004). For automotive applications, temporary maps have proven their worth. They are build up by different sensors for a short time scenery of the environment (Winner 2015). This paper formulates an independent and global coordinate system which holds multiple two-dimensional grid maps for small areas. The whole area is divided into patches, and for each timestep only one patch, namely the Region-Of-Interest (ROI) is loaded. As depicted in Figure 4 (center and right), the patches overlap at the point where the vehicle crosses the border from the inner to outer ROI to the outer margin. If the vehicle passes this border, a new patch map is loaded. This provides two advantages: First, the memory consumption is reduced to a minimum and second, drift over multiple maps can be reduced by realigning all maps subsequently. Our solution can be compared to the patch map approach by (Konrad et al. 2011). Konrad aligns all maps vertically and horizontally with an overlap at their margins. Compared to this, our approach is able to respect former recorded data by transforming it into the upcoming ROI.

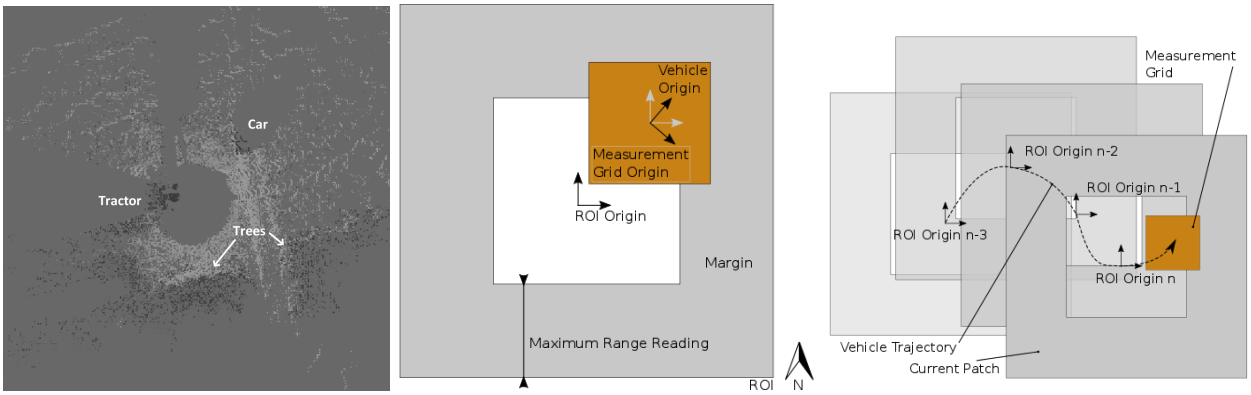

Figure 4. Left: Inverse sensor model as measurement grid of LIDAR for class "object". Center: Current patch as Region-Of-Interest. Right: Overlaid patches along a vehicle's trajectory

2.3.4 Mapping Uncertainty

Every ISM is influenced by the vehicle's pose uncertainty. This includes the latitude and longitude and the roll/pitch/yaw angles. Furthermore, because of the flat-plane assumption, the error caused by the assumed sensor height above the ground is respected as well. All uncertainties of every grid cell are modeled by a two-dimensional Gaussian function. To respect all Gaussian uncertainties in the ISM, all cell neighbours have to be taken into account. Thus, first an ISM without position uncertainties is created and then convolved by a Gaussian kernel $F \in \Re^{IxJ}$. To respect the fact that we deal with probabilities inside the ISM, we define the convolution function P^* for a single probability P of a cell $m_{x,y}$ at point (x,y) in the grid M as follows:

$$P^*(m_{x,y}) = logOdds^{-1} \sum_{i=x-I/2}^{x+I/2} \sum_{j=y-J/2}^{y+J/2} logOdds(F(i,j)(P(m_{i,j}) - 0.5) + 0.5)$$

3. Results and Discussion

3.1. Dataset

The evaluation of the grid mapping is performed on a dataset recorded at Research Centre Foulum, Denmark, in June 2015. The sensor platform described in section 2.1 is mounted in front of a tractor in a grass mowing scenario, recording over a 15 minute traversal in the field. Apart from naturally occurring elements in the field (shelterbelts, grass, ground, and water flooding), static obstacles (wells, a car, barrels, and adult and kid mannequin dolls) are placed and measured with precise GPS positions. The dataset also includes a single moving object (walking pedestrian). A ground truth map is generated by recording the field and obstacles with a Phantom 2 drone and manually annotating with per-pixel labeling. Figure 5 shows the orthophoto of the field with overlaid ground truth annotations.

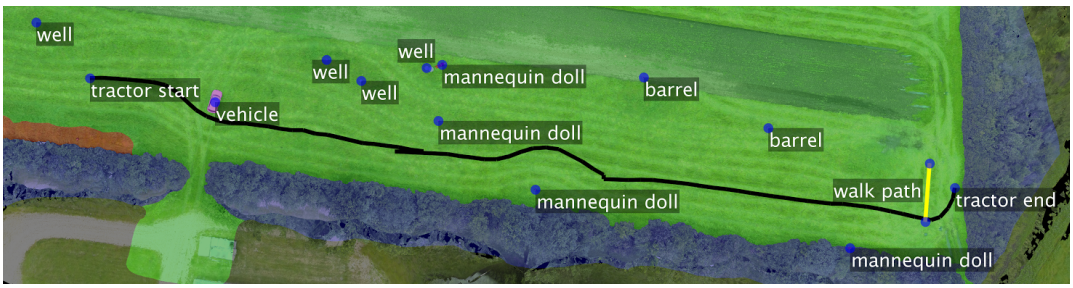

Figure 5. Orthophoto with static objects, tractor trajectory (black line) and human walk path (yellow line). An overlay shows the ground truth of vegetation (blue), ground (green) and non-traversable ground (red).

3.2. Evaluation and Results

To obtain the mapping results, the ISM methods are applied to their specific sensors to extract the measurement grids. To locate the measurement grid inside the current patch and globally, the extended Kalman filter by (Moore and Stouch 2016) is used, taking GPS, IMU, and GPS carrier measurements (Bevly and Cobb 2010) into account. As proposed in (Korthals, Skiba, and Krause 2016), multiple layers N of maps are needed to respect a diverse and heterogeneous sensor setup. This is used to overcome the drawback of the Bayesian update equation, which does not respect different sensor impacts or update rates. Thus, across each of the $N = 15$ sensor-algorithm-class sets, fusion is performed at a later stage by composing cell probabilities. In our implementation, two different fusion techniques are applied: First, the fusion based on a Superbayesian Independent Opinion Pool formula P_B (Pathak et al. 2007). It is applicable for the case when

separate occupancy grids with identical feature representations (e.g. set of maps for class "obstacle") are maintained. Second, a non-Bayesian fusion methods by taking the maximum P_M is applied to heterogeneous feature representations (e.g. set of maps for "vehicle" and "human"). It is worth mentioning that these fusion techniques are again cell-wise and therefore online applicable.

$$P_B(m) = \frac{\prod_N P_n(m)}{\prod_N P_n(m) + \prod_N (1 - P_n(m))}, \qquad P_M(m) = max_n\, P_n(m)$$

As evaluation metrics, precision, recall, F1 score, accuracy, True-Positive-Rate (TPR) and False-Positive-Rate (FPR) of the Receiver-Operator-Characteristic (ROC), and normalized entropy are calculated for all detected cells. For the given algorithms and sensors, the fusion and evaluation scores are not directly applicable. Even if the Bayesian framework allows the representation of the presence and absence of a feature, some algorithms do not make use of it. To name two examples, the LIDAR allows the deduction of free or occupied space based on its physical measurement principle. On the other hand, a camera based algorithm is fairly good for detecting the presence of a class, but easily fails in detecting the absence, due to e.g. a possible lack in the training set. Thus, the metrics recall, F1 score, TPR, and FPR can be calculated for LIDAR based detections, but not for camera and RADAR based detections. To give a better interpretation, the normalized entropy H_N of all true negative and true positive classified cells is used to calculate the remaining uncertainty normalized by a completely unknown map:

$$H(P(M)) = -\sum_{c \in M} P(c)\, log(P(c)) + (1 - P(c))\, log(1 - P(c)), \quad H_N(P(M)) = H(P(M))/H(P(M) \equiv 0.5)$$

This gives a quantitative value of the information gain among different setups where the range of the normalized entropy reaches from 0, meaning that there is no unknown space left, to 1, meaning the map is completely unknown.

Layers produced by the same sensor are fused by the maximum method to get a competitive fusion across algorithms, and the outcome of these layers is fused by the Superbayesian method to get a complementary fusion across different sensors as shown in Figure 6.

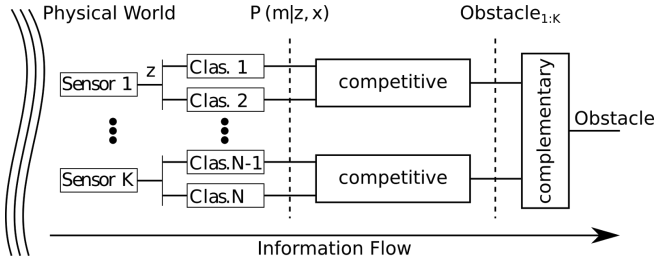

Figure 6. Fusion framework

Table 1. List of sensor setups. 1-3 use competitive fusion across classes, whereas 4-7 use complementary fusion.

Setup	Fusion	Sensors	Detection Algorithm	Input Classes	Output Classes
		Camera	SS	shelterbelt, human, vehicle	
1	Competitive	Camera	YOLO	human, vehicle	obstacle_C
		Camera	PED	human	
2	Competitive	LIDAR	SVM	object, vegetation	obstacle_L
3	Competitive	RADAR	KMA	radar	obstacle_R
4	Complementary	Camera, LIDAR	-	obstacle_C, obstacle_L	obstacle
5	Complementary	LIDAR, RADAR	-	obstacle_L, obstacle_R	obstacle
6	Complementary	Camera, RADAR	-	obstacle_C, obstacle_R	obstacle
7	Complementary	Camera, LIDAR, RADAR	-	obstacle_C, obstacle_L, obstacle_R	obstacle

Table 2. Evaluation scores for the different sensor setups (ill-defined scores omitted by "-")

Setup	Fusion	Precision	Recall	F1 score	Accuracy	TPR	FPR	Entropy
1	Maximum	0.889	-	-	0.889	-	-	0.984
2	Maximum	**0.897**	0.922	0.910	0.957	0.922	**0.0320**	0.821
3	Maximum	0.789	-	-	0.789	-	-	0.991
4	Superbayes	0.896	0.941	0.918	0.960	0.941	0.0342	0.819
5	Superbayes	0.889	0.944	0.916	0.960	0.944	0.0357	0.820
6	Superbayes	0.827	-	-	0.827	-	-	0.979
7	Superbayes	0.889	**0.958**	**0.922**	**0.961**	**0.958**	0.0376	**0.818**

For the evaluation, a constant map resolution of 10 cm per cell is used. To measure the impact of each sensor, all permutations of the sensors (camera, LIDAR, and RADAR) are performed as shown in Table 1. Particularly for the camera based detection, only classes representing objects are taken into account. For setup 1, 2, and 3, the fusion P_M is applied competitively, outputting "obstacle_C", "obstacle_L" and "obstacle_R" for camera, LIDAR, and RADAR respectively. These outputs are then fed into the complementary fusion P_B, outputting "obstacle". The results for all different setups are shown in Table 2. The first noticeable fact is the decrease of entropy for every complementary fusion. This shows, that with the introduction of new sources of information, the unknown area is reduced. Thus, the lowest entropy is evaluated for setup 7. The same is the case for the other scores, where setup 7 performs the best. The only exceptions arise for precision and FPR. For precision, the LIDAR performs better, but also has a bad recall resulting in the worst F1 score. This coincides with the FPR, as the number of misclassifications may rise with more sensors coming into play due to the fact, that in the evaluation scenario the sensor lobes do not fully overlap at all positions. Therefore, wrong classifications can not be corrected by sensor fusion.

As can be seen in Figure 7, misclassifications occur mainly at object borders. Due to the fact that the errors are evenly distributed around them, it can be assumed that they are caused by statistical errors from the sensors, the detection algorithm, or the vehicle's position uncertainty. To quantify this error, the standard deviations of all distinctive misclassified regions across obstacle borders are averaged with the result of $\sigma = 0.332$ m. In Figure 8, the final fused detection of all obstacle layers can be seen. To highlight one example, the car is almost perfectly detected with the only exception of the tail. Having in mind that the upper right edge of the car has not been seen by any sensor, the result of the fusion concept is even more convincing.

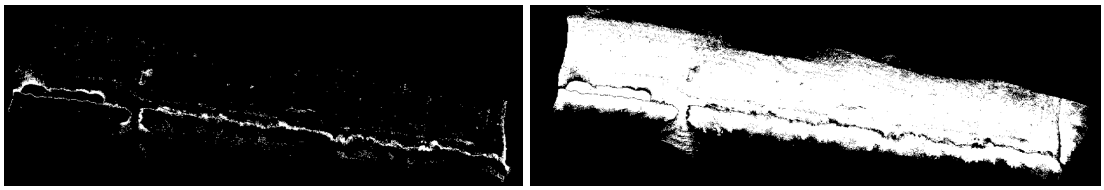

Figure 7. Binary mask created by setup 7 of false (left) and correct (right) classifications

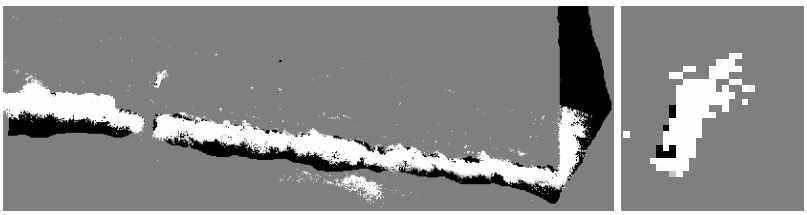

Figure 8. Left: Ground truth (black) with overlaid obstacle detection (white) by setup 7. Right: Magnified area of the car

4. **Conclusions**

In this work, we have presented a global mapping approach fusing information from a monocular color camera, a RADAR, and a LIDAR. For each sensor, we have introduced detection algorithms, mapping from raw sensor data to a number of 2D grid-based obstacle interpretations of the environment, such as "human", "vehicle", and "vegetation". These representations are first fused competitively for each sensor to provide a sensor-specific obstacle representation. Then, complementary fusion is used to fuse across sensor modalities, providing a final combined obstacle interpretation.

Based on data from a grass mowing scenario with various static obstacles, we have evaluated the proposed mapping approach for all combinations of sensors. We have shown that any combination of sensors performs better than the same sensors individually, and that we achieve a mapping accuracy for detected cells of 96% and an F1 score of 92%, when combining information across all three sensors. Future work will focus on introducing dynamic obstacles and training the fusion algorithm to weigh information from sensors and algorithms individually. Also, a more comprehensive evaluation from different fields and sensor setups is planned, investigating generalization performance of the proposed method.

Acknowledgements

This research is sponsored by the Innovation Fund Denmark as part of the project "SAFE - Safer Autonomous Farming Equipment" (project no. 16-2014-0).

This research and development project is funded by the German Federal Ministry of Education and Research (BMBF) within the Leading-Edge Cluster "Intelligent Technical Systems OstWestfalenLippe" (it's OWL) and managed by the Project Management Agency Karlsruhe (PTKA). The author is responsible for the contents of this publication.

References

Ahtiainen, J., T. Peynot, J. Saarinen, S. Scheding, and A. Visala. 2015. "Learned Ultra-Wideband RADAR Sensor Model for Augmented LIDAR-Based Traversability Mapping in Vegetated Environments." In *Information Fusion (Fusion), 2015 18th International Conference on*, 953–60.

Bertozzi, M., and A. Broggi. 1996. "Real-Time Lane and Obstacle Detection on the GOLD System." *Proceedings of Conference on Intelligent Vehicles*. doi:10.1109/IVS.1996.566380.

Bevly, D. M., and S. Cobb. 2010. *GNSS for Vehicle Control*. GNSS Technology and Applications Series. Artech House.

Christiansen, P., M. K. Hansen, K. A. Steen, H. Karstoft, and R. N. Jørgensen. 2015. "Advanced Sensor Platform for Human Detection and Protection in Autonomous Farming." In *Precision Agriculture '15*, 291–98.

Christiansen, P., R. Sørensen, S. Skovsen, C. D. Jæger, R. N. Jørgensen, H. Karstoft, and K. A. Steen. 2016. "Towards Autonomous Plant Production Using Fully Convolutional Neural Networks." In . Aarhus University.

Dalal, Navneet, and Bill Triggs. 2005. "Histograms of Oriented Gradients for Human Detection." In *Proceedings - 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005*. doi:10.1109/CVPR.2005.177.

Elfes, Alberto. 1990. "Occupancy Grids: A Stochastical Spatial Representation for Active Robot Perception." In *Proceedings of the Sixth Conference on Uncertainty in Artificial Intelligence*.

Everingham, Mark, Sma Eslami, and Luc Van Gool. 2013. "The Pascal Visual Object Classes Challenge–a Retrospective." *Homepages.Inf.Ed.Ac.Uk*. doi:10.1007/s11263-014-0733-5.

Freund, Yoav, and Robert E. Schapire. 1996. "Experiments with a New Boosting Algorithm." In *ICML*, 96:148–56.

Hähnel, Dirk. 2004. "Mapping with Mobile Robots."

Kohlbrecher, Stefan. 2011. "Grid-Based Occupancy Mapping and Automatic Gaze Control for Soccer Playing Humanoid Robots." *... Humanoid Soccer Robots ...*, no. October.

Konrad, Marcus, Magdalena Szczot, Florian Schüle, and Klaus Dietmayer. 2011. "Generic Grid Mapping for Road Course Estimation." *IEEE Intelligent Vehicles Symposium, Proceedings*, no. Iv: 851–56.

Korthals, Timo, Andreas Skiba, and Thilo Krause. 2016. "Evidenzkarten-Basierte Sensorfusion Zur Umfelderkennung Und Interpretation in Der Ernte." In *Informatik in Der Land-, Forst Und Ernährungswirtschaft*, 15–18.

Kragh, Mikkel, Rasmus N. Jørgensen, and Henrik Pedersen. 2015. "Object Detection and Terrain Classification in Agricultural Fields Using 3D Lidar Data." In *Computer Vision Systems*, 188–97. Lecture Notes in Computer Science. Springer International Publishing.

Long, Jonathan, Long Jonathan, Shelhamer Evan, and Darrell Trevor. 2015. "Fully Convolutional Networks for Semantic Segmentation." In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. doi:10.1109/cvpr.2015.7298965.

Moore, Thomas, and Daniel Stouch. 2016. "A Generalized Extended Kalman Filter Implementation for the Robot Operating System." In *Intelligent Autonomous Systems 13*, edited by E. Menegatti, N. Michael, K. Berns, and H. Yamaguchi, 335–48. Advances in Intelligent Systems and Computing 302. Springer International Publishing.

Mottaghi, Roozbeh, Xianjie Chen, Xiaobai Liu, Nam-Gyu Cho, Seong-Whan Lee, Sanja Fidler, Raquel Urtasun, and Alan Yuille. 2014. "The Role of Context for Object Detection and Semantic Segmentation in the Wild." In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 891–98. IEEE.

Munkres, James. 1957. "Algorithms for the Assignment and Transportation Problems." *Journal of the Society for Industrial and Applied Mathematics* 5 (1): 32–38.

Nam, Woonhyun, Piotr Dollár, and Joon Hee Han. 2014. "Local Decorrelation For Improved Detection." *Advances in Neural Information Processing Systems*, 1–9.

Pathak, Kaustubh, Andreas Birk, Jann Poppinga, and Sören Schwertfeger. 2007. "3D Forward Sensor Modeling and Application to Occupancy Grid Based Sensor Fusion." *Proceedings of the ... IEEE/RSJ International Conference on Intelligent Robots and Systems. IEEE/RSJ International Conference on Intelligent Robots and Systems* 2: 2059–64.

Redmon, Joseph, Santosh Divvala, Ross Girshick, and Ali Farhadi. 2016. "You Only Look Once: Unified, Real-Time Object Detection." http://arxiv.org/abs/1506.02640v3.

Reina, Giulio, and Annalisa Milella. 2012. "Towards Autonomous Agriculture: Automatic Ground Detection Using Trinocular Stereovision." *Sensors* 12 (9). Molecular Diversity Preservation International: 12405–23.

Stachniss, Cyrill. 2009. *Robotic Mapping and Exploration*.

Winner, Hermann. 2015. *Handbuch Fahrerassistenzsysteme - Grundlagen, Komponenten Und Systeme Für Aktive Sicherheit Und Komfort*.

Wu, Ting-Fan, Chih-Jen Lin, and Ruby C. Weng. 2004. "Probability Estimates for Multi-Class Classification by Pairwise Coupling." *Journal of Machine Learning Research: JMLR* 5 (December). JMLR.org: 975–1005.