Why did the Robot Cross the Road? - Learning from Multi-Modal Sensor Data for Autonomous Road Crossing

Noha Radwan Wera Winterhalter Christian Dornhege Wolfram Burgard

***Abstract*— We consider the problem of developing robots that navigate like pedestrians on sidewalks through city centers for performing various tasks including delivery and surveillance. One particular challenge for such robots is crossing streets without pedestrian traffic lights. To solve this task the robot has to decide based on its sensory input if the road is clear. In this work, we propose a novel multi-modal learning approach for the problem of autonomous street crossing. Our approach solely relies on laser and radar data and learns a classifier based on Random Forests to predict when it is safe to cross the road. We present extensive experimental evaluations using real-world data collected from multiple street crossing situations which demonstrate that our approach yields a safe and accurate street crossing behavior and generalizes well over different types of situations. A comparison to alternative methods demonstrates the advantages of our approach.**

Fig. 1. Autonomous navigation at a street intersection by combining information from different modalities. The figure shows our robotic platform at a street intersection, where it uses the data from the laser and radar sensors to decide when it is safe to cross.

I. Introduction

The last two decades have seen tremendous advances in the fields of mobile robotics and autonomous vehicles. Research initiatives have been active in attempting to solve the challenges of urban navigation such as traffic merging, navigating in narrow lanes, and handling intersections. Initiatives such as RoboCup and DARPA are aimed for the development of autonomous agents for complex tasks. Over time both initiatives expanded taking significant strides towards solving the next milestone including but not limited to rescue robotics, and autonomous urban driving.

We consider pedestrian robots that are designed to autonomously navigate on sidewalks among pedestrians in urban environments and provide assistance to users for tasks such as parcel delivery, guidance or surveillance. Similar to self-driving vehicles, they need to navigate within the environment and interact with surrounding vehicles in a safe manner. One of the key requirements for such robots is to properly perceive their environment. In order to ensure safe operation, the agents should be able to identify possibly dangerous situations and seek a plan that avoids them.

For both pedestrian robots and autonomous vehicles, street intersections pose a threat not only for them but also for surrounding traffic if not handled correctly. Approaches aiming to solve this challenging problem depend on the type of intersection, whether a traffic light regulated one, an unsignalized crossing or a zebra crossing. Detecting and recognizing traffic lights in a scene is a difficult problem due to the small size of the light, the presence of similarly colored objects in the scene and especially in the case of self-driving vehicles, a decision needs to be made almost instantaneously. Even if a traffic light is detected correctly it is always desirable to determine if it is safe to cross, e.g., when a speeding car or an ambulance is approaching.

For unsignalized crossing situations the problem is even harder. Without a clear signal, such as a traffic light, the robot must make a decision based on the behavior of surrounding vehicles. Current approaches to solve this problem employ the use of a vehicle-to-vehicle communication systems alongside with a reservation based approach [17]. However such approaches were developed with automated vehicles in mind, and would require equipping all vehicles with such a system to function properly.

In this paper, we present a novel approach that addresses the problem of autonomous road crossing. We formalize the problem as a classification decision as to when it is safe to cross the street. The presented approach is robust to the dynamics of the environment such as the width of the street, road curvature, lighting and weather conditions. We consider pedestrian robots operating on sidewalks and employ a multi-modal approach to solve the problem of perceiving vehicles in traffic (see Figure 1). Our robot is equipped with two electronically scanning radars (ESR) on the sides to cover long distances and as well as with multiple laser range scanners to perceive the vicinity of the robot. We compute tracks of objects in laser range data and use radar detections in the form of radial velocity, distance and approach angle over a fixed time interval. Using data labeled by humans, we train a Random Forest classifier to predict when it is safe to cross a street. In parallel, we introduce a real-world dataset that we make publicly available. The data was collected from different intersections in Freiburg, Germany over the course of two weeks, and contains over $1,200$ annotated scenes of different crossing scenarios. We evaluate our approach on this dataset and show that our algorithm is able to accurately determine safe situations. Furthermore, we compare the prediction accuracy and generalization capabilities of the Random Forest classifier with different classifiers and a baseline approach. Extensive experimental evaluations demonstrate the performance gain

All authors are with the University of Freiburg, Institute of Computer Science, 79110, Germany.

using the proposed method.

II. Related Work

Research on the problem of safe autonomous navigation across intersections has been an active topic in the context of self-driving vehicles. However, few approaches have addressed this problem for pedestrian robots. Bauer et al. present an autonomous pedestrian robot navigating in outdoor urban environments [3]. The robot is capable of navigating through signalized crossings by detecting and classifying traffic signals at intersections. Baker and Yanco use a vision-based system for autonomous street crossing targeted at assistive robots [1]. Using cameras mounted to both sides of the platform, they track oncoming vehicles to determine the intersection safety. Once the decision to cross the street has been made, they continue to track oncoming vehicles to maintain an updated measure of the intersection safety. Due to the short range of the camera, the approach can only detect nearby vehicles in a two-lane street.

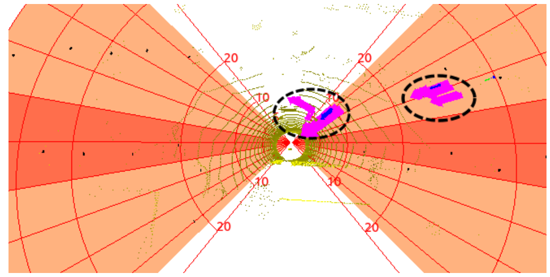
In the context of autonomous vehicles, the problem has been divided into smaller sub-problems depending on the intersection type. For traffic light regulated intersections, state-of-the-art approaches vary between combining computer vision-based methods with prior scene information to improve detection accuracy while reducing the search space within the image [2, 9, 13, 18].

Navigation through unsignalized intersections such as roundabouts is a more difficult problem, where the behavior of the vehicle is dependent on the action of surrounding agents. Several approaches have targeted the area of vehicle coordination to enable smooth interactions in intersection and merging scenarios [17]. Campos et al. present a decentralized solution for intersection crossing where local state constraints are used to enforce collision avoidance [6]. Similarly, Lee and Park develop a Cooperative Vehicle Intersection Control (CVIC) system designed to find safe driving trajectories for all vehicles approaching the intersection [15].

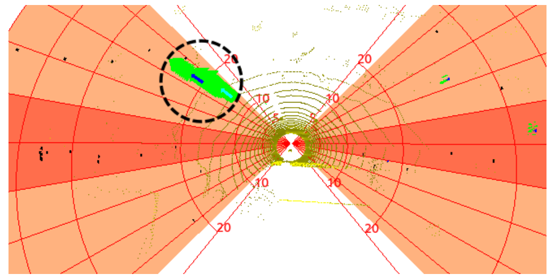
Recently, approaches have been developed that tackle the problem independent of the type of intersection and without the use of vehicle-to-vehicle communication systems [8, 19]. Dickmann et al. augment their vehicle setup with radar sensors mounted to the left and right, to enable information processing in roundabout and intersection scenarios. They process the raw radar data to build a tracker that is able to detect and track oncoming vehicles in roundabout crossings.

An alternative to the learning-based approach that we employ, some techniques model the problem as a behavioral prediction one. The aim is to compute the set of paths that an agent might follow starting from its current state [5, 10, 11]. Meissner et al. use a multi-sensor tracking system for classification of relevant objects [16]. This information is used to predict the motion of relevant objects which facilitates decision making in crossing scenarios.

The problem of safe navigation across intersections for mobile robots is quite different than for autonomous vehicles, as mobile robots do not need to make on the spot decision to stop or go. On the contrary, the robot can stand at the pedestrian crossing until the intersection is clear, whereas such behavior for an autonomous vehicle would be dangerous for oncoming traffic. Furthermore as pedestrian robots do not have the infrastructure to communicate with surrounding vehicles, they must rely solely on the sensory information to make their decision.



(a) $time = 0 \ sec$



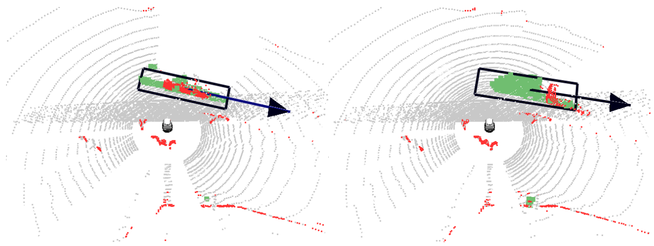
(b) $time = 5 \ sec$

Fig. 2. Time sequence example of the input radar data. The dashed circles highlight the positions of the tracked vehicles. In (a), two vehicles are approaching the robot as displayed by the pink arrows. At $time = 5 \ sec$, as shown in (b), we can observe one of the vehicles moving away from the robot which is visualized by green arrows.

Unlike behavioral prediction techniques, we do not attempt to forecast the motion of surrounding agents or introduce environment specific information such as number of lanes or road curvature. We are only interested in making a binary decision as to whether or not it is safe to cross the road. In order to make our approach suitable for the dynamic environment in which it is to be employed, we make the decision of crossing after observing oncoming traffic for a certain time interval. Thus, the behavior of surrounding vehicles is recorded for the classifier to use in making its decision. To the best of our knowledge, we are the first to present an autonomous street crossing approach for a pedestrian robot that uses automotive radars and laser scanners. The use of this sensory setup enables us to monitor oncoming traffic and make an informed decision based on vehicles up to $100 \ m$ away.

III. Learning to Cross the Street

We formulate the problem of safe autonomous street crossing as a binary classification task. The input to the classifier is the sensor data from the most recent K-second interval, while the output is a binary value as to whether it is safe or not to cross the street. Figure 2 shows a time sequence example of the sensory input data from the radars. Two radars are mounted perpendicular to the left and right of the movement direction of the robot in order to observe oncoming traffic. Tracked vehicles heading towards the robot are visualized by pink arrows, while vehicles moving away from the robot are visualized by green arrows. The size of the arrows increase with the magnitude of the velocity of the vehicles. In addition to the tracked radar data, our approach uses laser data that are preprocessed using an object tracker developed by Kümmerle et al. [14]. This approach clusters obstacles and provides bounding box information regarding their position, size and velocity (see Figure 3).



(a) t_1 (b) t_2

Fig. 3. Visualization of the output from the dynamic obstacle detection approach used from two consecutive time intervals. Tracked objects are visualized by a surrounding bounding box and an arrow displaying the direction of motion.

Given the described hardware setup with both radar and laser sensors, we track nearby objects $O_1, \cdots, O_m$ for a time interval K. We extract features from the selected interval to create a feature vector F, which along with a label L is used to train a classifier. We represent each object by its ID i, distance from the robot r_i, velocity with which it is approaching the robot v_i, and the angle with which it was detected α_i. Accordingly, an object is represented as a triple $O^T = (r, v, \alpha)$.

We create a feature vector for each time interval with a size of $m \times n \times k$ where n is the number of features for each object (in our case $n = 3$), and k is the size of interval. The number of vehicles tracked m is set to the maximum number of vehicles observed during an interval. During dataset processing, we observed the maximum number of vehicles tracked to be 60 during rush hour over a 15 $sec.$ long interval. Accordingly, we set m to this value. Objects are arranged in the vector with respect to their detection time, followed by distance to the robot with the closest object first. Under this representation, the final feature vector has the following format:

$$F = \begin{pmatrix} O_1^{t1} & O_1^{t2} & O_1^{t3} & \cdots \\ O_2^{t1} & O_2^{t2} & O_2^{t3} & \cdots \\ \vdots & \vdots & \vdots & \ldots \end{pmatrix}$$

We pass each feature vector as a training/testing sample to the classifier, along with the label L (0 representing an unsafe crossing situation and 1 a safe one). If fewer objects than the maximum are detected in the interval, the rest of the feature vector is padded with zeros.

In this paper we propose a Random Forest classifier [4] to learn the decision of when to cross. We compare it to a Support Vector Machine (SVM) [12] approach, a k-Nearest Neighbor (kNN) [7] method as well as a a baseline approach. The latter approach iterates over all detected objects within an interval, and independent of their temporal behavior decides if it is safe to cross or not. Therefore, we use the distance of the object from the robot and the detected velocity to compute the time to collision assuming the velocity remains constant. If the computed time is below a certain threshold for any of the objects throughout the time interval, the whole interval is considered unsafe.

IV. Dataset

We collected data from three different street crossings in Freiburg, Germany; two of which were traffic light regulated intersections and one a zebra crossing without traffic lights. From the traffic light regulated intersections, one was a T-junction shaped intersection, and the second was an intersection along a curved road. The zebra crossing intersection on the other hand was situated on a straight road. Figure 4 shows example images of the different intersections captured from the perspective of the robot. The data was gathered on different days at different times over the course of two weeks. To collect the data, we placed the robot on the side of the road facing the street, and recorded live traffic data from both sides of the street. Both traffic light regulated intersections contained an island in the middle of the road, therefore we also recorded data standing in the middle island facing each possible crossing direction.

(a) (b) (c)

Fig. 4. Images from the different streets where the data was recorded. Both situations are shown in (a) and (b) and have traffic lights for pedestrians and an island in the middle. The intersection in (c) is a zebra crossing without a middle island.

The dataset used for this paper is publicly available[1]. For each separate data file, we provide the output from the laser and radar trackers, along with the annotation and camera images captured from the frontal view of the platform.

Annotating the data proved to be a rather challenging problem for multiple reasons. First, the decision to cross or not must be made using only the information from this time interval without any knowledge of future or past intervals. Second, the time period for which an individual observes oncoming traffic before making a decision varies from one person to the other, rendering it difficult to assign a predetermined fixed value for it. In addition, depending on the traffic flow people often change their decision of crossing on the spot. Finally, different individuals have different crossing behaviors; in the same situation at an intersection, some might decide to cross while others choose a more conservative approach and wait for the next opportunity. Adding more difficulty to the problem, the crossing behavior varies within the same person depending on the type of intersection and the width of the street. These factors combined made the labeling procedure a rather tedious task, where we attempted to eliminate as much non-determinism as possible in order to enable our classifier to learn a meaningful classification strategy as close to human behavior as possible. For each data sample, the decision to cross is made at the end of the interval. We used a graphical user interface that combined the radar and laser views. Furthermore, we do not take into account any information regarding the intersection for making the decision, e.g. number of lanes, road curvature. We used three human annotators to label the data. In case of disagreement between the annotators, we chose majority voting over the decision of the annotators. We measured the inter-annotator agreement using Cohen's Kappa, and get a value of 0.47. The produced labels were saved with their

[1] http://www2.informatik.uni-freiburg.de/%7Eradwann/freiburg_street_crossing_dataset.html

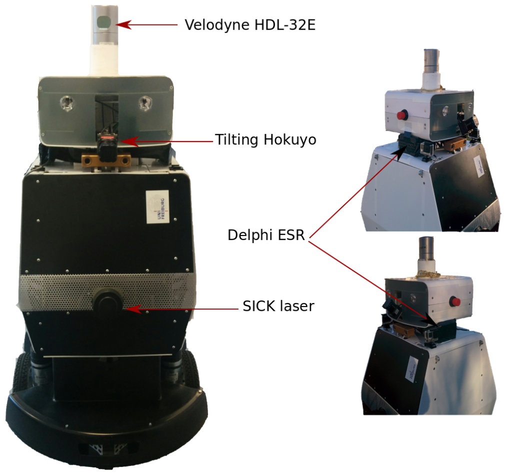
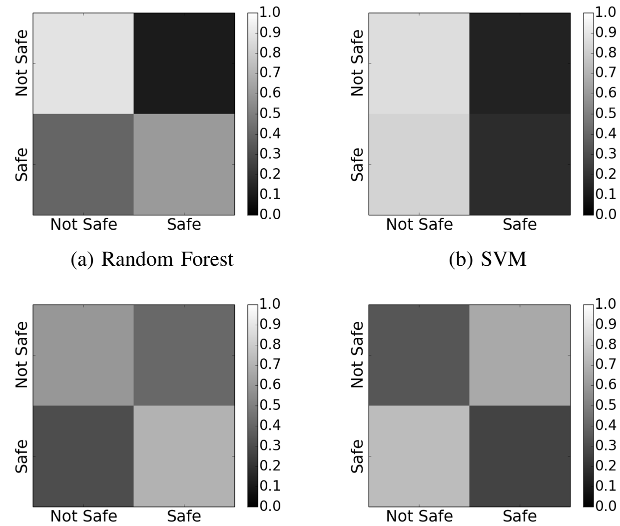


Fig. 5. The robot Obelix equipped with three laser scanners and two radars.



(a) Random Forest (b) SVM

(c) kNN (d) Baseline

Fig. 6. Confusion matrix for the various trained classifiers on the test dataset. The Random Forest classifier has the highest accuracy followed by the kNN classifier.

corresponding data point to be used for the classifier training.

V. Experiments

A. Hardware Setup

Our robot platform, Obelix [14], is equipped with several sensors. For this paper, we only relied on the three laser scanners, a Velodyne HDL-32E scanner, a tilting Hokuyo and a vertically mounted SICK scanner. In addtion, we employed two Delphi ESR radar sensors which are mounted to the left and right sides of the robot. Each radar provides both wide angle coverage at mid-range and high resolution coverage at long-range. The radar is designed specifically for the automotive industry, allowing the detection and tracking of adjacent vehicles and pedestrians across the width of the equipped platform. The long-range coverage can identify vehicles up to $174m$ with a field of view $\pm 10\,\text{deg}$, while the mid-range coverage has a shorter range of only $60m$ but with a much larger field of view $\pm 45\,\text{deg}$. Each radar provides tracked object information such as time at which the object was detected, object ID, range, radial velocity, radial acceleration and angle.

B. Experimental Setup

Following the data collection procedure, we divided each file into five second interval blocks. The resulting samples from different days were combined forming approximately 1,270 data points. We divided the data into a training and a test set with a $3:2$ split ratio. The collected sample data shows a slightly biased class distribution, with more non crossing examples than crossing ones with a ratio of $5:4$.

The parameters of the classifiers were selected by an exhaustive grid search over the hyperparameter space. We evaluated each parameter setting on the training data by applying leave-one-out-cross-validation. The training data was divided into five folds of equal size. We used four folds for training the model and one for evaluating the current parameter configuration. In the remainder of this paper, we report the results using the parameter configuration producing the best average precision and recall values. For the baseline classifier, the minimum time to collision threshold was set to $10sec$. This value was selected as a representative of the average time it took our platform to cross from one end of the sidewalk to the other.

C. Quantitative Results

In this section, we present different quantitative measures for comparing the proposed approach. Evaluating the different parameter configurations on the performance of the Random Forest classifier shows that the learning behavior of the classifier is robust to the selected parameters. We opted for a maximum tree depth of 100, a minimum sample size of 50 and an active variable size of 100. We evaluated the performance of the SVM classifier on the kernel type, C-value and γ-value, and obtain the best performance using a Sigmoid kernel with a C-value of 2.0 and a γ-value of 0.1. Cross validation showed that changing the values of either parameter did not lead to a significant performance improvement. For training the kNN classifier, we used a k-value of 8, which proved to provide the best compromise between precision and recall.

Figure 7 plots the precision and recall performance for the evaluated classifiers. In our problem setting, low precision is more dangerous than low recall, as low precision increases the risk of getting run over by an oncoming vehicle. On the other hand, low recall demonstrates a more conservative crossing approach where the robot would rather wait for the obstacle to pass than cross. Nonetheless, we do not encourage a very conservative approach as it could lead to the robot being caught in a deadlock situation unable to cross. Given our problem definition, the Random Forest classifier shows the best performance as it is able to balance between accurately determining when to cross the street and minimizing the waiting time for crossing.

The confusion matrix for the different classifiers is shown in Figure 6. Our Random Forest classifier shows the best accuracy with the lowest false positive rate in comparison to all other classifiers. The confusion matrix of the SVM classifier shows that it favors labeling examples as not safe to cross over safe, which indicates that the learned classifier is more likely to wait for longer periods of time. The kNN classifier shows slightly better performance compared to the

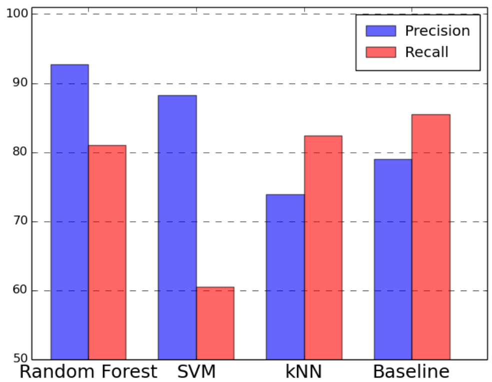


Fig. 7. Bar plot showing the precision and recall of the trained classifiers on the test data. Note that the y-axis of the plot starts from 50 to better highlight the differences between the classifiers. Our Random Forest classifier has the highest combined precision and recall values. The SVM classifier achieves the second highest precision but the lowest recall. On the other hand, the baseline classifier has the highest recall and low precision.

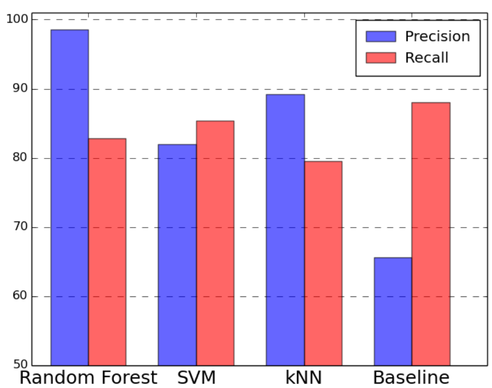


Fig. 8. Bar plot showing the precision and recall values of the evaluated classifiers, which were trained on data from two street crossing places and tested on sample data from a third place. Note that the y-axis of the plot starts from 50 to better highlight the differences between the classifiers. Our Random Forest classifier shows the best generalization capabilities with the highest combined precision and recall values, 98.6% and 82.8% respectively. The lowest precision is achieved by the baseline classifier with a value of 65.6%.

SVM, but with a higher number of mispredictions in comparison to the Random Forest. The baseline approach shows the worst accuracy, consistently confusing both classes.

D. Generalization Capabilities

The goal of this experiment is to measure the robustness of the presented approach with regard to the corresponding scenario. As previously mentioned, we collected our data from three different intersections. We used data from traffic light regulated intersections to create the training set, while we test the performance of the learned classifiers on data from the zebra crossing. After splitting the training data was observed to have the same class distribution as the original experimental setup, however the test data has a class distribution with more negative examples.

Figure 8 displays a bar plot of the precision and recall values for the trained classifiers. Our Random Forest classifier shows the best generalization capabilities with a precision value of 98.6% and recall of 82.8%. The SVM classifier has higher recall values in comparison with the initial setup, which we attribute to the unbalanced class distribution of the test set. On the contrary, the kNN classifier is able to generalize better with a 10% improvement in precision and a 3% improvement in recall relative to the initial setup. The baseline classifier on the other hand, shows the worst performance with a precision value slightly better than random guessing. This observed drop in performance for the baseline classifier can be attributed to the field of view of the radars. Since there is no full overlap between the field-of-view of the radars and the Velodyne, blind spots exist. The learning-based classifier approaches are able to learn about the presence of intermediate blind spots and hence can recover from these situations, whereas the baseline approach fails to capture them.

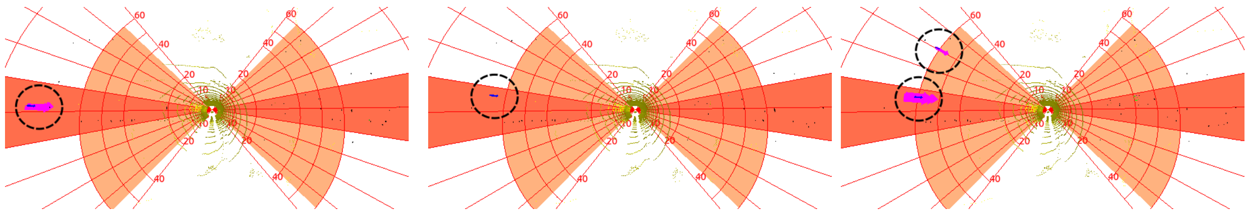
E. Qualitative Results

In this section, we present qualitative results on the presented dataset trained using the Random Forest classifier. Figure 9 and Figure 10 show examples of false negative predictions by the classifier. In Figure 9, the tracked vehicle slows down during the first half of the interval, then continues to speed up again in the remainder of it. In this example the decision of crossing varies within individuals, making it difficult to define what a ground-truth label should be. Figure 10 shows an example scenario in which a car appears to be approaching during the length of the interval with decreasing velocity. Ideally, we expect our classifier to learn in such cases that it is safe to cross, mimicking the behavior of humans at zebra crossings. However, the trained classifier is unable to learn such a behavior due to the insufficient number of examples showing similar situations in the training data. On the other hand, Figure 11 demonstrates a situation where the learned classifiers labels an unsafe situation as safe. In this example, a car coming from a side street outside of the sensor range of the robot is treated by the classifier as a false tracker detection leading to incorrect safe classification. We believe this occurs because the car appears at the very end of the interval and only for a few seconds, which closely resembles the characteristics of a ghost detection.

VI. Conclusions

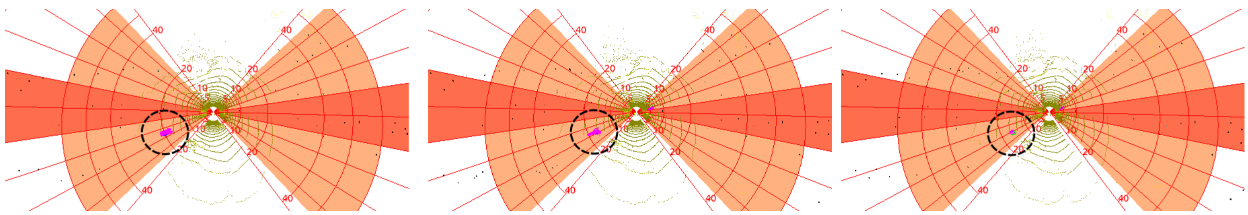
In this paper, we presented a novel approach based on Random Forests for learning to predict when it is safe to cross a street. We employ a pedestrian robot that uses multiple sensor modalities. Our approach takes into account information from laser and radar sensors to detect moving objects. Given such data over a short time interval, it decides whether it is safe to cross the street. We trained and evaluated a Random Forest classifier based on these modalities using real-world data from different places. The corresponding dataset has been made publicly available. The experimental results show that our classifier is robust to the type of intersection, generalizes well to different situations and outperforms different alternative approaches.

For future work, we would like to extend our approach into an end-to-end learning technique in which we continue to track oncoming traffic after the initial decision has been made. This would ensure safe operation in unpredictable dynamic situations that could occur during the traversal.



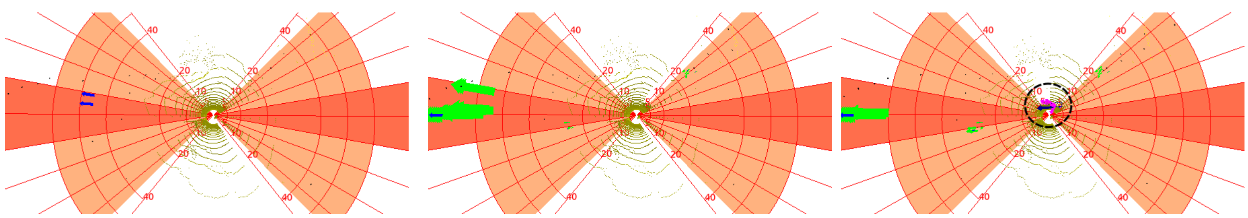
(a) $time = 0\ sec$ (b) $time = 2.5\ sec$ (c) $time = 5\ sec$

Fig. 9. Example of a misclassified sample from the test data, with a safe to cross label versus an unsafe to cross prediction. The dashed circle shown in the images highlights the positions of oncoming vehicles. At the first half of the interval a car appears to slow down, but at the last second it speeds up again. In such cases, the label selected during annotation varies greatly from one individual to the other.



(a) $time = 0\ sec$ (b) $time = 2.5\ sec$ (c) $time = 5\ sec$

Fig. 10. Visualization of a false negative classification on the test data. Throughout the entire interval the car appears to be slowing down. However, the classifier opts for a more conservative approach than the annotation.



(a) $time = 0\ sec$ (b) $time = 2.5\ sec$ (c) $time = 5\ sec$

Fig. 11. Example of an incorrect classification on a data point. The classifier predicted the interval to be safe to cross, whereas at the last second in the window a car is passing in front of the robot. Both trackers were unable to detect the car until the last second as it was outside of their sensing range due to the curvature of the road.

REFERENCES

[1] M. Baker and H. A. Yanco. Automated street crossing for assistive robots. In *Int. Conf. on Rehabilitation Robotics*, 2005.

[2] D. Barnes, W. Maddern, and I. Posner. Exploiting 3d semantic scene priors for online traffic light interpretation. In *Intelligent Vehicles Symposium (IV)*, 2015.

[3] A. Bauer, K. Klasing, G. Lidoris, Q. Mühlbauer, F. Rohrmüller, S. Sosnowski, T. Xu, K. Kühnlenz, D. Wollherr, and M. Buss. The autonomous city explorer: Towards natural human-robot interaction in urban environments. *Int. Journal of Social Robotics*, 1(2):127–140, 2009.

[4] L. Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[5] A. Broadhurst, S. Baker, and T. Kanade. Monte carlo road safety reasoning. In *Intelligent Vehicles Symposium (IV)*, 2005.

[6] G. D. D. Campos, P. Falcone, and J. Sjoberg. Autonomous cooperative driving: a velocity-based negotiation approach for intersection crossing. In *Int. Conf. on Intelligent Transportation Systems-(ITSC)*, 2013.

[7] T. Cover and P. Hart. Nearest neighbor pattern classification. *Transactions on Information Theory*, 13(1):21–27, 1967.

[8] J. Dickmann, N. Appenrodt, J. Klappstein, H. Bloecher, M. Muntzinger, A. Sailer, M. Hahn, and C. Brenk. Making bertha see even more: Radar contribution. *Access*, 3:1233–1247, 2015.

[9] N. Fairfield and C. Urmson. Traffic light mapping and detection. In *Int. Conf. on Robotics & Automation (ICRA)*, 2011.

[10] D. Ferguson, M. Darms, C. Urmson, and S. Kolski. Detection, prediction, and avoidance of dynamic obstacles in urban environments. In *Intelligent Vehicles Symposium (IV)*, 2008.

[11] J. Hardy and M. Campbell. Contingency planning over probabilistic obstacle predictions for autonomous road vehicles. *IEEE Transactions on Robotics*, 29(4):913–929, 2013.

[12] M. A. Hearst, S. T. Dumais, E. Osuna, J. Platt, and B. Scholkopf. Support vector machines. *Intelligent Systems and their Applications*, 13(4):18–28, 1998.

[13] Y. Jie, C. Xiaomin, G. Pengfei, and X. Zhonglong. A new traffic light detection and recognition algorithm for electronic travel aid. In *Int. Conf. on Intelligent Control and Information Processing (ICICIP)*, 2013.

[14] R. Kümmerle, M. Ruhnke, B. Steder, C. Stachniss, and W. Burgard. Autonomous robot navigation in highly populated pedestrian zones. *Journal on Field Robotics*, 32(4):565–589, 2015.

[15] J. Lee and B. Park. Development and evaluation of a cooperative vehicle intersection control algorithm under the connected vehicles environment. *Transactions on Intelligent Transportation Systems*, 13 (1):81–90, 2012.

[16] D. Meissner, S. Reuter, E. Strigel, and K. Dietmayer. Intersection-based road user tracking using a classifying multiple-model phd filter. *Intelligent Transportation Systems Magazine*, 6(2):21–33, 2014.

[17] J. Rios-Torres and A. A. Malikopoulos. A survey on the coordination of connected and automated vehicles at intersections and merging at highway on-ramps. *Transactions on Intelligent Transportation Systems*, 2016.

[18] Z. Shi, Z. Zou, and C. Zhang. Real-time traffic light detection with adaptive background suppression filter. *Transactions on Intelligent Transportation Systems*, 17(3):690–700, 2016.

[19] J. Wei, J. M. Snider, J. Kim, J. M. Dolan, R. Rajkumar, and B. Litkouhi. Towards a viable autonomous driving research platform. In *Intelligent Vehicles Symposium (IV)*, 2013.