Learning human motion features and trajectory predictions in large changing environments

Robin Eberhard, Stefan Dörr

***Abstract*—The ability to collaborate with humans is a key requirement of mobile robots to be versatile in a wide range of applications. From a technological perspective, this requires the robot to be aware of people in its close to midrange workspace. Due to safety purposes, most of todays mobile robots are equipped with 2D safety laser scanners mounted at leg-height. Using this sensor data for perceiving the environment without needing additional sensors is highly beneficial for economic reason and flexibility of the hardware design.**

This study presents a new approach for people detection and motion prediction in the robots workspace by solely processing 2D laser scan data. Making use of the latest advances in neural networks, fast scalable applications can be modeled without having knowledge about the inner workings. First, we present a leg detection tool, that operates on the raw laser scan data, classifying each point as human or non-human. The output is further used to learn localizing humans and predicting their trajectories. In our experiments, we show how our approach outperforms a state-of-the-art leg detection approach and is able to precisely localize humans as well as predicting their motion for a certain prediction horizon.

***Keywords*— *[keywords]* .**

I. Introduction

A wide variety of applications for mobile robots and automated guided vehicles (AGV) in industrial, entertainment or domestic environments require the robot to be aware of humans within its workspace as a basis for a non-disruptive navigation and efficient user interaction. Examples are sales assistance robots guiding and approaching customers in retail applications, care robots navigating through populated domestic scenes or AGVs sharing the workspace with workers in logistic environments. To allow the robots to operate in a shared workspace with humans, most mobile robots are nowadays equipped with 2D safety laser scanners mounted at leg-height to realize a safe collision avoidance with humans. While these sensors primary task is to assure the detection of an object with a certain minimum size within a certain predefined area, most of them also provide raw data in terms of range measurements of the environment. Using this sensor data to perform detection, tracking and motion prediction of humans is highly beneficial because it avoids the costs of additional sensors. Moreover, in some applications, there is no possibility to mount additional sensors at an appropriate position due to the task and hardware design of the robot.

In most robotic applications, object detection is resolved with vision based approaches due to the high depth of information one could gather from images. Lidar data, especially 2D lidar data, lacks this information depth since it only delivers a set of range points of the object's surface at the height of the scanner's mounting position. In general, this information lack makes it more difficult to distinguish relevant objects from other objects within the environment. For example, in our application, chair legs closely resemble human legs when viewing them from a certain position resulting in false positive detections.

In some previous work, laser-based human perception has already been approached with model-based solutions, that require prior knowledge about shapes and behavior of humans [1] [2]. The disadvantage of these approaches is that they depend on how well the manually designed model fits to the current scene and sensor characteristics of the used sensor. Additionally, it often requires expert knowledge for parameter tuning when deploying in a new application.

Learning based approaches have the potential to overcome these problems and further increase detection rates without needing any prior knowledge. By exploiting current advances in the field of machine learning, we present a new approach, where a neural network learns relevant characteristics on its own and places unique identifiers on humans, making it easy to track them over a long period of time. Having the information and history of locations, we can learn trajectories, behavior and intentions of humans in the presence of the robot. We provide a way to model the trajectories using Long short-term memory (LSTM) cells in combination with a Mixture density layer, which outputs a set of normal distributions for the predicted locations of the tracked humans.

This paper is organized as follows. Section 2 presents relevant related work. In section 3, we describe the model architecture and models and the basic formula of our algorithm followed by our experiments in Section xy. Section 4 concludes the paper.

II. State of the art

LIDAR scanners pose an attractive way to scan the environment, due to the simple interpretation of the data and the low computational power required. The idea to extract people from this input has been introduced in [3] where laser range measurements are grouped into blobs and objects. Moving objects are considered as people with the downside of not detecting immovable objects. In order to improve on this idea, a combination of LIDAR and vision data [4] can be applied. Information of leg positions are here extracted from the laser scanner, while a camera provides additional information through skin-colored face detection. Arras et al. proposed a machine learning algorithm [1], where pre-defined geometrical features are trained in a supervised learning approach. It was further adapted in [2] to expand the detection in retirement homes, allowing the classification of people with different walking aids. The algorithm allows to add additional features, which are weighted and therefore filtered by importance. However, the expert knowledge of geometrical features pose a downside to the algorithm as well as the computational power required during evaluation. Convolutional neural networks

therefore present a way to learn features without prior expertise. Since their introduction [5], they are the go-to standard for object detection [6], due to their versatility and their simple and fast implementation. Convolutional networks can be further expanded for segmentation of images [7], which gives a way to keep the input dimensionality while simultaneously detect and label objects.

Trajectory prediction has already been achieved in [8], where handwriting is generated using a mixture density network (MDN) [9]. These networks provide a way to model the most probable outcome as a function of multiple density terms. In order to predict from a history of input values, a Long short term memory (LSTM) [10] is included. This allows a way to store previous inputs and therefore predict outcomes when similarities are found. People based trajectory prediction has previously been achieved [11] using a similar approach. Making use of LSTM and MDN we therefore predict trajectories using unsupervised learning, requiring only positions of people as an input.

[***expand section with some work of vision based object detection, convolutional networks and other related work that was of interest for the selected approach***]

III. Model Architecture

Learning positions of people and identifying them requires a neural network to solve several tasks. First, the laser scans are classified by using a set of convolutional and pooling layers. The points labeled as belonging to legs are sorted out and typically clustered based on euclidian distance, in order to identify persons. This approach does not work well, when people are close together or even when they are moving, as the program would then either cluster too many legs together or see two legs as separate people respectively.

This can be countered by supplying the clustering algorithm with additional information. During the convolution step, the network learns features which are unique to legs in different situations. Those can be extracted and passed as additional parameters to the clustering algorithm, which not only allows higher accuracy, but also providing a pair of legs with a unique identifier.

Having identifiers and positions, they can then be fed into the third and last step of the setup, which aims to predict trajectories of individual people. The network takes a set of positions, that are run through an LSTM cell, which implicitly works as a memory for the program and holds the information necessary to build future trajectories.

A. Classification of the laser scan

In order to classify a laser scan, a snapshot of the current scan is taken in equal timesteps. Each snapshot contains one dimensional arrays of length N_{points} with distances r_i to the closest intersections from the laser beams.

From the one-dimensional array as input, the network would only find characteristics in the objective shapes of the laser scans. To also account for movement, N_{frames} snapshots are composed together. The desired outputs are probabilities for each point in the most current snapshot to be in either class leg or non-leg.

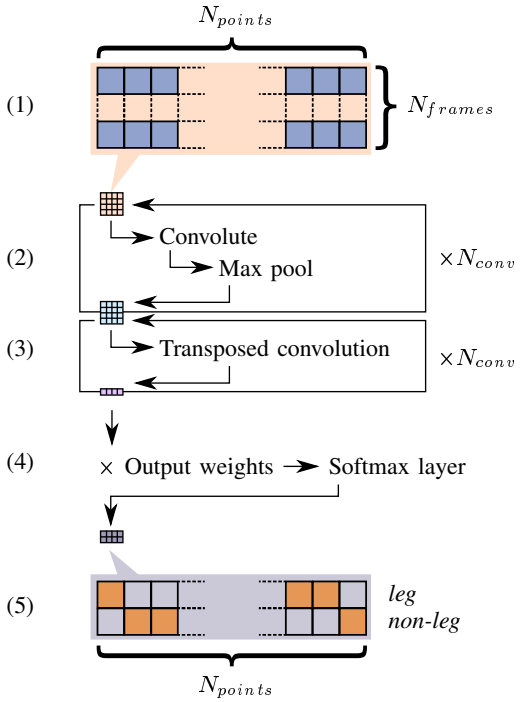

Fig. 1. **Laser scan classification.** (1) The input is a 2D array of $N_{points} \times N_{frames}$ containing distances from a laser scan. It is fed into (2), where features are extracted using convolutions and downscaled using a pooling layer. After N_{conv} iterations of (2), the now one-dimensional output is upscaled in (3) into the shape N_{points}. (4) multiplies the output and applies the softmax function, resulting in probabilities of each point belonging to class leg or non-leg in (5).

Therefore, the mathematical description of the model can be described by a function F that maps an input of size $[N_{frames}, N_{points}]$ to an output of size $[N_{points}, 2]$.

The output is modelled from the input as outlined in Figure 1 and described in more detail in the following. When convolutional layers are used for classification, a network will train the weights such that they represent characteristics for a class. In this way, the characteristics are first of geometrical nature and become more abstract the more layers there are.

The input is therefore fed into a convolutional layer and then downsampled using a max pooling layer. This is repeated N_{conv} times, after which the output is a one dimensional array of length smaller than N_{points}.

To gain a classification for each point in the most current laser scan, the output is upscaled using transpose convolutions, again extracting features. Multiplication by output weights and applying the softmax function results in the desired shape $[N_{points}, 2]$. Each entry consists of the two probabilities of a point being in class leg P_{leg} or non-leg P_{nonleg} with $P_{leg} + P_{nonleg} = 1$.

B. Clustering of leg points

In order to gain information about the current position and identity of people, the classified points are clustered together. This usually poses two major problems:

- The clustering has a high margin of error when legs of a single person are too far away or when legs of different people are together
- tracking of a person has to be handled externally.

With the following approach, we find a high accuracy and are also able to track people on certain features.

As the inputs are convoluted, the network finds characteristics, such as shapes and movement. In order to find the most important characteristics, a training was implemented to find weights v, such that parameters of people in different situations are easily separable. The ith convoluted layer has its weights w_i multiplied by v_i and the resulting cluster parameter p_i is then found by the sum of all weights:

$$p_i = v_i * \sum_j w_{i,j} \tag{1}$$

This introduces $2 * N_{convs}$ additional parameters to the clustering algorithm, highly increasing accuracy when used together with the euclidian distance. As the parameters have been chosen such that people in different situations are easily separable and given the unlikeliness of two people being in the same position in the same situation, we can additionally use the parameters to place a similar identifier on a single person over several frames. This allows to track them in order to predict their movements.

C. Trajectory prediction

With the information of position of individual people, we can learn to predict trajectories over some time. A trajectory of a single person can be modeled with a normal distribution, or if the path is unclear, a number of distributions N_{dist} are necessary.

This leads to the approach of using a mixture density network [9]. The network learns from parameters π, μ, σ and ρ according to [8], so that a distribution is given as:

$$\mathcal{N}(x|\mu,\sigma,\rho) = \frac{1}{2\pi\sigma_1\sigma_2\sqrt{1-\rho^2}} \exp\left[\frac{-Z}{2(1-\rho^2)}\right] \tag{2}$$

with

$$Z = \frac{(x_1-\mu_1)^2}{\sigma_1^2} + \frac{(x_2-\mu_2)^2}{\sigma_2^2} - \frac{2\rho(x_1-\mu_1)(x_2-\mu_2)}{\sigma_1\sigma_2}. \tag{3}$$

Therefore we can define the ensemble of distributions as

$$Pr(x|y) = \sum_{j=1}^{N_{dist}} \pi_j \mathcal{N}(x|\mu_j,\sigma_j,\rho_j) \tag{4}$$

for every input point x from the laser scan.

As the future movement depends on previous positions, the parameters μ, σ, ρ and π are derived from a recurrent network cell. A long-short term memory (LSTM) was used, which allows for information to be kept over a long time. This way, the network was trained on different patterns, so that it can recognize similar ones later in the evaluation. The LSTM cell will take the input positions and directly outputs the parameters for the mixture of normal distributions.

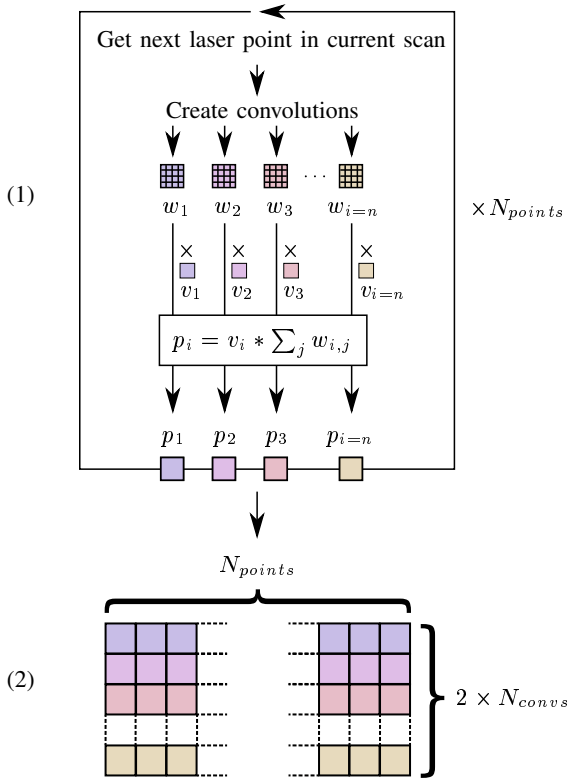

Fig. 2. **Convolution-based clustering.** (1) For every point in the current laser scan, N_{conv} convolutions are created. A clustering parameter p_i is derived from the weights w_i of the convolution and the trained weights v_i, which assumes that some convolutions are more important than others. (2) The resulting clustering matrix holds $n = 2 * N_{conv}$ parameters for each point in the current laser scan.

IV. Experimental results

A. Classification of laser data

In order to evaluate the leg detector, the robot was set up in a corridor, recording legs of people passing through over a course of three hours. Walls were separated from the corridor, giving a ground truth of the data. The leg detector receives the input from the laser scan and labels each point as leg or non-leg. In the comparison to the ground truth, we find, that the detector works best in close range, while becoming weaker as the range increases. This is demonstrated in Figure IV-A, where a short, 100 second long exposure was analyzed.

The decrease in the first half originates mainly from the difficulty of interpreting the laser data, as the points spread further out depending on the distance to the scanner. The second half additionally suggests too few data points as input to the convolutional network. At a distance of $2.2m$ we still find good results in a reasonable range around the robot, which we can use to evaluate a three hour long recording of the same corridor. We find the confusion matrix in Table IV-A, omitting false and true negatives as the sensor recorded only a specific wall in that range as negative data, making the input redundant. The long recording shows how well the network reacts to more arbitrary data, most importantly the accuracy is still high, which we can compare against the state of the art leg detector.

TABLE I. LEG DETECTION IN A RADIUS OF $2.2m$.

	Detected Label		
True Label	Leg	Non-Leg	Total
Leg	**121199** (84.0%)	**19415** (16.0%)	**11750**

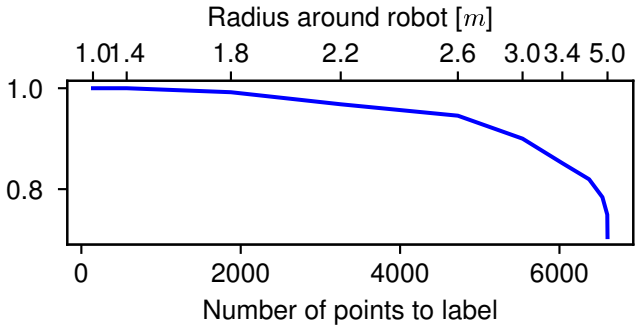

Fig. 3. **Distance dependency of the leg detector.** The diagram shows accuracies of the leg detection as a function of the number of points in a range around the laser scanner.

B. People detection using clusters

Detecting people from the output of the laser classification is achieved by clustering the results as explained in section III-B. We compare the detection to the algorithm provided in the official ROS-repository[1] based on [1], where the detection is based on a Kalman filter. A ten second long exposure from the corridor data was hand-labeled and compared to the outputs of the programs. Due to the dependency on the distance to the scanner, different radii were considered. We find the parameters:

- **True positives** (TP): The people that were detected correctly,
- **False positives** (FP): Non-human objects classified as persons,
- **False negatives** (FN): People that were not detected.

The parameters are combined into the $F - Measure \in [0, 1]$, separating good results (close to 1) from bad results (close to 0).

$$F = \frac{2 * precision * recall}{precision + recall} \tag{5}$$

with

$$precision = \frac{TP}{TP + FP} \tag{6}$$

$$recall = \frac{TP}{TP + FN} \tag{7}$$

V. CONCLUSION

ACKNOWLEDGMENT

The authors would like to thank...

[1] http://wiki.ros.org/leg_detector

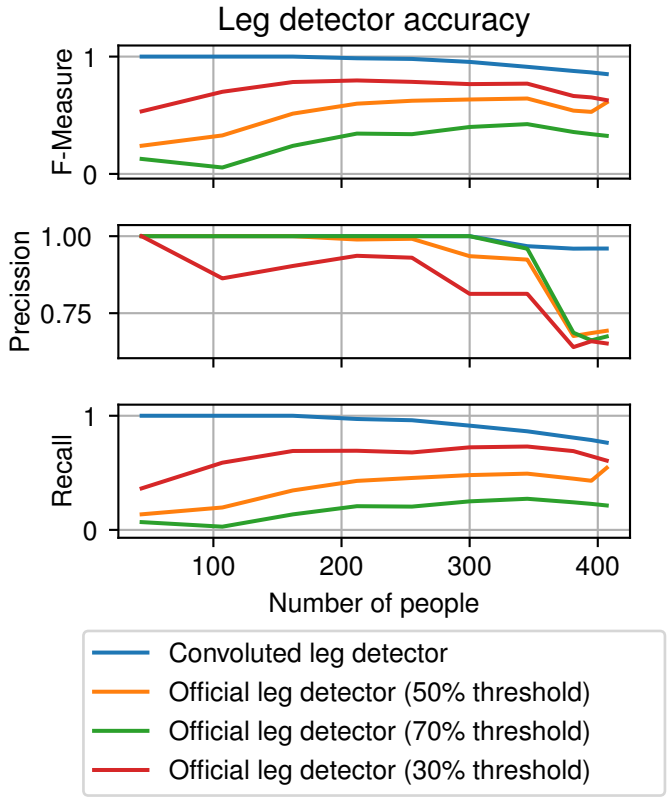

Fig. 4.

REFERENCES

[1] K. O. Arras, Óscar Martínez Mozos, and W. Burgard, "Using boosted features for detection of people in 2d range scans," in *IN PROC. OF THE IEEE INTL. CONF. ON ROBOTICS AND AUTOMATION*, 2007.

[2] C. Weinrich, T. Wengefeld, C. Schroeter, and H.-M. Gross, "People detection and distinction of their walking aids in 2d laser range data based on generic distance-invariant features," in *Robot and Human Interactive Communication, 2014 RO-MAN: The 23rd IEEE International Symposium on*. IEEE, 2014, pp. 767–773.

[3] A. Fod, A. Howard, and M. A. J. Mataric, "A laser-based people tracker," in *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No.02CH37292)*, vol. 3, 2002, pp. 3024–3029.

[4] M. Kleinehagenbrock, S. Lang, J. Fritsch, F. Lomker, G. A. Fink, and G. Sagerer, "Person tracking with a mobile robot based on multi-modal anchoring," in *Robot and Human Interactive Communication, 2002. Proceedings. 11th IEEE International Workshop on*. IEEE, 2002, pp. 423–429.

[5] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," vol. 86, no. 11, pp. 2278–2324.

[6] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, pp. 1097–1105. [Online]. Available: http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks

[7] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3431–3440.

[8] A. Graves, "Generating sequences with recurrent neural networks," *arXiv preprint arXiv:1308.0850*, 2013.

[9] C. M. Bishop, "Mixture density networks," 1994.

[10] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[11] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, and S. Savarese, "Social lstm: Human trajectory prediction in crowded spaces," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 961–971.

IMPORTANT INFORMATION DURING PRODUCTION

There are 2 todos left!