

UTILIZATION OF CONVOLUTION NEURAL NETWORK BASED ROAD DETECTION IN MOBILE ROBOT LOCALIZATION

J. Krejsa, S. Vechet*

Abstract: Mobile robot on-road navigation requires fusion of both global and local sensory information with an emphasis on the road detection processing. The paper deals with the road detection based on convolution neural networks (CNN) using commonly available tools such as TensorFlow and Keras. The road is defined by its linear boundaries. Network output is formed by the road definition together with classification parameters and serves as a local sensor in Kalman filter based localization. CNN based road detection is currently capable to successfully detect about 90% of images.

Keywords: Convolution Neural Network, Localization, Road Detection, Mobile Robot, Kalman Filtering.

1. Introduction

The localization of on-road mobile robot strongly depends on the ability to detect the road itself and robot position with the respect to the road, no matter the mechanism of the localization itself.

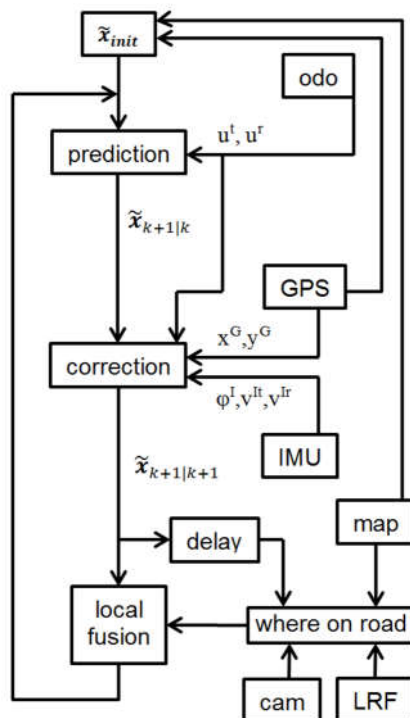


Fig. 1: Localization scheme.

Commonly used nonlinear Kalman filter based localization fuses the local and global sensory information, as summarized in fig. 1, with state vector $\mathbf{x}_k = [x_k^R, y_k^R, \phi_k^R, v_k^{Rl}, v_k^{Rr}]^T$ depicting the position of the robot in 2D plane together with generalized velocities, for more details see Krejsa (2018).

Road detection is nowadays mainly in the domain of image processing, with a number of approaches, recommended introduction is given by Bar Hillel (2014). With increasing volume of available visual data the convolution neural networks (CNN) are becoming more and more popular in recent years, giving impressive results in tasks such as road lane detection; see e.g. Kim (2014).

This encouraged us to attempt to use CNN for the detection of the road itself, in particular in the case of park pathways, where most of the common image processing methods give questionable results due to the changes in path surface and in many cases uncertain boundary of the road.

The paper gives detail about the image acquisition and labelling, CNN structure and introduces the criterion for determination whether the detection can be considered successful.

* doc. Ing. Jiří Krejsa, PhD., doc. Ing. Stanislav Věchet, PhD.: Institute of Thermomechanics AS CR, v.v.i., Technická 2, 616 69. Brno, Czech Republic, krejsa@fme.vutbr.cz

2. Methods

2.1. Road images acquisition and labeling

Successful use of CNN requires as large set of training data as possible. During the field tests for the Robotour robotic competition the large number of road images were acquired by mobile robot B2 in the city park roads. The images were acquired under various light conditions, roads in the park vary significantly in its surface and boundary type.

The road itself is in the images defined by linear boundaries of left and right side, in particular by four points A..D starting in left lower corner and running clockwise, therefore the AB defines the left boundary of the road and CD the right one.

The roads in images were labeled manually, together with the classification meta information corresponding to the set of meta = {road, openspace, crossroad, noroad}, the values are selfexplanatory. The examples of images together with the manually labeled road borders (where available) are shown in Fig. 2.



Fig. 2: Road images examples.

2.2. Convolution neural network structure and implementation

Several topologies of CNN were tested with slight variations in layer sizes, however, the influence of those variations on the detection quality was not of a significant influence. The structure used for further evaluation was as denoted in Tab. 1.

Tab. 1: CNN structure and parameters

Parameter	Value
Input size	480x360x3
Layer structure	conv-conv-pooling-dense-dense
Layer sizes	30 (7x7), 10 (3x3), 2x2, 90, 9
Activation functions	Relu (all layers)
Training set size	4857 images
Test set size	172 images

Commonly available computational framework TensorFlow (2018) with neural network extension Keras (2018) was used as the tool for CNN implementation. All the CNN related code was written in Python, further processing was written in Matlab.

3. Results

Trained CNN was tested on a test set consisting of the images not presented in the training set that included common road images, cross sections, images of road with obstacles and non-road images.

3.1. Successful detection definition

The output of CNN gives the set of points defining the boundary of the road found together with the classification of the result. The success / failure in road detection can not be calculated simply by the distance between detected / labeled boundary key points, therefore the following criterion was created in order to more meaningfully represent the quality of detection with respect to its intended use – fusion with other sensors to improve robot localization.

The criterion takes two variables into account. The first one is the maximum horizontal offset of the road boundary in the center of the image, denoted as Δx , the second one is the absolute value of angular error between the road center labelled manually and road center of the CNN output, denoted as $\Delta\phi$. The center of road is given as the line defined by the centers of front and back road boundaries (points AD and BC). In order to evaluate the road detection as successful, both variables must have values below the parametrical limits denoted as $\Delta\phi_{\max}$ and Δx_{\max} . More formally the success is defined as

$$pass = \begin{cases} true & (\Delta\phi_{\max} > \Delta\phi) \wedge (\Delta x_{\max} > \max(\Delta x_L, \Delta x_P)) \\ false & otherwise \end{cases} \quad (1)$$

where for the points $A = [x^A, y^A]$, $B = [x^B, y^B]$, ... of manual image labels and $A_N = [x_N^A, y_N^A]$, ... denoting the output of CNN the particular term for angle is as follows and similarly one can determine Δx_L and Δx_P for the offset of the left and right boundary in the center of the image

$$\Delta\phi = \left| \arctan \left(1 + \frac{(x^A + x^D)(y^B + y^C) - (x^B + x^C)(y^A + y^D)}{(x^A + x^D)(-x^A + x^B + x^C - x^D)} \right) - \arctan \left(1 + \frac{(x_N^A + x_N^D)(y_N^B + y_N^C) - (x_N^B + x_N^C)(y_N^A + y_N^D)}{(x_N^A + x_N^D)(-x_N^A + x_N^B + x_N^C - x_N^D)} \right) \right| \quad (2)$$

3.2. Road boundary detection

The examples of road boundary detection results are given in Fig. 3. Yellow dots correspond to the manual labeling of the image, red dots correspond to the output of CNN. The first row shows the correctly identified road, 3A is an easy to detect simple road, 3B is an example of road with shadow with a single boundary only. The second row shows the example of incorrect detection. 3C is an example of the road with obstacles that does not correspond to manual label, however, here the failure of CNN is questionable, as the road boundaries are hard to detect for human as well. 3D is an example of failure where CNN clearly detects the shadow as the road boundary.

The overall results are summarized in Table 2. Parameter values for the detection to be considered successful was set to $\Delta\phi_{\max}=10^\circ$ and $\Delta x_{\max}=8\%$ of image width.

Tab. 2: Road detection results. Openspace and crossroad are taken as single category

Image type	Success rate	Failure rate / description
Road	88 %	4 % openspace, 1 % no road, 7 % did not pass criteria
Openspace	65 %	34 % road, 1 % no road
No road	90 %	1 % road, 9 % openspace



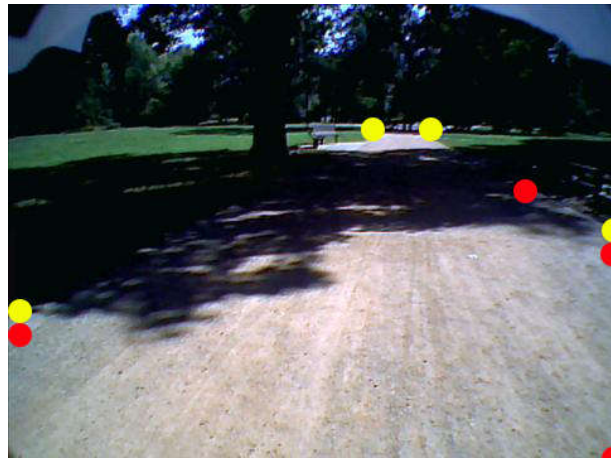
A: simple road – correct



B: single side road - correct



C: road with obstacle - questionable



D: deep shadow road - failure

Fig. 3: Road boundary detection results.

4. Conclusions and future work

The CNN proved that it can be used successfully for road detection with success rate almost 90%. We believe that further improvement could be reached with the higher number of images in the training set, mainly covering the hard-to-decide cases that are difficult to correctly label even for the human. In the future work we want to focus on extension of meta-information in the images, such as presence of obstacles, boundaries of openspaces, general definitions of cross-roads, etc.

Acknowledgement

The results were obtained with institutional support RVO 61388998 of the Institute of Thermomechanics AS CR v.v.i. The authors would like to thank A. Dulavová and K. Stehlíček for data labeling.

References

- Keras: The Python Deep Learning Library, available online at keras.io, retrieved 2018
- TensorFlow, available online at www.tensorflow.org, retrieved 2018
- Krejsa J. and Vechet S. (2018) Fusion of Local and Global Sensory Information in Mobile Robot Outdoor Localization Task. In: *Proceedings of the 2018 18th International Conference on Mechatronics – Mechatronika (ME) 2018*. Brno, pp 296-300.
- Bar Hillel A., Lerner R., Levi D., and Raz G., (2014) Recent progress in road and lane detection: a survey, *Machine Vision and Applications*, vol. 25, no. 3, pp. 727–745.
- Kim J. and Lee M. (2014) Robust Lane Detection Based on Convolutional Neural Network and Random Sample Consensus. In: *Proceedings of International Conference on Neural Information Processing (ICONIP) 2014*, pp 454-461