Combined Road Tracking for Paved Roads and Dirt Roads: Framework and Image Measurements

Bianca Forkel*, Jan Kallwies* and Hans-Joachim Wuensche

***Abstract*— We propose a modular framework for 3D tracking not only of paved roads but also of dirt roads. It is based on recursive state estimation of lane boundary points connected by clothoid pieces. While our tracking is flexible to integrate every kind of measurement, we specifically propose two image-based measurements. They combine traditional with modern computer vision: On the one hand, we show how to use directed edge detection to robustly measure road and lane boundaries. On the other hand, we introduce a innovative CNN-based measurement utilizing the self-similarity of (dirt) road areas.**

We demonstrate the performance of our approach in challenging scenarios. On a marked road, we achieve a median error of 0.13 m for the ego lane's boundaries in 25 m look-ahead. A difficult dirt road can also be tracked reliably with a look-ahead length of 25 m, resulting in a median error of 0.3 m. The tracking, as well as both measurements, are real-time capable.

I. Introduction and Related Work

Autonomous driving has been a very popular research topic in the last decades. A still challenging task is to establish autonomous driving outside of dedicated testing areas on more poorly developed roads. Therefore, we must not rely on maps, as we have to cope with poor GNSS conditions, esp. in forests or street canyons, and the lack of high precision maps in more rural places. So the key is an independent perception of the environment and especially the road to drive on.

A lot of research has been done on road detection and tracking, but every approach is either specialized on (marked) asphalt roads, *or* dirt roads. In this paper, however, we propose a common framework for all tasks: roads *and* dirt roads, with or without lane markings, single or multiple lane.

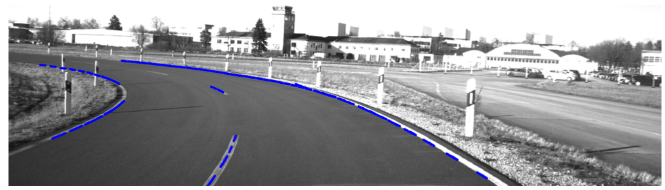
Road Model: The chosen road model determines for which scenarios the system is suitable. Geometric representations range from simple line models over generic smooth curves (i.e., splines) to clothoids. Clothoids are closely related to driving since their constant change in curvature is equivalent to a vehicle's constant steering rate. Thus, a classical approach is to model the ego lane with a single clothoid relative to the ego vehicle. Involving just three parameters – curvature, curvature change, and road width – this is a very efficient representation, which enabled lane following in the 1980s using impressively low computational power [1]. It is, however, quite limited in its application.

Modern approaches like [2] rely on more complex and flexible geometric models. Instead of modeling the road as a whole, the single lane boundaries are modeled by splines allowing to represent multiple lanes, branches, junctions, and exit ramps directly. The framework, however, relies on a flat world assumption and considers only two-dimensional positions and the heading angle and is thus not suitable for tracking dirt roads in hilly terrain.

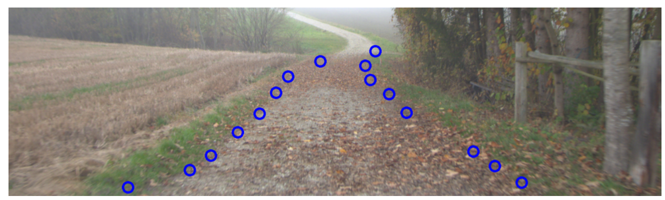
In this paper, we present a model based on the representation of lane boundaries using pieces of clothoids in full 3D space. It is suitable for complex road scenarios made up of "simple" geometric structures, as well as dirt roads with just one lane, but a greater geometric variety of the road course.

Measurements: The second essential component of a road tracking system is the measurement and detection process. Using our framework, all different kinds of measurements in all modalities could be incorporated into the road estimation process. However, in this paper, we would like to present two specific ones, both based on camera images.

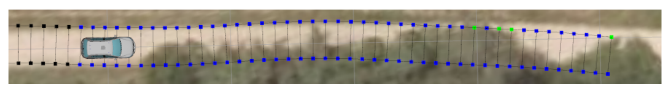
Camera images provide rich information about the course of the road and are thus often used for road tracking. To simplify the measurement process, it is often shifted to 3D. Therefore, the images are fused with depth information, e.g. from a LiDAR sensor, into 3D structures like multi-model grid maps [3], or transformed to a birds-eye-view based on a flat world assumption (e.g. [4], [5]). Another possibility is to perform stereo matching [6]. Nevertheless, this always imposes limitations – a longer look-ahead can be achieved if the measurements are performed directly within the images.



(a) CRONOS-based edge measurements of a marked road



(b) CNN-based similarity measurements of a dirt road



(c) Tracked 3D boundary points of the dirt road in **(b)**

Fig. 1: Examples for the proposed image measurements and visualization of a tracked dirt road on an aerial map. Points shown in blue were updated by a measurement within the last 3 s. Demo video on `https://www.mucar3.de/iv2021-road-tracking`.

All authors are with the Institute for Autonomous Systems Technology (TAS) of the Universität der Bundeswehr München, Neubiberg, Germany. Contact author email: `bianca.forkel@unibw.de`

**Bianca Forkel and Jan Kallwies contributed equally to this work.*

Roads: For roads, several approaches to detect lane markings in camera images have been proposed, with a recent trend to machine learning [7], [8], [9]. However, instead of processing the whole camera image, we want to take up another idea from the 1980s – to use the lane state estimation of the last time step to limit the search space for new measurements [1]. Despite lowering the computational effort, this makes the measurement quite robust against all kinds of disturbances and false positives.

Thus, in this paper, we propose the modernization and a significantly expanded and improved application of the CRONOS edge detection approach [10], [11, p. 132] – an extremely efficient method to extract edges of a known orientation using prior knowledge. It is especially suitable for paved roads (marked and unmarked).

Dirt Roads: Detecting the lane boundaries of a dirt road is, however, significantly more difficult. Classical edge detection often fails here due to the lack of high-contrast edges. Modern machine-learning-based approaches like semantic image segmentation show promising results. While semantic segmentation is suitable to segment the road, it requires labeled data covering the roads' appearance in the final application area for its offline training process. Since dirt roads show a great variety in their optical appearance, online adaptive approaches are better suitable for this task. A popular approach has been used at the DARPA Grand Challenge 2005 [12] using information obtained from a LiDAR sensor to train an image-classifier online. The authors of [3] and [13] considered the area underneath the ego vehicle from an accumulated multi-modal grid to infer features of the road.

In this paper, we present a new CNN-based image measurement that does not rely on other sensors or intermediate representations of the environment. It is able to segment the current camera image solely based on a rough initial indication of road areas in the image.

Contributions: In summary, the particular contributions of this paper are:

- A flexible framework for road and dirt road tracking in 3D space, based on recursive state estimation,
- a robust measurement for lane markings and road boundaries using CRONOS edge detection, and
- a new CNN-based image measurement for dirt roads.

Figure 1 shows some examples.

II. Road Model and System Architecture

The overall goal is an estimation of the road course in 3D. To allow more complex maneuvers like overtaking, we distinguish the different lanes of a road and thus track the single lane boundaries. The lane boundaries are modeled as connected clothoid pieces represented by point sequences. For multiple neighboring lane boundaries, it is sufficient to model and estimate the position and orientation of just one primary lane boundary. The neighboring points can then be defined relative to that primary point as it is depicted in Fig. 2. While Fig. 2 shows an example of a simple road with two lanes, this model could represent every possible road course as well as opening, closing, or branching lanes.

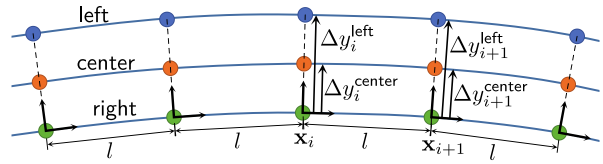


Fig. 2: Exemplary modeling of a road with two lanes. The position and orientation of one lane boundary are given directly.The position of the other lane boundaries are defined by their offset Δy relative to the primary lane boundary. Here, the primary lane boundary is the right, it could, however, also be an other one.

A set of neighboring lane boundary points $\mathbf{x}_i$ is defined by its primary point's position ${}^{\text{odom}}\mathbf{p}_i = \left(p_x\ p_y\ p_z\right)^\mathsf{T}$ and orientation, given by the quaternion ${}^{\text{odom}}\mathbf{q}_i = \left(q_x\ q_y\ q_z\ q_w\right)^\mathsf{T}$, in a stationary coordinate system, its curvature c, and the respective offsets $\Delta\mathbf{y}_i$ of the secondary lane boundary points

$$\mathbf{x}_i = \left(p_x\ p_y\ p_z\ q_x\ q_y\ q_z\ q_w\ c\ \Delta\mathbf{y}\right)^\mathsf{T}. \tag{1}$$

The length of the offset vector $\Delta\mathbf{y}_i$ depends on the number of lane boundaries being tracked. The width of lane k at point $\mathbf{x}_i$ is given by the neighboring lane boundary $k-1$:

$$d_i^k = \Delta y_i^k - \Delta y_i^{k-1} \tag{2}$$

Empirical evaluations have shown that a point distance l of 2 m between the primary lane boundary points is well suitable. The point distances of the other lane boundary points result from the curvatures at hand. The dead reckoning coordinate frame `odom` serves as a stationary coordinate system.

The usage of clothoids as the fundamental geometric structure implies a linear evolution of the curvature between two lane boundary points. Thus, the clothoid parameter of curvature change $\mathring{c}$ between two adjacent points is given by

$$\mathring{c}_i = (c_{i+1} - c_i)/l. \tag{3}$$

The current state of all lane boundary points is recursively estimated by solving a maximum a posteriori problem (described in Section III). It incorporates internal geometric information and constraints, as well as external information from a terrain estimation [14] and measurements (two approaches based on camera images are proposed in Section IV). The proposed measurements use the current state prediction as prior knowledge, such that the process follows the cycle:

1) Predict from last update
2) Measure lane boundary points using prediction
3) Update through optimization with new measurements

Accordingly, separate modules (ROS nodes) for tracking and measurements were implemented as shown in Fig. 3.

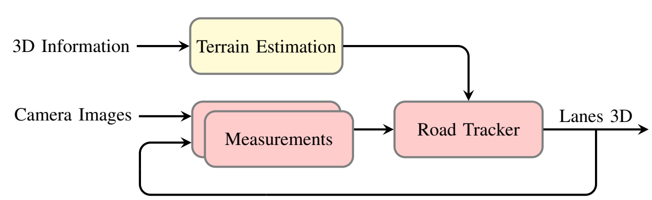


Fig. 3: Flowchart of the proposed road tracking pipeline. The red boxes show the contributions of this paper.

III. Road Tracker

To get a smooth estimate for the lane boundaries in consistency to both the measurements and an internal road model, a non-linear optimization of all lane boundary points $\mathbf{x}_i$ is performed by minimizing the least squares cost function

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \sum_i \left(\mathbf{c}_{\text{Meas.}}(\mathbf{x}_i) + \mathbf{c}_{\text{Int.}}(\mathbf{x}_i) + \mathbf{c}_{\text{Constr.}}(\mathbf{x}_i) \right). \quad (4)$$

The single cost functions $\mathbf{c}$ are described in the following.

A. External Information – Measurements

The formulation of the road estimation task as a maximum a posteriori problem allows to integrate and fuse all kinds of measurements by adding different residuals to the overall cost function. To select an appropriate cost function for a measurement approach, we differentiate between three kinds of measurements:

1) 2D measurements, e. g., image measurements, which leave one degree of freedom, that we determine from
2) Terrain height measurements, and
3) 3D measurements, which can be used directly.

All measurements presented in this work use the estimated lane boundary points from the last time step as prior knowledge and refer to exactly one lane boundary point. Thus, we avoid any association problem and can simply sum up all measurement residuals for all lane boundary points $\mathbf{x}_i^k$ neighbored to the primary point $\mathbf{x}_i$ (including the primary point itself):

$$\mathbf{c}_{\text{Meas.}}(\mathbf{x}_i) = \sum_k \left(\textstyle\sum_j \mathbf{r}^2_{\text{2D},j}\big(\mathbf{x}_i^k\big) + \mathbf{r}^2_{\text{Terrain}}\big(\mathbf{x}_i^k\big) + \textstyle\sum_j \mathbf{r}^2_{\text{3D},j}\big(\mathbf{x}_i^k\big) \right) \quad (5)$$

1) 2D Measurements: For every image measurement j of lane boundary point $\mathbf{x}_i^k$ a residual $\mathbf{r}_{\text{2D,j}}\big(\mathbf{x}_i^k\big)$ is evaluated. These measurements can originate from different time steps, different camera images, as well as different measurement approaches. We propose different image measurement approaches for roads and dirt roads in Section IV. They are integrated the same way: For all of them, the agreement between the estimated lane boundary point and the image measurements is determined considering the reprojection error in the image.

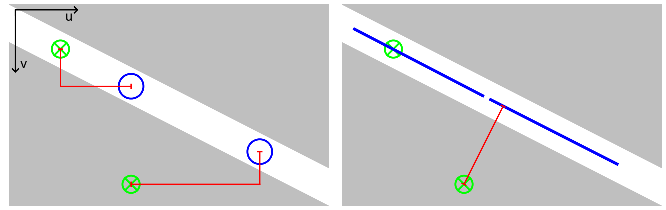
Given the pose of the ego vehicle at image capture time by the homogeneous transformation matrix ${}^{\text{veh}}\mathbf{H}^j_{\text{odom}}$ from the odometry coordinate system to the vehicle coordinate system, and the extrinsic and intrinsic calibration of the respective camera with ${}^{\text{cam}_j}\mathbf{H}_{\text{veh}}$ and ${}^{\text{img}_j}\mathbf{P}_{\text{cam}_j}$, the estimated lane boundary point is projected into the image:

$$ {}^{\text{veh}}\mathbf{p}_i^{*^k} = {}^{\text{veh}}\mathbf{H}^j_{\text{odom}} \cdot {}^{\text{odom}}\mathbf{p}_i^k \quad (6)$$

$$ {}^{\text{cam}_j}\mathbf{p}_i^{*^k} = {}^{\text{cam}_j}\mathbf{H}_{\text{veh}} \cdot {}^{\text{veh}}\mathbf{p}_i^{*^k} \quad (7)$$

$$ {}^{\text{img}_j}\mathbf{p}_i^{*^k} = {}^{\text{img}_j}\mathbf{P}_{\text{cam}_j} \cdot {}^{\text{cam}_j}\mathbf{p}_i^{*^k} \quad (8)$$

This way we obtain a prediction ${}^{\text{img}_j}\mathbf{p}_i^{*^k}$ for the image position of lane boundary point $\mathbf{x}_i^k$ in the image corresponding to measurement j. This predicted image position should match the measured image position.



(a) Point Measurements **(b)** Line Measurements

Fig. 4: Visualization of the image measurement residuals. Green: Predicted lane boundary points. Blue: Measurements. Red: Residuals to be minimized. Using point measurements (a), the upper point has a residual greater than zero even though it lies perfectly on the lane marking. This is resolved by using line measurements (b).

Two different cost functions are possible depending on the kind of measurement:

a) Point Measurement: The image coordinates can be compared straightforward to those of a measured point ${}^{\text{img}_j}\mathbf{y}_i^k$ using the residual function

$$\mathbf{r}_{\text{2D},j}\big(\mathbf{x}_i^k\big) = {}^{\text{img}_j}\mathbf{p}_i^{*^k} - {}^{\text{img}_j}\mathbf{y}_i^k. \quad (9)$$

As it is shown in Fig. 4a, this directly punishes deviations in u and v coordinates.

b) Line Measurement: Since it is difficult to recognize one concrete point of a continuous lane boundary, most approaches are not able to measure the exact position of a lane boundary point on an observed edge. Choosing an arbitrary point on the edge results in a wrong error as it is shown in Fig. 4. Thus, it is beneficial to measure a line instead of a point and consider the deviations of the predicted image position from that line. Given the line ${}^{\text{img}_j}\mathbf{g}_i^k$, the residual

$$\mathbf{r}_{\text{2D},j}\big(\mathbf{x}_i^k\big) = \mathrm{d}\Big({}^{\text{img}_j}\mathbf{p}_i^{*^k},\, {}^{\text{img}_j}\mathbf{g}_i^k\Big), \quad (10)$$

is evaluated, with d being the orthogonal distance of ${}^{\text{img}_j}\mathbf{p}_i^{*^k}$ to ${}^{\text{img}_j}\mathbf{g}_i^k$.

2) Terrain Height Measurements: Regardless of whether a point or a line was measured in an image, we are missing one degree of freedom as we do not know its distance from the camera along the pixel ray. However, we can take advantage of the fact that the road is on the ground and get the missing information based on the estimated ground height from a terrain estimation [14]. Hence, the terrain residual

$$\mathbf{r}_{\text{Terrain}}\big(\mathbf{x}_i^k\big) = \left(p_z - h_{\text{Terrain}}(p_x, p_y)\right) \cdot I_{\text{Terrain}}(p_x, p_y) \cdot w_{[\text{Terrain}]} \quad (11)$$

models the deviation of the z-coordinate of a lane boundary point ${}^{\text{odom}}\mathbf{p}_i^k = \big(p_x\ p_y\ p_z\big)^{\mathsf{T}}$ to the estimated terrain height h_{Terrain} with the corresponding estimated information (inverse covariance) I_{Terrain} at the respective position. The weighting factors w (including the weights used in Section III-B) have to be chosen according to the individual setup and application.

Taking together 2D image and terrain measurements, the lane boundary points are fully determined in 3D space.

3) 3D Measurements: Additionally or alternatively, 3D measurements, e. g., the measurement of a lane marking in the intensities of a LiDAR sensor or the measurement of the road border in a (colored) terrain grid, could be used. However, we will not go into that in this paper.

B. Internal Information

The advantage of using maximum a posteriori estimation over the Kalman Filter is that prior distributions are not limited to Gaussian distributions over the state vector. Instead, complex, non-linear functions can be used to represent the prior distribution. The corresponding cost function is composed of a clothoid consistency and a smoothness part:

$$\mathbf{c}_{\text{Int.}}(\mathbf{x}_i) = \mathbf{r}^2_{\text{Cloth.}}(\mathbf{x}_i) + \mathbf{r}^2_{\text{Smoothn.}}(\mathbf{x}_i) \tag{12}$$

a) Clothoid Consistency: The first part of Eq. (12) ensures consistency between two adjacent lane boundary points according to the clothoid model. To calculate the residual values, the parameters of the current point i are used to predict position and orientation of the following point $i+1$

$$^{\text{odom}}\mathbf{p}^*_{i+1} = {^{\text{odom}}}\mathbf{p}_i + {^{\text{odom}}}\mathbf{q}_i \cdot \Delta\mathbf{p}_{i\to i+1} \tag{13}$$

$$^{\text{odom}}\mathbf{q}^*_{i+1} = {^{\text{odom}}}\mathbf{q}_i \cdot \Delta\mathbf{q}_{i\to i+1}, \tag{14}$$

using the clothoid geometry model [11, p. 207]

$$\Delta\mathbf{p}_{i\to i+1} = \left(l,\ \tfrac{1}{2}c_i l^2 + \tfrac{1}{6}\mathring{c}_i l^3,\ 0\right)^\top \tag{15}$$

$$\Delta\mathbf{q}_{i\to i+1} = \left(\cos(\tfrac{\psi}{2}),\ 0,\ 0,\ \sin(\tfrac{\psi}{2})\right)^\top \tag{16}$$

$$\text{with } \psi = c_i \cdot l + \frac{1}{2}\mathring{c}_i \cdot l^2 \tag{17}$$

The residual vector itself is then given by:

$$\mathbf{r}_{\text{Cloth.},[1,2,3]} = \left({^{\text{odom}}}\mathbf{p}^*_i - {^{\text{odom}}}\mathbf{p}_i\right) \cdot w_{[\text{Cloth.Pos.}]} \tag{18}$$

$$\mathbf{r}_{\text{Cloth.},[4,5,6]} = \text{euler}\left({^{\text{odom}}}\mathbf{q}^*_i \cdot {^{\text{odom}}}\mathbf{q}^{-1}_i\right) \cdot w_{[\text{Cloth.Orient.}]} \tag{19}$$

Please note that the residual values for the orientation error are used in form of euler angles.

b) Smoothness: The second part of the internal residual function ensures a certain degree of smoothness and thus prevents unsteady jumps in the road course. The respective residuals penalize any change in road width (of all secondary lane boundaries k) and curvature between two successive lane boundary points:

$$\mathbf{r}_{\text{Smoothn.},[1]} = \sum_k \left(d^k_i - d^k_{i+1}\right) \cdot w_{[\text{SmoothLaneWidth}]} \tag{20}$$

$$\mathbf{r}_{\text{Smoothn.},[2]} = \left(c_i - c_{i+1}\right) \cdot w_{[\text{SmoothCurvature}]} \tag{21}$$

C. Constraints

In addition to the internal information, limitations of the state variables are modeled, since not all value ranges of the state variables are equally likely. For example, a minimum and a maximum lane width, as well as a maximum curvature, can be assumed. We model these limitations in form of an additional part of the cost function representing the respective prior distributions.

The residual function to limit a parameter value p within the range of $[\overline{p}_{\text{lim}} - \Delta p_{\text{lim}}; \overline{p}_{\text{lim}} + \Delta p_{\text{lim}}]$ is

$$r_{\text{lim}}(p) = \exp\left[\left(|p - \overline{p}_{\text{lim}}| - \Delta p_{\text{lim}}\right) \cdot w_{\text{lim}}\right]. \tag{22}$$

Here w_{lim} is a weighting parameter used to set the harshness of the limit. An exemplary resulting cost function and the respective probability distribution are visualized in Fig. 5.

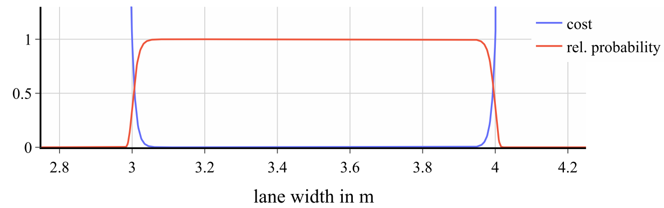


Fig. 5: The implemented model of parameter limitations with $w_{\text{lim}} = 50$. The blue graph shows the cost function for the lane width and the red graph the respective relative probability distribution. Lane widths between 3 and 4 meters are practically equally likely, whereas larger or smaller lane widths are associated with very high costs and are accordingly modeled to be extremely unlikely.

This kind of "soft" constraint enables better convergence behavior compared to strict, hard constrains due to its continuous gradients. The actually implemented cost function $\mathbf{c}_{\text{Constr.}}(\mathbf{x}^k_i)$ consists of limitation residuals in the form of Eq. (22) for the lane width d and the curvature c of all points $\mathbf{x}^k_i$ corresponding to lane boundary point i.

D. Implementation

Using the Ceres Solver [15], one single optimization problem for all lane boundary points in front of the car is solved. To ensure a minimum look-ahead, the lane boundaries are extended to the front using the estimated attributes of the frontmost point to predict a new lane boundary point.

IV. Image Measurements

This section presents two different measurements of road and dirt road borders in images for use with our tracker.

A. Edge Detection using CRONOS

CRONOS [10] is a very simple but effective edge detection method from the beginning of autonomous driving [11]. It is efficiently detecting edges of a particular direction by moving a rotated Prewitt convolution mask over an image window and evaluating the mask responses. If the mask is rotated according to the expected angle of a searched edge, it will not respond to unrelated edges in other directions (see Fig. 6).

To measure the lane boundary positions using CRONOS, the predicted lane boundary points are projected into the image. They are used to set up measurement windows and to calculate its expected edge angle, not only to rotate the convolution mask but also to decide whether we set up a horizontal and/or a vertical measurement window.

For inner lane boundaries, which are usually marked, we are explicitly looking for white lane markings. In the mask responses, they can be recognized by two successive, strong, dark-bright and bright-dark edges in a reasonable distance. As the width of lane markings in meters is standardized, the expected distance in pixels can be calculated.

For the outer road boundaries, we do not rely on white lane markings. On the one hand, with white posts or snow next to the road, looking for white blobs is prone to errors. On the other hand, roads or especially dirt roads do not need to be marked at all. Thus, for outer lane boundaries, simply selecting the innermost edge of a certain amplitude proved to be the most robust approach.

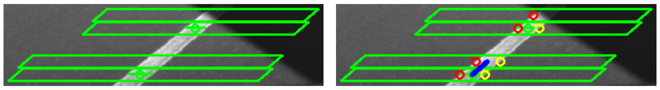
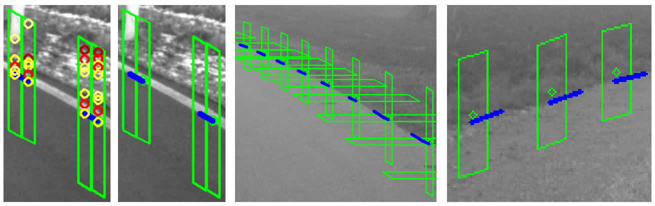


Fig. 6: Example for measuring an inner lane boundary: Predicted lane boundary points (circles), measurement windows, and expected lane marking width (lines) are shown in green. CRONOS responses are shown in red (dark-bright) and yellow (bright-dark). The blue line shows the resulting measurement. Note the two measurement windows per point. In the uppermost window the bright-dark edge is not detected, as it appears in another direction due to a shadow.



(a) (b) (c)

Fig. 7: Examples for measuring outer lane boundaries: (a) A lot of wrong CRONOS mask responses next to the road, handled by using the innermost edge. (b) Horizontal and vertical measurement windows for the same lane boundary points of an unmarked road. For clarity, the second measurement window for the line measurement is not displayed. (c) Measurements of a dirt road boundary. To decrease sensitivity for details, the image resolution is reduced.

The result of each (successful) measurement is one image point. As described in Section III-A.1, it is beneficial to measure image lines. Thus, we set up a second measurement window with a slight offset in distance for every lane boundary point. If an edge point of the same direction is found in both windows, we obtain a line measurement for the corresponding lane boundary point. Using two measurement windows also reduces the number of outliers, as the edge has to be found in both windows and, e. g., dirt will not show up in both windows at the same angle. Furthermore, by comparing the expected edge angle to the angle between the two points measured, remaining outliers can be easily rejected.

Figures 6 and 7 show different examples for measurements of inner and outer lane boundaries using CRONOS.

B. *CNN-based Similarity Measurement*

In order to obtain robust measurements on challenging dirt roads, we have developed a new measurement method. It is based on the observation that although the textures of dirt roads can appear quite different, they always have very similar image structures within an image, usually clearly different from the surrounding area. Thus, the fundamental idea is to compare image structures that are most likely to be part of the road with the image structures of the entire image. The full measurement process is depicted in Fig. 8.

The first step is to transform the input image into a 256-dimensional feature map using the first 3 blocks (i.e., 13 layers) of a ResNet-18 CNN [16]. It is downsampling the spatial image dimensions by a factor of 16.

The projections of the currently estimated lane boundary points (compare Section III-A.1) are used to derive image areas being highly probably part of the path (shown in green in Fig. 8). This area is used to generate the reference feature vector by averaging the respective pixels of the feature map.

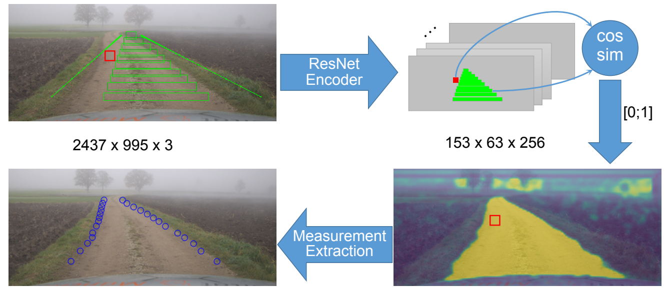


Fig. 8: Overview of the proposed CNN-based similarity measurement process. The reference feature vector is extracted from the areas marked in green. The pixel marked red in the feature map is an example of a pixel being compared with the reference vector. Due to the strong visual proximity of the image area to the rest of the trail, a similarity score of approximately one is obtained. The lower part of the figure illustrates the second step, the extraction of measurements for the given lane boundary points from the score image. Blue circles visualize the resulting image measurements.

Then, scores between 0 and 1 are obtained by calculating the cosine similarity between every pixel and the reference feature vector. In a second step, measurements for the visible lane boundary points are extracted from the resulting score image by finding transitions from low to high score values.

The relatively simple processing pipeline and the fact that the feature map only needs to be computed once for the entire image make the method very efficient and real-time capable. By computing the reference feature vector from the current image, the method is highly flexible and can be used to detect a wide variety of paths. Figure 9 shows some examples.

Annotation and training also turn out to be highly efficient. Image pixels just need to be annotated as "road" or "non-road". During the training of the CNN, the Cosine Embedding Loss[1] is used to learn an embedding that has the largest possible differences between "road" and "non-road" regions. We found even a very small training dataset of only 20 images to be sufficient for a large variability of scenes (see also Fig. 9). This makes the measurement method very attractive for road and path tracking.

[1] `https://pytorch.org/docs/stable/generated/torch.nn.CosineEmbeddingLoss.html`

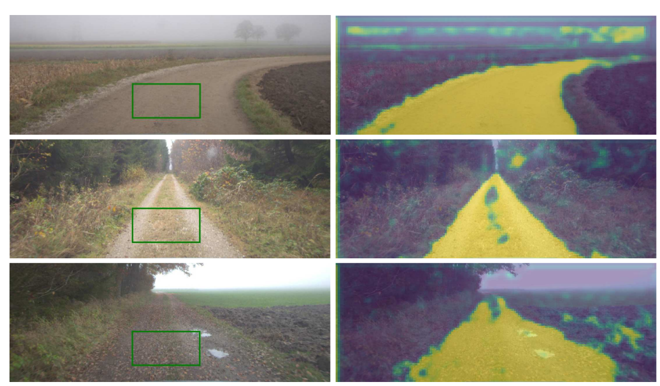


Fig. 9: Examples for the CNN-based similarity score on test images of unseen dirt roads. The reference feature vector is extracted from the area marked in green. The similarity between the single pixels and the reference feature vector is shown in an overlay from yellow for high similarity to blue for low similarity.

V. Experimental Results

This section shows that the proposed road tracking system is able to track different roads robustly.

A. Timing

Figure 10 illustrates the typical computation times of the single components, indicating that the cycle time of our image capturing process of 100 ms is not being exceeded. Thus, real-time road tracking is possible. All presented results were obtained using a single core of an Intel© Core™ i7-8700K CPU running at 3.70 GHz.

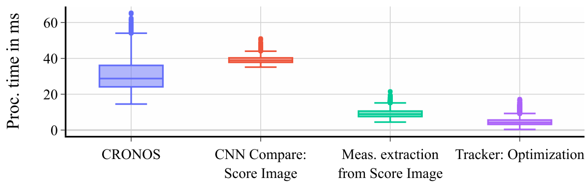


Fig. 10: Overview of the computing times of the various modules making up the tracking system.

B. Quantitative Results

A quantitative evaluation of our tracking system can be found in Fig. 11. Here, the tracking accuracy of the individual lane boundaries is compared in dependence on the ego-relative distance for a marked road and a dirt road. Both scenes include sharp curves with very limited visibility and partially extremely challenging lighting conditions. Nevertheless, the system is able to locate the road precisely.

As expected, the accuracy on a marked road is significantly better. However, due to the new adaptive measurement method, it is possible to track the dirt road with sufficient accuracy.

In the case of the marked two-lane road, the accuracy of the left lane boundary is significantly worse than that of the other two. This difference is because we are driving exactly between the right and the center lane boundary. The left lane boundary is significantly further away and, therefore, more difficult to observe.

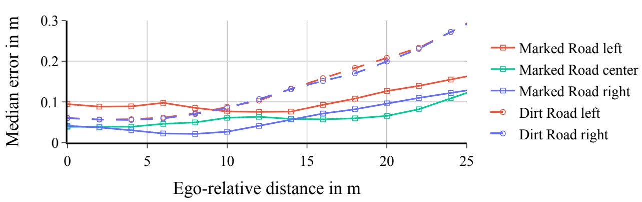


Fig. 11: Evaluation of the position accuracy of the tracked lane boundaries in dependence on the ego-relative distance for both a marked tarred road and a rural dirt road. The evaluation includes every single time-step, i.e., a world-fixed lane boundary point occurs at multiple ego-relative distances over time as we approach it.

The ground truth reference data used here has been obtained by a large-scale optimization process, including ego vehicle positions from a high accuracy INS system with RTK-GNSS and manually labeled lane boundaries in camera images.

The road courses evaluated here as well as the tracking of further scenes can be found in our demo video, available at `https://www.mucar3.de/iv2021-road-tracking`.

VI. Conclusion

In this paper, we present a flexible framework based on connected clothoid pieces, as well as two image-based measurements, for the combined tracking of paved roads and dirt roads. Future work will integrate 3D measurements.

While we show how to use traditional edge detection methods to robustly measure road boundaries using prior knowledge, we need a more sophisticated measurement for dirt roads. Based on a feature map extracted by ResNet, the cosine similarity of image pixels to a reference feature vector from a known road area is considered. This way, a score image is obtained, from which measurements are extracted.

We show the effectiveness of our approach in quantitative and qualitative results.

ACKNOWLEDGMENT

The authors gratefully acknowledge funding by the Federal Office of Bundeswehr Equipment, Information Technology and In-Service Support (BAAINBw).

References

[1] E. D. Dickmanns and B. D. Mysliwetz, “Recursive 3-D Road and Relative Ego-State Recognition,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, no. 2, pp. 199–213, 1992.

[2] A. Abramov, C. Bayer, C. Heller, and C. Loy, “A Flexible Modeling Approach for Robust Multi-Lane Road Estimation,” in *Proc. IEEE Intelligent Vehicles Symp. (IV)*, 2017, pp. 1386–1392.

[3] M. Manz, M. Himmelsbach, T. Luettel, and H.-J. Wuensche, “Detection and Tracking of Road Networks in Rural Terrain By Fusing Vision and LIDAR,” in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Syst. (IROS)*, 2011.

[4] J. Li, X. Mei, D. Prokhorov, and D. Tao, “Deep Neural Network for Structural Prediction and Lane Detection in Traffic Scene,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 3, pp. 690–703, 2017.

[5] A. Borkar, M. Hayes, and M. T. Smith, “Robust Lane Detection and Tracking with RANSAC and Kalman Filter,” in *Proc. IEEE Int. Conf. Image Processing (ICIP)*, 2009, pp. 3261–3264.

[6] P. Fleischmann, J. Kneip, and K. Berns, “An Adaptive Detection Approach for Autonomous Forest Path Following using Stereo Vision,” 2016, pp. 1–6.

[7] A. Abramov, C. Bayer, C. Heller, and C. Loy, “Multi-Lane Perception Using Feature Fusion Based on GraphSLAM,” in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Syst. (IROS)*, 2016, pp. 3108–3115.

[8] D. Neven, B. De Brabandere, S. Georgoulis, M. Proesmans, and L. Van Gool, “Towards End-to-End Lane Detection: an Instance Segmentation Approach,” in *Proc. IEEE Intelligent Vehicles Symp. (IV)*, 2018, pp. 286–291.

[9] Z. Qin, H. Wang, and X. Li, “Ultra Fast Structure-aware Deep Lane Detection,” in *Proc. European Conf. Comput. Vision (ECCV)*, 2020.

[10] B. Mysliwetz, “Parallelrechner-basierte Bildfolgen-Interpretation zur autonomen Fahrzeugführung,” Dissertation, Universität der Bundeswehr München, Fakultät für Luft- und Raumfahrttechik, Neubiberg, 1990.

[11] E. D. Dickmanns, *Dynamic Vision for Perception and Control of Motion*. Springer Verlag, 2007.

[12] S. Thrun, M. Montemerlo, H. Dahlkamp, D. Stavens, A. Aron, J. Diebel, P. Fong, J. Gale, M. Halpenny, G. Hoffmann *et al.*, “Stanley: The Robot That Won The DARPA Grand Challenge,” *J. Field Robotics*, vol. 23, no. 9, pp. 661–692, 2006.

[13] S. F. X. Bayerl and H.-J. Wuensche, “Detection and Tracking of Rural Crossroads Combining Vision and LiDAR Measurements,” in *Proc. IEEE Intelligent Transportation Syst. Conf. (ITSC)*, 2014.

[14] B. Forkel, J. Kallwies, and H.-J. Wuensche, “Probabilistic Terrain Estimation for Autonomous Off-Road Driving,” in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, 2021.

[15] S. Agarwal, K. Mierle, and Others, “Ceres Solver,” http://ceres-solver.org, 2017.

[16] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proc. IEEE Conf. Comput. Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.