(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
28 November 2019 (28.11.2019)

WIPO | PCT

(10) International Publication Number
WO 2019/224103 A1

(51) International Patent Classification:
*G06T 7/246* (2017.01) *G06K 9/00* (2006.01)
*G06T 7/277* (2017.01)

(21) International Application Number:
PCT/EP2019/062641

(22) International Filing Date:
16 May 2019 (16.05.2019)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
10 2018 112 177.6 22 May 2018 (22.05.2018) DE

(71) Applicant: CONNAUGHT ELECTRONICS LTD. [IE/IE]; Dunmore Road, Tuam, County Galway (IE).

(72) Inventors: NOGUERIA-OUTERELO, Manuel; c/o CTAG, Poligono Industrial A Granxa Calle A, parcelas 249-250, 36400 Porrino (Pontevedra) (ES). WESTHUES, Andreas; Auer Straße 54, 09366 Stollberg (DE). HUGHES, Ciaran; c/o Connaught Electronics Ltd., Dunmore Road, County Galway, Tuam (IE). STARR, Michael; c/o Connaught Electronics Ltd., Dunmore Road, County Galway, Tuam (IE).

(74) Agent: JAUREGUI URBAHN, Kristian; Laiernstr. 12, 74321 Bietigheim-Bissingen (DE).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

**Published:**
— *with international search report (Art. 21(3))*

**(54) Title:** LANE DETECTION BASED ON LANE MODELS

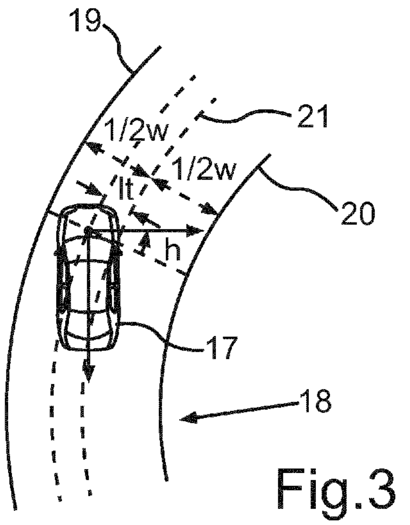



Fig.3

**(57) Abstract:** An improved tracking of a lane for a vehicle shall be provided. Thus, there is proposed a method including the steps of providing frames of the lane, extracting lane features (8) from the frames and providing a first lane model (10). Then, the first lane model (10) is updated (12) with the lane features. A second lane model (13) based on an outlier detection algorithm is provided. A tracked confidence value is calculated for the updated first lane model and a static confidence value for the static second lane model (13). On the basis of both confidence values it is decided whether to use the updated first lane model or the static second lane model for calculating actual information about the lane.

WO 2019/224103 A1

# Lane detection based on lane models

The present invention relates to a method of tracking a lane including the steps of providing frames of the lane, extracting lane features from the frames and providing a lane model. Furthermore, the present invention relates to a driver assistant system and a device each being configured to perform such a method. Additionally, the present invention is directed to a computer program product which is capable of performing the method of tracking a lane.

Diverse driver assistant systems are known from the prior art. Such driver assistant systems can have different automation levels, starting from level 1 such as for example the ACC (Adaptive Cruise Control), up to level 5, fully autonomous driving. For implementing these driving functions assisting the driver or also fully autonomous driving functions, it is required that such systems are able to detect objects, obstacles or other traffic participants ahead. In addition, it is also important in most of the systems to estimate whether such objects or obstacles are located in the lane or the driving area presumably to be travelled by the ego vehicle, for example to adapt the speed and/or to maintain a certain safety distance to a preceding vehicle. For example, if a preceding vehicle is detected, but it is not in the lane of the ego vehicle, thus, it either is not required to warn the driver or to automatically decelerate the vehicle in such a case. Therefore, it is required to know not only at a time whether objects ahead such as for example preceding vehicles are basically present, but also to know at a time as reliably as possible whether or not a vehicle preceding the ego vehicle is in the lane area to be travelled by the ego vehicle. Other specific applications for using a method of tracking a lane are possible.

Lane detection using camera systems was one of the first computer vision ADAS (Advanced Driver Assistance System) functions developed. Such lane detection systems may be based on lane models for multi-camera and fish eye camera systems.

Document WO 2013/022153 A1 describes an apparatus and a method for detecting a lane. The apparatus includes a camera module capturing an image. Further, the apparatus includes a control unit extracting a plurality of feature points from the image, carrying out lane fitting to connect the plurality of feature points with a single line and tracking the lane fitted. A display unit displays the lane tracked, wherein the lane fitting includes carrying out short-range fitting on the basis of feature points present in a short-range region among the plurality of feature points, determining an off-set representing lateral inclination of the lane on the basis of a result of the short-range fitting, and carrying out curve fitting on the basis of the offset.

Furthermore, document US 2013/0141520 A1 discloses a lane tracking system with a camera. The camera is configured to receive images of a road from a wide-angle field of view. One or more lane boundaries are detected, wherein each lane boundary includes a plurality of lane boundary points. A reliability-weighted model lane line is fitted to the plurality of points.

The object of the present invention is to provide a method of tracking a lane in an enhanced quality. Furthermore, a corresponding device and computer program product shall be provided.

According to the present invention this object is solved by a method, a device and a computer program product as defined in the independent claims.

Thus, there is provided a method of tracking a lane including the steps of

- providing frames of the lane,
- extracting lane features from the frames and
- providing a first lane model,
  as well as
- updating the first lane model with the lane features,
- providing a static second lane model based on an outlier detection algorithm,
- calculate a tracked confidence value for the updated first lane model and a static confidence value for the static second lane model and
- deciding on the basis of both confidence values whether to use the updated first lane model or the static second lane model for calculating actual information about the lane.

For instance, a vehicle driving on or at a lane may have to track the lane in order to provide assistance information for the driver as actual information about the lane. The vehicle may include one or more cameras (e.g. a multi-camera system or a fish eye camera system) for providing images (also called frames) of the lane. Lane features like heading angle, lateral position (with respect to the vehicle), lane width and curvature, are extracted from the frames. A first lane model for preferably predicting a lane position etc.

is provided. The first lane model is updated with actual lane features (like heading angle, lateral position, lane width or curvature).

Moreover, there is provided a static second lane model. The feature “static” means that the model is not updated with actual lane features. The second lane model is rather based on a fixed outlier detection algorithm. This algorithm distinguishes between outliers and inliers in accordance with pre-given criteria. With respect to the extracted lane features a confidence value is calculated for each model, for example on the basis of a statistical hypothesis. Specifically, there is calculated a tracked confidence value for the updated first lane model and a static confidence value for the static second lane model. With the aid of these confidence values it is decided to use the updated first lane model or the static second lane model for further calculations with respect to the lane tracked. Preferably, that model is used which delivers the higher confidence value. This guarantees that the calculated actual information about the lane (for instance the lane coordinates) is more precise.

In a preferred embodiment the updating of the first lane model is performed by predicting a new version of the first lane model with the help of Kalman filtering and matching the new version of the first lane model with the extracted lane features. Such procedure enables to easily update the model with a plurality of feature elements. These feature elements describe the lane and form a state vector for the Kalman filter.

Preferably, the first lane model and the second lane model track a lane geometry over time. Thus the lane models preferably provide actual information about lane dimensions, lane position and/or lane orientation. Such information may be classified with respect to the type of the lane.

In a further favourable development the lane features relate to lane markers and before updating the first lane model with lane markers, the lane markers are split into two separate sets for a left boundary and right boundary by calculating a separation line with the help of the first lane model. Preferably, the separation line is a second order polynomial. The sign of the lateral distance from the separation line may be used to classify the lane markers into left lane markers and right lane markers. Such separation into a left and right lane boundary ensures a high quality when fitting the corresponding lane model.

In a further embodiment the static second lane model is evaluated based on a best fit lane hypothesis for the extracted lane features by using a combination of random sample consensus and least square fit. Particularly, a specific lane model hypothesis may be calculated for every frame. This may be done in two steps: Random Sample Consensus (RANSAC) and least square fit. This results in actual model parameters which are used for the static lane model.

According to a further embodiment for each of a left and right boundary of the lane a pre-given number of features is used to calculate the static second lane model by performing polynomial fitting. For instance, six features are picked randomly for each boundary for calculating a best fit candidate model.

According to a further embodiment one of the first lane model and the second lane model is used in a first time step and this one model is still used in a second time step immediately following the first time step, if the direct confidence value and the static confidence value are below a pre-given threshold. This means that the lane model is fixed between two time steps. Consequently, a change of the lane model is only performed if the respective confidence value is high enough, i.e. is lying above the pre-given threshold.

Moreover, the direct confidence value may be calculated for a hypothesis related to the first lane model and the static confidence value is calculated for a hypothesis related to the second lane model on the basis of the lane features from the frames. Thus, lane markers can be reliably classified with respect to the two different models.

In another embodiment the confidence value for each of the first and second lane model is calculated by first calculating independent confidence values for the left and right boundary of the lane and then calculating an average of these independent confidence values for said confidence value for the respective first or second lane model. Thus, the different lane models can be weighted and compared. Specifically, for each model hypothesis a confidence value may be calculated in order to weight and compare the models. The average as a common confidence value is more reliable than each single confidence value.

In a preferred embodiment detected lane markers are separated into inliers and outliers with respect to the first lane model and only the inliers are used to update the first lane model. In this case a model fit is only performed with inlier markers. As a result the model fit has an improved quality.

The above-mentioned object may also be solved by a driver assistant system being designed to perform a method according to any method of tracking a lane as described above. Any other device for tracking a lane may be configured to perform the described methods. Modifications and advantages of the inventive method may also apply to the inventive driver assistant system or the inventive device.

Additionally, the above-object is also solved by a computer program product with program code means, which are in particular stored in a computer-readable medium, to perform the method of tracking a lane according to any one of the preceding claims when the computer program product is run on a computer device of an electronic control unit.

Further features of the invention are apparent from the claims, the Figures and the description of Figures. The features and feature combinations mentioned above in the description as well as the features and feature combinations mentioned below in the description of Figures and/or shown in the Figures alone are usable not only in the respectively specified combination, but also in other combinations without departing from the scope of the invention. Thus, implementations are also to be considered as encompassed and disclosed by the invention, which are not explicitly shown in the Figures and explained, but arise from and can be generated by separated feature combinations from the explained implementations. Implementations and feature combinations are also to be considered as disclosed, which thus do not have all of the features of an originally formulated independent claim. Moreover, implementations and feature combinations are to be considered as disclosed, in particular by the implementations set out above, which extend beyond or deviate from the feature combinations set out in the relations of the claims.

The present invention will now be described in more detail in connection with the attached Figures showing in:

Fig. 1 a principle block diagram of a lane sensing system;

Fig. 2 a block diagram of the processing scheme of a lane model module;

Fig. 3 a diagram of a lane model;

Fig. 4 a diagram for the separation of lane markers;

Fig. 5 a diagram for ego motion compensation and

Fig. 6 an activity diagram for the lane model module of Fig. 2.

The following embodiments represent preferred examples of the present invention.

Fig. 1 shows a lane-sensing system. A system input module 1 provides a front image, a left image, a right image and a rear image, for example. These images are sent to a lane mark detection module which detects lane marks from the images (herein also called frames). A transformation module 3 transforms the image information of the lane mark detection module 2 into world coordinates, for instance. Afterwards a feature tracking module 4 tracks the features like lane marks. For this analysis the feature tracking module 4 may receive vehicle geometry data from system input module 1.

The vehicle geometry information from system input module 1 together with feature tracking information from feature tracking module 4 are input into a lane position logic module 5, which also can be called lane tracking module. Thus the lane position logic module 5 receives information about detected lane marks (i.e. bounding boxes and inner feature positions, for example) preferably in vehicle coordinates. From this information the module 5 tries to fit a lane that consists of preferably two parallel second order polynomials as left and right boundaries and the position and orientation of the ego-vehicle within this lane. Furthermore, the module 5 may track the road geometry over time. The output of module 5 is passed to a lane boundary classification module 6 which classifies the type and colour of the lane boundary, for example. In a system output module 7 a lane type from the lane boundary classification module 6 and an estimated lane geometry for the current time stamp may be provided as results from the whole lane sensing system.

Fig. 2 shows a processing scheme of the lane position logic module or lane tracking module 5 of Fig. 1. Data segments 8 from the feature tracking module 4 are input into a matching unit 9 which matches a predicted lane from a lane prediction unit 10 with the data segments 8 from the feature tracking module 4. The lane prediction unit 10 is based on a first lane model. It may receive odometry data from an odometry unit 11. The matching unit 9 delivers information for an updating unit 12 which updates the lane tracking, i.e. it updates the first line model.

In parallel a computation unit 13 computes a best model from the measurements. Specifically, it calculates for each time stamp a best fit lane hypothesis from the incoming lane marker features, i.e. the data segments from feature tracking module 4. For this the computing module 13 preferably uses a combination of RANSAC and least square fit. The result may be called “static best model” (herein also called static second lane model).

A decision logic 14 uses confidence values for a pre-given hypothesis to decide whether to use the updated or tracked first model from updating unit 12 or the static second lane model from computing unit 13 to correct the lane tracking. For this correction, a correction unit 15 receives the respective information from the decision unit 14. As a result the lane model is updated for the lane prediction unit 10 or the lane model remains unchanged with respect to the previous time step. It is preferred to stick with the result of the previous time step if both the static second lane model and the tracked first lane model deliver too low confidence values. The actual lane model finally delivers output information 16 which may include classification information like the lane type or lane geometry.

A lane model class may combine two line classes for left and right boundary to a lane model. It contains methods for fitting the lane model through the 2D-feature positions. This class also calculates the overall confidence for the lane.

In the following an algorithm for the inventive method of tracking a lane is described in more detail. Preferably, the tracking of the first lane model is done with the help of Kalman filter. The Kalman filter state vector $\widehat{x_k}$, that describes the lane consists of the following feature elements:

$$\widehat{x_k} = \begin{pmatrix} \boldsymbol{heading\ angle} \\ \boldsymbol{lateral\ position} \\ \boldsymbol{lane\ width} \\ \boldsymbol{curvature} \end{pmatrix}$$

where heading angle h (compare Fig. 3) describes the rotation of the vehicle 17 within lane 18. Lane 18 has a left boundary 19 and a right boundary 20. These boundaries 19 and 20 define the lane width w of the lane 18. The lateral position lt of the vehicle 17 is the Euclidian distance of the vehicle origin from centre line 21 of the lane 18. The curvature of lane is the quadratic amount a of the second order polynomial of the road boundaries (see following description of least-square model fitting).

The state parameters are estimated based on 2D-features of the lane makers in vehicle coordinates. In contrast to what can be found in literature, in the present sensor model the 2D-features are not used directly to modify the state. Instead the 2D features are used to fit a lane for the current time stamp as an intermediate step. This fitted lane is the actual measurement for the filter update step. This leads to a sensor model in which the measurement vector $z_k$ consists of the same elements as the state vector which simplifies the observation model $H_k$ to a unit metrics. For simplification all elements of the state vector are estimated independently from each other. This means that in fact there are four independent Kalman filters – one for each element of the state vector.

The disadvantage of this model is that use is not made of any possible co-variants between the elements of the state vector. This means that the state co-variants matrix $P_k$ is in fact reduced to a variance vector $p_k$ with the same dimensions as $\widehat{x_k}$. On the other hand, the configuration of the filter is much simpler because each element of the filter can be tuned independently. The risk of over fitting problems is reduced, too.

In the tracking system (compare Fig. 2) always two measurements (i.e. two lane models) are calculated in parallel for each time stamp:

- the first lane model, i.e. the update model which is calculated by using lane marks of the current time stamp filtered by the help of the predicted lane model and
- the second lane model, i.e. the static (best) model, being calculated using lane marks of the current time stamp filtered by the help of a RANSAC

Later – in the lane position logic block 5 (compare Fig. 1) – it is decided by the help of confidence values which of the two measurements is used to update the Kalman filter state.

A separation of lane makers for left and right lane boundaries is performed before fitting the lane model. The lane markers 22 are split into two separate sets for the left and right boundary. This is done by calculating the separation line 21 from the predicted lane model. The separation line 21 is (like the model boundaries: predicted left boundary 19 and predicted right boundary 20) a second order polynomial with the same parameters as the two lane boundaries 19, 20 except that the constant amount ( c ) of the polynomial is set to mean value of the left and right boundary 19, 20 so that the line passes through the centre of the predicted lane. The sign of the lateral distance of the centre of the lane marker 22 to the separation line determines, if a lane marker 22 lies within a left boundary assignment area 23 or a right boundary assignment area 24 and thus will be assigned to the left or to the right boundary list.

In a preferred embodiment a calculation of a static lane model hypotheses is performed for the current frame. For example, the probability of the hypotheses is calculated that a lane marker of the current frame belongs to the left boundary of the lane. For every frame a best hypotheses lane model may be calculated. This is done in two steps. First a RANSAC (Random Sample Consensus) is used to select inlier features from the left and right feature sets. With these inliers a least square fit is performed to find the actual model parameters.

First, a pre-filtering using model-driven RANSAC is performed. From the list of features from the left and right boundary, n features (6 by default) are picked for each boundary. With these features a best fit candidate model is calculated using a polynomial fit (see calculations below). If a valid candidate was found, a confidence value for the fit is calculated and it is tested which of the left and right lane markers are close to the fitted polynomial. These steps are repeated until either all lane markers have been classified as inliers or the number of iterations reach a certain value which is configurable and may be 100 by default.

Furthermore, least-square model fitting is performed. This means that a model fit for each hypothesis model is done by a least square fitting method. The goal is to find a solution for the following equation:

$$\boldsymbol{Ax} = \boldsymbol{B} \tag{0.1}$$

where

$$\boldsymbol{A} = \boldsymbol{X'X} \tag{0.2}$$

and

$$\boldsymbol{B} = \boldsymbol{X'Y} \tag{0.3}$$

The solution vector $\boldsymbol{x}$ contains the polynomial coefficients of the fitted parabolic curve.

The matrix $\boldsymbol{X}$ is a $(n_l + n_r)$ x 4 matrix that is build out of the feature samples where $n_l$ the number of features for the left lane boundary is and $n_r$ is the number of features for the right lane boundary. The matrix has the following form:

$$\boldsymbol{X} = \begin{pmatrix} x_{l,1}^2 & x_{l,1} & 1 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ x_{l,n_l}^2 & x_{l,n_l} & 1 & 0 \\ x_{r,1}^2 & x_{r,1} & 0 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ x_{r,n_r}^2 & x_{r,n_r} & 0 & 1 \end{pmatrix}$$

With $x_{l,n}$and $x_{r,n}$ being the feature x-coordinates of the left and right feature positions. Matrix $\boldsymbol{Y}$ is a $(n_l + n_r)$ x 1 matrix containing the y values of the sample points:

$$\boldsymbol{Y} = \begin{pmatrix} y_{l,1} \\ \vdots \\ y_{l,n_l} \\ y_{r,1} \\ \vdots \\ y_{r,n_r} \end{pmatrix}$$

Therefore we can write for $\boldsymbol{A} = \boldsymbol{X}'\boldsymbol{X}$:

$$\boldsymbol{A} = \begin{pmatrix} \sum x^4 & \sum x^3 & \sum x_l^2 & \sum x_r^2 \\ \sum x^3 & \sum x^2 & \sum x_l & \sum x_r \\ \sum x_l^2 & \sum x_l & n_l & 0 \\ \sum x_r^2 & \sum x_r & 0 & n_r \end{pmatrix}$$

And for $\boldsymbol{B} = \boldsymbol{X}'\boldsymbol{Y}$:

$$\boldsymbol{B} = \begin{pmatrix} \sum (x^2 * y) \\ \sum x * y \\ \sum y_l \\ \sum y_r \end{pmatrix}$$

Where $x_l, x_r$ and $x$ refer to the x-positions for the left, right and both boundaries respectively and $y_l, y_r$ and $y$ refer to the y-positions for the left, right and both boundaries.

To solve equation (0.1) one can use Cramer's rule or any other valid mechanism.

A confidence value is calculated for each model hypothesis in order to weight and compare the models. The confidence value for the model is carried out by first calculating independent confidence values for the left and right boundary and then calculate the average as common confidence. The calculation of a confidence value is a usual statistical method for estimating a hypotheses concerning the belonging of a parameter to a pre-given interval.

A further essential step is to update the tracked first model. It has to be noted that the following notation $\widehat{\boldsymbol{x}}_{n|m}$ represents the estimate of the filter state at time n given observations up to and including at time m ≤ n.

Once the match of a lane model was successful, this model is kept and updated for subsequent time steps. In general, tracking with a Kalman filter is carried out in two steps: in the first step – the prediction step – the model state is transferred from the previous time stamp t-1 to the current time t by the help of a system model. In the present case the system model assumes that the lane model is fixed between two time steps. This is approximately true, if we assume small enough time increments between two successive processing steps. The error that is introduced by this assumption is indirectly inherited in the state variance $\boldsymbol{p}_{k|k-1}$ . So in the present case the prediction step consists only of the compensation of the vehicle ego motion (see next paragraph) and updating the model state variance values (i.e. increasing by the variance of the process noise $q_k$ . )

$$\boldsymbol{p}_{k|k-1} = \boldsymbol{p}_{k-1|k-1} + \boldsymbol{q}_k \qquad (0.4)$$

After the model has been transferred to the current time, the next step is to assign lane markers that have been detected in the current time step to the existing model. This step separates the lane markers in inliers and outliers with respect to the tracked model. With the inlier markers one can perform a model fit similar as for the static best match except that one does not need to perform RANSAC prior to the least square fit.

Ego motion compensation is now described in connection with Fig. 5. The left part of Fig. 5 shows vehicle 17 at time step t-1. The right part of Fig. 5 shows vehicle 17 at time step t. It follows lane 18 which lies between the left boundary 19 and the right boundary 20.

Before assigning lane markers of the current time step to the existing model the movement of the ego vehicle 17 has to be compensated. This ego motion compensation is performed in three steps:

1. From both lane boundaries 19, 20 of the t-1 lane model, select three points of the polynomial at x = -1m, x = 0m and x = +1m.

2. With the help of a given ego motion information (translation vector $\vec{s}$ and $\delta\omega$), transform these three points to the current vehicle coordinate system. This is done by first rotating the points around the new vehicle center with - $\delta\omega$ , and then shifting by -$\vec{s}$.

3. Perform a parabola fit through the three points for the left and right lane boundary 19, 20.

In a next step an association of lane markers to the existing model is analysed. This means that after the ego motion compensation has been carried out, the next step is to assign lane markers to the predicted model.

Afterwards an update-model is calculated e.g. on the basis of the following calculation steps. With the pre-filtered list of lane markers from the previous processing step a lane model is fitted with the help of the least square method. This least square operation is done in a similar way as for the static lane model but with some differences:

- If the association step before found lane marks for one of the two boundaries only, the lane is reconstructed by fitting the polynomial for this side and then mirror it to the other side with the help of the known lane width.
- In equation (0.2) of the least square estimator an additional weight for each feature is introduced. This weight is derived from, for example, the position variance of the feature or the detection confidence of the feature. In both cases a higher position variance/confidence leads to a lower weight on the result.

$$\boldsymbol{A} = \boldsymbol{X}'\boldsymbol{W}\boldsymbol{X} \tag{0.5}$$

$$\boldsymbol{A} = \begin{pmatrix} \sum wx^4 & \sum wx^3 & \sum w_l x_l^2 & \sum w_r x_r^2 \\ \sum wx^3 & \sum wx^2 & \sum w_l x_l & \sum w_r x_r \\ \sum w_l x_l^2 & \sum w_l x_l & \sum w_l & 0 \\ \sum w_r x_r^2 & \sum w_r x_r & 0 & \sum w_r \end{pmatrix}$$

where $w$ is all of the weights, $w_l$ is the weights for the left points and $w_r$ is the weights of the right weights.

In the least square estimator an additional regularisation factor is introduced for the curvature of the model, which prevents the curvature value from getting too high, i.e. overfitting of the model to the lane data is avoided. The regularization parameter is very straightforward to add. The cost function takes the additional parameters below :

$$S = \sum_{i=1}^{n} w_{r,j}\left(y_{r,i} - a_3 x_{r,i}^2 - a_2 x_{r,i} - a_1\right)^2 + \sum_{j=1}^{m} w_{l,j}\left(y_{l,j} - a_3 x_{l,j}^2 - a_2 x_{l,j} - a_0\right)^2 + \lambda \sum_{k=1}^{3} a_k^2 \quad (0.6)$$

Equivalent matrix form

$$(\boldsymbol{X}'\boldsymbol{X} + \lambda \boldsymbol{I})\boldsymbol{A} = \boldsymbol{X}^{\top}\boldsymbol{Y} \quad (0.7)$$

And with the weights described in (0.5) :

$$(\boldsymbol{X}'\boldsymbol{W}\boldsymbol{X} + \lambda \boldsymbol{I})\boldsymbol{A} = \boldsymbol{X}^{\top}\boldsymbol{W}\boldsymbol{Y} \quad (0.8)$$

See https://www.cs.ubc.ca/~schmidtm/Documents/2005_Notes_Lasso.pdf. Solving (0.8) is equivalent to minimising by (0.6) if one of the diagonals in the $\boldsymbol{I}$ above is set to zero (equivalent to $a_0$).

$$(\boldsymbol{X}^{\top}\boldsymbol{W}\boldsymbol{X} + \lambda \boldsymbol{K})\boldsymbol{A} = \boldsymbol{X}^{\top}\boldsymbol{W}\boldsymbol{Y} \quad (0.9)$$

Where K diagonal is set to zero in all elements except the one we want to suppress. For example, if we wish to suppress the curvature parameter, we would set the upper left element of *K* to 1, and all other entries to 0.

The lane position logic (LPL) block 5 (compare Fig. 1) decides whether to take the tracked first lane model or the static second (best fit) model to update the system output model based on the availability and the confidence of the calculated models. Fig. 6 shows the activity diagram for this LPL function block. The function starts at start point 25. At first an initial parameter concerning a valid update model is set to “false” in block 26. Then, in a first decision block 27 it is decided whether the input data relate to a first frame. If yes, a decision block 28 determines whether a static (best) lane model is valid. If no, the activity ends at exit 29.

If the decision in decision block 27 is no, a further decision block 30 determines whether the tracked first lane model is valid. If yes, a decision block 31 decides whether the static second (best) lane model is valid. If yes, a decision block 32 checks whether the static confidence value is higher than two times the tracked confidence value. If yes, and additionally if the decision from decision block 28 is yes, a variable concerning the update model is set to static second (best) lane model in block 33.

If the decision in decision block 30 is "no", a decision block 34 decides whether the static second (best) lane model is valid. If yes, a further decision block 35 decides whether the static confidence value is higher than the tracked confidence value. If yes, the prerequisite for block 33 is fulfilled to set the variable concerning the update model to "static second (best) value lane model".

If the decision in decision block 34 is "no" and also the decision of decision block 35 is "no", the algorithm ends at endpoint 36. The algorithm also ends at an endpoint 37 if the decision of decision block 31 is "no".

Furthermore, if the decision in decision block 32 is "no", the variable concerning the update model is set to "tracked first lane model" in block 38. As a consequence of blocks 33 and 38 the parameter relating to the presence of a valid update model is set to "true" in block 39. After block 39 the algorithm also ends at end point 36.

If necessary, a model update takes place. I.e. the last step of the tracking may be the update or correction step. Here the state vector (heading, lane width, lateral offset and curvature) is updated with the help of the selected model from the LPL block. For each of the four independent filters one can calculate four independent Kalman gain factors which may be bundled in the vector $k_k$

$$\boldsymbol{k}_k^{\text{feat}} = \frac{\boldsymbol{p}_{k|k-1}^{\text{feat}}}{\boldsymbol{p}_{k|k-1}^{\text{feat}} + \boldsymbol{r}_k^{\text{feat}}} \quad \forall \text{ feat} \in \{\text{heading, lateral offset, width, curvature}\} \tag{0.10}$$

where $\boldsymbol{r}_k^{\text{feat}}$ is the variance of the observation noise for each feature.
The Kalman gain is then used to update the new state and state variance:

$$\widehat{\boldsymbol{x}}_{k|k} = \widehat{\boldsymbol{x}}_{k|k} + \boldsymbol{k}_k(\boldsymbol{z}_k - \widehat{\boldsymbol{x}}_{k|k-1}) \tag{0.11}$$

$$\boldsymbol{p}_{k|k} = (I - \boldsymbol{k}_k)\boldsymbol{p}_{k|k-1} \tag{0.12}$$

where the measurement vector $\boldsymbol{z}_k$ used in eq. (0.11) consists of the estimated model parameters for the fitted model that have been chosen as best measurement in the LPL module (see previous section).

# Claims

1. Method of tracking a lane (18) including the steps of
   - providing frames of the lane (18),
   - extracting lane features (2) from the frames and
   - providing a first lane model (10),

   characterized by
   - updating (12) the first lane model (10) with the lane features,
   - providing a static second lane model (13) based on an outlier detection algorithm,
   - calculate a tracked confidence value for the updated first lane model and a static confidence value for the static second lane model (13) and
   - deciding (14) on the basis of both confidence values whether to use the updated first lane model or the static second lane model for calculating actual information about the lane (18).

2. Method according to claim1,
   characterized in that
   the updating (12) of the first lane model (10) is performed by predicting a new version of the first lane model with the help of Kalman filtering and matching the new version of the first lane model with the extracted lane features.

3. Method according to claim1 or 2,
   characterized in that
   the first lane model (10) and the second lane model (13) track a lane geometry over time.

4. Method according to one of the preceding claims,
   characterized in that
   the lane features relate to lane markers (22) and before updating the first lane model (10) with lane markers, the lane markers (22) are split into two separate sets for a

left boundary (19) and a right boundary (20) by calculating a separation line (21) with the help of the first lane model (10).

5. Method according to one of the preceding claims,
characterized in that
the static second lane model (13) is evaluated based on a best fit lane hypothesis for the extracted lane features by using a combination of random sample consensus and least square fit.

6. Method according to claim 5,
characterized in that
for each of a left and right boundary (19, 20) of the lane (18) a pregiven number of features is used to calculate the static second lane model (13) by performing polynomial fitting.

7. Method according to one of the preceding claims,
characterized in that
one of the first lane model (10) and the second lane model (13) is used in a first time step and this one model is still used in a second time step immediately following the first time step, if the tracked confidence value and the static confidence value are below a pregiven threshold.

8. Method according to one of the preceding claims,
characterized in that
the tracked confidence value is calculated for a hypothesis related to the first lane model (10) and the static confidence value is calculated for a hypothesis related to the second lane model (13) on the bases of the lane features from the frames.

9. Method according to claim 8,
characterized in that
the confidence value for each of the first and second lane model (10, 13) is calculated by first calculating independent confidence values for the left and right boundary (19, 20) of the lane (18) and then calculating an average of these independent confidence values as said confidence value for the respective first or second lane model (10, 13).

10. Method according to one of the preceding claims,
characterized in that
detected lane markers (22) are separated into inliers and outliers with respect to the first lane model (10) and only the inliers are used to update the first lane model (10).

11. Driver assistance system being designed to perform a method according to one of the preceding claims.

12. Device being configured to perform a method according to one of the claims 1 to 10.

13. Computer program product with program code means, which are in particular stored in a computer-readable medium, to perform the method of tracking a lane (18) according to any one of the preceding claims when the computer program product is run on a computer device of an electronic control unit.

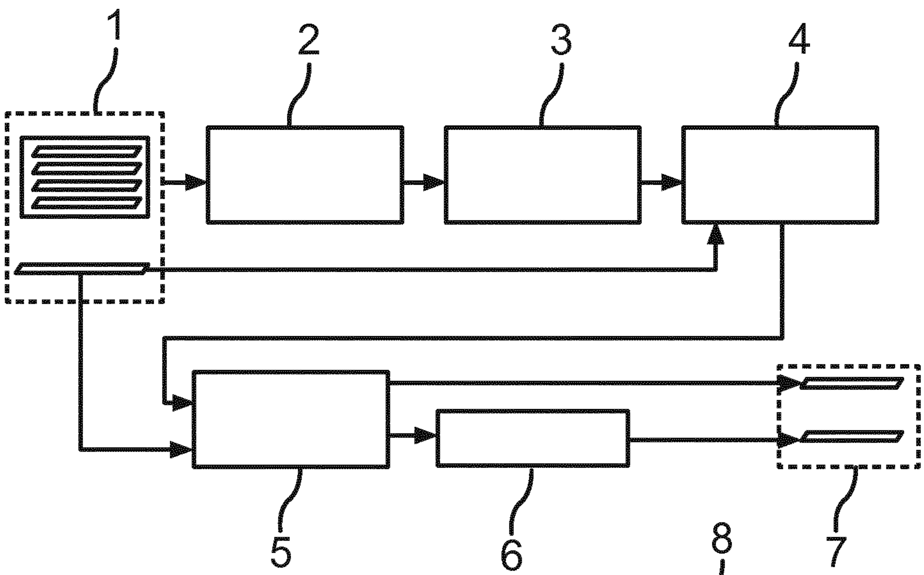



Fig.1

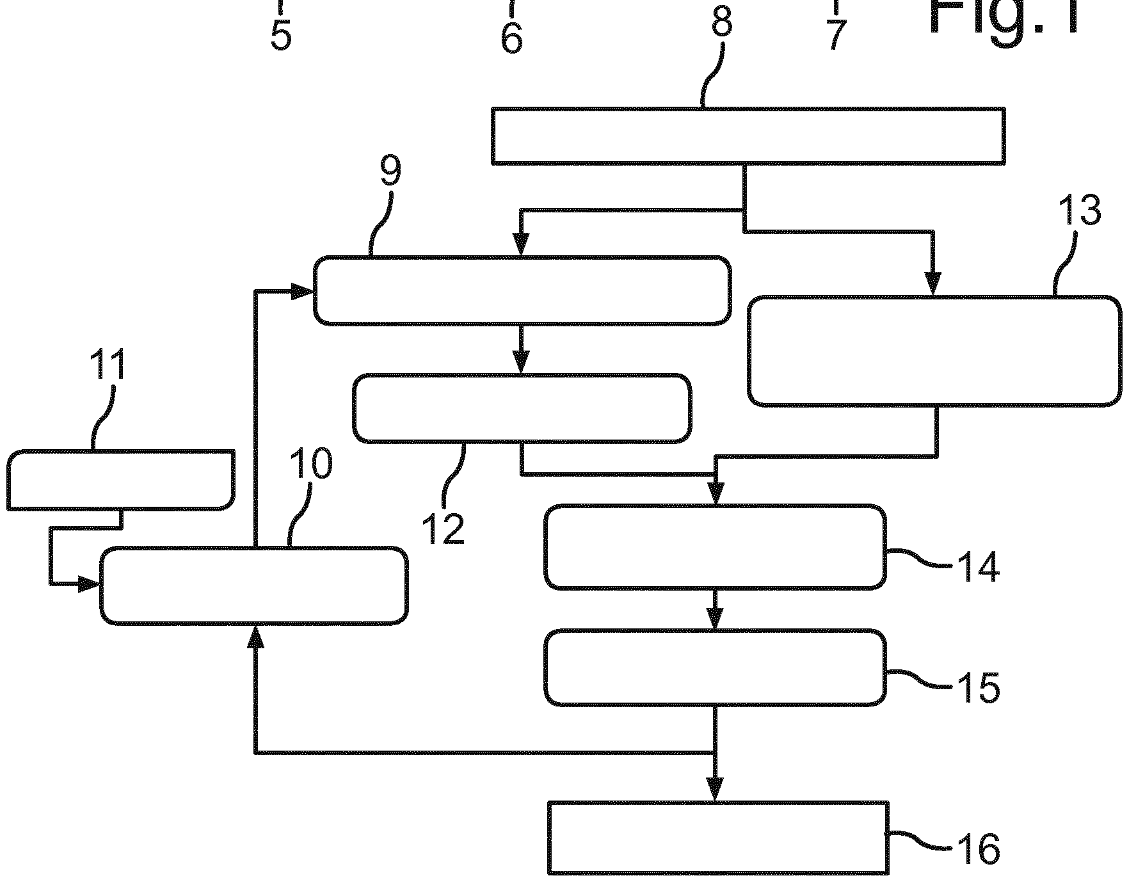



Fig.2

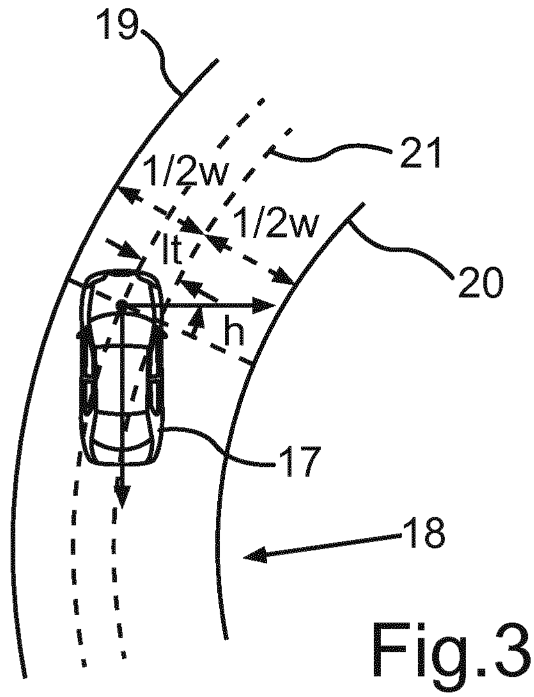



Fig.3

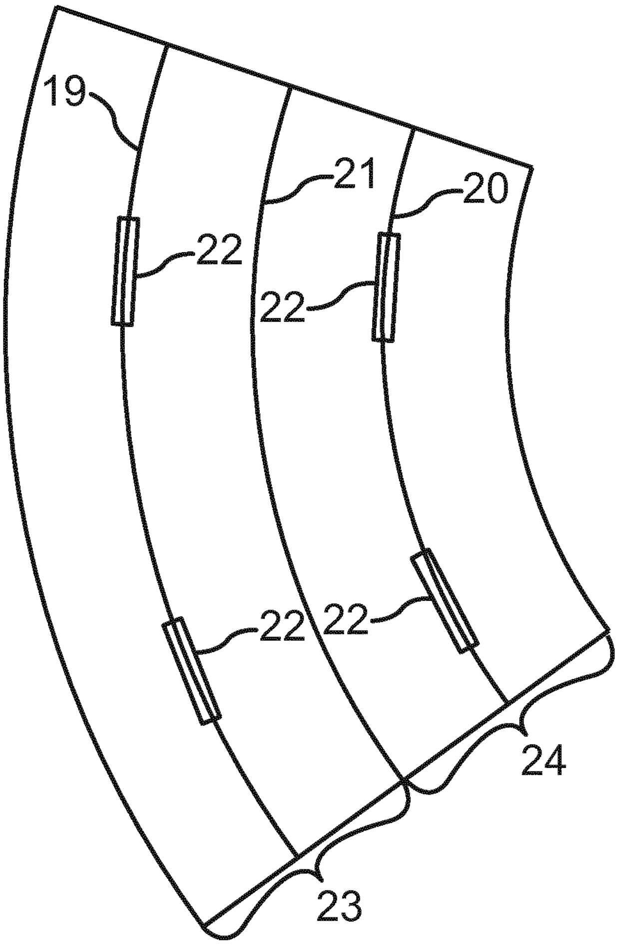
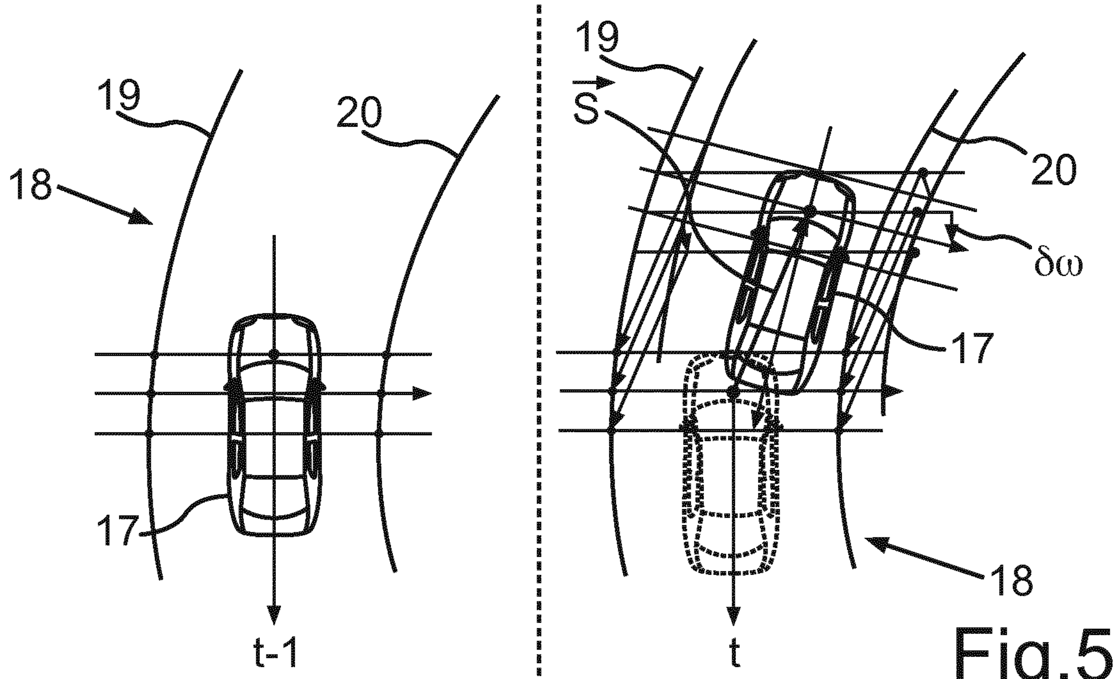
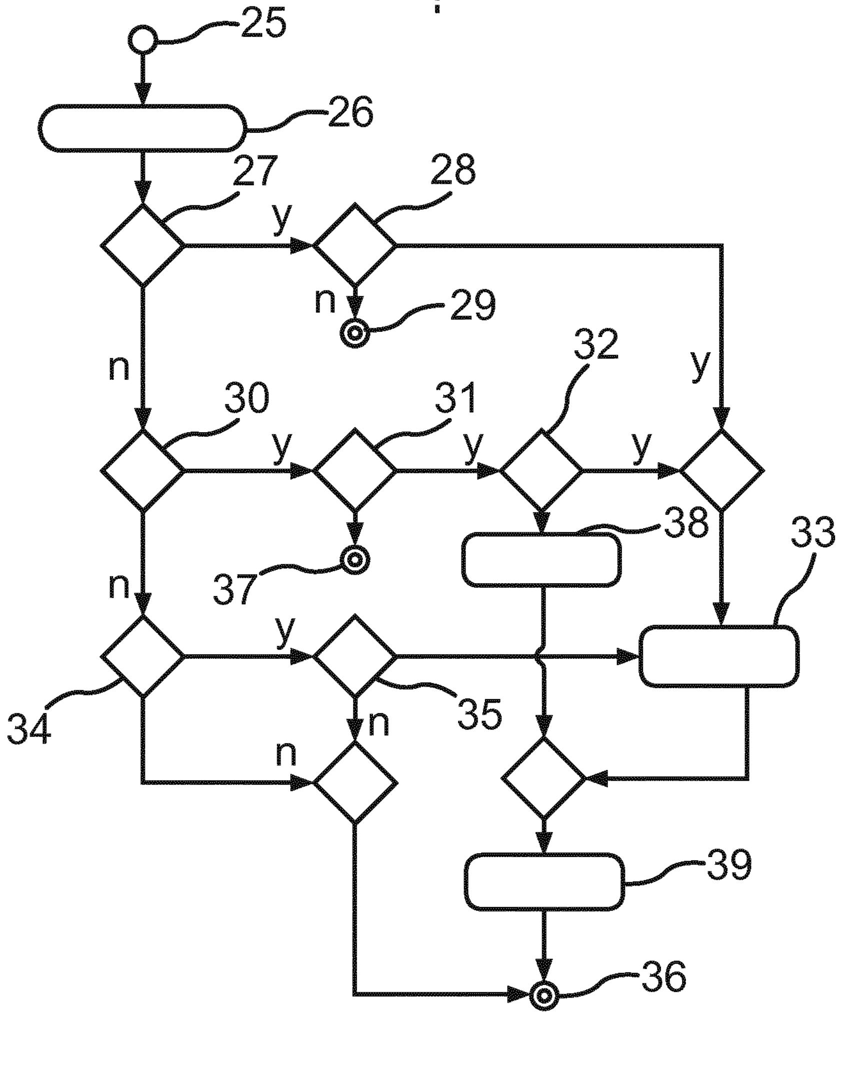



Fig.4



19
20
18
17
t-1
19
S
20
δω
17
18
t


Fig.5



25
26
27
28
y
n
29
n
y
30
31
32
y
y
y
38
33
37
n
y
34
35
n
n
39
36


Fig.6

# INTERNATIONAL SEARCH REPORT

International application No
PCT/EP2019/062641

**A. CLASSIFICATION OF SUBJECT MATTER**
INV. G06T7/246 G06T7/277
ADD. G06K9/00

According to International Patent Classification (IPC) or to both national classification and IPC

**B. FIELDS SEARCHED**

Minimum documentation searched (classification system followed by classification symbols)
G06T G06K

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

EPO-Internal, WPI Data

**C. DOCUMENTS CONSIDERED TO BE RELEVANT**

| Category* | Citation of document, with indication, where appropriate, of the relevant passages | Relevant to claim No. |
|---|---|---|
| A | Daniele Fontanelli ET AL: "A fast and low-cost vision-based line tracking measurement system for robotic vehicles", ACTA IMEKO, Vol. 4, No. 2, June 2015 (2015-06), pages 90-99, XP055609191, Retrieved from the Internet: URL:https://pdfs.semanticscholar.org/7aaf/1ed1a644308ef1fb25e48b42682e9f06b7b1.pdf [retrieved on 2019-07-26] abstract, section 3,pg. 96, Figs. 2 and 5 ----- | 1-13 |

☐ Further documents are listed in the continuation of Box C.

☐ See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance
"E" earlier application or patent but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"&" document member of the same patent family

| Date of the actual completion of the international search | Date of mailing of the international search report |
|---|---|
| 29 July 2019 | 09/08/2019 |

| Name and mailing address of the ISA/ | Authorized officer |
|---|---|
| European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016 | Borotschnig, Hermann |

2