Stereovision-Based Side Lane and Guardrail Detection

Radu Danescu, Stefan Sobol, Sergiu Nedevschi, Thorsten Graf

***Abstract*—This paper presents a fast and robust side lane and guardrail detection technique, based on the 3D information obtained from stereovision and on the already detected geometry of the current lane. The 3D points that belong to the road surface are selected using the road surface parameters inferred from the current lane detection, and they are used for detection of the side lane. The 3D points above the road are used in the detection of guardrails and fences, with the same technique. In both cases, the point sets are smoothed in an attempt to compensate for the variable density, and then analyzed through lateral distance histograms. The peaks of the histograms locate the position of the side lane delimiters and fences, while the shape of the histogram is used for the validation of the side lane or fence's existence. The results are then tracked for increased stability.**

I. Introduction

In knowing the position of our vehicle on the road, the estimation of the current lane's geometry and position is vital. The second most important piece of information about our position is the geometry of the neighboring lanes and/or the guardrails and fences neighboring the road. This knowledge becomes important because:

1. The driving assistance system will have a more detailed information about the environment, which otherwise will be composed only of the current lane and the detected obstacles. In a situation when a collision is imminent on the current lane (an obstacle is in front and the breaking system reacts too slowly) the system must know if there is possible lane change, and where. Knowing about the neighboring lanes and guardrails helps making the right decision.
2. In lane tracking, the situation of lane change will be better handled if the characteristics of the lane we switch to are known before we make the transition. In the absence of any information about neighboring lanes, the new lane is predicted as having the same width, leading to a possible error.
3. Knowledge about neighboring lanes will help the obstacle detection and tracking routines. Deciding to track objects that are only on our lane is not enough, because we may miss a lot of useful information. Knowledge about more lanes can help us identify all objects on the road, therefore all vehicles, and tracking them reliably.
4. The guardrails are, basically, obstacles, which must be detected and tracked. The obstacle detection/tracking routines, however, expect finite size obstacles, which can be modeled as cuboids, and they will not output robust results when confronted to the continuous nature of the guard rails. Thus, a special detection technique must be developed.

This paper presents a unified approach for the detection of side lanes and of continuous elevated structures neighboring the road, such as guardrails and fences. The key aspect of this approach is the use of 3D information provided by stereo processing.

The premises of this work are the stereo-based image processing environment presented in [4] and [5], which delivers real-time high accuracy edge-based 3D information using medium to high resolution grayscale images, and the stereovision-based lane detection algorithm, presented in [3], which reliably and accurately detects the current lane (the lane we are currently on). The stereovision setup and the reconstruction algorithm provide usable 3D points in a range of 10 – 100 m (by usable we mean that they have less than 10% distance error). The rate of false correspondence (leading to false points) is very low, due to the fact that only edge points are used for reconstruction, and the totally horizontal edges are eliminated.

II. Multilane Model

The current lane is modeled as a 3D surface, defined by the vertical and horizontal clothoid curves. Current lane detection means the continuous estimation of the vector $\mathbf{X_C}=(W,\ c_{h,0}\ ,\ c_{h,1}\ ,\ c_{v,0}\ ,X_{cw}\ ,\ \alpha,\ \gamma,\ \psi)^T$, composed of the following parameters (model similar to the one presented in [1], [2]):

W – the width of the lane

$c_{h,0}$ – horizontal curvature of the lane

$c_{h,1}$ – variation of the horizontal curvature of the lane

Manuscript received March 20, 2006. This work was supported by Volkswagen A.G., Germany

R. Danescu, S. Sobol, S. Nedevschi are with Technical University of Cluj-Napoca, Computer Science Department, 400020 Cluj-Napoca, ROMANIA (phone: +40-264-401457; fax: +40-264-594491; e-mail: Radu.Danescu@cs.utcluj.ro, Sergiu.Nedevschi@cs.utcluj.ro).

T. Graf is with Volkswagen AG, Electronic Research, 38436 Wolfsburg, GERMANY (+49-5361-9-29420, fax: +49 5361 9-72837; email: thorsten.graf@volkswagen.de)

$c_{v,0}$ – vertical curvature of the lane

X_{cw} – the lateral displacement of the car reference system from the lane reference system (lane center)

α, γ, ψ are the pitch, roll and yaw angles of the car (the rotation angles between the car reference system and the world reference system).

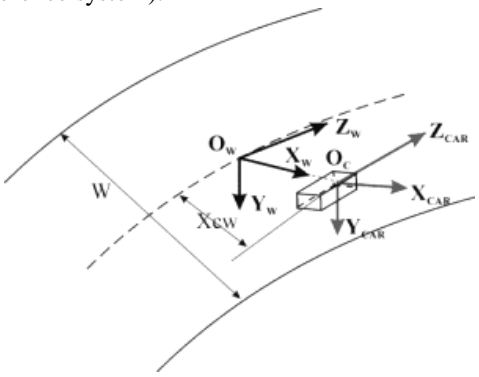

Fig. 1. The lane model.

These parameters describe the lane position and geometry through the following equations:

$$X_c = -X_{cw} - \psi Z + c_{h,0}\frac{Z^2}{2} + c_{h,1}\frac{Z^3}{6} \quad (1)$$

$$X_L = X_c - \frac{W}{2} \quad (2)$$

$$X_R = X_c + \frac{W}{2} \quad (3)$$

$$Y = Z\alpha + c_{v,0}\frac{Z^2}{2} + \gamma X \quad (4)$$

Equation (1) describes the horizontal profile - the variation of the lateral position (X) of the center of the lane with the distance Z. Equations (2) and (3) are expressing the lateral positions of the lane borders. Equation (4) describes the vertical position for any point on the road. The first two terms compose what we'll call the vertical profile, while the last term is due to the roll angle.

The current lane, described by the vector $\mathbf{X_C}$ and the equations 1-4 is already detected before we attempt the detection of side lanes, which we consider to be described by their width alone (orientation and curvature is shared by all lanes). The problem we propose to solve is the problem of the two neighboring lanes. We make this decision on the base of the following reasons:

a) This situation covers a very large number of road situations (its generality is high).

b) The visual information decreases with the lateral distance from the camera. The current lane is viewed best; the neighboring lanes have a smaller yet decent amount of information, and another type of lanes will be almost invisible

c) The Field of View of the camera imposes a minimum visibility distance (the distance from which something enters in the picture) on the lane borders, which increases with the lateral distance. Our stereo setup uses 16mm focal length lenses and 2/3" CCD cameras, which translates in a 30° lateral field of view. This field of view gives, for the average 3.5 m wide lane, the following minimum visibility distances: 6.5 m for the current lane delimiters, 19.5 m for the neighboring lane delimiters, and 32.6 m for the lanes further beyond.

Add to that the perspective effect, which makes the 3D features in the image to be smaller as their distance increases, and the result is that the third lane is almost impossible to see.

As a consequence, we define the road by the central curve (vertical and horizontal profile), a roll angle γ and the number and width of its lanes. Lanes are adjacent, and therefore they cannot have a different curvature, a different vertical profile, a.s.o. Therefore, our target scenario (current lane plus neighboring lanes) will be defined by the following vector: $\mathbf{X} = (\mathbf{X_{current}}\ W_L\ W_R)^T$ – the current lane geometry plus the widths of the neighboring lanes.

The multiple lane detection algorithm has two steps. The first is the detection of the current lane, using the method described in [3]. If the current lane is not available, there will be no search for the side lanes. Step two, the actual detection of the side lanes must deliver the following information:

- A side lane exists or not. This is the most important aspect: establishing the existence of a side lane
- The width of the side lane

A tracking algorithm will be applied after the side lane's detection, in order to filter the detection results and increase stability of the output.

III. ALGORITHM DESCRIPTION

A. *Preparing the input data*

The algorithm takes as input the set of 3D points and the parameters of the horizontal profile of the current lane (width, lateral offset, yaw angle, curvature and curvature variation). Out of the set of 3D points, only those that belong to the road surface are selected – those that comply with equation (4) whose parameters are already detected.

We assume that the side lanes have the same orientation as the current lane. This means that the yaw angle, the curvature and the curvature variation are the same. We shall transform the 3D space so that we shall have a space whose axis of symmetry is the central curve of the current lane, the one described by the equation (1).

Figure 2 shows a bird-eye view of an idealized set of 3D road points, with the apparent curvature. Figure 3 shows the

effect of transforming the 3D space, by subtracting from the X coordinate of each point the value X_C given by the equation (1) applied to the Z coordinate of the point. Basically, we straighten the curved point set, as we already know the curvature and orientation. If the assumptions were correct (that is, each side lane has the same orientation as the current lane, and the current lane is correctly detected) then the points belonging to lane delimiters must form, after space transformation, straight lines parallel to the Z axis of the coordinate system.

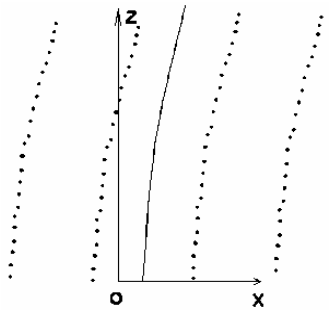

Fig. 2. Road points in the initial 3D space

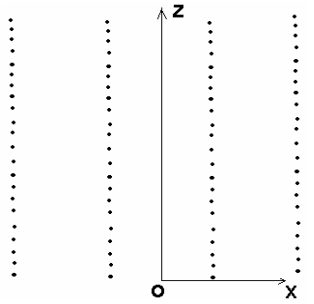

Fig. 3. Road points after space warping

After the warping of the 3D space to compensate for the curvature of the lanes, processing becomes much simpler. However, we still have the problem of the 3D point distribution, which is not uniform. We would like to have a set of points which is

- Distributed as uniformly as possible in the XOZ plane
- Completely contained in the XOZ plane

The last condition is somehow hard to understand. If we take the Y coordinate of each point out of consideration, each point is contained in the *XOZ* plane. But the problem is that the 3D space being continuous, not discrete, two points having similar X and Z coordinates, but different Y coordinates, will also become two points in the XOZ plane, even if they cannot be distinguished if seen in bird-eye projection. We don't want to have these "hidden" points in the future processing.

The second condition has given the idea for the solution of both problems. We will construct a discrete 3D space, by representing the space as a matrix of cells. The size of the cells is 20 cm on the Z-axis and 10 cm on the X-axis. Then, for each 3D point on the road surface (its Y coordinate being thus in a certain interval), we find the corresponding cell. If the cell is free (no point has been allocated to this cell before), we mark the cell as taken, and if the cell was previously taken we do nothing. This way, it doesn't matter how many points fall in the same cell, the only difference being whether we have a point in the cell or not. After all the points have been allocated, we take the cell matrix and transform the coordinates of the "full" cells into 3D coordinates. In this way we obtain a set of points having discrete 3D coordinates, and a distribution that is a lot more uniform than the original set.

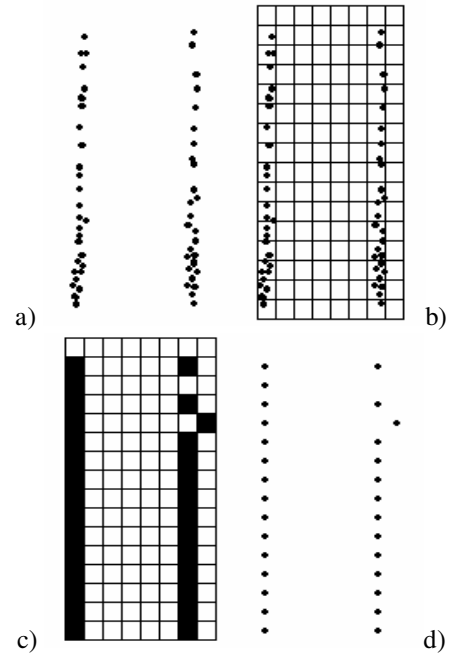

Fig. 4. Increasing the uniformity of the set of points: a) Original set of points, b) The original set of points and the discrete cell grid, c) The grid with the full and empty cells, d) The full cells are transformed in 3D points with discrete coordinates

Figure 5 shows a real set of stereovision-generated 3D road points, and figure 6 shows the filtered points inserted in search areas for the left and right side lane delimiters. The search areas are given by the prediction and by the prediction's uncertainty.

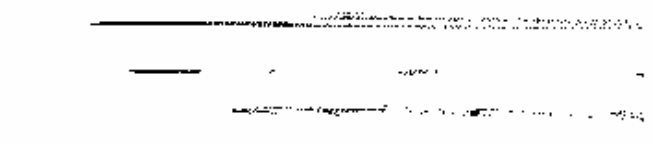

Fig. 5. A set of real 3D points viewed from above

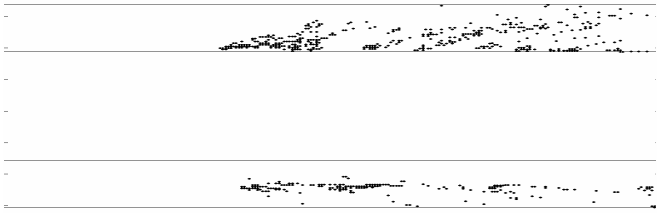

Fig. 6. The filtered set, in search areas for the left and right side lane

This basic technique of obtaining uniform 3D points can be altered in order to accommodate point classification techniques. The classification techniques are image space-based analysis algorithms, aimed at finding relevant lane delimiting features. One such classification algorithm is the search for dark-light-dark patterns, which indicate possible lane markings, which have higher priority over the rest of the points. The results of the classification algorithms are used to assign to each road 3D point a numerical weight in the interval (0.1, 1).

The cell matrix will hold, instead of true/false values (indicating whether a point has been inserted in the cell or not), weights. The initial value of each cell is zero, indicating that the cell is free. When processing the array of original 3D points, the corresponding cell for the point is computed. If the value of the cell is zero, the cell will take the value of the class weight of the 3D point. If the cell has a non-zero value, meaning that a point has already been inserted, the weight of the point is compared to the weight of the cell, and the weight of the cell will become the maximum of the two values. Thus, if more than one point corresponds to a grid cell, the weight of the cell is the maximum of the point weights.

After all the points have been processed, the set of discrete 3D points is created. The coordinates will be extracted as already described in the basic method, but an additional vector will be created, containing the weights of the points, these weights being the values of the cells.

Therefore, the input data for the side lane detection algorithm is composed of three vectors: X[], Z[] and W[], containing the warped discrete coordinates of the 3D points and their corresponding weights.

B. Side lane localization and validation

The filtered points that fall in search areas for the left and the right side lane (search areas are generated by prediction, using the past results and the associated uncertainty) are used for building two histograms. The first histogram represents the number of points for each discrete X-coordinate of the discrete 3D space, and the second one is the weighted histogram, where each entry represents the sum of weights of the points corresponding to an X coordinate of the discrete space.

The following image shows the two half-histograms for the right side lane search area. The weighted histogram has lower amplitude, because the weights are less or equal to 1.

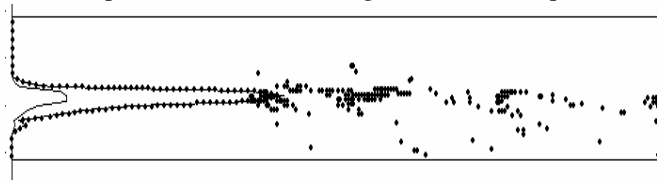

Fig. 7. The point count histogram (dotted line) and the weight sum histogram (continuous line) for the right side lane search area

In what follows, we'll keep our discussion to a single search area (a single side lane). The point count histogram is analyzed, and the maximum value, *HMax*, along with its position *XMax* is computed.

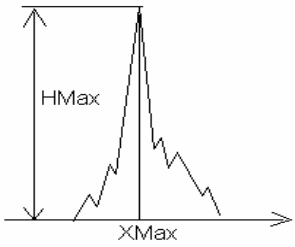

Fig. 8. The maximum of the histogram and its position

After finding the maximum, the histogram is smoothed by convolving it with a Gaussian kernel, on each side of the maximum. We don't perform a full convolution on the histogram because we don't want to move the position or to change the value of the maximum (as it may be a very sharp peak, if the points are very well aligned). After the smoothing, we perform a descent from the maximum position towards the left and the right, as long as the descending trend is kept. Therefore, we will find the maximum height on the left side and on the right side of the maximum, until we reach a local minimum. Also, besides the heights, the lengths of the descending paths are computed. Figure 9 illustrates the process:

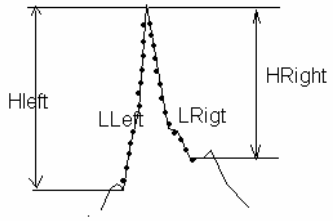

Fig. 9. Computing the height of the maximum continuous descent and the length of the way down

We select for future analysis the maximum descent height (out of the two possible paths), along with its corresponding length. Let's denote them by *MaxHeight* and *MaxPath*. Therefore, the parameters of the point count histogram are *HMax*, *XMax*, *MaxHeight*, *MaxPath*.

The weight sum histogram is also searched for the maximum and the maximum position. Let's denote them *WMax* and *WXMax*.

These parameters are the basis for the following reasoning:

If $HMax < 15$, the side lane is invalidated (no side lane)

If $HMax > 50$, the side lane is considered valid

If *HMax* is in the interval 15..50, the condition to become valid is:

$$MaxHeight > 10 \text{ and } \frac{MaxHeight}{MaxPath} > 0.7$$

This means that the maximum descent must be higher than a threshold and the descent must be steep.

If these conditions have been met, and the lane is valid so far, we analyze the position of the point count histogram maximum (*XMax*) with respect to the position of the weight sum histogram maximum (*WXMax*). If the distance between these positions is higher than 3, the lane is invalidated. Otherwise, the lane is considered valid, and the position final position of the side lane delimiter will be given by the position of the weight histogram maximum.

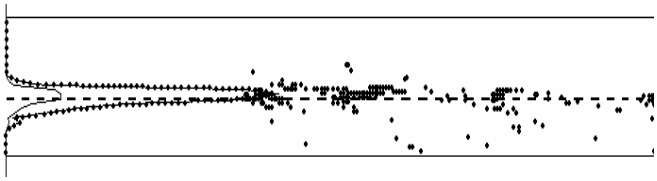

Fig. 10. The position of the side lane delimiter (heavy dashed line), is the position of the weight histogram maximum

C. *Using the obstacles in side lane validation*

A situation in which a side lane is invalidated is when an obstacle is present on it, in the interval (15-30m). This is the interval of maximum visibility of the side lane, and therefore it is highly improbable that correct side lane detection can be performed. The obstacles are detected by grouping the 3D points above the road surface into cuboids, as presented in [4].

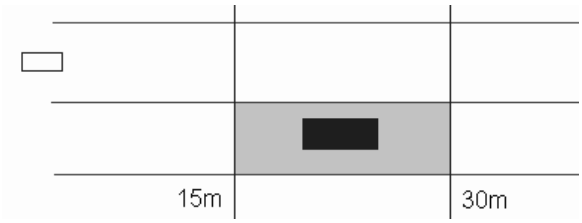

Fig. 11. Invalidating a side lane due to the presence of an obstacle in a key point

IV. DETECTION OF GUARDRAILS AND FENCES

Guardrails and fences are continuous obstacle-type objects that go alongside of the road. Due to their continuous nature, it is not feasible to attempt their detection/tracking in the same way we detect/track a vehicle, by grouping the 3D points into discrete cuboids. This approach will generate a set of objects formed from pieces of guardrail, objects that will differ greatly from one frame to another, and which are impossible to track.

Fig. 12. Side lane and a guardrail

The guard rail and the side lane delimiter have a lot of properties in common. They are continuous structures, and they go alongside of the road – they share a common orientation with the already detected current lane. Therefore, they both are described by their lateral position only.

The algorithm described in this paper detects the guard rails in the same way it detects side lanes. The only difference is the input data: instead of feeding it points that are contained in the road surface, we feed it points that are above the road surface. The histogram analysis will detect the long, continuous structures of elevated points as spikes, in the same way the side lane delimiters are detected.

V. RESULTS

The side lane and guardrail detection system has been integrated in the Stereo CAmera-Based Object Recognition

(SCABOR) application, application that is able to perform online (onboard) and offline (on recorded sequences) processing. The edge-based stereovision engine delivers about 5000 reliable 3D points per frame (frame size is 640x480), which allows the high level algorithms, such as the side lane detection, to operate very fast (the side lane detection routine takes less than 5 ms). The main target scenario is the highway (figure 13, 14), a structured environment with high-speed traveling velocities. The algorithm's performance is best in the designated target environment, correctly recognizing the side lane width and the guardrail's position with about 10 cm errors. However, the system produces dependable results also on rural roads, marked or even non-marked (figure 15), or on pre-urban roads (figure 16).

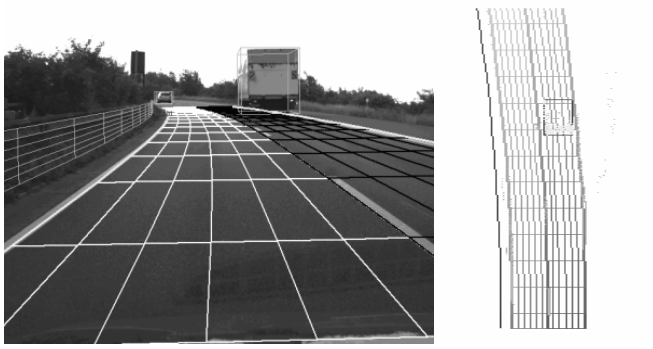

Fig. 13. Guard rail and side lane detected on highway

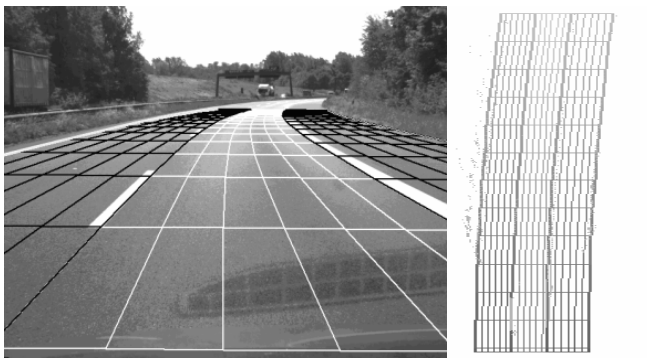

Fig. 14. Both neighboring lanes detected on highway

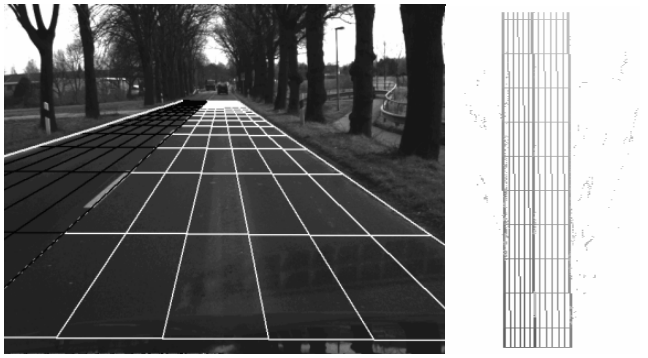

Fig. 15. Left side lane detected on a non-marked rural road

Fig. 16. Two guard rails detected on a pre-urban road

The quality of the side lane and guardrail results depends on the quality of the stereo reconstruction, which in turn depends on the quality of the camera calibration. The side structures, lane delimiters or guardrails, are closer to the periphery of the image, therefore subjected to a greater extent to radial distortion. They also tend to be more horizontal in the image, and thus more likely to be falsely matched. A careful calibration and a good image quality and resolution are the prerequisites for the algorithm's success.

VI. Conclusion

We have presented a robust and straightforward method for detecting the side lane delimiters and the position of the guardrails and fences using stereovision. The algorithm is based on the hypothesis that the side lane and the guardrails share the vertical and the horizontal profile (shape) with the current lane, which is already detected, and this allows the "straightening" of the 3D point set, making it suited for a histogram-based analysis. The simplicity, accuracy and robustness of the approach derive from the advantages of using a stereovision-based system for lane and road detection.

References

[1] E.D. Dickmanns, B.D. Mysliwetz, "Recursive 3-D Road and Relative Ego-State Recognition", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 14, Issue 2, February 1992

[2] J. Goldbeck, B. Huertgen, "Lane Detection and Tracking by Video Sensors", *In Proc.of IEEE International Conference on Intelligent Transportation Systems*, October 5-8, 1999, Tokyo Japan, pp. 74–79

[3] S. Nedevschi, R..Schmidt, T. Graf, R. Danescu, D. Frentiu, T. Marita, F. Oniga, C. Pocol, "3D Lane Detection System Based on Stereovision", *in Proc. of IEEE Intelligent Transportation Systems Conference (ITSC)*, Washington, USA, October 4-6, 2004, pp. 292-297

[4] S. Nedevschi, R. Schmidt, T. Graf, R. Danescu, D. Frentiu, T. Marita, F. Oniga, C. Pocol, "High Accuracy Stereo Vision System for Far Distance Obstacle Detection", *in Proc. of IEEE Intelligent Vehicles Symposium*, Parma, Italy, June 14-17, 2004, pp.161-166

[5] S. Nedevschi, R. Danescu, T. Marita, F. Oniga, C. Pocol, S. Sobol, T. Graf, R. Schmidt, "Driving Environment Perception Using Stereovision", *in Proc of IEEE Intelligent Vehicles Symposium, (IV2005)*, June 2005, Las Vegas, USA, pp.331-336