Ground truth evaluation of the Stixel representation using laser scanners

Conference Paper *in* Conference Record - IEEE Conference on Intelligent Transportation Systems · October 2010

DOI: 10.1109/ITSC.2010.5625017 · Source: IEEE Xplore

4 authors:

David Pfeiffer
Daimler
29 PUBLICATIONS **2,186** CITATIONS
SEE PROFILE

Sandino Morales
Terrabotics
31 PUBLICATIONS **350** CITATIONS
SEE PROFILE

Alexander Barth
Daimler
20 PUBLICATIONS **486** CITATIONS
SEE PROFILE

Uwe Franke
Daimler
177 PUBLICATIONS **22,326** CITATIONS
SEE PROFILE

Ground Truth Evaluation of the Stixel Representation Using Laser Scanners

David Pfeiffer[1], Sandino Morales[2], Alexander Barth[3] and Uwe Franke[1]
[1] Environment Perception, Daimler Research, Sindelfingen, Germany
[2] Department of Computer Science, University of Auckland, New Zealand
[3] Department of Photogrammetry, University of Bonn, Germany
{david.pfeiffer, uwe.franke}@daimler.com, pmor085@aucklanduni.ac.nz, alexander.barth@uni-bonn.de

***Abstract*— Modern real-time dense stereo vision provides precise depth information for nearly every pixel of an image, indicating stereo cameras as a key sensor for future vehicle safety systems. Efficient analysis of this large amount of data by different tasks running in parallel asks for a medium level representation that decouples application specific analysis from low-level vision. Recently, the so called "Stixel World" has been proposed. It models the objects in the scene, implicitly separates them from the ground plane, encodes the freespace to maneuver and thus represents the scene in a highly compact manner that supports different recognition tasks efficiently. The potential of this new representation depends on the accuracy that can be achieved. Therefore, this paper analyzes the precision of this representation using a high performance laser scanner as reference sensor. The statistical analysis confirms the high accuracy as expected from visual inspection.**

I. Introduction

Cameras are turning out to be a key sensor for safety systems of modern cars. Particularly stereo vision will play an essential role in traffic scene understanding. The three-dimensional perception of the environment will be the basis for sophisticated driver safety and comfort systems. This also includes life-saving collision avoidance systems such as emergency braking and avoidance by steering; with zero tolerance to malfunctioning. Besides vehicles and static obstacles, pedestrians can be recognized and their behavior can be anticipated.

The power of modern stereo vision becomes obvious when looking at Figure 1 and the corresponding 3D visualization of the same situation in Figure 2. Stereo algorithms such as *Semi-Global Matching* (SGM) [1] allow us to estimate the disparity of nearly every pixel with sub-pixel accuracy. Recently, Gehrig et al. presented a real-time implementation of this scheme using an FPGA [2]. An extension of that work delivers 400000 3D points per frame, 25 frames a second.

The vision systems currently on the market try to bridge the gap between features and objects as quick as possible using application specific heuristics and techniques. For example, stereo points are accumulated in occupancy grids, object boxes are tracked over time and lane markings are fed into Kalman filters. The concept of a shared medium level representation for different vision tasks has not yet found its way into this young but important area of research.

This will change in the future, when various applications use the images and the derived stereo data simultaneously. In order to facilitate a powerful architecture for those systems and to avoid the processing of all 3D points repeatedly an appropriate medium level representation is desirable to decouple low-level vision and application specific tasks. Such a representation should be:

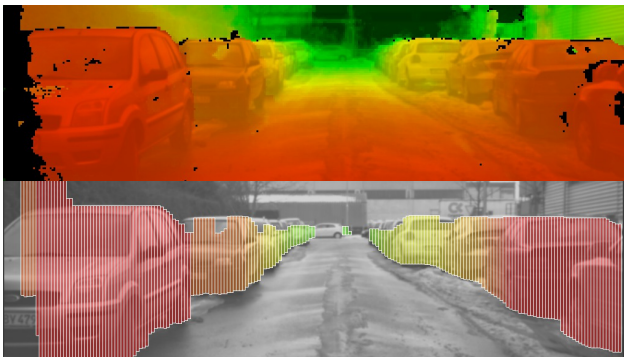

Fig. 1: Visualization of the disparity image computed using SGM and the extracted Stixel representation for an exemplary urban traffic scenario.

- *compact*: offering a significant reduction of the data volume.
- *complete*: information of interest is preserved.
- *stable*: small changes of the underlying data must not cause rapid changes within the representation.
- *robust*: outliers of the input data must have minimal or no impact on the resulting representation.

Recently, we proposed the *Stixel World* for the representation of traffic scenes and typical situations occurring in the world of autonomous mobile systems [3]. The idea is to represent the current 3D-situation by a set of rectangular sticks named *Stixels* as shown in Figure 1. Each Stixel is defined by its 3D position relative to the camera and stands vertically on the ground, having a certain height. Each Stixel limits the freespace and participates in approximating the object boundaries. All Stixels share the same width measured in pixels within the image.

Thus, several important tasks in object detection and recognition are solved implicitly or are strongly supported by this approach: The discrimination of object and ground, the segmentation of objects at different distances and the grouping of Stixels to objects, which turns out to be straightforward based on their spatial vicinity.

In order to use the Stixel representation in the context of

Fig. 2: Visualization of the triangulated SGM data and the 3D Stixel representation.

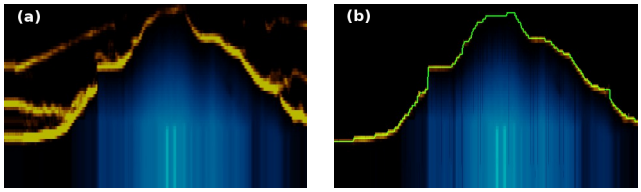

Fig. 3: Both figures illustrate a polar occupancy grid. Orange represents the likelihood for areas to be occupied, blue models the likelihood for street occurrence. The right grid is computed from the left one by applying the background subtraction. The green line corresponds to the freespace in polar coordinates obtained by dynamic programming.

safety critical driver assistance systems, the precision and reliability of the Stixel World are a major concern.

This contribution is based on prior work from Badino et al. [3]. We present an evaluation method using a calibrated high-precision Velodyne Laser Scanner [4]. This LIDAR sensor has 64 vertical beams and a horizontal resolution of up to 0.1 degree. It has an accuracy within centimeters up to distances of 70 m.

Section II sketches the necessary steps required to build the Stixel World from raw stereo data. Section III addresses our method to calibrate the used sensors and to generate ground truth data. We also explain the used evaluation scheme and discuss significant characteristics of the different sensors. Our experimental results are presented in Section IV. Section V concludes this contribution.

II. Building the Stixel World

In the Stixel World each Stixel represents the first object to encounter along each column of the image and thus encodes the distance, the location of the base point and the height of that object. The Stixel representation for a given situation is obtained in four steps: (1) generate a disparity image for the given stereo image pair, (2) determine the base points by computing the freespace using an occupancy grid, (3) perform a height segmentation to obtain the height of the objects and (4) extract the Stixel depth by using a histogram-based disparity registration scheme.

A. Dense Stereo

Stereo vision has been an active area of research for decades. For real-time stereo algorithms correlation-based approaches are popular (e.g. [5]). Among the top-performing algorithms in the Middlebury database [6], we found semi-global matching (SGM) [1] to be the most efficient. Even though the Middlebury database is a good platform for the comparison and rating of stereo algorithms under controlled conditions, detailed surveys on the accuracy and reliability of stereo algorithms in real-world and automotive scenarios are still an open topic [7].

Based on the SGM algorithm, Gehrig et al. have introduced the first real-time dense stereo implementation with a power consumption of less than 3 W [2]. This implementation runs on a Xilinx FPGA platform. For our purpose we use a variant of that implementation that is able to compute 1024×440 px disparity images at 25 Hz. An exemplary disparity image for a common urban traffic scene is depicted in Figure 1.

B. Freespace Computation

The freespace is computed from an occupancy grid in three steps: Obtaining a polar occupancy grid, background subtraction and dynamic programming.

Occupancy grids are used to stochastically model the likelihood of the environment to be occupied. Such grids are obtained by registering stereo disparities in their associated cells while considering the depth uncertainties known from the used stereo algorithm. The more stereo disparities are mapped to a specific cell, the higher is its likelihood to be occupied. In [8] we discussed several representations for occupancy grids in detail and found the polar column disparity grid (u, d) to be the most suitable to compute the freespace. This is due to the fact that an efficient search for freespace must be done in the direction of rays leaving the camera. In polar coordinates every grid column is, by definition, already in the direction of a ray. An exemplary polar column disparity grid is depicted in Figure 3a. For a better understanding of the spatial context the grid from Figure 3a has been remapped to a Cartesian representation shown in Figure 4.

Having such a polar representation, the task is to find the first visible relevant obstacle in the positive direction of depth. Looking at Figure 3a this means that the search must start from the bottom of the polar occupancy grid in vertical direction until an occupied cell is found. The space found in front of that cell is considered as freespace.

Every possible freespace solution is associated with a cost energy. Dynamic programming (DP) is used to find the optimal path cutting the polar grid from left to right. The output of the DP is a set of vector coordinates (u, d_b), where u is a column of the occupancy grid and d_b the disparity corresponding to the distance up to which freespace is available. Note that each freespace point (u, d_b) of the occupancy grid shown in Figure 3b indicates not only the limit of the freespace. It also describes the location of the base-point of the first obstacle located at that position as

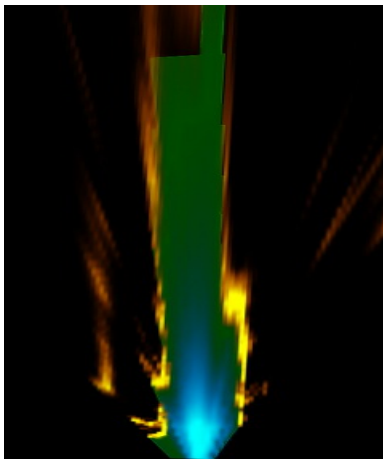

Fig. 4: This Cartesian occupancy grid is obtained by a transformation of the polar grid given in Figure 3a. The freespace polygon is overlaid using a green coloring.

illustrated in Figure 5 where the freespace is projected into the left image.

For every pair (u, d_b) a coordinate (x_u, z_u) is triangulated, which defines the corresponding 2D world point. The sorted collection of the points (x_u, z_u) plus the origin $(0, 0)$ form a polygon which defines the freespace area from the camera's point of view (see Figure 4) in Cartesian coordinates.

The next section briefly describes how to apply a second pass of dynamic programming in order to obtain the upper boundaries of the objects.

C. Height Segmentation

The height of the objects limiting the freespace is obtained by finding the optimum segmentation between foreground and background disparities. This is achieved by first computing a cost image from the disparity image and by then applying dynamic programming to find the upper boundary.

Given the set of freespace points (u, d_b) and their corresponding triangulated Cartesian coordinates (x_u, z_u), obtained in the previous step, the task is to find the optimum row position v_t where the upper boundary of the object at (x_u, z_u) is located.

From the disparity image a membership image is computed. Therefore every disparity $d_{u,v}$ votes for its membership to the foreground object given by the freespace disparity in column u and thus encodes, if it belongs to the object or not. Since all objects must have a positive height

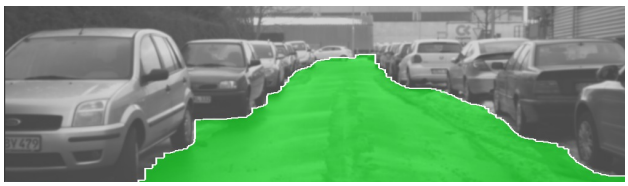

Fig. 5: Visualization of the freespace result after applying dynamic programming to the polar occupancy grid from Figure 3b.

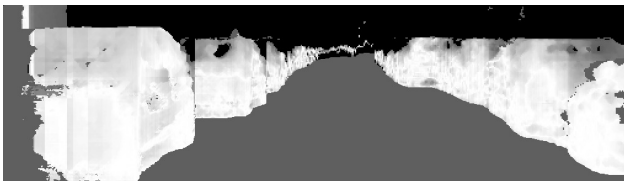

Fig. 6: Visualization of the membership votes with white meaning positive (belonging to the object), gray neutral and black negative.

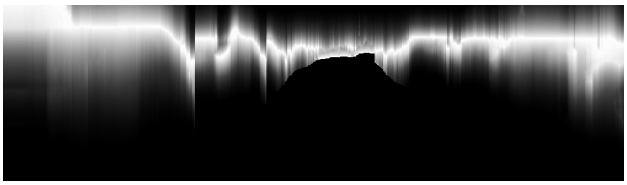

Fig. 7: Visualization of the cost image (data term) used for the DP in the height segmentation. Bright values represent a high likelihood to perform the cut.

only rows above the image coordinate (u, v_b) have to be considered, where v_b is the row position of the base point (x_u, z_u). Figure 6 shows the resulting membership votes for our exemplary scene. The cost image illustrated in Figure 7 shows the resulting costs when the single membership votes illustrated in Figure 6 are accumulated into cost hypotheses.

For the computation of the object heights DP is used to find the optimum path cutting the cost image from left to right. The height segmentation is explained in more detail in [3]. The segmentation result for our exemplary scene with the freespace shown in Figure 5 is depicted in Figure 8.

D. Stixel Extraction

Once the freespace and the height for every column has been computed, the extraction of the Stixel is straightforward. The properties base and top point v_b and v_t as well as the width of the Stixel span a rectangle where the Stixel is located within the image.

Since the occupancy grids are discretized to equidistant steps in disparities, the freespace vector inherits this finite resolution and is thus limited in precision. To overcome this limitation the disparities found within each Stixel are registered in a histogram while regarding the depth uncertainty known from SGM. A parabolic fit around the maximum delivers the new and more precise sub-pixel depth

Fig. 8: Result of the height segmentation after applying DP. The red line indicates the segmentation of foreground and background.

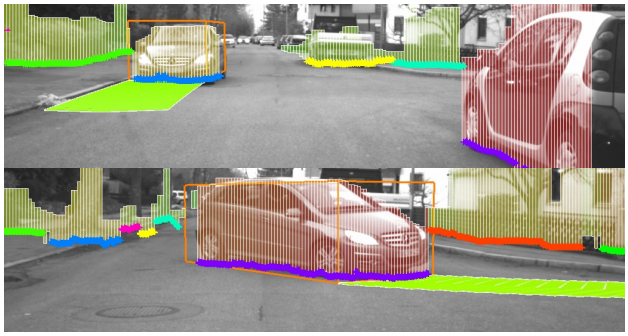

Fig. 9: Application of Stixels in a vehicle tracking system. The color bands at the bottom mark groups of Stixels, the boxes around the cars show their estimate orientation and the green "carpet" illustrates the predicted trajectory of the vehicle.

Fig. 10: Photograph of our test vehicle showing the installed stereo camera system behind the windshield and the Velodyne *LIDAR HDL-64E* mounted to the roof rack.

information. In addition this approach offers outlier rejection and noise suppression of the raw SGM input.

E. Application of the Stixel Representation

With respect to the previously mentioned criteria regarding a medium level representation the Stixel representation proves to decouple low-level data from high-level algorithms in a convincing fashion. It corresponds to a figure-ground segmentation that offers a precise contour approximation. By varying the width of the Stixels, the user can individually choose between compactness and the detail this medium level representation provides. Given a 1024×440 px image and a fixed width of 5 px for the Stixels, the whole scene is described in 205 Stixels, while every Stixel is defined by 2 parameters only (distance and height). The lateral position is given by the ordering, thus the set of Stixels additionally encodes the freespace available to maneuver. In total we achieve a reduction of the data volume of 99.9%: 400000 disparity measurements reduced to 410 values.

The extraction scheme proves to be robust against outliers in the input data. With minor changes of the input data between two consecutive frames this representation is still alike and does not require any complex reorganization (unlike tree-based structures or graphs).

Furthermore the Stixel representation is qualified to be grouped to objects due to their spatial vicinity as shown in Figure 2. Thus it is very suitable for tasks like control of attention, object detection and object tracking as done in the work of Barth et al. [9]. Stixels are grouped to clusters by using straightforward heuristics. The spatial orientation and silhouette of those groups are used as a prior and a constraint in a point-based vehicle tracking approach. Exemplary results are depicted in Figure 9.

The next section describes the process to generate real-world ground truth data and the used evaluation methods.

III. Ground Truth Generation

When looking at disparity images one can hardly make a statement regarding the quality of the 3D reconstruction. The conclusion that the results seem plausible is by no means sufficient. Consequently it is in our interest to rate the precision of the obtained results by validation to reference sensors, especially when dealing with traffic safety tasks or driver assistance systems. It is our goal to automate this process without the need of human interference, so the verification is performed on large data sets covering various real-world scenarios.

For this purpose 3D laser scanners are suitable as reference sensors. We rely on the *Velodyne LIDAR HDL-64E* , that generates $360°$ point scans of the environment and is well known from its application in the DARPA Urban Challenge contest [10]. While rotating it simultaneously records 3D point measurements in 64 vertically stacked scan-lines each with a rate of ≈ 20000 shots per second. Having a rotation rate of at least 5 Hz (15 Hz max.), this sensor achieves an angular resolution of up to 11 pts per degree while our stereo camera system yields a resolution of 22 px per degree and covers a total of $45°$ field of view. The particular characteristic of this sensor that qualifies it as our reference is the constant measurement accuracy that is within centimeters independent of the measured distance.

In order to compare the obtained 3D data against the results from the stereo and object detection algorithms, we have to calibrate the cameras relative to the LIDAR.

Knowing the intrinsic camera parameters, the relative orientation to the LIDAR and the base line of our stereo camera system we are able to compute virtual ground truth disparity images from the LIDAR 3D point clouds. Based on these images we will automatically verify the accuracy of the Stixel representation without the necessity to fall back to the pixel domain for a piecewise disparity comparison.

The following sections deal with the test setup and sensor calibration, the fusion and comparison of the data, special sensor characteristics as well as the quality rating of the Stixel representation.

A. Test Setup and Calibration

Our stereo vision system consists of two 1024×440 px cameras having a $45°$ field of view and a base line of 25 cm. They are attached to the front windshield close to the rear

view mirror while the LIDAR is mounted to the roof rack. An unavoidable deviation in position between these two sensors leads to overlaps of objects from different depths when 3D points obtained from the LIDAR are mapped into the image domain. This is a negative effect, since our primary concern is to record 3D points that are also visible within the image and to use them for verification. In order to minimize this effect we chose a mounting position as close to the left camera as possible. The sensor setup is shown in Figure 10.

The relative orientation between the camera system and the LIDAR consists of 6 parameters, a 3D translation and a 3D rotation. In order to determine these parameters we use the method presented by Lepetit et al. who offer a closed-form solution of the Perspective-n-Point problem [11]. Using known 3D points obtained from the LIDAR sensor and manually selected corresponding 2D image coordinates we are able to estimate the location and pose of our calibrated camera relatively to the LIDAR. Alternatively an iterative closest points approach can be used [12].

B. Data Fusion and Preparation

In order to associate data from the LIDAR to the stereo disparities, the stereo image pairs should be grabbed synchronously to the LIDAR data. This proves to be challenging for a couple of reasons. Firstly, the used sensors operate at different refresh rates of $5\,\mathrm{Hz}$ for the LIDAR and $25\,\mathrm{Hz}$ for the cameras. This leads to a worst case offset of $100\,\mathrm{ms}$ between these two sources. Secondly the working principle of the LIDAR is comparable with a vertical rolling shutter with $25\,\mathrm{ms}$ spent to capture content that is also visible within the stereo images. To avoid problems resulting from ego- and foreign-motion, our test-vehicle is standing still when recording the data. Additionally, dynamic scene content is avoided.

The 3D LIDAR measurements are transformed into the left camera coordinate system and projected to image coordinates. Their corresponding disparity values are then registered into a virtual disparity image. Such an exemplary ground truth disparity image is illustrated in Figure 11. For this contribution we do not interpolate or beautify the obtained LIDAR data. Assuming this would be favored, Dolson et al. presented a method to upsample range data with consideration of dynamic environments [13]. Also they are able to deal with the asynchronism and take into account that the LIDAR scan content for a given frame might be incomplete due to scan artifacts.

C. Quality Rating and Sensor Characteristics

In order to draw a conclusion regarding the measurement accuracy of the Stixel representation, the distance extracted for each Stixel is compared to the distance obtained from the LIDAR. The LIDAR distance measurement Z_L is determined by computing the mean of the inner 70% of the sorted LIDAR 3D points the Stixel covers within the image. Due to a significantly higher accuracy of the LIDAR compared to the stereo sensor this approach is regarded adequate with respect to precision.

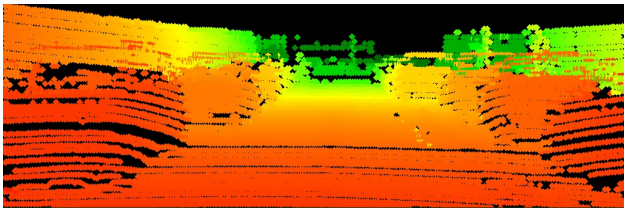

Fig. 11: Exemplary illustration of a real-world ground truth disparity image generated from 3D point clouds of the Velodyne LIDAR. For illustration purpose the points have been thickened. This is the raw data as obtained from the sensor, we do not apply any interpolation.

Fig. 12: The container right to the parked car was used for the evaluation of Stixels on objects that match our object constraints.

It is important to note that SGM and the LIDAR behave differently for specific object classes. A good example for that are reflective or (semi)transparent object parts such as windows, mirrors or puddles. While the LIDAR either looks right through those objects or follows the reflected ray the SGM usually smooths over these objects but still does not yield a result close to the theoretically optimum. This effect is clearly visible in the virtual disparity image illustrated in Figure 11, like in the windows of the parked cars. These data cannot be used for a ground truth evaluation.

Another concern results from the way we rate the precision of the Stixels. Due to our object model constraints (see Section I) we assume objects to have a nearly vertical pose. Accordingly we impose a constant disparity across the Stixel area by penalizing deviations. However many objects do not fully match this condition especially with an increasing height (e.g. engine hoods).

For these reasons we split the evaluation into two parts: At first we run the evaluation using a sequence with vertical objects, a well structured container we slowly approach. It features no reflective or transparent parts and complies with the imposed rectangularity constraint perfectly. A snapshot of the container sequence can be seen in Figure 12.

The second part of the evaluation is done within a real traffic environment at various positions in a narrow street with parking cars on both sides of the road. For this sequence we will limit the evaluation of stereo disparities beneath the Stixels to a height of $80\,\mathrm{cm}$ to avoid the effects mentioned above. Additionally it has a high severity for a couple of further reasons: At first we look at the objects in a very acute angle. Secondly those objects have a shiny and reflective surface (an effect that is amplified especially under that acute angle). Both factors result in a challenging scene for accuracy investigations of the depth estimation. The frame we used for our illustration (e.g. see Figure 8) is part of that sequence. Both sequences consist of more than 300 frames

and contribute a total of about 10000 Stixels each.

IV. Results

Three different types of curves were determined for the evaluation: The mean difference δ_S of the Stixel distance Z_S compared to the obtained LIDAR distance Z_L, the standard deviation σ_S of that error and the inter-Stixel standard deviation σ_L of the single LIDAR distance measurements. We decided not to consider the standard deviation of the SGM disparities, because those are highly correlated due to the SGM smoothness constraints and thus are misleadingly small. Literature regarding stereo evaluation methods often claims a disparity accuracy of up to $\sigma_d = 0.25\,\text{px}$ for reasonably textured areas. To aid the interpretation of our plots we also draw the resulting error curve when assuming this value as applicable.

A. Evaluation on Rectangular Objects

The error plot depicted in Figure 13 is generated by computing and evaluating Stixels for the container sequence. It is striking that the standard deviation of inter-Stixel LIDAR measurements is constant with $\sigma_L \approx 0.1\,\text{m}$. Still this is higher than expected. We reason this aspect by a weak calibration of the single laser beams.

It is apparent, that the mean error δ_S of the Stixels depth estimate increases with a square dependency. Up to 17 m it is below 0.2 m and rises to 0.45 m at a distance of 25 m. Furthermore the Stixels are always estimated as too far away. This is due to weaknesses in the complex processing chain including the camera and stereo calibration, the relative calibration between the stereo camera system and the LIDAR, the LIDAR calibration itself or the rectification.

Up to a 15 m distance the standard deviation of the Stixels mean error σ_S lies within the 3-sigma band of σ_L but increases rather strongly from there. At a distance of 25 m we obtain a standard deviation σ_S of approximately 0.7 m.

B. Evaluation on Arbitrary Shaped Objects

Figure 14 depicts the error plot for the non-constrained real-world sequence. We see that in this run the standard deviation σ_L of the inter-Stixel LIDAR measurements is not constant. By ranging from 0.3 m at 10 m to 1 m at a distance of 30 m it is significantly higher than in the plot shown in Figure 13. Yet the reason for that is not solely the LIDAR itself, but also our claim to have a constant distance across each Stixel. We draw the same conclusion regarding the over-estimation of distances. Up to a distance of 15 m the mean error δ_S of the Stixel distance is below 0.3 m and does not exceed 0.7 m at 30 m range. This approximately matches the curve shown in Figure 13. However the standard deviation σ_S increases significantly stronger with the distance. Up to 15 m it is similar to the LIDAR ($\sigma_S < 1.5 \cdot \sigma_L$) but already exceeds the $3\sigma_L$ at a distance of 26 m with 2.7 m.

C. Correction of Systematic Calibration Errors

The given plots indicate an interaction of various error sources. In this section we will address those of systematic

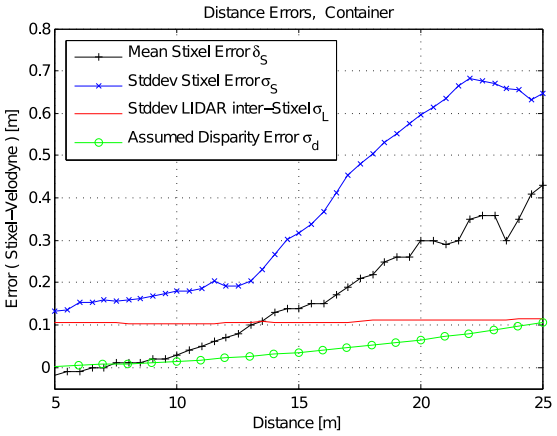

Fig. 13: Error curve of the evaluation using the container with the mean Stixel error δ_S (black), the standard deviation σ_S of that error (blue) and the standard deviation of the inter-Stixel LIDAR measurements σ_L (red). We also show the depth-dependent error, when a assuming a disparity error of $\sigma_d = 0.25\,\text{px}$ (green).

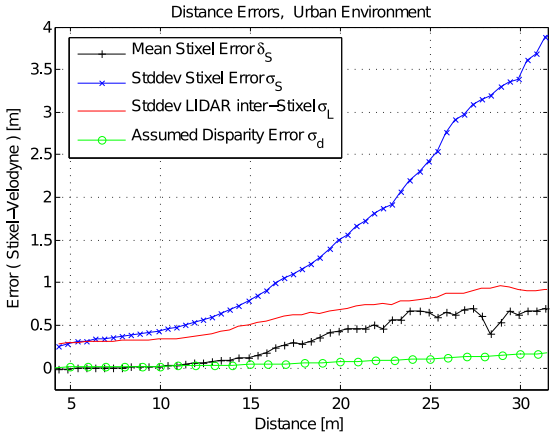

Fig. 14: The curves visualize the errors in depth estimation within real-world sequences. The notation of the graphs is identical to Figure 13.

nature. Since it is not applicable to have a permanent LIDAR supervisor running, we intend to use the container sequence as input to learn about the error statistics. We then apply this acquired knowledge to the street scenario.

The error curve of the mean Stixel error using the container follows a quadratic function of the form

$$f_e(z) = az^2 + b$$

with $a = 0.000886$ and $b = -0.0433$ quite precisely. Such a quadratic error in Cartesian space is caused by a disparity offset, which in turn can be explained by a squint angle offset in the stereo calibration. Here the factor 0.000886 equals to an offset error within the disparity space of 0.26 px and thus a squint error of 0.012 °. The constant term appears to result from an error of higher magnitude and is a remaining error in the longitudinal calibration between the stereo camera system and the LIDAR. Yet we will consider this value when

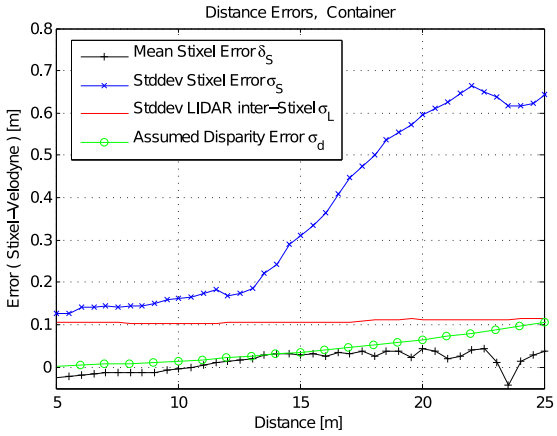

Fig. 15: Error curve of the evaluation using the container after applying the corrections to the camera calibration. The average error of the Stixel representation has been reduced significantly.

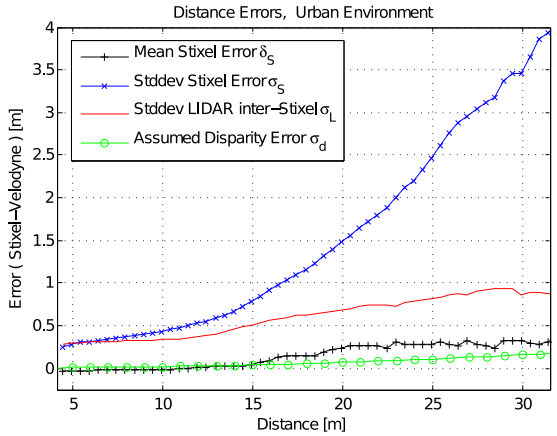

Fig. 16: The curves visualize the errors in depth estimation within real-world sequences when the correction obtained from the training sequence (container) is applied. The error is roughly reduced by a factor of two.

mapping from image to world coordinates and vice versa.

When applying this correction to our calibration we obtain the error curve for the container sequence as depicted in Figure 15. This correction results in the mean Stixel error δ_S to oscillate around the X-axis. Naturally this procedure has no effect on the disparity noise and thus does not help to reduce the standard deviation σ_S within the Stixels distances.

If we apply the same method to the scene with the parked cars using the statistics of the 10000 Stixels of container sequence as prior knowledge, we obtain the error curves given in Figure 16. In fact the error δ_S is reduced by a factor of 2, although that is not as good as in the container scenario itself. However this was to expect since the error statistics are clearly different between both sequences.

We reason the higher inaccuracy with the higher complexity this scenario offers and thus reveals remaining errors of a higher magnitude. Anyway we value a mean error δ_S of less then $0.4\,\mathrm{m}$ at $30\,\mathrm{m}$ distance under these conditions as a very good result. Besides this is close to the often claimed disparity standard deviation error of $\sigma_d = 0.25\,\mathrm{px}$.

V. Conclusion

In this contribution we have evaluated the accuracy of the Stixel World in real-world scenarios. For this purpose we used a high-performance laser scanner to generate real-world ground truth disparity images as reference input for the evaluation. The Stixel representation proves to be a very precise concept to efficiently model the environment while encoding the freespace and the height of objects. Evaluating against a reference is a necessity when dealing with safety relevant tasks and is a requirement to get aware of systematic errors in the processing chain of complex vision systems. This comparison revealed a handful of issues concerning our calibration and stereo processes that deserve dedicated consideration.

Regarding the compactness and flexibility of Stixels and the robustness of the described extraction scheme, the Stixel World is best suited for the application as a medium level representation in modern computer vision systems. By using Stixels one bridges the gap between raw input data and high-level vision by a meaningful relational structure to aid the organization in subsequent processing tasks as required in structured hierarchical vision system.

While this paper focused on the precision analysis our future work will be concerned with the robustness of this representation regarding phantoms and missed objects.

References

[1] H. Hirschmüller, "Accurate and efficient stereo processing by semi-global matching and mutual information," in *CVPR*, 2005.

[2] S. Gehrig, F. Eberli, and T. Meyer, "A real-time low-power stereo vision engine using semi-global matching," in *IEEE International Conference on Computer Vision Systems*, 2009.

[3] H. Badino, U. Franke, and D. Pfeiffer, "The stixel world - a compact medium level representation of the 3d-world," in *DAGM Symposium*, (Jena, Germany), September 2009.

[4] V. Headquarters, "High definition lidar hdl-64e s2," February 2010.

[5] U. Franke, "Real-time stereo vision for urban traffic scene understanding," in *Intelligent Vehicles 2000*, 2000.

[6] D. Scharstein and R. Szeliski, "Middlebury online stereo evaluation," 2002. http://vision.middlebury.edu/stereo.

[7] S. Morales, T. Vaudrey, and R. Klette, "Robustness evaluation of stereo algorithms on long stereo sequences," in *Intelligent Vehicles Symposium*, pp. 347 – 352, 2009.

[8] H. Badino, U. Franke, and R. Mester, "Free space computation using stochastic occupancy grids and dynamic programming," in *Workshop on Dynamical Vision, ICCV*, (Rio de Janeiro, Brazil), October 2007.

[9] A. Barth, D. Pfeiffer, and U. Franke, "Vehicle tracking at urban intersections using dense stereo," in *3rd Workshop on Behaviour Monitoring and Interpretation, BMI*, (Ghent, Belgium), pp. 47–58, 11 2009.

[10] R. Mason, J. Radford, R. Walters, D. Caldwell, B. Caldwell, and D. Kogan, "Darpa urban challenge," tech. rep., The Golem Group LLC, April 2007.

[11] V. Lepetit, F. Moreno-Noguer, and P. Fua, "Accurate non-iterative o(n) solution to the pnp problem," in *IEEE International Conference on Computer Vision*, (Rio de Janeiro, Brazil), October 2007.

[12] S. Rusinkiewicz and M. Levoy, "Efficient variants of the icp algorithm," in *3D Digital Imaging and Modeling*, 2001.

[13] J. Dolson, J. Baek, C. Plagemann, and S. Thrun, "Upsampling range data in dynamic environment," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2010.