Obstacle Detection Based on Single Frame Stereo Vision

Ciprian Pocol, Sergiu Nedevschi, Ion Giosan
Department of Computer Science
Technical University of Cluj-Napoca, Romania
{Ciprian.Pocol, Sergiu.Nedevschi, Ion.Giosan}@cs.utcluj.ro

***Abstract*—The obstacle detection from single stereo frames is a less investigated topic, while it is more tempting to add temporal information, like optical-flow (low-level) and obstacle tracking (high-level). A good understanding of obstacle detection in single frames is required for better results in obstacle detection from sequential frames. This survey uses a taxonomy that classifies the approaches based on their main processing space of the depth data. The methods for ground-obstacle separation are briefly detailed as well. At the end, there is a comparative analysis of the processing spaces and of the approaches of different research teams.**

Keywords: obstacle detection, single frame, stereo vision, survey

I. INTRODUCTION

One possibility to do 3D measurements in a generic scene, by using a moving camera, is a technique named structure from motion. In the same time, it computes the camera position relative to its position in the previous frame, this way simulating the principle of the stereo vision.

By using two distinct cameras, rigidly mounted on a rig, a real stereo vision system is obtained and it has several advantages over a mono vision system:

- the relative position of the two cameras is always the same and it can be computed with high accuracy by using specific calibration methods;
- the structure of the scene can be determined even when there is no motion;
- the calibration dramatically reduces the search space of features from one image to the other one, increasing the correspondence certainty and the depth precision.

Having the depth information available, it is tempting to go one step further and add optical flow information [1], even though the computation complexity increases significantly. The main objective of this paper is to provide a good understanding of the possibilities to detect obstacles from single stereo frames.

Some approaches use assumptions on the scene structure. For instance, it is assumed that the ground surface is planar, so that the difference of the IPM images (Inverse Perspective Mapping) emphasizes the obstacles [2]. The same assumption is used in [3] in order to quickly reconstruct the ground surface, by reducing the range of possible disparities; the approach is named "ground plane stereo".

This survey mainly focuses on approaches that aim to detect generic obstacles in generic ground scenes (using single stereo frames, as said before).

The obstacle detection and the ground detection may have common or similar parts. That's why the ground detection approaches will be briefly presented as well, because they perform the ground-obstacle separation. Some approaches detect the whole visible ground surface [11], while others detect only the limit of visible free space, which is actually the frontier between the ground surface and the beginning of the obstacles [16].

The source space of the 3D data is the perspective image enriched with depth represented by disparities. It is named disparity map and is also known as the U-V-disparity space; U and V being coordinates on the image plane. The U and V coordinates are defined relative to the optical center of the image [6]. Their resolution can differ from the image resolution: for instance one unit on the U axis = 2 pixels, in order to compress the data. The U-disparity and the V-disparity histograms are often used; they accumulate the pixels having the same (U, disparity) values and (V, disparity) values respectively.

From the disparity map, 3D points can be obtained; they are often represented in a polar or in a Cartesian space. Different approaches often use spaces that are derived from the disparity map or from the 3D space.

Due to the perspective geometry of the image formation, the coordinate system of the disparity map has a polar nature on both the lateral (U axis) and vertical (V axis) directions, while the depth's nature is based on disparities. On the other hand, the Cartesian coordinate system has its X and Y axes parallel with the image's U and V axes, while the Z axis represents the depth (some approaches may use different names for these axes).

Usually, any detection algorithm uses a main space and other secondary spaces:

- in order to use the real perception possibilities offered by the disparity space while using reasoning in the Cartesian space of the scene or vice-versa;
- in order to take into account coordinates that were lost when particular spaces were generated.

II. APPROACHES

The disparity based depth was firstly used, being the output of the stereo matching process. Later, 3D Cartesian points – metrically expressed – were obtained from the disparity map by the 3D reconstruction step.

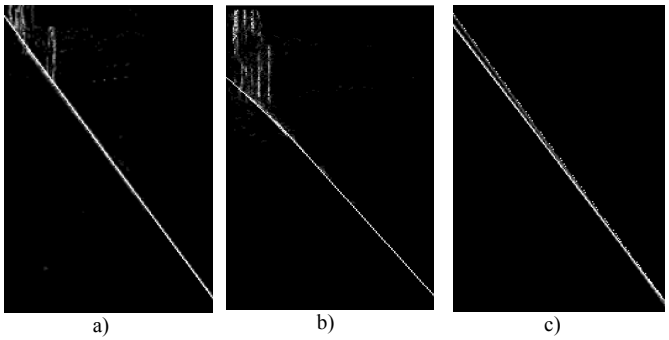
In the following, the existing detection approaches are presented. They are classified by the used depth

representation and by the main processing space that is used for obstacle detection.

A. *Depth represented by disparities*

1) *Approaches in the space of the V-disparity histogram*

At the beginning of the 2000s, Raphaël Labayrade has investigated the possibilities offered by the compact space of the V-disparity histogram [4], where each cell counts the points having the same (V, disparity) values. In this matrix, the rows correspond to the rows of the image (V) and the columns correspond to vertical planes of constant disparity, from small disparity (far range) to large disparity (near range): Fig. 1.a.



a) b) c)

Fig. 1. The V-disparity space.

The long white oblique line indicates the ground surface. In the case of a planar ground, the line is straight and is detectable by using the Hough transform. In the case of a non-planar road, it is fragment-able into quasi-planar parts (Fig. 1.b). The horizon is located at the end of the ground line.

The obstacles are localized by vertical lines. They are easily identified in a one-dimensional histogram that counts the points in the vertical planes of constant disparity. This histogram is easily obtained from the V-disparity histogram and its local maxima indicate the obstacles.

For each obstacle, its bounding rectangle in the image space can be determined (Fig. 2). The bottom (the contact with the ground) has the V coordinate of the ground line at the disparity of the current obstacle. The top can be determined in the V-disparity histogram, in the column of the current obstacle by going upwards from the ground line: the last cell with significant density is determined (actually, the search stops at the first insignificant cell). The left and the right limits can be identified by an analysis of the lateral distribution of the points having the disparity of the current obstacle and being placed between the top and bottom limits. The same analysis also differentiates multiple obstacles having the same depth.

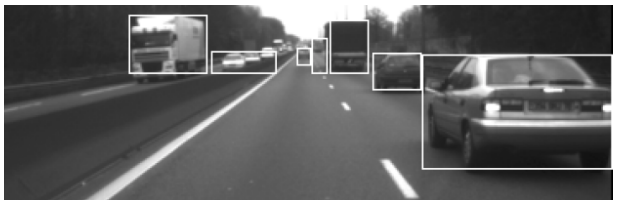


Fig. 2. Obstacles limits in the image space.

Fig. 1.c shows the superimposed ground lines from two consecutive images. It can be seen that it is important to take into account the ego-car oscillation and to compute the camera's pitch and height at every frame (this is ignored in similar approaches [24]).

In case that the ego-car has a significant roll angle relative to the road, supplementary processing is needed, as presented in [5]. A serious limitation of this approach comes when an obstacle does not pose a facet parallel with the image plane, getting spread over more disparities: in the V-disparity histogram it is diffuse.

2) *Approaches in the space of the U-disparity histogram*

In [6], the Labayrade's idea is extended by doing similar processing in the space of the U-disparity histogram also and aiming to detect any vertical planar surfaces. Anyway, non-particular oriented surfaces still remain undetected.

The detection and the labeling of the straight lines are independently done in the V-disparity histogram (Fig. 3) and the U-disparity histogram (Fig. 4). Then, in the image space, groups of pixels are built. All the pixels in a group have the same label in the V-disparity histogram and the U-disparity histogram. It is also accepted the case when all the pixels in the group have the same label in one of the histograms and are not labeled in the other one. Each group is encompassed by a polygon (Fig. 5) and their corresponding surfaces are divided into three classes:

- horizontal surfaces (ground, ceiling – the blue ones); they have a label in the V-disparity space only;
- vertical surfaces oriented towards the camera (obstacles – the red ones); they have labels in both spaces;
- vertical surfaces with other orientations (road side structures – the yellow ones); they have a label in the U-disparity space only.

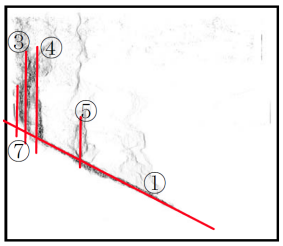


Fig. 3. The space of the V-disparity histogram.

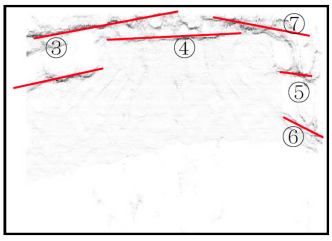


Fig. 4. The space of the U-disparity histogram.

The authors use more general formulas for the case when the camera pitch angle is significant.

The planar surfaces are not grouped into obstacles, although it would be useful in traffic scenarios and less important/applicable in indoor scenarios.

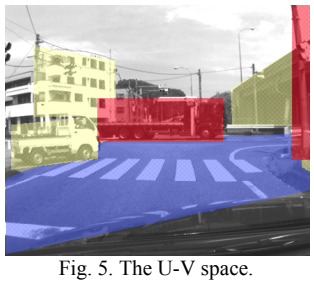


Fig. 5. The U-V space.

3) *Approaches in the U-V space*

The off-road scenes are aimed in [7]. The ground detection is done in the V-disparity space, but it is only possible when the ground's transversal profile is straight (despite the proposal at the end of the paper).

The obstacles are highlighted in the U-V space by aggregating pixels with similar disparities (Fig. 6), actually highlighting quasi-vertical surfaces. The detection is limited to a polar map of non-traversable areas, without detecting individual obstacles; in static off-road scenarios this is often enough.

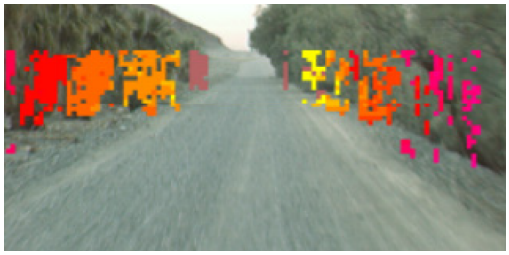


Fig. 6. Polar map of obstacle presence (the color encodes the depth).

***

In [8] off-road scenes are aimed again. On each column, the most frequent disparity is determined, getting to a stripe of pixels having that disparity (Fig. 7). Different filters are then applied in order to keep only those stripes that have obstacle specific properties. The criteria are: the pixels in the stripe must be compact, with limited gaps; neighbor stripes must have similar disparities as the current stripe (the neighborhood size depends on the depth); the stripe's slope must be significantly higher than the ground's local slope (determined in the V-disparity space); the metric height must be significant. It can be observed that the space of disparities is the main space, but Cartesian metrics are also used (pre-computed LUTs can avoid them). Results are shown in Fig. 8.

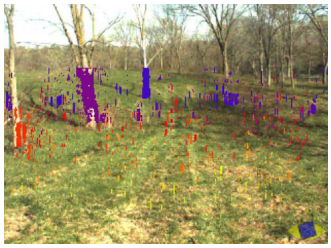


Fig. 7. Stripes of constant disparity.

Fig. 8. Final results, after applying the filters.

On each viewing direction, only the obstacle that appears the tallest in the image space will be kept. The narrow and laterally slanted obstacles may not be detected. The output is a polar map of obstacle presence.

B. Depth expressed metrically, Cartesian

1) Approaches in the X-Z space

In the older approaches of the Daimler team [26], the ground surface is supposed to be flat and horizontal. In this way, the interest 3D points [9] are those having the height between 20cm and 2m above the ground and are inside a 30x50m rectangle in front of the ego-car. The top-view Cartesian space is divided into 30x30cm cells. Thus, a bidirectional histogram is built, each cell counting the 3D points it contains; the height coordinate is dismissed. The cells having a significant number of points are grouped together. This is done by a connected components algorithm, configured to use a maximum distance between two connectable cells.

Each obtained obstacle is modeled by two rectangles, one in the top-view Cartesian space (Fig. 9) and one in the image space (Fig. 10). It can be observed that sometimes more real obstacles are grouped together.

The authors don't mention the fact that the density of the 3D points decreases (more than quadraticaly as the depth grows) and don't mention how they choose the threshold of the density of valid cells. Another missing clarification is about how they choose the maximum distance between two connectable cells.

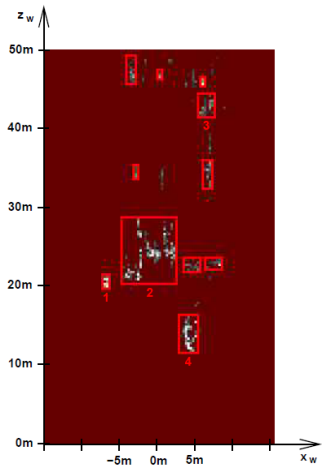


Fig. 9. Obstacle detection in the top-view Cartesian space.

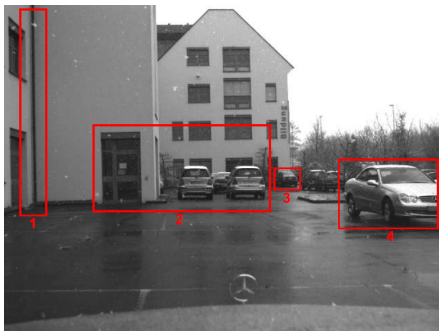


Fig. 10. The detected obstacles, in the image space, after re-considering the height coordinate (the background obstacles are not shown).

***

The team from Technical University of Cluj-Napoca [27] has an approach based on elevation maps, which separates the scene structures in three classes: ground, low platforms and obstacles. The approach is presented for the first time in [10] and culminates in [11]. The top-view Cartesian space is divided into 10x10cm cells having information such as the points density and the mean height of the points. With an elaborated algorithm, a parameterized surface is determined ($Y = -a \cdot X - a' \cdot X^2 - b \cdot Z - b' \cdot Z^2 - c$), which best models the ground surface. Thus, besides highlighting the localization of usual obstacles, low obstacles (such as sidewalks) are separately highlighted (Fig. 11).

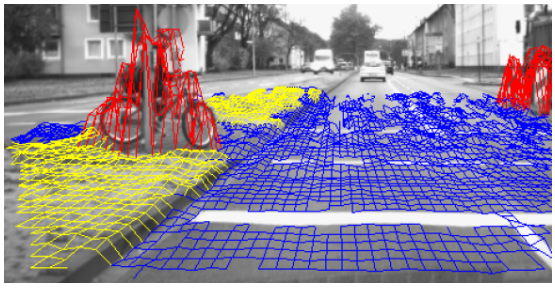


Fig. 11. Ground, low platforms and obstacles.

The cells belonging to obstacles are grouped by a simple algorithm by using a 3x3 vicinity [12] while the over-fragmentation is accepted.

Only the foreground obstacles are aimed to be detected. The polygonal contours of the visible frontier of the obstacles (Fig. 12) are determined by a radial scanning, from the ego-car position, by using an angular step that is adapted accordingly with the depth (not to miss small obstacles at higher depth). In order to detect both the low platforms and the obstacles behind them, two such scanning procedures are performed.

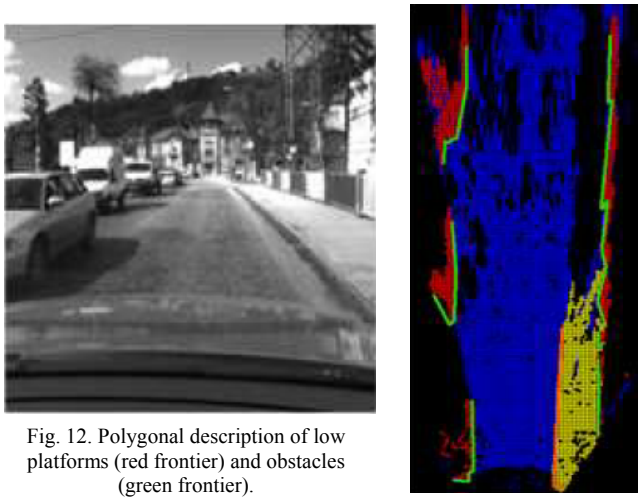


Fig. 12. Polygonal description of low platforms (red frontier) and obstacles (green frontier).

The advantage of this approach is that it can model any shape in a compact manner. Its disadvantage is that its implementation is a laborious one, because it works in the Cartesian space, while a polar space would simplify the radial scanning a lot.

2) Approaches in the U-f(Z) space

Another approach of the research group from the Technical University of Cluj-Napoca detects the ground by detecting the lane and detects the obstacles by a labeling process in a U-f(Z) space, which is a polar space.

The lane is detected in the Cartesian 3D space [13]. In the first step, the pitch angle of the ground is detected by using the 3D points situated in short range and at low height, where the ground is assumed to be quasi-planar. Then, the other parameters of a clotoidal lane model are determined, by using the 3D points belonging to the road markings (if detectable) or other types of delimiters: Fig. 13.

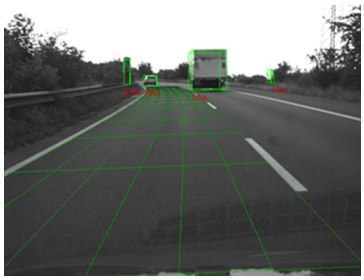


Fig. 13. Lane detection.

For obstacle detection, the used 3D points are those situated above the road, up to the height of the ego car: Fig. 14.

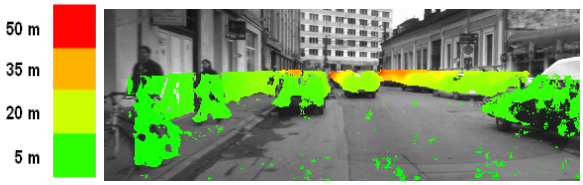


Fig. 14. The 3D points in the space of interest.

Similarly with the approach in [9] (described above), in several articles starting with [14] and culminating with [15], a top-view histogram of the 3D points is built (Fig. 15), but the used space – the U-f(Z) space – is a mixed one, having the advantage of a resolution that captures the perception possibilities of the stereo vision system. On the lateral axis, the cameras see the scene in a polar manner; that's why the U coordinate is used. On the depth axis, the Cartesian space is divided into intervals that capture both the 3D reconstruction quality and miscellaneous aspects regarding the consistency of the scene: reflective/transparent/smooth surfaces/different oriented surfaces etc. More exactly, the length of each interval is linearly related to the depth: $IntervalLength(Z) = k*Z$ (k is empirically chosen).

In this histogram, the problem of choosing the density threshold of the consistent cells is a simple one: the density tends to be constant, but still a compensation is needed, linearly with the Cartesian depth, because in the image space the height of the interest space lowers in such a fashion, leading to less points for far range. The problem of cells connectivity is also a simple one: the 3D points of an obstacle are placed in neighbor cells, regardless the depth.

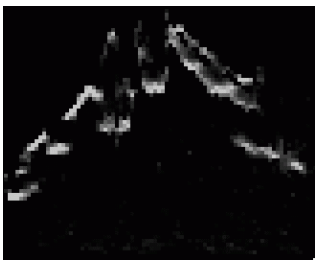


Fig. 15. The histogram of the 3D points from Fig. 14.

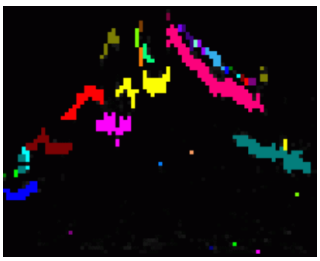


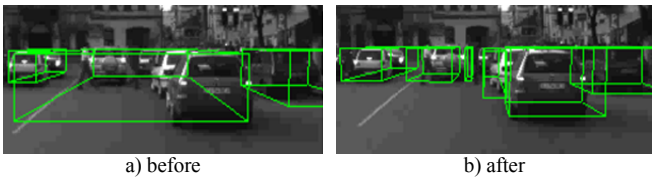
Fig. 16. The labeling of the obstacles from Fig. 14.

The grouping of the cells into obstacles is performed by a specialized labeling algorithm (Fig. 16):

- it groups the cell columns of each obstacle, from left to right;
- the cell columns are built by allowing gaps of at most the equivalent of 50cm;
- the length of each cell column is limited in order to avoid erroneous connections of different real obstacles on the same optical direction;
- the grouping of the cell columns allows a maximum lateral gap (the equivalent of 30cm) and a maximum depth difference.

In order to model the obstacles by confident cuboids, further processing steps are done:

- in order to reduce the reconstruction error, the frontier of each obstacle is refined both at the individual columns level and at the whole obstacle level;
- the concave obstacles are fragmented into two or more non-concave obstacles (Fig. 17);
- an analysis of the convex hull may decide the obstacle orientation (Fig. 18);
- some non-concave obstacles can still be far from having a cuboidal shape. They are fragmented as well (Fig. 19).



a) before b) after

Fig. 17. The fragmentation into obstacles without concavities.

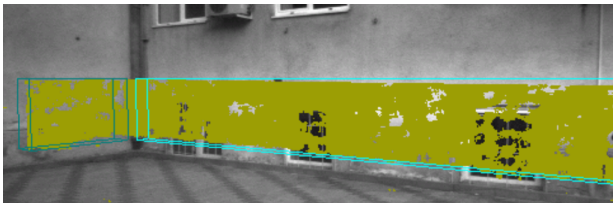
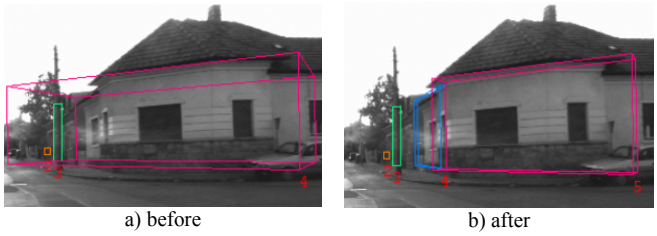


Fig. 18. Oriented obstacles.



a) before b) after

Fig. 19. The fragmentation into confident cuboids.

3) Approaches in the U-Z space

After implementing algorithms in the X-Z space, the Daimler Image Understanding group [26] has implemented algorithms in the U-Z space. In this space, they consider that for each optical direction, the structure of the scene is composed of: free space, a foreground obstacle and background structures.

The approach in [16] determines the frontier of all foreground obstacles, by analyzing a grid of the U-Z space (Fig. 20). This frontier divides the space in two areas: the free space and the background behind the foreground obstacles. In the same time, this is the ground-obstacles frontier.

The frontier consists of a single cell for each column (optical direction), leading to a cell chain of minimum cost, from left to right. Each cell has associated a cost which is the inverse of its density, meaning that high density cells are more eligible. Each pair of cells, belonging to adjacent columns, has associated a cost which is proportional with the depth difference of the two cells, meaning that it penalizes the depth variations. In order to allow real depth variations, this cost is saturated to a maximum value (several meters).

The optimum frontier is found by using an algorithm that determines the minimum cost path, from left to right. The algorithm processes a graph, having the U-Z cells as nodes and the pairs of cells on adjacent columns as arches. The authors use a dynamic programming algorithm, because it finds the optimum solution in the shortest time.

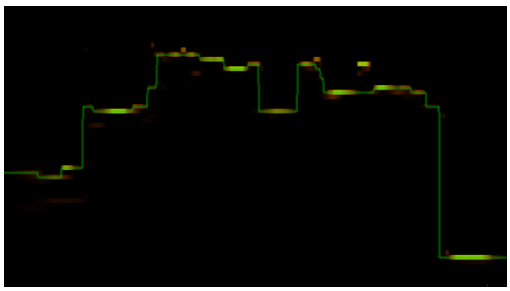


Fig. 20. The foreground obstacles frontier: a cell chain in the U-Z space.

Although it's beyond the scope of this survey, the usage of temporal data can be mentioned, as it is added in a flexible manner. The points density is accumulated over time, while taking into account the ego motion. Due to the fact that successive frames have different polar coordinate systems, the accumulation is done into a common Cartesian space: X-Z. The authors present mathematical fundaments of the implied spaces and the transformations between them, by using both the 3D measurements and their uncertainties. Thus, besides the cell level cost and the cell pair cost, the approach also adds the temporal depth variation cost, this way penalizing the difference relative to the frontier found in the previous frame. In order to allow real movements of the obstacles, this cost is saturated to a maximum value.

Fig. 21 shows the frontier in the image space. Up to this point, the authors don't aim to identify individual obstacles. It's only in the later papers when they'll do it.

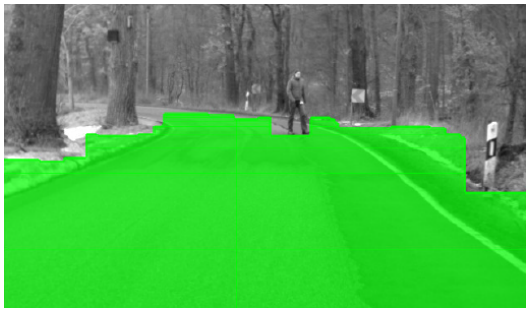


Fig. 21. The free space and the beginning of the foreground obstacles: a representation in the image space.

4) Approaches in the U-V space

The approach in [17] can deal with both on road and off-road scenarios. In the first step, it classifies the 3D points into on-the-ground and obstacle points. It works on the image space.

The first step is applied to each reconstructed pixel (3D point). At the beginning, all the pixels are marked as belonging to the ground, and then, the obstacle points are identified. The idea is that, for every point P_i, its neighbor points P_j are looked for, in a special vicinity. In principle, if the slope of their segment is above a threshold, the pair (P_i, P_j) indicates the presence of an obstacle and the two points are named *compatible*. Two points are indirectly compatible if there is a chain of compatible points that connects them. The used vicinity is composed of two vertical truncated cones, with their apex in P_i, like shown in Fig. 22. $\theta_{max} = 40^o$ represents the maximum slope that is climbable by the ego-car. A big slope that is not taller than $H_{min} = 20cm$, can be approached by the ego-car and is not considered as an obstacle. An $H_{max} = 1m$ is imposed in order to avoid connecting two points that are neighbor in the image spaces, but in the 3D space they are far away from each other by having both large depth difference and large height difference. Two points have a symmetrical relation, in the sense that one of them is above the other one or vice-versa. That's why it is enough to search in the upper cone.

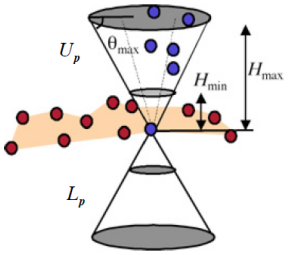


Fig. 22. The search space.

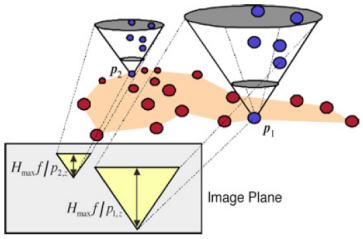


Fig. 23. Transforming the cones into triangles (and the truncated cones into trapezoids).

In practice, the 3D truncated cones are transformed into 2D truncated triangles – trapezoids – in the image space: Fig. 23. The authors discuss the possibility of having slanted truncated cones/trapezoids for the case when the ego-car

would be on a non-horizontal surface; the angles can be measured by IMUs. The authors prove that the search time of the compatible points is a linear one.

In the second step, the pairs of compatible points are grouped into individual obstacles. This is done by using a graph: the nodes are the 3D points and the arches are the pairs of compatible 3D points. An algorithm identifies the connected components (sub-graphs) that represent individual obstacles: Fig. 24. Further reasoning can be done. For instance, small obstacles (caused by wrongly reconstructed 3D points) are rejected, by analyzing the obstacle size in both the image space and the 3D space.

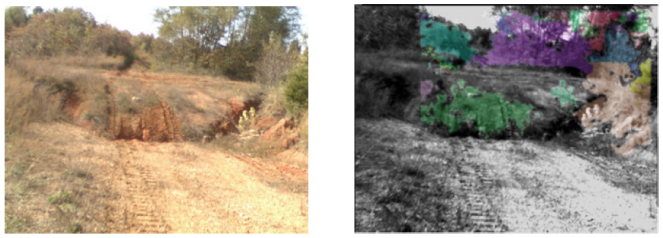


Fig. 24. The grouping into individual obstacles.

An advantage of this approach is that it can detect obstacles that have unreconstructed parts, as long as all the reconstructed points are compatible, directly or indirectly. The drawback is that, such a generic approach can go wrong in many cases, because it relies on a point pair level classification and connectivity.

***

Compared to [16] (see above), in [18], several important improvements are presented:

- The used 3D points are only those placed above the ground by using a separate algorithm for ground detection [19]
- The frontier between the free space and the beginning of the foreground obstacles is still determined in the U-Z space. In the processing step, in order to eliminate the background structures, along each column, only the first consistent cell is kept.
- The height of the obstacles is determined in the image space (U-V). In this context, the notion of "stixels" is being introduced: an array of vertical rectangles, laying on the ground-obstacles frontier and stretching up to the superior limit of the obstacles (Fig. 25).

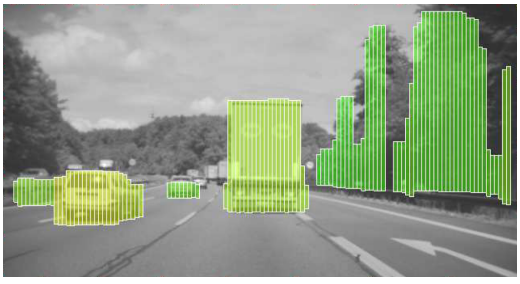


Fig. 25. Obstacles delimited by stixels.

- To determine the superior limit of the obstacles, an analysis in the image space (U-V) is done. The optimum solution is found in a similar manner as for the ground-obstacles frontier – based on costs (Fig. 26). The cost of a U-V cell reflects the depth difference relative to the ground-obstacles frontier, on the current column. The cost of a pair of cells belonging to adjacent columns reflects the height difference of those cells. In practice, the authors use more elaborated versions of these costs, combining Cartesian metrics and U-V-disparity metrics.

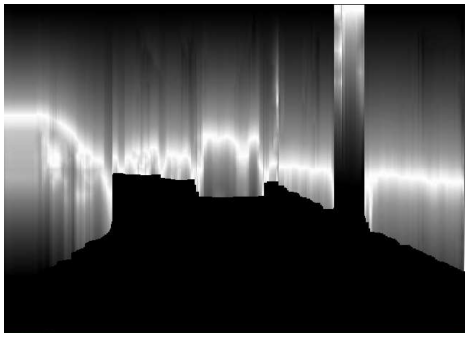


Fig. 26. The cost map that is used for separating the foreground from the background; cheaper cells are shown brighter (the scene differs from the one in Fig. 25!).

In [20], compared to [16] and [18], the next important enhancements are presented:

- The ground-obstacle frontier is now detected in the U-V space, being more compact than the U-Z space, especially for bigger depth and also because it is the space where the occlusions are assessable. The main idea remains the same: a dynamic programming algorithm determines the optimum cells path, from left to right, by using costs associated to cells and costs associated to pairs of cells from adjacent columns.
- Over consecutive frames, dynamic stixels are tracked.

An issue is that, this approach detects the stixels in two consecutive steps: ground-obstacles frontier detection and obstacles height detection. This way, the output errors of the first step can be amplified by the second one. It also lacks the attention paid to background (or semi-occluded) obstacles. In dynamic scenes, such obstacles may quickly become foreground obstacles and become subject for collision with the ego-car.

To overcome these problems, the approach in [21] uses a single step to determine not only the foreground stixels but also the multiple stixels for every optical direction, when there are multiple obstacles at different depth planes (Fig. 27). The stixels extraction is modeled as a typical MAP estimation problem (MAP = maximum a posteriori probability) and it is solved by dynamic programming.

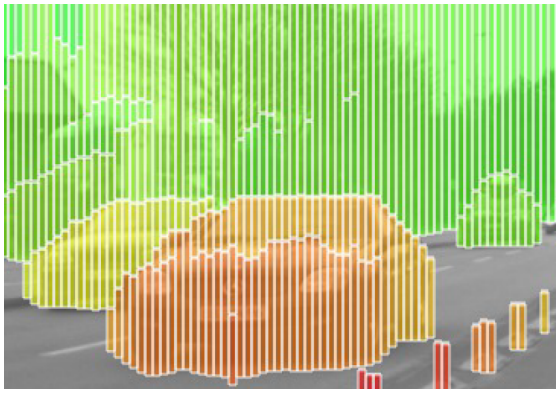


Fig. 27. Multiple stixels for any optical direction.

The grouping of stixels (Fig. 28) is also done in a MAP estimation fashion [22]: each pair of consecutive stixels has a cost that penalizes the depth difference; motion differences may also count.

Similar more complex approaches can be found in [23].

Fig. 28. Detected obstacles.

5) Approaches in the X-Y-Z space

The division of the 3D Cartesian space into cubic cells of constant size, small enough for a confident obstacle discretization, produces a lot of empty cells as compared to the cells containing 3D points. In order to avoid this drawback, in [24], the VisLab team from the Parma University [25] uses an octomap/octree organization of the 3D space. The space of interest is divided into eight equal-sized spaces. Each such space is recursively divided in the same manner till a minimum size is reached (25cm). The division is done only for those spaces containing 3D points, leading to a minimalist structure. These cells are named voxels: Fig. 29.

The authors don't explicitly specify the model they use for the ground surface, but they say that only the voxels situated at positive height are used. This means that the ground is assumed to be flat and that the ego-car doesn't oscillate with respect to the ground.

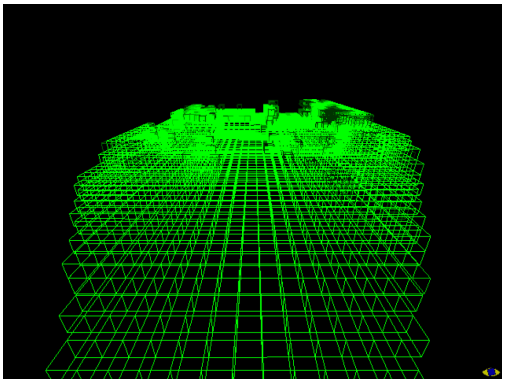


Fig. 29. The occupied space is divided into voxels.

The grouping of the voxels into obstacles is done by a region-growing algorithm. It starts from an initial voxel and adds new voxels that are not further than a threshold distance (of Chebyshev type) and that have similar color. The color similarity is computed between the color of the candidate voxel and the average color of the group; both the RGB and the HSL values are used. The used thresholds are not explained.

A real obstacle can have more than one color. That's why, the next step aggregates the groups that are close to each other: Fig. 30. Small un-aggregated groups are rejected; they might be due to the stereo reconstruction error.

The method may group together real different neighbor obstacles.

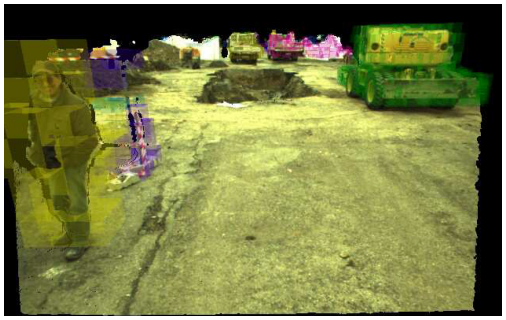


Fig. 30. Grouping of voxels into obstacles.

III. CONCLUSIONS

In order to achieve good detection results from sequential stereo frames, a good understanding of the detection from single stereo frames is firstly needed. Thus, the aim of this paper is to present and analyze existing obstacle detection approaches from single stereo frames. The ground-obstacle separation techniques are briefly discussed as well.

The analyzed approaches are classified by the used depth information and by the main processing space, because the used algorithms and the detection quality depend on such aspects.

After a deep understanding of these approaches, it is possible to conclude comparisons between the processing spaces and between the approaches.

A. Comparison between the processing spaces

The 3D space of the real scene is too vast compared to the 3D data resulted from the image space. That's why different obstacle detection approaches try to reduce the processing space.

A quick comparison between the Cartesian spaces and the disparity based spaces could be: the Cartesian spaces don't deform the reality and the disparity based spaces follow the real possibilities of perceiving the scene through stereo vision.

In what follows, the processing spaces will be sorted, starting with the weaker ones (the order is approximate and it can differ when it comes to the demands of concrete applicative contexts!):

- **The V-disparity space**. Its sole advantage is that it is compact for ground detection when the ground is transversely flat. It can't differentiate multiple obstacles at the same depth; the obstacles that are not parallel with the image plane get diffused especially at short depth.
- **The Cartesian 3D space**. It is useful only as an independent space when multiple sensors are used simultaneously or when integrating successive frames of a moving stereo vision system. Disadvantages: if it is divided into voxels with constant resolution, it is too coarse for near range or too fine for far range – adaptive resolution requires complex algorithms; the useful data is sparse; for poor stereo reconstruction, the obstacles get discontinuous.
- **The U-V spaces** with the variants **U-V-disparity** and **U-V-Z**. Advantages: the data is pretty compact; being the image space itself, it allows direct relation with different image processing algorithms and there is a 1-to-1 relation with the 3D points. Disadvantages: areas of non-reconstructed pixels may fragment

obstacles; neighbor pixels of the same obstacle can have large depth variations.

- **The X-Z Cartesian space**. Advantages: like the 3D Cartesian space, it is good as a common space for multiple sensors/multiple stereo frames; it allows Euclidian analysis in the space of the real scene. Disadvantages: like for the 3D Cartesian space, a constant resolution is not suited for the whole depth range; the 3D points density drops fast as the depth grows (more than quadraticaly).
- **The U-disparity space**. Advantages: the data is pretty compact; even when the reconstruction is poor (large errors or non-reconstructed points), the obstacles may still look contiguous; it well depicts the obstacles distribution in the scene. Disadvantage: for a tall interest space, increases the possibility of merging obstacles that have the same depth and the same lateral position.
- **The U-Z space**. Advantages: it is compact; it combines the polar lateral perception with the Cartesian depth perception. Disadvantage: the perception consistency of the obstacles placed at different depths is empirically interpreted. **The U-f(Z) space** may counteract the points spread in a custom way.

As a conclusion, for the best results, the obstacle identification and localization should be done in the U-disparity space (which best matches the perception possibilities of the stereo vision) and the post-processing should include reasoning in the 2D/3D Cartesian spaces (which are closer to the structure of the real scene).

B. The approaches of the research teams

Some teams and individuals had constantly worked in the field [25, 26, 27], while others had sporadically worked.

Some important teams are:

- The VisLab team [25], from the University of Parma, Italy, led by professor Alberto Broggi, started at the beginning of the '90s and got renown for pioneering initiatives (e.g. autonomous vehicles). As regarding the obstacle detection from single stereo frames, the processing space that they mainly used is the U-V space (the image space), the used depth was disparity based or Cartesian and it allowed natural approaches, similar with the human vision. Unfortunately, their approaches are often simple, with important limitations.
- The Daimler Team [26], Germany, led by professor Uwe Franke, started at the beginning of the '90s and got renown for seriousness and sometimes for approaches that others avoided (e.g., 6D-vision). As regarding the obstacle detection from single stereo frames, they had weaker approaches in the past (e.g. [9] by using the X-Z space). But, starting from 2007, they have perfected an approach that generates an optimum segmentation of the obstacles, based on costs associated to different possible decisions. The algorithm is scalable by being able to easily integrate different spatial/temporal information and decisions, in a probabilistic manner. In [23], the authors present failing cases and propose solutions. The limitations come from the fact that all the cases are processed in the same generic manner.
- The IPPRRC team [27], from the Technical University of Cluj-Napoca, Romania, led by professor Sergiu Nedevschi, started in 2001 and got renown for elaborated approaches (e.g., lane/ground detection). As regarding the obstacle detection from single stereo frames, they identify the obstacle areas in the U-f(Z) space and then they do further processing steps in order to get obstacles modeled as confident cuboids [15]. The limitations are caused by the forced modeling of non-cuboidal obstacles as cuboids. An alternative approach models the obstacles as polygonal frontiers [12].

Raphaël Labayrade et al. and later Zhencheng Hu et al. explained the obstacle and road detection possibilities in the V-disparity and U-disparity spaces. Although their methods are limited to detect surfaces with particular orientations (belonging to most obstacles), they have the merit of deepening the knowledge about the consistency of the disparity based information.

Roberto Manduchi et al. propose a ground-obstacle separation method, including off-road scenarios, the processing being done in the image space. Unfortunately, they end up with modeling each obstacle as a set of connected 3D points.

Although many authors classify their own approaches as being robust, with good results, their works are not further used.

IV. REFERENCES

[1] U. Franke, C. Rabe, H. Badino, and S. Gehrig, "6D-vision: fusion of stereo and motion for robust environment perception," in Deutsche Arbeitsgemeinschaft für Mustererkennung (DAGM), 2005, pp. 216-223.

[2] M. Bertozzi and A. Broggi, "GOLD: A parallel real-time stereo vision system for generic obstacle and lane detection," in IEEE Trans. on Image Processing, vol. 7, Jan. 1998, pp. 62-81.

[3] T. Williamson and C. Thorpe, "A trinocular stereo system for highway obstacle detection," in Intl. Conf. on Robotics and Automation, 1999, pp. 2267-2273, vol. 3.

[4] R. Labayrade, D. Aubert and J.-P. Tarel, "Real time obstacle detection in stereovision on non flat road geometry through V-disparity representation," in IEEE Intelligent Vehicles Symposium, June 2002, pp. 646-651, vol. 2.

[5] R. Labayrade and D. Aubert, "A single framework for vehicle roll, pitch, yaw estimation and obstacles detection by stereovision," in IEEE Intelligent Vehicles Symposium, June 2003, pp. 31-36.

[6] Z. Hu and K. Uchimura, "U-V-Disparity: An efficient algorithm for Stereovision Based Scene Analysis," in IEEE Intelligent Vehicles Symposium, June 2005, pp. 48-54.

[7] A. Broggi, C. Caraffi, R. I. Fedriga and Paolo Grisleri, "Obstacle detection with stereo vision for off-road vehicle navigation," in IEEE Computer Society Conference on Computer Vision and Pattern Recognition, June 2005, pp. 65.

[8] C. Caraffi, S. Cattani and P. Grisleri, "Off-Road path and obstacle detection using decision networks and stereo vision," in IEEE Intl. Conf. on Intelligent Transportation Systems, Dec. 2007, pp. 607-618.

[9] S. Gehrig, J. Klappstein and U. Franke, „Active stereo for intersection assistance," in Vision, Modeling, and Visualization, Nov. 2004, pp. 29-35.

[10] F. Oniga, S. Nedevschi, M. M. Meinecke, T. B. To, "Road Surface and Obstacle Detection Based on Elevation Maps from Dense Stereo", in IEEE Intl. Conf. on Intelligent Transportation Systems, 2007, pp. 859-865.

[11] F. Oniga and S. Nedevschi, "Processing dense stereo data using elevation maps: Road surface, traffic isle, and obstacle detection," in IEEE Trans. on Vehicular Technology, March 2010, vol. 59, pp. 1172-1182.

[12] A. Vatavu, S. Nedevschi, and F. Oniga, "Real time environment representation in driving scenarios based on

object delimiters extraction," in Lecture Notes in Electrical Engineering, 2011, vol. 85, pp. 255-267.

[13] S. Nedevschi, R. Schmidt, T. Graf, R. Danescu, D. Frentiu, T. Marita, F. Oniga, C. Pocol, "3D lane detection system based on stereovision," in IEEE Intelligent Transportation Systems Conference, Oct. 2004, pp. 161-166.

[14] S. Nedevschi, R. Danescu, D. Frentiu, T. Marita, F. Oniga, C. Pocol, R. Schmidt, T. Graf, "High Accuracy Stereo Vision System for Far Distance Obstacle Detection," in IEEE Intelligent Vehicles Symposium, June 2004, pp. 292-297.

[15] C. Pocol, S. Nedevschi and M.-M. Meinecke, "Obstacle detection based on dense stereovision for urban ACC systems," in Proc. Workshop on Intelligent Transportation, March 2008, pp. 13-18.

[16] H. Badino, U. Franke and R. Mester, "Free space computation using stochastic occupancy grids and dynamic programming," in Intl. Conf. on Computer Vision, 2007.

[17] R. Manduchi, A. Castano, A. Talukder, L. Matthies, "Obstacle detection and terrain classification for autonomous off-road navigation," in Autonomous Robots, Jan. 2005, vol. 18, pp. 81-102.

[18] H. Badino, U. Franke and D. Pfeiffer, „The stixel world - a compact medium level representation of the 3D-world," in Deutsche Arbeitsgemeinschaft für Mustererkennung (DAGM), 2009, pp. 51-60.

[19] A. Wedel, U. Franke, H. Badino and D. Cremers, "B-spline modeling of road surfaces for freespace estimation," in Intelligent Vehicles Symposium, June 2008, pp. 828-833.

[20] D. Pfeiffer and U. Franke, "Efficient representation of traffic scenes by means of dynamic stixels," in Intelligent Vehicles Symposium, June 2010, pp. 217-224.

[21] D. Pfeiffer and U. Franke, "Towards a global optimal multi-layer stixel representation of dense 3D data," in British Machine Vision Conference, 2011, pp. 511-512.

[22] D. Pfeiffer, F. Erbs and U. Franke, "Pixels, stixels, and objects," in Workshop on Computer Vision in Vehicles Technology, European Computer Vision Conference, vol. 7585, 2012, pp. 1-10.

[23] F. Erbs, B. Schwarz, U. Franke, "From stixels to objects – a conditional random field based approach," in IEEE Intelligent Vehicles Symposium IV, June 2013, pp. 586-591.

[24] A. Broggi, S. Cattani, M. Patander, M. Sabbatelli and P. Zani, "A full-3D voxel-based dynamic obstacle detection for urban scenario using stereo vision," in Proc. IEEE Intl. Conf. on Intelligent Transportation Systems, Oct. 2013, pp. 71-76.

[25] The Artificial Vision and Intelligent Systems Laboratory (VisLab) of Parma University: http://www.vislab.it

[26] The Daimler's Image Understanding group has most achievements presented on: http://www.6d-vision.com

[27] Image Processing and Pattern Recognition Research Centre (IPPRRC), Technical University of Cluj-Napoca: http://cv.utcluj.ro