https://doi.org/10.2352/ISSN.2470-1173.2019.15.AVM-032

Integration of Advanced Stereo Obstacle Detection with Perspectively Correct Surround Views

Christian Fuchs and Dietrich Paulus; Active Vision Group, University of Koblenz-Landau; Koblenz, Germany

Abstract

The safe navigation of large commercial vehicles implies the extended need for the surveillance of the direct surroundings. Nowadays surround view systems show heavy distortions and only have a limited range around the vehicle which may not be far enough to detect obstacles or persons early. We present a novel method which fuses an advanced perspectively correct surround view with an advanced obstacle detection. The method proposed is based on stereo vision and uses geometric modelling of the environment using a grid map data structure. The grid map is processed by a refinement algorithm to overcome limitations of the grid map when approximating shapes of the obstacles which are highlighted in the surround view.

Introduction

Monitoring the direct vehicle surroundings is a crucial task to both human drivers and (semi-)autonomously operated vehicles. In case of large commercial vehicles and combination-vehicles, the task becomes more difficult. Especially in driving maneuvers which include direction changes, the problem raises. Surround view systems are suitable to increase the operational safety in these scenarios. Current state-of-the-art methods - especially in commercial applications - only cover a short range (approx. 1.2 meters) around the vehicle and do not provide geometric correct views. Both limitations are not suitable for large commercial vehicles, which need a higher range for the surround views and perspectively correct views to gain appropriate support for navigation. To accomplish a reliable surveillance of the vehicle surroundings, the integration of advanced environment analysis in combination with a visualization of hazardous areas and obstacles is important. In addition, nowadays applications suffer from heavy distortions caused by misleading assumptions incorporated in a widely used homography-based method underneath. An automated analysis of the vehicle surroundings identifying obstacles and highlighting them in the views depicts an added value for these systems and aims towards safer navigation of these vehicles.

Recently, German authorities have started a legislative initiative on making dead space surveillance systems required for commercial vehicles, which stresses the need for a technical solution.

The need for extended ranges and additional valuable environmental information with surround view systems motivates further investigation of these topics. In order to create a sophisticated monitoring and surveillance system especially for large vehicles, we extend a method previously developed at our lab. It incorporates an advanced environment modelling technique based upon stereo data which shows a superior geometric precision. The more precise the geometric modelling, the better the perspective mapping achieved with our method gets. To enhance the value of the monitoring system, a stereo-based advanced obstacle detection algorithm is used to fuse relevant detected objects into the view.

Related Work regarding our approach is presented in the next section. The fundamentals of our work are then introduced. The extension regarding the iterative grid refinement algorithm and the stereo-obstacle detection is explained afterwards. We finally present resulting views computed using publicly available datasets.

Related Work

The following paragraphs summarize relevant aspects of research topics which must be considered in order to compute dense surround views with integrated obstacle detection.

Camera Calibration and Modelling

The correct description of the imaging process within a camera is a vital prerequisite when using a (digital) camera as a measuring device. As cameras are prone to uncertainties in their production process, adequate calibration techniques are mandatory to estimate correct geometric parametrizations.

Fundamental work on this issue has been published by Tsai and Lenz [18] and Tsai [38] including a refined method by Zhang [41]. The underlying methods enable a robust camera parameter estimation, which is widely used nowadays. However, camera lenses with a more complex geometric modelling need adapted methods concerning the distortion models, e.g. fish-eye, wide-angle or catadioptric lenses. Geyer and Daniilidis [10] published an approach for catadioptric cameras, Scaramuzza [30] proposes a method for both fish-eye and catadioptric cameras – to mention two extended methods. Especially the method by Scaramuzza [30] is commonly used in automotive applications.

Image-based Rendering

In the field of image-based rendering, the computation of virtual views is a widely discussed topic. A survey of different approaches is given by Shum and Kang [34]. The publication gives an overview of different techniques and their underlying geometric modelling strategy. However, most approaches use pre-computed or pre-modeled knowledge about the environment and are therefore not applicable in the context of surround views which are targeted towards real-time needs. Only implicit geometry methods are reasonable in this context, e.g. [17, 44, 40].

Image-based rendering methods focus on the interpolation resp. extrapolation with a camera pose close to the pose of the real camera. Up to the best of the authors' knowledge, no publication from the field of image-based rendering addresses *extensive* pose changes between real and virtual camera, which is needed for surround view computations. Yet, fundamental work is presented in these publications, although they do not include a solution for the

issue investigated.

Perspective Transformations and Virtual Views

Surround views, also known as bird's eye views, cannot be captured directly as this would imply a camera position above the vehicle. This is technically impossible as the cameras have to be mounted on the vehicle itself. Therefore, computation algorithms to compute a *virtual* camera's view are mandatory.

An obvious approach towards view transformation is perspective geometry. Hartley and Zisserman [12] and Vincent and Laganiere [39] describe the idea behind this approach:

Let all objects visible in the field of view of a camera be located on a plane in *3-D* space. Using at least four points in the *2-D* image space, the *3-D* plane can be described (projective plane). Given corresponding *2-D* points in image coordinates of the desired virtual field of view, the transformation of the projection of the plane from the captured view to the desired view can be defined using a homography matrix $H \in \mathbb{P}^{2\times 2}$. Assuming the cameras to have a constant imaging process (fixed lenses) and unaltered geometric relations between the real and virtual camera, matrix H is considered constant.

The homography matrix describes the transformation/warping between the two views and can be applied to the whole image thus transforming to the virtual view. Several publications and state-of-the-art implementations utilize homographies with various optimizations regarding camera distortions and computation cost, e.g. [20, 22, 37, 29].

However, approaches using homographies have several drawbacks which we described in prior publication s [5, 6]. The so called *Homography Shadowing Effect* explains the unnatural distortions which occur as soon as the plane assumption regarding the *3-D* location of the objects visible is violated. Yet, homographies are still used in nowadays surround view systems. Depending on the application, a perspective correctness of the view is vital for the surround view system, e.g. in off-road or construction site scenarios.

In a first step, we proposed a point-based approach towards perspectively correct bird's views which exploits depth information from stereo images and works without homographies [5]. We advanced the idea with an underlying geometric model of the ground combined with partially planar texturing algorithm in [7]. With this publication, we extend our algorithms with an iterative cell refinement and advanced stereo-based obstacle detection in order to more precisely model the ground and to include the visualization of non-ground obstacles.

Stereo and Stereo Odometry

Stereo vision enables the extraction of depth information by matching images captures from different views at the same time. It is the key to extract *3-D* information without the need for knowledge about the objects in the scene or image sequences. The basic principles behind the stereo imaging geometric – or in more general multi-view geometry – is summarized by Hartley and Zisserman [12]. Given camera intrinsics and geometric relation between the cameras, a matching between the two images enables the computation of disparities which represent the perspective mismatch and therefore the depth at a location resp. an image area.

The matching approaches separate in two main types: Keypoint-based and block-based matching. Keypoint-based algorithms match single points described by feature descriptors, e.g. [19, 21, 28], between the images and compute the corresponding disparities, e.g. [11, 15]. These algorithms tend to have a high precision in depth, but only result in sparse depth information in relation to the overall pixels in the image.

Block-matching algorithms focus on dense disparity data and try to match pixel blocks between the images. A popular method was proposed by Hirschmüller *et al.* [14]. They use mutual information and pixel-wise matching in their *semiglobal matching (SGM)* algorithm. The algorithm has been optimized to different applications and is used in lots of automotive applications such as in [33, 13].

Stereo vision is used in lots of application in the automotive field already (e. g. [26, 3, 16] and many more). The idea of using stereo vision in order to create perspectively correct surround views has not yet been discusses, up to the best of our knowledge. In a previous publication [5] we use stereo vision and transform the resulting colored point clouds into virtual camera views to create a perspectively correct surround view. The approach shows good properties towards the correct geometric handling but has sparse result images when heavy shifts in the camera rotations are applied. In another recent approach [7] we created a grid map based geometric environment model and approximate the ground space with a closed surface mesh. The surface mesh's uses triangles as geometric primitives which itself are planar. The planar patches are then properly texturized using the camera images.

A lot of different stereo datasets have been acquired and published, e.g. [27, 31, 32]. Geiger, Lenz and Urtasun [9] published the *KITTI Stereo Benchmark*, which has become very popular over the past years for automotive applications. The collection contains datasets for different purposes with different sensor modalities. The KITTI odometry datasets contain both color and gray-level stereo datasets and *3-D*-LIDAR data. We use these datasets in our work.

As the plan is to temporarily integrate the stereo data, a precise knowledge about the relative movement between two cameras is mandatory. Visual odometry is a key technique to compute the relative poses directly from stereo frames. We define a pose as a combination of position orientation, which is represented by a tuple of a *3-D*-vertex and a rotation quaternion. We follow a method proposed by Cvišić and Petrović [4] in this publication. It is amongst the best-ranking algorithms on the KITTI odometry benchmark [9] at the moment.

Closed Surface Representation

The representation of closed surfaces in the context of environment modelling follows two main techniques: Variants of heightmaps and voxel-based truncated signed-distance function (TSDF) [25].

Heightmaps are widely used in SLAM technologies. They are optimized simultaneously to the pose of the sensor/camera: Heightmaps are used as the underlying data structure in various SLAM algorithms [35, 43, 42]. They solve the localization and orientation (pose) problem at the same time the geometry is optimized. Yet, none of these publication focusses on the issue of high-quality texturing.

Motooka et al. [23] show impressive quality in a photometrically optimized texturing of a heightmap. They use a large num-

ber of camera images. However, they state that their method is not suitable for an online application.

Tanner et al. [36] address large area mapping and utilize TSDFs. They show a detailed colored mesh representation of the environment. Yet, they focus on large area mapping rather than texture details. They apply one color per voxel which results in a texture resolution of one color per 10 cm.

A *3-D* reconstruction approach is presented by Gallup, Frahm and Pollefeys [8]. They cluster depth images into a voxel structure and combine them with a heightmap. Their focus is on a continuous heightmap at the cost of texturing quality.

Geometric Grid Map Modelling

To create the perspectively correct surround view, depth data from a multi-stereo setup is accumulated and integrated into a surface model of the environment to represent the environments geometry. The approach is extended from a previous method we published [7] and integrates refinements steps which match the real geometry of the scene more precisely.

We use a grid map as the basic data structure. The grid map is orthogonal and equidistant and has a fixed reference to the world coordinate system. It is also aligned to the world coordinate system which is defined on a corner of a grid cell. This implies that two of the world coordinate system's axes (*xy*-plane) define the bases of the grid map.

Parameter $g \in \mathbb{R}$ defines the side length of a single cell and determines the area covered by each grid cell. The grid map samples the assumed ground plane of the world coordinate system. As the vehicle moves over the grid map, it is important to maintain the grid map local around the vehicle, as only the direct vehicle surroundings are of particular interest:

As we plan to accumulate *3-D* data in the grid, it is important to maintain the world reference and adapt the vehicles current pose to the fixed reference. For each vehicle pose with reference to the world coordinate system and thus the origin of the overall grid map, the current vehicle cell which corresponds to the vehicles pose can be determined. This cell is denoted as the vehicle or center cell c_c. Over the system runtime, the vehicle is expected to move large distances, so that an iterative update of the grid map is needed to keep the amount of data manageable.

The grid map data structure is extended to a *local shifting grid map* which always uses the center cell c_c as the center of the grid map while maintaining the world reference and the sampling of the ground plane.

Therefore, the grid definition must be extended with the grid extents: Parameter $e \in \mathbb{N}$ defines the number of cells around the center cell c_c. The extents of the local shifting grid map are given by the index range $([-e,e],[-e,e]) \in \mathbb{Z}^2$ which yields $(2 \cdot e+1)^2$ cells in the grid. As soon as the vehicle enters a center cell at timestep τ which defers from the center cell at timestep $\tau-1$ ($c_{c;\tau} \neq c_{c;\tau-1}$), the grid map data structure must be updated as shown in Figure 1.

The *3-D* point cloud from the stereo system is added to the grid map for each time step. In order to integrate the data correctly, a visual odometry approach [4] is utilized to compute the corresponding delta poses of the vehicle. The relations of the coordinate systems involved are shown in Figure 3. The *3-D* points get accumulated in their corresponding cell. Afterwards, each grid cell is processed independently. Depending on the heights of

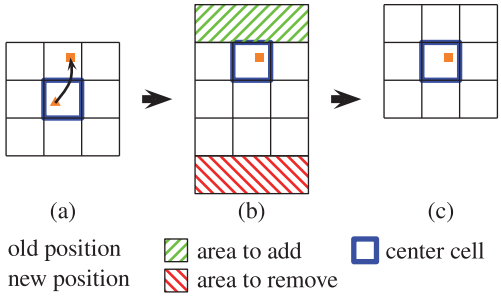


Figure 1: Local Shifting Grid Map: As soon as the vehicle's *2-D* ground position leaves the old reference center cell (a), the center cell is updated using the new position (b). The new center cell causes the selection of the cells to add resp. to remove from the grid (b). The grid's extents are then updated according to the new center cell (c). The contents of the grid cells remaining in the grid are maintained while updating the extents.

the *3-D* points and their original distance to the sensor as a quality criterion, a heuristic algorithm [7, 24] is applied to determine if the cell is an obstacle or a flat cell. The decision concerning an obstacle classification is made upon a statistic measurement of the height spread within the cell. In case of a flat cell, an appropriate cell height is computed from the points' heights and assigned to the cell. As soon as the points of – for example – a vertical object lies within a cell and the thresholds for the obstacle detection are exceeded, the whole cell is regarded an obstacle cell.

The grid map with its topology and accumulated and post-processed data is used to compute a closed surface approximation of the ground. In order to create partially planar surface elements, a triangle mesh is created upon the grid. The surface has its vertices at the corners of the grid's cells. The vertex β at the origin of cell $c_{u,v}$ is computed as follows (with $\xi(u,v)$ the height of cell $c_{u,v}$):

$$\beta(u,v) = \left(g \cdot u, g \cdot v, \frac{1}{4} \cdot \sum_{i=-1}^{0} \sum_{j=-1}^{0} \xi(u+i, v+j) \right)^{\mathrm{T}}$$

An example for the closed surface can be seen in Figure 2. In order to create a surround view, a texture layer is added to the grid map. The resolution of the texture layer is defined with parameter $r \in \mathbb{N}$ and yields a texture resolution of $r \cdot r$ pixels per cell.

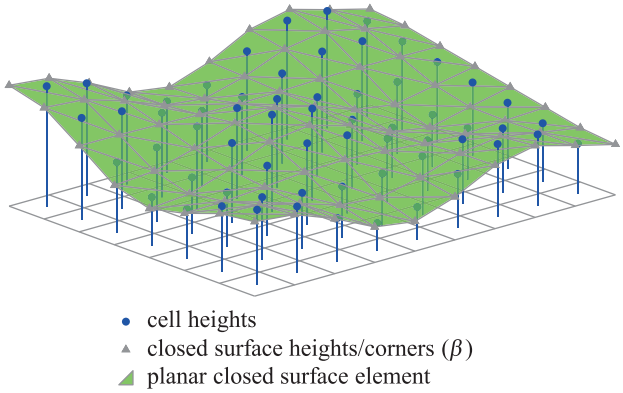


Figure 2: Example for a closed surface over a grid map. The single surface elements are partially planar.

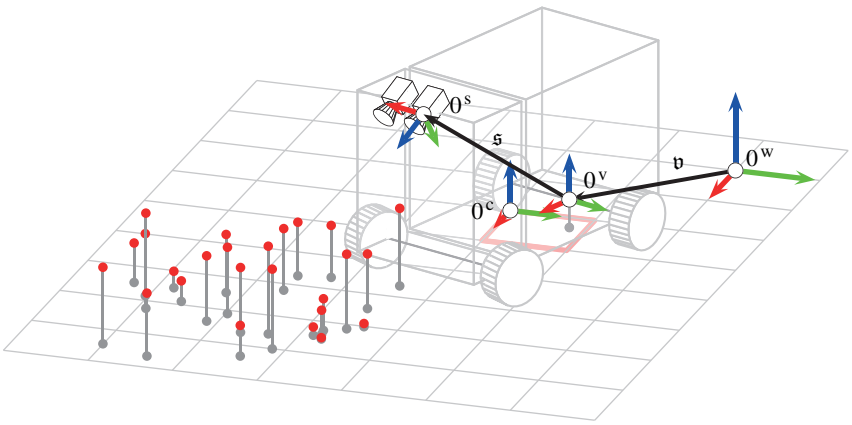


Figure 3: Grid map with world coordinate system (w), vehicle coordinate system (v), center cell c_c (c) and sensor coordinate systems (s). Pose $\mathfrak{v}$ transforms from world the vehicle coordinate system. The sensor pose $\mathfrak{s}$ defines the coordinate system of the sensor data within the vehicle. The *3-D* points are transformed from the coordinate system of the camera (s) for the grid's cells.

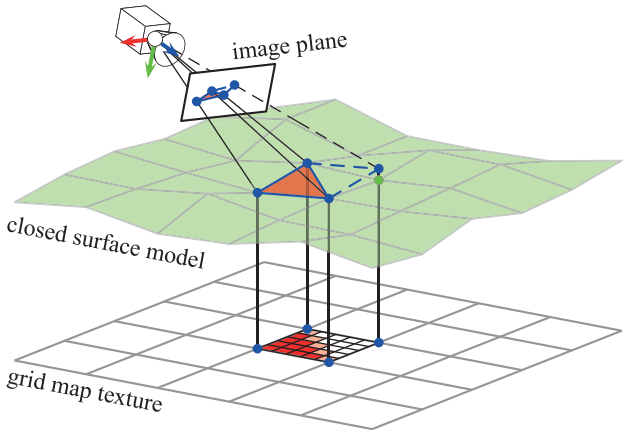


Figure 4: Texture extraction from camera images using the closed surface model.

As the relative position of the cameras to the closed surface model is known, the camera images can now be used to texturize it. As the partially planar elements approximate the real world, they can be used as a reference geometry to project the camera images onto.

Figure 4 illustrates the process: The projection of a triangle to the camera at a known relative pose is described by the known camera intrinsic. It is therefore possible to determine the area in the image which covers the part of the world that is approximated by the triangle. As the triangle is partially planar, a homography can be used to formulate the warping process between the projected triangle excerpt in the image and the corresponding texture patch. However, at least four points are needed to formulate a homography matrix. A virtual fourth point can be added to the triangle by computing a co-planar point on the triangle plane.

Iterative Surface Refinement

The method proposed decides on a cell basis, if the corresponding space is regarded as an obstacle or not. This implies, that an obstacle can only be narrowed down up to the precision of the grid's resolution $g \in \mathbb{R}$. However, it is important to approximate the obstacle representation more precisely. We therefore use a two-staged approach for the obstacle representation in the method proposed. At first, the heuristic approach explained above – which allows a simple and therefore fast processing – towards the obstacle detection is used. In a second stage, we refine promising candidate obstacle cells to enlarge the bottom area which leads to a larger area which can be visualized in a surround view.

In order to select candidate cells, the grid topology of the single cells is used. It only makes sense to revisit previously as obstacles classified cells, when the cell is neighboring non-obstacle cells. The basic idea behind this is to be able to enlarge the nearby ground approximation and the texturable area. In order to maintain reasonable splitting, we limit the texture resolution $r \in \mathbb{N}$ to $r = 2^n$ with $n \in \mathbb{N}$.

Let $\psi : \mathbb{Z}^2 \to \{0;1\}$ be a Boolean function which describes if a cell c in the local shifting grid map G is classified as an obstacle in the first stage. Using this function, a neighbor relation is defined, which describes cells that share an edge with the reference cell in the grid. In case of the orthonormal equidistant grid structure, the neighbor relation equals to a 4-neighborhood as known from image processing. The neighbor relation $\rho(u,v)$ for a cell $c_{u,v}$ at position $(u,v) \in \mathbb{Z}^2$ in the grid is defined as:

$$\rho(u,v) = \left\{ c_{u-1,v}, c_{u,v+1}, c_{u+1,v}, c_{u,v-1} \right\}$$

The set of obstacle cells, which are considered as candidates for the grid refinement, is denoted as Ψ:

$$\Psi = \left\{ c_{i,j} \,\middle|\, \psi(c_{i,j}) \wedge \neg \left(\bigwedge_{a \in \rho(i,j)} \psi(a) \right) \right\}$$

Given an obstacle cell $c_{u,v} \in \Psi$, the corresponding cell point cloud $C_{u,v}$ is transformed so that the components of the point which refer to the grid plane (xy-plane, see Figure 3) are normalized to locally normalized relative cell coordinates, leaving the z-component (height) unaltered. This yields for the normalized point cloud $O_{u,v}$:

$$O_{u,v} = \left\{ \left(\left(p_x^c \cdot g^{-1}, p_y^c \cdot g^{-1}, p_z \right)^T, d \right) \,\middle|\, (p,d) \in C_{u,v} \right\}$$

Algorithm 1 Recursive cell splitting

```
1:  procedure SUBDIVIDECELL(O, r)
2:      if r < 1 or |O| < t then                ▷ t ∈ ℕ a threshold
3:          return { }
4:      S ← { }
5:      for i := 0 to 1 do
6:          for j := 0 to 1 do
7:              B ← D_{i,j}(O)                  ▷ Compute sub-cell
8:              if ¬ψ(B) then                   ▷ Non-obstacle cell?
9:                  S ← S ∪ {B}
10:             else
11:                 S ← S ∪ SUBDIVIDECELL(B, 0.5 · r)
        return S
```

Now let the obstacle cell $c_{u,v} \in \Psi$ be a candidate for the grid refinement. The corresponding normalized point cloud $O_{u,v}$ is subdivided iteratively to approximate possible ground space in the cell area. The area of cell $c_{u,v}$ is split into four equally sized sub-cells $D_{i,j}$ which are combined in the set $S_{u,v}$:

$$S_{u,v} = \{D_{0,0}, D_{1,0}, D_{0,1}, D_{1,1}\}$$

The computation of the sub-cells in the fist stage is computed as follows:

$$D_{i,j}(O_{u,v}) = \{(A \cdot p, d) \,|\, (p,d) \in O_{u,v}, \\ p_x \in [i \cdot 0.5, i \cdot 0.5 + 0.5], \\ p_y \in [j \cdot 0.5, j \cdot 0.5 + 0.5]\}$$

Matrix $A \in \mathbb{R}^{3\times3}$ is a is a scaling matrix which normalizes the points $p \in \mathbb{R}^3$ within the sub-cells in an analogue way:

$$A = \begin{pmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

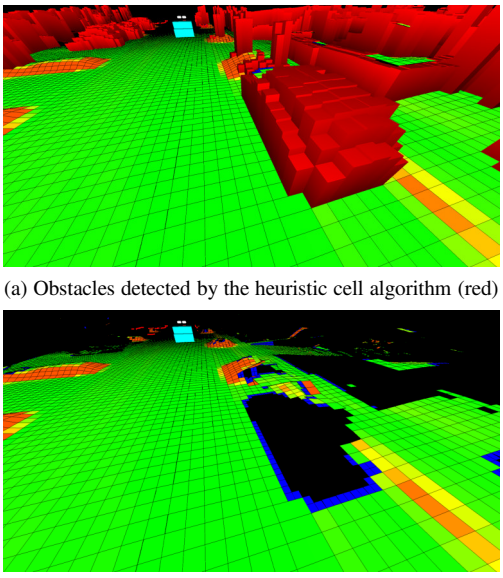
The subdivision of the cells can be applied iteratively as shown in Algorithm 1. In case a sub-cell is classified as ground, it can be used to extend the closed surface mesh. Otherwise the iterative refinement recurses until at least one of the termination criteria is met: Too few points left for the sub-cell (threshold) or the potentially left texture gain drops below the area of one pixel.

The principle behind the iterative grid refinement is depicted in Figure 6.

Advanced Stereo Obstacle Detection

The obstacle detection based on a heuristic approach with the iterative grid refinement is capable of approximation the ground geometry in an advanced way. However, it is desirable to provide more precise information concerning the nearest obstacles to the vehicle and its shape. The goal is to fuse this information with the perspectively surround view in order to provide accurate information about nearby obstacles to the driver/algorithms.

Based on the work by Badino, Franke and Pfeiffer [1], we integrate an adapted stereo obstacle detection approach which generalizes their method while tailoring it to the surround view scenario.



(a) Obstacles detected by the heuristic cell algorithm (red)

(b) Refined sub-cells (blue)

Figure 5: Example for the grid refinement: The obstacle cells as red solids (a) are refined by the blue sub-cells (b)

The authors directly work on the disparity maps of the stereo camera in their "Stixel World" approach and use a camera pose which looks in direction of the horizon parallel to the ground as a prerequisite. Given the disparity image, they search for the nearest obstacle to the camera per disparity image column. Due to the camera pose constraint, each column of the disparity image represents a sampling when rotating in yaw direction in means of the camera/the vehicle.

Using columns wise histograms of the disparities (and a spline-based approximation of the ground to cut out ground areas) which are represented in a joint occupancy grid, they start a local maxima search in order to find the nearest obstacle to the camera. Obstacle candidates are then combined over the single image columns to find connected obstacle instances. In further steps, the heights of the objects found are determined.

Their approach shows impressive results when the camera pose prerequisite is fulfilled. However, the approach is not directly adaptable to the scenario discussed here, as the (stereo) cameras used for surround view generation usually are not aligned towards the horizon yet facing downwards towards the ground.

We propose and use a generalization of the approach by Badino, Franke and Pfeiffer [2] in order to adapt their method to the surround view scenario. Given the initial stereo point cloud, a coordinate system equal to a camera facing the horizon is generated. This coordinate system has its origin in the camera's coordinate system but is rotated adequately. Let pose $\mathfrak{a}$ describe the transformation between the camera's coordinate system c and the coordinate system h of the virtual camera facing the horizon. With $T_\mathfrak{a} \in \mathbb{P}^{3\times3}$ the transformation matrix resulting from pose $\mathfrak{a}$, the point cloud can be transformed to the desired coordinate system from the initial point cloud P^c in the virtual camera's coordinate system:

$$P^\mathrm{h} = \{T_\mathfrak{a} \cdot \tilde{p} \,|\, p \in P^\mathrm{c}\}$$

The original pixel topology of disparity image is of course not available anymore in the transformed point cloud P^h. How-

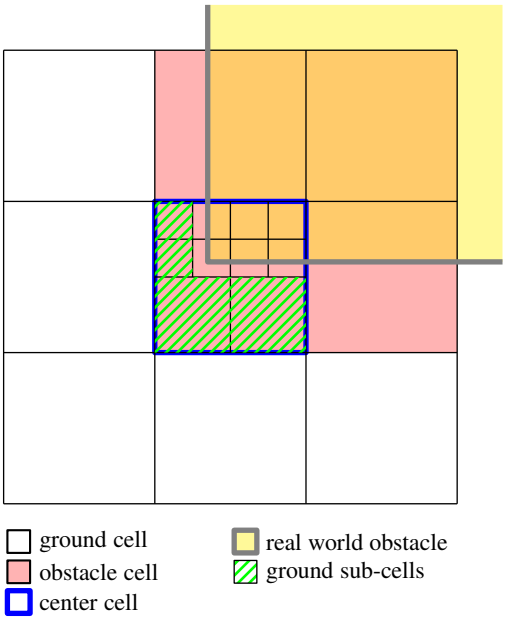


Figure 6: Principle of the iterative grid refinement

ever, an alternative to the disparity image column can be set up by clustering the point cloud using a discretized yaw angle (relative to the camera's orientation).

Given a sample step Δ_α – which can easily be derived from the camera intrinsic to match the opening angle of image column in the original disparity image – the candidates for the column-based histogram can be extracted. At this point, the disparities itself have already been transformed to *3-D* points. It is reasonable to focus on the *distances* of the points to the virtual view for the histogram, as the semantic is appropriate compared to the disparities.

With this generalization of the method, we can apply it to the stereo data from cameras violating the prerequisite in the publication by Badino, Franke and Pfeiffer [2]. As the height of the obstacle is not of importance in means of surround view computation as we view the obstacle from a bird's view, no sophisticated solution is needed here. An overview of the steps involved in the stereo obstacle detection is given in Figure 7. The obstacles extracted from the stereo images using the method described are fused into the surround view algorithm as described above and enable the clear presentation of obstacle boundaries.

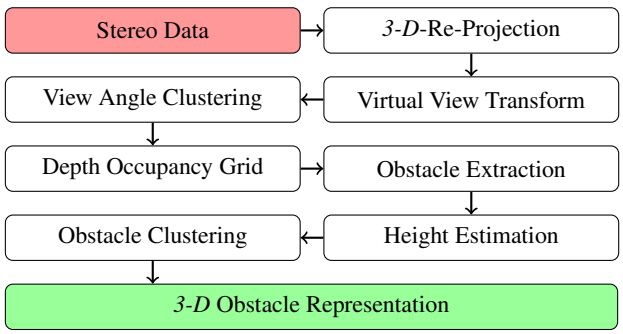


Figure 7: Components overview for the Advanced Stereo Obstacle Detection

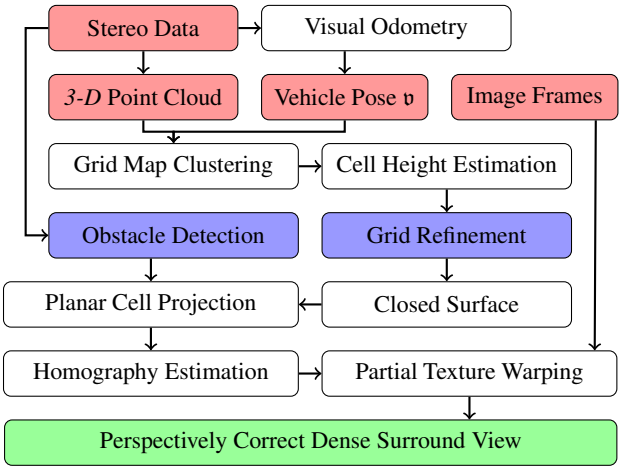


Figure 8: Overview of the system's components and the dataflow between the components. Red boxes indicate input data, the green box indicates the output. The blue boxes mark the topics discussed in deep in this publication.

Test Results and Conclusion

We propose an approach towards perspectively correct surround views, which conducts a geometric modelling based on a local shifting grid map. The environment model is used to create a closed surface representation of the vehicle surroundings which provides partially planar surface elements. These elements are used to achieve a high quality texturing of the vehicle surroundings and thus a high quality surround view.

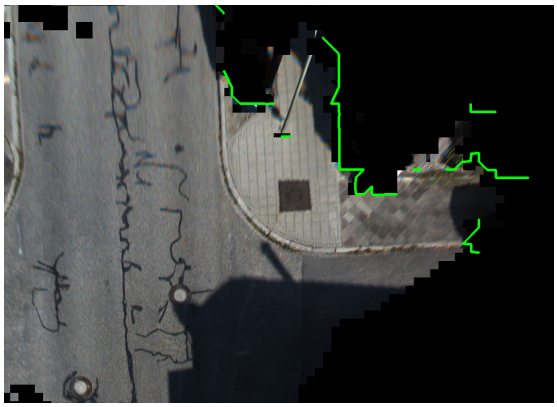
We use the stereo data from the same sensors in three different ways: We first compute *3-D* point data from the data and then use it for both geometric modelling and obstacle detection. The stereo frames are also used to solve the visual odometry problem in order to compute relative poses between the timesteps. These poses are needed for the temporal integration of the *3-D* data in the grid map. The accumulated *3-D* points are then used to create a heightmap which is the basis for our closed surface model. The grid map is then refined to further enlarge the ground surface approximation in order to increase the viewable area in the surround view. The original camera images are used to texture the *3-D* model of the environment.

The advanced stereo obstacle detection enables the precise integration and visualization of non-ground objects within the surround view which clearly marks an advantage. Figure 8 shows the components of the system developed and explains the data flow within.

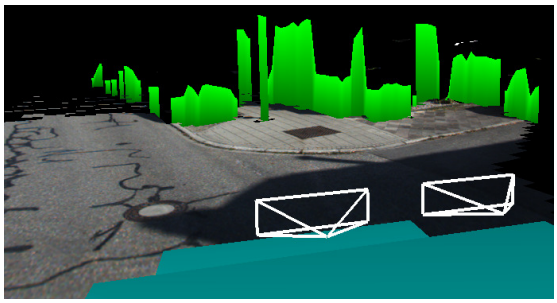
To enable the reproduction and comparison of our method, we rely on a publicly available dataset as input to compute the result images presented (KITTI Odometry Datasets [9]). The results range up to $\approx 9\,\mathrm{m}$ around the vehicle and were computed using the following parameters:

$$e = 30, \qquad g = 3.5^{-1}\,\mathrm{m} \qquad r = 15\,\mathrm{px}$$

Figures 9, 10 and 11 show results computed by our method. The shape of the obstacles (e.g. cars, lantern, house, distribution boxes) in Figure 9 is clearly visible in the surround view/ground texture. Even the small lantern was detected correctly. The two cyclists – which are very important to be detected – in Figure 10



(a) Ground Texture (orthographic).
Green lines mark obstacles detected by the advanced algorithm.



(b) *3-D* View with Car Pose.
The green walls mark obstacle boundaries found.

(c) Scene image (KITTI odometry dataset #1, frame 102 [9]))

Figure 9: Perspectively correct dense surround view result using KITTI odometry dataset #1.

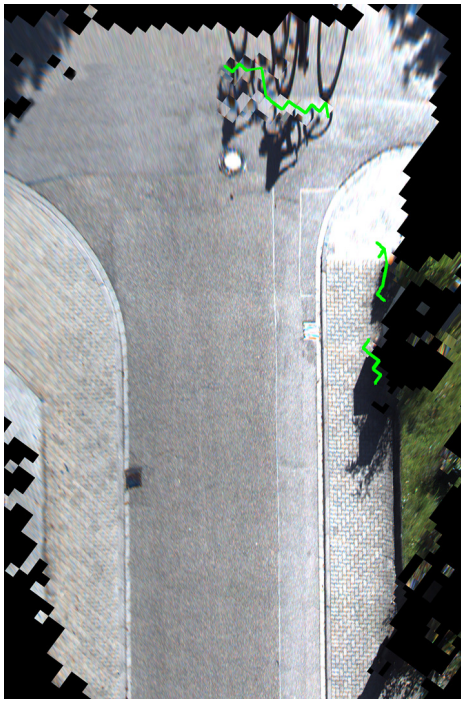
are detected and marked correctly. The car in the shadow in Figure 11 is highlighted in the ground texture and therefore visible despite of the dark areas in the original images.

The approach proposed shows sophisticated image quality. The geometric modelling enables a perspective correct view, although roughness is present on the ground. The grid refinement enables to enlarge the visible ground space so that the area covered by the surround view is increased. The obstacles by the advanced stereo based algorithm highlight potentially dangerous borders around the vehicle and pose a gain in safety for the operator/driver.

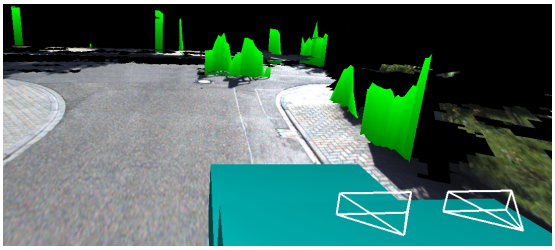
For further work it is planned to enhance the texture resolution and the geometric modelling to gain an even better representation of the environment.

References

[1] H. Badino, U. Franke, and D. Pfeiffer. The Stixel World - A Compact Medium Level Representation of the 3D-World. In J. Denzler, G. Notni, and H. Süße, editors, *Pattern Recognition*, pages 51–60,



(a) Ground Texture (orthographic).
Green lines mark obstacles detected by the advanced algorithm.

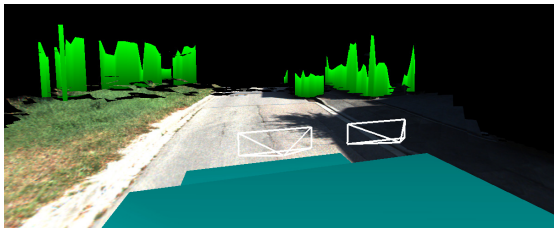


(b) *3-D* View with Car Pose.
The green walls mark obstacle boundaries found.

(c) Scene image (KITTI odometry dataset #5, frame 2070 [9]))

Figure 10: Perspectively correct dense surround view result using KITTI odometry dataset #5.

(a) Ground Texture (orthographic).
Green lines mark obstacles detected by the advanced algorithm.



(b) *3-D* View with Car Pose.
The green walls mark obstacle boundaries found.

(c) Scene image (KITTI odometry dataset #3, frame 316 [9]))

Figure 11: Perspectively correct dense surround view result using KITTI odometry dataset #3.

Berlin, Heidelberg, 2009. Springer Berlin Heidelberg.

[2] H. Badino, R. Mester, T. Vaudrey, and U. Franke. Stereo-based free space computation in complex traffic scenarios. *Proceedings of the IEEE Southwest Symposium on Image Analysis and Interpretation*, pages 189–192, 2008.

[3] A. Broggi, C. Caraffi, R. I. Fedriga, P. Grisleri, and I. Parma. Obstacle Detection with Stereo Vision for Off-Road Vehicle Navigation. In *IEEE Conference on Computer Vision and Pattern Recognition*, page 65, San Diego, 2005.

[4] I. Cvišić and I. Petrović. Stereo odometry based on careful feature selection and tracking. In *2015 European Conference on Mobile Robots, ECMR 2015 - Proceedings*, 2015.

[5] C. Fuchs and D. Paulus. Perspectively Correct Bird's Views Using Stereo Vision. In *Autonomous Vehicles and Machines Conference, IS&T Electronic Imaging 2017*. IS&T Digital Library, 2017.

[6] C. Fuchs and D. Paulus. Perspectively correct construction of virtual views. In *Proceedings of the 6th International Conference on Pattern Recognition Applications and Methods - Volume 1: ICPRAM*, pages 626–632. Scitepress, 2017.

[7] C. Fuchs and D. Paulus. Dense Surround View Computation with Perspective Correctness. In *Autonomous Vehicles and Machines Conference, IS&T Electronic Imaging 2018*, pages 282–1–282–8, Springfield, USA, 2018. Society for Imaging Science and Technology (IS&T).

[8] D. Gallup, J.-M. Frahm, and M. Pollefeys. A Heightmap Model for Efficient 3D Reconstruction from Street-Level Video. *Int. Conf. on 3D Data Processing, Visualization and Transmission*, 6, 2010.

[9] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3354–3361, 2012.

[10] C. Geyer and K. Daniilidis. Catadioptric projective geometry. *International Journal of Computer Vision*, 45(3):223–243, 2001.

[11] W. E. Grimson. Computational experiments with a feature based stereo algorithm. *IEEE transactions on pattern analysis and machine intelligence*, 7(1):17–34, 1985.

[12] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, New York, NY, USA, 2 edition, 2003.

[13] S. Hermann and R. Klette. Iterative semi-global matching for robust driver assistance systems. In *Asian Conference on Computer Vision*, pages 465–478, 2013.

[14] H. Hirschmüller. Stereo processing by semiglobal matching and mutual information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(2):328–341, 2008.

[15] R. Horaud and T. Skordas. Stereo correspondence through feature grouping and maximal cliques. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(11):1168–1180, 1989.

[16] C. G. Keller, C. Hermes, and D. M. Gavrila. Pattern Recognition: 33rd DAGM Symposium, Frankfurt/Main, Germany, August 31 – September 2, 2011. Proceedings. In R. Mester and M. Felsberg, editors, *DAGM*, chapter Will the P, pages 386–395. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.

[17] S. Laveau and O. Faugeras. 3-D scene representation as a collection of images. *Proceedings of 12th International Conference on Pattern Recognition*, 1, 1994.

[18] R. K. Lenz and R. Y. Tsai. Techniques for calibration of the scale factor and image center for high accuracy 3-D machine vision metrology. *IEEE Transactions on Pattern Analysis and Machine In-*

telligence, 10(5):713–720, 1988.

[19] S. Leutenegger, M. Chli, and R. Y. Siegwart. BRISK: Binary robust invariant scalable keypoints. In *2011 IEEE International Conference on Computer Vision (ICCV)*, pages 2548–2555, 2011.

[20] Y. C. Liu, K. Y. Lin, and Y. S. Chen. Bird's-eye view vision system for vehicle surrounding monitoring. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 4931 LNCS:207–218, 2008.

[21] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[22] L. Luo, I. Koh, S. Park, R. Ahn, and J. Chong. A software-hardware cooperative implementation of bird's-eye view system for camera-on-vehicle. *2009 IEEE International Conference on Network Infrastructure and Digital Content*, pages 963–967, 2009.

[23] K. Motooka, S. Sugimoto, M. Okutomi, and T. Shima. 360-Degree 3D Ground Surface Reconstruction Using a Single Rotating Camera *. In *Proceedings of 7th Workshop on Planning, Perception and Navigation for Intelligent Vehicles (PPNIV2015)*, pages 147–152, 2015.

[24] F. Neuhaus, N. Wojke, C. Winkens, B. Krayer, D. Paulus, and M. Häselich. Autonomous 3d Terrain Mapping and Object Localization for the Spacebot Camp 2015. In *International Symposium on Artificial Intelligence, Robotics and Automation in Space (i-SAIRAS)*, 2016.

[25] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohi, J. Shotton, S. Hodges, and A. Fitzgibbon. KinectFusion: Real-time dense surface mapping and tracking. In *2011 10th IEEE International Symposium on Mixed and Augmented Reality*, pages 127–136, 2011.

[26] D. Pfeiffer and U. Franke. Towards a Global Optimal Multi-Layer Stixel Representation of Dense 3D Data. *Procedings of the British Machine Vision Conference 2011*, pages 51.1–51.12, 2011.

[27] D. Pfeiffer, S. Gehrig, and N. Schneider. Exploiting the power of stereo confidences. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 297–304, 2013.

[28] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski. ORB - an efficient alternative to SIFT or SURF. In *IEEE International Conference on Computer Vision (ICCV)*, pages 2564–2571, 2011.

[29] T. Sato, A. Moro, A. Sugahara, T. Tasaki, A. Yamashita, and H. Asama. Spatio-temporal bird's-eye view images using multiple fish-eye cameras. *Proceedings of the 2013 IEEE/SICE International Symposium on System Integration*, pages 753–758, 2013.

[30] D. Scaramuzza. *Omnidirectional vision: from calibration to robot motion estimation*. PhD thesis, ETH Zürich, 2007.

[31] T. Scharwächter, M. Enzweiler, U. Franke, and S. Roth. Efficient Multi-Cue Scene Segmentation. *German Conference on Pattern Recognition (GCPR)*, 2013.

[32] T. Scharwächter, M. Enzweiler, S. Roth, and U. Franke. Stixmantics: a medium-level model for real-time semantic scene understanding. In *Computer Vision ECCV 2014*, pages 533–548. Springer International Publishing, 2014.

[33] T. Scharwachter, M. Schuler, and U. Franke. Visual guard rail detection for advanced highway assistance systems. *IEEE Intelligent Vehicles Symposium, Proceedings*, (Iv):900–905, 2014.

[34] H.-Y. Shum and S. B. Kang. A review of image-based rendering techniques. *Proc. SPIE Visual Communications and Image Processing*, pages 2–13, 2000.

[35] S. Sugimoto, K. Motooka, and M. Okutomi. Direct generation of regular-grid ground surface map from in-vehicle stereo image sequences. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 600–607, 2013.

[36] M. Tanner, P. Pinies, L. M. Paz, and P. Newman. DENSER Cities: A System for Dense Efficient Reconstructions of Cities. *arXiv preprint arXiv:1604.03734*, 2016.

[37] B. Thomas, R. Chithambaran, Y. Picard, and C. Cougnard. Development of a cost effective bird's eye view parking assistance system. *2011 IEEE Recent Advances in Intelligent Computational Systems*, pages 461–466, 2011.

[38] R. Y. Tsai. A Versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology Using Off-the-Shelf TV Cameras and Lenses. *IEEE Journal on Robotics and Automation*, 3(4):323–344, 1987.

[39] E. Vincent and R. Laganière. Detecting planar homographies in an image pair. *2nd International Symposium on Image and Signal Processing and Analysis*, 0(2):182–187, 2001.

[40] F. Vogt, S. Krüger, J. Schmidt, D. Paulus, H. Niemann, W. Hohenberger, and C. H. Schick. Light fields for minimal invasive surgery using an endoscope positioning robot. *Methods of information in medicine*, 43(4):403–408, 2004.

[41] Z. Zhang. A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11):1330–1334, 2000.

[42] J. Zienkiewicz, A. Davison, and S. Leutenegger. Real-time height map fusion using differentiable rendering. In *IEEE International Conference on Intelligent Robots and Systems*, volume 2016-Novem, pages 4280–4287, 2016.

[43] J. Zienkiewicz, A. Tsiotsios, A. Davison, and S. Leutenegger. Monocular, real-time surface reconstruction using dynamic level of detail. In *Proceedings - 2016 4th International Conference on 3D Vision, 3DV 2016*, pages 37–46, 2016.

[44] S. Zinger, L. Do, and P. H. N. De With. Free-viewpoint depth image based rendering. *Journal of Visual Communication and Image Representation*, 21(5-6):533–541, 2010.

Author Biography

Christian Fuchs received a Diploma degree in Computer Science from the University of Koblenz-Landau in 2011. He works as a research associate in the Active Vision Group. His primary research interests are 3D pose estimation, stereo vision and driver assistance systems.

Dietrich Paulus obtained a Bachelor degree in Computer Science from University of Western Ontario, London, Canada, followed by a diploma (Dipl.-Inf.) in Computer Science and a PhD (Dr.-Ing.) from Friedrich-Alexander University Erlangen-Nuremberg, Germany. He obtained his habilitation in Erlangen in 2001. Since 2001 he is at the Institute for Computational Visualistics at the University Koblenz Landau, Germany where he became a full professor in 2002. His primary interests are computer vision and robot vision.

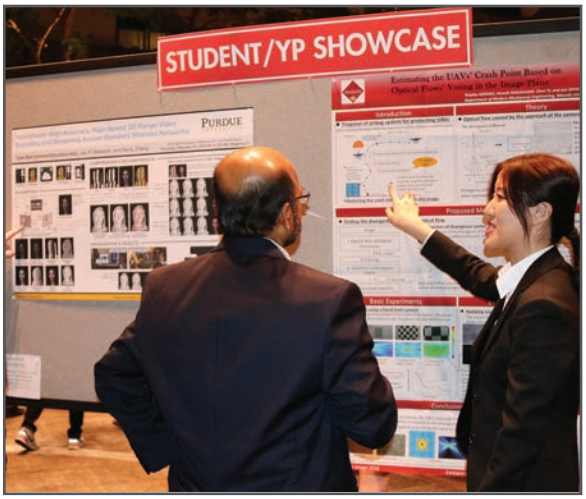
JOIN US AT THE NEXT EI!

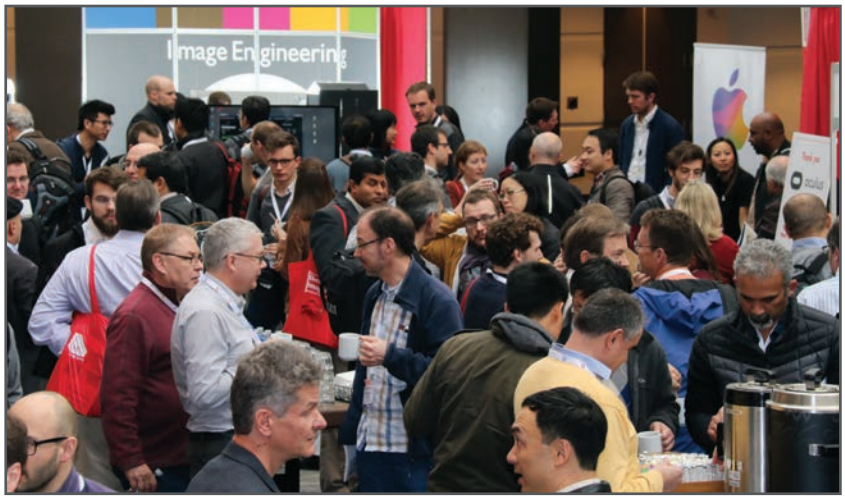
IS&T International Symposium on

Electronic Imaging

SCIENCE AND TECHNOLOGY

Imaging across applications . . . Where industry and academia meet!





• SHORT COURSES • EXHIBITS • DEMONSTRATION SESSION • PLENARY TALKS •
• INTERACTIVE PAPER SESSION • SPECIAL EVENTS • TECHNICAL SESSIONS •

www.electronicimaging.org

IS&T
imaging.org