UAV-based Autonomous Image Acquisition with Multi-View Stereo Quality Assurance by Confidence Prediction

Christian Mostegel Markus Rumpler Friedrich Fraundorfer Horst Bischof
Institute for Computer Graphics and Vision, Graz University of Technology*
{surname}@icg.tugraz.at

Abstract

In this paper we present an autonomous system for acquiring close-range high-resolution images that maximize the quality of a later-on 3D reconstruction with respect to coverage, ground resolution and 3D uncertainty. In contrast to previous work, our system uses the already acquired images to predict the confidence in the output of a dense multi-view stereo approach without executing it. This confidence encodes the likelihood of a successful reconstruction with respect to the observed scene and potential camera constellations. Our prediction module runs in real-time and can be trained without any externally recorded ground truth. We use the confidence prediction for on-site quality assurance and for planning further views that are tailored for a specific multi-view stereo approach with respect to the given scene. We demonstrate the capabilities of our approach with an autonomous Unmanned Aerial Vehicle (UAV) in a challenging outdoor scenario.

1. Introduction

In this paper, we address the problem of UAV-based image acquisition for dense monocular 3D reconstruction with high-resolution images at close range. The aim is to acquire images in such a way that they are suited for processing with an offline dense multi-view stereo (MVS) algorithm, while at the same time fulfilling a set of quality requirements. These requirements include coverage, ground resolution and 3D accuracy and can be assessed geometrically. However, determining how well the images are suited for a specific MVS algorithm is much harder to model. To extract depth from 2D images, MVS approaches have to establish correspondences between the images. To solve this challenging task every MVS approach has to make some assumptions. These assumptions vary from approach to approach, but the most popular assumptions include saliency, local planarity and a static environment. If some of these assumptions are violated the MVS algorithm will not be able to reconstruct the scene correctly. Up to now, this problem was widely ignored by monocular image acquisition approaches [8, 22, 41, 31, 2, 27, 35], which often leads to missing parts in the resulting 3D reconstructions [41, 22]. In this work, we propose a solution for this problem via machine learning. The main idea is to predict how well the acquired images are suited for the dense MVS algorithm directly during the acquisition. While this is already useful for quality assurance, we take this idea one step further and use the acquired images to plan the optimal camera constellation with respect to the observed scene structure. Within this context, we demonstrate that the likelihood of a successful 3D reconstruction depends on the combination of scene structure, triangulation angle and the used MVS algorithm. We further refer to the prediction of this likelihood as *MVS confidence prediction*.

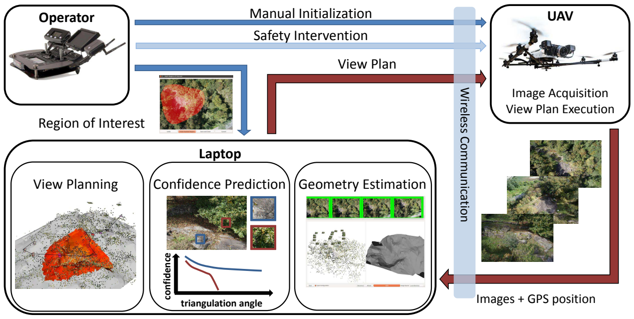

Figure 1. Autonomous Image Acquisition. After a manual initialization, our system loops between view planning and autonomous execution. Within the view planning procedure, we leverage machine learning to predict the best camera constellation for the presented scene and a specific dense MVS algorithm. This MVS algorithm will use the recorded high resolution images to produce a highly accurate and complete 3D reconstruction off-site in the lab.

This MVS confidence prediction is related but not equal to the (two-view) stereo confidence prediction, which is a topic of increasing interest in the domain of stereo vision [15, 46, 33, 29]. In stereo vision, the confidence encodes the likelihood that an already computed depth value is correct, whereas in our case it encodes the likelihood

*The research leading to these results has received funding from the EC FP7 project 3D-PITOTI (ICT-2011-600545) and from the Austrian Research Promotion Agency (FFG) as Vision+ project 836630 and together with OMICRON electronics GmbH as Bridge1 project 843450.

that we will be able to compute a correct 3D measurement later-on. Despite this difference, the training of both tasks is closely related and requires a large amount of data. Up to now, obtaining this training data was a tedious and time consuming task, evolving manual interaction [24, 14, 28], synthetic data [6, 34, 28] and/or 3D ground truth acquisition with active depth sensors [47, 14, 28, 40]. In [29], we present a new way of obtaining this training data for stereo vision. The main idea is to use multiple depthmaps (computed with the same algorithm) from different view points and evaluate consistencies and contradictions between them to collect training data. In this work, we extend this completely automatic approach to multi-view stereo.

After training, our system operates completely on-site (Fig. 1). For estimating the scene geometry, we use the already acquired images for performing incremental structure-from-motion (SfM) [21] and incremental updates of an evolving mesh [20]. Both modules run concurrently in real-time and deliver the camera poses of the acquired images and a closed surface mesh representation of the scene.

Based on this information, we plan future camera positions that maximize the quality of a later-on dense 3D reconstruction. This task falls in the domain of view planning, which has been shown to be NP-hard [48]. Consequently, a wide range of very task specific problem simplifications and solutions were developed in the communities of robotics [39, 36, 32, 18, 13, 12, 10, 7, 5, 4, 30, 50, 44, 19, 9, 41], photogrammetry [27, 26, 3, 1] and computer vision [35, 43, 49, 22, 8, 31, 16]. What kind of simplification is chosen strongly depends on the used sensor, the application scenario and the time constraints. In this work, we propose a set of simplifications that allows us to compute a view plan in a fixed time-frame. In contrast to active depth sensors, a single 3D measurement in monocular 3D reconstruction requires multiple images to observe the same physical scene part. Thus our first simplification is to remove this inter-camera-dependency by planning triplets of cameras as independent measurement units. Second, we introduce the concept of surrogate cameras (cameras without orientation) to reduce the dimensionality of the search space. Finally, we lower the visibility estimation time through inverse scene rendering. In contrast to the works above, our formulation allows us to evaluate a large number of potential camera poses at low cost, while the runtime can be adjusted to the acquisition requirements.

In the following, we first describe the training and setup of our MVS confidence predictor. Then we describe our fixed-time view planning strategy. In our experiments, we evaluate the performance and stored information of the confidence predictor on a challenging outdoor UAV dataset. In the same domain, we finally evaluate our autonomous image acquisition system with respect to quality and completeness of the resulting 3D reconstructions.

2. Multi-View Stereo Confidence Prediction

Given a specific scene structure (e.g. vegetation) and a camera constellation, the MVS confidence encodes the likelihood that a dense reconstruction algorithm will work as intended. With ”work as intended” we mean that if a scene part is observed by a sufficient number of cameras then the algorithm should be able to produce a 3D measurement within the theoretical uncertainty bounds for each pixel that observes this scene part [29]. The first matter we address in this section is how we can generate training data to predict the MVS confidence without any hard ground truth. Therefore we extend our approach for stereo vision [29] to multi-view stereo. Then we outline our machine learning setup and explain how we can use this setup to predict the MVS confidence in real-time during the image acquisition.

2.1. MVS Training Data Generation

As it is extremely tedious to come by 3D ground truth, the basic idea of [29] is to use self-consistency and self-contradiction from different view points for generating labeled training data. This approach is related to depthmap fusion, but outputs 2D label images instead of depthmaps. Pixels that are associated with consistent depth values become positive training data, while inconsistent depth values lead to negative training data. This data is then used for training a pixel-wise binary classification task. The main challenge during the training data generation is to keep the false positive rate (consistent but incorrect) and the false negative rate (correct but inconsistent) as low as possible, while labeling as many pixels as possible.

In [29], we start by computing a depthmap for each stereo pair in the dataset. A single depthmap can be interpreted as the 3D reconstruction of a camera cluster with two cameras and a fixed baseline. In the case of multi-view stereo, we can choose an arbitrary number of cameras per cluster in any constellation. As this general case has too many degrees of freedom to be estimated efficiently, we limit ourselves to three cameras per cluster, which is also the standard minimum number of cameras for most MVS approaches (e.g. [11, 37]). Within this triplet of cameras, the most important factor is the baseline between the cameras or more precisely the triangulation angle between the cameras and the scene. This triangulation angle can be freely chosen. We want to use this property to learn the relationship between MVS confidence and the triangulation angle so that we can choose the right camera constellation for the presented scene in our view planning approach. In theory, a large triangulation angle between cameras is beneficial as it reduces the 3D uncertainty. However, in practice a large triangulation makes it more difficult to find correspondences between the images, especially when the 3D structure is highly complex. To learn this relationship, we first generate a large variety of triangulation angles in the

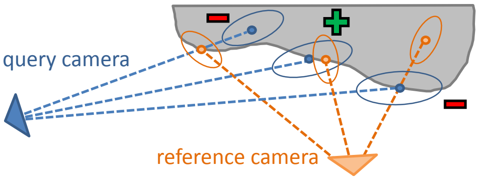

Figure 2. Consistency voting. A positive vote (center) is only cast if the reference measurement is within the uncertainty boundary of the query measurement. A negative vote is either cast if a reference measurement blocks the line of sight of the query camera (left) or the other way around (right).

training data. We randomly sample image triplets from a fixed number (t) of triangulation angle bins, while ensuring that the images have sufficient overlap. For each of these camera triplets, we execute the chosen dense MVS algorithm and project the resulting 3D reconstruction back into the images to obtain one depthmap per image. Using these depthmaps, we can proceed with the training data generation in three stages [29].

The first stage has the purpose of reducing the influence of all consistent but incorrect measurements. In practice, we can observe that the likelihood that two measurements of independent 3D reconstructions[1] are consistent but incorrect at the same time decreases as the relative view point difference increases. Thus, we analyze how well each measurement is supported by reference reconstructions from a sufficiently different view point. We treat a reference measurement as sufficiently different if the view angle difference $\alpha_{\text{diff}} > \alpha_{\text{min}}$ or the scale difference $s_{\text{res,query}} > s_{\text{min}}$ is sufficiently large. We compute these values as $\alpha_{\text{diff}} = \measuredangle(\overrightarrow{\mathbf{p}_{\text{query}}\mathbf{c}_{\text{ref}}}, \overrightarrow{\mathbf{p}_{\text{query}}\mathbf{c}_{\text{query}}})$ and $s_{\text{res,query}} = \text{res}_{\text{ref}}/\text{res}_{\text{query}}$ with $\text{res}_{\text{x}} = f_{\text{x}}/\|\mathbf{c}_{\text{x}} - \mathbf{p}_{\text{query}}\|$, where $\mathbf{c}_{\text{x}}$ is the mean camera center and f_{x} the mean focal length of a camera triplet. If a reference measurement fulfills one of the two conditions, we increment the *support* of the query measurement by one. Note that as in [29], reference measurements from a similar view point are only allowed to increment the *support* once.

In the second stage, we let the parts of the depthmaps with at least one *support* vote on the consistency of all depthmap values. The voting process proceeds analog to [29]. For each query measurement, we collect positive and negative votes as shown in Fig. 2. The votes are weighted with their *support* and their inverse 3D uncertainty [29]. Based on the voting outcome, all pixels with at least one vote are then either assigned a positive or a negative label.

The third stage requires more changes to generalize to multi-view stereo. While in [29] this stage only has the purpose of detecting outliers, in our case we also have to detect missing measurements. More precisely, we have to detect if the MVS algorithm failed to produce any output in a region where it should have been geometrically possible and use this case as a negative training sample. For detecting these missing parts we use a combination of a depthmap augmentation [29] and two surface meshes. We use two meshes with slightly different object boundaries to account for errors in the meshes. To construct these meshes, we first use all available images in the dataset to compute a sparse point cloud [38]. From this point cloud we robustly extract a surface mesh [23, 51], and then shrink and expand this mesh for our purpose. The shrunken mesh is obtained by performing three iterations of neighbor-based smoothing. In each iteration a vertex moves half the distance to the average position of the vertices that share an edge with this vertex. For the second mesh, we expand the shrunken mesh again. For this purpose, we compute a vector by averaging the motion vectors of a vertex and its neighbors from the shrinking procedure. Each vertex is then moved twice the vector length in the opposite direction of this vector. If the depthmap augmentations and the two meshes agree that some part of the scene is missing, the corresponding pixels are used as negative training samples.

2.2. Machine Learning Setup

For view planning, we want to know which camera constellation will give us a good chance of getting a complete and accurate 3D reconstruction. To help with this task, we want to use the already acquired images during the acquisition. For training, we pose the problem as a pixel-wise classification task. During run-time, we compute the MVS confidence depending on the triangulation angle and the scene around the pixel of interest. For this task, we chose Semantic Texton Forests (STFs) [45]. We selected this approach for three main reasons. First, this approach is very fast in the execution phase as it operates directly on the input image (without any feature extractions or filtering). Second, STFs have shown a reasonable performance in semantic image segmentation. Third, it is possible to store meta information in the leaves of the forest. We use this property to store the triangulation angle under which a sample was obtain (or failed to obtain). This does not influence the learning procedure, but allows us to predict the reconstruction confidence in dependence of the triangulation angle at evaluation time.

During the image acquisition, we want to compute the MVS confidence in real-time on a specific computer for a specific high-resolution camera. Thus we provide two ways to reduce the prediction time to the operator's needs. First, we restructure the STF leaf nodes to contain a fixed number (b) of angular bins with one confidence value for each bin. Second, we can make use of the property that the confidence prediction is in general a smooth function for a specific type of object (see Sec. 4.1.2). Thus, we evaluate the MVS confidence on a regular grid and compute a confidence image with b channels for each input image.

[1] 3D reconstructions that were produced with the same MVS algorithm from independent image sets.

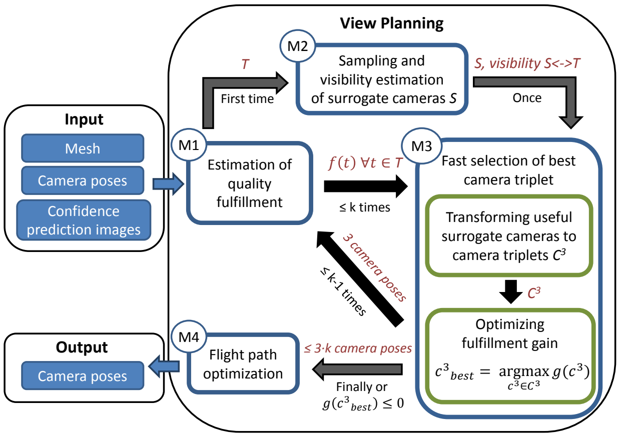
View Planning
Input
Mesh
Camera poses
Confidence prediction images
M1
Estimation of quality fulfillment
T
First time
M2
Sampling and visibility estimation of surrogate cameras S
S, visibility S<->T
Once
$f(t)\ \forall t \in T$
≤ k times
M3
Fast selection of best camera triplet
Transforming useful surrogate cameras to camera triplets C^3
C^3
Optimizing fulfillment gain
$c^3{}_{best} = \underset{c^3 \in C^3}{\operatorname{argmax}}\, g(c^3)$
3 camera poses
≤ k-1 times
≤ 3·k camera poses
Finally or $g(c^3{}_{best}) \leq 0$
M4
Flight path optimization
Output
Camera poses

Figure 3. View planning. Our algorithm tries to find the k next best camera triplets for improving the acquisition quality. Next to the arrows, we show the data communication between our submodules (M1-M4) in red and in black we show how often this data is computed. S is the set of surrogate cameras, T the set of considered unfulfilled triangles and C^3 the set of camera triplets generated from the surrogate cameras.

3. View Planning

The aim of our view planning approach is to plan a set of useful camera poses in a fixed time frame. As the view planning problem is NP-hard, we have to make several simplifications to constrain the computation time. One of our most prominent simplifications is that we plan equilateral camera triplets instead of single cameras. On the one hand, this lets us directly integrate our MVS confidence prediction and, on the other hand, we can treat each camera triplet as an independent measurement unit. In Fig. 3 we show an overview of our approach, which we use to guide the reader through our algorithm and its submodules. As input our approach requires a snapshot of the estimated geometry (mesh and camera poses), as well as the pre-computed MVS confidence images. Further, the operator has to label a region of interest in one of the images (Fig. 6), and define the desired quality constraints (ground resolution and 3D accuracy).

Estimation of quality fulfillment (M1). The aim of this submodule is to estimate how well our desired quality constraints are currently fulfilled by the already captured images. For this estimation, we need the already acquired images and their camera poses C_t as well as the surface mesh. First, we bring all mesh triangles within the region of interest to approximately the same size through iteratively splitting them until the maximum edge length equals the average edge length before splitting. Within the region of interest we then randomly select a fixed number N_t of triangles. Next we determine the visibility information between these triangles and C_t through rendering the mesh. Based on the information which cameras see which triangles, we evaluate how well the desired quality constraints are currently fulfilled. We compute the fulfillment separately for each triangle using four fulfillment functions.

(1) The *coverage* is modeled as a Boolean with $f_{cov} = 1$ if a triangle is visible in a minimum of c cameras and $f_{cov} = 0$ otherwise. (2) The *resolution* requirement (px/m^2) is defined as $f_{res} = \frac{r}{r_d}$ (truncated above 1) for a desired resolution r_d. (3) The fulfillment of the *3D uncertainty* requirement is defined as $f_{unc} = \frac{a_d}{\sqrt{u}}$ (truncated above 1) for a desired accuracy a_d. Here, u stands for the maximum Eigen value of the covariance matrix related to a triangle's centroid [17]. (4) The last fulfillment function is the output of our MVS confidence prediction algorithm f_{conf} (Sec. 2).

For evaluating these functions, we generate all possible combinations of camera triplets from the cameras that observe a triangle t ($c^3 \in C^3_t$). We then evaluate the combined fulfillment function as:

$$f(t, c^3) = (\alpha f_{res} + (1 - \alpha) f_{unc}) \cdot f_{cov} \cdot f_{conf} \qquad (1)$$

This formulation allows the operator to define the relative weight α between desired ground resolution and 3D accuracy, while the coverage and MVS confidence encode the chances of a successful reconstruction. The overall fulfillment of a triangle t is computed as $f(t) = \max_{c^3 \in C^3_t} f(t, c^3)$ (2).

Based on the fulfillment information, we now further reduce the number of considered triangles to a triangle set T. We guide this reduction such that we end up with triangles that have a low fulfillment but are well distributed over the scene of interest. Thus, we randomly select a fixed number N_v of triangles from a piece-wise constant distribution, where the chance of selecting a triangle t is weighted with $w(t) = 1 - f(t)/f_{conf}(t)$. We remove f_{conf} from the weight to avoid bias towards structures that might not be reconstructible at all.

Surrogate cameras (M2). In this submodule, we use the concept of surrogate cameras to estimate the visibility of mesh triangles from a large number of possible camera positions. Thus, we first randomly sample a fixed number N_p of 3D positions in the free space of the scene. These 3D positions represent the camera centers of surrogate cameras. A surrogate camera has an unlimited field of view and thus also no orientation at this point (later we will transform this surrogate camera into an equilateral camera triplet). The usage of surrogate cameras allows us to reformulate the visibility estimation problem and to estimate which surrogate cameras are visible from a given triangle instead of the other way around. The benefit of this formulation is that we are able to control execution time of the visibility estimation with the number of considered triangles instead of the number of considered camera poses. This enables us to evaluate a high number of camera positions at low cost. For each triangle $t \in T$, we place a virtual camera in the scene. The camera center of a virtual camera is set to the triangle's centroid and the optical axis to the triangle's normal. We set the focal length of this camera such that we get a fixed field of view ϕ. Now we use the virtual cameras for rendering the scene, i.e. the mesh and the 3D points that define the centers of the surrogate cameras. The resulting visibility links are stored in the surrogate cameras.

Finding the best camera triplet (M3). To find the best camera triplet at a low computational cost, we guide the transformation from surrogate cameras to camera triplets such that we only need to evaluate potentially useful and feasible camera constellations. Thus, we first compute the potential fulfillment gain $g_{pot}(t)$ of a surrogate camera with respect to a linked triangle t. Formally, we define $g_{pot}(t) = max_\alpha\{f(t, c^3_\alpha) - f(t), 0\}$, for a hypothetical equilateral camera triplet c^3_α, that has the surrogate camera in its center and where each camera directly faces towards the triangle. The triangulation angle α defines the distance between the cameras in the b steps of the predicted MVS confidence, which we evaluate with the confidence image of the closest already captured image (with respect to the surrogate camera) that observes the triangle.

Using this potential gain information, we determine in which direction the surrogate cameras should face. Therefore, we perform a weighted mean shift clustering on the rays towards the linked triangles. As a weight we use the fulfillment gain and the bandwidth is set to the minimum camera opening angle. The winning cluster (i.e. the cluster with the highest potential fulfillment gain) is chosen to define the general viewing direction of the surrogate camera. Then we update the visibility information and the potential gains of the now oriented surrogate cameras.

Given the orientation, we generate b camera triplets for each surrogate camera, one for each confidence bin. For each camera triplet c^3 we efficiently check the distance to obstacles [25] and compute the fulfillment gain of c^3 as

$$g(c^3) = \sum_{t \in T_{c^3}} max\{f(t, c^3) - f(t), 0\}, \tag{3}$$

where T_{c^3} are the triangles that are visible from c^3. Over all triplets, we find the best camera triplet as

$$c^3_{\text{best}} = \arg \max_{c^3 \in C^3} g(c^3), \tag{4}$$

where C^3 is the set of all generated camera triplets[2]. If $g(c^3_{\text{best}})$ is greater than zero and we have not yet planned k camera triplets, we add c^3_{best} to the set of already acquired images (C_t) and plan a new camera triplet. Otherwise, we pass all planned camera triplets with positive gain on to the flight path optimization.

Flight path optimization (M4). This module minimizes the travel distance between the camera poses and ensures that the resulting images can be registered by the geometry estimation module. First, we reorder the camera poses with a greedy distance minimization using the last captured image as a starting point. Then we check if the taken images can be connected to the given set of images respecting the capture sequence. We assume that this is the case if an image has a minimum overlap o_{min} with at least one of the previously captured images. If this is not the case we sample camera poses which fulfill this property along the trajectory from the closest previously captured camera pose to the target camera pose. This results in a view plan that ensures a successful sequential registration of the planned image set.

4. Experiments

We split our evaluation in two main parts. The first part evaluates the performance and the information which is stored by our confidence prediction approach. The second part focuses on our autonomous acquisition system and how it performs in a real world experiment.

4.1. Confidence Prediction

In the first part of this section we benchmark the performance of our training data generation and the prediction performance of the Semantic Texton Forest (STF) [45] on the KITTI dataset [14]. In the second part, we use a challenging multi-view dataset to evaluate what the system can learn about two different multi-view stereo approaches in relation to scene structure and camera constellation.

In all our experiments, we used the same STF setup. We implemented the STF in the random forest framework of Schulter et al. [42]. We only use STF in its basic form (without image-level prior [45]). This means that the split decision is made directly on the image data (Lab color space) within a patch of the size 27 × 27. We trained 20 trees with a maximum depth of 20. For the split evaluation, we used the Shannon Entropy, minimum leaf size for further splitting of 50, 5000 node tests, 100 thresholds and 1000 random training samples at each node. For all our experiments, we extracted approximately 4 million training patches for each class in training.

4.1.1 KITTI2012 Dataset

In this experiment, we apply our approach to the scenario of street-view dense stereo reconstruction using the KITTI dataset [14], which provides a semi-dense depth ground truth recorded with a Lidar.

For learning, we follow the same procedure as in [29] and use the 195 sequences of 21 stereo pairs of the testing dataset for automatically generating our label images.

[2] For the implementation, we can drastically reduce the number of evaluations by using the potential gain. If we start with the surrogate camera with the highest potential gain, we can stop if $\sum_t g_{pot}(t)$ of the evaluated surrogate camera is zero or smaller than the current best gain.

We treat each stereo pair as a distinct cluster and use a semi-global matcher with left-right consistency check (SURE [37]) as the query algorithm. As in [29], we evaluate the label accuracy and the average Area Under the Sparsification Curve (AUSC), although with a slightly different setup. While stereo confidence prediction [29] tries to decide which depth values cannot be trusted from an already computed depthmap, our aim is to predict which kind of structures cause more problems than others. Thus, we remove all regions from the Lidar ground truth, which are not visible in both color images (including object occlusions).

With this setup we reach a labeling accuracy of **98.7%** while labeling 35% of the ground truth pixels (which is very similar to the results in [29]). For the sparsification we obtain a relative AUSC of 3.15 (obtained AUSC divided by optimal AUSC). This means that the AUSC is **39%** lower than random sparsification with 5.15. This is a strong indication that the system learned to predict regions which are difficult to reconstruct for the semi-global matcher.

For the matter of completeness, we also analyze the sparsification performance of the STF [45] with the exact same setup as in [29] (including the training data generation). With this setup STF reaches a relative AUSC of 6.63. It is not surprising that STF cannot reach the sparsification performance of stereo specific sparsification approaches (e.g. left-right difference with 2.81), as the STF only uses color information of a single image and thus has no chance to reason about occlusions. Nevertheless, the STF was able to extract some high level knowledge in which regions the chances of failure are higher and thus still obtains a 31.4% lower AUSC value than random sparsification (9.65).

4.1.2 Val Camonica Dataset

For the second dataset, we have chosen a reconstruction scenario in a closed real-world domain, where the task is the 3D reconstruction of prehistoric rock art sites in the Italian valley of Val Camonica. The recorded dataset consists of over 5000 images of 8 different sites (see supplement), which contain a well-defined set of 3D structures (mainly rock, grass, trees, bridges, signs and markers). These structures dominate nearly all sites in the region (hundreds), which makes this a perfect example for learning and predicting domain specific properties of a query algorithm.

For generating camera triplets we used $t = 5$ triangulation bins. The lowest triangulation angle bin starts at a minimum angle of 4° and ranges to double that value, where the next bin starts. On each resulting triplet we execute a query algorithm three times at different image resolutions (levels 1, 2 and 3 of an image pyramid). We evaluate two query algorithms for the dense 3D reconstruction. The first query algorithm is based on semi-global matching SURE [37], but can use more than two views for improving the reconstruction accuracy. In contrast, our second query algorithm PMVS [11] tries to densify an initial sparse 3D reconstruction through iterative expansion.

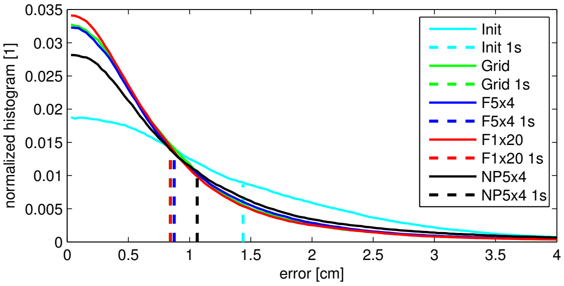

Figure 5. Error histogram on the rock surface. We show the normalized histograms of the error distribution and the 1 σ bound in which 68.3% of all measurements lie. Grid and F1x20 share the same error bound.

For the quantitative evaluation of this experiment, we performed leave-one-out cross validation across the 8 sites, i.e. we train on 7 sites and test on the remaining. This led to the following classification accuracies: PMVS: 81.1% (STD: 4.2%) and SURE: 65.3% (STD: 6.1%). Within this context, we also analyzed the influence of regular grid sampling on the prediction performance. For small grid sizes the classification error stays nearly the same (relative error increase is below 1% for 4 pixels), while for larger grid sizes it declines gradually (below 3% for 16 pixels and below 7% for 64 pixels). This means that regular sampling can drastically reduce the computational load of the prediction with only a small decrease of the prediction performance.

Now let us analyze what the system learned about the two algorithms in relation to scene structures and triangulation angle. In Fig. 4 we show the confidence prediction for six different structures. From this experiment we can draw several conclusions. First, the 3D structure of the scene has a significant influence on how well something can be reconstructed under a given triangulation angle. The more non-planar a structure is, the harder it is to reconstruct at large triangulation angles. Second, the two analyzed approaches react very differently to a change in triangulation angle. While for SURE the confidence is always highest for very small angles, PMVS' confidence stays constant for smooth surfaces. In the case of non-planarity, SURE is clearly more robust than PMVS.

4.2. Autonomous Image Acquisition

To evaluate our image-acquisition approach in this scenario, we first run different view planning algorithms on-site and then analyze the effective reconstruction output, which is computed off-site. As we also desire a reconstruction of the surrounding environment (which is dominated by vegetation), we use SURE [37] as an MVS algorithm. For this experiment, we run three versions of the proposed approach. The first version is our full approach (F5x4), where

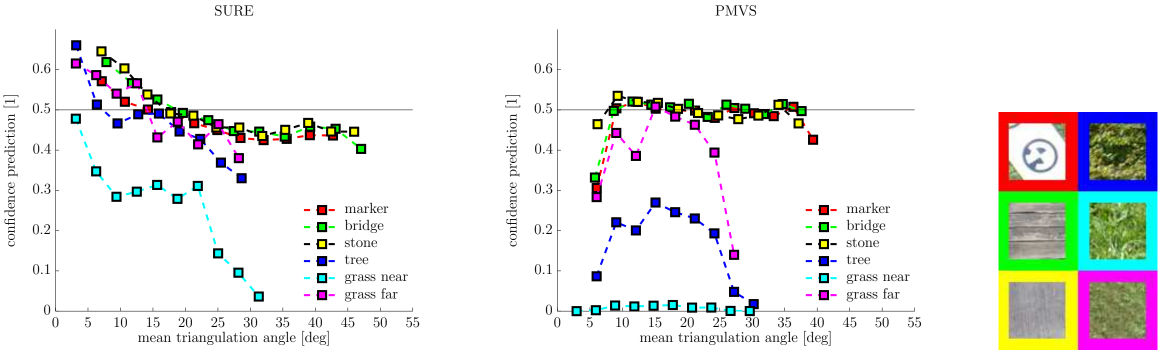

Figure 4. Dependence of the confidence prediction on the triangulation angle and the 3D structure. On the right, we display the patches (50×50px) which we used to produce the curves. These curves show the confidence prediction within angular bins (20 bins between min and max). The curves stop if less than 1% of the collected triangulation angles fall within a bin. For both approaches (SURE and PMVS), there is a significant difference between smooth surfaces (marker, bridge, stone) and high frequency structures (tree, grass). The predicted confidence is to some extent correlated with the degree of non-planarity of a structure. While grass viewed from far away is quite easy to reconstruct, the same grass viewed close up becomes very hard to reconstruct. For both approaches, the chance for reconstructing highly non-planar structures above 30° is virtually zero.

	Init	Grid	F5x4	F1x20	NP5x4	G+F5x4	G+F1x20	G+NP5x4
cov	53.5 ± 1.2	56.0 ± 1.2	$\mathbf{65.6} \pm 1.6$	$\mathbf{66.6} \pm 1.4$	56.7 ± 1.4	$\mathbf{69.5} \pm 1.5$	67.0 ± 1.5	57.2 ± 1.2
f_{res}	17.9 ± 1.3	43.9 ± 2.6	42.2 ± 2.6	$\mathbf{47.5} \pm 2.7$	29.3 ± 2.3	$\mathbf{52.8} \pm 2.7$	$\mathbf{55.3} \pm 2.7$	46.8 ± 2.6
f_{unc}	15.5 ± 0.3	$\mathbf{22.8} \pm 0.4$	21.2 ± 0.5	20.7 ± 0.5	19.9 ± 0.5	$\mathbf{27.7} \pm 0.5$	26.2 ± 0.4	25.6 ± 0.4
f	16.7 ± 0.8	$\mathbf{33.4} \pm 1.5$	31.7 ± 1.5	$\mathbf{34.1} \pm 1.6$	24.6 ± 1.4	$\mathbf{40.2} \pm 1.6$	$\mathbf{40.7} \pm 1.6$	36.2 ± 1.5

Table 1. Fulfillment statistics in percent. We show the coverage of the region of interest cov, the resolution fulfillment f_{res} and the uncertainty fulfillment f_{unc}, as well as the overall fulfillment f as defined in Sec. 2. We display the mean value and the standard deviation over the three surface meshes. We mark all results within the standard deviation of the best method with a bold fond. In the first column we show the results with only the 19 initialization images, then we show the four standalone approaches. The last three columns show a combination of the standard grid approach (Grid) with the other approaches.

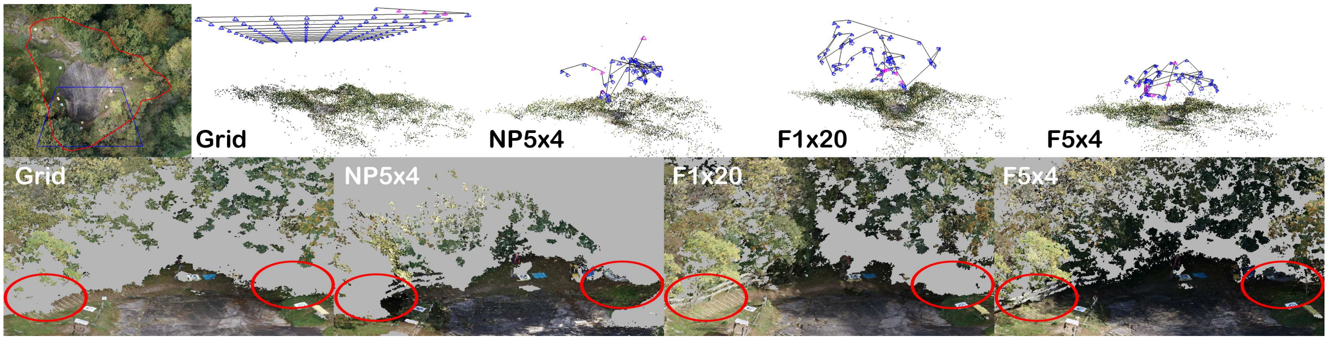

Figure 6. Resulting 3D reconstructions. On the top left, we show one of the images acquired by the UAV with the region of interest in red. The other four columns show all view plans. The blue cameras are regular or triplet cameras, while the pink cameras ensure sufficient overlap for sequential registration. In the bottom row we show all four reconstructions (the field of view is marked blue in the top left image). Note that our approach has the best coverage underneath the trees. The point colors vary due to changing illumination conditions.

we let the algorithm plan 4 camera triplets per iteration for a total of 5 iterations. In the second version (F1x20), we let our approach plan the same number of total triplets (20) but in a single iteration, i.e. we disable the incremental geometry updates. The third version (NP5x4) is exactly the same as F5x4 but without the prediction to constrain the triangulation angle. As a baseline method, we use grid planning with 80 percent overlap. All approaches share the same set of parameters. The fulfillment requirements were set to $c = 3$, $r_d = 8$ mm and $a_d = 8$ mm with $\alpha = 0.5$. The safety distance was set to 5 m at a maximum octree [25] resolution of 2 m and the minimum camera overlap for registration to $o_{min} = 50\%$. The triangulation angle was binned in $b = 9$ steps of 5° from 0° to $\gamma_{max} = 45°$. For the inverse visibility estimation we set the parameters such that the planning approximately takes 5 seconds per planned triplet, i.e. $N_t = 2000$, $N_p = 5000$ and $N_v = 200$ with $\phi = 120°$. This parameters resulted in an effective execution time per triplet of 5.98 seconds (STD: 2.19) over all experiments on a HP EliteBook 8570w. The confidence was evaluate on a regular grid with a step size of 8 pixels, which resulted in a confidence prediction time of ~2 sec/image. We acquire the images with a Sony Nex-5 16Mpx camera mounted on an Asctec Falcon8 octocopter.

For this experiment, we focus on one site in Val Camonica, namely Seradina Rock 12C. The rock surface (17×13 m) is covered with prehistoric rock carvings and is partly occluded by the surrounding vegetation (Fig. 6). We placed 7 fiducial markers in circle around the rock of interest and measured them with a Leica total station. These markers can be automatically detected in the images and are used for geo-referencing the offline reconstructions [38]. Additionally, a ground truth mesh of the rock (not the surroundings) was obtained through terrestrial laser scanning (TLS) in the same coordinate system two years before. The mesh has a resolution of 8 mm edge length and the accuracy of the laser scanner (Riegl VZ-400) is 5 mm. We use this mesh to evaluate the resulting 3D uncertainty.

To evaluate the coverage and the requirement fulfillment, we first obtain a geo-referenced sparse reconstruction from all flights on the day of the experiment (∼500 images). Then we obtain three meshes, one based on [23, 51] and the two others as described in Sec. 2. As we know that these meshes will contain errors, we only use these meshes as a guideline for the evaluation. Within the region of interest, we split all triangles to have a maximum edge length of 8 cm. For each taken image, we first compute the triangle visibility. Then we produce a depthmap from all SURE 3D points linked to the image. If the measured depth is either larger than or within 24 cm of the triangle depth, we accept the 3D point as a valid measurement of the triangle. Based on the links of the 3D measurement, we then compute the fulfillment of the triangle analog to Sec. 2. Finally, this results in a set of fulfillment and coverage scores over all triangles in the region of interest.

In field, all approaches were initialized with 19 images taken in grid at a height of 50 m above the lowest point of the site. The region of interest was marked in one of the initialization images, such that it is centered on the rock and includes a few meters of the surrounding vegetation (Fig. 6). Landing and take-off are performed manually, while the view plans are executed autonomously by the UAV.

Results. For each of our approach variants, we executed SURE only on the three images of the triplets. Like this we can evaluate the general success rate of view planning variants in analyzing on which triplets SURE succeeded to produce any 3D output. Without the confidence prediction the success rate is very low (**18%** for **NP5x4**). This shows the gap between theory and practice. While in theory a large triangulation leads to a small 3D uncertainty, the matching becomes much more difficult and only flat surfaces survive. However, with the proposed confidence prediction we were able to reach a prefect success rate for our full approach (**100%** for **F5x4**), and still reached an acceptable success rate without the reconstruction updates (**80%** for **F1x20**).

In Table 1 we display the effective fulfillment statistics of all approaches in the region of interest. Of the standalone approaches, F1x20 and Grid take the lead, but are closely followed by F5x4. The worst performance was reached by NP5x4. While the dense grid performs well on the overall fulfillment, we can see a **10% gap** in the scene coverage, where F5x4 and F1x20 lead with nearly equal results. F1x20 performs slightly better than F5x4, because F1x20 found a sweet spot in the center above the rock for a single triplet where it was able to drop below the tree line and acquire a close up of the rock.

If we combine the results of the dense grid (Grid) with the proposed approach, we achieve the overall best results. All evaluated measures improve significantly, which is an indication of a symbiosis between the approaches. This suggests that for the given scene (which is quite flat for many scene parts) an initial grid reconstruction with a subsequent refinement with the proposed approach is recommended. Note that if the scene complexity increases and a grid plan can no longer be executed safely (e.g. underneath a forest canopy or indoors), our planning approach is still applicable.

If we take a look at the error distribution in relation to the ground truth of the rock surface (Fig. 5), we can see that our approach and grid planning achieve very similar results. Note the Grid only covered 87.4% of the rock surface, while all others covered significantly more: F5x4 covered 97.9%, F1x20 94.7% and NP5x4 94.0%. This is a very promising result, as we only allowed our approach to use the planned triplets and no combination between them, while we put no such restrictions on the Grid approach. Furthermore, many of the triplets focused on the surrounding vegetation and the overall number of acquired images by our approach is lower than for the Grid approach (60 vs. 108 images). Thus, our approach achieved a high accuracy at a higher coverage with fewer images, which can also be observed visually in Fig. 6 and the supplementary material.

5. Conclusion

In this paper we presented a novel autonomous system for acquiring close-range high-resolution images that maximize the quality of a later-on 3D reconstruction. We demonstrated that this quality strongly depends on the planarity of the scene structure (complex structures vs. smooth surfaces), the camera constellation and the chosen dense MVS algorithm. We learn these properties from unordered image sets without any hard ground truth and use the acquired knowledge to constrain the set of possible camera constellations in the planning phase. In using these constraints, we can drastically improve the success of the image acquisition, which finally results in a high-accuracy 3D reconstruction with a significantly higher scene coverage compared to traditional acquisition techniques.

References

[1] A. H. A., B. Sargeant, T. Erfani, S. Robson, M. Shortis, M. Hess, and J. Boehm. Towards fully automatic reliable 3d acquisition: From designing imaging network to a complete and accurate point cloud. *Robotics and Autonomous Systems*, 62(8):1197 – 1207, 2014. 2

[2] A. H. Ahmadabadian, S. Robson, J. Boehm, and M. Shortis. Image selection in photogrammetric multi-view stereo methods for metric and complete 3d reconstruction. *Proc. SPIE*, 8791:879107–879107–11, 2013. 1

[3] A. H. Ahmadabadian, S. Robson, J. Boehm, and M. Shortis. Stereo-imaging network design for precise and dense 3D reconstruction. *The Photogrammetric Record*, 29(147):317–336, 2014. 2

[4] K. Alexis, C. Papachristos, R. Siegwart, and A. Tzes. Uniform coverage structural inspection path-planning for micro aerial vehicles. In *Intelligent Control (ISIC), 2015 IEEE International Symposium on*, pages 59–64, Sept 2015. 2

[5] A. Bircher, M. Kamel, K. Alexis, M. Burri, P. Oettershagen, S. Omari, T. Mantel, and R. Siegwart. Three-dimensional coverage path planning via viewpoint resampling and tour optimization for aerial robots. *Autonomous Robots*, pages 1–20, 2015. 2

[6] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black. A naturalistic open source movie for optical flow evaluation. In *European Conference on Computer Vision (ECCV)*, 2012. 2

[7] C. Dornhege, A. Kleiner, and A. Rolling. Coverage search in 3D. In *Safety, Security, and Rescue Robotics (SSRR), 2013 IEEE International Symposium on*, pages 1–8, Oct 2013. 2

[8] E. Dunn and J.-M. Frahm. Next best view planning for active model improvement. In *British Machine Vision Conference (BMVC)*, pages 1–11, 2009. 1, 2

[9] B. Englot and F. Hover. Planning complex inspection tasks using redundant roadmaps. In *Int. Symp. Robotics Research*, 2011. 2

[10] C. Forster, M. Pizzoli, and D. Scaramuzza. Appearance-based active, monocular, dense reconstruction for micro aerial vehicles. In *Proceedings of Robotics: Science and Systems*, Berkeley, USA, July 2014. 2

[11] Y. Furukawa and J. Ponce. Accurate, dense, and robust multiview stereopsis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(8), 2010. 2, 6

[12] E. Galceran, R. Campos, N. Palomeras, D. Ribas, M. Carreras, and P. Ridao. Coverage path planning with real-time replanning and surface reconstruction for inspection of three-dimensional underwater structures using autonomous underwater vehicles. *Journal of Field Robotics*, 32(7):952–983, 2015. 2

[13] E. Galceran and M. Carreras. A survey on coverage path planning for robotics. *Robotics and Autonomous Systems*, 61(12):1258 – 1276, 2013. 2

[14] 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 (CVPR)*, 2012. 2, 5

[15] R. Haeusler, R. Nair, and D. Kondermann. Ensemble learning for confidence measures in stereo vision. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 305–312, June 2013. 1

[16] S. Haner and A. Heyden. Optimal view path planning for visual SLAM. In *Proceedings of the 17th Scandinavian Conference on Image Analysis*, SCIA'11, pages 370–380, Ystad, Sweden, 2011. Springer-Verlag. 2

[17] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2 edition, Apr. 2004. 4

[18] L. Heng, A. Gotovos, A. Krause, and M. Pollefeys. Efficient visual exploration and coverage with a micro aerial vehicle in unknown environments. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 1071–1078. IEEE, 2015. 2

[19] G. A. Hollinger, B. Englot, F. Hover, U. Mitra, and G. S. Sukhatme. Uncertainty-Driven View Planning for Underwater Inspection. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2012. 2

[20] C. Hoppe, M. Klopschitz, M. Donoser, and H. Bischof. Incremental surface extraction from sparse structure-from-motion point clouds. In *British Machine Vision Conference (BMVC)*, pages 94–1, 2013. 2

[21] C. Hoppe, M. Klopschitz, M. Rumpler, A. Wendel, S. Kluckner, H. Bischof, and G. Reitmayr. Online feedback for structure-from-motion image acquisition. In *British Machine Vision Conference (BMVC)*, volume 2, page 6, 2012. 2

[22] C. Hoppe, A. Wendel, S. Zollmann, K. Pirker, A. Irschara, H. Bischof, and S. Kluckner. Photogrammetric camera network design for micro aerial vehicles. In *Proceedings of the Computer Vison Winterworkshop*, Mala Nedelja, Slovenia, 2012. 1, 2

[23] P. Labatut, J.-P. Pons, and R. Keriven. Efficient multi-view reconstruction of large-scale scenes using interest points, delaunay triangulation and graph cuts. In *International Conference on Computer Vision (ICCV)*, pages 1–8, Oct 2007. 3, 8

[24] L. Ladick, P. Sturgess, C. Russell, S. Sengupta, Y. Bastanlar, W. Clocksin, and P. Torr. Joint optimization for object class segmentation and dense stereo reconstruction. *International Journal of Computer Vision*, 100(2):122–133, 2012. 2

[25] B. Lau, C. Sprunk, and W. Burgard. Efficient grid-based spatial representations for robot navigation in dynamic environments. *Robotics and Autonomous Systems*, 61(10):1116 – 1130, 2013. Selected Papers from the 5th European Conference on Mobile Robots (ECMR 2011). 5, 7

[26] M. Liu, M. Salzmann, and X. He. Discrete-continuous depth estimation from a single image. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 716–723, June 2014. 2

[27] R. A. Martin, I. Rojas, K. Franke, and J. D. Hedengren. Evolutionary view planning for optimized uav terrain modeling in a simulated environment. *Remote Sensing*, 8(1):26, 2016. 1, 2

[28] M. Menze and A. Geiger. Object scene flow for autonomous vehicles. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 2

[29] C. Mostegel, M. Rumpler, F. Fraundorfer, and H. Bischof. Using self-contradiction to learn confidence measures in stereo vision. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 1, 2, 3, 5, 6

[30] C. Mostegel, A. Wendel, and H. Bischof. Active Monocular Localization: Towards Autonomous Monocular Exploration for Multirotor MAVs . In *IEEE International Conference on Robotics and Automation (ICRA)*, 2014. 2

[31] C. Munkelt, A. Breitbarth, G. Notni, and J. Denzler. Multi-View Planning for Simultaneous Coverage and Accuracy Optimisation. In *British Machine Vision Conference (BMVC)*, 2010. 1, 2

[32] M. Nieuwenhuisen and S. Behnke. Layered mission and path planning for mav navigation with partial environment knowledge. In *Intelligent Autonomous Systems 13*, pages 307–319. Springer, 2014. 2

[33] M.-G. Park and K.-J. Yoon. Leveraging stereo matching with learning-based confidence measures. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 101–109, June 2015. 1

[34] M. Peris, A. Maki, S. Martull, Y. Ohkawa, and K. Fukui. Towards a simulation driven stereo vision system. In *International Conference on Pattern Recognition (ICPR)*, 2012. 2

[35] M. Pistellato, F. Bergamasco, A. Albarelli, and A. Torsello. Dynamic Optimal Path Selection for 3D Triangulation with Multiple Cameras. In *International Conference on Image Analysis and Processing (ICIAP)*, pages 468–479. Springer, 2015. 1, 2

[36] D. Rainville, J.-P. Mercier, C. Gagné, P. Giguere, D. Laurendeau, et al. Multisensor placement in 3D environments via visibility estimation and derivative-free optimization. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 3327–3334. IEEE, 2015. 2

[37] M. Rothermel, K. Wenzel, D. Fritsch, and N. Haala. SURE: Photogrammetric Surface Reconstruction from Imagery. In *Proceedings LC3D Workshop*, 2012. 2, 6

[38] M. Rumpler, S. Daftry, A. Tscharf, R. Prettenthaler, C. Hoppe, G. Mayer, and H. Bischof. Automated end-to-end workflow for precise and geo-accurate reconstructions using fiducial markers. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2(3):135, 2014. 3, 8

[39] S. A. Sadat, J. Wawerla, and R. Vaughan. Fractal trajectories for online non-uniform aerial coverage. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 2971–2976, May 2015. 2

[40] D. Scharstein, H. Hirschmller, Y. Kitajima, G. Krathwohl, N. Nei, X. Wang, and P. Westling. High-resolution stereo datasets with subpixel-accurate ground truth. In X. Jiang, J. Hornegger, and R. Koch, editors, *Pattern Recognition*, volume 8753 of *Lecture Notes in Computer Science*, pages 31–42. Springer International Publishing, 2014. 2

[41] K. Schmid, H. Hirschmüller, A. Dömel, I. Grixa, M. Suppa, and G. Hirzinger. View Planning for Multi-View Stereo 3D Reconstruction Using an Autonomous Multicopter. *Journal of Intelligent and Robotic Systems*, 65:309–323, 2012. 1, 2

[42] S. Schulter, C. Leistner, P. Wohlhart, P. M. Roth, and H. Bischof. Accurate object detection with joint classification-regression random forests. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014. 5

[43] W. R. Scott, G. Roth, and J.-F. Rivest. Pose error effects on range sensing. In *15th International Conference on Vision Interface*, 2002. 2

[44] W. R. Scott, G. Roth, and J.-F. Rivest. View planning for automated three-dimensional object reconstruction and inspection. *ACM Computing Surveys (CSUR)*, 35:64–96, 2003. 2

[45] J. Shotton, M. Johnson, and R. Cipolla. Semantic texton forests for image categorization and segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, June 2008. 3, 5, 6

[46] A. Spyropoulos, N. Komodakis, and P. Mordohai. Learning to detect ground control points for improving the accuracy of stereo matching. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1621–1628, June 2014. 1

[47] C. Strecha, W. von Hansen, L. Van Gool, P. Fua, and U. Thoennessen. On benchmarking camera calibration and multi-view stereo for high resolution imagery. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, June 2008. 2

[48] G. H. Tarbox and S. N. Gottschlich. Planning for complete sensor coverage in inspection. *Computer Vision and Image Understanding*, 61:84–111, 1995. 2

[49] M. Trummer, C. Munkelt, and J. Denzler. Online Next-Best-View Planning for Accuracy Optimization Using an Extended E-Criterion. In *International Conference on Pattern Recognition (ICPR)*, pages 1642–1645, 2010. 2

[50] J. I. Vasquez-Gomez, L. E. Sucar, and R. Murrieta-Cid. Hierarchical Ray Tracing For Fast Volumetric Next-Best-View Planning. In *International Conference on Computer and Robot Vision*, 2013. 2

[51] H.-H. Vu, P. Labatut, J.-P. Pons, and R. Keriven. High accuracy and visibility-consistent dense multiview stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(5):889–901, May 2012. 3, 8