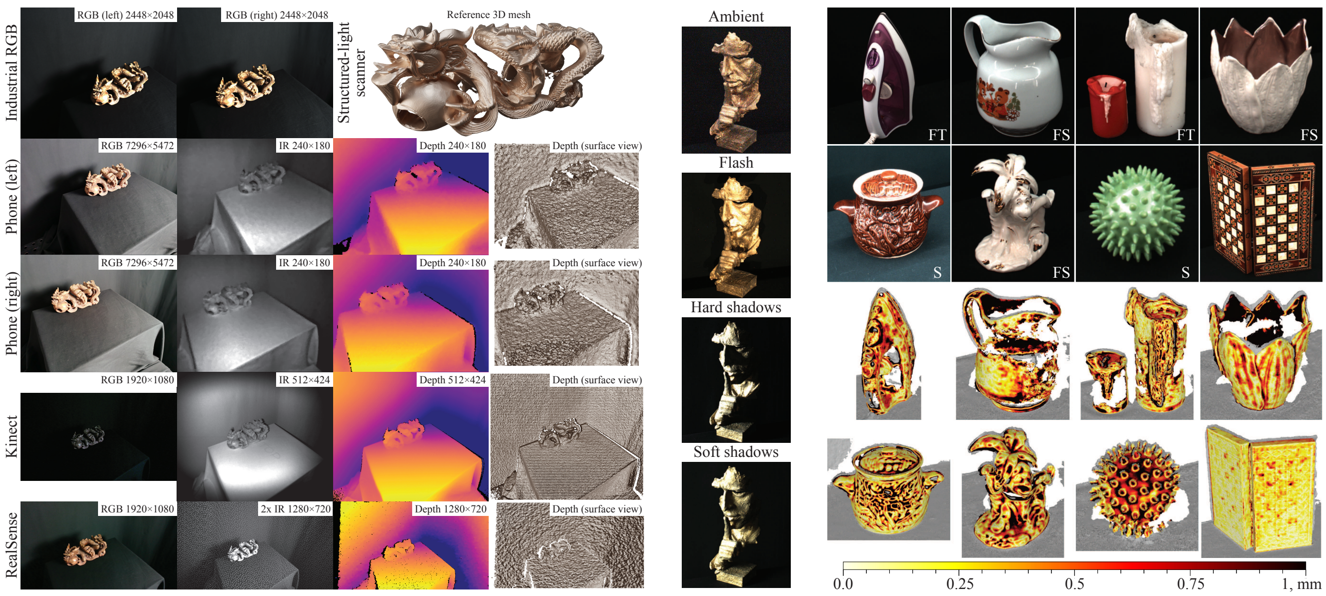
Multi-sensor large-scale dataset for multi-view 3D reconstruction

Oleg Voynov[1,2] Gleb Bobrovskikh[1] Pavel Karpyshev[1] Saveliy Galochkin[1]
Andrei-Timotei Ardelean[1] Arseniy Bozhenko[1] Ekaterina Karmanova[1] Pavel Kopanev[1]
Yaroslav Labutin-Rymsho[3] Ruslan Rakhimov[1] Aleksandr Safin[1] Valerii Serpiva[1]
Alexey Artemov[4*] Evgeny Burnaev[1,2] Dzmitry Tsetserukou[1] Denis Zorin[5]

[1]Skolkovo Institute of Science and Technology [2]Artificial Intelligence Research Institute
[3]Moscow Engineering Physics Institute [4]Technical University of Munich [5]New York University



(a) Data captured with all sensors (b) Lighting examples (c) Objects from dataset and 3D reconstruction error

Figure 1. Our dataset contains RGB-D data captured (a) with 7 different devices, (b) under various lightning conditions. (c) We focus on materials challenging for 3D reconstruction algorithms, such as featureless (F), highly specular with sharp reflections (S), or translucent (T), as illustrated with reconstructions produced by state-of-the-art algorithms (compare with an "easy" object on the bottom right).

Abstract

We present a new multi-sensor dataset for multi-view 3D surface reconstruction. It includes registered RGB and depth data from sensors of different resolutions and modalities: smartphones, Intel RealSense, Microsoft Kinect, industrial cameras, and structured-light scanner. The scenes are selected to emphasize a diverse set of material properties challenging for existing algorithms. We provide around 1.4 million images of 107 different scenes acquired from 100 viewing directions under 14 lighting conditions. We expect our dataset will be useful for evaluation and training of 3D reconstruction algorithms and for related tasks. The dataset is available at `skoltech3d.appliedai.tech`.

[*]Work partially performed while at Skolkovo Institute of Science and Technology.

1. Introduction

Sensor data used in 3D reconstruction range from highly specialized and expensive CT, laser, and structured-light scanners to video from commodity cameras and depth sensors; computational 3D reconstruction methods are typically tailored to a specific type of sensors. Yet, even commodity hardware increasingly provides multi-sensor data: for example, many recent phones have multiple RGB cameras as well as lower resolution depth sensors. Using data from different sensors, RGB-D data in particular, has the potential to considerably improve the quality of 3D reconstruction. For example, multi-view stereo algorithms (*e.g.*, [55, 82]) produce high-quality 3D geometry from RGB data, but may miss featureless surfaces; supplementing RGB images with depth sensor data makes it possible to have more complete reconstructions. Conversely, commodity depth sensors often lack resolution provided by RGB cameras.

Learning-based techniques substantially simplify the challenging task of combining data fom multiple sensors. However, learning methods require suitable data for training. Our dataset aims to complement existing ones (*e.g.*, [13, 28, 31, 58, 63, 83]), as discussed in Section 2, most importantly, by providing multi-sensor data and high-accuracy ground truth for objects with challenging reflection properties.

The structure of our dataset is expected to benefit research on 3D reconstruction in several ways.

- *Multi-sensor data.* We provide data from *seven* different devices with high-quality alignment, including low-resolution depth data from commodity sensors (phones, Kinect, RealSense), high-resolution geometry data from a structured-light scanner, and RGB data at different resolutions and from different cameras. This enables supervised learning for reconstruction methods relying on different combinations of sensor data, in particular, increasingly common combination of high-resolution RGB with low-resolution depth data. In addition, multi-sensor data simplifies comparison of methods relying on different sensor types (RGB, depth, and RGB-D).
- *Lighting and pose variability.* We chose to focus on a setup with controlled (but variable) lighting and a fixed set of camera positions for all scenes, to enable high-quality alignment of data from multiple sensors, and systematic comparisons of algorithm sensitivity to various factors. We aimed to make the dataset large enough (1.39 M images of different modalities in total) to support training machine learning algorithms, and provide systematic variability in camera poses (100 per object), lighting (14 lighting setups, including "hard" and diffuse lighting, flash, and backlighting, as illustrated in Figure 1b), and reflection properties these algorithms need.
- *Object selection.* Among 107 objects in our dataset, we include primarily objects that may present challenges to existing algorithms mentioned above (see examples in Figure 1c); we made special effort to improve quality of 3D high-resolution structured-light data for these objects.

Our focus is on RGB and depth data for individual objects in laboratory setting, similar to the DTU dataset [28], rather than on complex scenes with natural lighting, as in Tanks and Temples [31] or ETH3D [58]. This provides means for systematic exploration and isolation of different factors contributing to strengths and weaknesses of different algorithms, and complements more holistic evaluation and training data provided by datasets with complex scenes.

2. Related work

Many datasets for tasks related to 3D reconstruction were developed (see, for example, a survey of datasets related to simultaneous localization and mapping (SLAM) [38]); we only discuss datasets most closely related to ours. A summary of comparisons to key datasets from previous work is shown in Table 1.

Dataset	Sensor types	RGB, MPix	Depth, MPix	HR geo.	Poses /scene	Lighting	# Scenes	# Frames
DTU [28]	RGB (2) SLS	2		✓	49/64	8	80	27K
ETH3D [58]	RGB TLS	24 —		✓	10–70	U	24	11K
TnT [31]	RGB TLS	8 —		✓	150–300	U	21	148K
BlendedMVG [83]	unknown	3/0.4			20–1000	U	502	110K
BigBIRD [63]	RGB (5) RGB-D (5)	12 1.2	— 0.3		600	1	120	144K
ScanNet [13]	RGB-D	1,3	0,3		NA	U	1513	2.5M
Ours	RGB (2) RGB-D 1 (2) RGB-D 2 RGB-D 3 SLS	5 40 2 2 —	— 0.04 0.2 0.9 —	✓	100	14	107	877K

Table 1. **Comparison of our dataset to the most widely used related datasets.** U indicates uncontrolled lighting; frames are counted per sensor, *i.e.*, all data from an RGB-D sensor are counted as a single frame. The number of separate images acquired may be considerably larger (1.4 M for our dataset). All scenes, from both training and testing sets, were counted.

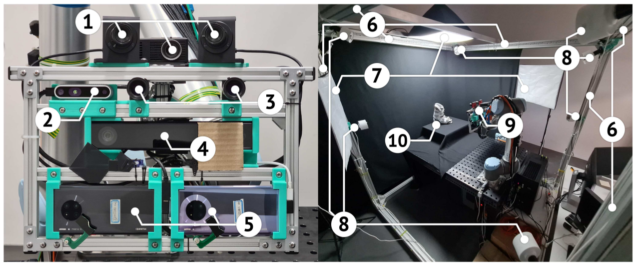
RGB datasets with high-resolution 3D ground truth. A number of datasets are designed for multi-view stereo (MVS) methods, such as PatchMatch-based [4, 55, 80, 81], learning-based [9, 21, 34, 40, 70, 76, 82, 87], or hybrid methods [19, 33]. These datasets are also used for evaluation of methods reconstructing an implicit surface representation encoded by a neural network from a set of RGB images [45, 46, 71, 84], and in the novel view synthesis task [2, 42, 50, 51, 72].

Datasets from this category typically include high-resolution RGB, either photo [1, 28, 58, 83] or video [31], and high-resolution 3D ground truth obtained with a structured-light scanner (SLS) [1] or a terrestrial laser scanner (TLS) [31, 58]. The Middlebury datasets [52, 53, 59] focus on two-frame stereo, providing accurate ground truth for disparity in addition to RGB.

In this group, the DTU dataset [28] with controlled lighting and high-resolution ground truth is most often used for training learning-based MVS methods. Most other MVS datasets, while containing some images of isolated objects, focus on complete scenes, often collected with hand-held, freely positioned cameras [31, 58, 66].

Compared to previously developed datasets with high-resolution 3D scanner data, we provide the largest number of sensors, objects, and lighting conditions, and the most challenging objects.

Datasets with low-resolution depth. Datasets designed for tasks like SLAM, object recognition and segmentation are often collected using low-resolution depth sensors like Microsoft Kinect or Intel RealSense [7, 13, 44, 47, 63, 64]; some of these datasets combine high-resolution RGB with low-resolution depth (*e.g.*, [13, 63]) but do not provide high-



1 RangeVision Spectrum structured light scanner
2 Intel RealSense D435 RGB-D sensor
3 The Imaging Source DFK 33UX250 industrial RGB cameras
4 Microsoft Kinect v2 ToF RGB-D sensor
5 2x Huawei Mate 30 Pro phones with ToF sensors
6 LED strips (ambient light)
7 3x Softboxes (diffuse light)
8 7x Directional lights (hard light)
9 Camera rig
10 Object being scanned

Figure 2. **Our acquisition setup** (view in zoom). We included a diverse set of seven commonly used RGB and RGB-D sensors, mounting them on a shared metal rig to aid data alignment. We constructed a metal frame surrounding the scanning area and installed various light sources to provide 14 lighting setups.

resolution ground truth for depth. A notable exception is CoRBS dataset [73], but it contains only four scenes. The degree of control over camera position and lighting in these datasets varies from complete [11, 63] to none (*e.g.*, [47]).

These datasets are often used for qualitative evaluation of non-learning-based depth fusion methods [15, 27, 57, 77] which produce voxel- or surfel-based surface representations from depth maps; they are also used to train learning-based methods which produce voxel-based surface representation from RGB [43, 67], depth [14, 16], or RGB-D [17] data. These methods are likely to benefit from including our high-resolution depth data in training.

Synthetic datasets ShapeNet [8] and ModelNet [78] are often used to train learning-based depth fusion methods, *e.g.*, [25, 41, 74, 75]. Synthetic ICL-NUIM SLAM benchmark [22] is used for evaluation, for example, in [15, 57, 77]. Such approach is limited by the differences between real-world and synthetic data.

Synthetic benchmarks [6, 32, 35] allow to generate large training sets easily via simulation of the acquisition process. However, real-world data is still required to faithfully model sensors, train generators, and test trained algorithms. Our dataset can be used for these purposes.

Datasets with multiple depth resolutions. Recently proposed RGB-D-D [23] and ARKitScenes [5] datasets make a step in a similar direction to ours, pairing low-resolution depth data acquired by smartphones with medium resolution (0.3 MPix) time-of-flight data and high-resolution laser scans, respectively.

Our dataset contains inputs from multiple depth sensors of low and high resolutions along with associated registered higher resolution RGB images, providing a framework for evaluating and training both depth fusion and RGB-D fusion algorithms previously trained on synthetic data, as well as developing new ones. Having multiple depth resolutions also supports applications such as depth map super-resolution [26, 65, 69, 79] and depth map completion [29, 88].

Device	#	RGB	Depth	IR	Intr.	Extr.	Rec.
DFK 33UX250	2	✓*	—	—	✓	✓	—
Mate 30 Pro	2	✓*	✓	✓	✓	✓	—
RealSense D435	1	✓*	✓*	✓✓*	✓	✓	—
Kinect v2	1	✓*	✓	✓	✓	✓	—
Spectrum	1	—	✓	—	—	—	✓

Table 2. **Composition of our dataset.** We provide RGB, depth, and IR images, intrinsic (Intr.) and extrinsic (Extr.) calibration parameters, and a reference mesh reconstruction (Rec.). The data marked with * is captured per lighting setup.

Similarly to our dataset, a concurrent work [62] complements RGB-D sequences from Intel RealSense with registered ground-truth captured by structured-light scanning. We provide registered depth images from common sensors at three levels of accuracy, and a controlled lighting variation.

3. Dataset

3.1. Overview

Our dataset consists of 107 scenes with a single everyday object or a small group of objects on a black background, see examples in Figure 1 and a complete list of scenes in the supplementary material.

We collected the dataset using multiple sensors mounted on Universal Robots UR10 robotic arm with 6 degrees of freedom and sub-millimeter position repeatability. We used the sensors shown in Figure 2 on the left: (1) RangeVision Spectrum structured-light scanner (SLS), (2) two The Imaging Source DFK 33UX250 industrial RGB cameras, (3) two Huawei Mate 30 Pro phones with time-of-flight (ToF) depth sensors, (4) Intel RealSense D435 stereo RGB-D camera, and (5) Microsoft Kinect v2 ToF RGB-D camera.

We surrounded the scanning area with a metal frame to which we attached the light sources, shown in Figure 2 on the right: seven directional lights, three diffuse soft-boxes, and LED strips which imitate ambient light. We also used flashlights on the phones as the light source moving with the camera. To prevent cross-talk between depth sensors we added external shutters that close the infrared (IR) projector of one sensor while the others are imaging.

For each scene, we moved the camera rig through 100 positions on a sphere with a radius of 70 cm around the object, using the same trajectory for all scenes, and collected the data using 14 lighting setups. For each device, except the SLS, we collected raw RGB, depth, and IR images, including both left and right IR images for RealSense. In total, we collected 15 raw images per scene, camera position, and lighting setup: 6 RGB, 5 IR, and 4 depth images, as illustrated in Table 2 and Figure 1a. As the IR and depth data from ToF sensors of the phones and Kinect is unaffected by the lighting conditions, we captured this data once per camera position. For the SLS we collected partial scans from 27 positions and combined them into a single scan, as we describe further.

In our dataset, we included a large number of objects with challenging and varied surface material properties, as shown in Table 3. A set of qualitative labels corresponding to the key surface reflection parameters were assigned to each object based on visual estimation of these parameters. For example, *Specularity* represents the ratio of specular to diffuse reflectance for one of the dominant materials of the object, and *Reflection sharpness* characterizes how sharp the reflectance function peak is. These labels and their relation to performance of 3D reconstruction methods are discussed in the supplementary material.

3.2. Data acquisition and post-processing

Here, we summarize the main steps of our data acquisition, including selection of camera settings, camera calibration, and preparation of objects for scanning, and then describe our post-processing pipeline, which employs new methods for high-accuracy data registration and automatic generation of occlusion-valid reference depth images from SL scans.

The data acquisition procedure for each scene consisted of the following steps:

1. We automatically selected optimal camera exposure and gain settings for the scene.
2. We performed a preliminary SL scan, and if it was incomplete or low quality due to surface reflection properties, we applied a vanishing opaque matte coating to the object.
3. We scanned the object with the SLS from 27 viewpoints.
4. For coated objects, we accelerated the coating sublimation with hot air while keeping the object stationary. We then collected additional validation SL scans from five viewpoints to verify that the object was not deformed during coating removal.
5. We acquired RGB and low-resolution depth sensor data.

Sensor exposure/gain selection is critical for obtaining useful data: uniform settings result in frequent over- or underexposure due to variations of object surface properties. Hardware auto-exposure, being biased by the black background, proved to be inadequate for our setup and produced too high exposure/gain. Instead, we designed an auto-exposure algorithm inspired by [60, 61], which we used to find optimal camera settings for each scene, lighting, and sensor individually.

To find the *minimal noise* camera settings, we extracted the foreground mask of the scene from the images with and without the object using the method of [36], and then maximized the Shannon entropy of the foreground image w.r.t. the exposure value while keeping the gain at minimum. We then additionally maximized the entropy w.r.t. the gain value while keeping the found exposure value, to prevent the underexposure of very dark objects. To find the *real-time / high-noise* camera settings we swapped exposure and gain, optimizing the gain first while keeping the exposure at 30 FPS, and then optimizing the exposure while keeping the found gain.

Feature	Possible values	Dataset statistics
Specularity	diffuse/low/medium/high	40/15/35/24
Refl. sharpness	no refl./low/medium/high/very high	26/20/23/29/10
Translucency	none/low/medium/high/transparent	93/7/7/5/2
Mirror-like	yes/no	11/96
Metallic	yes/no	9/99
Texture type	color/3D/imperfections	9/24/43
Geom. features	small/medium/no dominant/flat	19/22/47/30

Table 3. **Surface material properties in our dataset.**

We obtained *minimal noise* camera settings for each lighting variant, and *real-time* settings only for the dim flash and ambient lighting. We picked the settings for a single position of the rig per scene and used these settings for all positions. For RealSense, we also picked the optimal power of the IR projector out of 12 options, selecting the one leading to the lowest number of depth pixels missing a value.

Preparation of objects for 3D scanning. For our dataset we picked the objects with surface material properties that challenge common sensors and reconstruction algorithms. However, these properties often challenge the SLS too, making it hard to obtain reliable high-resolution reference data. To get the highest-quality SL scans we applied a temporary coating to about half of the scanned objects (see Figure 3).

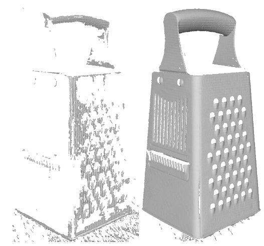


Figure 3. A partial scan without (left) and with coating (right).

Post-processing of structured-light scans. To simplify the use of the SL data for evaluation and training we merged raw partial scans into a single clean surface reconstruction. For this, we globally aligned partial scans using a variant of the iterative closest point algorithm, initialized with calibrated scanner poses. We assessed the alignment quality by comparing the inter-scan distance to the scanner resolution and in case of poor alignment, due to a lack of geometrical features on the object, re-scanned the object with attached 3D markers. From the aligned scans, we reconstructed a surface mesh using screened Poisson Surface Reconstruction [30] with the cell size of 0.3 mm, corresponding to a conservative estimate of the scanner resolution. Finally, we cleaned the surface keeping only the vertices close to the raw scans and manually removing the scanning and reconstruction artifacts and the supporting stand. We refer to the resulting surface as *the SL scan*.

For the temporary coating of the scanned objects we used Aesub Blue scanning spray. It sublimates from the surface at room temperature in a few hours, which we reduced to 5–15 minutes by slightly heating the object with a heat gun. To detect potential object deformations, we aligned the five validation scans to the SL scan and visually checked for

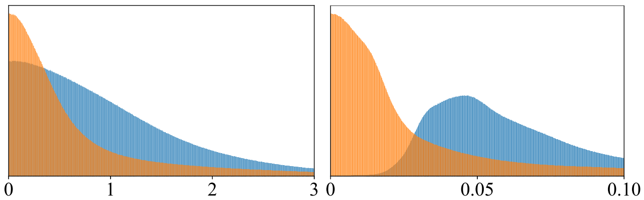


Figure 4. **Calibration refinement results.** Left: distribution of COLMAP reconstruction errors in mm before camera pose refinement (blue) and after (orange). Right: distribution of relative Kinect depth errors w.r.t. SLS before correction (blue) and after (orange).

distance variation indicating deformation. If a deformation was localized only to an isolated part of the object (*e.g.*, a power cord) we removed this part from the scan, otherwise, we excluded the whole object from the dataset.

Camera calibration. To measure the trajectories of the sensors and their intrinsic camera parameters we used the calibration pipeline of [56]. It is based on generic camera models known to result in a more accurate calibration than parametric camera models, commonly used in benchmarks related to ours, including DTU and ETH3D.

Application of this calibration pipeline to our camera rig as is was numerically unstable due to the large variability of sensor properties, such as field of view and resolution. To eliminate instability we split the calibration procedure into several steps, as we describe in the supplementary material.

For all RGB and depth sensors and the SLS, we obtained central generic camera models with several thousands parameters and the trajectory of the sensor in the global coordinate system. The mean calibration error for different sensors at different steps of the procedure was in the range 0.04–0.4 px, or approximately in the range 0.024–0.15 mm.

We observed thermal effects to contribute noticeably to calibration results, in particular, calibration errors drop by factors of 2–10 if the devices warm up after power-on to stable operating temperature. To reduce thermal drift in our data we pre-warmed all devices for 1 hour before calibration and at the beginning of each day of scanning.

Refinement of camera poses. Limited rig stiffness and variations in positioning of the robotic arm lead to a noticeable deviation of the camera trajectory during scanning from the calibrated trajectory, up to several pixels in image space. We additionally refined individual sensor poses per scene w.r.t. SL scan using a new approach inspired by [10, 18].

The idea of these works is to optimize a photometric discrepancy between a sensor image and a render of a reference texture on the scan to the image plane, with respect to the pose of the sensor. In [10], geometric features of the scan, such as normals, are used as the reference texture, while in [18], poses for multiple viewpoints are optimized simultaneously and for each viewpoint the projections from images from all other viewpoints are used as the reference. In both cases, the mutual information between intensity distributions of the sensor image and the render is used as the discrepancy measure, and it is optimized using derivative-free NEWUOA method [49]. We used the same overall idea but changed both key components of the method: the reference texture and the discrepancy measure.

The modification of the original method was required as the previously proposed variants did not yield sufficiently accurate results for our data. The variant of [10] assumes the presence of photometric similarity between the render of the scan surface and the photo of the real surface, which is often not present, *e.g.*, for a smooth surface with color texture, and, importantly, relies on the match between the silhouettes of the rendered scan and the real object in the photo, which is often violated as not all parts of the object visible in the photo may be scanned. Simultaneous optimization of multiple viewpoints in [18] often results in pose drift.

Our approach is to use the raw SLS camera images as the reference texture, since the scan is constructed from these images and they are perfectly aligned. Instead of mutual information between intensity distributions, which we found to be often unstable, we optimized the smooth L_1 distance in deep CNN feature space. Inspired by [37], for feature extraction we used a model pre-trained specifically for feature matching [20].

For some camera poses and sensors, the direct alignment to the SLS images was unstable, as these images were captured under SLS illumination, which differs significantly from the illumination used for the other cameras. To further stabilize the alignment procedure for the whole dataset, we split it into three steps. First, we aligned, directly to the SLS reference texture, a single image from an industrial RGB camera, for a viewpoint selected for each scene individually to maximize the alignment tightness. Then, we aligned the images from the industrial RGB camera for the remaining viewpoints successively, using the already aligned images as the reference. Finally, for all the other sensors, we aligned each image individually to the image from the industrial RGB camera captured at the same position of the rig.

Using this sequential sensor alignment procedure we obtained subpixel alignment accuracy in most cases. In Figure 4 left, we show that refinement of camera poses reduces the error of a 3D reconstruction method.

Refinement of depth camera calibration. The lower quality depth sensors were calibrated by the manufacturers using calibration systems different from the one we used for the SLS and the camera trajectories. This leads to a misalignment between the lower quality depth data and the SL scan, which can be represented as a composition of a 3D rigid transform, scaling and a non-linear warping [24, 68, 86, 89].

To reduce the misalignment, we used a correction model $d_{\text{undist}} = S(u, v)S(d_{\text{raw}})$, which transforms the raw depth measurement d_{raw} at pixel (u, v) in the depth image to the depth value in the 3D space of the SL scan, using the cubic basis splines $S(u, v)$ and $S(d)$ defined on regular grids. We

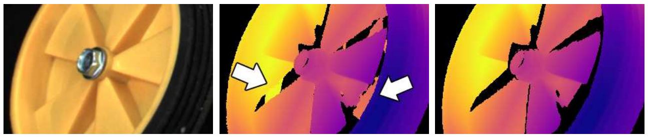


Figure 5. **Rendering SL depth with occluding surface.** Although the real surface of the object is solid (left), its SL scan is incomplete, so the background parts of the object bleed through in the rendered depth map (middle). Occluding surface helps to discard incorrect depth values (right).

trained the model on depth images of a calibration pattern, and as the reference used the calibration data from the same system that we used for camera calibration.

In Figure 4 right, we show that refinement of depth camera calibration reduces the error of a low quality sensor depth w.r.t. the SL scan.

Rendering SL scans. Occluding surface. In many tasks that we target, the ground-truth 3D data is needed in the form of depth maps. At the same time, the SL scanning often does not capture the complete object due to its geometric limitations, so a direct rendering of a depth map from the SL scan is likely to produce incorrect large depth values at some points because closer parts of the surface are absent from the scan, as illustrated in Figure 5 middle.

Previous works have addressed this problem by discarding depth values either inconsistent with an MVS reconstruction from RGB images [39], or occluded by a complete surface reconstructed from the scan and adjusted manually [58]. We follow the latter, more reliable approach, but propose a fully automatic method for building the occluding surface.

An *occluding surface* for an object is any surface fully enclosing the object. Our goal is to construct an occluding surface tight to the real surface of the object, using the information about visibility of the real surface from the viewpoint of the SLS camera during scanning. The depth values rendered from the SL scan that differ from the values rendered from such an occluding surface by more than a certain threshold can then be identified as spurious and removed from the generated depth maps.

To find the occluding surface we assume that the convex hull of all partial SL scans C encloses the part of the object that we are going to render. Conceptually, we find the occluding surface as $C \setminus \cup_i \mathrm{Cone}(v_i, S_i)$, where the cone $\mathrm{Cone}(v_i, S_i)$ encloses all the points between the surface of the partial scan S_i and the respective viewpoint of the SLS camera v_i. Intuitively, we carve the free space between the scanner and the object out of the convex hull, for all positions of the scanner.

The direct computation of such an occluding surface by a sequence of boolean operations on closed meshes is extremely expensive as the number of triangles for each mesh can be very large. Instead, we compute an accurate approximation by sampling all surfaces involved (we used the sampling distance of 0.1 mm) and keeping only the points that are in the interior of the convex hull and in the exterior of all cones, which we check using depth testing. We then reconstruct the occluding surface from these samples using screened Poisson Surface Reconstruction.

The occluding surface obtained with this method can be used to generate accurate, occlusion-valid depth images from SL data as shown in Figure 5 right.

4. Experimental evaluation

We demonstrate the challenges of our dataset for RGB-based 3D reconstruction methods, and additionally, demonstrate one of the uses of our dataset as a benchmark to compare methods of 3D reconstruction from different modalities. For this, we tested 5 methods which reconstruct the surface only from RGB data, 3 methods which use only depth data, and one method which uses both modalities.

Methods. COLMAP [54, 55] is an RGB-to-3D reconstruction pipeline. It implements a non-learning-based multi-view stereo (MVS) method: from multi-view RGB images, the depth maps are estimated per-view and then fused into a point cloud representing the reconstructed 3D surface. COLMAP is commonly evaluated on benchmarks similar to ours, so we include it as a baseline. ACMP [81] is a non-learning, PatchMatch-based MVS method with planar prior, which shows a strong performance on benchmarks such as Tanks and Temples (TnT). VisMVSNet [87] and UniMVS-Net [48] are learning, plane-sweeping-based MVS methods with top performance on TnT benchmark. NeuS [71] is a method which reconstructs the surface as a signed distance function (SDF) represented with a neural network which is fitted directly to RGB images via differentiable rendering. TSDF Fusion [12, 27] is a classical non-learning-based depth-fusion approach. It reconstructs a truncated SDF (TSDF) of a surface on a voxel grid via iterative integration of depth maps. RoutedFusion [74] is a learning-based extension, which performs depth map integration using neural networks. SurfelMeshing [57] is a non-learning-based depth-fusion method that reconstructs the surface using a surfel cloud as an intermediate representation. Neural RGB-D surface reconstruction [3] is a recent method which, similarly to NeuS, reconstructs the TSDF of a surface represented with a neural network, but in contrast to NeuS, fits the network to RGB *and depth* images.

Data. We tested the methods under the most favorable conditions available in our dataset. We obtained the reconstructions using full-resolution images from all 100 viewpoints for each scene, and the ground-truth camera poses and intrinsic camera models. For each data modality we used the highest-quality option: the photos from one of the industrial RGB cameras with the minimal noise settings, taken under ambient light to rule out the effects of lighting, and the depth maps from the ToF sensor of Kinect, of a higher resolution than of the phones, and more stable than RealSense.

For the learning-based methods we tested the models trained on prior datasets: DTU [28] and BlendedMVS [83] for MVSNets, and ModelNet [78] for RoutedFusion. We provide more details in the supplementary material.

Measures. We evaluated the reconstructions w.r.t. the full SL scans using Precision, Recall, and F-score quality measures, similarly to the prior benchmarks [31,58]. Precision measures the reconstruction accuracy: we calculated it as the percentage of the reconstruction points closer to the reference than a certain distance threshold. Recall measures the reconstruction completeness: we calculated it as the percentage of the *reference* points closer to the reconstruction than a threshold. F-score is the harmonic mean of these two numbers, which, to be high, requires the reconstruction to be both accurate and complete.

For a careful calculation of Precision and Recall two problems have to be considered. First, both the reconstruction and the reference may have varying point densities, which will cause uneven contribution of different parts of the surface to the value of the measure. Second, the reference SL scans are incomplete, so the distance from some reconstruction points to the SL scan does not represent the distance to the real surface of the object, specifically, if the point lies near the missing part of the surface. We addressed these problems similarly to [58], extending the approach using our new occluding surface. Specifically, we calculated each measure in small cells of 3D space and then took the average as the final value, and used only the points for which the distance to the real surface of the object is certain. We describe this in more detail in the supplementary material.

Results. In Figure 6 we show the average performance on our dataset for the tested methods. Each curve shows the number of scenes that the respective method reconstructs with a better value of the measure than the value on the X axis. We tested Neural RGB-D surface reconstruction only on a fraction of scenes, due to its long running time, so we exclude the curve for this method.

In Figure 7 we show reconstructions for several scenes. The top part of the figure shows the results produced by methods of different types: an RGB-only UniMVSNet, an RGB-only NeuS with neural representation, Neural RGB-D surface reconstruction, and a depth-only TSDF Fusion. The middle part shows the best result per scene w.r.t. Recall. The bottom part shows the best result w.r.t. Precision.

In these figures we show the measures calculated with the threshold of 0.5 mm. In the supplementary material we show distributions of the measures for other values of the threshold, distributions of additional quality measures, and additional reconstruction visualizations.

Discussion. The best method w.r.t. Recall (ACMP) reconstructs all scenes in our dataset on at least 53%, with the distance threshold of 0.5 mm, but only half of the scenes on 80% or more. The best method w.r.t. Precision and the overall quality represented with F-score (VisMVSNet) reconstructs all scenes with at least 32% accuracy and only 14 scenes with accuracy higher than 80%. This demonstrates that our dataset contains plenty of challenges for state-of-the-art 3D reconstruction methods. In particular, featureless parts of the surface, especially with sharp reflections, are often missing in the reconstruction, as illustrated in Figure 7. At the same time, VisMVSNet significantly outperforms UniMVSNet w.r.t. Precision and F-score and performs almost as well w.r.t. Recall. This is opposite to their relative performance on prior benchmarks (TnT and DTU), which indicates that our dataset poses a different set of challenges.

Comparison of the methods of different types shows that reconstructions from depth maps are significantly less accurate than reconstructions from RGB images, although sometimes may be more complete, as illustrated at the top of Figure 7. This demonstrates that these modalities can complement each other. Similarly, NeuS, which uses a neural surface representation, fills-in the areas of the surface challenging for RGB-based methods, but often inaccurately. Remarkably, Neural RGB-D surface reconstruction produces the surface of a significantly lower quality in comparison to VisMVSNet and NeuS, while using the same input RGB images and additionally the depth maps. This illustrates that there is a room for development of 3D reconstruction methods that effectively use both modalities, and we believe that our dataset will facilitate the development of such methods.

5. Conclusion

We presented a new dataset for evaluation and training of 3D reconstruction algorithms. Compared to prior datasets the distinguishing features of ours include a large number of sensors of different modalities and resolutions, depth sensors in particular, selection of scenes presenting difficulties for many existing algorithms, and high-quality reference data for these scenes. Our dataset can support training and evaluation of methods for many variations of 3D reconstruction tasks, in particular, learning-based 3D surface reconstruction from multi-view RGB-D data.

The main (intentional) limitation of our dataset is the use of the laboratory setting: the focus on static isolated objects with easy-to-separate background, the same camera trajectory for all scenes, the laboratory lighting. Another possible limitation is a small range of object sizes, limited by the physical size of the setup.

Acknowledgements. The authors acknowledge the use of Skoltech supercomputer Zhores [85] for obtaining the results presented in this paper. E. Burnaev and O. Voynov were supported by the Analytical center under the RF Government (subsidy agreement 000000D730321P5Q0002, Grant No. 70-2021-00145 02.11.2021).

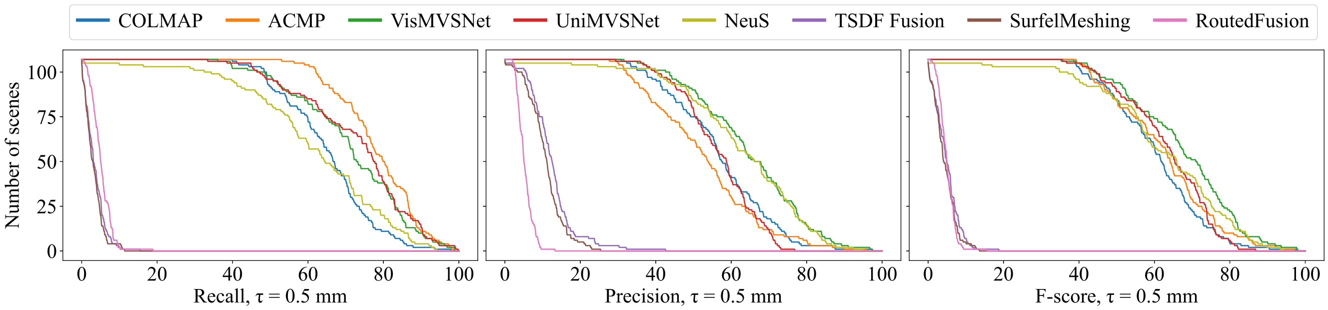


Figure 6. **Average performance per method** as the number of scenes with reconstruction quality better than the value on the X axis.

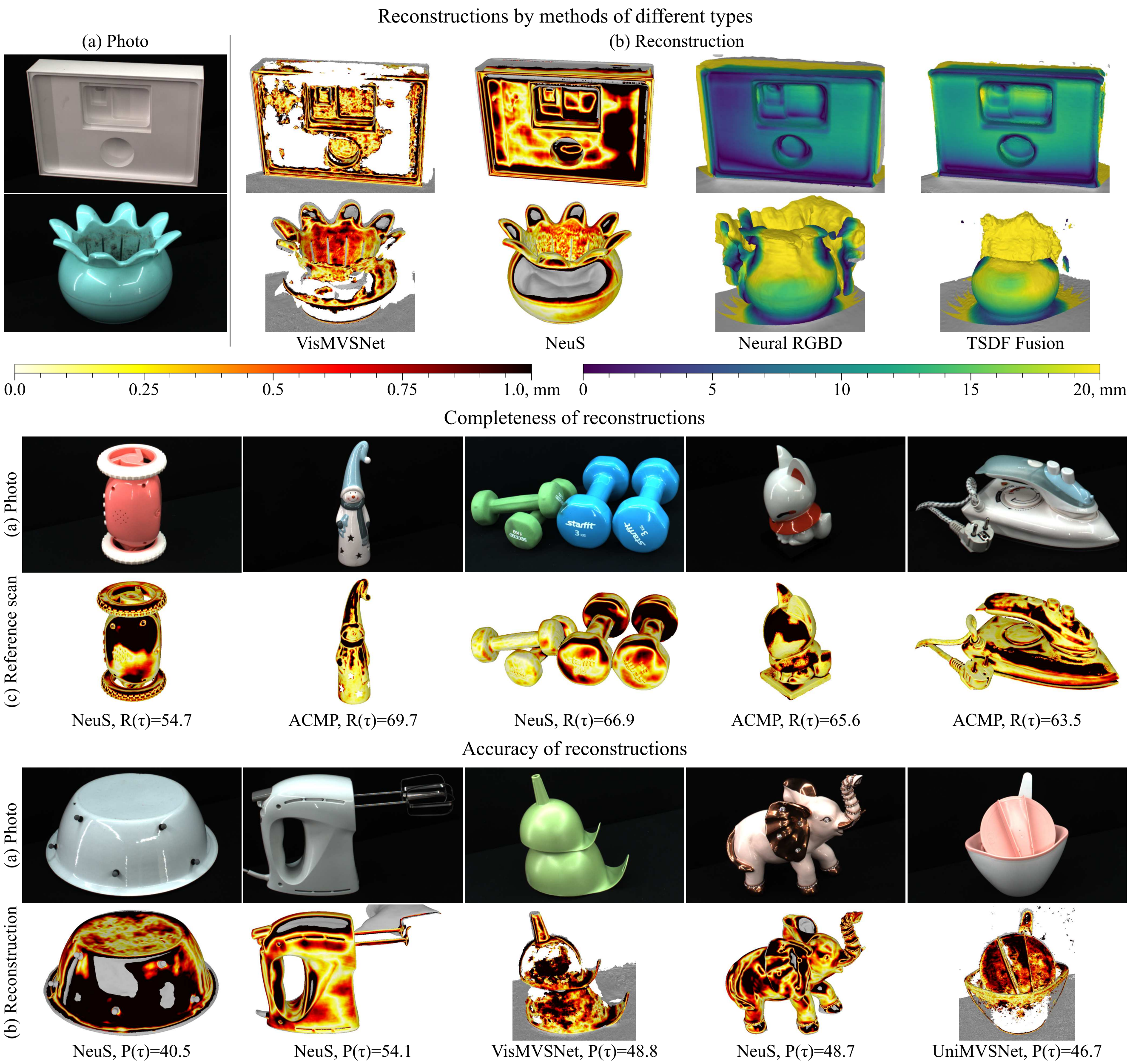


Figure 7. **Qualitative results.** (a) A photo of the scene. (b) Reconstruction with color-coded distance to the SL scan. (c) The SL scan with color-coded distance to the reconstruction. The two bottom parts show only the best result per scene and the respective value of Recall or Precision for $\tau = 0.5$ mm. Grey color represents undefined distance to the real surface, as explained in the supplementary material.

References

[1] Henrik Aanæs, Rasmus Ramsbøl Jensen, George Vogiatzis, Engin Tola, and Anders Bjorholm Dahl. Large-scale data for multiple-view stereopsis. *International Journal of Computer Vision*, 120(2):153–168, 2016. 2

[2] Kara-Ali Aliev, Artem Sevastopolsky, Maria Kolos, Dmitry Ulyanov, and Victor Lempitsky. Neural point-based graphics. In *European Conference on Computer Vision*, pages 696–712. Springer, 2020. 2

[3] Dejan Azinović, Ricardo Martin-Brualla, Dan B Goldman, Matthias Nießner, and Justus Thies. Neural rgb-d surface reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6290–6301, 2022. 6

[4] Connelly Barnes, Eli Shechtman, Adam Finkelstein, and Dan B Goldman. PatchMatch: A randomized correspondence algorithm for structural image editing. *ACM Transactions on Graphics (Proc. SIGGRAPH)*, 28(3), Aug. 2009. 2

[5] Gilad Baruch, Zhuoyuan Chen, Afshin Dehghan, Tal Dimry, Yuri Feigin, Peter Fu, Thomas Gebauer, Brandon Joffe, Daniel Kurz, Arik Schwartz, and Elad Shulman. ARKitscenes - a diverse real-world dataset for 3d indoor scene understanding using mobile RGB-d data. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*, 2021. 3

[6] Matthew Berger, Joshua A Levine, Luis Gustavo Nonato, Gabriel Taubin, and Claudio T Silva. A benchmark for surface reconstruction. *ACM Transactions on Graphics (TOG)*, 32(2):1–17, 2013. 3

[7] Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niessner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3d: Learning from rgb-d data in indoor environments. *International Conference on 3D Vision (3DV)*, 2017. 2

[8] Angel X. Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. ShapeNet: An Information-Rich 3D Model Repository. Technical Report arXiv:1512.03012 [cs.GR], Stanford University — Princeton University — Toyota Technological Institute at Chicago, 2015. 3

[9] Jaesung Choe, Sunghoon Im, Francois Rameau, Minjun Kang, and In So Kweon. Volumefusion: Deep depth fusion for 3d scene reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 16086–16095, October 2021. 2

[10] Massimiliano Corsini, Matteo Dellepiane, Federico Ponchio, and Roberto Scopigno. Image-to-geometry registration: a mutual information method exploiting illumination-related geometric properties. In *Computer Graphics Forum*, volume 28, pages 1755–1764. Wiley Online Library, 2009. 5

[11] D. Cremers and K. Kolev. Multiview stereo and silhouette consistency via convex functionals over convex domains. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(6):1161–1174, 2011. 3

[12] Brian Curless and Marc Levoy. A volumetric method for building complex models from range images. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques - SIGGRAPH '96*, pages 303–312, Not Known, 1996. ACM Press. 6

[13] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. ScanNet: Richly-annotated 3D reconstructions of indoor scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5828–5839, 2017. 2

[14] Angela Dai, Christian Diller, and Matthias Niessner. Sg-nn: Sparse generative neural networks for self-supervised scene completion of rgb-d scans. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 3

[15] Angela Dai, Matthias Nießner, Michael Zollhöfer, Shahram Izadi, and Christian Theobalt. BundleFusion: Real-Time Globally Consistent 3D Reconstruction Using On-the-Fly Surface Reintegration. *ACM Transactions on Graphics*, 36(3):1–18, July 2017. 3

[16] Angela Dai, Daniel Ritchie, Martin Bokeloh, Scott Reed, Jürgen Sturm, and Matthias Nießner. Scancomplete: Large-scale scene completion and semantic segmentation for 3d scans. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4578–4587, 2018. 3

[17] Angela Dai, Yawar Siddiqui, Justus Thies, Julien Valentin, and Matthias Niessner. Spsg: Self-supervised photometric scene generation from rgb-d scans. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1747–1756, June 2021. 3

[18] Matteo Dellepiane and Roberto Scopigno. Global refinement of image-to-geometry registration for color projection. In *2013 Digital Heritage International Congress (DigitalHeritage)*, volume 1, pages 39–46. IEEE, 2013. 5

[19] Simon Donne and Andreas Geiger. Learning non-volumetric depth fusion using successive reprojections. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 2

[20] Hugo Germain, Guillaume Bourmaud, and Vincent Lepetit. S2dnet: Learning accurate correspondences for sparse-to-dense feature matching. *arXiv preprint arXiv:2004.01673*, 2020. 5

[21] Xiaodong Gu, Zhiwen Fan, Siyu Zhu, Zuozhuo Dai, Feitong Tan, and Ping Tan. Cascade cost volume for high-resolution multi-view stereo and stereo matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2

[22] A. Handa, T. Whelan, J.B. McDonald, and A.J. Davison. A benchmark for RGB-D visual odometry, 3D reconstruction and SLAM. In *IEEE Intl. Conf. on Robotics and Automation, ICRA*, Hong Kong, China, May 2014. 3

[23] Lingzhi He, Hongguang Zhu, Feng Li, Huihui Bai, Runmin Cong, Chunjie Zhang, Chunyu Lin, Meiqin Liu, and Yao Zhao. Towards fast and accurate real-world depth super-resolution: Benchmark dataset and baseline. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9229–9238, June 2021. 3

[24] Daniel Herrera, Juho Kannala, and Janne Heikkilä. Joint depth and color camera calibration with distortion correction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(10):2058–2064, 2012. 5

[25] Jiahui Huang, Shi-Sheng Huang, Haoxuan Song, and Shi-Min Hu. Di-fusion: Online implicit 3d reconstruction with deep priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8932–8941, June 2021. 3

[26] Tak-Wai Hui, Chen Change Loy, , and Xiaoou Tang. Depth map super-resolution by deep multi-scale guidance. In *Proceedings of European Conference on Computer Vision (ECCV)*, pages 353–369, 2016. 3

[27] Shahram Izadi, Andrew Davison, Andrew Fitzgibbon, David Kim, Otmar Hilliges, David Molyneaux, Richard Newcombe, Pushmeet Kohli, Jamie Shotton, Steve Hodges, and Dustin Freeman. KinectFusion: real-time 3D reconstruction and interaction using a moving depth camera. In *Proceedings of the 24th annual ACM symposium on User interface software and technology - UIST '11*, page 559, Santa Barbara, California, USA, 2011. ACM Press. 3, 6

[28] Rasmus Jensen, Anders Dahl, George Vogiatzis, Engil Tola, and Henrik Aanæs. Large scale multi-view stereopsis evaluation. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 406–413. IEEE, 2014. 2, 7

[29] Junho Jeon and Seungyong Lee. Reconstruction-based pairwise depth dataset for depth image enhancement using cnn. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018. 3

[30] Michael Kazhdan and Hugues Hoppe. Screened poisson surface reconstruction. *ACM Transactions on Graphics (ToG)*, 32(3):1–13, 2013. 4

[31] Arno Knapitsch, Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. Tanks and temples: Benchmarking large-scale scene reconstruction. *ACM Transactions on Graphics (ToG)*, 36(4):1–13, 2017. 2, 7

[32] Sebastian Koch, Yurii Piadyk, Markus Worchel, Marc Alexa, Cláudio Silva, Denis Zorin, and Daniele Panozzo. Hardware design and accurate simulation for benchmarking of 3D reconstruction algorithms. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021. 3

[33] Andreas Kuhn, Christian Sormann, Mattia Rossi, Oliver Erdler, and Friedrich Fraundorfer. Deepc-mvs: Deep confidence prediction for multi-view stereo reconstruction. In *2020 International Conference on 3D Vision (3DV)*, pages 404–413, 2020. 2

[34] Vincent Leroy, Jean-Sebastien Franco, and Edmond Boyer. Shape reconstruction using volume sweeping and learned photoconsistency. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018. 2

[35] Andreas Ley, Ronny Hänsch, and Olaf Hellwich. Syb3r: A realistic synthetic benchmark for 3d reconstruction from images. In *European Conference on Computer Vision*, pages 236–251. Springer, 2016. 3

[36] Shanchuan Lin, Andrey Ryabtsev, Soumyadip Sengupta, Brian L Curless, Steven M Seitz, and Ira Kemelmacher-Shlizerman. Real-time high-resolution background matting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8762–8771, 2021. 4

[37] Philipp Lindenberger, Paul-Edouard Sarlin, Viktor Larsson, and Marc Pollefeys. Pixel-perfect structure-from-motion with featuremetric refinement. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5987–5997, 2021. 5

[38] Yuanzhi Liu, Yujia Fu, Fengdong Chen, Bart Goossens, Wei Tao, and Hui Zhao. Simultaneous localization and mapping related datasets: A comprehensive survey. *arXiv preprint arXiv:2102.04036*, 2021. 2

[39] Keyang Luo, Tao Guan, Lili Ju, Yuesong Wang, Zhuo Chen, and Yawei Luo. Attention-aware multi-view stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1590–1599, 2020. 6

[40] Xinjun Ma, Yue Gong, Qirui Wang, Jingwei Huang, Lei Chen, and Fan Yu. Epp-mvsnet: Epipolar-assembling based depth prediction for multi-view stereo. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 5732–5740, October 2021. 2

[41] Marko Mihajlovic, Silvan Weder, Marc Pollefeys, and Martin R. Oswald. Deepsurfels: Learning online appearance fusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14524–14535, June 2021. 3

[42] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *European Conference on Computer Vision*, pages 405–421. Springer, 2020. 2

[43] Zak Murez, Tarrence van As, James Bartolozzi, Ayan Sinha, Vijay Badrinarayanan, and Andrew Rabinovich. Atlas: End-to-end 3d scene reconstruction from posed images. In *European Conference on Computer Vision (ECCV)*, 2020. 3

[44] Pushmeet Kohli Nathan Silberman, Derek Hoiem and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *ECCV*, 2012. 2

[45] Michael Niemeyer, Lars Mescheder, Michael Oechsle, and Andreas Geiger. Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3504–3515, 2020. 2

[46] Michael Oechsle, Songyou Peng, and Andreas Geiger. Unisurf: Unifying neural implicit surfaces and radiance fields for multi-view reconstruction. *arXiv preprint arXiv:2104.10078*, 2021. 2

[47] Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. Colored point cloud registration revisited. In *ICCV*, 2017. 2, 3

[48] Rui Peng, Rongjie Wang, Zhenyu Wang, Yawen Lai, and Ronggang Wang. Rethinking depth estimation for multi-view stereo: A unified representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8645–8654, 2022. 6

[49] Michael JD Powell. The newuoa software for unconstrained optimization without derivatives. In *Large-scale nonlinear optimization*, pages 255–297. Springer, 2006. 5

[50] Ruslan Rakhimov, Andrei-Timotei Ardelean, Victor Lempitsky, and Evgeny Burnaev. Npbg++: Accelerating neural point-based graphics. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15969–15979, 2022. 2

[51] Gernot Riegler and Vladlen Koltun. Stable view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12216–12225, 2021. 2

[52] Daniel Scharstein, Heiko Hirschmüller, York Kitajima, Greg Krathwohl, Nera Nešić, Xi Wang, and Porter Westling. High-resolution stereo datasets with subpixel-accurate ground truth. In *German Conference on Pattern Recognition*, pages 31–42. Springer, 2014. 2

[53] Daniel Scharstein, Heiko Hirschmüller, York Kitajima, Greg Krathwohl, Nera Nešić, Xi Wang, and Porter Westling. High-resolution stereo datasets with subpixel-accurate ground truth. In *German conference on pattern recognition*, pages 31–42. Springer, 2014. 2

[54] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-Motion Revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 6

[55] Johannes Lutz Schönberger, Enliang Zheng, Marc Pollefeys, and Jan-Michael Frahm. Pixelwise View Selection for Unstructured Multi-View Stereo. In *European Conference on Computer Vision (ECCV)*, 2016. 1, 2, 6

[56] Thomas Schops, Viktor Larsson, Marc Pollefeys, and Torsten Sattler. Why having 10,000 parameters in your camera model is better than twelve. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2535–2544, 2020. 5

[57] Thomas Schops, Torsten Sattler, and Marc Pollefeys. SurfelMeshing: Online Surfel-Based Mesh Reconstruction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(10):2494–2507, Oct. 2020. 3, 6

[58] Thomas Schops, Johannes L Schonberger, Silvano Galliani, Torsten Sattler, Konrad Schindler, Marc Pollefeys, and Andreas Geiger. A multi-view stereo benchmark with high-resolution images and multi-camera videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3260–3269, 2017. 2, 6, 7

[59] Steven M Seitz, Brian Curless, James Diebel, Daniel Scharstein, and Richard Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *2006 IEEE computer society conference on computer vision and pattern recognition (CVPR'06)*, volume 1, pages 519–528. IEEE, 2006. 2

[60] Inwook Shim, Tae-Hyun Oh, Joon-Young Lee, Jinwook Choi, Dong-Geol Choi, and In So Kweon. Gradient-based camera exposure control for outdoor mobile platforms. *IEEE Transactions on Circuits and Systems for Video Technology*, 29(6):1569–1583, 2018. 4

[61] Ukcheol Shin, Jinsun Park, Gyumin Shim, Francois Rameau, and In So Kweon. Camera exposure control for robust robot vision with noise-aware image quality assessment. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1165–1172. IEEE, 2019. 4

[62] Rakesh Shrestha, Siqi Hu, Minghao Gou, Ziyuan Liu, and Ping Tan. A real world dataset for multi-view 3d reconstruction. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part VIII*, pages 56–73. Springer, 2022. 3

[63] Arjun Singh, James Sha, Karthik S Narayan, Tudor Achim, and Pieter Abbeel. BigBIRD: A large-scale 3D database of object instances. In *2014 IEEE international conference on robotics and automation (ICRA)*, pages 509–516. IEEE, 2014. 2, 3

[64] Shuran Song, Samuel P Lichtenberg, and Jianxiong Xiao. SUN RGB-D: A RGB-D scene understanding benchmark suite. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 567–576, 2015. 2

[65] Xibin Song, Yuchao Dai, Dingfu Zhou, Liu Liu, Wei Li, Hongdong Li, and Ruigang Yang. Channel attention based iterative residual learning for depth map super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 3

[66] Christoph Strecha, Wolfgang Von Hansen, Luc Van Gool, Pascal Fua, and Ulrich Thoennessen. On benchmarking camera calibration and multi-view stereo for high resolution imagery. In *2008 IEEE conference on computer vision and pattern recognition*, pages 1–8. Ieee, 2008. 2

[67] Jiaming Sun, Yiming Xie, Linghao Chen, Xiaowei Zhou, and Hujun Bao. Neuralrecon: Real-time coherent 3d reconstruction from monocular video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15598–15607, June 2021. 3

[68] Alex Teichman, Stephen Miller, and Sebastian Thrun. Unsupervised intrinsic calibration of depth sensors via slam. In *Robotics: Science and systems*, volume 248, page 3, 2013. 5

[69] Oleg Voynov, Alexey Artemov, Vage Egiazarian, Alexander Notchenko, Gleb Bobrovskikh, Evgeny Burnaev, and Denis Zorin. Perceptual deep depth super-resolution. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5653–5663, 2019. 3

[70] Fangjinhua Wang, Silvano Galliani, Christoph Vogel, Pablo Speciale, and Marc Pollefeys. Patchmatchnet: Learned multi-view patchmatch stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2

[71] Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. *NeurIPS*, 2021. 2, 6

[72] Qianqian Wang, Zhicheng Wang, Kyle Genova, Pratul P Srinivasan, Howard Zhou, Jonathan T Barron, Ricardo Martin-Brualla, Noah Snavely, and Thomas Funkhouser. Ibrnet: Learning multi-view image-based rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4690–4699, 2021. 2

[73] Oliver Wasenmüller, Marcel Meyer, and Didier Stricker. CoRBS: Comprehensive rgb-d benchmark for slam using kinect v2. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, March 2016. 3

[74] Silvan Weder, Johannes Schonberger, Marc Pollefeys, and Martin R. Oswald. RoutedFusion: Learning Real-Time Depth Map Fusion. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4886–4896, Seattle, WA, USA, June 2020. IEEE. 3, 6

[75] Silvan Weder, Johannes L. Schonberger, Marc Pollefeys, and Martin R. Oswald. Neuralfusion: Online depth fusion in latent space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3162–3172, June 2021. 3

[76] Zizhuang Wei, Qingtian Zhu, Chen Min, Yisong Chen, and Guoping Wang. Aa-rmvsnet: Adaptive aggregation recurrent multi-view stereo network. In *Proceedings of the IEEE/CVF*

International Conference on Computer Vision, pages 6187–6196, 2021. 2

[77] Thomas Whelan, Renato F Salas-Moreno, Ben Glocker, Andrew J Davison, and Stefan Leutenegger. ElasticFusion: Real-time dense SLAM and light source estimation. *The International Journal of Robotics Research*, 35(14):1697–1716, Dec. 2016. 3

[78] Zhirong Wu, Shuran Song, Aditya Khosla, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shape modeling. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, USA, June 2015. 3, 7

[79] Chuhua Xian, Kun Qian, Zitian Zhang, and Charlie C. L. Wang. Multi-Scale Progressive Fusion Learning for Depth Map Super-Resolution. *arXiv e-prints*, page arXiv:2011.11865, Nov. 2020. 3

[80] Qingshan Xu and Wenbing Tao. Multi-scale geometric consistency guided multi-view stereo. *Computer Vision and Pattern Recognition (CVPR)*, 2019. 2

[81] Qingshan Xu and Wenbing Tao. Planar prior assisted patchmatch multi-view stereo. *AAAI Conference on Artificial Intelligence (AAAI)*, 2020. 2, 6

[82] Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long Quan. Mvsnet: Depth inference for unstructured multi-view stereo. *European Conference on Computer Vision (ECCV)*, 2018. 1, 2

[83] Yao Yao, Zixin Luo, Shiwei Li, Jingyang Zhang, Yufan Ren, Lei Zhou, Tian Fang, and Long Quan. Blendedmvs: A large-scale dataset for generalized multi-view stereo networks. *Computer Vision and Pattern Recognition (CVPR)*, 2020. 2, 7

[84] Lior Yariv, Yoni Kasten, Dror Moran, Meirav Galun, Matan Atzmon, Basri Ronen, and Yaron Lipman. Multiview neural surface reconstruction by disentangling geometry and appearance. *Advances in Neural Information Processing Systems*, 33, 2020. 2

[85] Igor Zacharov, Rinat Arslanov, Maksim Gunin, Daniil Stefonishin, Andrey Bykov, Sergey Pavlov, Oleg Panarin, Anton Maliutin, Sergey Rykovanov, and Maxim Fedorov. "zhores"-petaflops supercomputer for data-driven modeling, machine learning and artificial intelligence installed in skolkovo institute of science and technology. *Open Engineering*, 9(1):512–520, 2019. 7

[86] Bernhard Zeisl and Marc Pollefeys. Structure-based auto-calibration of rgb-d sensors. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5076–5083. IEEE, 2016. 5

[87] Jingyang Zhang, Yao Yao, Shiwei Li, Zixin Luo, and Tian Fang. Visibility-aware multi-view stereo network. *British Machine Vision Conference (BMVC)*, 2020. 2, 6

[88] Yinda Zhang and Thomas Funkhouser. Deep Depth Completion of a Single RGB-D Image. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 175–185, Salt Lake City, UT, USA, June 2018. IEEE. 3

[89] Qian-Yi Zhou and Vladlen Koltun. Simultaneous localization and calibration: Self-calibration of consumer depth cameras. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 454–460, 2014. 5