Volume Sweeping: Learning Photoconsistency for Multi-View Shape Reconstruction

Vincent Leroy, Jean-Sébastien Franco, Edmond Boyer

Noname manuscript No.
(will be inserted by the editor)

Volume Sweeping: Learning Photoconsistency for Multi-View Shape Reconstruction

Vincent Leroy[1,2] · Jean-Sébastien Franco[1] · Edmond Boyer[1]

the date of receipt and acceptance should be inserted later

Abstract We propose a full study and methodology for multi-view stereo reconstruction with performance capture data. Multi-view 3D reconstruction has largely been studied with general, high resolution and high texture content inputs, where classic low-level feature extraction and matching are generally successful. However in performance capture scenarios, texture content is limited by wider angle shots resulting in smaller subject projection areas, and intrinsically low image content of casual clothing. We present a dedicated pipeline, based on a per-camera depth map sweeping strategy, analyzing in particular how recent deep network advances allow to replace classic multi-view photoconsistency functions with one that is learned. We show that learning based on a volumetric receptive field around a 3D depth candidate improves over using per-view 2D windows, giving the photoconsistency inference more visibility over local 3D correlations in viewpoint color aggregation. Despite being trained on a standard dataset of scanned static objects, the proposed method is shown to generalize and significantly outperform existing approaches on performance capture data, while achieving competitive results on recent benchmarks.

Keywords Multi View Stereo Reconstruction · Learned Photoconsistency · Performance Capture · Volume Sweeping

Vincent Leroy
E-mail: vincent.leroy@naverlabs.com
Jean-Sébastien Franco
E-mail: jean-sebastien.franco@inria.fr
Edmond Boyer
E-mail: edmond.boyer@inria.fr
[1] Univ. Grenoble Alpes, Inria, CNRS, Grenoble INP*, LJK 38000 Grenoble, France * Institute of Engineering Univ. Grenoble Alpes
[2] NAVER LABS Europe, 6 chemin de Maupertuis, Meylan, 38240, France

1 Introduction

In this paper, we examine the problem of multi-view shape reconstruction of production-realistic performance capture sequences. Such sequences may contain arbitrary casual clothing and motions, and have specific capture set assumptions due to the particular lighting and camera positioning of these setups. Multi-view 3D reconstruction is a popular and mature field, with numerous applications involving the recording and replay of captured 3D scenes, such as 3D content creation for broadcast and mobile applications, or the increasingly popular virtual and augmented reality applications with 3D user avatars. An essential and still improvable aspect in this matter, in particular with performance capture setups, is the fidelity and quality of the recovered shapes, our goal in this work.

Multi-view stereo (MVS) based methods have attained a good level of quality with pipelines that typically comprise feature extraction, matching stages and 3D shape extraction. Interestingly, very recent works have re-examined stereo and MVS by introducing features and similarity functions automatically inferred using deep learning. The main promise of this methodological shift is to include better data-driven priors, either in 2D [1,2,3,4] as improvement over classic 2D features, or in 3D to account for relative view placement and local or global shape priors [5,6,7]. These novel MVS methods have been shown to outperform classic learning-free methods on static scene benchmarks [8].

Our main goal is to examine whether these data-driven improvements transfer to the more complex case of live performance capture, where a diverse set of additional difficulties arise with respect to typical MVS setups. Typical challenges for these capture situations include smaller visual projection areas of objects of in-

Fig. 1 Challenging scene captured with a passive RGB multi-camera setup [9]. (*left*) one input image, (*center*) reconstructions obtained with classical 2D features [10], (*right*) proposed solution. Our results validate the key improvement of a CNN-learned disparity to MVS for performance capture scenarios. Results particularly improve in noisy, very low contrast and low textured regions such as the arm, the leg or even the black skirt folds, which can be better seen in a brightened version of the picture in Figure 17.

terest, due to wider necessary fields of view for capturing motion; occlusion and self-occlusion of several subjects interacting together; lack of texture content typical of real-life subject appearance and clothing; or motion blur with fast moving subjects such as sport action scenes (see Figure 14). To the best of our knowledge, existing learning-based MVS schemes report results on static datasets such as DTU [11] or ShapeNet [12] but have not yet been demonstrated on performance capture data with the aforementioned typical issues.

We present a detailed framework for this purpose, which casts the problem as a fusion of per view depth maps as inspired by recent fusion methods [13], each depth map extracted using a learned multi-view photoconsistency function. Our approach performs multi-view matching within local volumetric units of inference. Contrary to previous methods, our volumetric unit is defined in a given view's own reference, so as to capture camera inherent 3D dependencies, specifically for the purpose of per-view decision. Instead of inferring occupancies, we infer disparity scores to ease training and to focus the method more on photometric configurations than local shape patterns. We sweep viewing rays with this volumetric receptive field, a process we coin *volume sweeping*, and embed the algorithm in a multi-view depth-map extraction and fusion pipeline followed by a geometric surface reconstruction. With this strategy, we validate that CNN-based MVS outperforms classical MVS approaches in performance capture scenarios. In particular, we obtain high precision geometric results on complex sequences, outperforming both existing CNN-based and classic non-learning methods on a large set of capture datasets. These diverse results are obtained using only a DTU subset as training data, which evidences the generalization capabilities of our network.

This article is an extended version of [14] that provides a complete and self-contained description of the proposed method, with more details about the pipeline from [10] along with the detailed volume sweeping and surface extraction algorithms. Several supplementary experiments were performed to give more insights on the contribution and study the influence of the parameters. We finally challenged the generalization properties of our network on multiple dataset that were not seen during training with competitive results compared to both hand-crafted and learned state of the art.

2 Related Work

Multi-view stereo reconstruction is an active and longstanding vision problem [15]. Stereo and MVS-

based approaches are increasingly being used for high fidelity capture applications [16,17,18,19,11,20,21], possibly complementing other strategies such as depth-based reconstruction [13,22,23,24] by addressing shortcomings that include limited range, sensitivity to high contrast lighting, and interference when increasing the number of viewpoints. While considering various shape representations, for instance point clouds [16], fused depth maps [25], meshes [26,27], or volumetric discretizations [28,29,30], most MVS methods infer 3D shape information by relying on the photoconsistency principle that rays observing the same scene point should convey similar photometric information.

In it simplest form, such similarity can be measured by considering projected color variances among views, as used in early works [28] with limited robustness. In stereo and short baseline situations, simple normalized forms of 2D window correlation are sufficient to characterize similarity under simple lighting and constrast changes, using *e.g.* ZNCC, SSD, SHD. For broader geometric and photometric resilience, various features based on scale-invariant gradient characterizations [31,32,33] have been designed, some specialized for the dense matching required for the MVS problem [34]. More recently, image features have been successfully applied to performance capture sequences in *e.g.* [20,10]. Generally, MVS methods characterize photoconsistency either with a symmetric, viewpoint agnostic, combination of all pairwise similarities [35], or with a per image depth map determination through sweeping strategies [36,25]. The latter sweeping approaches have the advantage of simplifying the scene parametrization of occlusions [37,38], which we leverage for our approach and show to yield a robustness advantage over other strategies in our experiments.

While classic MVS approaches have been generally successful, recent works aimed at learning stereo photoconsistency have underlined that additional priors and more subtle variability co-dependencies are still discoverable in real world data. Several works leverage this by learning how to match 2D patch pairs for short baseline stereo, letting deep networks infer what features are relevant [1,2,3,4]. More recent works extend this principle to short baseline MVS, with symmetric combination of 2D learned features [39], or wide baseline sparse capture scenarios [40,41].

Most of these methods however use a 2D receptive field for stereo matching. The intuition that volumetric 3D receptive fields may be more informative and ease CNN inference and has been explored by some recent works [5,6,7,8], an assertion that the presented approach further verifies. While casting correlations in 3D as well, our approach proposes several key differences. Contrary to the latter, our volumetric receptive field is projective in the camera coordinate frame, similar to some binocular stereo [42] or image-based rendering [43] works. This allows for sweeping along viewing rays, which was proven to be a robust search strategy for binocular stereo plane sweeping [38]. It also enables a per frame approach, with depth estimations, that appears to be more flexible than a global reasoning over all frames. This scheme also avoids decorrelating camera resolution and 3D receptive field resolution, as with *e.g.* voxels, the volumetric receptive field being defined as a backprojection along pixel rays. Additionally, this volumetric receptive field learns local pairwise correlations, a lower level and easier task than learning occupancy grid patterns. Our evaluation substantiates the aforementioned robustness benefits on a number of qualitative (7.3) and quantitative experiments (7.2) with challenging dynamic capture datasets, showing in particular the improvements over 2D receptive fields (7.1).

3 Method Overview

Our main objective is to study multi-view photoconsistency within the context of multi-view stereo reconstructions. We consider for that purpose the reconstruction framework, largely adopted over the last decade, that consists in first estimating per camera depth maps, followed by depth fusion and surface extraction. This framework allows to reason at the pixel level, enabling therefore each camera to provide local details on the observed surface with local estimations. This is in contrast with global strategies that consider photoconsistency at the shape level, with for instance voxels as in [6]. Comparisons between strategies are provided in the experiment section (see section 7.2).

Regarding depth map estimation, we propose to replace the traditional handcrafted photoconsistency measures used to estimate depths with a learned version. This version is based on CNNs and exploits their ability to learn local photometric configurations near surfaces observed from multiple viewpoints. As depicted in Figure 2, our approach takes as input a set of calibrated images and outputs a 3D mesh obtained by fusing depth maps. Depths along pixel viewing rays are obtained using a volume sweeping strategy that samples multi-view photoconsistency along rays and identifies the maxima. For a depth point candidate along a viewing ray, the photoconsistency is estimated using a discretized 3D volumetric projective grid centered on that point. In such a 3D grid, color inputs from the primary camera are paired with color inputs from another camera at each volume element of the grid around the depth point candidate. For a given depth candidate, we

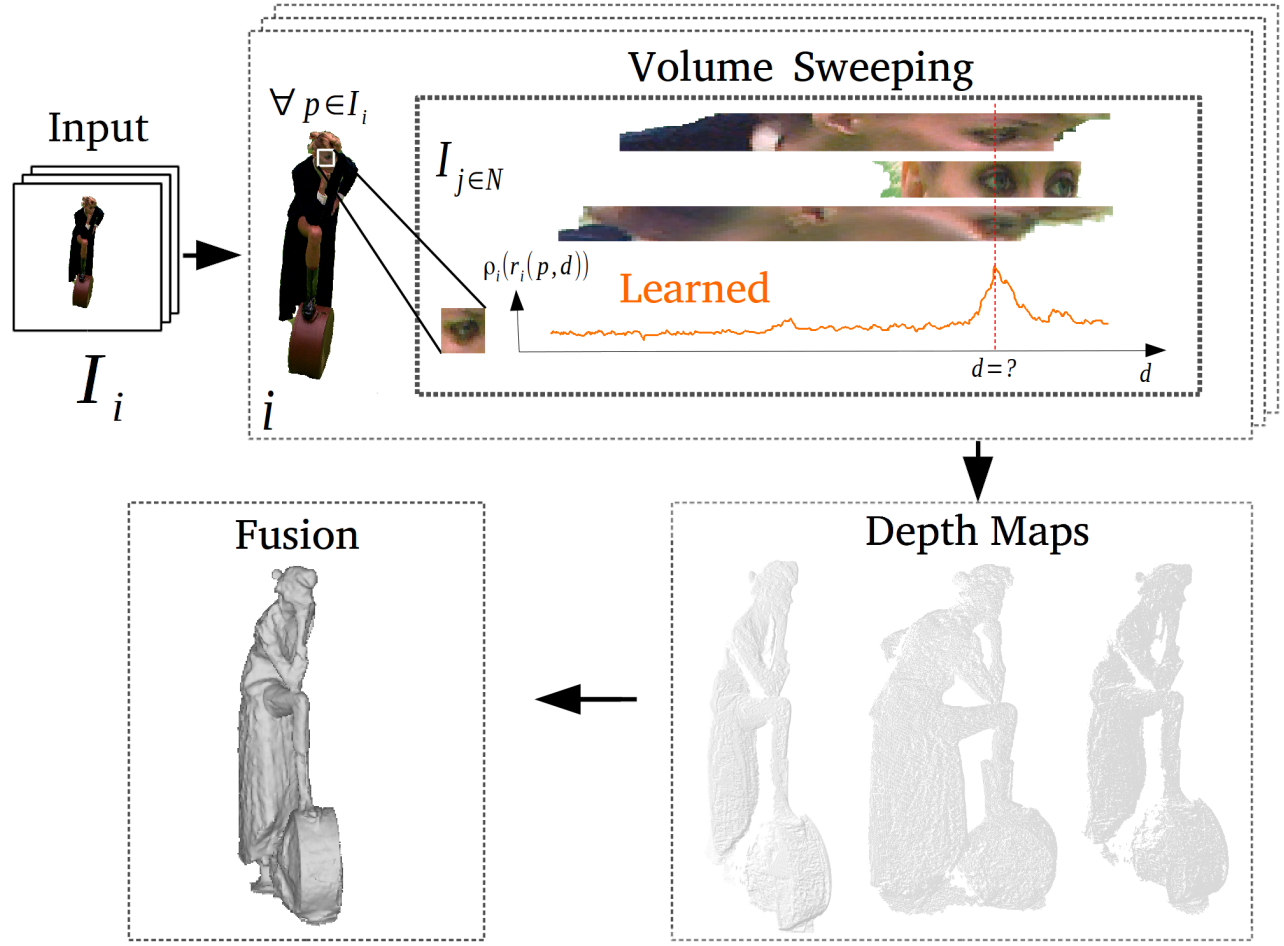

Fig. 2 Method overview. Depth maps, for all input image I_i, are obtained by maximizing, along viewing lines, a learned function that measures photoconsistency at a given depth d along the viewing line of a given pixel p. Depth maps are then fused into an implicit form from which the zero set surface is extracted.

collect all such paired color volume grids for every other camera than the primary. A trained CNN is used to recognize the photoconsistent configurations given pairs of color samples within the 3D grid. The key aspects of this strategy are:

- The per camera approach, which, by construction, samples the photoconsistency at a given location as captured and thus enables more local details to be revealed compared to a global approach, as shown in Figure 17.
- The 3D receptive field for the photoconsistency evaluation, which resolves some 2D projection ambiguities that hindered 2D based strategies.
- The learning based strategy using a convolutional neural network, which outperforms traditional photometric features when evaluating the photoconsistency in dynamic captured scenes, as demonstrated by our experiments.

The following sections focus on our main contributions, namely the 3D volume sampling in section 4.1 and the learning based approach in sections 4.2,4.3 for the photoconsistency evaluation. We then describe in section 5 our depth map evaluation procedure, derived from a *winners-take-all* strategy suitable to our capture scenario. These depth maps are then fused into an implicit form, from which, without loss of generality, we extract the zero-level set using the surface extraction technique described in section 6.

4 Learning Photoconsistency

Our reconstruction approach takes as input N images $\{I_i\}_{i=1}^N$, along with their projection operators $\{\pi_i\}_{i=1}^N$, and computes depth maps, for the input images, which are subsequently fused into a 3D implicit form. This section explains how these maps are estimated.

Given a pixel p in an input image I_i, the problem is therefore to find the depth d at which its viewing ray intersects the observed surface. The point along the ray of pixel p at depth d is noted $r_i(p,d)$. Our approach searches along viewing rays using a likelihood function for a point to be on the surface given the input color pairs in the evaluation volume. In contrast to traditional methods that consider handcrafted photoconsistency measures, we learn this function from multiview datasets with ground-truth surfaces. To this purpose we build a convolutional neural network which, given a reference camera i and a query point $x \in \mathbb{R}^3$, maps a local volume of color pair samples around x to a scalar photoconsistency score $\rho_i(x) \in [0..1]$. The photoconsistency score accounts in practice for color information from camera i at native resolution, and for other camera colors in addition to their relative orientations as implicitly encoded in the volume color pair construction. These important features allow our method to adapt to specific ray incidences. Its voluntarily asymmetric nature also allows subsequent inferences to automatically build visibility decisions, *e.g.* deciding for occlusion when the primary camera i's color is not confirmed by other view's colors. This would not have been possible with a symmetric photoconsistency function such as [39].

We thus cast the photoconsistency estimation as a binary classification problem from these color pairs around the location x, with respect to the reference image I_i and the other images. In the following, we first provide details about the 3D sampling regions before describing the CNN architecture used for the classification and its training. We then explain the volume sweeping strategy that is subsequently applied to find depths along rays.

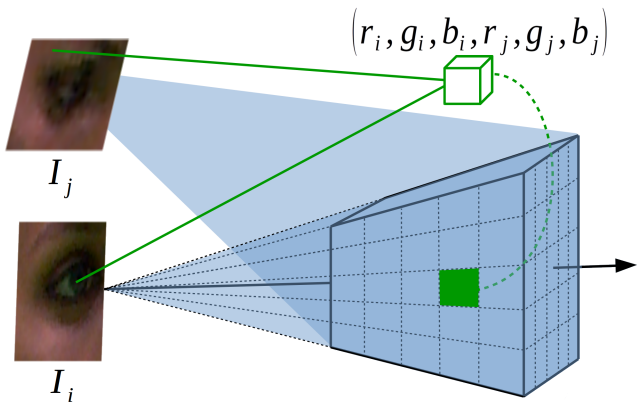

Fig. 3 The 3D volume used to estimate photoconsistency along rays from the reference image I_i. k^3 samples within the volume are regularly distributed along viewing rays and contain color pairs as back-projected from images I_i and I_j. At a given depth along a ray from I_i any image $I_j \neq i$ can define such a pairwise comparison volume.

4.1 Volume Sampling

In order to estimate photoconsistency along a viewing ray, a 3D sampling region is moved along that ray at regular distances. Within this region, pairs of colors backprojected from the images are sampled. Each pair contains a color from the reference image I_i and its corresponding color in another image I_j. Samples within the 3D region are taken at regular depths along viewing rays in the reference image (see Figure 3). The corresponding volume is a truncated pyramid that projects onto a 2D region of constant and given pixel dimension in the reference image. This allows the 3D sampling to adapt to the camera properties, *e.g.* pixel resolution and focal length.

More precisely, we denote $r_i(p,d)$ the 3D location at depth d along the viewing line back-projected from pixel p in the reference image I_i. The k^3 input sample grid used to compare pairs of colors from images $\{i,j\}_{j\neq i}$ is then the set of back-projected rays in a k^2 window centered on p, regularly sampled from depth $d - k\lambda/2$ to $d + k\lambda/2$ around $r_i(p,d)$, with λ chosen such that spacing in the depth direction is equal to the inter-pixel distance from the reference camera at that depth. Every sample contains the reference color of the originating pixel in image I_i and the color of the point projected in image I_j.

Volume sampling is always performed with the same orientation and ordering with respect to the reference camera. Convolutions are thus consistently oriented with respect to the camera depth direction.

Volume Size In practice, we choose $k = 8$. Our strategy is to learn pairwise photoconsistent configurations along rays. This way, decisions for the surface presence are conditioned to the observation viewpoints, which implicitly enforce visibility rules since only one 3D point per ray can be detected. This is in contrast to more global strategies where such per viewpoint visibility is less easy to impose, as with regular voxel grids, *e.g.* [6] with 32^3 or 64^3 grids. In addition, by considering the surface detection problem alone, and letting the subsequent step of fusion integrate depth in a robust and consistent way, we simplify the problem and require little spatial coherence, hence allowing for small grids. We provide a more detailed study of the performances of the classifiers with various depth values in section 7.1.

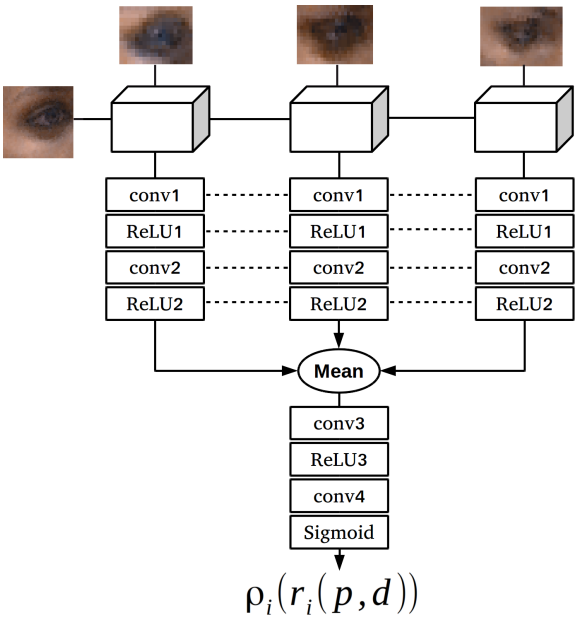

Fig. 4 CNN architecture. Each cube is a pairwise comparison volume with k^3 samples that contain 6 valued vectors of RGB pairs and over which 3D convolutions are applied. The output score $\rho_i(r_i(p,d)) \in [0..1]$ encodes the photoconsistency measure at depth d along the ray from pixel p in image I_i.

4.2 Multi-View Neural Network

As explained in the previous section, at a given point x along a viewing ray from image I_i we can build $N-1$ color volumes with pairs of views $(I_i, I_{j\neq i})$. Each volume is composed of k^3 cells with pairs of RGB values. In order to detect whether the surface is going through x, we use siamese encoders similar in spirit to [39], with however 3D volumes instead of 2D patches. Each encoder considers as input a pairwise color volume and provides a feature. Features from all color volumes at x are then averaged and fed into a final decision layer. Weight sharing and averaging are chosen to achieve camera order invariance.

The network is depicted in Figure 4. The inputs are the $N-1$ color volumes of size $k^3 \times 6$ where RGB pairs are concatenated at each sample within the volume. Convolutions are performed in 3D over the 6 valued vectors of RGB pairs. The first layers (encoders) of the network process every volume in parallel, with shared weights. Every encoder is a sequence of two convolutions followed by non-linearities, and max-pooling with stride. Both convolutional layers consist of respectively 16 and 32 filters of kernel $4 \times 4 \times 4$, followed by a Rectified Linear Unit (ReLU) and a max-pooling with kernel $2 \times 2 \times 2$ with stride 2. We then average the obtained $2 \times 2 \times 2 \times 32$ features and feed the result to a 128 filter $1 \times 1 \times 1$ convolutional layer, followed by a ReLU and a final $1 \times 1 \times 1$ decision layer, for a total of $72K$ parameters . The network provides a score $\rho_i(r_i(p,d)) \in [0..1]$ for the photoconsistency at depth d along the ray from pixel p in image I_i.

We experimented this network using different configurations. In particular, instead of averaging pairwise comparison features, we tried max-pooling which did not yield better results. Compared to the volumetric solution proposed by [6], the number of parameters is an order of magnitude less. As mentioned earlier, we believe that photoconsistency is a local property that requires less spatial coherence than shape properties.

4.3 Network Training

The network was implemented using TensorFlow [44] and trained from scratch using the DTU Robot Image Dataset [11], which provides multiview data equipped with *ground-truth* surfaces that present an accuracy of $0.5mm$. From this dataset 11 million k^3 color volumes were generated, from which we randomly chose 80 percent for training, and the remaining part for evaluation. Both positive and negative samples were equally generated by randomly sampling volumes up to $20cm$ away from ground truth points, where a volume is considered as positive when it contains at least μ ground truth points. In theory, the network could be trained with any number of camera pairs, however, in practice, we randomly choose from one up to 40 pairs. Training was performed with the binary cross entropy function as loss. Model weights are optimized by performing a Stochastic Gradient Descent, using Adaptive Moment Estimation [45] on $560,000$ iterations with batch size of 50 comparisons, and with a random number of compared cameras (from 2 up to 40). Since our sampling grids are relatively small and camera dependent, we are able to generate enough sample variability for training, without the need for data augmentation.

5 Depth Estimation

As previously noted, our main motivation is to reconstruct live dynamic scenes, typically humans in motion. In such cases, it is advantageous to focus on the foreground objects in the observed scene rather than modeling the full scene. To this purpose, we limit the search domain for depths along viewing rays to a region defined by image silhouettes. In the following we explain how such a region is defined and we detail then the volume sweeping we adopt to identify image depths.

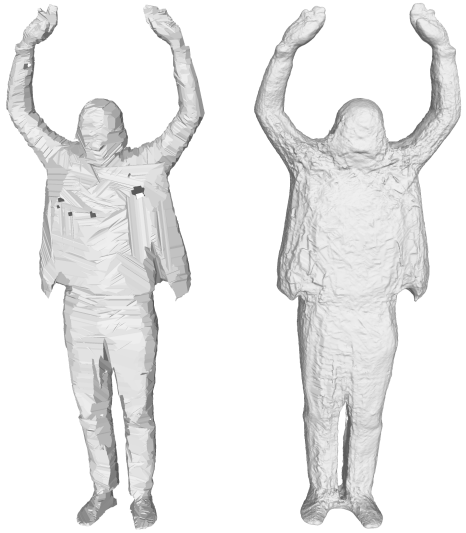

Fig. 5 Left: the Confidence Volume with $\alpha = \beta = 54$, equivalent to the Visual hull with the 54 cameras that see the subject; Right: the Confidence Volume with $\alpha = \beta = 10$.

5.1 Confidence Volume

We assume we are given a set of N images $\{I_i\}_{i=1}^N$ observed with a set C of calibrated cameras with known projections $\{\pi_i\}_{i=1}^N$ and centers $\{c_i\}_{i=1}^N$. We assume we are also given a set of silhouettes $\{\Omega_i\}_{i=1}^N$, often available in multi-view scenarios by performing image segmentation, for instance background subtraction with constrained capture environments. The silhouettes are generally imprecise, as a result of multiple causes, including color ambiguities, that plague the segmentation. However, the redundant information they provide over several viewpoints can be used to restrict the search domains along viewing rays to segments that are likely to intersect the object surface. To this purpose we define the confidence volume V as:

$$V = \{x \in \mathbb{R}^3 : \exists^{>\alpha} i \, (\pi_i(x) \in I_i) \\ \wedge \exists^{>\beta} i \, (\pi_i(x) \in \Omega_i)\}, \tag{1}$$

as the locus of points in $\mathbb{R}^3$ which project in $i > \alpha$ images and $\beta \leq i$ silhouettes to which they belong. When $\beta = i$, V is simply the visual hull with i images. α, β are two user defined constants that restrict weakly supported depth predictions with α and enable predictions away from the exact visual hull when $\beta < \alpha$. Intuitively, V is a dilated version of the visual hull in the space region seen by at least α images, as shown in fig 5. As explained in the following section, the intersection of a viewing ray with V defines the starting point of the depth search interval along that ray.

5.2 Volume Sweeping

In order to estimate pixel depths, the sampling volume introduced in section 4.1 is swept along their viewing rays while computing multi-view photoconsistency using the network detailed in section 4.2. For every camera, we sample therefore along viewing rays, test possible depth values, and choose the best candidate with respect to the network score. In practice, a reference view I_i is only compared to the other views I_j such that $\cos(\theta_{ij}) > 0.5$, where θ_{ij} is the angle between the optical axes of camera i and j. Then, we sample rays from camera i through every pixel p and build colored volumes at every candidate depth, starting at the intersection with the confidence volume introduced in the previous section. Once the probability of surface presence is computed for every candidate, we define the estimated depth d_i as:

$$d_i = \underset{d \in [d_{min}, d_{max}]}{\operatorname{argmax}} \; (\rho_i(r_i(p, d))), \tag{2}$$

where $\rho_i(r_i(p, d))$ is the consistency measure along the ray from p in image I_i, as estimated by the network. $[d_{min}, d_{max}]$ is the search range with: $d_{min} = d_V(p)$ the intersection of the viewing ray at p with the confidence volume; d_{max} such that the search is stopped when the accumulated photoconsistency score reaches a given value ρ_{max}, in a *winner-takes-all* surface detection strategy.

$$\int_{x=d_{min}}^{d_{max}} \rho_i(r_i(p, x)) \mathrm{d}x \leq \rho_{max} \tag{3}$$

Depths for all pixels and from all images are further fused using a truncated signed distance function (TSDF) [46]. The following section explains how we define and extract the zero level-set of the TSDF.

6 Surface Extraction

We explained in the previous section how to compute depth maps for every viewpoint. We now have to fuse them into an implicit form, namely the TSDF [46] from which we can extract the zero-level set that corresponds to the reconstructed surface, which appears in *black* in Figure 6. Contrary to previous works [47, 24, 13, 22], we do not store TSDF values in a regular voxel grid but we rather devise a simple yet efficient sampling procedure derived from Voronoï Tesselation strategies, that specifically accommodates multi-view capture scenarios. It is worth mentioning that other works such as [27] also make use of irregular sampling strategies for MVS, but in a volumetric graph-cut framework.

6.1 Implicit Form Definition

For a point $x \in \mathbb{R}^3$, the truncated signed distance $TD(x) \in \mathbb{R}$ to the surface is defined as the weighted average of all camera contributions $F_i(x), i \in C$:

$$F_i(x) = \begin{cases} \min(\mu, \eta(x)) & iif\ \eta(x) \geq -\mu, \\ \emptyset & \text{otherwise}, \end{cases} \\ \eta(x) = d_i(\pi_i(x)) - \|c_i - x\|, \tag{4}$$

and:

$$TD(x) = \frac{\sum\limits_{i \in C_x} \rho_i(x) F_i(x)}{\sum\limits_{i \in C_x} \rho_i(x)}, \tag{5}$$

where $C_x = \{i \in C : F_i(x) \neq \emptyset\}$ and $\rho'_i(x)$ the photo-consistency measure (4) of the estimated depth along the ray from camera i passing through x. If d_i is undefined at x, *e.g.* x is outside the camera visibility domain, then camera i does not contribute to the TSDF. When no camera contributes at x but x is inside the confidence volume V then it is considered as inside, *i.e.* $TD(x) < 0$. Note that contributions are weighted by the normalized photoconsistency measure which means that when cameras disagree about the photoconsistency at x, cameras with higher measures have an increased impact whereas cameras with low detection probability measures only marginally impact the reconstruction.

6.2 Extraction Procedure

From the previously defined TSDF, we extract the surface using a sampling strategy based on ray casting and Voronoï Tessellation. Figure 6 provides a 2D example of the main steps of the algorithm that are as follows:

1. (*orange*) Sample points inside the implicit form defined by the TSDF. This is achieved by randomly selecting pixels in all images and computing the point, along each pixel rays, inside but close to the surface according to the TSDF. The process is iterated until a user defined number of 3D points is reached.
2. (*blue*) Determine the Voronoï diagram: given the points inside the shape surface, a Voronoï diagram of this set of points is computed.
3. (*green*) Clip the Voronoï diagram with the zero level set of the TSDF. This operation extracts the intersection of the Voronoï cells with the surface to form an oriented mesh.

In the above strategy, sampling points close to the surface, and originating from image viewpoints, ensures that the 3D discretization is denser on the surface than inside the volume and also denser on surface regions observed by more images. The latter enables more precision to be given to surface regions for which more image observations are available.

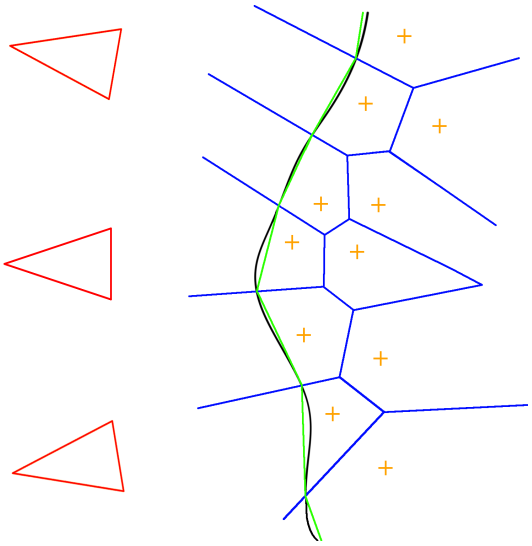

Fig. 6 Our surface extraction procedure. The zero-level set of the implicit form (**black**) is observed by different cameras (**red**). They are used to provide the inside samples (**orange**) that will be used as the centroids for the Voronoï tessellation. This tessellation is finally clipped at the zero-level set and the final surface (**green**) can be extracted.

We visualize in figure 7 an example of extracted surface. We show 2 of the 40 input views in the top row, and our reconstruction in the middle. The bottom side of the bust is never seen by any camera. We show in the bottom row the difference in sampling resulting from the observation of the shape. The horizontal bottom side of the model is never observed, yet still correctly reconstructed. On the other hand, the triangles of the mesh in that area are much larger than the ones in the vertical upper part, which is observed by more cameras. This strategy allows for complete reconstructions of captured shapes with an adaptive sampling density depending on the observations of the object, focusing more samples in the regions where the details can be recovered.

Runtime The full pipeline allows us to reconstruct one time frame, *i.e.* 68 images (2048×2048), in approximately 30 to 40 minutes using two NVIDIA Titan X GPUs, depending on the number of pixels that observe the shapes.

7 Experiments

Our main goals in this section are (i) to evaluate whether and how our learned photoconsistency contributes with respect to existing methods and (ii) to ver-

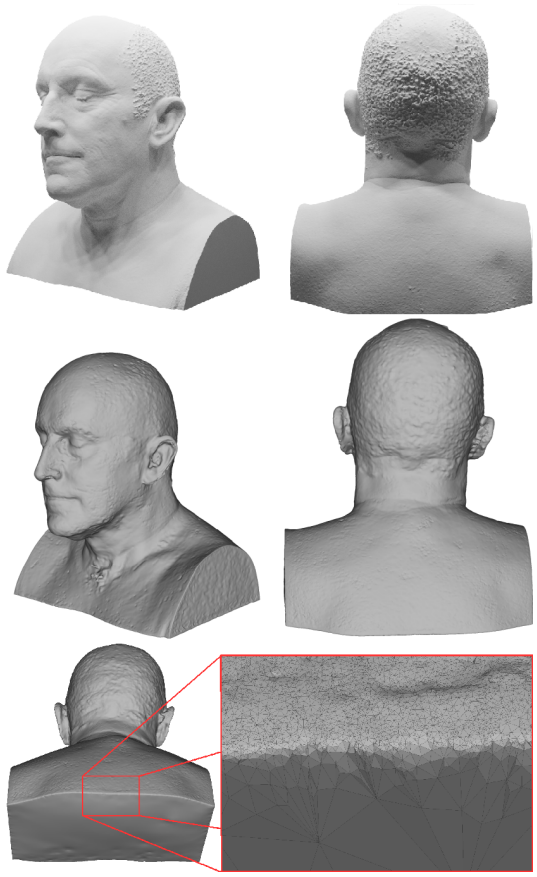

Fig. 7 Two points of view of a synthetic model (*top*) and the result of our reconstruction (*middle*). A close-up of the extracted surface (*bottom*) at the limit between well-observed and unseen regions. The top part of the close-up is seen by many cameras whereas the bottom part is never observed.

ify whether these transfer to the more complex case of generic 3D capture scenes in practice, *e.g.* humans with complex clothing. To this aim, we perform various evaluations to verify and quantify the benefit of our learned multi-view similarity. We start by providing multiple validation experiments to justify the choices for the learning and reconstruction strategies in 7.1. Second, for comparison purposes, we apply in 7.2 our depth estimation approach in the static case using the [11] benchmark and compare it to state of the art MVS methods, both handcrafted and learning based. We make use of the standard *accuracy* and *completeness* metrics, both averaged and median, for which the evaluation code is provided by the authors.

Finally, we build experiments to test the main claim of improvement with production capture data in 7.3. To this goal we use several dynamic sequences captured on different platforms, which exhibit typical difficulties of such data. In particular, we mainly focus on the Kinovis acquisition platform [9], which consists of 68 RGB cameras, of resolution 2048×2048 with focal lengths varying from $8mm$ to $25mm$. We achieve very significant qualitative improvements compared to the state of the art approaches both learning-based [6,8], and handcrafted [10], without fine-tunning and despite the difference of capture setup used for training. We also compare to [23] on an example provided by the authors and achieve slightly better quality using only half the available information.

7.1 Validation

We previously formulated the problem of surface detection along viewing rays as a binary classification problem, as explained in section 4. In order to assess the benefit of our volumetric strategy, we first focus on different classifiers performances. We provide in 7.1.1 receptive field comparisons on the training dataset, this to enhance the advantage of casting and learning correlations in 3D. Additionally, section 7.1.2 provides a study of the depth hyperparameter of the receptive field of our network. Then, since preliminary results of [14] seemed to show a better robustness to a larger baseline, we design an experiment with cameras that are further apart to better quantify this improvement in section 7.1.3. We finally provide in 7.1.4 an ablation study of the accumulator described in 3 to validate its importance in the depth estimation procedure, in the performance capture scenario.

Section 7.1.2 shows that a volume size of $8 \times 8 \times 8$ is a preferred trade-off, thus will be used from now on, when not specified.

7.1.1 Classifiers Study

In this paragraph, we compare performances of different classifiers based on various receptive fields:

1. Zero-Mean Normalized Cross Correlation (*ZNCC*): ZNCC is applied over the samples within the volumetric support region.
2. Learning (CNN) with a planar support: a planar equivalent of our volumetric solution, with the same architecture and number of weights, in a fronto-facing plane sweeping fashion.
3. Learning (CNN) with a volumetric support: our solution described in the previous sections.

Figure 8 shows, with the classifiers' ROC curves, that the most accurate results are obtained with a volumetric support and learning. Intuitively, a volumetric sampling region better accounts for the local non-planar geometry of the surface than planar sampling regions.

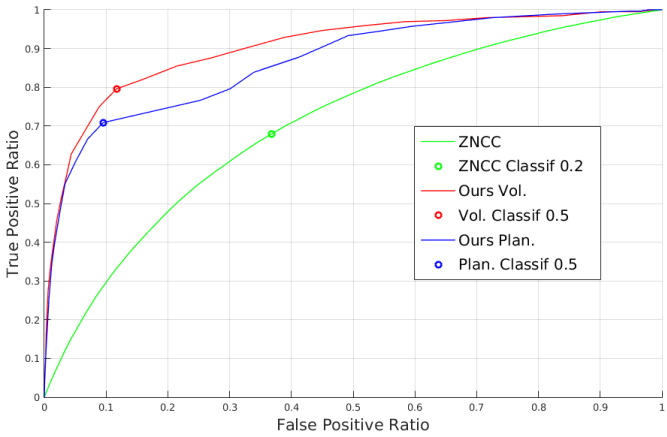

Fig. 8 ROC Curves of three different classifiers, ZNCC, planar and volumetric supports, on the DTU Dataset [11]. Circles represent thresholds that optimize sensitivity + specificity with the values 0.2, 0.5 and 0.5 respectively.

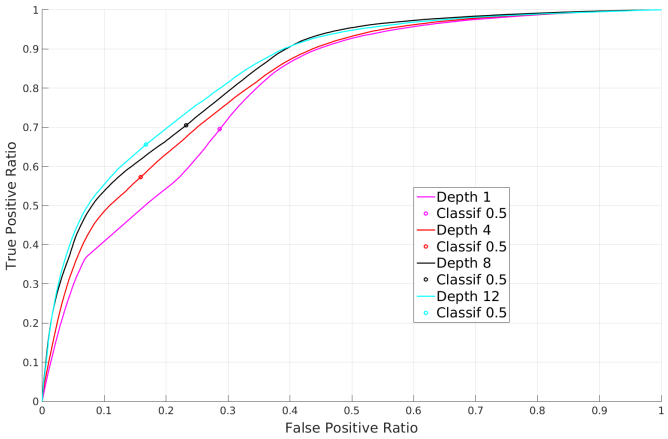

Fig. 9 ROC Curves of four different classifiers using 8×8 receptive fields with various depths. Circles represent thresholds that optimize sensitivity + specificity.

This graph also emphasizes the significantly higher discriminative ability of learned correlations compared to deterministic ones.

7.1.2 Volume Sampling

To further demonstrate this, we then proceed to a study on the impact of the depth parameter of the sampling volume. While keeping a 8×8 pixels reprojection on the images, we study the performances on classifiers with receptive fields varying in depth. Figure 9 shows classifiers performances with depth values ranging from 1 to 12. To perform this experiment, we had to diminish the networks number of parameters to fit the 12 depth training in memory and keep reasonable training and testing times, explaining the worse performances compared to previous ROC curves. This experiment demonstrates that the more information the network gathers along the ray the better the detection of the surface is. We choose a depth of 8 as it gives the best trade-off between computational complexity and performance.

7.1.3 Baseline Study

We now evaluate the robustness to various baselines by accounting for a higher number of cameras and more distant cameras in the classification. Table 1 shows the accuracy of the classifiers with a varying number of cameras and for the optimal threshold values in Figure 8. As already noticed in the literature, *e.g.* [16, 19], a planar receptive field gives better results with a narrow baseline and the accuracy consistently decreases when the inter-camera space grows with additional cameras. In contrast the classifier based on a volumetric support exhibits more robustness to the variety in the camera baselines. This appears to be an advantage with large multi-camera setup as it enables more cameras to contribute and reduces hence occlusion issues.

Camera #	5	20	49
ZNCC	64.98	65.46	65.58
Ours Plan.	80.67	77.87	75.92
Ours Vol.	**82.95**	**84.84**	**83.45**

Table 1 Classifier accuracy (%).

To push this experiment further, we design an experiment to test the robustness of our approach on a sparse capture platform, with lower scene coverage and wider baseline. Since no ground truth exists for this kind of performance capture scenario, we simulate it using of a realistic rendering engine to create a synthetic dataset. Similar to [9] in terms of camera parameters and capture volume, we chose to render only 10 randomly placed cameras, evenly distributed on an hemisphere around the capture volume. The average spacing between a camera and its 10 closest neighbors is $8.03m$ in this case, where it is $2.5m$ for the 68 POV kinovis platform and $0.188m$ in the 49 POV DTU case. For this experiment, we set the neighboring camera acceptation threshold $\cos(\theta_{ij})$ to 0.1, meaning that we accept almost orthogonal cameras. The synthetic cameras render the scene using Filmic Blender [48], a photorealistic configuration for Blenders Cycles ray-tracing engine. The images are generated with random parameters, *i.e.* the cameras parameters vary, in terms of position, orientation, focal length, and pixels number of samples, the latter directly affecting sensor noise. With this platform, we rendered a dozen of models such as procedu-

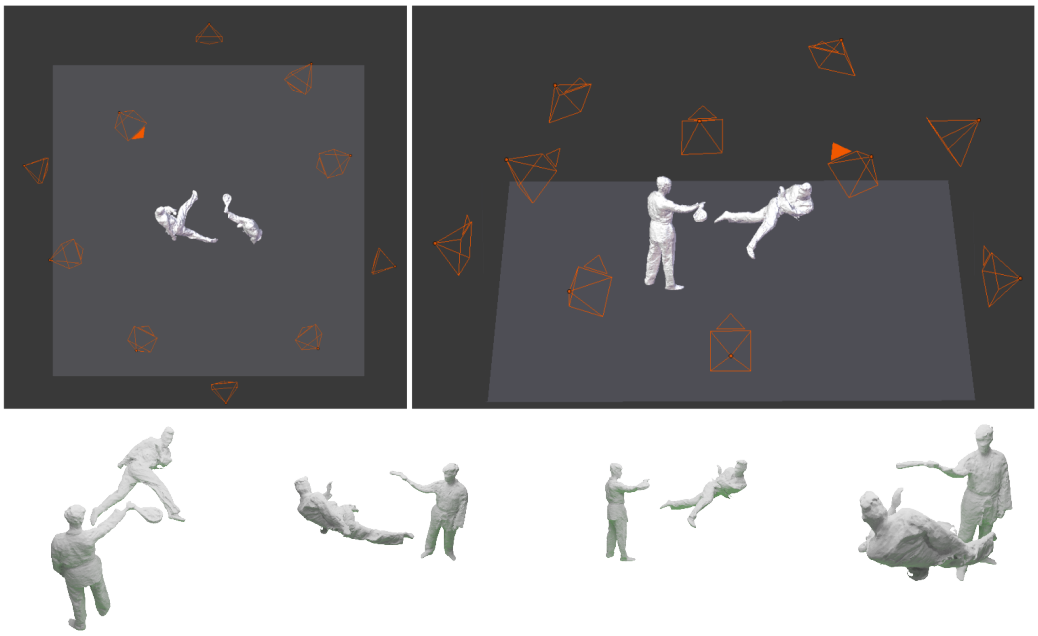

Fig. 10 An example of sparse synthetic performance capture data generation. *(top)* Top and side view of the 10 cameras positioned around a surface. *(bottom)* Four examples of generated points of view.

rally generated geometric shapes, real life reconstructions or CAD models with various appearances. The multiview networks are trained from scratch on these synthetic examples, and evaluated on unseen synthetic data. Figure 10 shows an example of our synthetic platform as well as the generated synthetic data. We show in figure 11 the impact of a volumetric support: when the baseline between the cameras becomes extreme, it offers more robustness compared to a planar support, which appears very slanted in the compared view. Even though it is only a synthetic dataset, we believe that it gives interesting insights on the versatility of our volume sweeping strategy for the performance capture scenario. A qualitative result of this improved robustness is shown in figure 12. The area of the face is highly occluded, and the volumetric support helps recovering a smoother surface. Also note the details of the belt: the volume allows a sharp reconstruction of finer details, where a plane cannot handle finer geometry details.

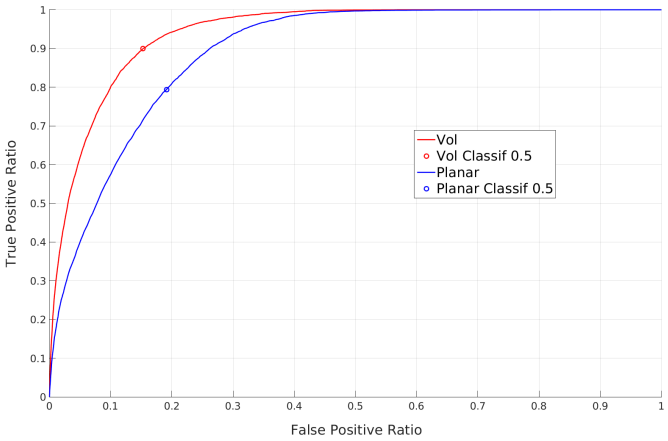

Fig. 11 ROC Curves of two different classifiers using planar and volumetric receptive fields, on the sparse synthetic data. Circles represent thresholds that optimize sensitivity + specificity.

7.1.4 Accumulation Term

We now provide a qualitative experiment to justify the use of the accumulation term in equation 3 in figure 13. This figure demonstrates the importance of the accumulation scheme in the performance capture scenario. The noisy photoconsistency in this case leads to a lot of false positives, creating extreme holes in the reconstructions when not using the accumulation scheme, *i.e.* $\rho_{max} \rightarrow \infty$. The addition of this term ($\rho_{max} = 1.6$) allows for smooth and faithful reconstructions, still containing most of the important geometric details.

7.2 Quantitative Comparisons

In this section, we compare our solution to various state-of-the-art methods using the DTU Robot Image Dataset [11]. We use the standard accuracy and completeness metrics to quantify the quality of the esti-

Fig. 12 (*Left*) 3 input images, (*middle*) plane based classifier, (*right*) volumetric classifier. The face is highly occluded in many views (*left*) making the reconstruction noisy and inaccurate when using a planar support whereas the volume counterpart yields smoother and more accurate details.

Fig. 13 (*top row*) Input images of captured subjects. (*middle row*) Reconstructions without probability accumulation along rays. (*bottom row*) Results with accumulation.

mated surface as described in [49], that is we define *accuracy* for a point of the reconstructed shape as the smallest Euclidean distance to the *ground-truth*, and the *completeness* of a point of the *ground-truth* as the smallest Euclidean distance to the reconstructed shape. For both metrics, we compare the average and median values over all the points of the shapes. To diminish the impact of far outliers in the metrics, we make use of the default thresholding parameter of [49]. We compare to Furukawa et al. [16], Campbell et al. [50] and Tola et al. [34], that are well-known handcrafted strategies, as well as to additional learning-based results from Ji et al. [6] and Hartmann et al. [39]. To conduct a fair comparison with [39], that is a patch based approach building a depthmap with a network comparable to ours, we use the result of our volume sweeping approach on the same depth map. When performing reconstructions on the DTU, we did not use the accumulation scheme in 3, *i.e.* $\rho_{max} \rightarrow \infty$. To speed up computations, we limit the search along a viewing ray to $5mm$ around a coarse depth estimation based on image descriptors [51]. Depths are sampled every $0.5mm$. As a post processing step, we simply add a soft bilateral filter, similarly to [39], accounting for color, spatial neighborhood, and probability of the detection.

Reconstruction results are depicted in table 2. We achieve quality on par with the best performing methods on this dataset, with a median accuracy and completeness in the range of the ground truth accuracy that we measured around $0.5mm$. It should be noticed that the best accuracy is obtained by Tola et al. [49] which tend to favor accuracy against completeness whereas Campbell et al. [50], in a symmetric manner, tend to favor completeness against accuracy. We obtain more balanced results on the 2 criteria, similarly to the widely used approach by Furukawa et al. [16], with however better performances. We also outperform the recent learning based method Surfacenet [6] on most measures in this experiment.

Compared to Hartmann et al. [39], and under similar experimental conditions, our approach give better results with 2 orders of magnitude less parameters, thereby confirming the benefit of volumetric supports over planar ones. Compared to Surfacenet. [6] (cube size $64 \times 64 \times 64$, sample step $0.4mm$) we obtain reconstructions of slightly better quality with an order of magnitude less parameters.

7.3 Qualitative Evaluation and Generalization

One of our main goals is to verify whether a learning based strategy generalizes to real life dynamic data and how it compares to state-of-the-art approaches in this

Measure	Acc.		Compl.	
	Mean	Med.	Mean	Med.
Tola et al. [49]	**0.448**	**0.205**	0.754	0.425
Furukawa et al. [16]	0.678	0.325	0.597	0.375
Campbell et al. [50]	1.286	0.532	**0.279**	**0.155**
Ji et al. [6]	0.530	0.260	0.892	0.254
Ours (*fused*)	0.490	0.220	0.532	0.296
Hartmann et al. [39]	1.563	0.496	1.540	0.710
Ours (*depthmap*)	**0.599**	**0.272**	**1.037**	**0.387**

Table 2 Reconstruction accuracy and completeness.

case. To this purpose, we focus our qualitative evaluation on two different dynamic capture datasets, both drastically different from the training one. We first perform, in section 7.3.1, reconstructions of dynamic RGB sequences captured by the Kinovis platform [9]. We then test, in section 7.3.2, our reconstruction method on a different real life dynamic dataset, captured with the active setup of [23] and compare to their results. It is important to note that the network previously trained on the DTU Dataset [11] was kept as such without any fine tuning at all times in this section.

7.3.1 Kinovis Data

We first focus on data captured by [9], that is a hemispherical setup with 68 cameras of various focal lengths. In this scenario, standard MVS assumptions are often violated, *e.g.* wide baseline, specular surfaces, motion blur and occlusions, challenging therefore the reconstruction methods. A video demonstrating our results and providing comparisons on dynamic sequences is available online: `https://hal.archives-ouvertes.fr/hal-01849286`.

Most general purpose MVS methods we tested tend to fail in the performance capture scenario, either providing incomplete or low resolution results, or being extremely noisy. Figure 16 illustrates the reconstruction obtained using COLMAP [52], which is a handcrafted general purpose MVS pipeline based on *PatchMatch Stereo* [53]. Both methods perform overall correctly, as seen on the left side of the figure. However, [52] (*top-right*) struggles to recover fine-grained details while keeping the noise and artifacts level low, contrary to our approach (*bottom-right*). This results demonstrates the benefit of a dedicated method in the context of performance capture.

In order to assess the performances of our learned photoconsistency term, we compared in figure 1 to [10], which is a patch based sweeping method using traditional image features and specifically designed for this scenario. Both methods share a significant part of their pipeline, except for the photoconsistency evaluation, thus providing good insights about the benefits of the

Fig. 14 (*top*) Input images, (*middle*) result with [10], (*bottom*) result with our method. Motion blur and low contrast are visible in the input images . Best viewed magnified.

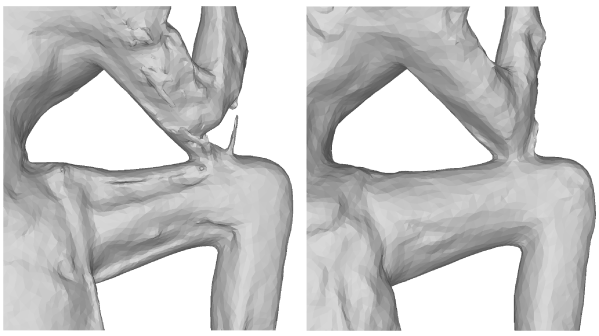

Fig. 15 Close up view of the arm region in Figure 1. (*left*) Results from [10], (*right*) our reconstruction

Fig. 16 (*top*) Reconstruction using COLMAP [52], (*bottom*) our result.

proposed learned term. Even though [10] performs well in contrasted regions, the patch based descriptors reach their limits in image regions with low contrast or low resolution. Figure 15 and 14 give such examples. They show that our strategy helps recovering finer surface details, while strongly decreasing noise in low contrast regions. The results obtained also demonstrate strong improvements in surface details, such as dress folds, that were undetected by the deterministic approach. In addition, they demonstrate lower levels of noise, particularly in self-occluded regions, and more robustness to motion blur as with the toes or tongue-in-cheek details that appear in Figure 14-bottom.

We then compare to the recent learning based approach [6] using the code available online (see Figure 17). Reconstructions with this approach were limited to a tight bounding box and different values for the volume sampling step were tested. The best results were obtained with a $2mm$ step. To conduct a fair comparison with our method, all points falling outside the visual hull were removed from the reconstruction. In this scenario, the point cloud obtained using [6] appeared to be very noisy and incomplete (see Figure 17-middle), plaguing the subsequent surface extraction step. Fig-

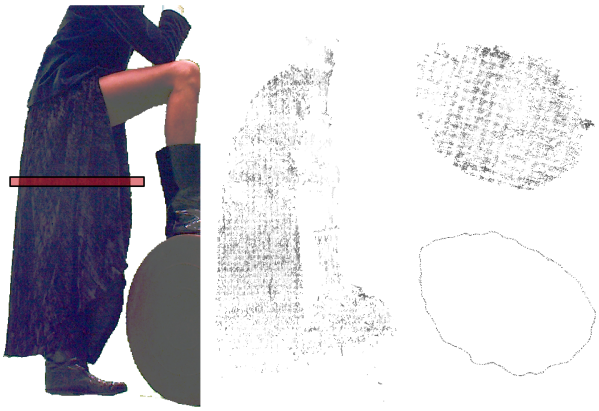

Fig. 17 Qualitative comparison with [6]. (*Left*) input image with the horizontal section in *red*, (*middle*) point cloud with [6], (*right-top*) point cloud horizontal section with [6] (*right-bottom*) point cloud horizontal section with our approach.

ure 17-left also shows a horizontal section of the model in a poorly contrasted image region of the dress. The global strategy used in [6] wrongly reconstruct many surface points inside the shape volume (top figure), as a result of the ambiguous appearance of the dress. In contrast, our approach (bottom figure) correctly identify surface points by maximizing learned correlations along viewing rays.

In addition to this, we also compare to results of [8] provided by the authors in Figure 18. This method outputs a rather dense colored point cloud but similarly to results from [6], extracting a smooth surface from this point cloud remains a difficult task due to strong noise and missing data. Since the method uses custom and undocumented calibration parameters, it was not straightforward to remove points lying outside the visual hull. Moreover, the precision of the point cloud from [8] restricts its usage for performance capture and realistic reconstructions rendering. Figure 19 provides a close-up of the face of a subject. The level of detail of the result from [8] is not fine enough to correctly capture facial details, compared to the density of our output surface.

Fig. 18 (*top*) Results provided by [8] on the kick 540 sequence. (*middle*) Poisson Reconstruction of the output point cloud of [8]. (*bottom*) Our result.

7.3.2 Active Capture Platform

Finally, we compare our reconstructions of a scene captured with results from the active system of [23]. This setup consists of 52 RGB cameras mounted as stereo pairs but also differs from the previous dynamic capture scenario, as it also features an active system, projecting random infrared dots on the shape. 52 infrared cameras, also paired on stereo rigs then capture the reprojected spots on the shape, resulting in highly contrasted images, allowing to disambiguate the photoconsistency computation, especially in textureless regions without interfering with the visible appearance of the subject. In figure 20, we compare to results provided by the authors. While [23] make use of all the data available, we restrict our method to work with RGB images only. On the other hand, we allow cameras that are far apart to participate in the computation of the photoconsistency. Our results demonstrate the quality of our method's results, showing detailed reconstructions competitive with the results of [23] even though we only use the passive system, *i.e.* half of the available information. Figure 21 displays a close-up of the face of the subject. Our method allows to recover high-frequency facial details, such as the shape of the nostrils or the lips commissures, thus providing highly faithful reconstructions.

Fig. 19 Point clouds density comparison between results provided by [8] (*left*) and our output (*right*). Best viewed magnified.

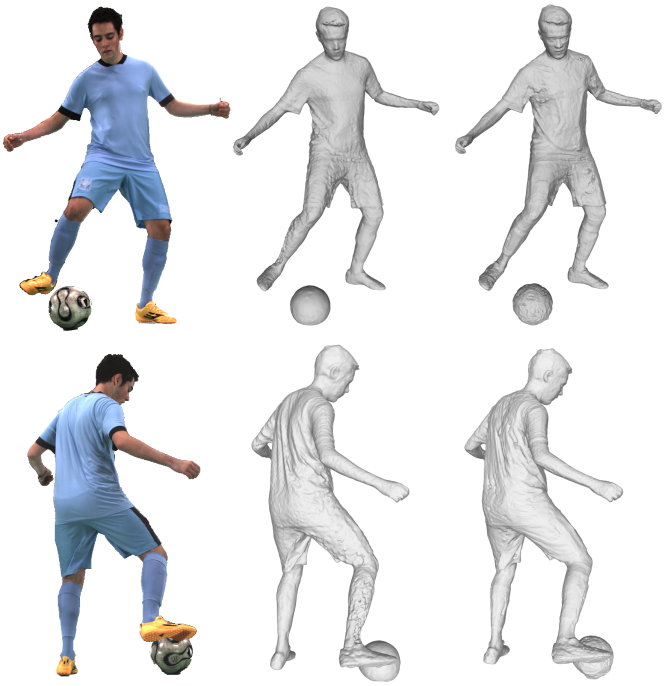

Fig. 20 Two points of view of a subject from [23] (*left*). (*middle*) Reconstruction provided by the authors. (*right*) Results using our learning strategy.

Fig. 21 Close up of the face of the subject from [23] (*left*). The reconstruction provided by the authors (*middle*) is very smooth compared to our result (*right*).

8 Conclusion and Future Works

We presented a learning framework for surface reconstruction in passive multi-view scenarios. Our solution consists in a N-view volume sweeping, trained on static scenes from a small scale dataset equipped with ground truth. Thanks to this new model, we validate the improvement of CNN-learned MVS photoconsistency in the case of complex and dynamic performance capture, with significant challenges typical of these datasets such as low light areas and low texture content and perceived resolution. This result is achieved with an order of magnitude less training parameters than previous comparable learned MVS works, showing significant network generalization from a training performed only on static DTU inputs, fully leveraging the high quality ground truth now available with these datasets. Thanks to our local strategy, our method achieved significantly improved detail recovery and noise reduction in complex real life scenarios, outperforming all existing approaches in this case.

The discretization of the volume around a query point involves a lot of redundancy and is a computationally expensive step for both training and inference. Moreover, even when optimized to process several neighboring depths in parallel, it remains rather memory inefficient. A possible future work could be to find a continuous representation for colored rays crossing the volume of interest, that could be used to infer surface presence probability in a similar manner with a much lighter computational cost.

Finally, we believe our approach is a first step towards a data-driven method to unify shape from silhouette and multi-view stereo inference, as made possible by the wide baseline robustness and general volumetric receptive field of our network, with the prospect of increased automation and quality.

Acknowledgements

This work was conducted at the INRIA Grenoble. Funded by France National Research grant ANR-14-CE24-0030 ACHMOV. Images 1 - 2 - 15 - 17 of Anja Rubik courtesy of Ezra Petronio and Self Service Magazine. Geometric model in figure 7 courtesy of 3DScanStore [54].

References

1. Žbontar, J., LeCun, Y.: Stereo matching by training a convolutional neural network to compare image patches. J. Mach. Learn. Res. **17**(1) (January 2016)

2. Luo, W., Schwing, A.G., Urtasun, R.: Efficient deep learning for stereo matching. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016. (2016) 5695–5703
3. Zagoruyko, S., Komodakis, N.: Learning to compare image patches via convolutional neural networks. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015. (2015)
4. Ummenhofer, B., Zhou, H., Uhrig, J., Mayer, N., Ilg, E., Dosovitskiy, A., Brox, T.: Demon: Depth and motion network for learning monocular stereo. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017. (2017) 5622–5631
5. Choy, C.B., Xu, D., Gwak, J., Chen, K., Savarese, S.: 3d-r2n2: A unified approach for single and multi-view 3d object reconstruction. In: ECCV. (2016)
6. Ji, M., Gall, J., Zheng, H., Liu, Y., Fang, L.: Surfacenet: An end-to-end 3d neural network for multiview stereopsis. In: The IEEE International Conference on Computer Vision (ICCV). (2017)
7. Kar, A., Häne, C., Malik, J.: Learning a multi-view stereo machine. In: Proc. Neural Information Processing Systems (NIPS). (2017)
8. Yao, Y., Luo, Z., Li, S., Fang, T., Quan, L.: Mvsnet: Depth inference for unstructured multi-view stereo. ECCV (2018)
9. : Kinovis inria platform. `https://kinovis.inria.fr/inria-platform/`
10. Leroy, V., Franco, J.S., Boyer, E.: Multi-View Dynamic Shape Refinement Using Local Temporal Integration. In: IEEE, International Conference on Computer Vision 2017, Venice, Italy (2017)
11. Jensen, R.R., Dahl, A.L., Vogiatzis, G., Tola, E., Aanæs, H.: Large scale multi-view stereopsis evaluation. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23-28, 2014. (2014) 406–413
12. Chang, A.X., Funkhouser, T., Guibas, L., Hanrahan, P., Huang, Q., Li, Z., Savarese, S., Savva, M., Song, S., Su, H., Xiao, J., Yi, L., Yu, F.: ShapeNet: An Information-Rich 3D Model Repository. Technical Report arXiv:1512.03012 [cs.GR], Stanford University — Princeton University — Toyota Technological Institute at Chicago (2015)
13. Newcombe, R.A., Fox, D., Seitz, S.M.: Dynamicfusion: Reconstruction and tracking of non-rigid scenes in real-time. In: IEEE Conference on Computer Vision and Pattern Recognition, (CVPR) 2015, Boston, MA, USA, June 7-12, 2015. (2015) 343–352
14. Leroy, V., Franco, J., Boyer, E.: Shape reconstruction using volume sweeping and learned photoconsistency. In: Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part IX. (2018)
15. Seitz, S.M., Curless, B., Diebel, J., Scharstein, D., Szeliski, R.: A comparison and evaluation of multi-view stereo reconstruction algorithms. In: 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2006), 17-22 June 2006, New York, NY, USA. (2006) 519–528
16. Furukawa, Y., Ponce, J.: Accurate, dense, and robust multi-view stereopsis. In: 2007 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2007), 18-23 June 2007, Minneapolis, Minnesota, USA. (2007)
17. Strecha, C., von Hansen, W., Gool, L.V., Fua, P., Thoennessen, U.: On benchmarking camera calibration and multi-view stereo for high resolution imagery. 2008 IEEE Conference on Computer Vision and Pattern Recognition (2008)
18. Gall, J., Stoll, C., Aguiar, E.D., Theobalt, C., Rosenhahn, B., peter Seidel, H.: Motion capture using joint skeleton tracking and surface estimation. In: CVPR. (2009)
19. Oswald, M.R., Cremers, D.: A convex relaxation approach to space time multi-view 3d reconstruction. In: ICCV Workshop on Dynamic Shape Capture and Analysis (4DMOD). (2013)
20. Mustafa, A., Kim, H., Guillemaut, J., Hilton, A.: Temporally coherent 4d reconstruction of complex dynamic scenes. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016. (2016) 4660–4669
21. Schöps, T., Schönberger, J.L., Galliani, S., Sattler, T., Schindler, K., Pollefeys, M., Geiger, A.: A multi-view stereo benchmark with high-resolution images and multi-camera videos. In: IEEE Conf. on Computer Vision and Pattern Recognition (CVPR). (2017)
22. Innmann, M., Zollhöfer, M., Nießner, M., Theobalt, C., Stamminger, M.: Volumedeform: Real-time volumetric non-rigid reconstruction. In: Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VIII. (2016)
23. Collet, A., Chuang, M., Sweeney, P., Gillett, D., Evseev, D., Calabrese, D., Hoppe, H., Kirk, A.G., Sullivan, S.: High-quality streamable free-viewpoint video. ACM Trans. Graph. (2015)
24. Dou, M., Khamis, S., Degtyarev, Y., Davidson, P., Fanello, S.R., Kowdle, A., Escolano, S.O., Rhemann, C., Kim, D., Taylor, J., Kohli, P., Tankovich, V., Izadi, S.: Fusion4d: Real-time performance capture of challenging scenes. ACM Trans. Graph. (2016)
25. Merrell, P., Akbarzadeh, A., Wang, L., michael Frahm, J., Nistr, R.Y.D.: Real-time visibility-based fusion of depth maps. In: Int. Conf. on Computer Vision and Pattern Recognition. (2007)
26. Starck, J., Hilton, A.: Surface capture for performance-based animation. IEEE Comput. Graph. Appl. **27**(3) (May 2007)
27. Labatut, P., Pons, J., Keriven, R.: Efficient multi-view reconstruction of large-scale scenes using interest points, delaunay triangulation and graph cuts. In: IEEE 11th International Conference on Computer Vision, ICCV 2007, Rio de Janeiro, Brazil, October 14-20, 2007. (2007) 1–8
28. Kutulakos, K.N., Seitz, S.M.: A theory of shape by space carving. International Journal of Computer Vision (2000)
29. Cremers, D., Kolev, K.: Multiview stereo and silhouette consistency via convex functionals over convex domains. IEEE Trans. Pattern Anal. Mach. Intell. **33**(6) (June 2011)
30. Ulusoy, A.O., Geiger, A., Black, M.J.: Towards probabilistic volumetric reconstruction using ray potentials. In: 3D Vision (3DV), 2015 3rd International Conference on. (2015)
31. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision **60**(2) (2004) 91–110
32. Bay, H., Tuytelaars, T., Gool, L.J.V.: SURF: speeded up robust features. In: Computer Vision - ECCV 2006, 9th

European Conference on Computer Vision, Graz, Austria, May 7-13, 2006, Proceedings, Part I. (2006) 404–417
33. Mikolajczyk, K., Schmid, C.: A performance evaluation of local descriptors. In: 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2003), 16-22 June 2003, Madison, WI, USA. (2003)
34. Tola, E., Lepetit, V., Fua, P.: DAISY: an efficient dense descriptor applied to wide-baseline stereo. IEEE Trans. Pattern Anal. Mach. Intell. **32**(5) (2010) 815–830
35. Pons, J.P., Keriven, R., Faugeras, O.: Multi-view stereo reconstruction and scene flow estimation with a global image-based matching score. IJCV **72**(2) (2007)
36. Collins, R.T.: A space-sweep approach to true multi-image matching. In: CVPR. (1996)
37. Furukawa, Y., Hernández, C.: Multi-view stereo: A tutorial. Foundations and Trends in Computer Graphics and Vision (2015)
38. Gallup, D., Frahm, J., Mordohai, P., Yang, Q., Pollefeys, M.: Real-time plane-sweeping stereo with multiple sweeping directions. In: 2007 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2007), 18-23 June 2007, Minneapolis, Minnesota, USA. (2007)
39. Hartmann, W., Galliani, S., Havlena, M., Van Gool, L., Schindler, K.: Learned multi-patch similarity. In: The IEEE International Conference on Computer Vision (ICCV). (Oct 2017)
40. Huang, Z., Li, T., Chen, W., Zhao, Y., Xing, J., LeGendre, C., Ma, C., Luo, L., Li, H.: Deep volumetric video from very sparse multi-view performance capture. European Conference on Computer Vision (ECCV) (2018)
41. Gilbert, A., Volino, M., Collomosse, J., Hilton, A.: Volumetric performance capture from minimal camera viewpoints. European Conference on Computer Vision (ECCV) (2018)
42. Kendall, A., Martirosyan, H., Dasgupta, S., Henry, P., Kennedy, R., Bachrach, A., Bry, A.: End-to-end learning of geometry and context for deep stereo regression. In: Proceedings of the International Conference on Computer Vision (ICCV). (2017)
43. Flynn, J., Neulander, I., Philbin, J., Snavely, N.: Deepstereo: Learning to predict new views from the world's imagery. Conference on Computer Vision and Pattern Recognition (CVPR) (2016)
44. Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G.S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., Zheng, X.: TensorFlow: Large-scale machine learning on heterogeneous systems (2015) Software available from tensorflow.org.
45. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In Bengio, Y., LeCun, Y., eds.: 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings. (2015)
46. Curless, B., Levoy, M.: A volumetric method for building complex models from range images. In: Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 1996, New Orleans, LA, USA, August 4-9, 1996. (1996)
47. Izadi, S., Kim, D., Hilliges, O., Molyneaux, D., Newcombe, R.A., Kohli, P., Shotton, J., Hodges, S., Freeman, D., Davison, A.J., Fitzgibbon, A.W.: 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, Santa Barbara, CA, USA, October 16-19, 2011. (2011) 559–568
48. : Filmic blender. `https://sobotka.github.io/filmic-blender/`
49. Tola, E., Strecha, C., Fua, P.: Efficient large-scale multi-view stereo for ultra high-resolution image sets. Mach. Vis. Appl. (2012)
50. Campbell, N.D.F., Vogiatzis, G., Hernández, C., Cipolla, R.: Using multiple hypotheses to improve depth-maps for multi-view stereo. In: Computer Vision - ECCV 2008, 10th European Conference on Computer Vision, Marseille, France, October 12-18, 2008, Proceedings, Part I. (2008) 766–779
51. Tola, E., Lepetit, V., Fua, P.: A fast local descriptor for dense matching. In: 2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2008), 24-26 June 2008, Anchorage, Alaska, USA. (2008)
52. Schönberger, J.L., Zheng, E., Pollefeys, M., Frahm, J.M.: Pixelwise view selection for unstructured multi-view stereo. In: European Conference on Computer Vision (ECCV). (2016)
53. Bleyer, M., Rhemann, C., Rother, C.: Patchmatch stereo - stereo matching with slanted support windows. In Hoey, J., McKenna, S.J., Trucco, E., eds.: British Machine Vision Conference (BMVC). (2011)
54. : 3d scanstore. `https://www.3dscanstore.com`