TAPA-MVS: Textureless-Aware PAtchMatch Multi-View Stereo

Andrea Romanoni
Politecnico di Milano, Italy
andrea.romanoni@polimi.it

Matteo Matteucci
Politecnico di Milano, Italy
matteo.matteucci@polimi.it

Abstract

One of the most successful approaches in Multi-View Stereo estimates a depth map and a normal map for each view via PatchMatch-based optimization and fuses them into a consistent 3D points cloud. This relies on photo-consistency to evaluate the goodness of a depth estimate. It generally produces very accurate results, however, the reconstructed model often lacks completeness, especially in correspondence of broad untextured areas where the photo-consistency metrics are unreliable. Assuming the untextured areas piecewise planar, in this paper we generate novel PatchMatch hypotheses so to expand reliable depth estimates in neighboring untextured regions. At the same time, we modify the photo-consistency measure such to favor standard or novel PatchMatch depth hypotheses depending on the textureness of the considered area. Finally, we propose a depth refinement step to filter out wrong estimates and to fill gaps on both the depth and normal maps, while preserving discontinuities. Our method proved its effectiveness against several state of the art algorithms in the publicly available ETH3D dataset containing a wide variety of high and low-resolution images.

1. Introduction

Multi-View Stereo (MVS) aims at recovering a dense 3D representation of the scene perceived by a set of calibrated images, for instance, to map cities, to create a digital library of cultural heritage or to help robots navigating an environment. Thanks to the availability of public datasets [21, 24, 9], several successful MVS algorithms have been proposed in the last decade, and their performance keeps increasing.

Depth map estimation represents one of the fundamental and most challenging steps on which most MVS methods rely. Depth maps are then fused together directly into a point cloud [31, 18], or into a volumetric representation, such as a voxel grid [17, 3] or Delaunay triangulation [11, 26, 10, 14]. In the latter case a 3D mesh is extracted and can be further refined via variational methods [26, 2, 13, 16] and eventually labelled with semantics [15].

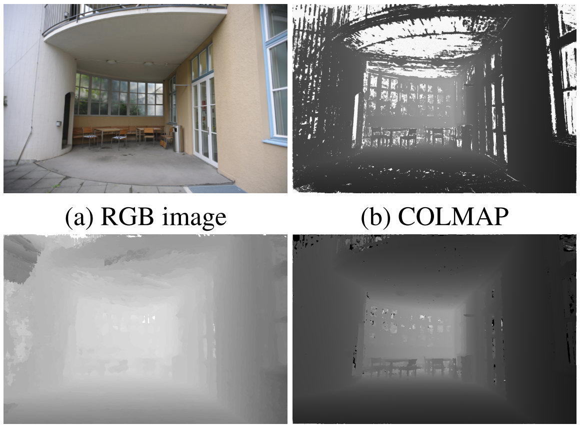

(a) RGB image
(b) COLMAP
(c) DeepMVS
(d) TAPA-MVS

Figure 1. Example of the depth map produced by the proposed method with respect to the state-of-the-art

Although Machine Learning methods have begun to appear [7, 27, 30], PatchMatch-based algorithms, emerged some years ago, are still among the top performing approaches for efficient and accurate depth map estimation. The core idea of PatchMatch, pioneered by Barnes *et al*. [1] and extended for depth estimation by Bleyer *et al*. [4], is to choose for each pixel a random guess of the depth and then propagate the most likely estimates to their neighborhood. Starting from this idea, Schönberger *et al*. [18] recently proposed a robust framework able to jointly estimate the depth, the normals, and the pixel-wise camera visibility for each view.

One of the major drawbacks of PatchMatch methods is that most of the untextured regions are not managed correctly (Figure 1(b)). Indeed the optimization highly relies on the photometric measure to discriminate which random estimate is the best guess and to filter out unstable estimates. The depth of the untextured regions is hard to define with enough confidence since they are homogeneous and thus, the photometric measure alone hardly discerns neighboring regions.

In this paper, we specifically address the untextured regions drawback by leveraging on the assumption that they are often piecewise flat (Figure 1(d)). The framework presented, named TAPA-MVS, proposes:

- a metric to define the textureness of each image pixel; it serves as a proxy to understand how much the photo-consistency metric is reliable.
- to subdivide the image into superpixels and, for each iteration of the optimization procedure, to fit one plane for each superpixel; for each pixel, a new depth-normal hypothesis together with the textureness defined before, is evenly integrated in the optimization framework considering the likelihood of the plane fitting procedure.
- a novel depth refinement method that filters the depth and normal maps and fills each missing estimates with an approximate bilateral weighted median of the neighbors.

We tested the proposals against the 38 sequences of the publicly available ETH3D dataset [19] (Section 6) and the results show that our method is able to significantly improve the completeness of the reconstruction while preserving a very good accuracy. Our approach improves COLMAP results also in fountain-P11, HerzJesu-P8 [24], Tower of London and NotreDame [28] (see Supplementary materials).

2. Patch-Match for Multi-View Stereo

The PatchMatch seminal paper by Barnes *et al.* [1] proposed a general method to efficiently compute an approximate nearest neighbor function defining the pixelwise correspondence among patches of two images. The idea is to use a collaborative search exploiting local coherency. PatchMatch initializes each pixel of an image with a random guess about the location of the nearest neighbor in the second image. Then, each pixel propagates its estimate to the neighboring pixels and, among these estimates, the most likely is assigned to the pixel itself. As a result the best estimates spread along the entire image.

Bleyer *et al.* [4] re-framed this method into the stereo matching realm. Indeed, for each image patch, stereo matching looks in the second image for the corresponding patch, *i.e.* the nearest neighbor in the sense of photometric consistency. To improve its robustness the matching function is defined on slanted support windows. Heise *et al.* [6] produce smoother depth estimates while preserving edges discontinuities, by regularizing the estimate with quadratic relaxation.

The natural extension of PatchMatch from pair-wise stereo matching to Multi-View Stereo was proposed by Shen [23]. The author selects a subset of camera pairs depending on the number of shared points computed by Structure from Motion and their mutual parallax angle. Then he estimates a depth map for the selected subset of camera pairs through a simplified version of the method of Bleyer *et al.* [4]. The algorithm refines the depth maps by enforcing consistency among multiple views, and it finally merges the depth maps into a point cloud.

Galliani *et al.* [5] modify the PatchMatch propagation scheme so to better exploit parallelization of GPUs. Differently, from Shen [23], they aggregate, for each reference camera, a set of matching costs computed from different source images. One of the major drawbacks of these approaches is the decoupled depth estimation and camera pairs selection. Xu and Tao [29] recently proposed an attempt to overcome this issue; they extended [5] with a more efficient propagation pattern and, in particular, their optimization procedure jointly considers all the views and all the depth hypotheses.

Rather than considering the whole set of images to compute the matching costs, Zheng *et al.* [31] proposed an elegant method to deal with view selection. They framed the joint depth estimation and pixel-wise view selection problem into a variational approximation framework. Following a generalized Expectation Maximization paradigm, they alternate between depth update with PatchMatch, keeping the view selection fixed, and pixel-wise view inference with the forward-backward algorithm, keeping the depth fixed.

Schönberger *et al.* [18] extended this method to jointly estimate per-pixel depths and normals, such that, differently from [31], the knowledge of the normals enables slanted support windows to avoid the fronto-parallel assumption. Then they add view-dependent priors to select views that more likely induce robust matching cost computation.

The PatchMatch based methods described thus far, have been proven to be among the top performing approachs in several MVS benchmarks [22, 24, 9, 20]. However, some issues are still open. In particular, most of them strongly rely on photo-consistency measures to discriminate among depth hypotheses. Even if this works remarkably for textured areas and the propagation scheme partially induces smoothness, untextured regions are often poorly reconstructed. For this reason, we propose two proxies to improve the reconstruction where untextured areas appear. On the one hand, we seamlessly extend the probabilistic framework to explicitly detect and handle untextured regions by extending the set of PatchMatch hypotheses. On the other side, we complete the depth estimation with a refinement procedure to fill the missing depth estimates.

3. Review of the COLMAP framework

In this section we review the state-of-the-art framework proposed by Schönberger *et al.* [18] which builds on top of the method presented by Zheng *et al.* [31]. Let note that in the following, we express the coordinate of the pixel only with a value l, since both frameworks sweep independently every single line of the image alternating between rows and columns.

Given a reference image $\mathbf{X}^{\mathrm{ref}}$ and a set of source images $\mathbf{X}^{\mathrm{src}} = \{X^m | m = 1 \dots M\}$, the framework estimates the depth θ_l and the normal $\mathbf{n}_l$ of each pixel l, together with a binary variable $Z_l^m \in \{0, 1\}$, which indicates if l is visible in image m. This is framed into a Maximum-A Posteriori (MAP) estimation where the posterior probability is:

$$P(\mathbf{Z}, \theta, \mathbf{N} | \mathbf{X}) = \frac{P(\mathbf{Z}, \theta, \mathbf{N}, \mathbf{X})}{P(\mathbf{X})} = \\ = \frac{1}{P(\mathbf{X})} \prod_{l=1}^{L} \prod_{m=1}^{M} \left[P\left(Z_{l,t}^m | Z_{l-1,t}^m, Z_{l,t-1}^m \right) \right. \\ \left. P\left(X_l^m | Z_l^m, \theta_l, \mathbf{n}_l, X^{ref} \right) P\left(\theta_l, \mathbf{n}_l | \theta_l^m, \mathbf{n}_l^m \right) \right], \quad (1)$$

where L is the number of pixels considered in the current line sweep, $\mathbf{X} = \left\{ \mathbf{X}^{\mathrm{src}}, \mathrm{X}^{\mathrm{ref}} \right\}$ and $\mathbf{N} = \{\mathbf{n}_l | l = 1 \dots L\}$. The likelihood term

$$P\left(X_l^m | Z_l^m, \theta_l \right) = \begin{cases} \frac{1}{NA} \exp\left(-\frac{(1-\rho_l^m(\theta_l))^2}{2\sigma_\rho^2} \right) & \text{if } Z_l^m = 1 \\ \frac{1}{N}\mathcal{U} & \text{if } Z_l^m = 0, \end{cases} \quad (2)$$

represents the photometric consistency of the patch X_l^m, which belongs to a non-occluded source image m and is around the pixel corresponding to the point at l, with respect to the patch X_l^{ref} around l in the reference image. The photometric consistency ρ is computed as a bilaterally weighted NCC, $A = \int_{-1}^{1} \exp\left\{ -\frac{(1-\rho)^2}{2\sigma_\rho^2} \right\} d\rho$ and the constant N cancels out in the optimization. The likelihood term $P\left(\theta_l, \mathbf{n}_l | \theta_l^m, \mathbf{n}_l^m \right)$ represents the geometric consistency and enforces multi-view depth and normal coherence. Finally $P\left(Z_{l,t}^m | Z_{l-1,t}^m, Z_{l,t-1}^m \right)$ favors image occlusion indicators which are smooth both spatially and along the successive iteration of the optimization procedure.

Being Equation (1) intractable, Zheng *et al.* [31] proposed to use variational inference to approximate the real posterior with a function $q(\mathbf{Z}, \theta, \mathbf{N})$ such that the KL divergence of the two functions is minimized. Schönberger *et al.* [18] factorize $q(\mathbf{Z}, \theta, \mathbf{N}) = q(\mathbf{Z}) q(\theta, \mathbf{N})$ and, to estimate such approximation, they propose a variant of the Generalized Expectation-Maximization algorithm [12]. In the E step, the values $(\theta, \mathbf{N})$ are kept fixed, and, in the resulting Hidden Markov Model, the function $q(Z_{l,t}^m)$ is computed by means of message passing. In the M step, viceversa, the values of $Z_{l,t}^m$ are fixed, the function $q(\theta, \mathbf{N})$ is constrained to the family of Kroneker delta functions $q(\theta_l, \mathbf{n}_l) = q(\theta_l = \theta_l^*, \mathbf{n}_l^*)$. The new optimal values of θ_l and $\mathbf{N}_l$ are computed as:

$$\left(\hat{\theta}_l^{\,\mathrm{opt}}, \hat{\mathbf{n}}_l^{\mathrm{opt}} \right) = \operatorname*{argmin}_{\theta_l^*, \mathbf{n}_l^*} \frac{1}{|S|} \sum_{m \in S} \left(1 - \rho_l^m \left(\theta_l^*, \mathbf{n}_l^* \right) \right), \quad (3)$$

where S is a subset of sources images, randomly sampled according to a probability $P_l(m)$. Probability $P_l(m)$ favors images not occluded, and coherent with three priors which encourage good inter-cameras parallax, similar resolution and camera, front-facing the 3D point defined by $\theta_l^*, \mathbf{n}_l^*$.

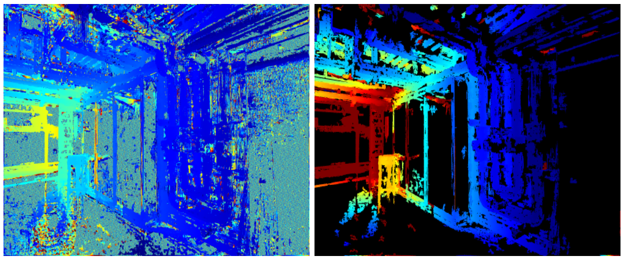

(a) (b)

Figure 2. Depth map after the first iteration (a). Unstable regions have been filtered in (b).

According to the PatchMatch scheme proposed in [18], the pair $(\theta_l^*, \mathbf{n}_l^*)$ evaluated in Equation (3) is chosen among the following set of hypotheses:

$$\left\{ (\theta_l, \mathbf{n}_l), \left(\theta_{l-1}^{\mathrm{prp}}, \mathbf{n}_{l-1} \right), \left(\theta_l^{\mathrm{rnd}}, \mathbf{n}_l \right), \left(\theta_l, \mathbf{n}_l^{\mathrm{rnd}} \right), \right. \\ \left. \left(\theta_l^{\mathrm{rnd}}, \mathbf{n}_l^{\mathrm{rnd}} \right), \left(\theta_l^{\mathrm{prt}}, \mathbf{n}_l \right), \left(\theta_l, \mathbf{n}_l^{\mathrm{prt}} \right) \right\}, \quad (4)$$

where $(\theta_l, \mathbf{n}_l)$ comes from the previous iteration, $(\theta_{l-1}, \mathbf{n}_{l-1})$ is the estimate from the previous pixel of the scan, $\left(\theta_l^{\mathrm{rnd}}, \mathbf{n}_l \right)$ is a random hypothesis and finally, θ_l^{prt} and $\mathbf{n}_l^{\mathrm{prt}}$ are two small perturbations of the estimates θ_l and $\mathbf{n}_l$.

4. Textureness-Aware Joint PatchMatch and View Selection

The core ingredient that makes a Multi-View Stereo algorithm successful is the quality and the discriminative effectiveness of the stereo comparison among patches belonging to different cameras. Such comparison relies on a photometric measure, computed as Normalized Cross Correlation or similar metrics such as Sum of Squared Differences (SSD), or Bilateral Weighted NCC. The major drawback arises in correspondence of untextured regions. Here the discriminative capabilities of NCC become unreliable because all the patches belonging to the untextured area are similar among each other.

Under these assumptions, the idea behind our proposal is to segment images into superpixels such that each superpixel would span a region of the image with a texture mostly homogeneous and it likely stops in correspondence to an image edge. Then, we propagate the depth/normal estimates belonging to photometrically stable regions around the edges to the entire superpixel. In the following we assume the first iteration of the framework presented in Section 3 is executed so that we have a very first estimation of the depth map, which is reliable only in correspondence of highly textured regions (Figure 2).

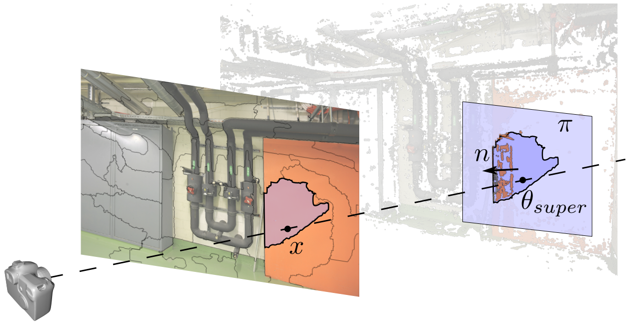


Figure 3. Depth hypothesis generation. The depth θ is the distance from the camera to the the plane π, estimated with the 3D points corresponding to the superpixel extracted on the image.

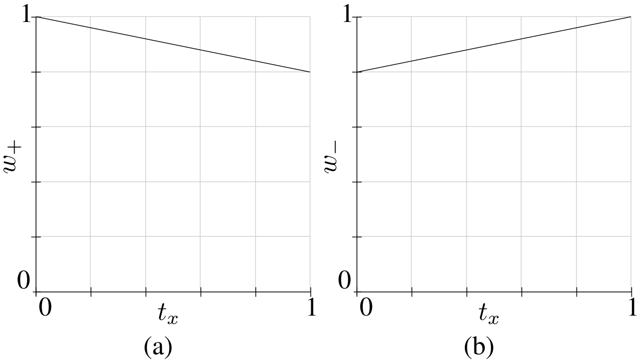


Figure 4. Weights adopted to tune the photo-consistency and the geometric cost according to the textureness t_x

4.1. Piecewise Planar Hypotheses generation

The idea of the method is to augment the set of PatchMatch depth hypotheses in Equation 4 with novel hypotheses that model a piecewise planar prior corresponding to untextured areas.

In the first step we extract the superpixels $\mathcal{S} = \{s_1, s_2, \ldots, s_{N_{super}}\}$ of each image by means of the algorithm SEEDS [25]. Since, a superpixel s_k generally contains homogeneous texture, we assume that each pixel covered by a superpixel s_k roughly belongs to the same plane.

After running the first iteration of depth estimation, we filter out the small isolated speckles of the depth map obtained (in this paper, with area smaller than $\frac{imagearea}{5000}$). As a consequence, the area of s_k in the filtered depth map likely contains a set $\mathcal{P}_k^{\text{inl}}$ of reliable 3D points estimates which roughly corresponds to real 3D points. In the presence of untextured regions, these points mostly belong to the areas near edges (Figure 2).

We fit a plane π_k on the 3D points in $\mathcal{P}_k^{\text{inl}}$ with RANSAC, classifying the points farther than 10 cm from the plane as outliers. Let us define $\hat{\theta}_x$ the tentative depth hypothesis for a pixel x corresponding to the 3D point on the plane π_k and $\hat{\mathbf{n}}_x$ the corresponding plane normal (Figure 3) Then, let us define the inlier ratio $r_k^{inl} = \frac{\text{num. inliers}}{|\mathcal{P}_k^{\text{inl}}|}$, whose value expresses the confidence of the plane estimate.

The actual hypotheses $(\theta_x, \mathbf{n}_x)$ for a pixel $x \in s_k$ is generated as follows. To deal with fitting uncertainty, we first define $P\left((\theta_x, \mathbf{n}_x) = (\hat{\theta}_x, \hat{\mathbf{n}}_x)\right) = r_k^{inl}$; so that if the value v_{ran} sampled from a uniform distribution is $v_{ran} <= r_k^{inl}$ then $\theta_x = \hat{\theta}_x$. To propagate the hypotheses from superpixels with good inlier ratio to the neighbors with bad one, if $v_{ran} > r_k^{inl}$ the value of θ_x is sampled from the neighboring superpixels belonging to a set $\mathcal{N}_k$. Since we aim at spreading the depth hypotheses among superpixels with a similar appearance, we sample from $\mathcal{N}_k$ proportionally to the Bhattacharya distance among the RGB histograms of s_k and the elements of $\mathcal{N}_k$.

Experimentally, we noticed that the choice of N_{super}, i.e., the number of superpixels, influences how the untextured areas are treated and modeled in our method. With small values of N_{super} large areas of the images are nicely covered, but at the same time, limited untextured regions are improperly fused. Vice-versa, a big N_{super} better models small regions while underestimating large areas. For this reason, we choose to adopt both a coarse and a fine superpixel segmentation of the image such that both small and large untextured areas are modeled properly. Therefore, for each pixel, we generate two depth hypotheses: $(\theta_x^{\text{fine}}, \mathbf{n}_x^{\text{fine}})$ and $(\theta_x^{\text{coarse}}, \mathbf{n}_x^{\text{coarse}})$. In our experiments we choose $N_{super}^{\text{fine}} = \frac{imagewidth}{20}$ and $N_{super}^{\text{coarse}} = \frac{imagewidth}{30}$.

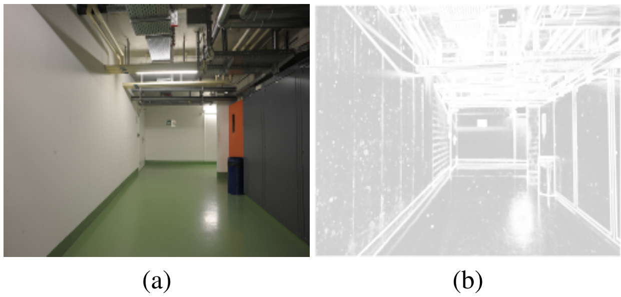
4.2. Textureness-Aware Hypotheses Integration

To integrate the novel hypotheses into the estimation framework, it is possible to simply add $(\theta_x^{\text{fine}}, \mathbf{n}_x^{\text{fine}})$ and $(\theta_x^{\text{coarse}}, \mathbf{n}_x^{\text{coarse}})$ to the set of hypotheses defined in Equation 4. However, in this case, these hypotheses would be treated with no particular attention to untextured areas. Indeed, the optimization framework would compare them against the baseline hypotheses relying on the photo-consistency metric; in the presence of flat evenly colored surfaces, the unreliability of the metric would still affect the estimation process. Instead, the goal of the proposed method is to favor $(\theta_x^{\text{fine}}, \mathbf{n}_x^{\text{fine}})$ and $(\theta_x^{\text{coarse}}, \mathbf{n}_x^{\text{coarse}})$ where the image presents untextured areas, so to guide the optimization to choose them instead of other guesses.

For these reasons, we first define a pixel-wise textureness coefficient to measure the amount of texture that surrounds a pixel x. With a formulation similar to those presented in [26], we define it as:

$$t_x = \frac{Var_x + \epsilon_{var}}{Var_x + \frac{\epsilon_{var}}{t_{min}}} \tag{5}$$

where Var_x is the variance of the 5x5 patch around pixel x, ϵ_{var} is a constant we fixed experimentally at 0.00005,



(a) (b)

Figure 5. Visualization of the textureness coefficients computed on image (a)

i.e., two order of magnitude smaller than the average variance we found in the ETH3D training dataset (Section 6), finally, $t_{min} = 0.5$ is the minimum value we choose for the textureness coefficient; the higher the variance, the closer the coefficient is to 1.0. Figure 5 shows an example of a textureness coefficients image.

To seamlessly integrate the novel hypotheses we use the textureness coefficient to reweight the photometric-based cost $C_{photo} = 1 - \rho(\theta, \mathbf{n})$ (Equation 3). Given a pixel x let define two weights: $w^{+}(x) = 0.8 + 0.2 \cdot t_x$ and $w^{-}(x) = 1.0 - 0.2 \cdot t_x$.

We use the metric $\bar{C}_{photo} = w^{-} \cdot C_{photo}$ for the hypotheses contained in the set of Equation 4 and $\bar{C}_{photo} = w^{+} \cdot C_{photo}$ for $(\theta_x^{\text{fine}}, \mathbf{n}_x^{\text{fine}})$ and $(\theta_x^{\text{coarse}}, \mathbf{n}_x^{\text{coarse}})$ so that regions with low texture favors novel hypotheses. Vice-versa, it is better to force a higher geometric consistency C_{geom} when we are dealing with the novel hypothesis in the presence of untextured areas. So to keep the formulation simple we use w^{+} and w^{-} again turning $\bar{C}_{geom} = w^{+} \cdot C_{geom}$ for the standard set of hypotheses and $\bar{C}_{geom} = w^{-} \cdot C_{geom}$ for the proposed ones.

5. Joint Depth and Normal Depth Refinement

The hypotheses proposed in the previous section improve the framework estimate accuracy and completeness in correspondence of untextured regions. However, two issues remain open. First, the filtering scheme adopted in [18] filters out all the estimates that are not photometrically and geometrically consistent among the views. Due to their photometric instability, the photo-consistency check removes most of the new depth estimates corresponding to unfiltered areas; therefore, in our case, we neglect this filtering step.

This leads us to the second issue. The resulting depth map contains wrong and noisy estimates sparsely spread along the depth image (Figure 6(a)). For this reason, we complemented the estimation process with a depth refinement step. To get rid of wrong estimates that have not converged to a stable solution, we first apply a classical speckles filter to remove small blobs containing non-continuous depths values. We fixed, experimentally, the maximum speckle size of continuous pixels to $\frac{imagearea}{5000}$. We consider two pixels as continuous when the depth difference is at most 10% of the scene size.

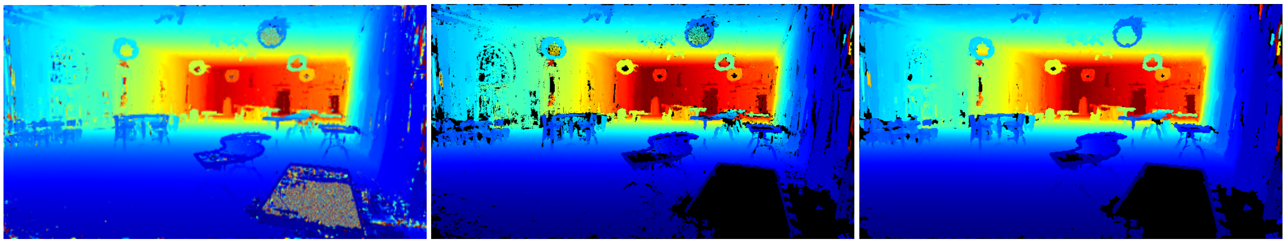
The output of the filtering procedure contains now small regions where the depth and normal estimates are missing (Figure 6(b)). To recover them, we designed the following refinement step. Let x_{miss} be a pixel where depth and normal estimates are missing and $\mathcal{N}_{\text{miss}}$ the set of neighboring pixels. The simplest solution is to fill the missing estimate by averaging the depth and normal values contained in $\mathcal{N}_{\text{miss}}$. A better choice would be to weight the contribution to the average with the bilateral coefficients adopted in the bilateral NCC computation; they give more importance to the pixels close to x_{miss} both in the image and in color space.

To better deal with depth discontinuities, we can improve even further the refinement process by using a weighted median of depth and normal instead of the weighted average. The pixel-wise median and, in particular, the weighted median is computational demanding, thus, to approximate the median computation, we populate a three bins histogram with the depths of the pixels in $\mathcal{N}_{\text{miss}}$. We choose the bin with the highest frequency so to get rid of the outliers, and we compute a bilaterally weighted average of the depth and normals that populates this bin (Figure 6(c)). The computed depth/normal values are assigned to x_{miss}.

6. Experiments

We tested the proposed method on an Intel(R) Xeon(R) CPU E5-2687W with a GeForce GTX 1080 against the publicly available ETH3D datasets [19], fountain-P11, HerzJesu-P8 [24]. In the Supplementary materials we also reported a qualitatively comparison against Tower of London and NotreDame [28].

ETH3D dataset is split into test/training and low-/high-resolution for a total of 38 sequences. Parameter tuning is only permitted with the training sequences that are available with the ground truth. The comparison is carried out by computing the distance from the 3D model to the ground-truth (GT) 3D scans and vice-versa; then, accuracy, completeness, and F1-score are computed considering the percentage of model-to-GT distances below a fixed threshold τ. For a complete description of the evaluation procedure, we refer the reader to [19]. To generate the 3D model out of the depth map estimated with the proposed method, we adopted the depth filtering and fusion implemented in COLMAP. Since depth estimate corresponding to untextured regions can get noisy, we changed the default fusion parameter such that the reprojection error permitted, is more strict (half for high-resolution sequences a quarter for low-resolution ones). On the other hand, even the normal estimate could be noisy, but, usually, the corresponding depths are reasonable. For this reason, we allow for larger normal



Original Depth Map(a) After Speckles Removal(b) After Depth Refinement(c)

Figure 6. Depth map refinement

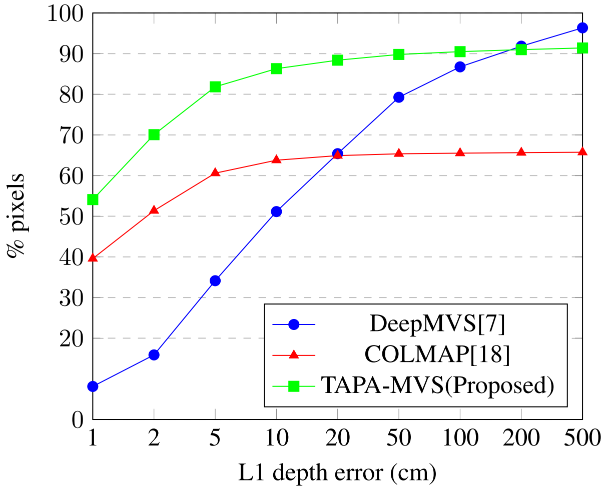


Figure 7. Depth map error distribution

errors (double the normal error permitted by COLMAP) and demand the outlier filtering to the reprojection error check.

Table 1 shows the F1-scores computed with a threshold of 2 cm, which is the default values adopted for the dataset leaderboard. TAPA-MVS, *i.e.*, the method proposed in this paper, is ranked first according to the overall F1-score of both the Training and Test sequences. It is worth noticing that TAPA-MVS, improves significantly the results of the baseline COLMAP framework. The reason for such successful reconstruction has to be ascribed to the texture aware mechanism which is able to accurately reconstruct the photometrically stable areas and to recover the missing geometry where the photo-consistent measure is unreliable. Figure 8 shows the models recovered by TAPA-MVS and the top performing algorithms in some of the ETH3D sequences. The models reconstructed by TAPA-MVS are significantly more complete and contain less noise.

To further test the effectiveness of our method, we compared directly the accuracy of the depth map in the 13 training high-resolution sequences against the baseline COLMAP [18] and the recent deep learning-based DeepMVS [29]. Figure 7 illustrates the error distribution, *i.e.*, the percentage of pixels in the depth maps whose error is lower than a variable threshold (x-axis). TAPA-MVS clearly shows better completeness with respect to both methods, especially when considering small errors. In Figure 9 we define image regions with respect to increasing textureness values relying of the term t_x described in Section 4.2. Given a value v in the x-axis, we consider the image areas where the textureness coefficient $t_x < v$ and we plot in the three graphs the percentage of pixels in these areas with a depth error lesser than 10cm, 20 cm or 50cm. These graphs demonstrate the robustness of the proposed method against untextured regions, indeed even in low-textured areas, the percentage of pixel correctly estimated is comparable to the highly textured regions.

Finally we compared our method againts DeepMVS and COLMAP in fountain-P11 and HerzJesu-P8; even id they captures two textured scenes the proposed method still estimate better depth maps.

Ablation study To assess the effectiveness of all the proposal of the paper, Table 2 shows the accuracy, completeness and F1-score of our method in the training high-resolution sequences whose ground truth is publicly available. In the table, the rows represent increasing values of the distance threshold τ. We listed the results without the Texture Weighting (TW), without using the Coarse or the Fine Superpixels (CS and FS) and finally without the Depth Refinement step (DR). We also added to the comparison the COLMAP performance [18] which is the baseline algorithm prior to the novel steps suggested by this paper.

As expected COLMAP achieves the best accuracy at the cost of lower completeness since it produces depth estimates only in correspondence of textured regions. The data clearly shows that all the single proposal described in the previous sections are crucial to the balance between model completeness and accuracy obtained by TAPA-MVS. In particular, texture weighting is fundamental to avoid the framework treating the proposed hypothesis with the same importance as the old ones, no matter how much texture the image contains, this induces, in some cases, severe errors that led the optimization into local minima. The Superpixels plane fitting steps are both relevant to obtain good guesses

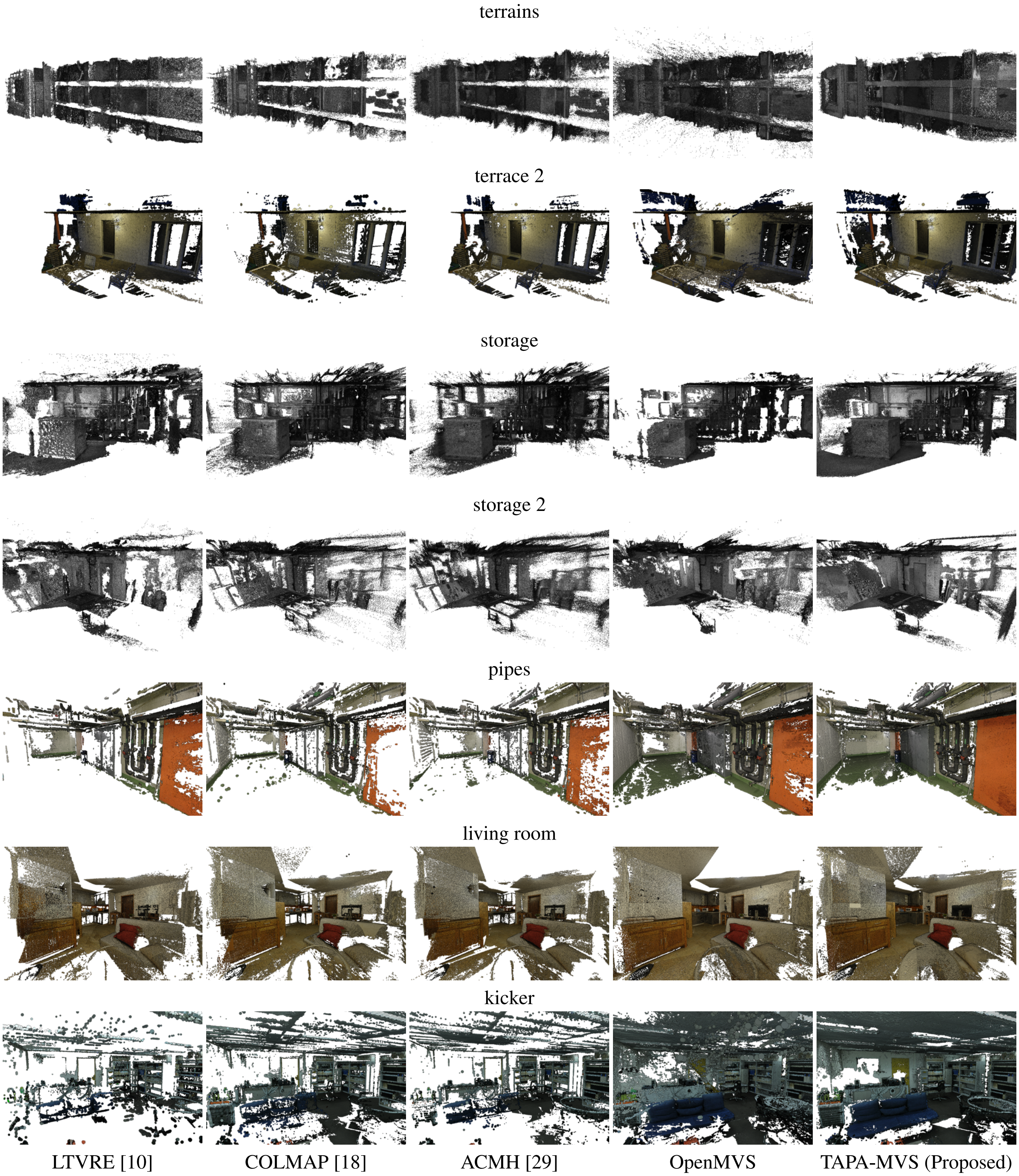


Figure 8. Results on ETH3D

Method	Training sequences			Test sequences		
	Overall	Low-Res	High-Res	Overall	Low-Res	High-Res
TAPA-MVS (Proposed)	**71.42**	55.13	**77.69**	**73.13**	**58.67**	79.15
OpenMVS	70.44	**55.58**	76.15	72.83	56.18	**79.77**
ACMH [29]	65.37	51.50	70.71	67.68	47.97	75.89
COLMAP [18]	62.73	49.91	67.66	66.92	52.32	73.01
LTVRE [10]	59.44	53.25	61.82	69.57	53.52	76.25
CMPMVS [8]	47.48	9.53	62.49	51.72	7.38	70.19

Table 1. f1 scores on the ETH3D Dataset with tolerance τ =2cm (used by default for the dataset leaderboard page).

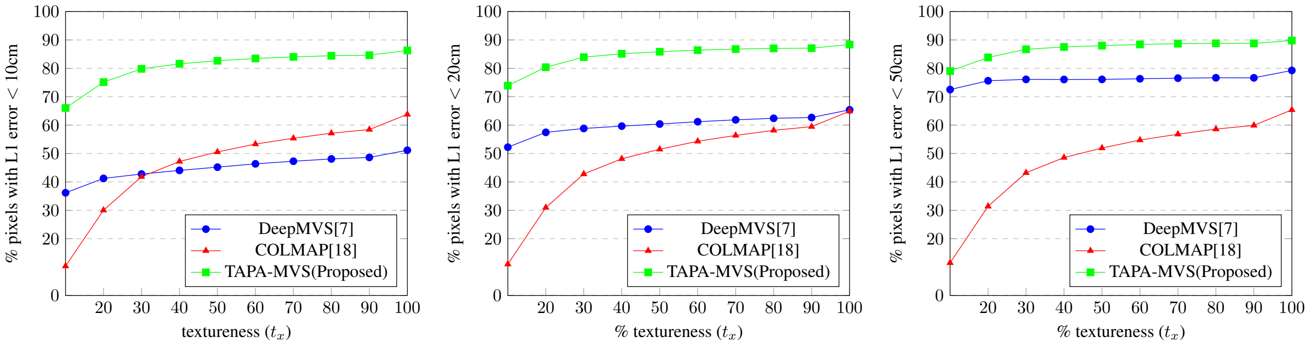


Figure 9. Percentage of pixels with error $<$ 10cm, 20cm and 50cm with respect to textureness

τ	COLMAP[18]			w/o TW			w/o CS			w/o FS			w/o DR			TAPA-MVS		
	C	A	F1	C	A	F1	C	A	F1	C	A	F1	C	A	F1	C	A	F1
1	38.65	**84.34**	51.99	32.68	74.40	44.58	41.72	75.30	53.18	41.35	75.10	52.86	47.78	72.13	56.31	**51.66**	75.37	**60.85**
2	55.13	**91.85**	67.66	52.57	85.70	63.08	64.13	85.98	72.54	63.69	85.77	72.26	64.27	83.32	71.84	**71.45**	85.88	**77.69**
5	69.91	**97.09**	80.5	69.31	94.08	78.62	81.08	93.69	86.68	80.84	93.58	86.51	78.62	92.51	84.37	**84.83**	94.31	**88.91**
10	79.47	**98.75**	87.61	78.10	96.91	85.64	88.80	96.53	92.38	88.61	96.45	92.22	86.33	95.94	90.47	**90.98**	96.79	**93.69**
20	88.24	**99.37**	93.27	84.93	98.34	90.53	93.64	98.12	95.77	93.61	98.05	95.72	91.26	97.75	94.25	**94.72**	98.23	**96.38**
50	96.03	**99.70**	97.78	92.07	99.30	95.19	97.33	99.23	98.25	97.54	99.20	98.34	95.65	99.21	97.23	**97.60**	99.30	**98.41**

Table 2. Ablation study: without Texture Weighting (TW), Coarse Superpixels (CS), Fine Superpixels (FS), Depth Refinement (DR)

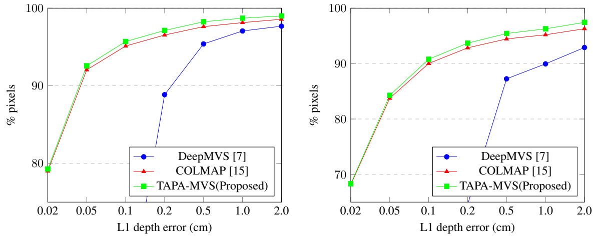


Figure 10. Error distribution in fountain-P11 (left) and HerzJesu-P8 (right) datasets

for untextured regions. Finally, depth refinement not only improves the completeness of the results but, by filtering out wrong estimates and replacing them with a careful neighbors interpolation at the missing estimate, it improves the accuracy as well.

7. Conclusions and Future Works

We presented a PatchMatch-based framework for Multi-View Stereo which is robust in correspondence of untextured regions. By choosing a set of novel PatchMatch hypotheses, the optimization framework expands the photometrically stable depth estimates, corresponding to image edges and textured areas, to the neighboring untextured regions. We demonstrated that a modification of the cost function used by the framework to evaluate the goodness of such hypotheses is needed, in particular, by favoring the novel ones when the textureness is low. We finally propose a depth refinement method that improves both reconstruction accuracy and completeness.

In the future, we plan to build a complete textureness-aware MVS pipeline including also a mesh reconstruction and refinement stages. In particular, we are interested in a robust meshing stage embedding piecewise planar priors, where the point clouds regions correspond to untextured areas. Moreover, we would like to define a mesh refinement method that balances regularization and data-driven optimization depending on image textureness.

References

[1] C. Barnes, E. Shechtman, A. Finkelstein, and D. Goldman. Patchmatch: A randomized correspondence algorithm for structural image editing. *ACM Transactions on Graphics-TOG*, 28(3):24, 2009.

[2] M. Blaha, M. Rothermel, M. R. Oswald, T. Sattler, A. Richard, J. D. Wegner, M. Pollefeys, and K. Schindler. Semantically informed multiview surface refinement. *International Journal of Computer Vision*, 2017.

[3] M. Blaha, C. Vogel, A. Richard, J. D. Wegner, T. Pock, and K. Schindler. Large-scale semantic 3d reconstruction: an adaptive multi-resolution model for multi-class volumetric labeling. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3176–3184, 2016.

[4] M. Bleyer, C. Rhemann, and C. Rother. Patchmatch stereo-stereo matching with slanted support windows. In *BMVC*, volume 11, pages 1–11, 2011.

[5] S. Galliani, K. Lasinger, and K. Schindler. Massively parallel multiview stereopsis by surface normal diffusion. *The IEEE International Conference on Computer Vision (ICCV)*, June 2015.

[6] P. Heise, S. Klose, B. Jensen, and A. Knoll. Pm-huber: Patchmatch with huber regularization for stereo matching. In *Computer Vision (ICCV), 2013 IEEE International Conference on*, pages 2360–2367. IEEE, 2013.

[7] P.-H. Huang, K. Matzen, J. Kopf, N. Ahuja, and J.-B. Huang. Deepmvs: Learning multi-view stereopsis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2821–2830, 2018.

[8] M. Jancosek and T. Pajdla. Multi-view reconstruction preserving weakly-supported surfaces. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 3121–3128. IEEE, 2011.

[9] R. Jensen, A. Dahl, G. Vogiatzis, E. Tola, and H. Aanæs. Large scale multi-view stereopsis evaluation. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 406–413. IEEE, 2014.

[10] A. Kuhn, H. Hirschmüller, D. Scharstein, and H. Mayer. A tv prior for high-quality scalable multi-view stereo reconstruction. *International Journal of Computer Vision*, 124(1):2–17, 2017.

[11] P. Labatut, J.-P. Pons, and R. Keriven. Efficient multi-view reconstruction of large-scale scenes using interest points, delaunay triangulation and graph cuts. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8. IEEE, 2007.

[12] R. M. Neal and G. E. Hinton. A view of the em algorithm that justifies incremental, sparse, and other variants. In *Learning in graphical models*, pages 355–368. Springer, 1998.

[13] A. Romanoni, M. Ciccone, F. Visin, and M. Matteucci. Multi-view stereo with single-view semantic mesh refinement. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 706–715, 2017.

[14] A. Romanoni and M. Matteucci. Efficient moving point handling for incremental 3d manifold reconstruction. In *Image Analysis and ProcessingICIAP 2015*, pages 489–499. Springer, 2015.

[15] A. Romanoni and M. Matteucci. A data-driven prior on facet orientation for semantic mesh labeling. In *2018 International Conference on 3D Vision (3DV)*, pages 662–671. IEEE, 2018.

[16] A. Romanoni and M. Matteucci. Mesh-based camera pairs selection and occlusion-aware masking for mesh refinement. *Pattern Recognition Letters*, 125:364–372, 2019.

[17] N. Savinov, C. Häne, L. Ladicky, and M. Pollefeys. Semantic 3d reconstruction with continuous regularization and ray potentials using a visibility consistency constraint. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5460–5469, 2016.

[18] J. L. Schönberger, E. Zheng, J.-M. Frahm, and M. Pollefeys. Pixelwise view selection for unstructured multi-view stereo. In *European Conference on Computer Vision*, pages 501–518. Springer, 2016.

[19] T. Schops, J. L. Schonberger, S. Galliani, T. Sattler, K. Schindler, M. Pollefeys, and A. 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.

[20] T. Schöps, J. L. Schönberger, S. Galliani, T. Sattler, K. Schindler, M. Pollefeys, and A. Geiger. A multi-view stereo benchmark with high-resolution images and multi-camera videos. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[21] S. M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *Computer vision and pattern recognition, 2006 IEEE Computer Society Conference on*, volume 1, pages 519–528. IEEE, 2006.

[22] S. M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *Computer vision and pattern recognition, 2006 IEEE Computer Society Conference on*, volume 1, pages 519–528. IEEE, 2006.

[23] S. Shen. Accurate multiple view 3d reconstruction using patch-based stereo for large-scale scenes. *IEEE transactions on image processing*, 22(5):1901–1914, 2013.

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

[25] M. Van den Bergh, X. Boix, G. Roig, and L. Van Gool. Seeds: Superpixels extracted via energy-driven sampling. *International Journal of Computer Vision*, 111(3):298–314, 2015.

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

[27] K. Wang and S. Shen. Mvdepthnet: Real-time multiview depth estimation neural network. In *2018 International Conference on 3D Vision (3DV)*, pages 248–257. IEEE, 2018.

[28] K. Wilson and N. Snavely. Robust global translations with 1dsfm. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2014.

[29] Q. Xu and W. Tao. Multi-view stereo with asymmetric checkerboard propagation and multi-hypothesis joint view selection. *arXiv preprint arXiv:1805.07920*, 2018.

[30] Y. Yao, Z. Luo, S. Li, T. Fang, and L. Quan. Mvsnet: Depth inference for unstructured multi-view stereo. *European Conference on Computer Vision (ECCV)*, 2018.

[31] E. Zheng, E. Dunn, V. Jojic, and J. Frahm. Patchmatch based joint view selection and depthmap estimation. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1510–1517, June 2014.