HAL open science

Deep Multi-View Stereo Gone Wild

Francois Darmon, Benedicte Bascle, Jean-Clement Devaux, Pascal Monasse, Mathieu Aubry

To cite this version:

Francois Darmon, Benedicte Bascle, Jean-Clement Devaux, Pascal Monasse, Mathieu Aubry. Deep Multi-View Stereo Gone Wild. 2021 International Conference on 3D Vision (3DV), Dec 2021, London, United Kingdom. pp.484-493, 10.1109/3DV53792.2021.00058 . hal-04315771

HAL Id: hal-04315771

https://hal.science/hal-04315771v1

Submitted on 30 Nov 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Deep Multi-View Stereo Gone Wild

François Darmon[1,2], Bénédicte Bascle[1], Jean-Clément Devaux[1], Pascal Monasse[2], and Mathieu Aubry[2]

[1]Thales LAS France
[2]LIGM (UMR 8049), École des Ponts, Univ. Gustave Eiffel, CNRS, Marne-la-Vallée, France
http://imagine.enpc.fr/~darmonf/wild_deep_mvs/

Abstract

Deep multi-view stereo (MVS) methods have been developed and extensively compared on simple datasets, where they now outperform classical approaches. In this paper, we ask whether the conclusions reached in controlled scenarios are still valid when working with Internet photo collections. We propose a methodology for evaluation and explore the influence of three aspects of deep MVS methods: network architecture, training data, and supervision. We make several key observations, which we extensively validate quantitatively and qualitatively, both for depth prediction and complete 3D reconstructions. First, complex unsupervised approaches cannot train on data in the wild. Our new approach makes it possible with three key elements: upsampling the output, softmin based aggregation and a single reconstruction loss. Second, supervised deep depthmap-based MVS methods are state-of-the art for reconstruction of few internet images. Finally, our evaluation provides very different results than usual ones. This shows that evaluation in uncontrolled scenarios is important for new architectures.

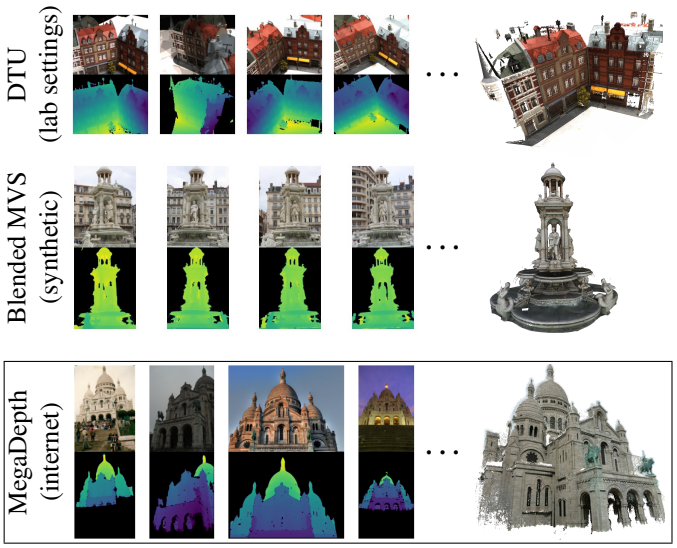

Figure 1: We apply deep multi-view stereo networks to internet images and study the influence of architecture, supervision and training data over the quality of the reconstructed 3D models.

1. Introduction

Deep Learning holds promises to improve Multi-View Stereo (MVS), i.e. the reconstruction of a 3D scene from a set of images with known camera parameters. However, deep MVS algorithms have mainly been developed and demonstrated in controlled settings, e.g. using pictures taken with a single camera, at a given time, specifically for 3D reconstruction. In this paper, we point out the challenges in using such approaches to perform 3D reconstructions "in the wild" and propose an evaluation procedure for such a task. We analyze quantitatively and qualitatively the impact of network architecture and training for 3D reconstructions with a varying number of images. We give particular attention to unsupervised methods and propose a new approach which performs on par with state of the art while being much simpler.

The reference dataset for the development of the vast majority of deep MVS approaches [35, 36, 34, 3, 39, 21] has been the DTU [12] dataset, consisting of 80 small scenes in sequences of 49 to 64 images. It was acquired in a lab, together with a reference 3D model obtained with a structured light scanner. While the scale of this dataset makes it suitable to train deep networks, it corresponds to a relatively simple reconstruction scenario, where many images taken in similar conditions are available. Recently, deep MVS approaches have started to evaluate their performance on larger outdoor datasets, such as *Tanks and Temples* [18] (T&T) and ETH3D [25]. For example, VisMVSNet [42] outperformed all classical methods on the T&T dataset. Yao *et al.* [37] have recently proposed a synthetic dataset, BlendedMVS, specifically designed to train deep MVS approaches to gen-

eralize to such outdoor scenarios. We aim to go further and evaluate whether deep MVS networks can generalize to the more challenging setting of 3D reconstruction from Internet image collections. We are particularly interested in the challenging case with few images available for the reconstruction, where classical approaches struggle to reconstruct dense point clouds, and we propose an evaluation protocol for such a setup.

Such a generalization to datasets with different properties, where obtaining ground truth measurements is challenging, has been a major motivation for unsupervised deep MVS approaches [4, 17, 11, 31]. We found however that many of these methods cannot train the network on real data. We thus propose a simpler alternative providing results on par with the best competing methods. It relies on minor modifications of the standard MVSNet [35] architecture and is simply trained by comparing the images warped using the predicted depth.

Our experiments give several interesting insights. First, we show that existing unsupervised method are not suited for images in the wild but our proposed method is. Second, we show that depthmap-based MVS methods are state-of-the-art for reconstruction of few internet images. Third, in the wild evaluation provides very different results than evaluation on DTU and T&T. We therefore advocate for its use when developing new architectures. Our main contributions are:

- We introduce a new experimental setup[1] for comparing different algorithms on images in the wild.
- We use this protocol to compare methods along three axes: network supervision, network architecture and training dataset.
- We show that existing unsupervised methods fail on images in the wild and thus we introduce a new unsupervised method.

2. Related Work

The goal of MVS is to reconstruct the 3D geometry of a scene from a set of images and the corresponding camera parameters. These parameters are assumed to be known, e.g., estimated using structure-from-motion [23]. Classical approaches to MVS can be roughly sorted in three categories [5]: direct reconstruction [6] that works on point clouds, volumetric approaches [26, 19] that use a space discretization, and depth map based methods [7, 24] that first compute a depth map for every image then fuse all depth maps into a single reconstruction. The latter approach decomposes MVS in many depth estimation tasks, and is typically adopted by deep MVS methods.

Deep Learning based MVS Deep Learning was first applied to patch matching [41, 40] and disparity estimation [16]. Several approaches tackled the MVS problem with deep learning, using either a volumetric approach [13, 22] or sparse point cloud densification [28]. MVSNet [35] introduced a network architecture specific to depth map based MVS and end-to-end trainable. Further works improved the precision, speed and memory efficiency of MVSNet. [36, 33] propose a recurrent architecture instead of 3D convolutions. MVS-CRF [32] uses a CRF model with learned parameters to infer depth maps. Many approaches introduce a multi-step framework where a first depth map is produced with an MVSNet-like architecture, then the results are refined either with optimization [36, 39], graph neural networks [3, 2] or multiscale neural networks [9, 34, 42, 38]. Another line of work aims at improving the per-view aggregation by predicting aggregation weights [21, 42]. All these methods were evaluated in controlled settings and it is unclear how they would generalize to internet images.

Unsupervised training is a promising approach for such a generalization. [17, 4, 11, 31] propose unsupervised approaches using several photometric consistency and regularization losses. However, to the best of our knowledge, these methods were never evaluated on Internet-based image datasets. Besides, their loss formulation is composed of many terms and hyperparameters, making them difficult to reproduce and likely to fail on new datasets. In this paper we propose a simpler unsupervised training approach, which we show to perform on par with these more complex ones on standard data, and generalizes to Internet images.

Datasets and evaluation There exist many datasets for 3D reconstruction. Most use active sensing [8, 18, 29]. Synthetic ones [1] provide an alternative way to acquire large-scale data but they cannot be used to predict reliable performance on real images. Specific datasets were introduced for MVS, such as ETH3D [25], Tanks & Temples (T&T) [18] or DTU [12] datasets. Most deep MVS methods train on the DTU dataset [12] and evaluate generalization on T&T [18]. DTU provides large-scale data but in a laboratory setting. The limited generalization capability of networks trained on DTU was highlighted in [37]. In this paper, a network trained on the synthetic dataset BlendedMVS was shown to perform better on T&T than its counterpart trained on MegaDepth [20]. [37] claims that this is due to the lack of accuracy on MegaDepth depth maps. We also compare training deep MVS approaches on DTU, BlendedMVS and MegaDepth but for MVS reconstruction from Internet images, and additionally compare performances obtained in supervised and unsupervised settings.

3. MVS Networks in the Wild

In this section, we present standard deep depthmap-based MVS architectures and training which we compare, and propose some modifications, including our new unsupervised

[1] Our PyTorch code and data are available at https://github.com/fdarmon/wild_deep_mvs.

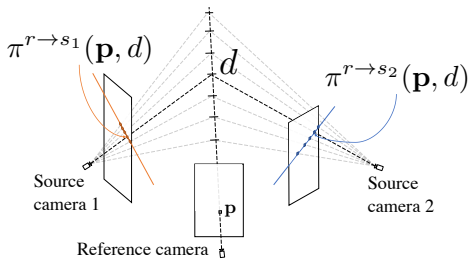

Figure 2: Notations: A pixel $\mathbf{p}$ in reference frame is projected on source frames at a position depending on its depth.

training, to better handle images in the wild.

3.1. Network Architectures

Overview and notations Starting from a set of calibrated images, deep MVS methods typically try to estimate a depth map for each image. The depth maps are then fused into a 3D point cloud using an independent method. The MVS network will take as input a *reference image* $\mathbf{I}^r$ and a set of *source images* $\mathbf{I}^{s_1} \dots \mathbf{I}^{s_N}$, and try to predict the depth $\mathbf{D}^r$ of the reference image.

Multi-view stereo architectures build on the idea that depth can be used to establish correspondences between images. More formally, as visualized in Figure 2, we write $\pi^{r\to s_k}(\mathbf{p}, d)$ the 2D position of the projection in the camera associated to $\mathbf{I}^{s_k}$ of the 3D point at depth d on the ray associated to the pixel $\mathbf{p} \in \mathbb{R}^2$ in $\mathbf{I}^r$. If d is the ground truth depth of the pixel $\mathbf{p}$ then the appearance of the reference image should be consistent with all the appearances of the source images $\mathbf{I}^{s_k}$ at pixels $\pi^{r\to s_k}(\mathbf{p}, d)$.

Similar to the classic plane sweep stereo methods, the best performing deep MVS methods explicitly sample candidate depths and build a cost volume that computes for every pixel a score based on the correspondences given in every view by the candidate depths. This score can be seen as the cost of assigning a given depth to a pixel. The depth map can be extracted from the cost volume by looking at the minimum in the depth axis. MVSNet [35] was the first deep architecture implementing this idea.

MVSNet MVSNet starts by extracting a feature representation of each input image using a shared convolutional neural network (CNN). We write $\mathbf{F}^r$ the feature map of the reference image and $\mathbf{F}^{s_1} \dots \mathbf{F}^{s_N}$ the feature maps of the source images. The network also takes as input the depth range $[d_{\min}, d_{\max}]$ to consider. It is usually provided in standard datasets, and we explain in section 4.1 how we defined them in the context of reconstruction in the wild. The information from the feature maps are then aggregated into a 3D cost volume C for depth d regularly sampled from $d_{\min}$ to $d_{\max}$.

$$\mathbf{C}[\mathbf{p}, d] = \phi\left(\mathbf{F}^r[\mathbf{p}]; \{\mathbf{F}^{s_k}[\pi^{r\to s_k}(\mathbf{p}, d)], k = 1 \dots N\}\right), \tag{1}$$

where $\mathbf{F}[\cdot]$ denotes the interpolation of the feature map $\mathbf{F}$ and ϕ is an aggregation function. In MVSNet the channel-wise variance is used for aggregation:

$$\phi(\mathbf{f}_r; \{\mathbf{f}_{s_1}, \dots \mathbf{f}_{s_N}\}) = \mathrm{var}(\mathbf{f}_r, \mathbf{f}_{s_1}, \dots \mathbf{f}_{s_N}). \tag{2}$$

The cost volume only encodes local information. A further step of 3D convolutions is then performed to propagate this local information. The depth map is finally extracted from the refined cost volume using a differentiable argmin operation.

Softmin-based aggregation The variance aggregation of MVSNet can be problematic when applied to images in the wild. Indeed, we can expect large viewpoint variations and many occlusions. Both would lead to large appearance differences in the pixels corresponding to the actual depth, and such outlier images would have a strong influence on the variance. Instead, we propose an alternative aggregation:

$$\phi(\mathbf{f}_r; \{\mathbf{f}_{s_1}, \dots \mathbf{f}_{s_N}\}) = \frac{\sum_{k=1}^{N} \exp(-\lambda \|\mathbf{f}_r - \mathbf{f}_{s_k}\|^2)(\mathbf{f}_r - \mathbf{f}_{s_k})^2}{\sum_{k=1}^{N} \exp(-\lambda \|\mathbf{f}_r - \mathbf{f}_{s_k}\|^2)}, \tag{3}$$

where λ is a learnable parameter. This score has two main differences with the variance aggregation. First, it breaks the symmetry between the reference and source images, each source feature is compared to the reference instead of the mean feature. Second, in our aggregation, the contribution of each source feature is weighted using a softmin weight. The intuition is that the features most similar to the ones of the reference image should be given higher weight compared to features that are completely different, which might be outliers. We will show in our experiments that such an aggregation is particularly beneficial in the presence of a high number of potential source images.

Multi-scale architectures. A known limitation of MVSNet is its large memory consumption. The consequence is that the resolution, both for image and depth is limited by the available memory. Several methods have been proposed to increase the resolution without increasing the memory requirements. Among them are the multiscale approaches that first scan coarsely the whole depth range using low resolution feature maps then refine the depth at higher resolutions. We used two successful multi-scale approaches: CVP-MVSNet [34] and Vis-MVSNet [42]. CVP-MVSNet starts from a downsampled version of the images to predict a coarse depth map using the same architecture as MVSNet. Then, at each scale, this depth map is upsampled and refined using the same network until full scale prediction is reached. This architecture is very light and can work with a variable number of scales. Vis-MVSNet uses a fixed number of scales with a different network for each scale. Moreover,

contrary to CVP-MVSNet, the lowest scale features are extracted from the full scale image using strided convolutions. Finally, Vis-MVSNet leverages a specific multi-view aggregation. It first predicts a confidence for each source image using a Bayesian framework [15], then uses the confidence to aggregate the pairwise cost volumes associated with every source image. While this aggregation is complex, we believe it is one of the key ingredients that allows Vis-MVSNet to generalize well to Internet images.

3.2. Training

3.2.1 Supervised Training.

We first consider the case where a partial ground truth depth map D_{GT} is available, together with a binary mask M_{GT} encoding pixels for which the ground truth is valid. Supervised training is typically performed using the $\ell 1$ loss between the predicted depth map D and the ground truth depth map D_{GT}, weighted by the mask M_{GT} and normalized by the scene's depth range,

$$l(D, D_{GT}, M_{GT}) = \frac{\|M_{GT}(D - D_{GT})\|}{(d_{\max} - d_{\min})\|M_{GT}\|}. \quad (4)$$

3.2.2 Unsupervised Training

Motivated by the difficulty to obtain ground truth depth maps, unsupervised approaches attempt to rely solely on camera calibration. Approaches proposed in the literature typically combine a photometric consistency term with several additional losses. Instead, we propose to rely exclusively on the structural similarity between the reference image and the warped source images.

Similarly to MVS^2 [4], we adopt a multi-reference setting where each image in a batch is successively used as a reference image. Let us consider a batch of images $\mathcal{I} = \{\mathbf{I}^1, ..., \mathbf{I}^{N+1}\}$ and the associated depth maps $\mathcal{D} = \{D^1, ..., D^{N+1}\}$ predicted by our MVS network. We write $\mathbf{W}^{b\to a}(D^a)$ the image resulting from warping $\mathbf{I}^b$ on $\mathbf{I}^a$ using the estimated depth map D^a:

$$\mathbf{W}^{b\to a}(D^a)[\mathbf{p}] = \mathbf{I}^b[\pi^{a\to b}(\mathbf{p}, D^a[\mathbf{p}])]. \quad (5)$$

Note that this warped image is not defined everywhere, since the 3D point defined from the pixel $\mathbf{p}$ and the depth $D^a[\mathbf{p}]$ in $\mathbf{I}^a$ might project outside of $\mathbf{I}^b$. Even when the warped image is defined, it might not be photoconsistent in case of occlusion. We thus follow MVS^2 [4] and define an occlusion aware mask in $\mathbf{I}^a$, $M^{b\to a}(D^a, D^b)$, based on the consistency between the estimated depths D^a and D^b in I^a and I^b (see [4] for details).

We then define the loss for the batch as:

$$l(\mathcal{I}, \mathcal{D}) = \sum_{r=1}^{N} \frac{\sum_{s\neq r} \|M^{s\to r}(D^r, D^s)\text{SSIM}\left(\mathbf{I}_r, \mathbf{W}^{s\to r}(D^r)\right)\|}{\sum_{s\neq r} \|M^{s\to r}(D^r, D^s)\|} \quad (6)$$

where SSIM is the structural similarity [43], which enables to compare images acquired in very different conditions. To avoid the need for further regularization outlined by other unsupervised approaches, we use the same approach as [27]: the depth maps, which are predicted at the same resolution as the CNN feature maps, are upsampled to the resolution of the original images before computing the loss.

Note that this loss is much simpler than in MVS^2 [4], a combination of $\ell 1$ color loss, $\ell 1$ of the image gradients, SSIM and census transform. However, we show that it leads to results on par with state of the art and can be successfully applied to train a network from Internet image collections.

4. Benchmark of Deep MVS Networks in the Wild

To analyze the performance of different networks, we define training and evaluation dataset, evaluation metrics, as well as reference choices for several steps of the reconstruction pipeline, including reference depth to consider and a strategy to merge estimated depths into a full 3D model.

4.1. Data

Training datasets We experiment training on the DTU [12], BlendedMVS [37] and MegaDepth [20] datasets. Training an MVS network requires images with their associated calibration and depth maps. It also requires to select the set of source images for each reference image and the depth range for building the cost volume. DTU is a dataset captured in laboratory conditions with structured light scanner ground truth. It was preprocessed, organized, and annotated by Yao *et al.* [35], who generated depth maps by meshing and rendering the ground truth point cloud. BlendedMVS was introduced in order to train deep models for outdoor reconstruction. It is composed of renderings of 3D scenes blended with real images to increase realism. Depth range and list of source views are given in the dataset. These datasets have reliable ground truth but there is a large domain gap between their images and Internet images. Thus, we also experiment training on MegaDepth, which is composed of many Internet images with pseudo ground truth depth and sparse 3D models obtained with COLMAP [24, 24]. We generate a training dataset from MegaDepth, excluding the training scenes of the Image Matching Benchmark [14] which we will use for evaluation. We selected sets of training images by looking at the sparse reconstruction. We randomly sample a reference image and two source images uniformly among all the images that have more than 100 reconstructed 3D points, with

a triangulation angle above 5 degree, in common with the reference. Once these sets of three images are sampled, we select the 3D points observed by at least three images and use their minimum and maximum depth as $d_{\min}$ and $d_{\max}$.

Depth map evaluation To evaluate depth map prediction, we used the synthetic BlendedMVS scenes, as well as the images from YFCC-100M [30, 10] used in the image matching benchmark of [14]. Indeed, [14] provides 14 sets of few thousands of images with filtered ground truth depth maps obtained with COLMAP [23, 24].

3D reconstruction evaluation Because our ultimate goal is not to obtain depth maps but complete 3D reconstructions, we also select data to evaluate it. We used the standard DTU test set, but also constructed an evaluation setting for images in the wild. We start from the verified depth maps of the Image Matching Workshop [14]. We fuse them into a point cloud with COLMAP fusion using very conservative parameters: reprojection error below half a pixel and depth error below 1%. The points that satisfy such conditions in 20 views are used as reference 3D models. We manually check that the reconstructions were of high enough quality. The scenes *Westminster Abbey* and *Hagia Sophia interior* were removed since their reconstruction was not satisfactory. We then randomly selected sets of 5, 10, 20, and 50 images as test image sets and compare the reconstructions using only the images in these sets to the reference reconstruction obtained from the reference depth maps of a very large number of images. While the reference reconstructions are of course biased toward COLMAP reconstructions, we argue that the quality and extension of the reconstructions obtained from the small image sets is much lower than the reference one, and thus evaluating them with respect to the reference makes sense. Moreover, we never observed any case where part of the scene was reconstructed using deep MVS approaches on the test image sets and not in the reference reconstruction.

4.2. Metrics

Depth estimation We follow the same evaluation protocol as BlendedMVS [37]. The predicted and ground truth depth maps are first normalized by $(d_{\max} - d_{\min})/128$ which makes the evaluation comparable for images seeing different depth ranges. We use the following three metrics: end point error (EPE), the mean absolute error between the inferred and the ground truth depth maps; $\mathbf{e_1}$, the proportion in % of pixels with an error larger than 1 in the scaled depth maps; $\mathbf{e_3}$, the proportion in % of pixels with an error larger than 3.

3D reconstruction On DTU [12] we follow the standard evaluation protocol and use as metrics: Accuracy, the mean distance from the reconstruction to ground truth, completeness from ground truth to reconstruction, and the overall metric defined as the average of the previous two.

For our evaluation in the wild, we use the same metrics as T&T [18] : precision, recall and F-score at a given threshold. The reference reconstruction is known only up to a scale factor therefore the choice of such threshold is not straightforward. Let D_k be the ground truth depth map of image I_k, we define the threshold as:

$$t = \underset{k}{\text{median}} \left(\underset{\|\mathbf{p}-\mathbf{p}'\|=2}{\text{median}} \|D_k[\mathbf{p}]\mathbf{p} - D_k[\mathbf{p}']\mathbf{p}'\| \right). \quad (7)$$

4.3. Common Reconstruction Framework

To fairly compare different deep MVS approaches for 3D reconstruction, we need to fuse the predicted depth maps in a consistent way. Several fusion methods exist, each with variations in the point filtering and the parameters used. For DTU reconstructions, we use fusibile [7] with the same parameters for all compared methods. Since fusibile is only implemented for fixed resolution images, we use a different method for Internet images. We use the fusion procedure of COLMAP [24] with a classic pre-filtering step similar to the one used in MVSNet: we only keep depths consistent in three views. The consistency criterion is based on three thresholds: the reprojected depth must be closer than 1% to original depth, the reprojection distance must be less than 1 pixel, and the triangulation angle must be above $1°$.

5. Experiments

We first provide an ablation study of our unsupervised approach and compare it with state of the art unsupervised methods (Section 5.1). Then, we compare the different network architecture and supervisions for depth map estimation (Section 5.2) and full 3D reconstruction (Section 5.3).

5.1. Unsupervised Approach Analysis

We validate and ablate our unsupervised approach on both DTU 3D reconstruction and YFCC depth map estimation in Table 1. We test the importance of upsampling the depth map as a way to regularize the network, compare variance and softmin aggregation functions as well as results with and without occlusion aware masking. We also compare our approach with existing work.[2] The main observation is that upsampling is the key factor to obtain meaningful results. Softmin aggregation is consistently better than variance for both datasets and evaluations. Occlusion masking also consistently improves the results, by a stronger margin on YFCC where occlusions are expected to be more common. Using upsampling and occlusion masking, our simple training loss is on par with state of the art unsupervised methods on DTU but is is much better for YFCC trainings where competing method have poor results.

[2] We use open source implementations of M3VSNet and JDACS provided by the authors.

Table 1: Ablation study of unsupervised learning. We compare networks trained with and without upsampling (Up.) of the predicted depth map, with and without occlusion masking (Occ.) and using as aggregation function (ϕ) either Variance aggregation (V, eq. 2) or our Softmin aggregation (S, eq. 3).

ϕ	Up.	Occ.	DTU reconstruction			YFCC depth maps		
			Prec.	Comp.	Over.	EPE	e_1	e_3
V			1.000	0.803	0.901	34.74	91.88	80.99
V	✓		0.614	0.580	0.597	21.68	67.48	48.44
S	✓		**0.607**	0.560	0.584	21.88	66.39	46.61
V	✓	✓	0.610	0.545	0.578	18.75	63.07	43.17
S	✓	✓	0.608	**0.528**	**0.568**	**18.22**	**61.97**	**41.34**
Unsup. MVSNet[17]			0.881	1.073	0.977			
MVS^2 [4]			0.760	**0.515**	0.637			
M^{3}VSNet [11]			0.636	0.531	0.583	40.57	82.45	69.91
JDACS [31]			**0.571**	**0.515**	**0.543**	34.37	92.36	80.45

5.2. Depth Map Prediction

In Table 2 we compare different approaches for depth prediction on the YFCC data and BlendedMVS validation set. We study different architectures, the standard MVSNet architecture with either the standard variance aggregation or our proposed softmin aggregation and two state-of-the-art multiscale architecture CVP-MVSNet [34] and Vis-MVSNet [42].[3] We compare results for networks trained on DTU, BlendedMVS (B) and MegaDepth (MD) with either the supervised $\ell 1$ loss or our occlusion masked unsupervised loss (except for BlendedMVS which we only use in the supervised setting since it is a synthetic dataset designed specifically for this purpose).

MVSNet architecture As expected, the best performing networks on a given dataset are the ones trained with a supervised loss on the same dataset. Confirming our analysis in the unsupervised setup, we can also see that MVSNet modified with our softmin-based aggregation systematically outperforms the variance-based aggregation. The results for unsupervised settings are more surprising. First, networks trained on DTU in the unsupervised setting generalize better both to blended-MVS and YFCC, hinting that unsupervised approaches might lead to better generalization even without changing the training data. Also, the unsupervised network trained on MegaDepth performs better on YFCC than the supervised networks trained on BlendedMVS, outlining again the potential of unsupervised approaches.

Multi-scale architectures CVP-MVSNet, which is our best performing architecture on DTU (see next section) and achieves the second best results in terms of e_1 error on BlendedMVS, did not achieve satisfactory results on YFCC images even when supervised on MegaDepth. This may be because at the lowest resolution the architecture takes as input a low resolution downsampled version of the image, on which correspondences might be too hard to infer for images taken in very different conditions.

[3]For both CVP-MVSNet and Vis-MVSNet, we adapted the public implementation provided by the authors

Table 2: Direct depth map evaluation: Comparison of architectures MVSNet [35], our MVSNet with softmin aggregation (MVSNet-S), CVP-MVSNet [34] and Vis-MVSNet [42], trained on BlendedMVS (B), DTU or MegaDepth (MD), with or without depth supervision (Sup.).

Archi	Train	Sup.	BlendedMVS			YFCC		
			EPE	e_1	e_3	EPE	e_1	e_3
MVSNet	B	✓	**1.49**	**21.98**	**8.32**	21.56	67.93	49.75
	DTU	✓	4.90	39.25	22.01	32.41	79.74	65.42
	DTU		3.94	31.68	16.61	24.61	70.77	53.44
	MD	✓	2.29	30.08	12.37	**17.61**	**62.54**	43.09
	MD		3.24	32.94	15.25	18.75	63.07	**42.17**
MVSNet-S	B	✓	**1.35**	**25.91**	**8.55**	20.98	69.57	49.86
	DTU	✓	4.18	42.07	19.87	27.18	78.25	60.53
	DTU		4.05	32.50	16.01	21.07	68.64	50.35
	MD	✓	1.91	32.91	11.93	**15.57**	**60.44**	**38.18**
	MD		2.88	33.44	14.14	18.22	61.97	41.34
CVP-MVSNet	B	✓	**1.90**	**19.73**	**10.24**	40.07	85.88	76.25
	DTU	✓	10.99	46.79	35.59	73.69	95.29	90.69
	DTU		5.25	29.33	19.77	45.36	89.81	82.17
	MD	✓	3.07	24.33	14.40	34.39	**78.49**	**66.57**
	MD		3.39	21.67	12.94	**32.74**	78.92	66.99
Vis-MVSNet	B	✓	**1.47**	**18.47**	**7.59**	19.60	64.98	46.38
	DTU	✓	3.70	30.37	18.16	27.46	72.89	58.37
	DTU		7.22	37.03	20.75	38.32	73.24	56.17
	MD	✓	2.05	22.21	10.14	**16.01**	**56.71**	**38.20**
	MD		3.88	31.64	17.50	19.21	66.27	47.34

On the contrary, Vis-MVSNet supervised on MegaDepth data is the best performing approach in terms of e_1 error. However, its performance is worse in the unsupervised setting compared to MVSNet. This is likely explained by the complexity of the loss function of Vis-MVSNet. In the original paper, the loss combines pairwise estimated depth maps and final depth maps. The pairwise loss is computed in a Bayesian setting [15] and our direct adaptation of this loss by replacing $\ell 1$ with photometric loss might be too naive.

Number of source views Until now, all experiments were performed using one reference view and four source views as input to the network at test time. However, in real scenarios, one might have access to more views, and it is thus important for the network to benefit from an increased number of views at inference time. We thus investigate the use of more source images at test time, still training with two source images only, and report the results in Figure 3. The results for MVSNet architecture with variance-based aggregation get worse as the number of source views is increased. This may be because the relevant information in the cost volume is more likely to be masked by noise when the number of source images increases. On the contrary, the performances of our softmin-based aggregation or Vis-MVSNet aggregation improve when using a higher number of source views.

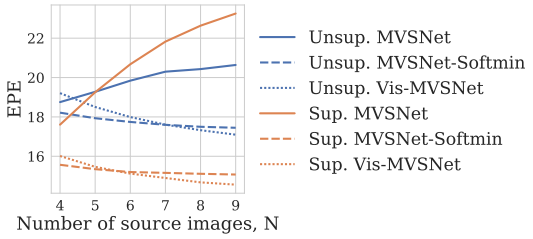

Figure 3: EPE on YFCC depth map estimation as a function of the number of source images N for networks trained on MegaDepth, supervised (Sup.) or unsupervised (Unsup.). Contrary to the other architectures MVSNet with variance aggregation does not benefit from using more images.

5.3. 3D Reconstruction

We now evaluate the quality of the full 3D reconstruction obtained with the different architectures. The predicted depth maps are combined with a common fusion algorithm and the evaluation is performed on the fused point cloud. We first report the results on DTU in Table 3. Interestingly, our softmin-based adaptation degrades performance when training on DTU in a supervised setting, but not in an unsupervised setting. Also note that the performance on DTU, and in particular the completeness of the reconstruction, is largely improved when the networks are trained on MegaDepth data. The best performance is even obtained in an unsupervised setting with CVP-MVSNet. However, this network performs poorly on YFCC data.

Finally, we evaluate the quality of the 3D reconstructions obtained from small sets of YFCC images. For fair comparison with COLMAP [24], the depth maps are evaluated with up to 19 source images for these experiments, except when using the original MVSNet architecture with variance aggregation, which does not benefit from using more images and which we continue to use with four source images. Since depth estimation on YFCC images completely failed with CVP-MVSNet as well as with networks trained on DTU, we did not perform reconstructions in these cases. Quantitative results are reported in Table 4 and qualitative results are shown in Figure 4.

Most of the trends observed for depth map prediction can also be observed in this experiment. In particular, networks trained on MegaDepth perform better than those trained on BlendedMVS, our softmin-based aggregation improves results for the MVSNet architecture, in particular when many views are available and in the unsupervised setting, and Vis-MVSNet outperforms MVSNet in the supervised setting.

A new important observation is that the unsupervised MVSNet performs better than the supervised version, especially for the recall metric. This can also be seen in Figure 4 where the reconstructions appear more complete in the unsupervised setting. We argue that this is a strong result that advocates for the development of unsupervised approaches suitable for the more advanced architectures. Interestingly, this result was not noticeable in our depth map evaluation.

Table 3: 3D reconstruction evaluation on DTU. Comparison of trainings on BlendedMVS, MegaDepth and DTU for various architectures with or without supervision ('Sup.' column).

Archi.	Training data	Sup.	DTU reconstructions Acc.	Comp.	Overall
MVSNet [35]	Blended	✓	0.487	0.496	0.491
	DTU	✓	**0.453**	**0.488**	**0.470**
	DTU		0.610	0.545	0.578
	MD	✓	0.486	0.547	0.517
	MD		0.689	0.645	0.670
MVSNet [35] softmin	Blended	✓	0.631	0.738	0.684
	DTU	✓	**0.598**	0.531	**0.564**
	DTU		0.609	**0.528**	0.568
	MD	✓	0.625	0.548	0.586
	MD		0.690	0.614	0.652
CVP-MVSNet [34]	Blended	✓	0.364	0.434	0.399
	DTU	✓	0.396	0.534	0.465
	DTU		**0.340**	0.586	0.463
	MD	✓	0.360	0.376	0.368
	MD		0.364	**0.370**	**0.367**
Vis-MVSNet [42]	Blended	✓	0.504	0.411	0.458
	DTU	✓	**0.434**	0.478	0.456
	DTU		0.456	0.596	0.526
	MD	✓	0.438	**0.345**	**0.392**
	MD		0.451	0.873	0.662

Another significant observation is that Vis-MVSNet trained with supervision on MegaDepth quantitatively outperforms COLMAP when using very small image sets (5 or 10 images) with both higher precision and higher recall. This is also a very encouraging result, showing that Deep MVS approaches can provide competitive results in challenging conditions. However, one should note that this quantitative result is not obvious when looking at the qualitative results where either COLMAP or Vis-MVSNet achieves better looking results, depending on the scene.

6. Conclusion

We have presented a study of the performance of deep MVS methods on Internet images, a setup that, to the best of our knowledge has never been explored. We discussed the influence of training data, architecture and supervision. For this last aspect, we introduced a new unsupervised approach which outperforms state of the art. Our analysis revealed several interesting observations, which we hope will encourage systematic evaluation of deep MVS approaches with challenging Internet images and stimulate research on unsupervised approaches, to ultimately have deep learning bring clear performance boosts in this important scenario.

Acknowledgments

This work was supported in part by ANR project EnHerit ANR-17-CE23-0008 and was performed using HPC resources from GENCI–IDRIS 2020-AD011011756. We thank Tom Monnier, Michaël Ramamonjisoa and Vincent Lepetit for valuable feedback.

Table 4: 3D reconstruction evaluation on our 'in the wild' benchmark. We compare several architectures trained with and without supervision ('Sup.' column) on BlendedMVS (Blended) and MegaDepth (MD). We report the best between using four source views (*) and all source views (up to 19).

Archi.	Training data	Sup.	5 images			10 images			20 images			50 images		
			Prec.	Rec.	F-score	Prec.	Rec.	F-score	Prec.	Rec.	F-score	Prec.	Rec.	F-score
MVSNet [35]	Blended	✓	**84.68**	11.51	0.1916	**91.37***	23.27*	0.3583*	**91.46***	36.46*	0.5173*	**93.40***	51.50*	0.6608*
	MD	✓	84.31	14.84	0.2387	91.05*	27.62*	0.4151*	89.94*	39.55*	0.5460*	92.07*	52.69*	0.6675*
	MD		80.44	**15.31**	**0.2439**	90.17*	**31.05***	**0.4546***	88.15*	**43.84***	**0.5822***	92.22*	**57.27***	**0.7039***
MVSNet [35] softmin	Blended	✓	74.67	2.14	0.0395	83.66	11.68	0.1897	83.43	19.64	0.2974	**89.93**	37.85	0.5240
	MD	✓	**84.60**	11.70	0.1936	**88.65**	30.99	0.4443	**85.93**	43.16	0.5645	87.69	57.10	0.6870
	MD		82.02	**14.71**	**0.2371**	83.67	**37.63**	**0.5095**	79.09	**51.40**	**0.6144**	81.74	**63.66**	**0.7129**
Vis-MVSNet [42]	Blended	✓	87.43	13.19	0.2158	87.98	35.82	0.4987	85.66	53.65	0.6541	89.26	64.99	0.7503
	MD	✓	**91.44**	**21.72**	**0.3330**	**93.23**	**48.22**	**0.6305**	**90.25**	**64.40**	**0.7485**	**90.55**	**72.88**	**0.8067**
	MD		84.25	8.90	0.1491	84.19	32.45	0.4569	81.10	53.27	0.6352	86.13	68.03	0.7579
COLMAP [24]			89.95	17.30	0.2785	93.20	42.76	0.5797	94.88	62.68	0.7511	96.78	73.64	0.8346

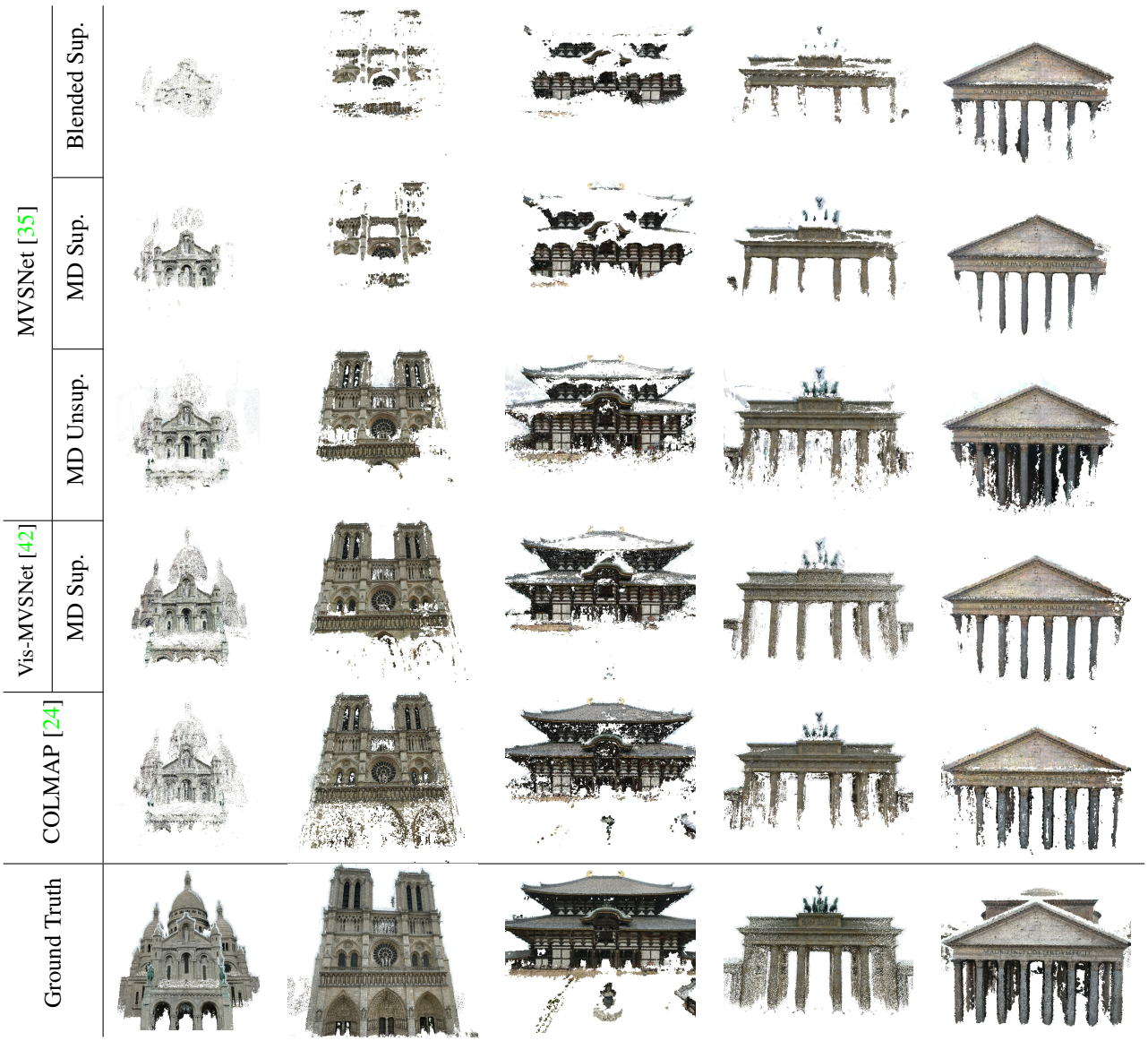

Figure 4: Comparison of reconstructions from ten images using different architectures trained on BlendedMVS (Blended) or MegaDepth (MD), supervised (Sup.) or unsupervised (Unsup.).

References

[1] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black. A naturalistic open source movie for optical flow evaluation. In A. Fitzgibbon et al. (Eds.), editor, *European Conf. on Computer Vision (ECCV)*, Part IV, LNCS 7577, pages 611–625. Springer-Verlag, Oct. 2012. 2

[2] Rui Chen, Songfang Han, Jing Xu, et al. Visibility-aware point-based multi-view stereo network. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020. 2

[3] Rui Chen, Songfang Han, Jing Xu, and Hao Su. Point-based multi-view stereo network. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1538–1547, 2019. 1, 2

[4] Yuchao Dai, Zhidong Zhu, Zhibo Rao, and Bo Li. MVS^2: Deep unsupervised multi-view stereo with multi-view symmetry. In *2019 International Conference on 3D Vision (3DV)*, pages 1–8. IEEE, 2019. 2, 4, 6

[5] Yasutaka Furukawa and Carlos Hernández. Multi-view stereo: A tutorial. *Foundations and Trends® in Computer Graphics and Vision*, 9(1-2):1–148, 2015. 2

[6] Yasutaka Furukawa and Jean Ponce. Accurate, dense, and robust multiview stereopsis. *IEEE transactions on pattern analysis and machine intelligence*, 32(8):1362–1376, 2009. 2

[7] Silvano Galliani, Katrin Lasinger, and Konrad Schindler. Massively parallel multiview stereopsis by surface normal diffusion. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, December 2015. 2, 5

[8] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The KITTI dataset. *The International Journal of Robotics Research*, 32(11):1231–1237, 2013. 2

[9] Xiaodong Gu, Zhiwen Fan, Siyu Zhu, Zuozhuo Dai, Feitong Tan, and Ping Tan. Cascade cost volume for high-resolution multi-view stereo and stereo matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2495–2504, 2020. 2

[10] Jared Heinly, Johannes L Schonberger, Enrique Dunn, and Jan-Michael Frahm. Reconstructing the world* in six days*(as captured by the Yahoo 100 million image dataset). In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3287–3295, 2015. 5

[11] Baichuan Huang, Can Huang, Yijia He, Jingbin Liu, and Xiao Liu. M^{3}VSNet: Unsupervised multi-metric multi-view stereo network. *arXiv preprint arXiv:2005.00363*, 2020. 2, 6

[12] Rasmus Jensen, Anders Dahl, George Vogiatzis, Engin Tola, and Henrik Aanæs. Large scale multi-view stereopsis evaluation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 406–413, 2014. 1, 2, 4, 5

[13] Mengqi Ji, Juergen Gall, Haitian Zheng, Yebin Liu, and Lu Fang. Surfacenet: An end-to-end 3D neural network for multiview stereopsis. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2307–2315, 2017. 2

[14] Yuhe Jin, Dmytro Mishkin, Anastasiia Mishchuk, Jiri Matas, Pascal Fua, Kwang Moo Yi, and Eduard Trulls. Image matching across wide baselines: From paper to practice. *International Journal of Computer Vision*, 129(2):517–547, Oct 2020. 4, 5

[15] Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. 4, 6

[16] Alex Kendall, Hayk Martirosyan, Saumitro Dasgupta, Peter Henry, Ryan Kennedy, Abraham Bachrach, and Adam Bry. End-to-end learning of geometry and context for deep stereo regression. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 66–75, 2017. 2

[17] Tejas Khot, Shubham Agrawal, Shubham Tulsiani, Christoph Mertz, Simon Lucey, and Martial Hebert. Learning unsupervised multi-view stereopsis via robust photometric consistency. *CVPR workshop*, 2019. 2, 6

[18] Arno Knapitsch, Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. Tanks and Temples: Benchmarking large-scale scene reconstruction. *ACM Transactions on Graphics (ToG)*, 36(4):1–13, 2017. 1, 2, 5

[19] Kiriakos N Kutulakos and Steven M Seitz. A theory of shape by space carving. *International journal of computer vision*, 38(3):199–218, 2000. 2

[20] Zhengqi Li and Noah Snavely. Megadepth: Learning single-view depth prediction from internet photos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2041–2050, 2018. 2, 4

[21] Keyang Luo, Tao Guan, Lili Ju, Haipeng Huang, and Yawei Luo. P-MVSNet: Learning patch-wise matching confidence aggregation for multi-view stereo. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 10452–10461, 2019. 1, 2

[22] Zak Murez, Tarrence van As, James Bartolozzi, Ayan Sinha, Vijay Badrinarayanan, and Andrew Rabinovich. Atlas: End-to-end 3d scene reconstruction from posed images. In *ECCV*, 2020. 2

[23] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4104–4113, 2016. 2, 5

[24] Johannes L Schönberger, Enliang Zheng, Jan-Michael Frahm, and Marc Pollefeys. Pixelwise view selection for unstructured multi-view stereo. In *European Conference on Computer Vision*, pages 501–518. Springer, 2016. 2, 4, 5, 7, 8

[25] Thomas Schöps, Johannes L. Schönberger, Silvano Galliani, Torsten Sattler, Konrad Schindler, Marc Pollefeys, and Andreas Geiger. A multi-view stereo benchmark with high-resolution images and multi-camera videos. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1, 2

[26] Steven M Seitz and Charles R Dyer. Photorealistic scene reconstruction by voxel coloring. *International Journal of Computer Vision*, 35(2):151–173, 1999. 2

[27] Xi Shen, François Darmon, Alexei A Efros, and Mathieu Aubry. Ransac-flow: generic two-stage image alignment. In *16th European Conference on Computer Vision*, 2020. 4

[28] Ayan Sinha, Zak Murez, James Bartolozzi, Vijay Badrinarayanan, and Andrew Rabinovich. Deltas: Depth estimation by learning triangulation and densification of sparse points. In *ECCV*, 2020. 2

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

[30] Bart Thomee, David A Shamma, Gerald Friedland, Benjamin Elizalde, Karl Ni, Douglas Poland, Damian Borth, and Li-Jia Li. YFCC100M: The new data in multimedia research. *Communications of the ACM*, 59(2):64–73, 2016. 5

[31] Hongbin Xu, Zhipeng Zhou, Yu Qiao, Wenxiong Kang, and Qiuxia Wu. Self-supervised multi-view stereo via effective co-segmentation and data-augmentation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 2021. 2, 6

[32] Youze Xue, Jiansheng Chen, Weitao Wan, Yiqing Huang, Cheng Yu, Tianpeng Li, and Jiayu Bao. MVSCRF: Learning multi-view stereo with conditional random fields. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4312–4321, 2019. 2

[33] Jianfeng Yan, Zizhuang Wei, Hongwei Yi, Mingyu Ding, Runze Zhang, Yisong Chen, Guoping Wang, and Yu-Wing Tai. Dense hybrid recurrent multi-view stereo net with dynamic consistency checking. In *European Conference on Computer Vision*, pages 674–689. Springer, 2020. 2

[34] Jiayu Yang, Wei Mao, Jose M Alvarez, and Miaomiao Liu. Cost volume pyramid based depth inference for multi-view stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4877–4886, 2020. 1, 2, 3, 6, 7

[35] Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long Quan. Mvsnet: Depth inference for unstructured multi-view stereo. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 767–783, 2018. 1, 2, 3, 4, 6, 7, 8

[36] Yao Yao, Zixin Luo, Shiwei Li, Tianwei Shen, Tian Fang, and Long Quan. Recurrent MVSNet for high-resolution multi-view stereo depth inference. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5525–5534, 2019. 1, 2

[37] Yao Yao, Zixin Luo, Shiwei Li, Jingyang Zhang, Yufan Ren, Lei Zhou, Tian Fang, and Long Quan. BlendedMVS: A large-scale dataset for generalized multi-view stereo networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1790–1799, 2020. 1, 2, 4, 5

[38] Hongwei Yi, Zizhuang Wei, Mingyu Ding, Runze Zhang, Yisong Chen, Guoping Wang, and Yu-Wing Tai. Pyramid multi-view stereo net with self-adaptive view aggregation. In *European Conference on Computer Vision*, pages 766–782. Springer, 2020. 2

[39] Zehao Yu and Shenghua Gao. Fast-MVSNet: Sparse-to-dense multi-view stereo with learned propagation and Gauss-Newton refinement. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1949–1958, 2020. 1, 2

[40] Sergey Zagoruyko and Nikos Komodakis. Learning to compare image patches via convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4353–4361, 2015. 2

[41] Jure Zbontar and Yann LeCun. Computing the stereo matching cost with a convolutional neural network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1592–1599, 2015. 2

[42] Jingyang Zhang, Yao Yao, Shiwei Li, Zixin Luo, and Tian Fang. Visibility-aware multi-view stereo network. In *British Machine Vision Conference*, 2020. 1, 2, 3, 6, 7, 8

[43] Zhou Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004. 4