Supplementary Material for DepthInSpace: Exploitation and Fusion of Multiple Video Frames for Structured-Light Depth Estimation

Mohammad Mahdi Johari
Idiap Reserach Institute, EPFL
mohammad.johari@idiap.ch

Camilla Carta
ams OSRAM
camilla.carta@ams.com

François Fleuret
University of Geneva, EPFL
francois.fleuret@unige.ch

1. Notation

We use the same notation of the main paper here. We name our loss functions as photometric loss $\mathcal{L}_{ph}$, smoothness loss $\mathcal{L}_{s}$, multi-view loss $\mathcal{L}_{mv}$, and pseudo-ground truth loss $\mathcal{L}_{pgt}$. For the DIS-MF model, we assume there are only two frames and we intend to merge Frame j's feature map $\boldsymbol{\phi_j}$ into Frame i's feature map $\boldsymbol{\phi_i}$. $\boldsymbol{X_i}$ represents 3D point cloud of Frame i obtained using imperfect disparities and camera intrinsic parameters. We also define warped features $\boldsymbol{\phi_{j\to i}} = w^{j\to i}(\boldsymbol{\phi_j})$, and warped points $\boldsymbol{X_{j\to i}} = w^{j\to i}(\boldsymbol{X_j})$, where $w^{j\to i}(\cdot)$ stands for bilinear 2D warping via the optical flow $\boldsymbol{F_{i\to j}}$.

2. Validation Binary Masks for the Multi-View Loss Function

As explained in the main paper, our multi-view loss function is defined as:

$$\boldsymbol{\mathcal{L}_{mv}^{ij}} = \left| \left\langle \boldsymbol{X_i} - w^{j\to i}\big(\boldsymbol{T_{j\to i}} \times [\boldsymbol{X_j}, \vec{\mathbf{1}}]\big) \right\rangle_z \right| \times \boldsymbol{M'_{j\to i}} \tag{1}$$

where $\boldsymbol{T_{j\to i}} \in \mathbb{R}^{3\times 4}$ is the transformation matrix consisting of ego motion parameters, $\vec{\mathbf{1}}$ is an all one matrix, and $\langle\cdot\rangle_z$ operator returns the depth z of its input 3D vector. In Equation 1, $\boldsymbol{M'_{j\to i}}$ is a binary mask map validating warped points and preventing the network from being trained with noisy gradient information. In our DIS-SF model, two criteria must be met in order for a warped point to be indicated as valid. The first one is optical flow forward-backward consistency, suggested in [18, 10]:

$$\boldsymbol{M'_{FB}} = |\boldsymbol{F_{i\to j}} + w^{j\to i}(\boldsymbol{F_{j\to i}})|^2 < 0.01 \times (|\boldsymbol{F_{i\to j}}|^2 + |w^{j\to i}(\boldsymbol{F_{j\to i}})|^2) + 0.5 \tag{2}$$

which is exactly the binary mask map that is used in the fusion block of our DIS-MF model in the main paper. The second criterion excludes outlier points whose depths with respect to the camera position of the target frame has a considerable distance from the depths of their corresponding points in the target frame itself, *i.e.*:

$$\boldsymbol{M'_O} = \left| \left\langle \boldsymbol{X_i} - w^{j\to i}\big(\boldsymbol{T_{j\to i}} \times [\boldsymbol{X_j}, \vec{\mathbf{1}}]\big) \right\rangle_z \right| < \tau \tag{3}$$

where τ is a threshold set to 10 cm. Accordingly, the binary mask map in our DIS-SF model is defined as the intersection of these two criteria: $\boldsymbol{M'_{j\to i}} = \boldsymbol{M'_{FB}} \odot \boldsymbol{M'_O}$.

Instead, for the DIS-MF model, we define more strict and accurate criteria thanks to having access to imperfect disparities information. After checking optical flow forward-backward consistency in Equation 2, we check if the optical flow is consistent with the rigid flow derived from camera motion parameters and imperfect disparities. Therefore, this constraint excludes pixels that either are warped with an erroneous optical flow or contain a significant error in their initially predicted depths:

$$\boldsymbol{M'_{RF}} = \left| \langle \boldsymbol{K} \times \boldsymbol{X_i} \rangle_{uv} - \left\langle w^{j\to i}(\boldsymbol{K} \times \boldsymbol{T_{j\to i}} \times [\boldsymbol{X_j}, \vec{\mathbf{1}}]) \right\rangle_{uv} \right| < 1 \tag{4}$$

where $\boldsymbol{K}$ is the camera intrinsic matrix, $\vec{\mathbf{1}}$ is an all one matrix, and $\langle\cdot\rangle_{uv}$ operator denotes mapping the 3D points on the image plane by dividing the operands by their depth components z. We map the points on the image plane because we are

interested in taking into account the validity of the depth of the frame z_j independent of the depth of target frame z_i. Thus, if a particular warped point has an accurate initially predicted depth to some degree, but the target point's depth is inaccurate, the warped point is still considered valuable for participating in the loss function, and in fact, it encourages the network to correct the depth of the target point. The threshold in Equation 4 is set to 1, and it means that the warped point passes this criterion if it falls into its corresponding matched point's neighborhood of size one pixel on the target image plane.

Lastly, to prevent the network from being greedy in aggregating and fusing and also to encourage it to preserve sharp edges in the disparity map, we introduce another criterion that relates to the visual consistency of the warped and original normalized ambient images. This constraint specially excludes faulty warped points near the edges in the image and encourages the network to preserve the edges and geometry of the objects in the scene:

$$\boldsymbol{M'_{VC}} = |\boldsymbol{A_i} - w^{j \to i}(\boldsymbol{A_j})| < 0.01 \tag{5}$$

This criterion is defined on the normalized ambient images, so the 0.01 threshold in Equation 5 means points are accepted if their gray intensity levels differ less than one percent of the whole intensity level's range. Having these criteria together, we define the validation binary mask in our DIS-MF model as: $\boldsymbol{M'_{j \to i}} = \boldsymbol{M'_{FB}} \odot \boldsymbol{M'_{RF}} \odot \boldsymbol{M'_{VC}}$. An ablation study on the validation masks of DIS-MF is provided in Experiment 4 of Table 5 in Section 5.3.

3. Optical Flow Versus Back-Projection Projection Technique

As we briefly discussed in the main paper, while it is possible to find matched pixels using the back-projection projection technique, as it is used in CTD[12], we opted for utilizing direct optical flow predictions for finding matched pixels among frames.

Optical flow provides us with two advantages. Firstly, the back-projection projection technique uses the momentary estimated depth and camera motion parameters to find matched pixels. As a result, the errors in the predicted depths propagate through the algorithm and degrade its performance. Also, it causes rapid fluctuations of the optimization landscape of the loss function. In contrast, we use a pre-trained optical flow network that finds matched pixels independently of the performance of depth estimation itself and provides a more reliable and stable loss function to optimize. Secondly, leveraging optical flow enables us to define strict binary masks (described in Section 2) to exclude incorrectly warped pixels and only retain valuable data for supervising the training.

However, the drawback is that optical flow networks are designed to work with RGB images, while we provide them with ambient images that include only luminance information. To be more specific, we had to convert the ambient images to RGB format before feeding them to LiteFlowNet [6]. This conversion would degrade optical flow prediction performance, but it does not degrade the performance of our depth estimation models proportionately. The reason is that we have made our models robust to the performance of optical flow thanks to our multiple masking criteria (Section 2). We only retain correctly warped pixels, so a lower optical flow performance results in fewer pixels for supervision, not more faulty pixels. An ablation study on the effect of the optical flow accuracy on our models' performances is provided in Section 5.5.

4. Implementation Details

Synthetic Datasets: We use the renderer tool provided by [12] to generate the synthetic datasets. Following their approach, we populate the scene with a subset of chair meshes from the ShapeNet Core dataset [3] randomly scaled and rotated, placed at a distance between 0.5–3 m from the camera. A randomly slanted background plane at a distance between 2–5 m is also employed in the scene. Like [12], the camera is randomly translated within $20 \times 20 \times 20$ cm for capturing frames of each video sequence. However, we use our own camera's parameters to capture data from the scene. The baseline between our camera and projector is 2.46 cm, and the image resolution is 512×432 pixels. As we already mentioned in the main article, three different dot patterns are exploited in our experiments resulting in three synthetic datasets.

Real Dataset: We use an Artec Eva 3D scanner to scan 4 different 3D scenes containing various objects. The scanner has a few limitations, such as having limited depth range of 0.3–1.2 m and capturing only 100 depth frames per scan. For this reason, we take multiple partial scans of each scene and register them together using the point-to-plane variant of the ICP algorithm [13], as mentioned in the main article. Once we obtain a 3D point cloud that covers the scene in a suitable way, the Ball-Pivoting algorithm [1] is applied to derive a 3D mesh of the scene. This mesh is used as ground truth. Subsequently, we scan each scene with our own structured-light sensor as to capture 148 pairs of dot images and ambient images. The sensor we use is equipped with a programmable switch, enabling the projector to be on and off, so it can capture dot images and ambient images alternately at the rate of 15 fps each. Given the capturing rate, each pair of dot image and ambient image

Model	λ_1	λ_2	λ_3
DIS-SF	0.4	0.0	0.2
DIS-FTSF	0.4	0.1	0.2
DIS-MF	0.8	0.0	0.2

Table 1. The values of the coefficients used in our models' aggregate loss functions in Equation (6). For ablation study of searching for these hyperparameters, refer to Section 5.2

captures the same scene approximately. We make use of these frames as 37 video sequences containing 4 frames each. As a result, 148 video sequences are obtained in total from all 4 scenes. Applying a block-matching technique on the dot images, we extract 148 depth images for each scene. The same ICP algorithm [13] is then used to align the depth images acquired using the sensor and the ground truth 3D mesh in order to obtain transformation matrices among the frames. We made the real dataset, along with the instructions to replicate the synthetic datasets, publicly available.

Loss Functions: Our proposed models' loss functions are explained in detail in the main paper. Here we clarify the coefficients used in the aggregate loss function. We introduced the aggregate loss function in the paper as:

$$\mathcal{L} = \frac{1}{N}\sum_{i\in\Gamma}(\mathcal{L}_{ph}^{i} + \lambda_1\mathcal{L}_{s}^{i} + \lambda_2\mathcal{L}_{pgt}^{i}) + \frac{1}{N(N-1)}\sum_{i,j\in\Gamma}\lambda_3\mathcal{L}_{mv}^{ij} \tag{6}$$

where Γ denotes the image samples representing the same scene, $\mathcal{L}_{ph}$ is the photometric loss, $\mathcal{L}_{s}$ is the smoothness loss, $\mathcal{L}_{mv}$ is the multi-view loss, and $\mathcal{L}_{pgt}$ is the pseudo-ground truth loss. The coefficients in Equation (6) take different values for each model. These values are presented in Table 1. As is shown in the table, the coefficient of the smoothness loss λ_1 is set differently in our DIS-SF and DIS-MF models. Since our binary masks are less strict in DIS-SF, it allows more pixels in flat regions or backgrounds into $\mathcal{L}_{mv}$. This promotes smoothness of depth maps, and accordingly, λ_1 is set to a smaller value in DIS-SF. In Section 5.2, we provide detailed ablation study of searching for these hyperparameters.

Training Details: We have trained our models with a single NVIDIA Tesla V100 GPU. We exploit Adam [9] optimizer with a learning rate of 10^{-4} for training. We set the batch size to 8 for training the DIS-SF and DIS-FTSF models and set it to 4 for the DIS-MF model due to GPU memory limitation.

Prior to training, we pre-save the outputs of LiteFlowNet [6] and provide them as the optical flow predictions to our model. Since there are 4 frames in each video sequence of our datasets, we need to pre-save 12 optical flow maps for each sequence. The reason is that we need both forward and backward optical flow data for each unique pair of frames in a video sequence.

We also pre-save the results of our DIS-SF model before training our DIS-MF model, and similarly, we pre-save the outputs of DIS-MF prior to training the DIS-FTSF model. These pre-saving steps make data preparation more convenient and result in an efficient training process. The time of training of our models, as well as the CTD [12] model, on synthetic datasets and fine-tuning on the real dataset is presented in Table 2.

5. Ablation Study

This section provides ablation studies regarding our introduced loss functions, setting hyperparameters, and design choices of our DIS-MF network architecture. All the experiments are evaluated on the synthetic dataset with the projection dot pattern from our industry partner.

Dataset	CTD[12]	DIS-SF	DIS-FTSF	DIS-MF
Synthetic	40	36	18	85
Real	7	6	3	12

Table 2. Approximate time (hours) of training/fine-tuning of the models on synthetic and real datasets with a single NVIDIA Tesla V100 GPU. Note that these reported times are measured assuming all required data are available. Therefore, the dependency of our models on each other's outputs should also be considered. For example, for obtaining a trained DIS-FTSF model on a synthetic dataset from scratch, we train the DIS-SF model for 36 hours, the DIS-MF model for 85 hours, and lastly, the DIS-FTSF model for 18 hours approximately.

Loss	$o(0.5)$	$o(1)$	$o(2)$	$o(5)$
Supervised by SGM Outputs	21.1	15.9	12.0	7.91
CTD Loss Functions	3.38	1.71	0.85	0.28
CTD Loss Functions + $\mathcal{L}_{pgt}$	2.62	1.25	0.58	0.14
$\mathcal{L}_{ph}$	40.5	22.1	9.36	1.15
$\mathcal{L}_{ph} + \mathcal{L}_{s}$	3.52	1.38	0.67	0.23
$\mathcal{L}_{ph} + \mathcal{L}_{mv}$	3.22	1.69	0.86	0.24
$\mathcal{L}_{ph} + \mathcal{L}_{s} + \mathcal{L}_{mv}$	2.31	1.24	0.62	0.19
$\mathcal{L}_{ph} + \mathcal{L}_{s} + \mathcal{L}_{mv} + \mathcal{L}_{pgt}$	**1.96**	**0.95**	**0.45**	**0.12**

Table 3. Ablation study of individual loss terms: the photometric loss $\mathcal{L}_{ph}$, the smoothness loss $\mathcal{L}_{s}$, the multi-view loss $\mathcal{L}_{mv}$, and the pseudo-ground truth loss $\mathcal{L}_{pgt}$. Since we share the same network architecture for single-frame depth estimation with CTD [12], we could fairly contrast the superiority of our loss functions with respect to CTD's. The last row, which represents our DIS-FTSF model, yields overall the best results in terms of the percentage of outliers. Also, the effectiveness and generalizability of fine-tuning with pseudo-ground truth labels $\mathcal{L}_{pgt}$ are shown when used for either fine-tuning our DIS-SF model or the CTD model.

5.1. Efficacy of the Loss Functions

Here, we attempt to distinguish the effectiveness of each of our individual loss functions. Table 3 summarizes the quantitative evaluation of our DIS-SF network when it is trained with different combinations of loss functions. Since we use exactly the same network architecture of CTD [12] in DIS-SF, it is also fair to compare the results with the case that the network is trained with loss functions employed in CTD. Furthermore, to show the generalization of the concept of fine-tuning with pseudo-labels, we fine-tuned the CTD model with our $\mathcal{L}_{pgt}$ as an additional loss function and observed improvement in the outputs as it is indicated in Table 3. Lastly, we also examined the performance if we naively supervised the network with valid outputs resulting from the SGM algorithm [4].

Figure 1 also depicts some examples of our DIS-SF network outputs when it is trained with different combinations of loss functions. The samples demonstrate how our loss functions complement each other and form a robust depth estimation model altogether.

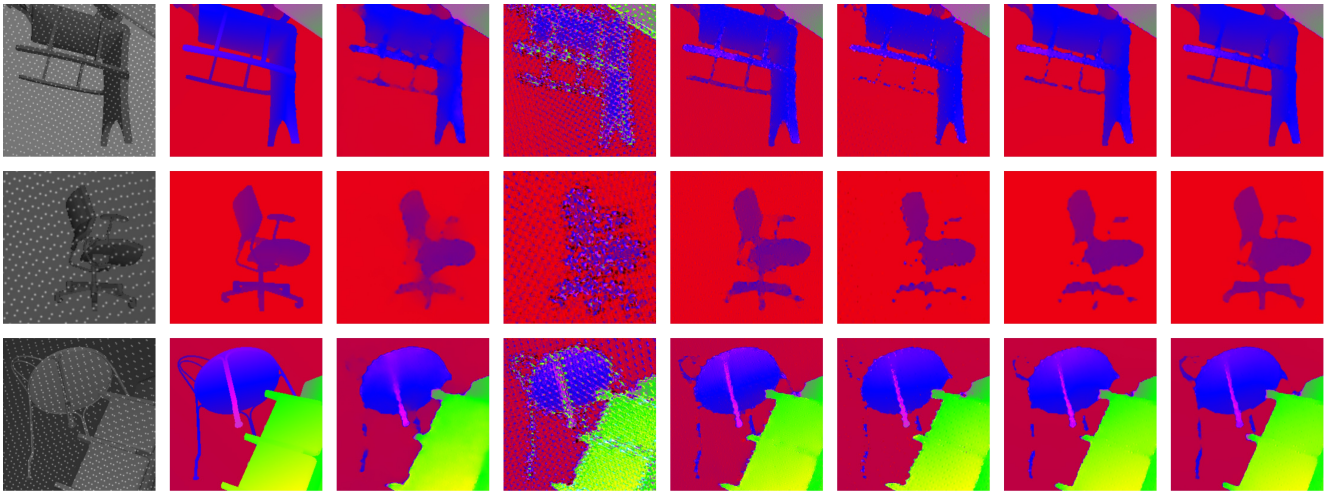


Figure 1. Qualitative ablation study of our individual loss functions. For each experiment, our DIS-SF network is trained with a different combination of our loss functions. (a) Input dot image with projected pattern. (b) Ground truth disparity map. (c) Trained with CTD [12] loss functions. (d) Trained with $\mathcal{L}_{ph}$. (e) Trained with $\mathcal{L}_{ph} + \mathcal{L}_{s}$. (f) Trained with $\mathcal{L}_{ph} + \mathcal{L}_{mv}$. (g) Trained with $\mathcal{L}_{ph} + \mathcal{L}_{s} + \mathcal{L}_{mv}$. (h) Our final single-frame model trained with $\mathcal{L}_{ph} + \mathcal{L}_{s} + \mathcal{L}_{mv} + \mathcal{L}_{pgt}$.

Model	$\lambda_1(\mathcal{L}_s)$	$\lambda_3(\mathcal{L}_{mv})$	$o(0.5)$	$o(1)$	$o(2)$	$o(5)$
DIS-SF	0.2	0.1	2.84	1.45	0.68	0.20
	0.2	0.2	2.48	1.31	0.64	0.19
	0.2	0.3	2.41	1.35	0.70	0.19
	0.2	0.4	2.35	1.26	0.66	0.19
	0.4	0.1	2.34	1.26	0.63	0.19
	0.4	0.2	**2.31**	**1.24**	**0.62**	**0.19**
	0.4	0.3	2.37	1.28	0.64	0.19
	0.4	0.4	2.49	1.29	0.67	0.20
	0.6	0.1	2.37	1.27	0.65	0.19
	0.6	0.2	2.32	1.24	0.63	0.19
	0.6	0.3	2.42	1.28	0.65	0.19
	0.6	0.4	2.44	1.31	0.68	0.19
	0.8	0.1	2.35	1.25	0.65	0.19
	0.8	0.2	2.40	1.28	0.66	0.19
	0.8	0.3	2.43	1.30	0.67	0.20
	0.8	0.4	2.50	1.38	0.69	0.20
DIS-MF	0.2	0.1	2.19	0.93	0.43	0.12
	0.2	0.2	1.98	0.93	0.44	0.12
	0.2	0.3	1.99	0.96	0.47	0.13
	0.4	0.1	1.92	0.86	0.39	0.11
	0.4	0.2	1.87	0.84	0.39	0.11
	0.4	0.3	1.81	0.88	0.42	0.11
	0.6	0.1	2.06	0.86	0.36	0.10
	0.6	0.2	1.97	0.84	0.37	0.10
	0.6	0.3	1.70	0.80	0.38	0.11
	0.8	0.1	1.96	0.82	0.35	0.10
	0.8	0.2	**1.58**	**0.71**	**0.32**	**0.10**
	0.8	0.3	1.66	0.77	0.36	0.10
	1.0	0.1	1.88	0.75	0.34	0.10
	1.0	0.2	1.72	0.75	0.34	0.10
	1.0	0.3	1.62	0.76	0.35	0.10

Table 4. Ablation study of searching for hyperparameters of our aggregate loss function. It is noticeable that even poor choices of hyperparameters for the DIS-MF model still result in better performance than all experiments with the DIS-SF model. For further details on why separate experiments were needed for the DIS-SF and DIS-MF models, refer to Section 5.2 and 4.

5.2. Searching for Hyperparameters

Table 4 represents the experiments we conducted for searching for the hyperparameters of our aggregate loss function for the DIS-SF and DIS-MF models. As we previously discussed, the hyperparameters should take different values due to the different mask criteria we defined in Section 2. In the DIS-SF model, the mask criteria are less strict, and as a result, smoothness of disparity maps is inherently promoted in the multi-view loss $\mathcal{L}_{mv}$, whereas smoothness loss $\mathcal{L}_s$ in DIS-MF is emphasized directly. The experiments in Table 4 support this hypothesis.

5.3. DIS-MF Network Architecture

The quantitative analysis of four experiments is presented in Table 5: 1. Effect of the number of fused frames. 2. Effect of the number of fusion blocks and their channel size in the overall network architecture. 3. Efficacy of various processing elements in our DIS-MF network. 4. Contribution of validation binary masks, introduced in Section 2, on our model's performance. Moreover, qualitative analysis of these four experiments are provided in Figures 2, 3, 4, and 5.

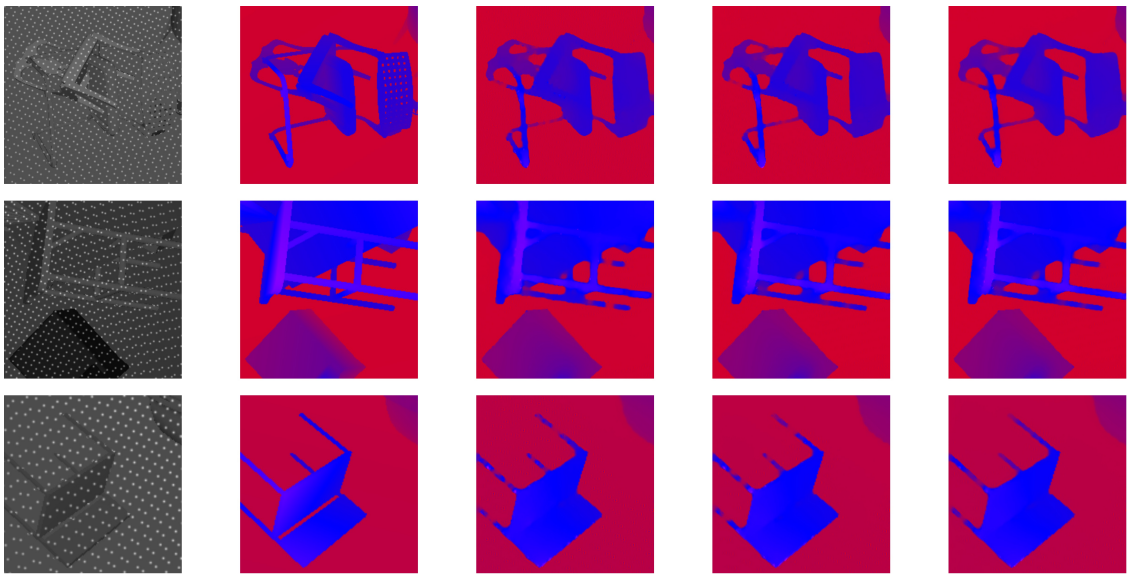
In particular, it is notable that despite the hugeness of our network and its robustness against variation of the number of fusion blocks or their channel size, according to Table 5, the performance still degrades when we remove continuous 3D convolution and perform fusion only on the 2D grid map. As also discussed in the main paper, in the context of depth estimation from conventional RGB camera, works like DeepV2D [16], DeepMVS [5], DeepSFM [17], and DPSNet [7]

N	Fusion Blocks		Network Components			Binary Masks Criteria			Metrics [%]				Examples
	Num.	Ch.	2D Fus.	3D Fus.	Ref.	M'_{FB}	M'_{RF}	M'_{VC}	$o(0.5)$	$o(1)$	$o(2)$	$o(5)$	
Experiment 1: Effect of the number of fused frames N.													
2	4	32	●	●	●	●	●	●	1.90	0.83	0.37	0.10	Fig. 2.c
3	4	32	●	●	●	●	●	●	1.73	0.75	0.34	0.10	Fig. 2.d
4	4	32	●	●	●	●	●	●	**1.58**	**0.71**	**0.32**	**0.10**	Fig. 2.e
Experiment 2: Effect of the number of fusion blocks and their channel size.													
4	2	48	●	●	●	●	●	●	1.59	0.72	0.33	0.10	Fig. 3.c
4	4	32	●	●	●	●	●	●	**1.58**	**0.71**	**0.32**	**0.10**	Fig. 3.d
4	6	24	●	●	●	●	●	●	1.62	0.72	0.33	0.10	Fig. 3.e
4	8	16	●	●	●	●	●	●	1.60	0.73	0.33	0.10	Fig. 3.f
Experiment 3: Each processing component's efficacy in the network architecture.													
4	4	32	○	●	●	●	●	●	1.58	0.72	0.33	0.10	Fig. 4.c
4	4	32	●	○	●	●	●	●	1.74	0.74	0.34	0.11	Fig. 4.d
4	4	32	●	●	○	●	●	●	1.98	1.04	0.50	0.13	Fig. 4.e
4	4	32	●	●	●	●	●	●	**1.58**	**0.71**	**0.32**	**0.10**	Fig. 4.f
Experiment 4: Impact of the validation binary masks on the performance.													
4	4	32	●	●	●	●	○	○	2.19	1.11	0.53	0.14	Fig. 5.c
4	4	32	●	●	●	●	○	●	1.63	0.71	0.33	0.10	Fig. 5.d
4	4	32	●	●	●	●	●	○	1.88	0.91	0.42	0.11	Fig. 5.e
4	4	32	●	●	●	●	●	●	**1.58**	**0.71**	**0.32**	**0.10**	Fig. 5.f

Table 5. Ablation study of our DIS-MF network architecture. This table summarizes four experiments, in each of which the effect of a subset of particular design choices is examined independently. Firstly, this study shows how the performance of DIS-MF improves with higher numbers of fused frames N. The second experiment examines how many cascaded fusion blocks and with which channel size lead to better results (The network architecture is presented in Figure 3 of the main paper). The channel size and the number of blocks are selected such that computation resources become comparable. As the results suggest, our proposed architecture is robust against variations of these parameters. The third experiment evaluates the efficacy of processing components in our network architecture. As introduced in the main paper in detail (Section 3.2 in the main paper), our fusion block contains aggregating convolution modules both on the 2D grid map and in the continuous 3D space. Here, the contribution of each part to the final performance is shown (*e.g.*, the $o(0.5)$ metric deteriorates from **1.58** % to **1.74** % when we perform aggregation only on the 2D grid map). For further details on why this comparison is important, please refer to Section 5.3. Also, the effectiveness of the refinement structure in the post-processing part of our network architecture is investigated in this experiment. Lastly, this study demonstrates how each of the binary mask criteria in Section 2 affects our DIS-MF model's performance in the fourth experiment. Qualitative analyses corresponding to each of these experiments are provided in Figures 2, 3, 4, and 5.

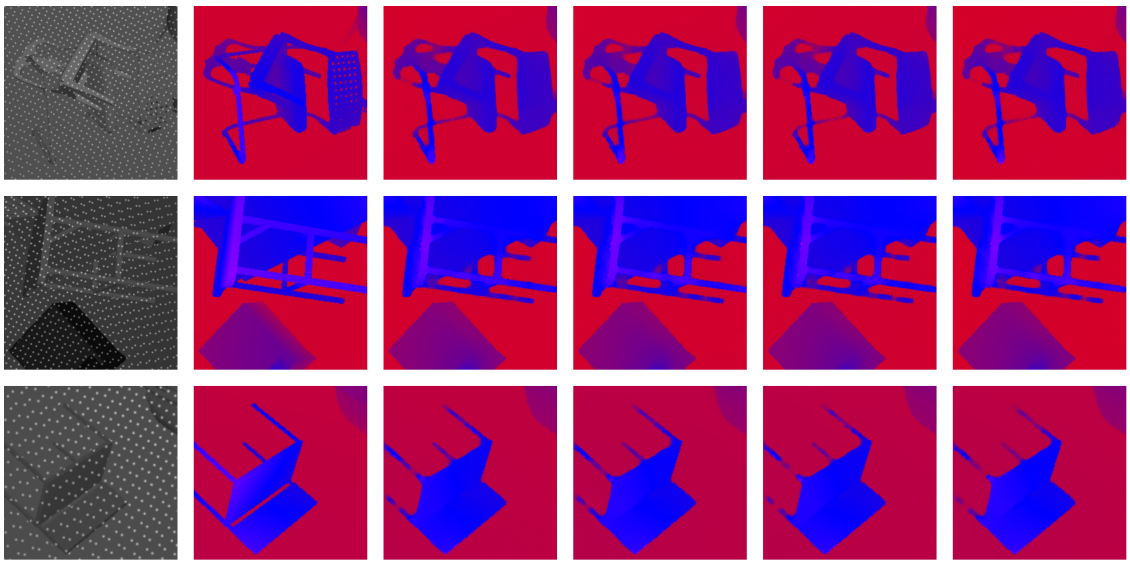
attempt to fuse information of multiple frames on the 2D grid. We cannot directly compare our work with them because they are designed to operate on RGB images. Still, we attempt to demonstrate how leveraging fusion both on the 2D grid and in the 3D space improves the performance in the structured-light setup depth estimation.

Lastly, Figure 6 exemplifies the behavior of each binary mask criterion in Section 2 and shows how these masks exclude low confidence pixels and prevent them from supervising the training.



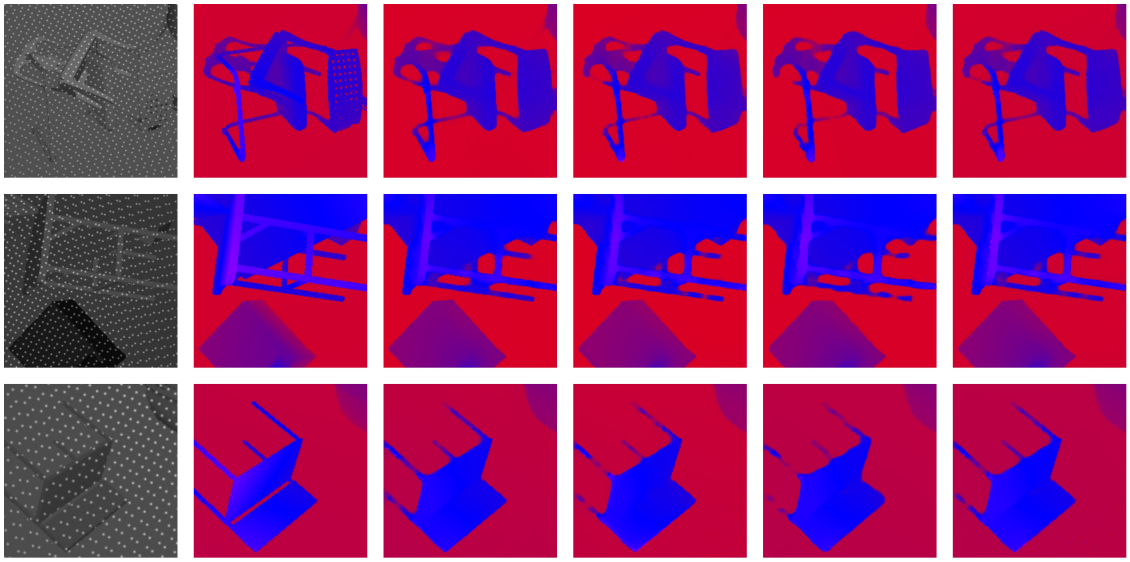
(a) Dot Image (b) GT (c) $N = 2$ (d) $N = 3$ (e) $N = 4$

Figure 2. Qualitative analysis of Experiment 1 in Table 5, where we examine the effect of the number of fused frame N on DIS-MF performance. (a) Input dot image. (b) Ground truth disparity map. (c) DIS-MF's output with $N = 2$. (d) DIS-MF's output with $N = 3$. (e) DIS-MF's output with $N = 4$. As expected according to Table 5, DIS-MF performs better at completing disparity maps as the number of fused frames increases.



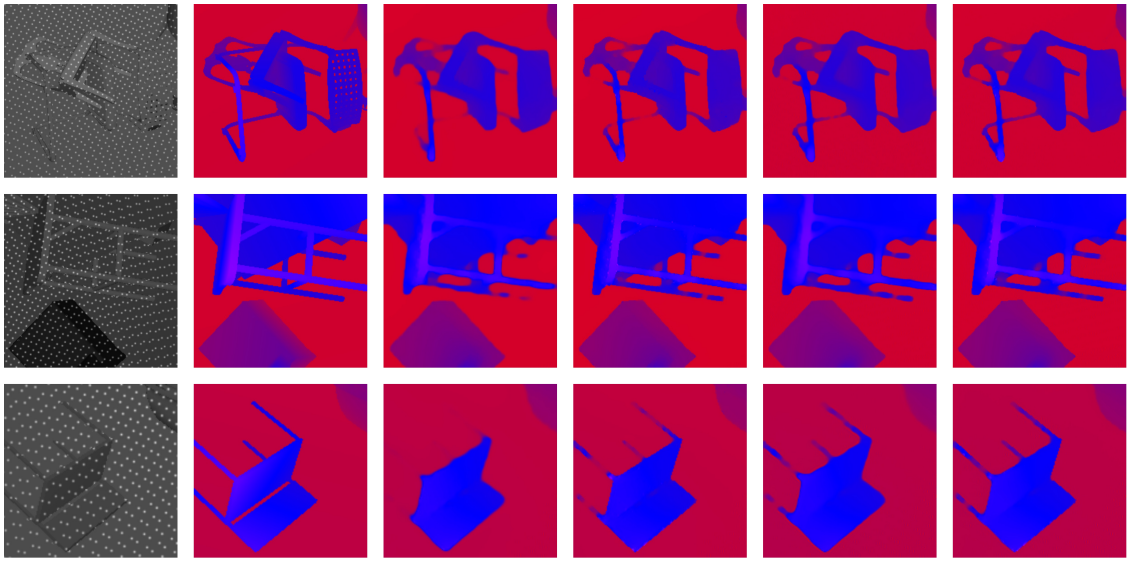
(a) (b) (c) (d) (e) (f)

Figure 3. Qualitative analysis of Experiment 2 in Table 5, where we examine the effect of the number of fusion blocks and their channel size on DIS-MF performance (The architecture is presented in Figure 3 in the main paper). (a) Input dot image. (b) Ground truth disparity map. (c) DIS-MF's output with 2 cascaded fusion blocks, each of which with 48 channels. (d) DIS-MF's output with 4 cascaded fusion blocks, each of which with 32 channels. (e) DIS-MF's output with 6 cascaded fusion blocks, each of which with 24 channels. (f) DIS-MF's output with 8 cascaded fusion blocks, each of which with 16 channels. As expected according to Table 5, DIS-MF performs robustly with different choices of parameters, and having 4 fusion blocks with 32 channels produces higher quality disparity maps.



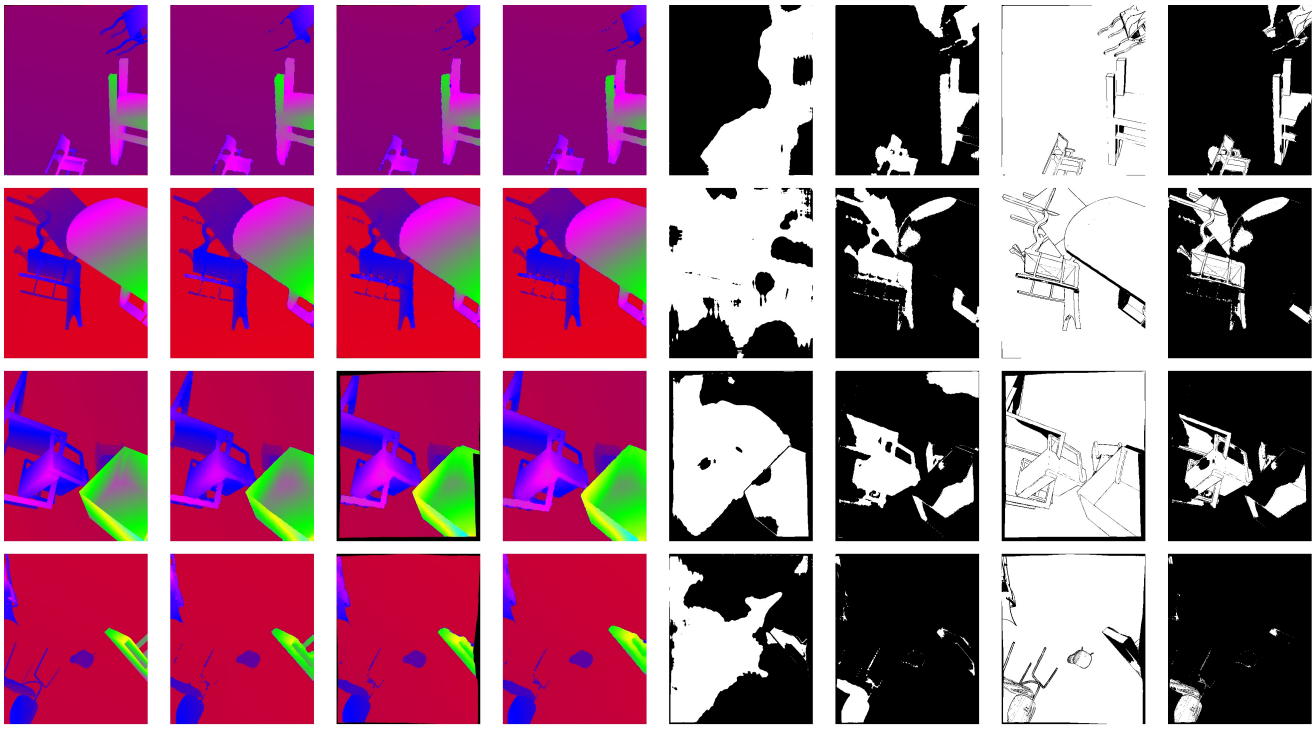
(a) (b) (c) (d) (e) (f)

Figure 4. Qualitative analysis of Experiment 3 in Table 5, where we examine the efficacy of each processing component on DIS-MF performance (Components are introduced in Section 3.2 in the main paper). (a) Input dot image. (b) Ground truth disparity map. (c) DIS-MF's output when fusion is done only in the continuous 3D space. (d) DIS-MF's output when fusion is done only on the 2D grid map. (e) DIS-MF's output when the refinement structure (post-processing part in Figure 3 in the main paper) is removed from the network architecture. (f) DIS-MF's output with all processing components. This figure, in particular, contrasts the capability of DIS-MF in completing disparity maps when fusion is performed in the continuous 3D space versus when it is done on the 2D grid map. Additionally, by comparing columns (c) and (f), one observes that fusing only in the 3D space results in fewer missing points compared to having both 2D and 3D fusion in the architecture. However, the edge fattening artifact, which is inherent to continuous 3D convolution, is more pronounced in the disparity maps generated by aggregating data only in the 3D space. Overall, employing both 2D and 3D fusion components produces balanced disparity maps concerning completing missing points and preserving edges and discontinuities in the disparity maps.



(a) (b) (c) (d) (e) (f)

Figure 5. Qualitative analysis of Experiment 4 in Table 5, where we examine the impact of validation binary masks, introduced in Section 2, on DIS-MF performance (a) Input dot image. (b) Ground truth disparity map. (c) DIS-MF's output when only $\boldsymbol{M'_{FB}}$ is used to select pixels for $\boldsymbol{\mathcal{L}_{mv}}$. (d) DIS-MF's output when $\boldsymbol{M'_{FB}}$ and $\boldsymbol{M'_{VC}}$ are used to select pixels for $\boldsymbol{\mathcal{L}_{mv}}$. (e) DIS-MF's output when $\boldsymbol{M'_{FB}}$ and $\boldsymbol{M'_{RF}}$ are used to select pixels for $\boldsymbol{\mathcal{L}_{mv}}$. (f) DIS-MF's output when all $\boldsymbol{M'_{FB}}$, $\boldsymbol{M'_{RF}}$, and $\boldsymbol{M'_{VC}}$ are used to select pixels for $\boldsymbol{\mathcal{L}_{mv}}$. It is noticeable how $\boldsymbol{M'_{RF}}$ contributes to completing the disparity map while $\boldsymbol{M'_{VC}}$ properly preserves the edges and discontinuities.



(a) GT (b) $\boldsymbol{D_i}$ (c) $\boldsymbol{D_{j\to i}}$ (d) $\boldsymbol{D_j}$ (e) $\boldsymbol{M'_{FB}}$ (f) $\boldsymbol{M'_{RF}}$ (g) $\boldsymbol{M'_{VC}}$ (h) $\boldsymbol{M'_{j\to i}}$

Figure 6. Visualizing the behavior of binary masks criteria in Section 2 when Frame j is being used for supervision and is warped on the Frame i's grid map. This figure demonstrates how each criterion selects high confidence pixels for training and prevents low confidence pixels from participating in the supervision and destabilizing training. (a) Frame i's ground truth disparity map. (b) Frame i's imperfect disparity map ($\boldsymbol{D_i}$). (c) Frame j's imperfect disparity map warped on the Frame i's grid map ($\boldsymbol{D_{j\to i}}$). (d) Frame j's imperfect disparity map ($\boldsymbol{D_j}$). (e) Optical flow forward-backward consistency mask $\boldsymbol{M'_{FB}}$. (f) Optical flow and rigid flow consistency mask $\boldsymbol{M'_{RF}}$. (g) Visual consistency of ambient images mask $\boldsymbol{M'_{VC}}$. (h) The intersection of all binary masks $\boldsymbol{M'_{j\to i}} = \boldsymbol{M'_{FB}} \odot \boldsymbol{M'_{RF}} \odot \boldsymbol{M'_{VC}}$.

Model	$o(0.5)$	$o(1)$	$o(2)$	$o(5)$
CTD	3.38	1.71	0.85	0.28
DIS-SF	2.31	1.24	0.62	0.19
CTD Out. + DIS-MF Net. + CTD Loss	2.33	1.14	0.60	0.20
CTD Out. + DIS-MF Net. + DIS Loss	1.97	0.83	0.37	0.12
DIS-MF	**1.58**	**0.71**	**0.32**	**0.10**

Table 6. Analysis of our multi-frame fusion efficacy when it takes CTD outputs as imperfect input disparities.

Model	$o(0.5)$	$o(1)$	$o(2)$	$o(5)$
DIS-SF + LiteFlowNet	2.31	1.24	0.62	0.19
DIS-SF + GMA	**2.14**	**1.12**	**0.56**	**0.14**
DIS-MF + LiteFlowNet	**1.58**	**0.71**	0.32	0.10
DIS-MF + GMA	1.62	**0.71**	**0.31**	**0.09**

Table 7. Analysis of the effect of using different optical flow networks (LiteFlowNet [6] and GMA [8]) on DIS models' performances.

5.4. Utilizing the Fusion Architecture in Different Setups

This section analyzes the effect of three factors in our DIS-MF model's performance: the imperfect input disparities, the fusion architecture, and our proposed multi-view-based loss function. In this regard, we applied our DIS-MF network to CTD outputs and trained the fusion model once with CTD loss function and once with ours. Comparison of the results with the original CTD and DIS models in Table 6 discriminates the individual efficacy of our proposed loss and fusion architecture and shows these two contributions are complementary to each other. Also, the table suggests that these two are not necessarily required to be applied to DIS-SF imperfect disparities, as they can significantly improve the quality of CTD outputs.

5.5. Robustness to Optical Flow Predictions

As discussed in Section 3, our training process is robust to optical flow prediction errors thanks to our validation masks introduced in Section 2. This robustness allows us to utilize the lightweight optical flow network LiteFlowNet [6] in our models. To evaluate this robustness, we also trained our networks on the synthetic data with the state-of-the-art optical flow model on MPI Sintel [2], GMA [8], which has **69.4**% higher accuracy but is slower than LiteFlowNet. The results in Table 7 show that despite the huge gap between the GMA and LiteFlowNet accuracies, the gain that GMA brings to our models is marginal. It is also notable that since our validation masks in DIS-MF are stricter than in DIS-SF on account of having access to imperfect disparities (see Section 2), DIS-MF is more robust to the optical flow performance.

6. Additional Qualitative Results

Here we first present an extended qualitative analysis of existing models along with our proposed models in Figures 7, 8, 9, and 10. Each figure presents sample images from one of the datasets introduced in the main paper. The color bars provided for disparity error maps (in terms of pixels) in the figures are also valid for the images in the main article.

Furthermore, we adopt a different approach to visualize depth maps of objects from the synthetic dataset in Figure 11 and from the real dataset in Figure 12. To visualize the depth maps, we rely on color-coded depth values and 3D rendering of the scene. The depths are rendered in OpenGL [15], where every point of the mesh has a color that corresponds to its depth value in the Turbo colormap [11]. In addition, we place a spotlight at the same position as the camera to light the scene. This allows us to better appreciate the quality of the computed depths as small changes of the values of the normals in the mesh impact the lighting. Finally, we consider that edges over 10 cm in the mesh are invalid and exclude them from the rendered images.

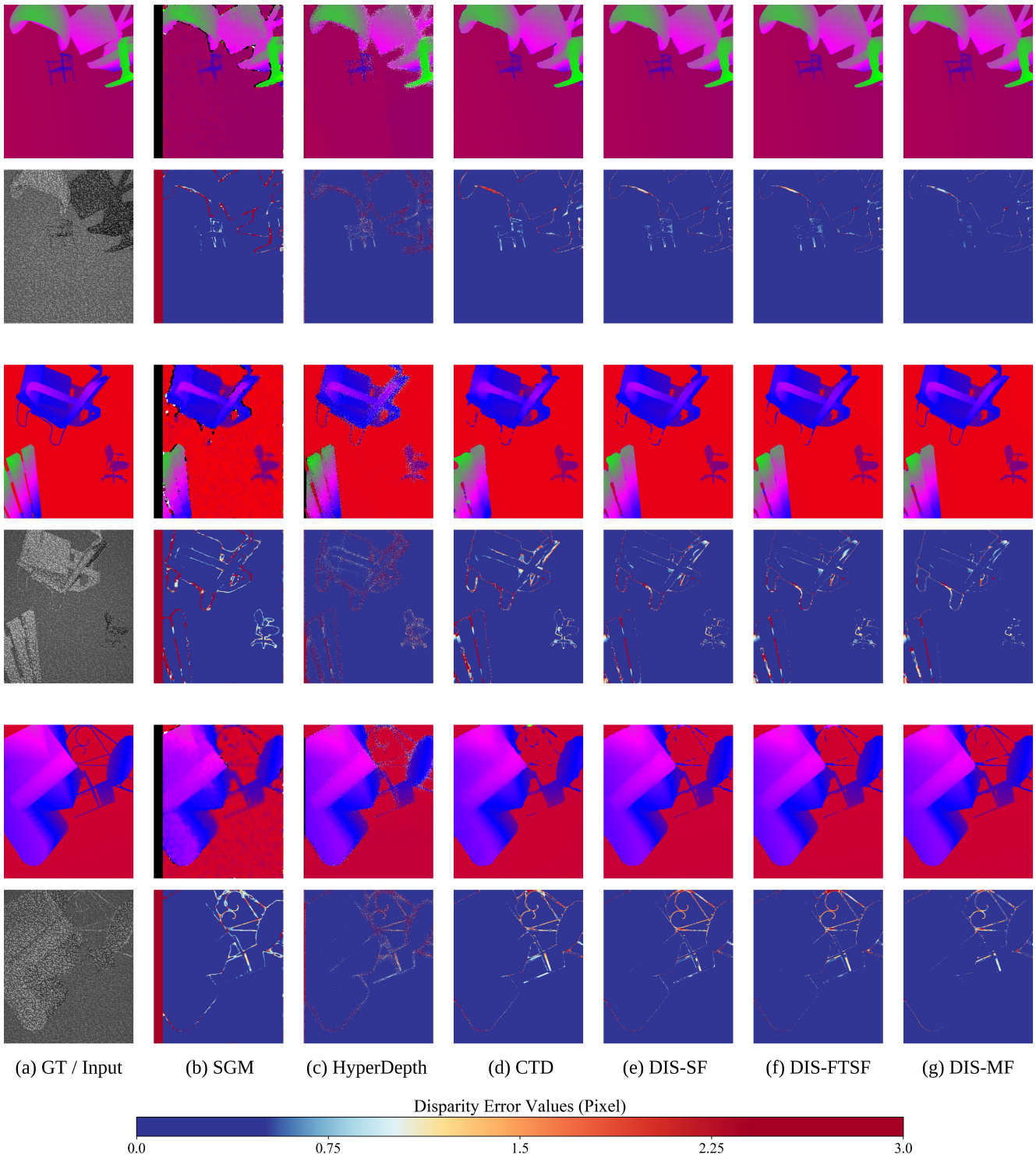


Figure 7. Additional full-size qualitative results of the implemented methods and their corresponding error maps. All samples are taken from the synthetic dataset rendered with Kinect dot pattern. (a) Ground truth disparity map and input dot image. (b) The SGM algorithm [4]. (c) HyperDepth [14]. (d) CTD [12]. (e) Our DepthInSpace Single-Frame (DIS-SF) model. (f) Our DepthInSpace Fine-Tuned Single-Frame (DIS-FTSF) model. (g) Our DepthInSpace Multi-Frame (DIS-MF) Model.

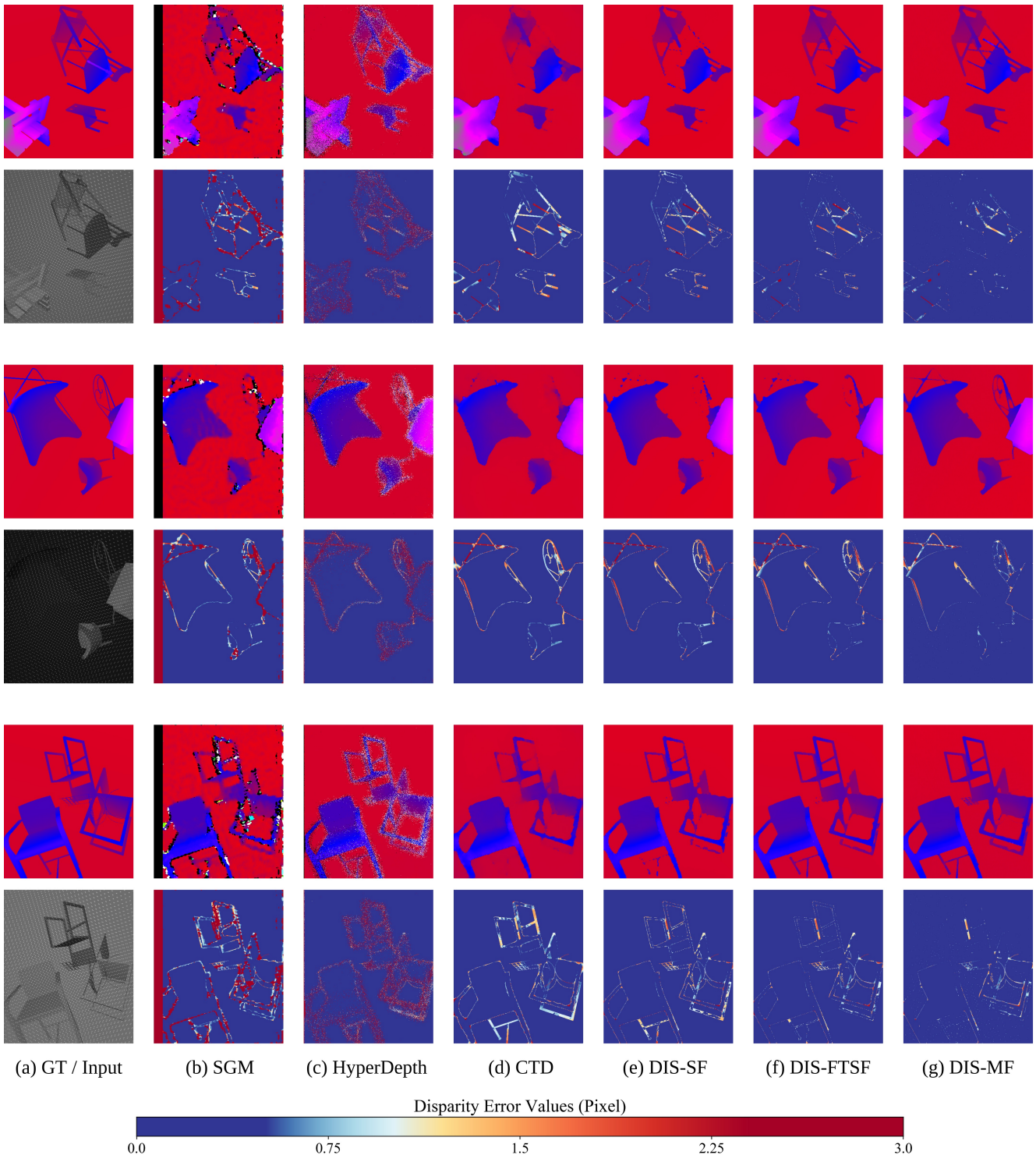


Figure 8. Additional full-size qualitative results of the implemented methods and their corresponding error maps. All samples are taken from the synthetic dataset rendered with our own theoretical dot pattern. (a) Ground truth disparity map and input dot image with projected pattern. (b) The SGM algorithm [4]. (c) HyperDepth [14]. (d) CTD [12]. (e) Our DepthInSpace Single-Frame (DIS-SF) model. (f) Our DepthInSpace Fine-Tuned Single-Frame (DIS-FTSF) model. (g) Our DepthInSpace Multi-Frame (DIS-MF) Model.

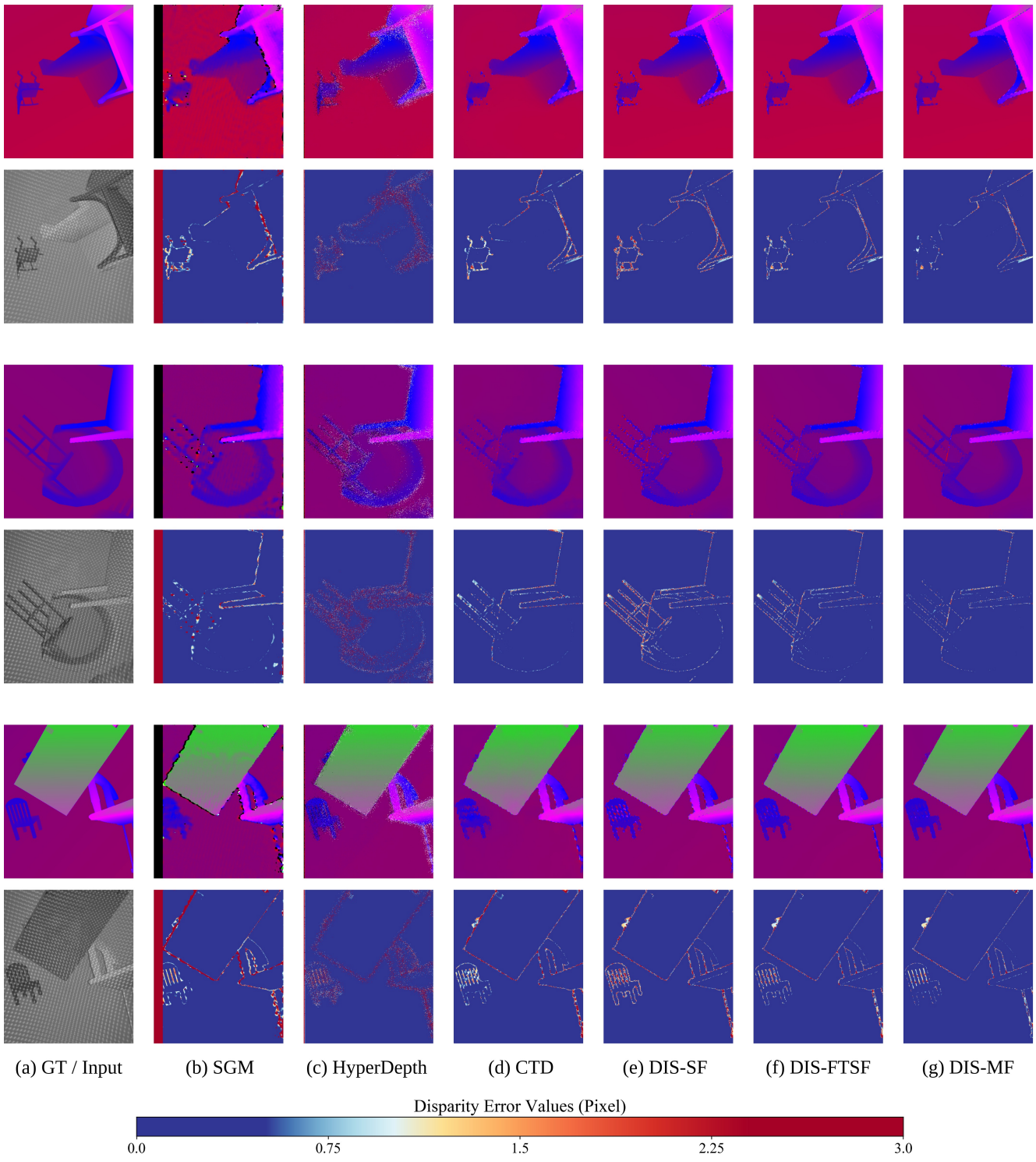


Figure 9. Additional full-size qualitative results of the implemented methods and their corresponding error maps. All samples are taken from the synthetic dataset rendered with our own dot pattern observed in an actual laboratory setup. (a) Ground truth disparity map and input dot image. (b) The SGM algorithm [4]. (c) HyperDepth [14]. (d) CTD [12]. (e) Our DepthInSpace Single-Frame (DIS-SF) model. (f) Our DepthInSpace Fine-Tuned Single-Frame (DIS-FTSF) model. (g) Our DepthInSpace Multi-Frame (DIS-MF) Model.

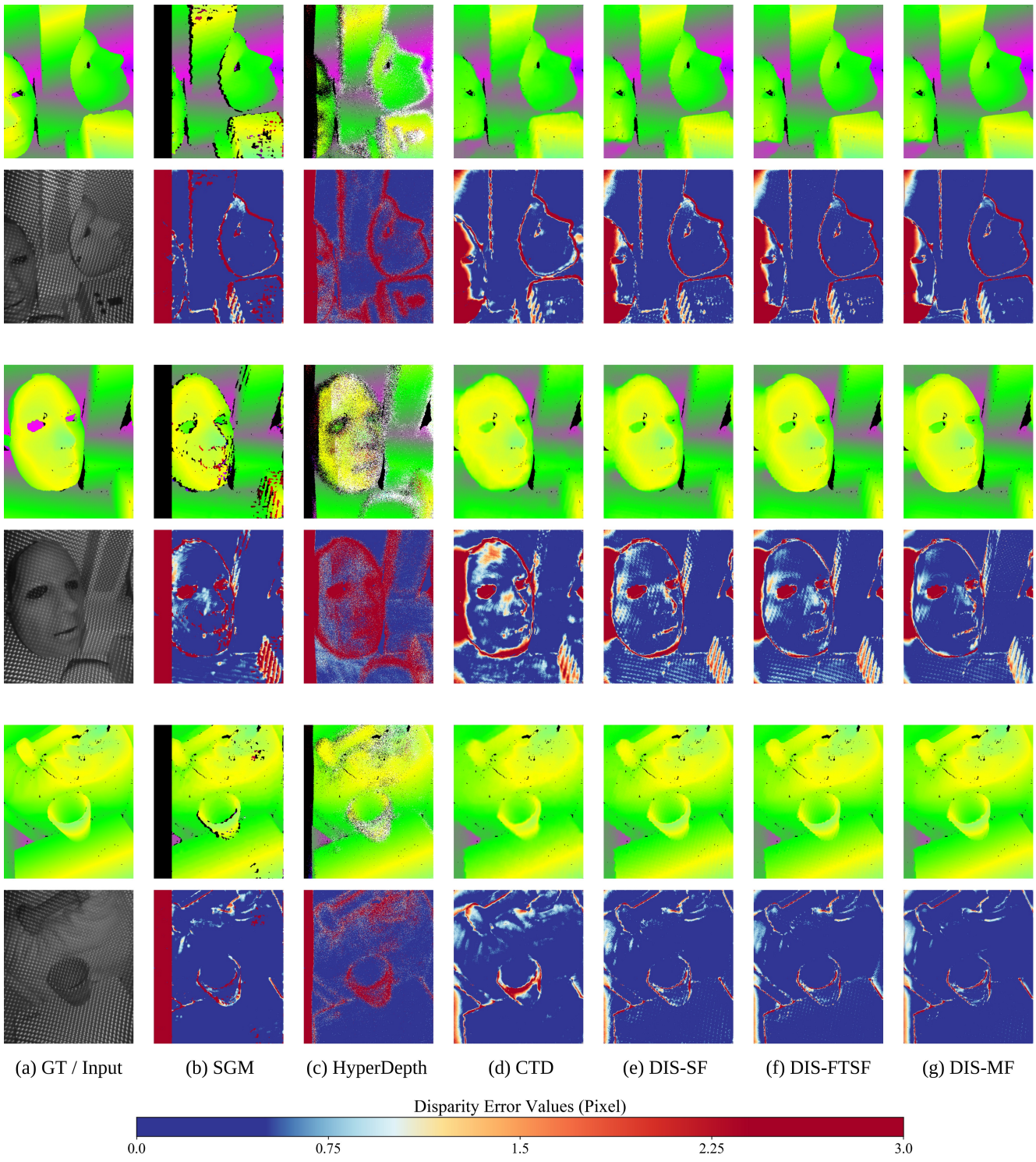
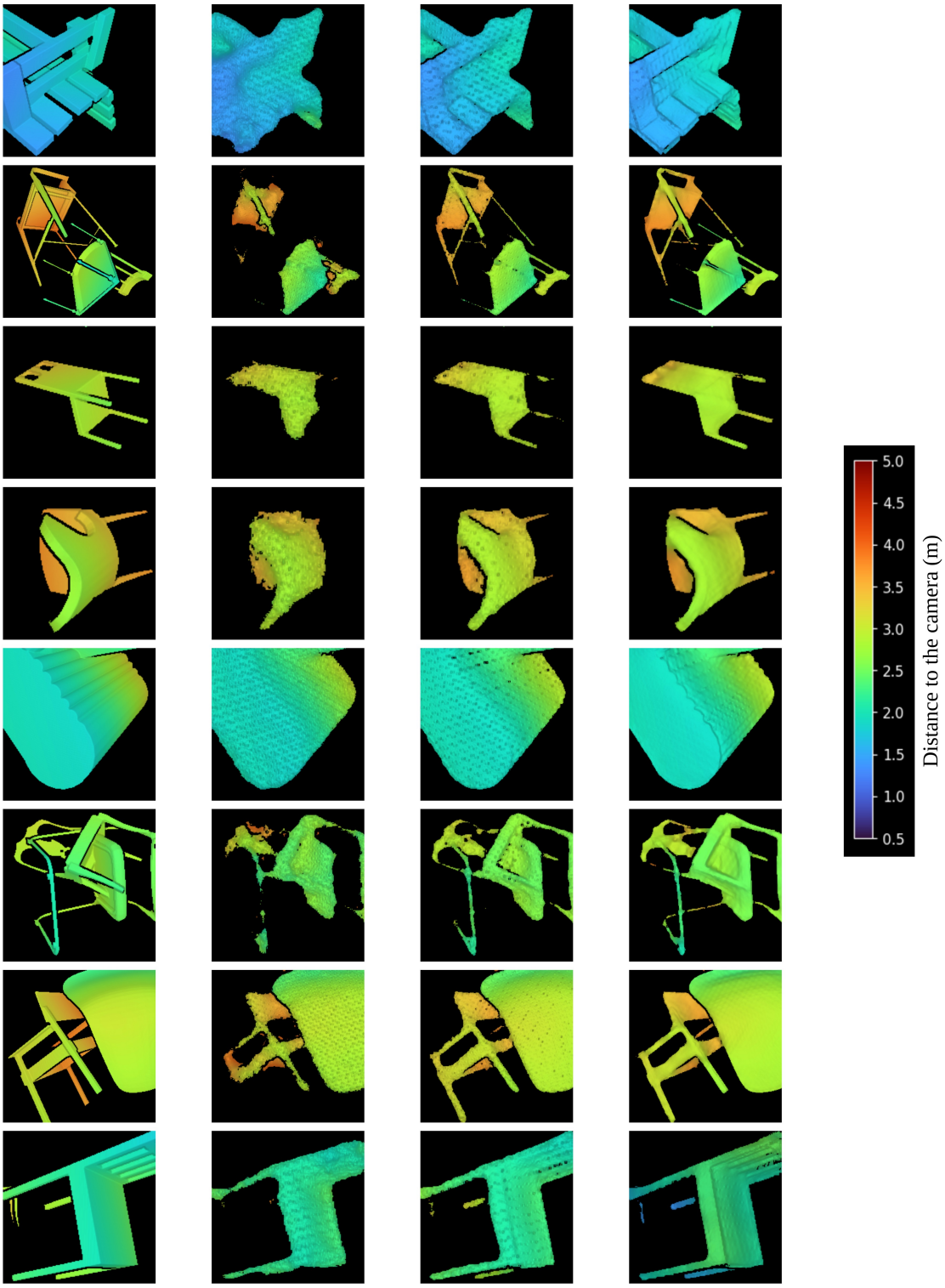


Figure 10. Additional full-size qualitative results of the implemented methods and their corresponding error maps. All samples are taken from the real dataset. (a) Ground truth disparity map and input dot image. (b) The SGM algorithm [4]. (c) HyperDepth [14]. (d) CTD [12]. (e) Our DepthInSpace Single-Frame (DIS-SF) model. (f) Our DepthInSpace Fine-Tuned Single-Frame (DIS-FTSF) model. (g) Our DepthInSpace Multi-Frame (DIS-MF) Model. Points for which the ground truth data is unavailable are excluded from evaluation.



(a) GT (b) CTD (c) DIS-FTSF (d) DIS-MF

Figure 11. Qualitative analysis of the depth maps rendered in OpenGL [15] and presented in the Turbo color map [11]. Samples are taken from the synthetic dataset. This analysis shows how our models outperform the state-of-the-art model, CTD [12], in preserving details of the 3D objects and producing sharp edges. (a) Ground truth depth map. (b) CTD [12]. (c) Our DIS-FTSF model. (d) Our DIS-MF model. For further information about 3D rendering of depth maps, refer to Section 6. The colorbar represents depth values in meters.

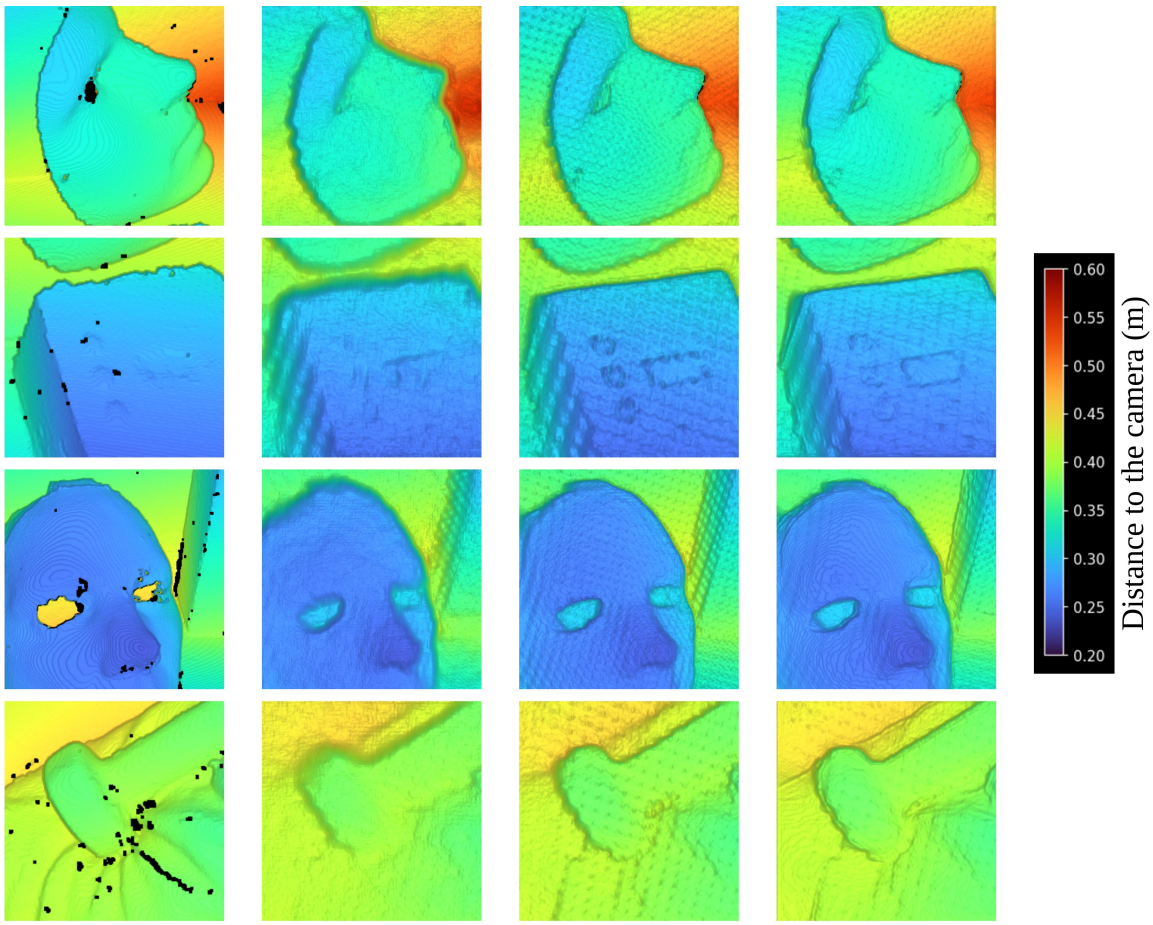


Figure 12. Qualitative analysis of the depth maps rendered in OpenGL [15] and presented in the Turbo color map [11]. Samples are taken from the real dataset. This analysis shows how our models outperform the state-of-the-art model, CTD [12], in preserving details of the 3D objects and producing sharp edges. It is noticeable that in some regions (*e.g.*, the top edge of the box in the second row in column (a)), ground truth depths are noisy, and it is due to the limitations of the 3D scanner we used to capture ground truth depths. Since all evaluated methods are self-supervised, their performances are not affected by the ground truth noise. (a) Ground truth depth map. (b) CTD [12]. (c) Our DIS-FTSF model. (d) Our DIS-MF model. For further information about 3D rendering of depth maps, refer to Section 6. The colorbar represents depth values in meters.

References

[1] F. Bernardini, J. Mittleman, H. Rushmeier, C. Silva, and G. Taubin. The ball-pivoting algorithm for surface reconstruction. *IEEE Transactions on Visualization and Computer Graphics*, 5(4):349–359, 1999. 2

[2] Daniel J Butler, Jonas Wulff, Garrett B Stanley, and Michael J Black. A naturalistic open source movie for optical flow evaluation. In *European conference on computer vision*, pages 611–625. Springer, 2012. 10

[3] Angel X. Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. ShapeNet: An Information-Rich 3D Model Repository. Technical Report arXiv:1512.03012 [cs.GR], Stanford University — Princeton University — Toyota Technological Institute at Chicago, 2015. 2

[4] Heiko Hirschmuller. Stereo processing by semiglobal matching and mutual information. *IEEE Transactions on pattern analysis and machine intelligence*, 30(2):328–341, 2007. 4, 11, 12, 13, 14

[5] Po-Han Huang, Kevin Matzen, Johannes Kopf, Narendra Ahuja, and Jia-Bin Huang. Deepmvs: Learning multi-view stereopsis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2821–2830, 2018. 5

[6] Tak-Wai Hui, Xiaoou Tang, and Chen Change Loy. Liteflownet: A lightweight convolutional neural network for optical flow estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8981–8989, 2018. 2, 3, 10

[7] Sunghoon Im, Hae-Gon Jeon, Stephen Lin, and In So Kweon. Dpsnet: End-to-end deep plane sweep stereo. In *International Conference on Learning Representations*, 2018. 5

[8] Shihao Jiang, Dylan Campbell, Yao Lu, Hongdong Li, and Richard Hartley. Learning to estimate hidden motions with global motion aggregation. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2021. 10

[9] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *International Conference on Learning Representations*, 12 2014. 3

[10] Simon Meister, Junhwa Hur, and Stefan Roth. UnFlow: Unsupervised learning of optical flow with a bidirectional census loss. In *AAAI*, New Orleans, Louisiana, Feb. 2018. 1

[11] Anton Mikhailov. Turbo, an improved rainbow colormap for visualization. *Google AI Blog*, 2019. 10, 15, 16

[12] Gernot Riegler, Yiyi Liao, Simon Donne, Vladlen Koltun, and Andreas Geiger. Connecting the dots: Learning representations for active monocular depth estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7624–7633, 2019. 2, 3, 4, 11, 12, 13, 14, 15, 16

[13] Szymon Rusinkiewicz and Marc Levoy. Efficient variants of the icp algorithm. In *Proceedings third international conference on 3-D digital imaging and modeling*, pages 145–152. IEEE, 2001. 2, 3

[14] Sean Ryan Fanello, Christoph Rhemann, Vladimir Tankovich, Adarsh Kowdle, Sergio Orts Escolano, David Kim, and Shahram Izadi. Hyperdepth: Learning depth from structured light without matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5441–5450, 2016. 11, 12, 13, 14

[15] Dave Shreiner, Graham Sellers, John Kessenich, and Bill Licea-Kane. *OpenGL programming guide: The Official guide to learning OpenGL, version 4.3*. Addison-Wesley, 2013. 10, 15, 16

[16] Zachary Teed and Jia Deng. Deepv2d: Video to depth with differentiable structure from motion. In *International Conference on Learning Representations*, 2019. 5

[17] Xingkui Wei, Yinda Zhang, Zhuwen Li, Yanwei Fu, and Xiangyang Xue. Deepsfm: Structure from motion via deep bundle adjustment. In *European conference on computer vision*, pages 230–247. Springer, 2020. 5

[18] Yuliang Zou, Zelun Luo, and Jia-Bin Huang. Df-net: Unsupervised joint learning of depth and flow using cross-task consistency. In *Proceedings of the European conference on computer vision (ECCV)*, pages 36–53, 2018. 1