

Sparse Depth Super Resolution Support Materials

Jiajun Lu

University of Illinois at Urbana Champaign

jlu23@illinois.edu

David Forsyth

University of Illinois at Urbana Champaign

daf@illinois.edu

1. Depth from No Samples

This section includes our two ways to combine with other methods. Because only results of [1] are available, we can only process the results. Since the source of [4] is available, we make more modifications. More results in Figure 1.

1.1. Ours Combined with Deep Network [1]

We first use 8 by 8 fixed sampling to subsample the depth, then use our method combining subsampled depth and image to get segmentations. For each segment, we use a weighted median filter $LS()$ to process the segments. Last, we upsample the depth using our smoothing method. The $LS()$ works like this. For each sample point, find its nearby samples, then throw away the top and bottom 20% of depth samples and calculate the mean; then use that value to replace the original value. To better demonstrate the influence of our spatial model, we also include numbers for applying $LS()$ filter on its own depth. The results are almost not influenced, see Table 1. Notice that the change in error metri is very small, but the appearance of the reconstructions has changed significantly. The spatial structure of our reconstruction is very different to that of [1]. There are now lots of sharp depth boundaries, and the depth is less heavily smoothed.

1.2. Ours Combined with Depth Transfer [4]

Our objective function includes three terms: depth term, SIFT match term and segments consistency term. The depth term forces the reconstructed depth to be similar to the smoothed weighted median range. The SIFT match term includes the SIFT matching error. The consistency term forces the adjacent segments to come from same image.

$$\begin{aligned} & \underset{\mathbf{w}}{\operatorname{argmin}} \lambda_1 \sum_s \sum_p \left(\sum_i w_{si} \mathbf{d}_{si} - f(\mathbf{d}_s) \right)^2 + \\ & \lambda_2 \sum_{s,i,p} w_{si} \mathbf{m}_{si} + \lambda_3 \sum_{s_1, s_2 \in \text{Neighbor}} \sum_{i_1 \neq i_2} A_{s_1} A_{s_2} w_{s_1 i_1} w_{s_2 i_2} \\ & \text{subject to } \sum_i w_{si} = 1 \end{aligned}$$

s is the index of the segments, i is the index of the candidate depth and p is the index of the pixels in each segment. w_{si} is the weight of the i th candidate in s th segment, \mathbf{d} is the candiate warped depth, \mathbf{m} is the SIFT error, and A_s is the size of the segment s . Function $\mathbf{p}\mathbf{d}_s = f(\mathbf{d}_s)$ is calculated as follows. Let \mathbf{d}_s be the candidate warped depth maps from [4] approach, we need the per pixel median range candidates $\mathbf{m}\mathbf{d}_s$, and the corresponding per pixel SIFT error \mathbf{se}_s . Depth prior is the mean depth of the nearest 200 images (calculated using downsampled images), and is written as $\mathbf{d}\mathbf{p}_s$. The processed depth is calculated as

$$\begin{aligned} \mathbf{r}\mathbf{d}_s &= \sum \frac{\mathbf{m}\mathbf{d}_s}{(\mathbf{se}_s + 1)^2} \\ r &= \alpha * \max(\text{std1}(\mathbf{m}\mathbf{d}_s), \text{std2}(\mathbf{r}\mathbf{d}_s)) \\ \mathbf{p}\mathbf{d}_s &= LS(\mathbf{r}\mathbf{d}_s * (1 - r) + r * \mathbf{d}\mathbf{p}_s) \end{aligned}$$

We nomalized the weights calculated by SIFT Error and the standard deviation. $\text{std1}()$ calculates the standard deviation of pixels on the same location. $\text{std2}()$ calculates the standard deviation of pixels in the near region. $LS()$ is same as above.

After solving the optimization, we have a rough depth map and we use the first two terms in the objective function to calculate a maching score. Then, find a best matching score in each sampling grid. Using this low resolution depth map and our depth super resolution method, we can recover a better depth map.

2. Extension to Optical Flow

Our method works well on data that has clear boundaries co-aligned with image boundaries, and is relatively smooth everywhere else. Optical flow satisfies these criteria and results for optical flow are included in Figure 2.

3. Effects of Different Sampling Methods

The results of Fixed Sampling and Gaussian Sampling are similar. Numerically, for images, Gaussian Sampling is generally about 2% (RMSE) better than Fixed Sampling, and for video, Gaussian Sampling is generally about 5%

Method	relative	log10	rmse	Method	relative	log10	rmse	Method	relative	log10	rmse
[4]	0.374	0.134	1.12	Ours + [4]	0.364	0.132	1.04				
[1] Coarse	0.231	n/a	0.881	Ours + [1] Coarse	0.230	n/a	0.879	$LS()$ + [1] Coarse	0.231	n/a	0.881
[1] Fine	0.218	n/a	0.914	Ours + [1] Fine	0.218	n/a	0.913	$LS()$ + [1] Fine	0.218	n/a	0.914

Table 1. Our method imposes an image derived spatial model on depth samples. We compare depth reconstructions from no data, only applying $LS()$ filter we mentioned above and subsampling the results of these methods then interpolating these samples with our approach. Only applying $LS()$ filter almost does not influence the results. Note the improvement in error, likely because our spatial model captures relations between depth and image appearance well and so suppresses a tendency to oversmooth.

(RMSE) better than Fixed Sampling. The visual difference is small and hard to tell.

4. Results on Images

This section includes more results on images of four datasets. Results on 4 images of Middlebury in Figure 3. More results on Middlebury in Figure 4. More results on NYU in Figure 5. More results on Sintel in Figure 6. More results on Gesture in Figure 7.

[RMSE], smaller is better

Method	Art			Books			Moebius		
	4	8	16	4	8	16	4	8	16
[2]	0.0200	0.0347	0.0544	0.0113	0.0147	0.0294	0.0090	0.0135	0.0266
[3]	0.0401	0.0411	0.0472	0.0233	0.0236	0.0267	0.0213	0.0215	0.0246
Ours	0.0178	0.0232	0.0291	0.0093	0.0109	0.0146	0.0072	0.0098	0.0128

Table 2. Results on Middlebury dataset, images chosen by [2].

5. Results on Videos

This section includes one frame of 12 by 12 times, 24 by 24 times, 48 by 48 times, 64 by 64 times (if any) super resolution results of six videos, in Figure 8, Figure 9, Figure 10, Figure 11, Figure 12 and Figure 13.

6. Applications

Figure 14 shows the comparison of object insertion using different depth. In the hand tracking, this tracker reports a score that evaluates how well it appears to have tracked a sequence, by scoring the best alignment between frames and the model. For typical sequences, the tracker score reported for our reconstructed depth is as good as, or better than, that reported for raw Kinect depth (Table 3).

Ratio	Kinect	Near	[2]	Mine
12×12	-0.6900	-0.7164	-0.7775	-0.6889
24×24	-0.6900	-0.7242	-0.8191	-0.6948

Table 3. Reported average tracker score (larger values indicate a better track) for sequences reconstructed from different subsampling rates, compared with raw Kinect data. Our method smoothes depths while preserving edges sufficiently well that tracks from reconstructed depths are occasionally slightly better than tracks from raw Kinect data.

References

- [1] D. Eigen, C. Puhrsch, and R. Fergus. Depth map prediction from a single image using a multi-scale deep network. *arXiv preprint arXiv:1406.2283*, 2014.
- [2] D. Ferstl, C. Reinbacher, R. Ranftl, M. Ruether, and H. Bischof. Image guided depth upsampling using anisotropic total generalized variation. In *Proceedings International Conference on Computer Vision (ICCV), IEEE*, December 2013.
- [3] K. He, J. Sun, and X. Tang. Guided image filtering. In *Proceedings of the 11th European Conference on Computer Vision: Part I, ECCV’10*, pages 1–14, Berlin, Heidelberg, 2010. Springer-Verlag.
- [4] K. Karsch, C. Liu, and S. B. Kang. Depthtransfer: Depth extraction from video using non-parametric sampling. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 2014.

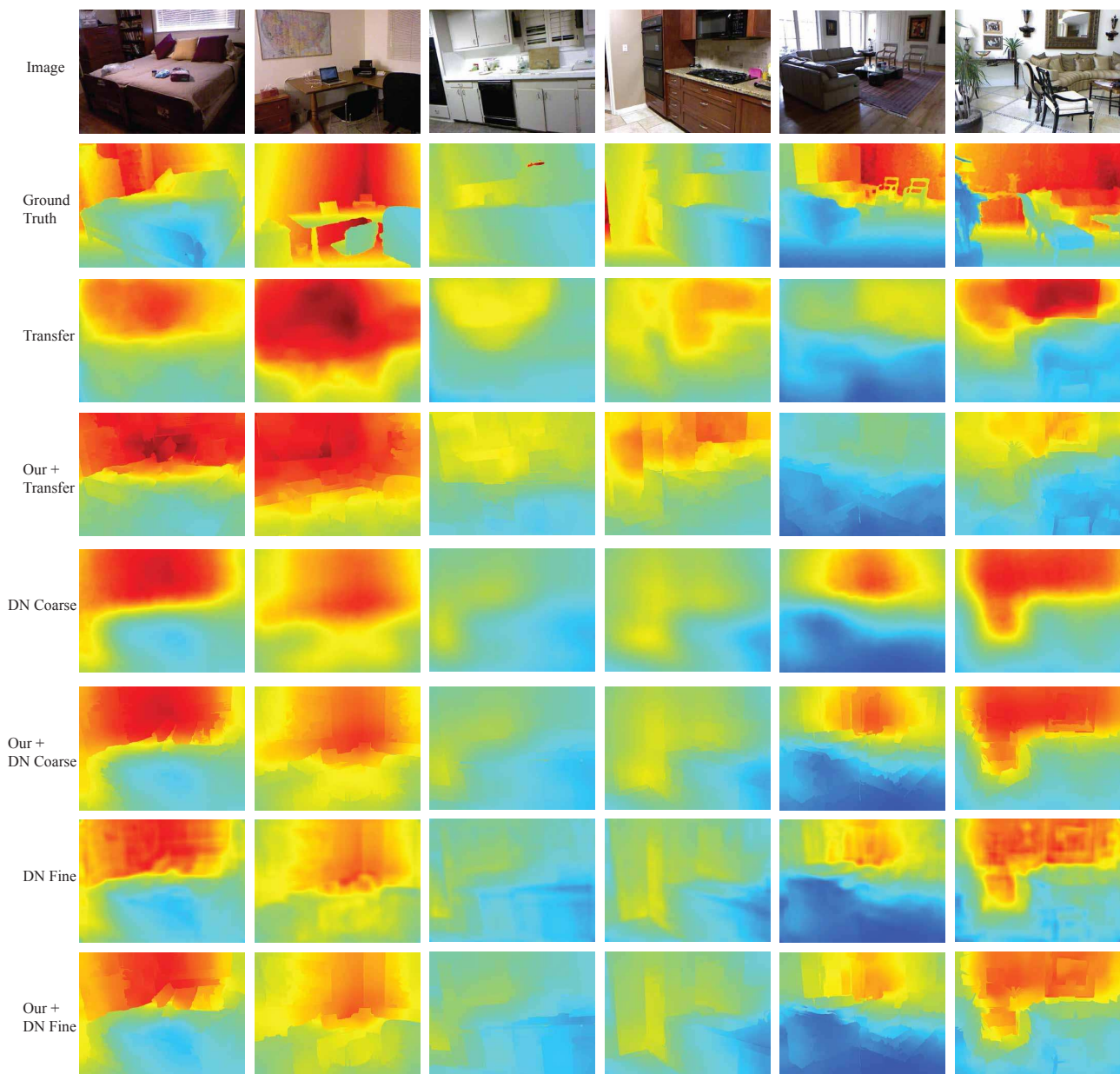


Figure 1. Results for depth from no samples. See also in movie.

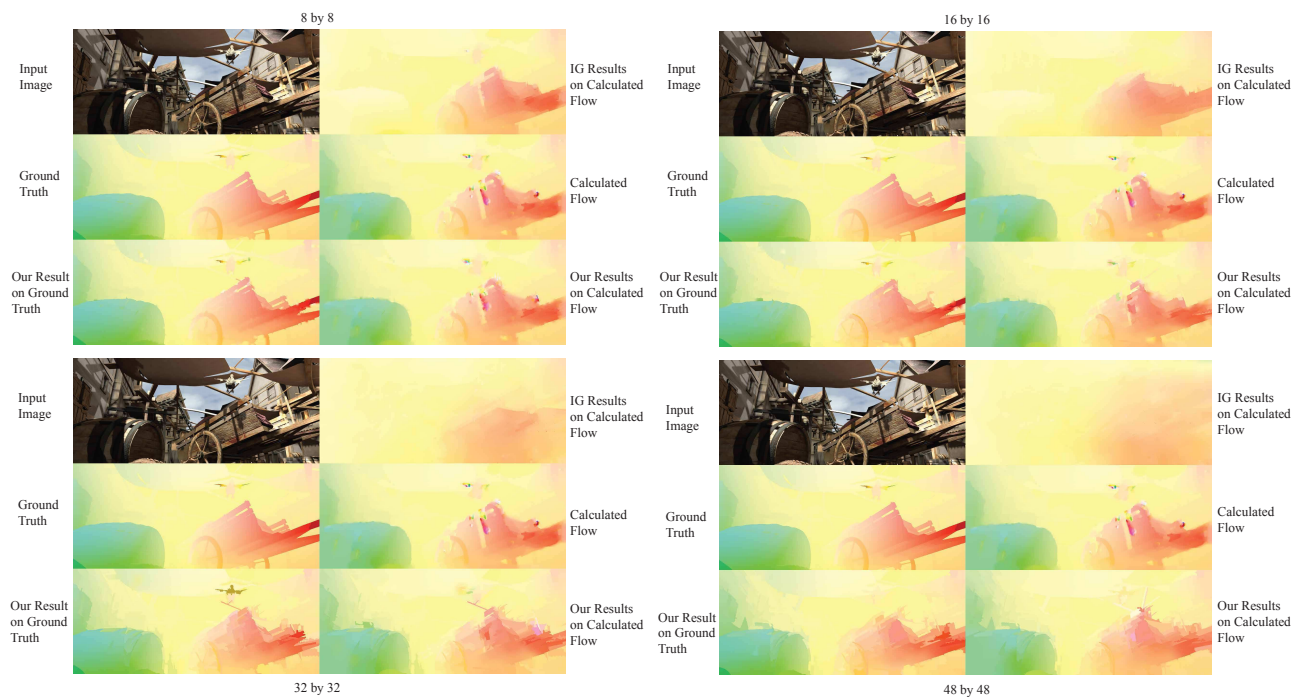


Figure 2. Extension to optical flow

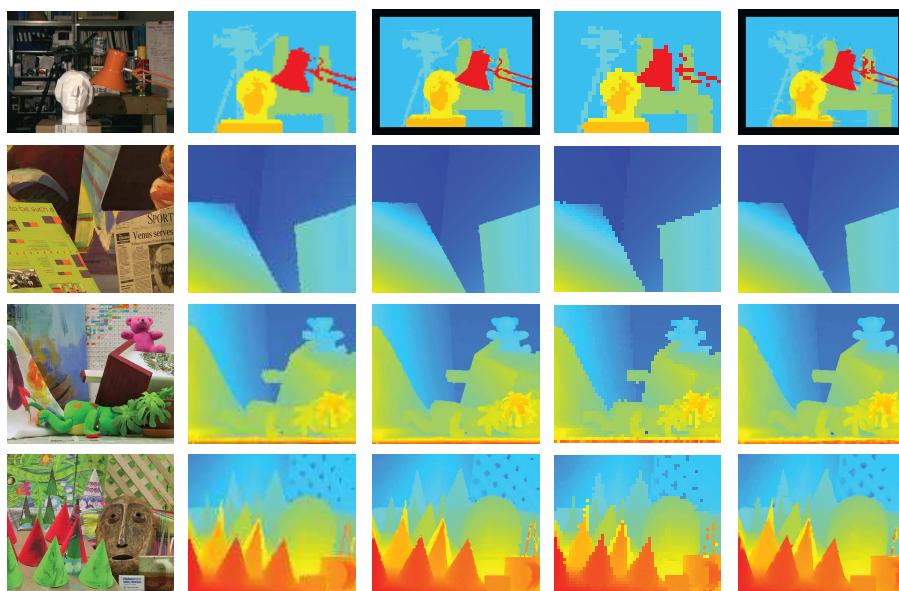


Figure 3. Results for 4 Middlebury images.

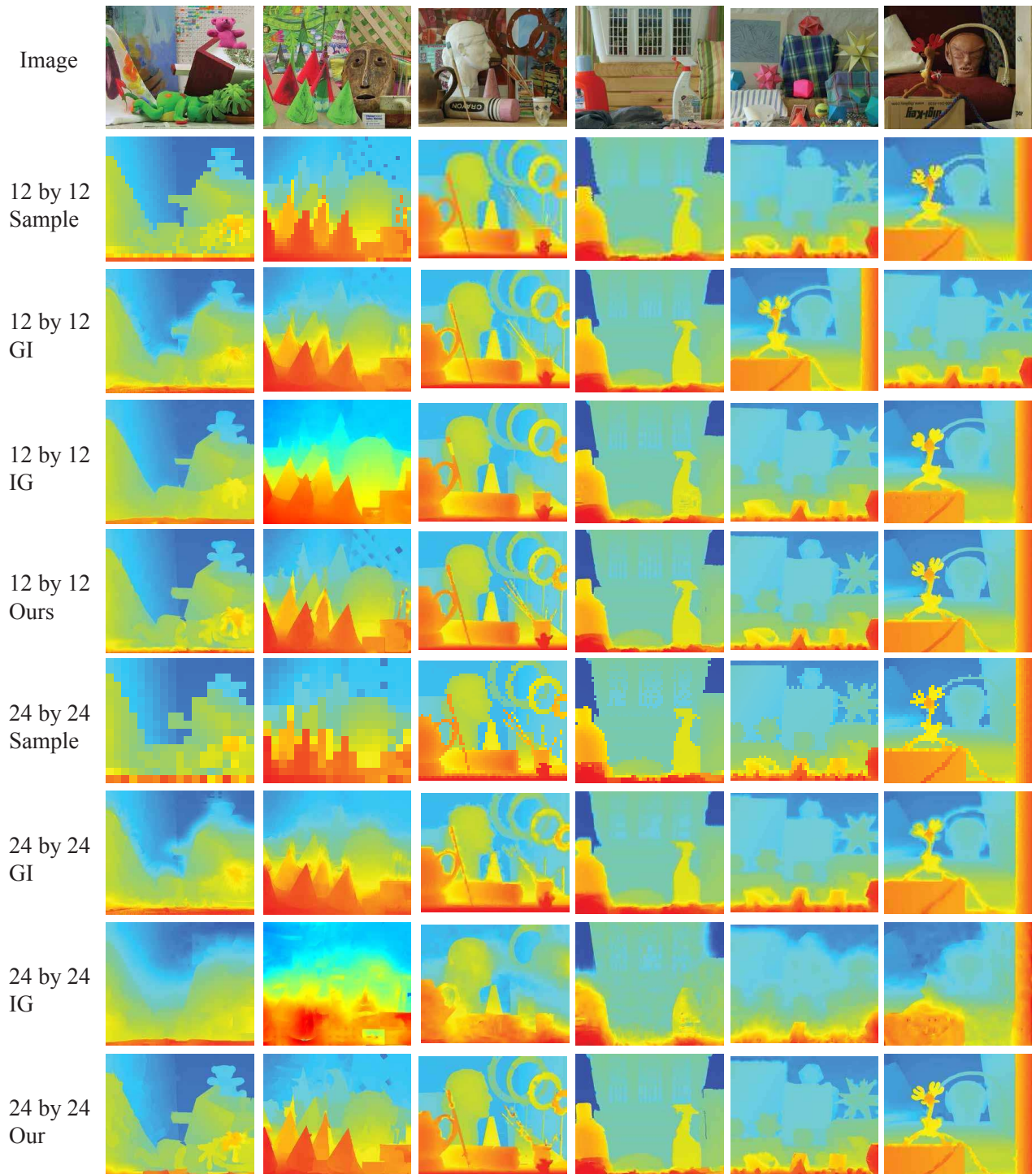


Figure 4. More results for Middlebury dataset (GI= [3]; IG= [2]).

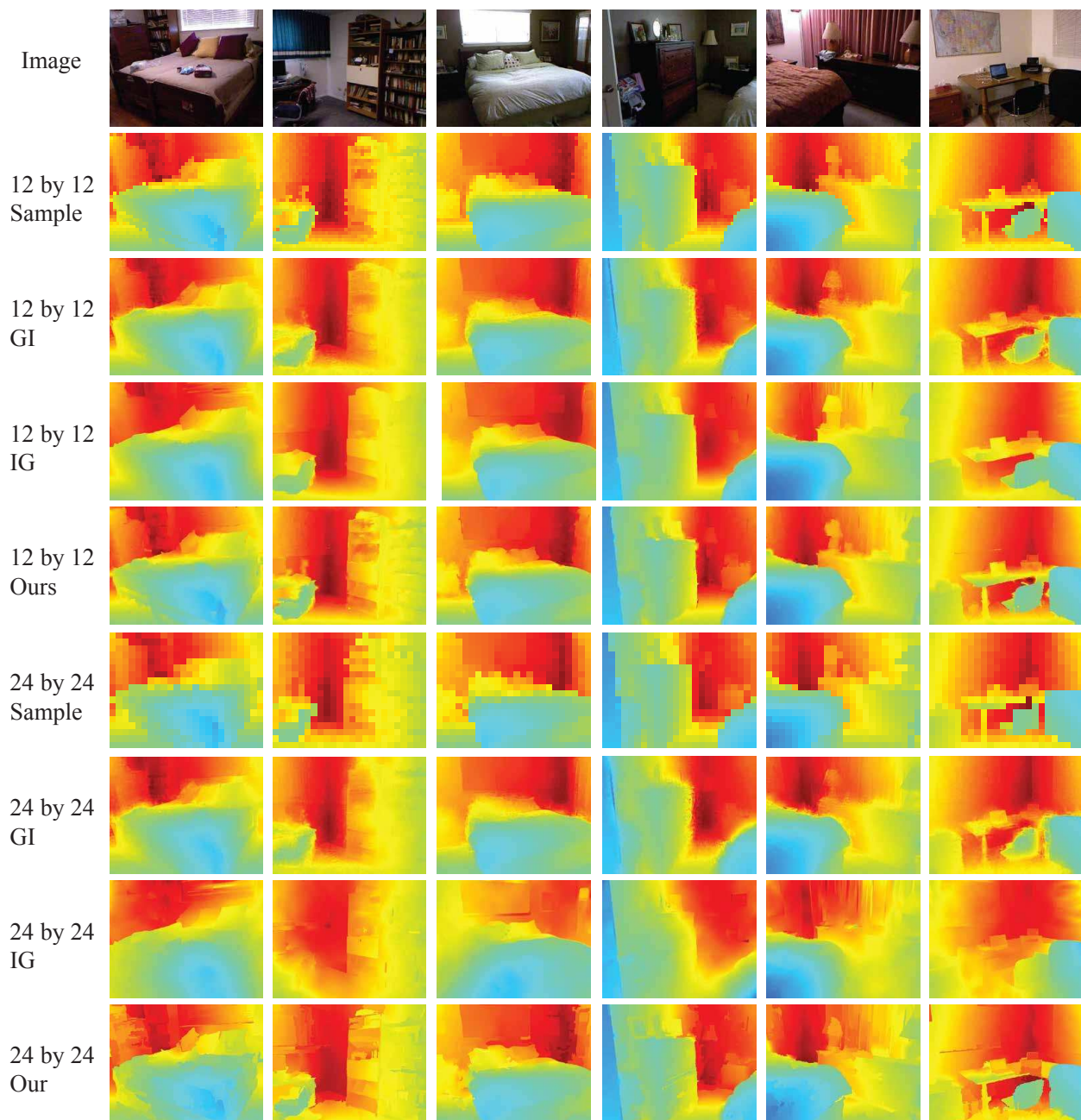


Figure 5. More results for NYU dataset (GI= [3]; IG= [2]).

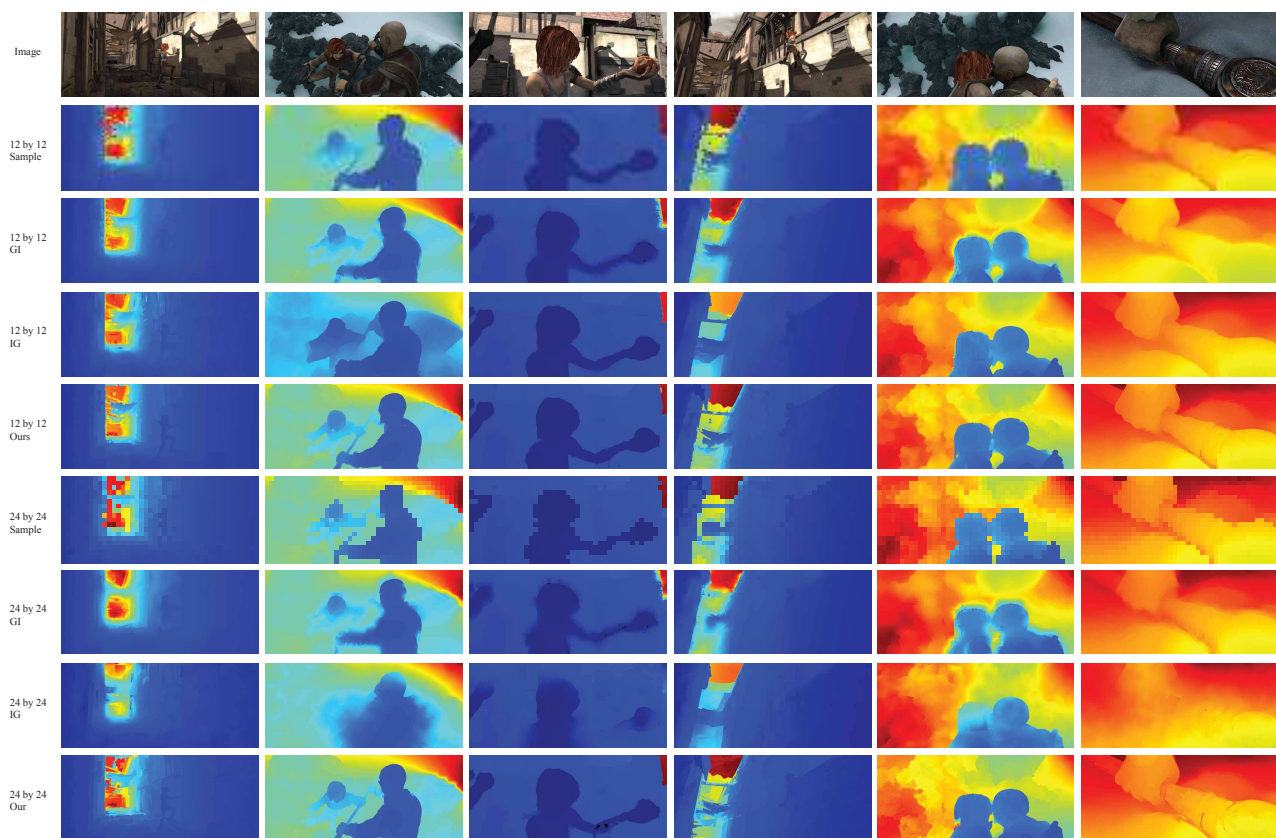


Figure 6. More results for Sintel dataset (GI= [3]; IG= [2]).

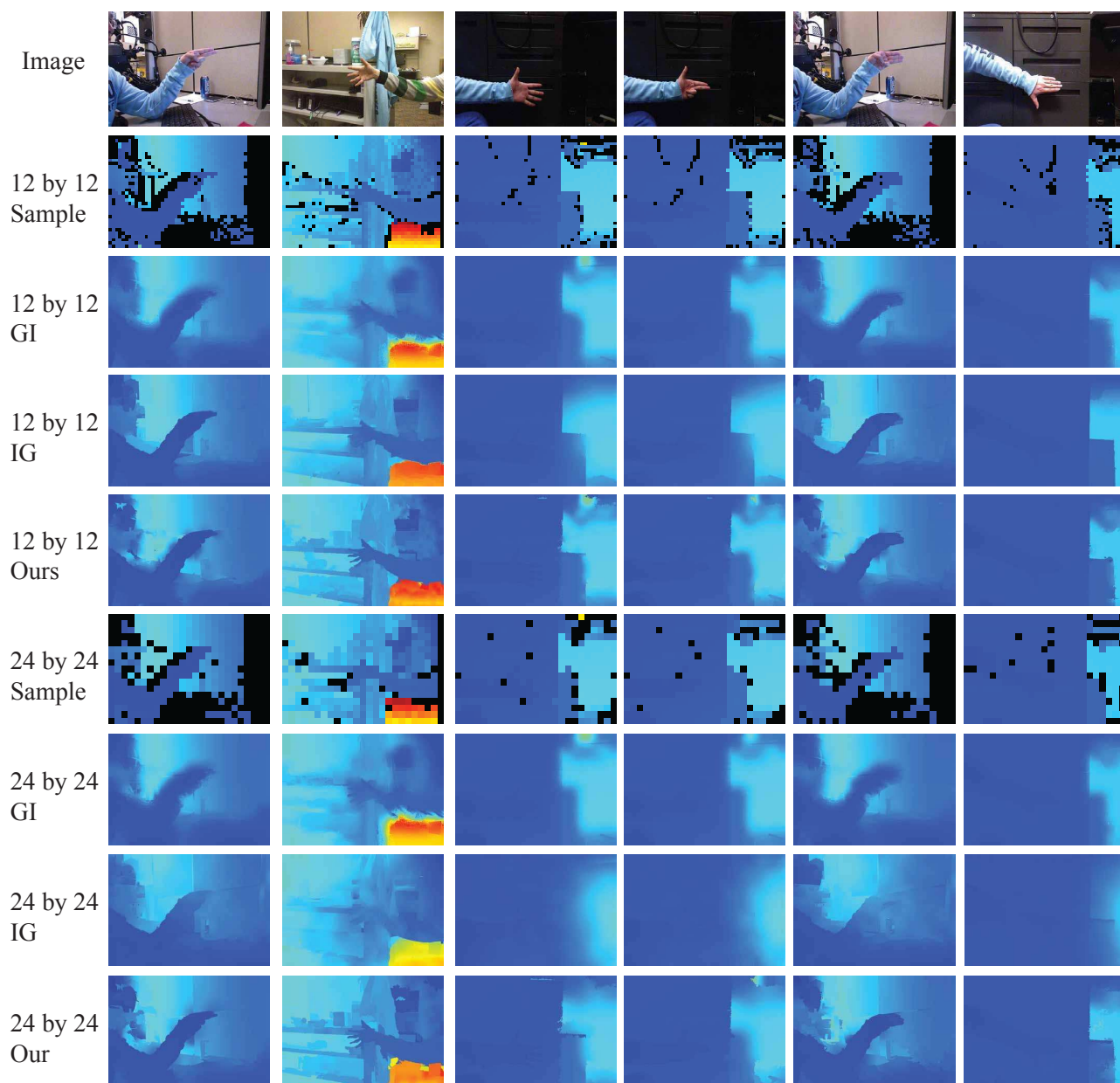


Figure 7. More results for Gesture dataset (GI= [3]; IG= [2]).

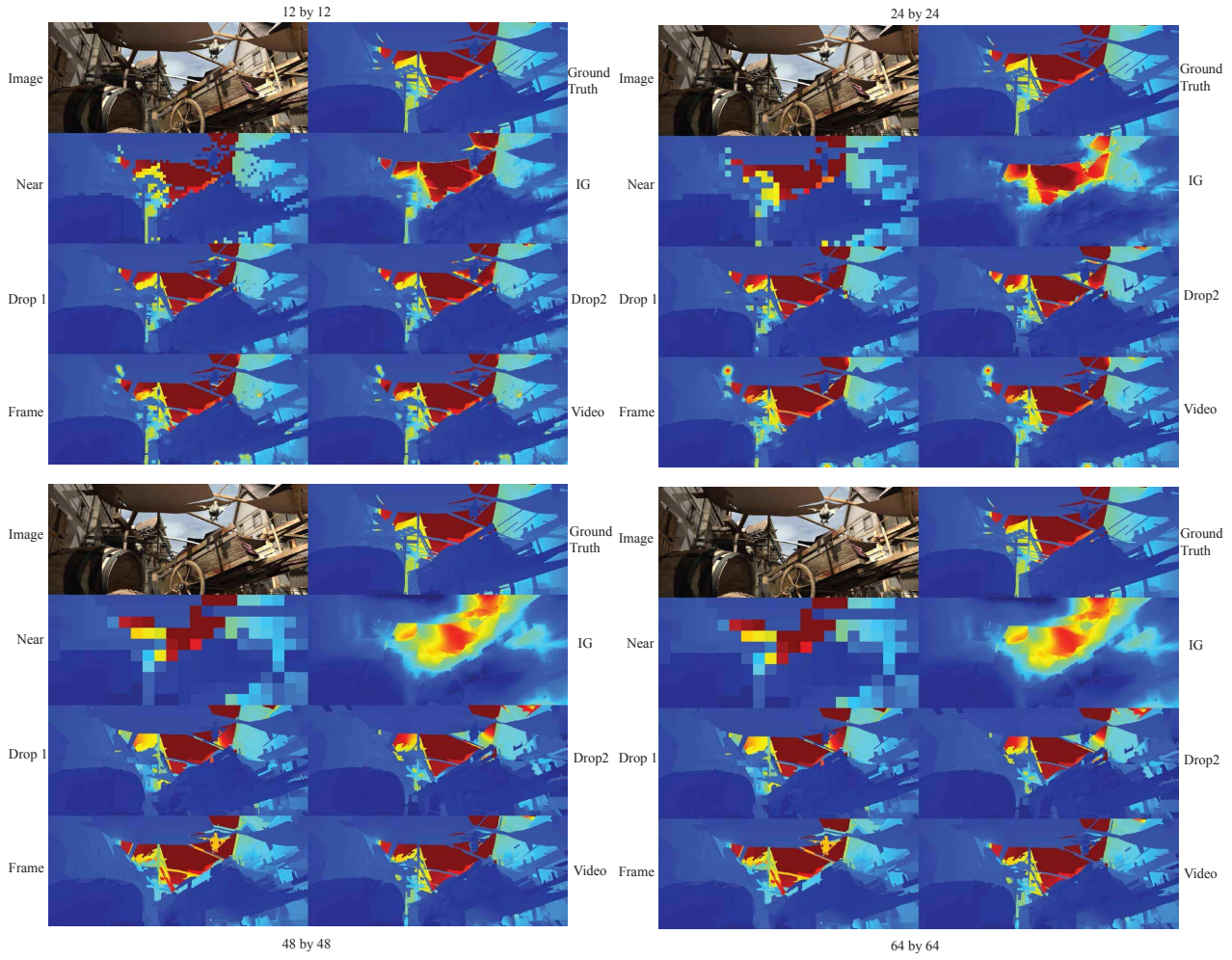


Figure 8. Depth super resolution from video on market sequence (Drop1=skip 1 depth frame out of every 2 depth frames; Drop2=skip 2 depth frames out of every 3 depth frames; Near=nearest neighbor; IG= [2]).

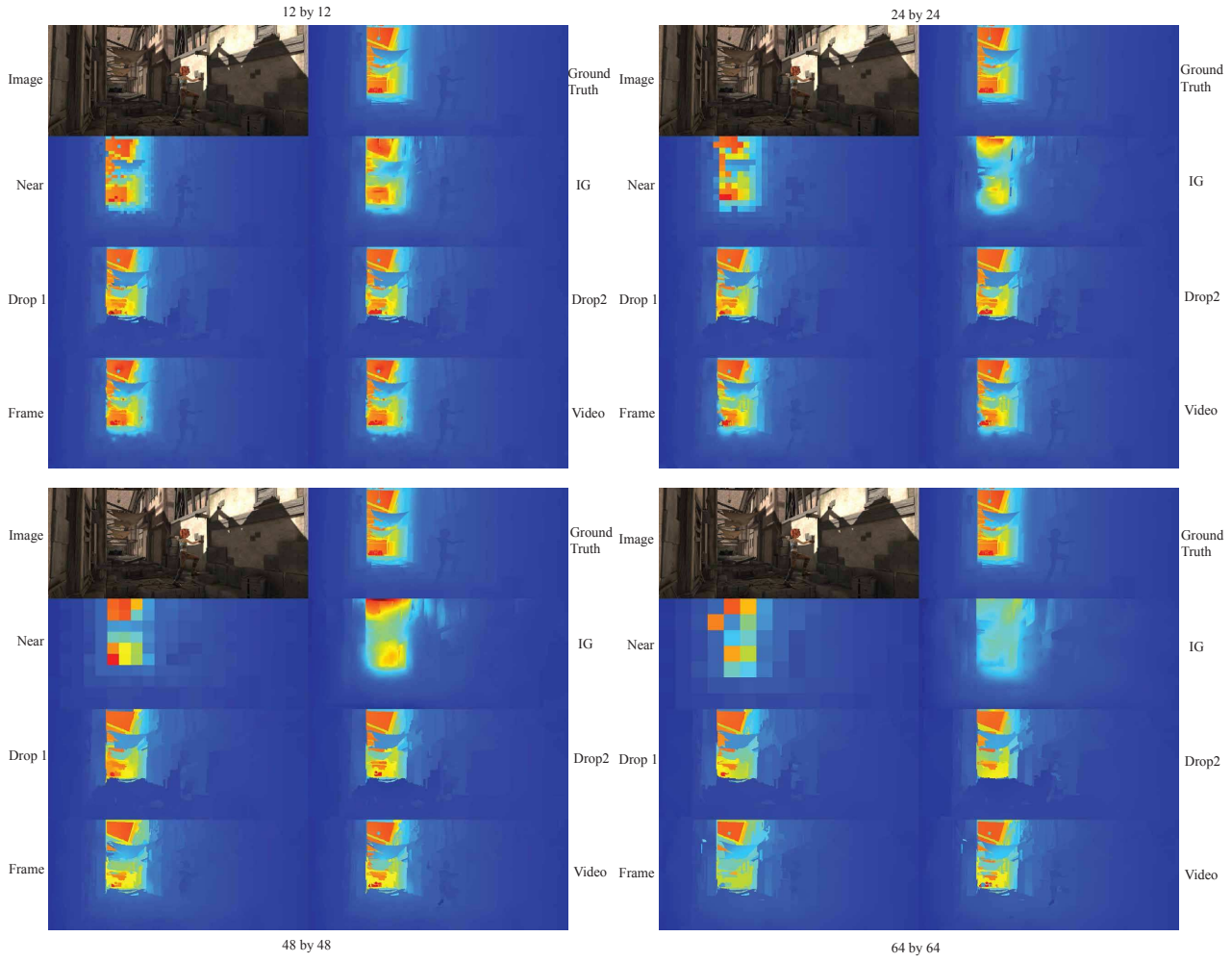


Figure 9. Depth super resolution from video on alley sequence (Drop1=skip 1 depth frame out of every 2 depth frames; Drop2=skip 2 depth frames out of every 3 depth frames; Near=nearest neighbor; IG= [2]).

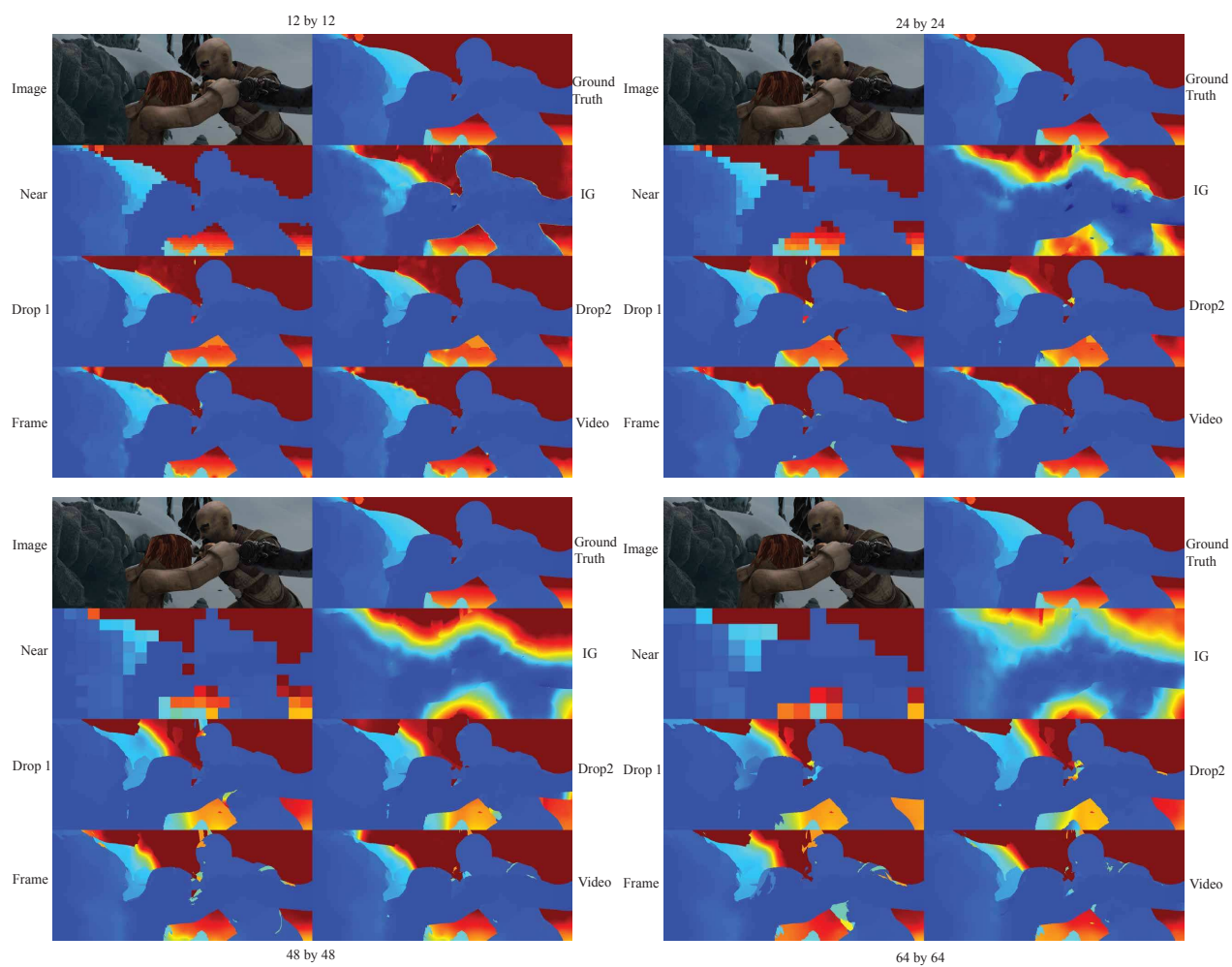


Figure 10. Depth super resolution from video on ambush sequence (Drop1=skip 1 depth frame out of every 2 depth frames; Drop2=skip 2 depth frames out of every 3 depth frames; Near=nearest neighbor; IG= [2]).

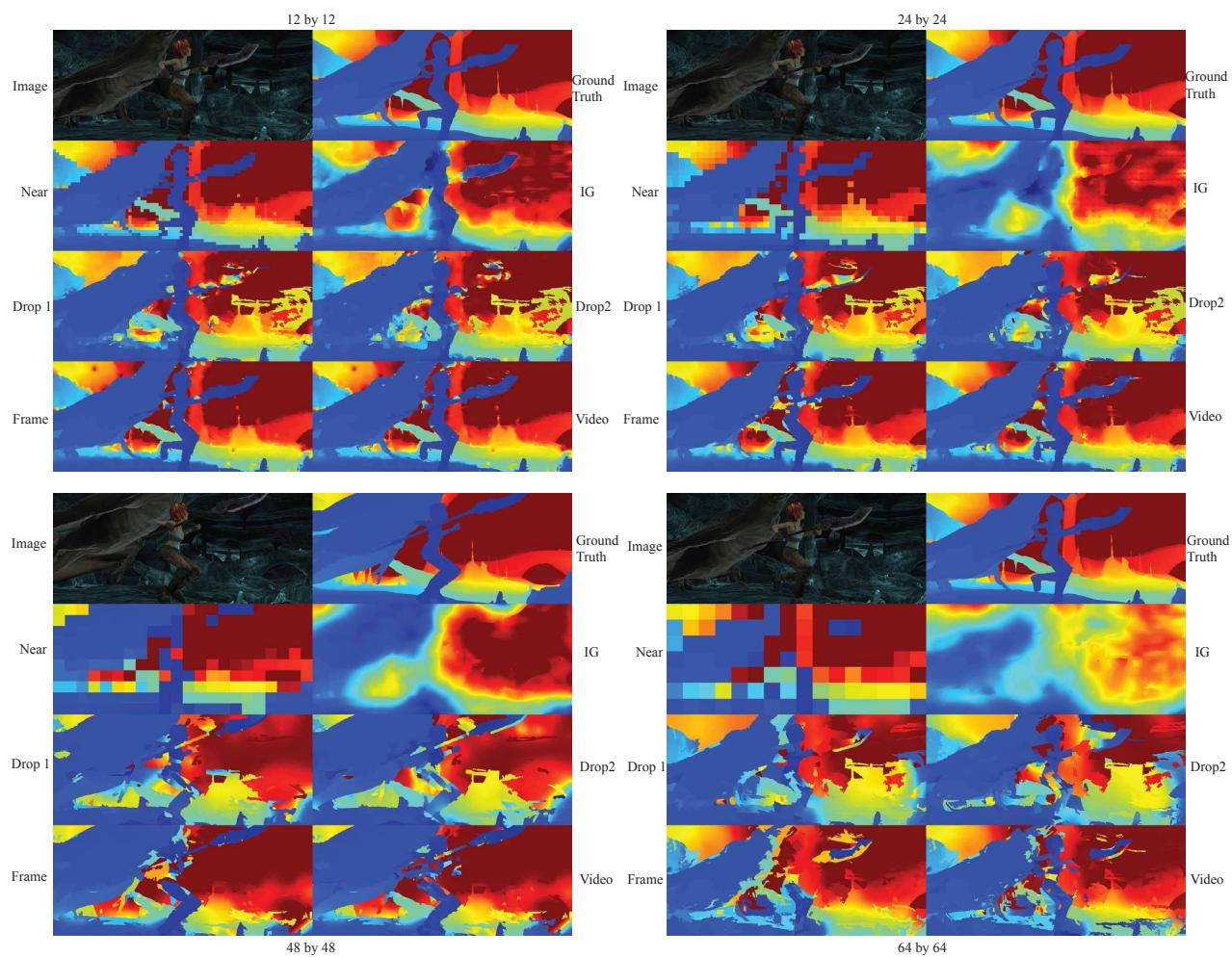


Figure 11. Depth super resolution from video on cave sequence (Drop1=skip 1 depth frame out of every 2 depth frames; Drop2=skip 2 depth frames out of every 3 depth frames; Near=nearest neighbor; IG= [2]).

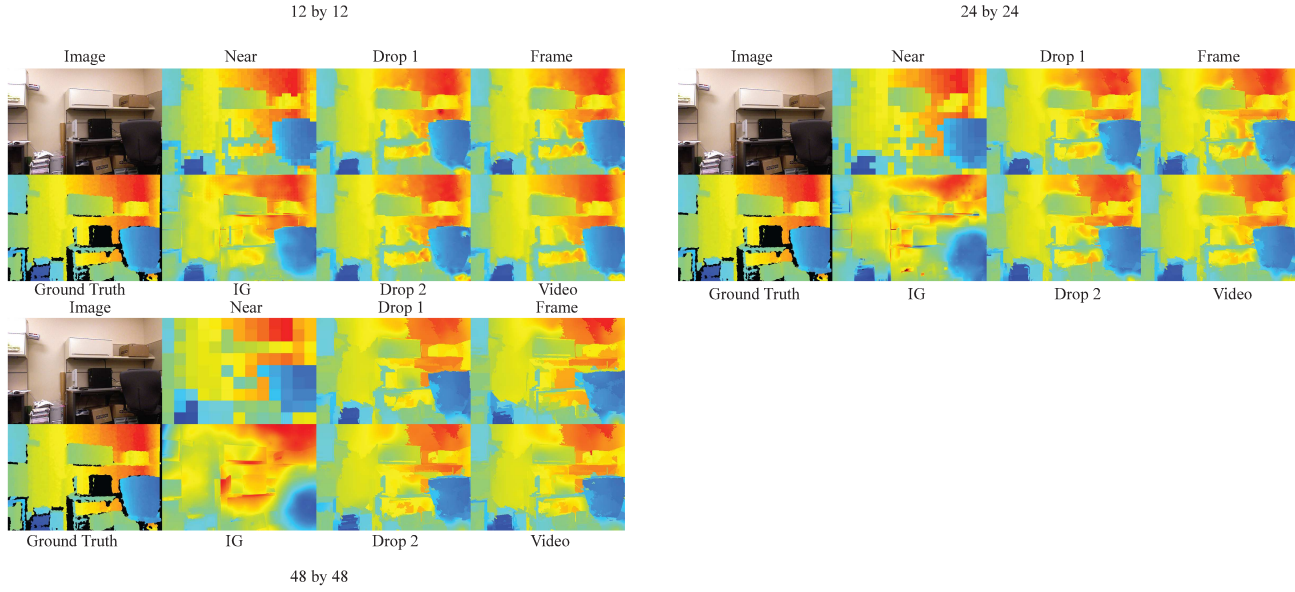


Figure 12. Depth super resolution from video on office sequence (Drop1=skip 1 depth frame out of every 2 depth frames; Drop2=skip 2 depth frames out of every 3 depth frames; Near=nearest neighbor; IG= [2]).

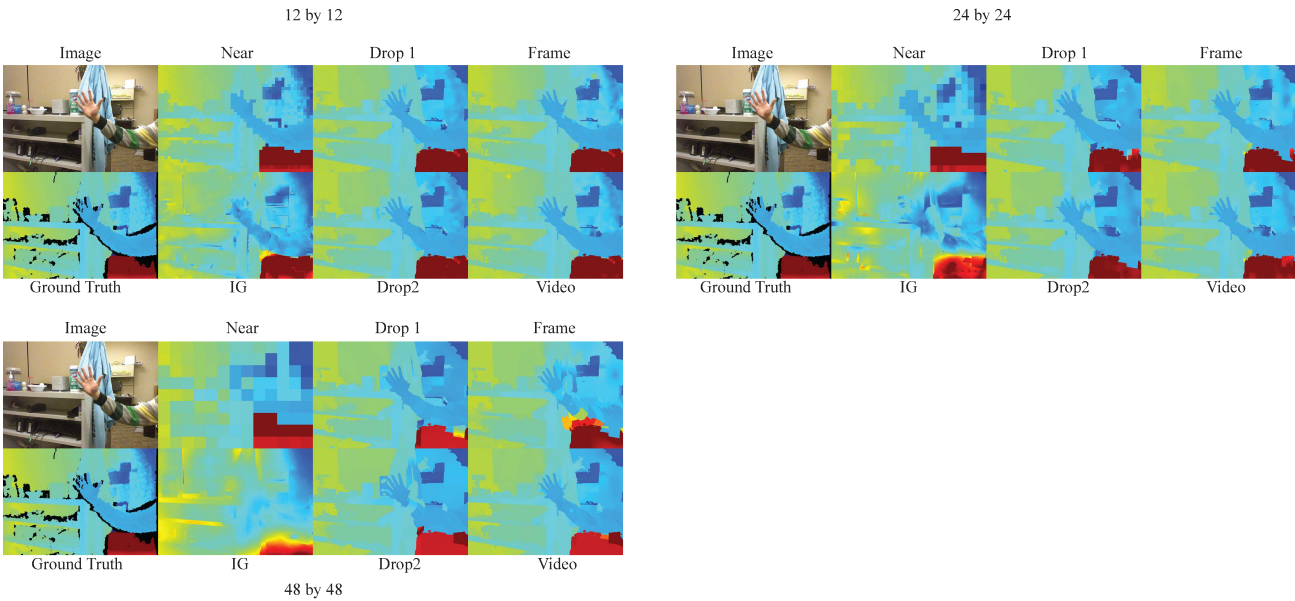


Figure 13. Depth super resolution from video on gesture sequence (Drop1=skip 1 depth frame out of every 2 depth frames; Drop2=skip 2 depth frames out of every 3 depth frames; Near=nearest neighbor; IG= [2]).



Figure 14. Object insertion using image depth transfer depth and our 24×24 times super resolution depth. Transfer depth fails in many places, especially boundaries, and our results are accurate everywhere. See also in movie.