TACO: Taming Diffusion for in-the-wild Video Amodal Completion

Supplementary Material

We provide details on data curation, implementation, additional qualitative results, and a discussion of limitations in *supplementary materials*.

A. Data Curation Details

In this section, we elaborate on the details of how we curate the Object-video-Overlay (OvO) dataset, including how to overlay occluders consistently throughout the video in Appendix A.1, how we apply image transformation techniques to augment our dataset with image-level datasets in Appendix A.2, and other curation details in Appendix A.3.

A.1. Overlay Occluders Consistently

To ensure consistent occlusions, it is crucial to maintain continuous change in the occluders' properties across video frames. This includes the occluders' position, $\mathbf{p}$, which specifies their position, and scale, $\mathbf{s}$, which determines their size. We employ two heuristic strategies in OvO-Easy and OvO-Hard dataset to generate occlusions.

OvO-Easy Occluders are selected from Objaverse [4, 5] and SA-1B [10]. Appropriate occlusion positions are identified for the first and last frames of the video. The occlusion rate, defined as the ratio of the occluded area to the total area of the object, is constrained to lie between 0.3 and 0.7 in both the first and last frames. For the intermediate frame i , the occlusion position $\mathbf{p}_i$ and scale $\mathbf{s}_i$ are determined through linear interpolation, ensuring smooth transitions:

$$\mathbf{p}_i = \frac{i}{N}\cdot(\mathbf{p}_{\text{ed}}-\mathbf{p}_{\text{st}})+\mathbf{p}_{\text{st}},\ \mathbf{s}_i = \frac{i}{N}\cdot(\mathbf{s}_{\text{ed}}-\mathbf{s}_{\text{st}})+\mathbf{s}_{\text{st}}, \quad \text{(S.1)}$$

where $\mathbf{p}_{\text{st}}, \mathbf{s}_{\text{st}}, \mathbf{p}_{\text{ed}}, \mathbf{s}_{\text{ed}}$ denotes the occlusion position and scale in the first and last frame, and N stands for the total frame number. An example is illustrated in Fig. S.1. Apart from a relatively low occlusion rate, some objects are fully visible in intermediate frames.

OvO-Hard In contrast, OvO-Hard begins by selecting an initial occlusion position $\mathbf{p}_{\text{st}}$ and scale $\mathbf{s}_{\text{st}}$ in the first frame. The position and size of the occluder are then dynamically adjusted throughout the video, guided by changes in the bounding box of the occluded object:

$$\mathbf{p}_i = \mathbf{c}_i - \mathbf{c}_{\text{st}} + \mathbf{p}_{\text{st}},\ \mathbf{s}_i = max(\frac{\mathbf{h}_i}{\mathbf{h}_{\text{st}}}, \frac{\mathbf{w}_i}{\mathbf{w}_{\text{st}}})^{1/3}\cdot \mathbf{s}_{\text{st}}, \quad \text{(S.2)}$$

where $\mathbf{c}_{\text{st}}$ and $\mathbf{c}_i$ stands for the center of the bounding box in the initial frame and frame i, $\mathbf{h}_i$, $\mathbf{w}_i$, $\mathbf{h}_{\text{st}}$, and $\mathbf{w}_{\text{st}}$ stands for the height and width of the $i-th$ frame and the initial frame. In OvO-Hard, we also apply image feathering techniques to blend the occluder more naturally into the image. The occlusion rate of the initial frame is constrained to lie between 0.4 and 0.8.

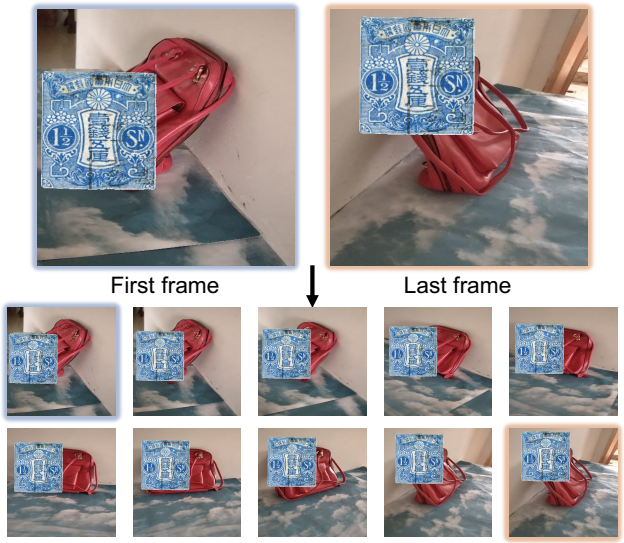

Figure S.1. **Illustration on consistent occlusion.** In OvO-Easy, we first select a proper position and scale in the first and last frame, and then interpolate in intermediate frames. The stamp is the occluder in the example.

A.2. Image Transformations

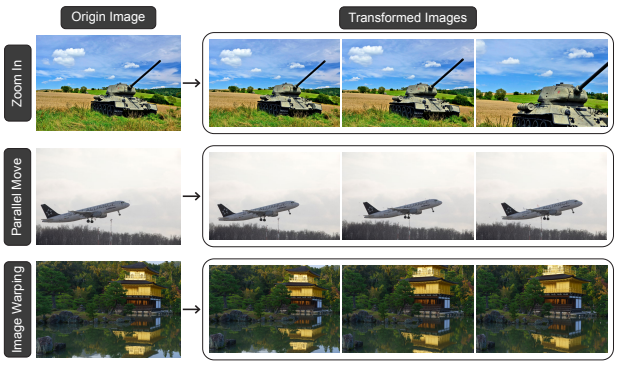

Figure S.2. **Image transformations.** We augment our OvO dataset by incorporating image-level datasets and image transformation techniques to simulate videos.

In addition to introducing more severe occlusions, we further augment our dataset in OvO-Hard using the image-level dataset SA-1B [10]. Specifically, we apply three image transformation techniques: zooming, parallel moving,

and image warping. An illustration of these techniques is shown in Fig. S.2. The zooming transformation is implemented by center-cropping the image, while image warping is achieved through a homography transformation. For parallel moving, we first segment the complete foreground object using the provided annotations [10, 13]. The background area is then inpainted, after which the foreground and background are shifted in parallel to create a motion effect. For instance, in Fig. S.2, the plane visually appears to move to the right.

A.3. Other Details

In this section, we will elaborate on the data sources, the process of occluder selection, and the details of amodal check including heuristic rules and manual filtering.

Data sources To ensure the diversity of our dataset OvO, we conducted experiments using subsets of four datasets. After applying the amodal check, the final dataset consists of approximately 90K videos from MVImgNet [22], 11K videos from SA-V [14], 10K videos from Bdd100k [21], and 24K videos from SA-1B [10]. For constructing OvO-Easy, we use data from MVImgNet [22] and SA-V [14]. For the more challenging OvO-Hard, we further add data from Bdd100k [21] and SA-1B [10]. Additionally, we reserve approximately 1K videos as the test sets for OvO-Easy and OvO-Hard, which are used to benchmark our method against various baselines.

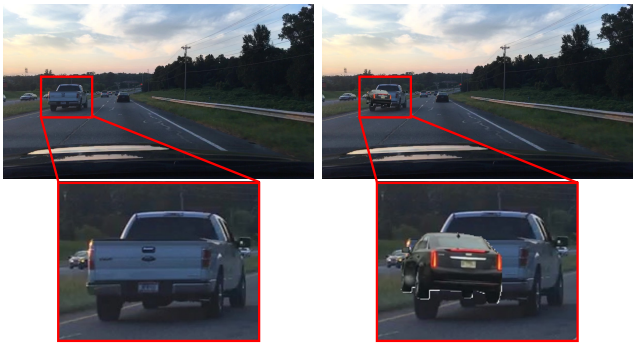

Figure S.3. **Illustration on Bdd100K occluders.** We select occluders from Bdd100K since many of the occluders in SA-1B are not applicable to the autonomous driving context.

Occluders In OvO-Easy, we select approximately 50K occluders from a subset of SA-1B [10] and around 20K occluders from a subset of Objaverse [5]. For occluders sourced from SA-1B, objects are segmented using the provided annotations, ensuring that the occluder occupies a significant portion of the image. For occluders from Objaverse, we render rotation videos consisting of 40 frames using Blender [2]. In OvO-Hard, occluders from Objaverse are excluded to enhance realism. Additionally, when curating data pairs for Bdd100k [21], we select occluders directly from Bdd100k [21] to maintain domain relevance, as many occluders from SA-1B and Objaverse are not applicable to the autonomous driving context. An example on the occluders of Bdd100k is shown in Fig. S.3.

Figure S.4. **Illustration on the heuristic amodal check.** We apply three heuristic rules to check whether an object is complete.

Amodal check We apply three heuristic rules to filter out candidates likely to be incomplete, as illustrated in Fig. S.4. In the first example, the mask touches the image boundary, indicating potential incompleteness. The second example shows a mask with numerous internal holes, suggesting an inaccurate segmentation. In the third example, the pillow is positioned behind the foreground bag, failing the depth consistency check. Despite these heuristic rules, some incomplete object candidates remain. To address this, we leverage crowdsourcing to further filter out incomplete samples that have incorrectly passed the heuristic amodal check.

B. Implementation Details

In this section, we provide detailed explanations of the training and inference process in Appendix B.1, the dataset, metrics and baselines used for amodal completion and segmentation in Appendix B.2, the dataset curation and reconstruction pipeline for object reconstruction in Appendix B.3, the intermediate results for pose estimation in Appendix B.4, and details on user study in Appendix B.5.

B.1. Training and Inference Details

We trained our model on the OvO-Easy dataset for 7 epochs and continued training on the OvO-Hard dataset for another 7 epochs, resulting in a total training time of approximately 6 days using 8 NVIDIA A800 (80G) GPUs. Due to computational constraints, all input and target videos were resized to a resolution of 384×384. The batch size per GPU was set to 4, yielding a total batch size of 32. To balance the dataset, data pairs from SA-V [14] were sampled twice

per epoch. As noted in prior work [11, 16], Stable Video Diffusion (SVD)[1] is not robust to variations in resolution. To address this, we further fine-tuned the model on the Bdd100k subset for 8 epochs at a resolution of 640×384, taking around 20 hours. This is because the typical resolution used in autonomous driving scenarios significantly differs from those in MVImgNet[22] and SA-V [14]. We employed the SVD version configured to predict 14 frames. To handle additional visible mask inputs, extra channels in the first layer were added after concatenation and initialized to zero. A freeze motion bucket and fixed frame rate were used for simplicity. During inference, conditional samples were generated using the EDM sampler with 50 steps [9], taking approximately 20 seconds to produce an output video.

B.2. Amodal Completion and Segmentation

Test Dataset In addition to the test split of OvO-Easy and OvO-Hard, we curate two additional datasets, Kubric-Static and Kubric-Dynamic, using the Kubric simulator [8] to benchmark the generalizability of the models. For Kubric-Static, we randomly select 2 to 5 objects from GSO [6], along with a background dome. The objects are placed in a static scene with randomized scales and positions. A rotating video is rendered using blender [2] by rotating the camera around the scene, capturing each object's modal and amodal masks as well as amodal RGB images. From the rendered videos, we filter 365 samples with occlusions for benchmarking, ensuring that at least 10 out of 14 frames in each video contain occluded objects. For Kubric-Dynamic, we randomly select 2 to 3 falling objects and combine them with 2 to 4 static objects placed on the ground as well as a background dome. A linearly changing camera trajectory is applied before rendering each frame. Using the same filtering criteria as in Kubric-Static, we select 323 videos with occlusions for benchmarking.

Metrics Since a significant portion of the synthesized frames is white, this can inflate the PSNR, SSIM, and LPIPS values. To address this, we crop the synthesized amodal object using a dilated bounding box of the ground-truth (GT) amodal mask and compute the PSNR, SSIM, LPIPS, and CLIP-T metrics within this specific region.

Baselines Since pix2gestalt [13] can only generate images at a resolution of 256×256, we resize all results to this resolution when calculating the metrics. For the E^2FGVI baseline, we observed significantly degraded performance when masking out all but the visible object area. To address this, we first change the background to white and then use the same dilated bounding box of the ground-truth (GT) amodal mask as the inpainting mask. An illustrative comparison of the inpainting masks and the corresponding results is provided in Fig. S.5.

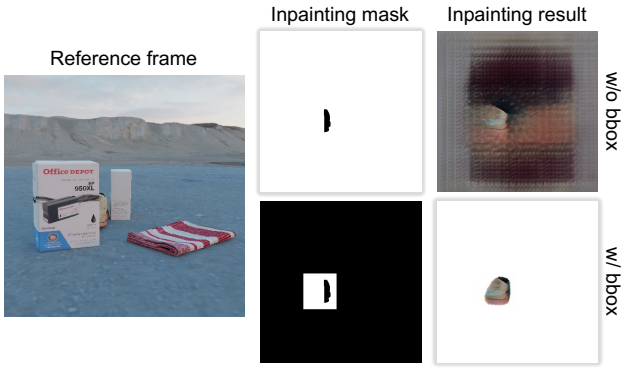

Figure S.5. **Comparison on inpainting mask.** We try two kind of inpainting mask for the E^2FGVI baseline, and the one with bounding box constraint is significantly better.

B.3. Object Reconstruction

Dataset We also use objects from GSO [6] to create compositional occluded scenes. Specifically, we first place a primary object at the center of the scene. Then, we select 3 to 6 additional objects along with a background dome and arrange them around the central object to ensure occlusion occurs. Finally, we rotate the camera around the center of the scene to generate an occluded video. A total of 20 scenes are composed for benchmarking.

Reconstruction We directly apply NeRF2Mesh [15] to the synthesized results for reconstruction, where images with better cross-view consistency result in higher reconstruction quality. To enhance performance, NeRF2Mesh also requires an object mask during reconstruction. We obtain this mask by thresholding the synthesized images.

B.4. Pose Estimation

We select a video in YCB-Video [18] and utilize SAM2 [14] to acquire the visible masks of an object throughout the video. An intermediate result (synthesized video of the amodal object) using our method is shown in Fig. S.6.

B.5. User Study

We select 20 videos from unseen datasets including ScanNet++ [3, 20], BridgeData [7, 17], YouTube-VOS [19], YCB-Video [18], and various Internet videos. Each questionnaire contains 16 questions, asking participants to evaluate the completion results across three dimensions: 1) Content-Quality: The overall quality of the completed content, 2) Content-Consistency: Temporal consistency of the completed object, and 3) Content-Plausibility: Whether the completed object is semantically aligned with its visible part. A snapshot of the questionnaire interface is shown in Fig. S.11. We gathered feedback from 36 participants, resulting in 576 valid responses.

Figure S.6. **Intermediate results on YCB-Video.** We visualize the intermediate synthesized video on YCB-Video.

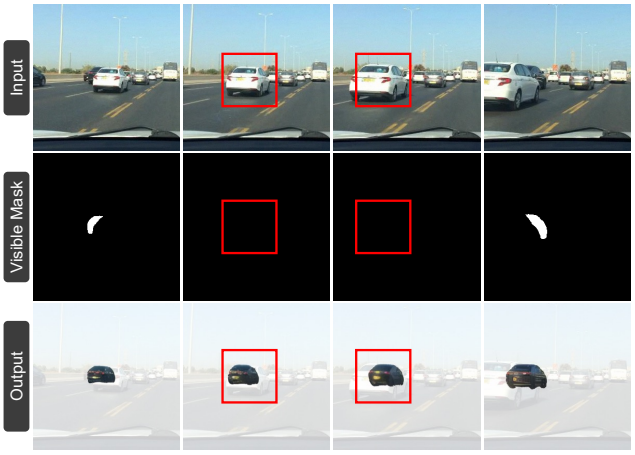

Figure S.7. **Revealing missing objects.** Our method is able to reveal missing objects by inferring from neighboring frames.

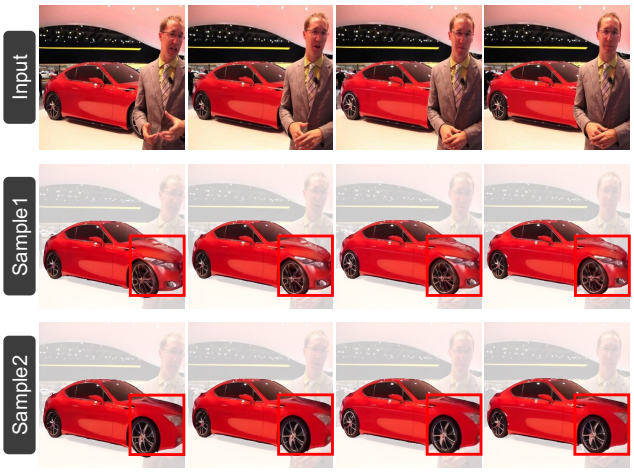

Figure S.8. **Diversity in sampling.** We can sample multiple reasonable results due to the inherent ambiguity in occluded area.

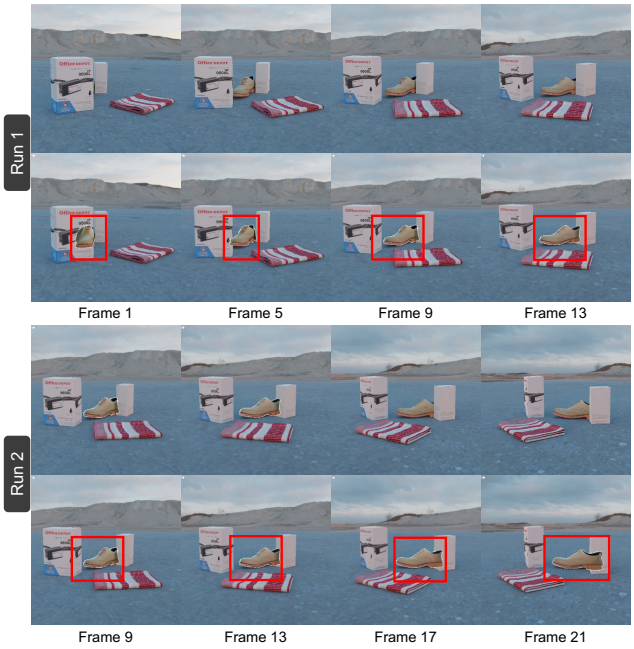

Figure S.9. **Tackling long videos.** Our method is able to extend beyond 14 frames by progressive generation.

C. Additional Results

We highly recommend browsing the website, which contains comparisons with baselines, more qualitative results on diverse datasets, and examples of long videos.

C.1. Image Amodal Completion

We also evaluate our method on two image-based datasets to test whether our model still works for image-level tasks. We replicate each image 14 times to simulate static videos.

- **BSDS-A**: We evaluate amodal segmentation on BSDS-A [12], using the same test split as *pix2gestalt* (P2G). Since each image has multiple annotations and the specific annotations used in P2G are unavailable, we randomly select one annotation per image, resulting in 730 objects across 200 images. We perform single-shot inference for both methods, which may cause discrepancies from the 16-shot setting in P2G paper. Nonetheless, the overall conclusion remains consistent.
- **Kubric-Img**: We test on 750 images from the Kubric-Static dataset with occlusion rates between 30% and 70% to evaluate both amodal completion and segmentation.

Quantitative results in Tabs. S.1 and S.2 as well as visualized results in Fig. S.12 show our method performs on

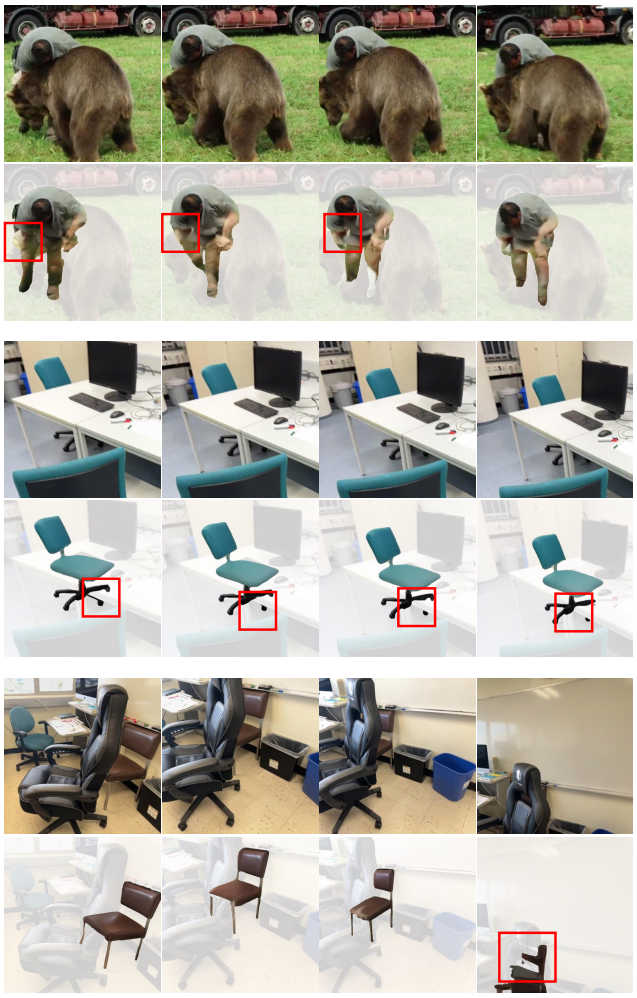

Figure S.10. **Failure cases.** Our method has limitations under certain challenging conditions. For instance, it may produce blurry results in areas with complex occlusions, as seen in the hand region of the first example. Similarly, it struggles to handle extremely fine-grained and heavily occluded structures, as demonstrated in the leg area of the second example. Additionally, our method may fail to perform effectively during drastic and sudden camera movements as shown in the last frame in the third example.

Table S.1. BSDS-A

	Modal	P2G	Ours
IoU	57.7	**68.9**	67.6

Table S.2. Kubric-Img

	PSNR	SSIM	LPIPS	IoU
P2G	**17.578**	0.781	0.153	**75.6**
Ours	17.471	**0.782**	**0.146**	74.4

par with image-based methods, indicating its ability to synthesize plausible amodal content, *even without video context*. We attribute this to the OvO-Hard dataset design, where persistent occlusion forces the model to learn genuinely consistent amodal representations rather than merely rely on video cues. Ensuring such consistency makes VAC innately more complex than image-based tasks.

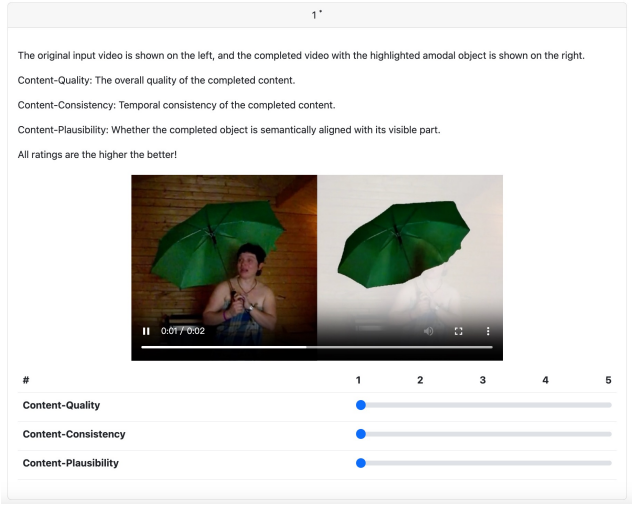

Figure S.11. **Interface of the user study.**

Figure S.12. Visualization on BSDS-A

C.2. More Comparisons

We provide qualitative comparison on ScanNet [3, 20], BridgeData [7, 17], and YCB-Video [18] in Fig. S.13. More visualized comparisons on Bdd100k [21] and YouTube-VOS are provided in Fig. S.14. The original resolution of Bdd100k [21] used for inference is 640×384, we crop out the region of interest for better visualization. Similarly, for pix2gestalt [13], we crop a square area as input, as the method requires square images.

C.3. More Qualitative Results

We provide more qualitative results of our method on on ScanNet [3, 20] in Fig. S.15, Fig. S.16, and Fig. S.17. More qualitative results on BridgeData [7, 17] are provided in Fig. S.18, Fig. S.19, and Fig. S.20. More qualitative results on YouTube-VOS [19] are provided in Fig. S.21 and Fig. S.22. More qualitative results on YCB-Video [18] are provided in Fig. S.23 and Fig. S.24. More intuitive visualized results on various datasets and in-the-wild videos are provided in the local website attached in *supplementary materials*.

C.4. Revealing Missing Objects

An object may be completely invisible (occluded by other objects) in a video clip. We observe that our method is able to reveal completely missing objects by aggregating information from neighboring frames in certain cases. For example, as shown in Fig. S.7, the black car is completely occluded by the white car in front of it in the middle frames,

and our method is able to hallucinate the position, shape, and appearance of the black car.

C.5. Tackling Long Videos

To extend beyond 14 frames, we introduce a sliding window mechanism for progressively generating subsequent frames, as illustrated in Fig. S.9. In the first run, the initial 14 frames are selected, and the amodal object is synthesized. The synthesized object is then blended back into the video, as shown in the second row. For the second run, subsequent frames are concatenated with the previously blended frames as the input video. The visible masks of the overlap frames are acquired by thresholding the synthesized object. This sliding window approach allows our method to effectively generate videos exceeding 14 frames. Two additional qualitative examples are available on the local website.

C.6. Diversity in Sampling

Since amodal completion possesses inherent ambiguity, we can synthesize multiple reasonable results, and a qualitative example indicating the diversity is shown in Fig. S.8.

C.7. Failure Cases

We illustrate several failure cases in Fig. S.10. In the first example, nearly the entire legs and arms of the human are occluded, representing a severely occluded scenario. Under such conditions, our method may produce low-quality outputs, such as blurry hands and occasionally missing legs. In the second example, the chair is heavily occluded by the white table. While it can be inferred that the chair has thin legs, our method may struggle with accurately reconstructing thin structures in certain frames. In the third example, the camera undergoes drastic movements, and the brown chair becomes heavily occluded from specific viewpoints. Our method struggles to recover accurate and consistent results when the object is almost missing.

D. Limitations and Negative Impacts

Limitations We have discussed about several failure cases in Appendix C.7. Additionally, our method is sensitive to resolution variations due to the constraints of the SVD architecture. While we can extend beyond 14 frames, generating extremely long videos remains a challenge. We hope that advancements in more powerful video diffusion models will help address these issues in the future. Moreover, as our method incorporates data from only a few datasets, as mentioned in Appendix A.3, its performance may degrade when generalizing to vastly different domains, such as human interactions with other objects, as shown in the example in Fig. S.10. Incorporating more diverse training data and curating realistic occlusion scenarios could help mitigate this issue.

Negative Impacts The use of diffusion models to generate content raises significant ethical concerns, including potential privacy violations and the risk of generating biased content. These models can be misused to spread misinformation or serve deceptive purposes, eroding trust and causing societal harm. Additionally, they may produce misleading or false outputs, causing potential challenges in fields where accuracy and reliability are crucial.

References

[1] Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023. 3

[2] Blender Online Community. *Blender - a 3D modelling and rendering package*. Blender Foundation, Stichting Blender Foundation, Amsterdam, 2018. 2, 3

[3] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 3, 5, 8, 10, 11, 12

[4] Matt Deitke, Ruoshi Liu, Matthew Wallingford, Huong Ngo, Oscar Michel, Aditya Kusupati, Alan Fan, Christian Laforte, Vikram Voleti, Samir Yitzhak Gadre, et al. Objaverse-xl: A universe of 10m+ 3d objects. *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. 1

[5] Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3d objects. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 1, 2

[6] Laura Downs, Anthony Francis, Nate Koenig, Brandon Kinman, Ryan Hickman, Krista Reymann, Thomas B McHugh, and Vincent Vanhoucke. Google scanned objects: A high-quality dataset of 3d scanned household items. In *International Conference on Robotics and Automation (ICRA)*, 2022. 3

[7] Frederik Ebert, Yanlai Yang, Karl Schmeckpeper, Bernadette Bucher, Georgios Georgakis, Kostas Daniilidis, Chelsea Finn, and Sergey Levine. Bridge data: Boosting generalization of robotic skills with cross-domain datasets. *arXiv preprint arXiv:2109.13396*, 2021. 3, 5, 8, 13, 14, 15

[8] Klaus Greff, Francois Belletti, Lucas Beyer, Carl Doersch, Yilun Du, Daniel Duckworth, David J Fleet, Dan Gnanapragasam, Florian Golemo, Charles Herrmann, et al. Kubric: A scalable dataset generator. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 3

[9] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. 3

[10] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment

anything. In *International Conference on Computer Vision (ICCV)*, 2023. 1, 2

[11] Ruining Li, Chuanxia Zheng, Christian Rupprecht, and Andrea Vedaldi. Puppet-master: Scaling interactive video generation as a motion prior for part-level dynamics. *arXiv preprint arXiv:2408.04631*, 2024. 3

[12] David Martin, Charless Fowlkes, Doron Tal, and Jitendra Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *International Conference on Computer Vision (ICCV)*, 2001. 4

[13] Ege Ozguroglu, Ruoshi Liu, Dídac Surís, Dian Chen, Achal Dave, Pavel Tokmakov, and Carl Vondrick. pix2gestalt: Amodal segmentation by synthesizing wholes. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 2, 3, 5

[14] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. Sam 2: Segment anything in images and videos. In *International Conference on Learning Representations (ICLR)*, 2025. 2, 3

[15] Jiaxiang Tang, Hang Zhou, Xiaokang Chen, Tianshu Hu, Errui Ding, Jingdong Wang, and Gang Zeng. Delicate textured mesh recovery from nerf via adaptive surface refinement. In *International Conference on Computer Vision (ICCV)*, 2023. 3

[16] Basile Van Hoorick, Rundi Wu, Ege Ozguroglu, Kyle Sargent, Ruoshi Liu, Pavel Tokmakov, Achal Dave, Changxi Zheng, and Carl Vondrick. Generative camera dolly: Extreme monocular dynamic novel view synthesis. In *European Conference on Computer Vision (ECCV)*, 2024. 3

[17] Homer Rich Walke, Kevin Black, Tony Z Zhao, Quan Vuong, Chongyi Zheng, Philippe Hansen-Estruch, Andre Wang He, Vivek Myers, Moo Jin Kim, Max Du, et al. Bridgedata v2: A dataset for robot learning at scale. In *Conference on Robot Learning (CoRL)*, 2023. 3, 5, 8, 13, 14, 15

[18] Yu Xiang, Tanner Schmidt, Venkatraman Narayanan, and Dieter Fox. Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes. *arXiv preprint arXiv:1711.00199*, 2017. 3, 5, 8, 16, 17

[19] Ning Xu, Linjie Yang, Yuchen Fan, Dingcheng Yue, Yuchen Liang, Jianchao Yang, and Thomas Huang. Youtube-vos: A large-scale video object segmentation benchmark. *arXiv preprint arXiv:1809.03327*, 2018. 3, 5, 9, 18, 19

[20] Chandan Yeshwanth, Yueh-Cheng Liu, Matthias Nießner, and Angela Dai. Scannet++: A high-fidelity dataset of 3d indoor scenes. In *International Conference on Computer Vision (ICCV)*, 2023. 3, 5, 8, 10, 11, 12

[21] Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2, 5, 9

[22] Xianggang Yu, Mutian Xu, Yidan Zhang, Haolin Liu, Chongjie Ye, Yushuang Wu, Zizheng Yan, Chenming Zhu, Zhangyang Xiong, Tianyou Liang, et al. Mvimgnet: A large-scale dataset of multi-view images. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 2, 3

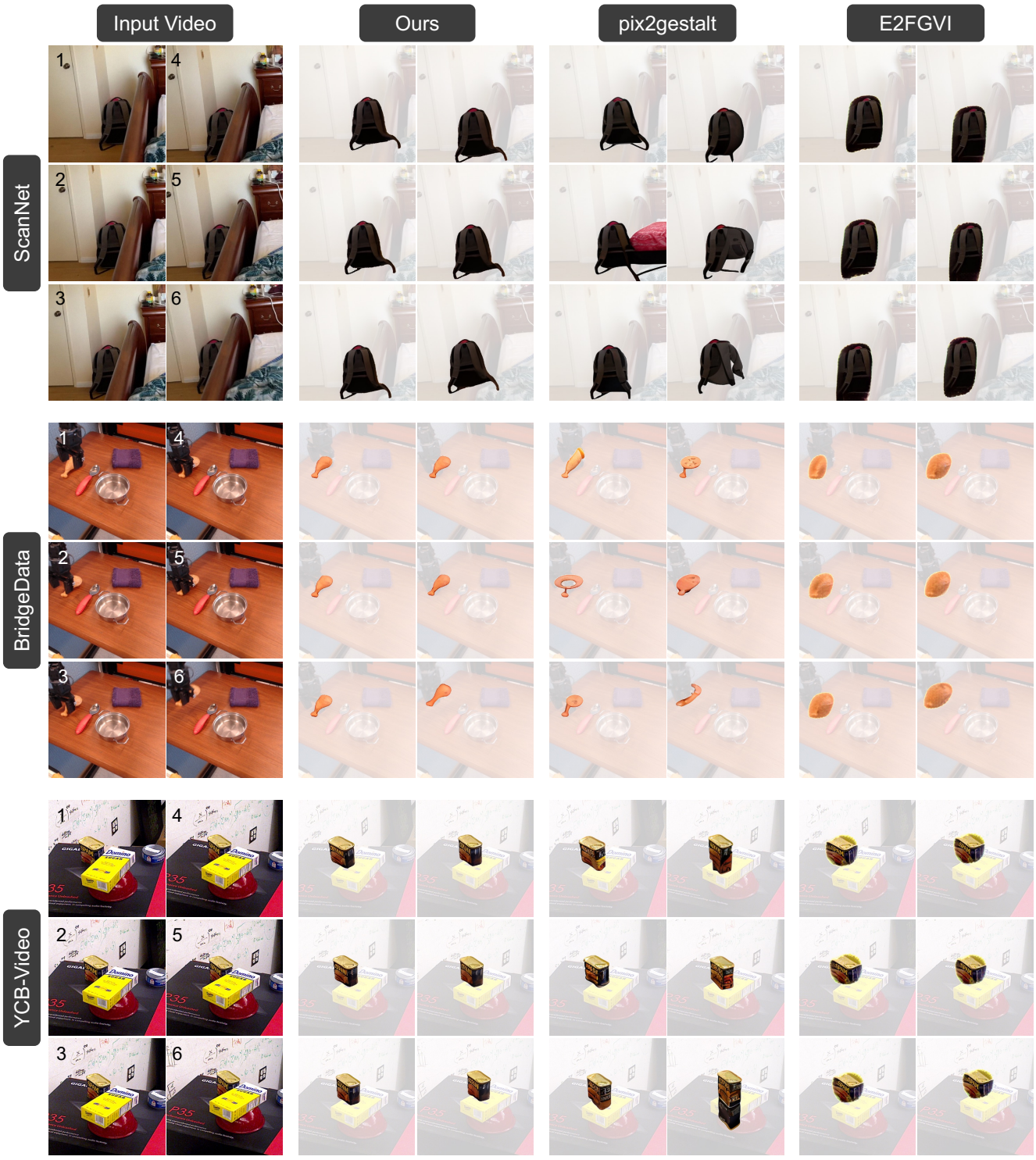

Figure S.13. **Qualitative comparison on ScanNet [3, 20], BridgeData [7, 17], and YCB-Video [18].**

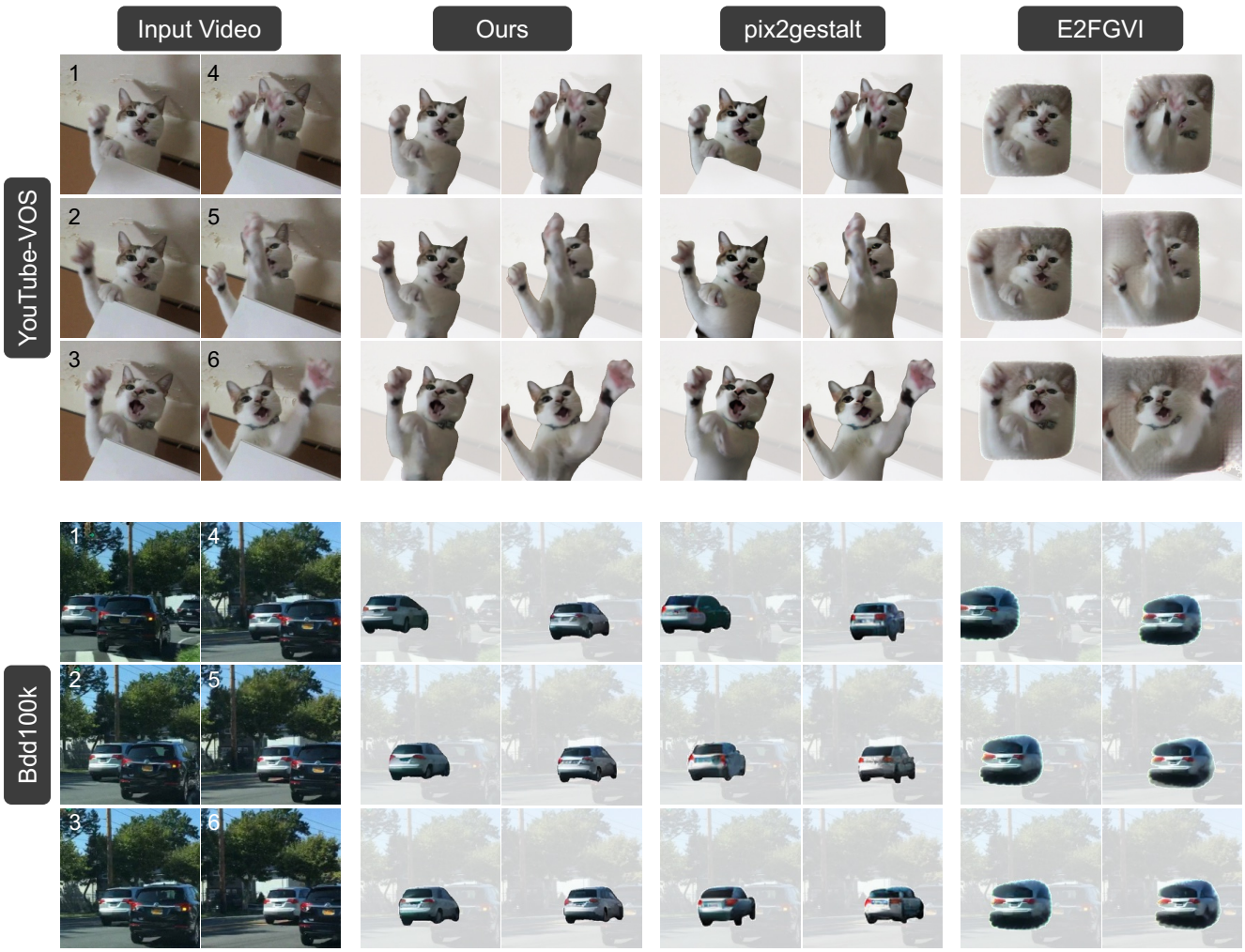

Figure S.14. **Qualitative comparison on YouTube-VOS [19] and Bdd100k [21].**

Figure S.15. **Qualitative results on ScanNet [3, 20].**

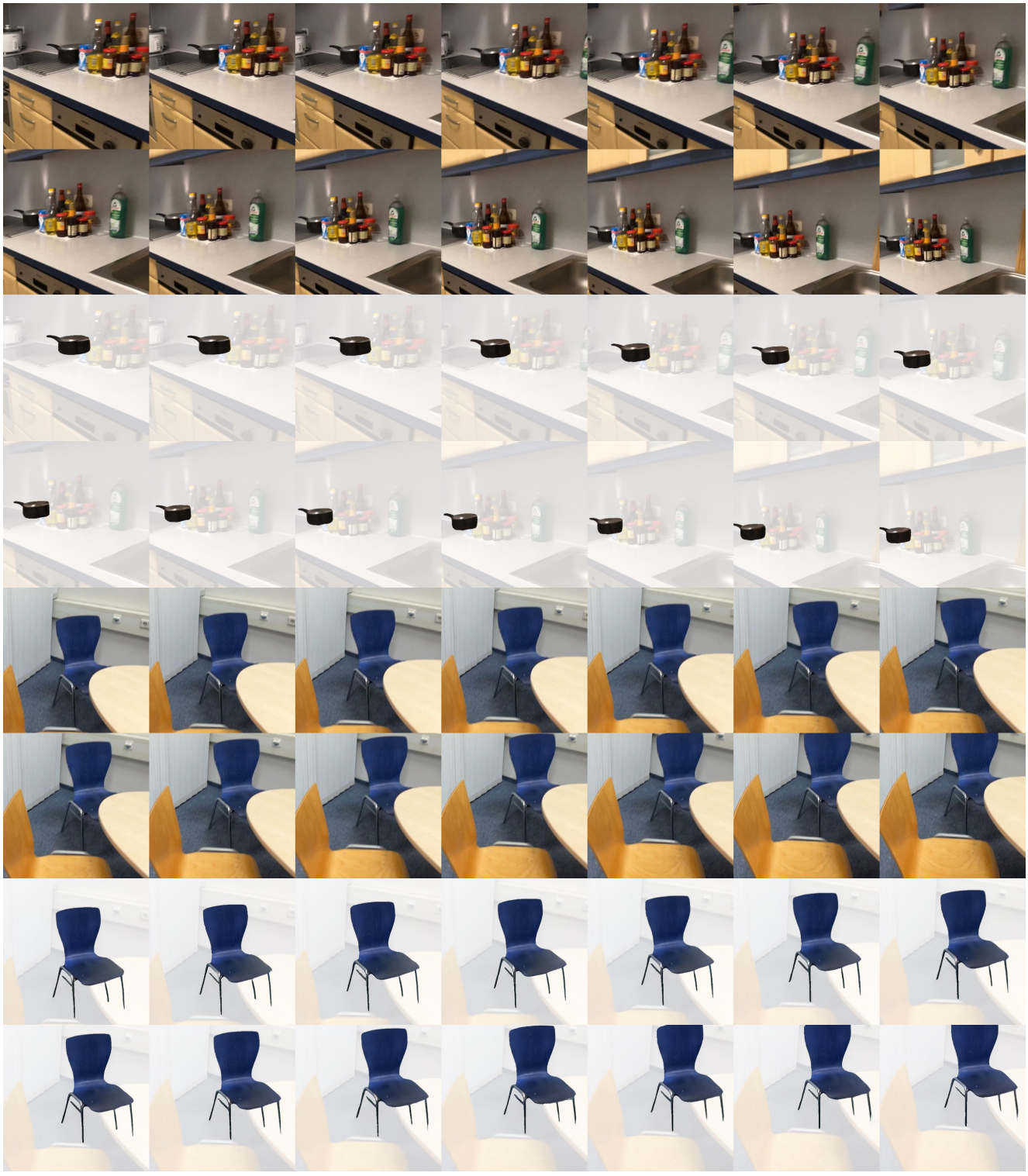

Figure S.16. **Qualitative results on ScanNet [3, 20].**

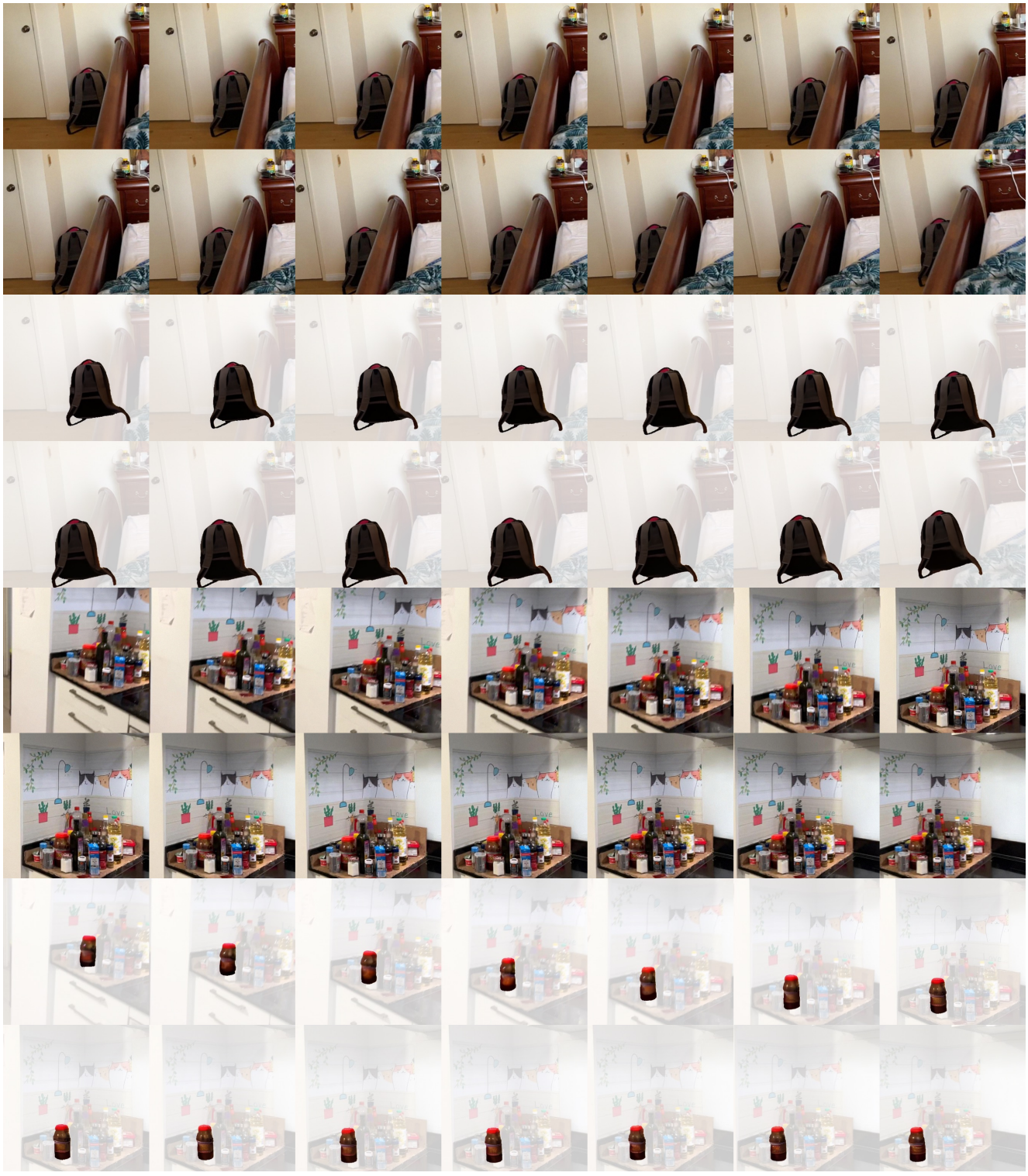

Figure S.17. **Qualitative results on ScanNet [3, 20].**

Figure S.18. **Qualitative results on BridgeData [7, 17].**

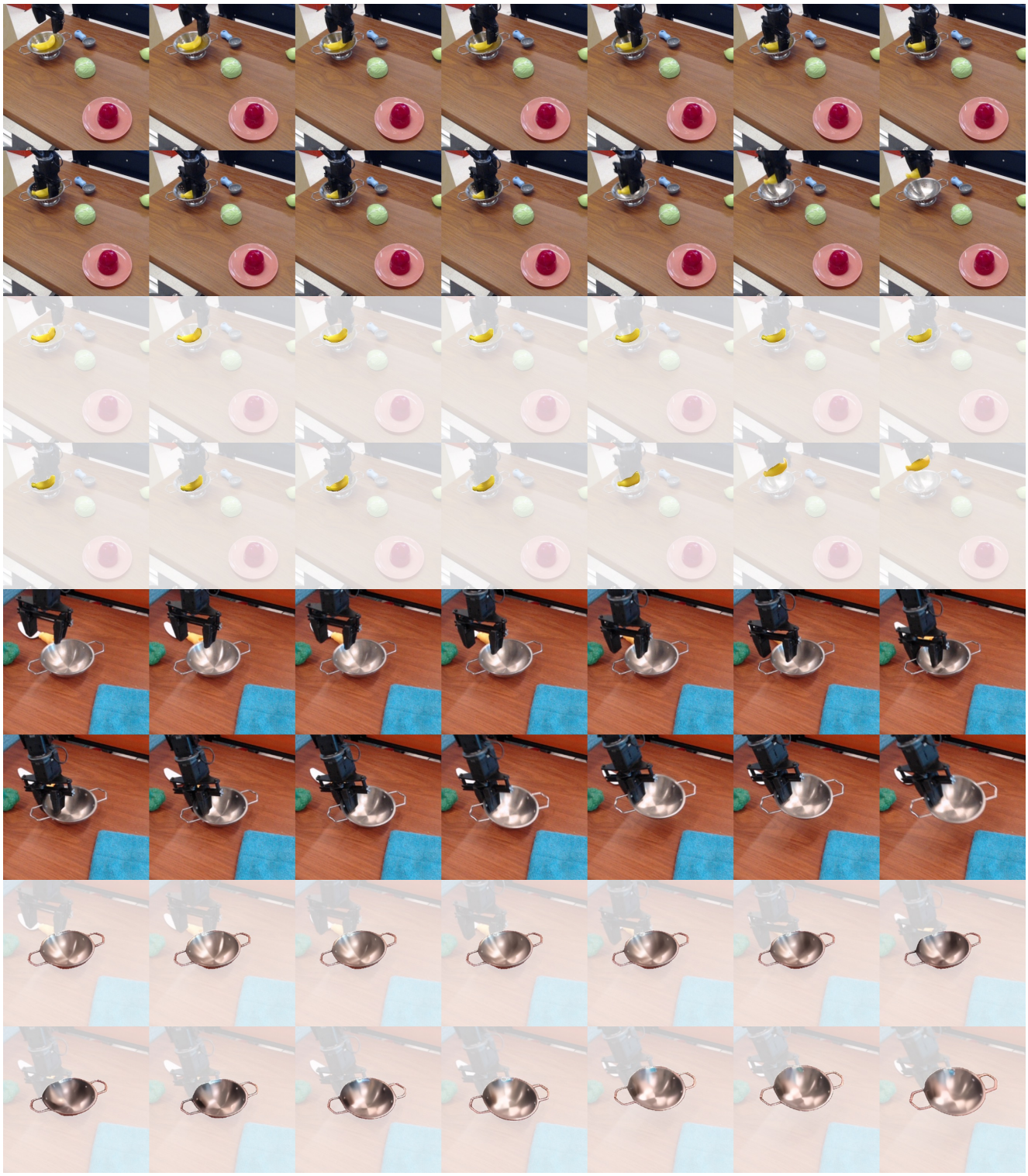

Figure S.19. **Qualitative results on BridgeData [7, 17].**

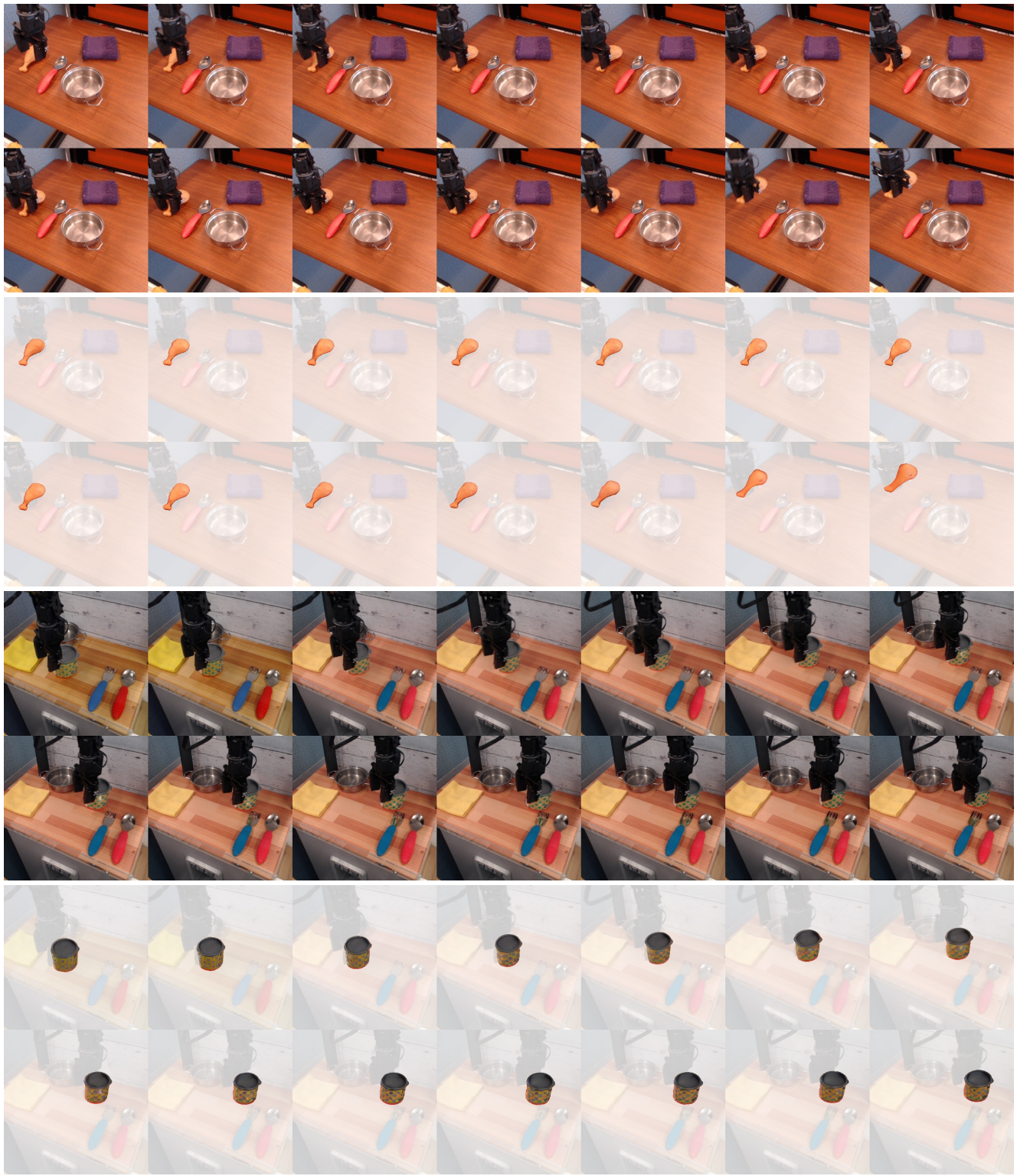

Figure S.20. **Qualitative results on BridgeData [7, 17].**

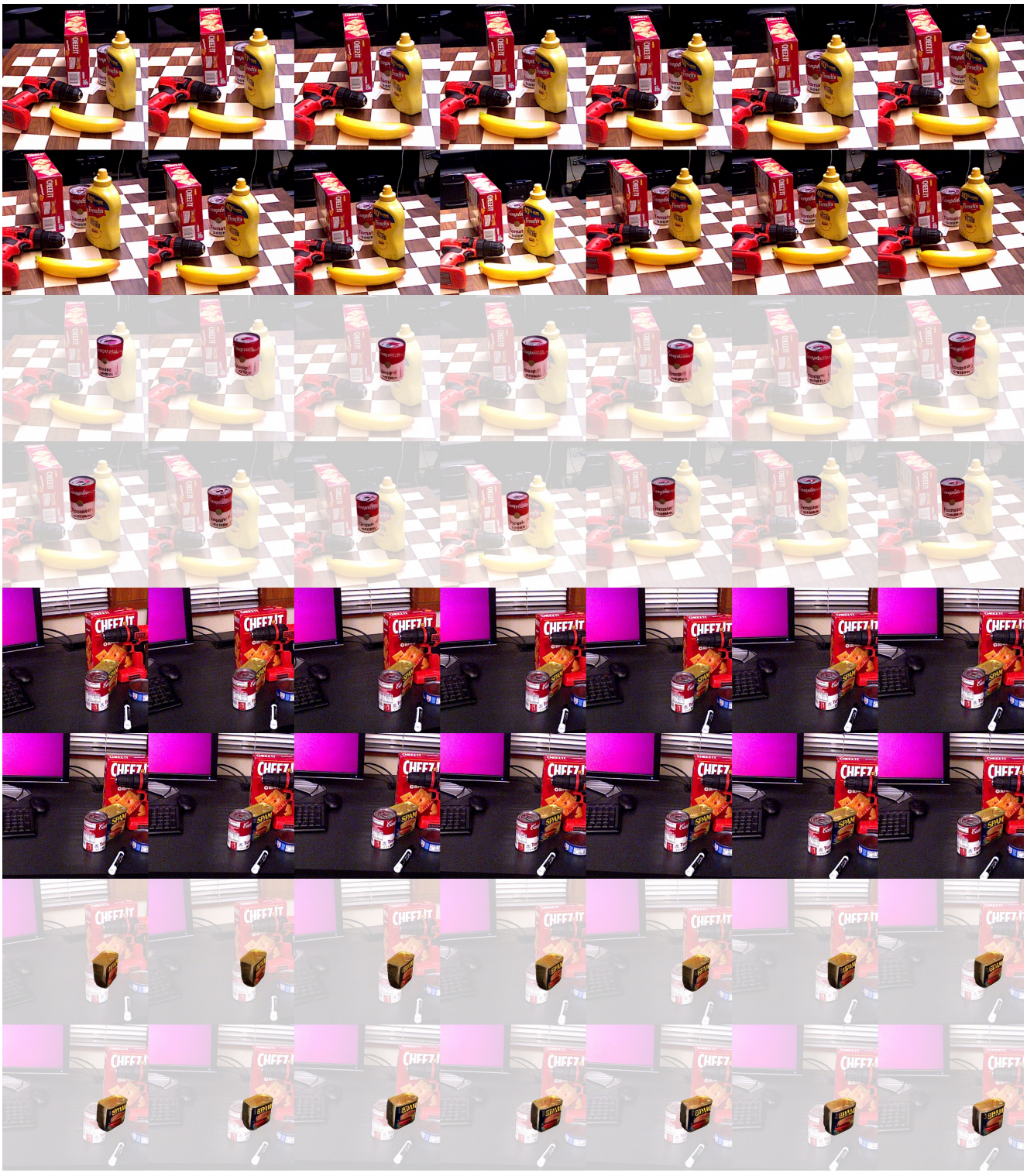

Figure S.21. **Qualitative results on YCB-Video [18].**

Figure S.22. **Qualitative results on YCB-Video [18].**

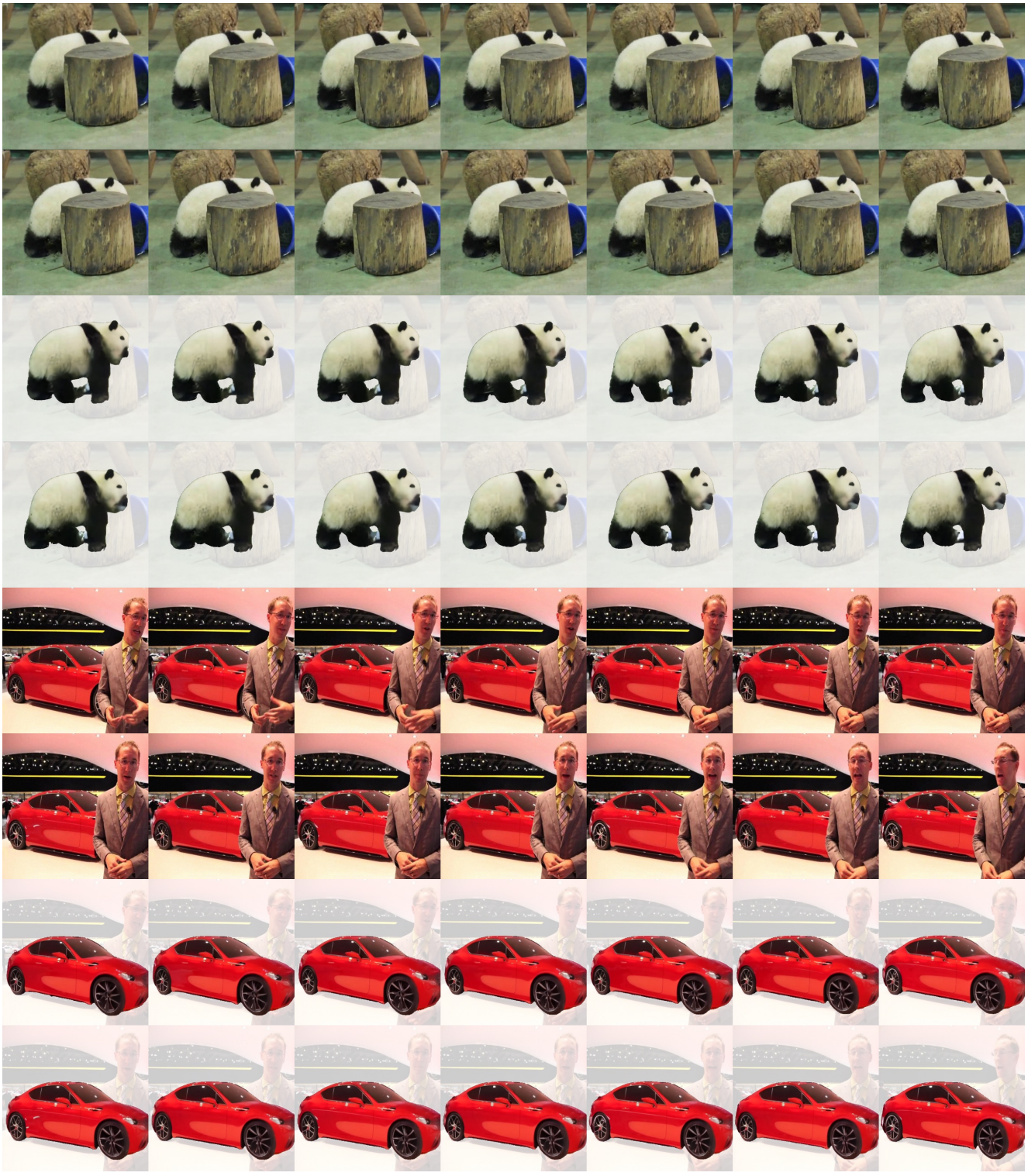

Figure S.23. **Qualitative results on YouTube-VOS [19].**

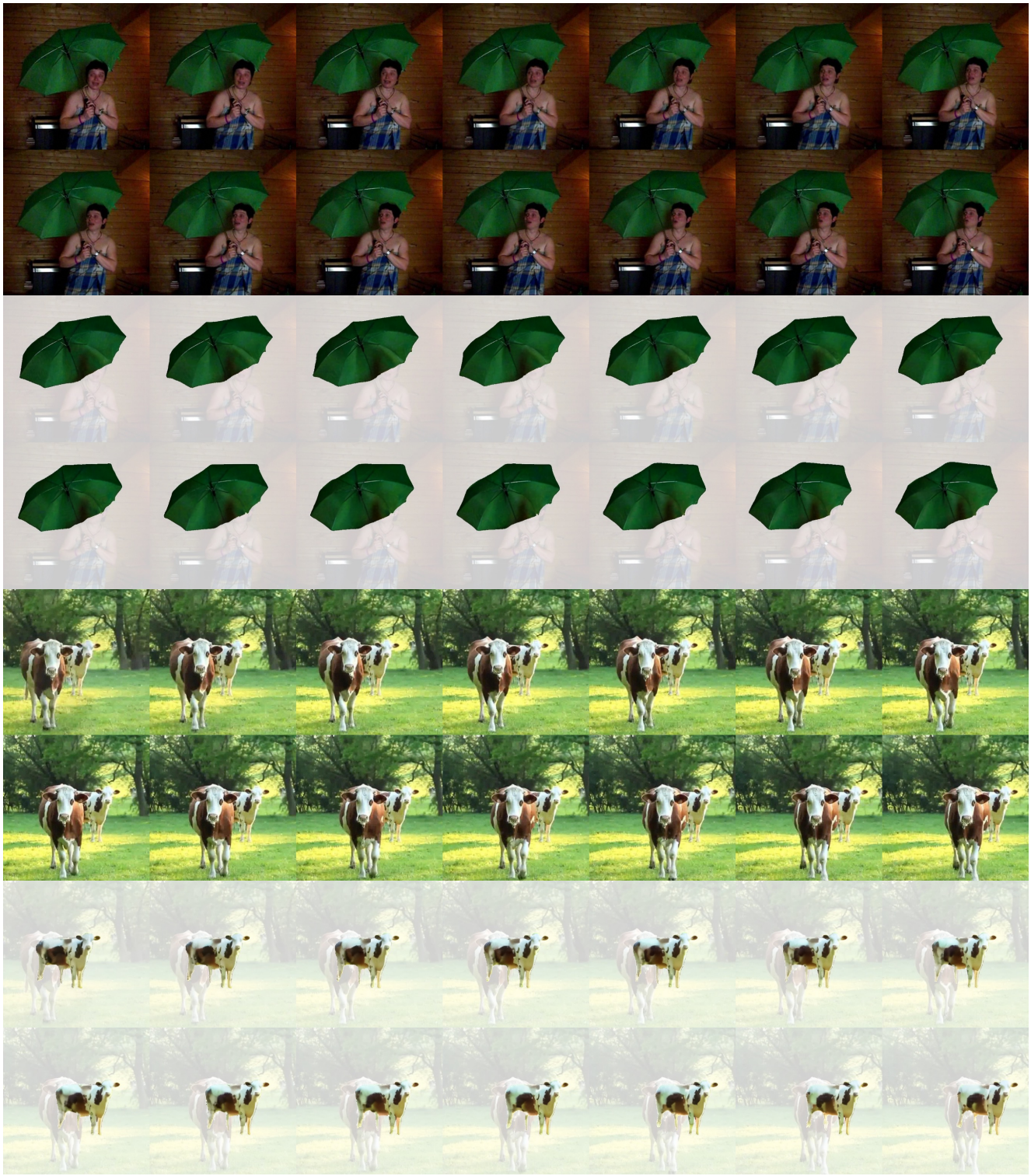

Figure S.24. **Qualitative results on YouTube-VOS [19].**