Supplemental Document: Fine-Grained Spatially Varying Material Selection in Images

JULIA GUERRERO-VIU, Universidad de Zaragoza - I3A, Spain and Adobe Research, United Kingdom
MICHAEL FISCHER, Adobe Research, United Kingdom
ILIYAN GEORGIEV, Adobe Research, United Kingdom
ELENA GARCES, Adobe Research, France
DIEGO GUTIERREZ, Universidad de Zaragoza - I3A, Spain
BELEN MASIA, Universidad de Zaragoza - I3A, Spain
VALENTIN DESCHAINTRE, Adobe Research, United Kingdom

This supplemental document includes additional information on the following topics:

- (S1) Implementation Details
- (S2) Datasets
- (S3) Additional Results

S1 IMPLEMENTATION DETAILS

S1.1 Full Implementation Details

We train our model on our DuMaS dataset (800,000+ images) for 10 epochs using the Adam optimizer, with a learning rate of 1e-4 on 4 A100-40GB GPUs, using the DDPS distributed strategy and a batch size of 4 images per GPU. During training, we sample random crops, and apply random exposure, saturation, and brightness augmentations. From the image encoder, we extract four intermediate features (both local and global tokens) from transformer blocks at indices 2, 5, 8, and 11. For our multi-resolution module, we use different aggregation resolutions per ViT block. After that, the spatial processing further upscales the aggregated features at each block. As mentioned in the main paper, our model uses two resolution levels ($n = 2$), and the details of the aggregation resolutions are as follows:

- I_1 input has resolution $r \times r$, r being the input resolution of the ViT ($r = 518$ for DINOv2). Given the patch size of DINOv2, the output features of the image encoder $F_{1,j}$ have resolution $r/14$ at all four transformer blocks.
- I_2 input has resolution $2r \times 2r$. After rearranging the 4 tiles in which the image was split, the output features of the image encoder $F_{2,j}$ have resolution $r/7$ at all four transformer blocks.
- For blocks $j = 1$ and $j = 2$, we aggregate features at the highest resolution ($r/7$), then the spatial processing module upscales the aggregated features $F_{agg,1}$ and $F_{agg,2}$ to resolution $r/2$.
- For block $j = 3$, we aggregate features at resolution $r/14$, while the spatial processing module upscales the aggregated features $F_{agg,3}$ to resolution $r/7$.
- For block $j = 4$, we aggregate features at resolution $r/14$, while the spatial processing module keeps the resolution of the aggregated features $F_{agg,4}$ at $r/14$.

We use area downsampling and bilinear upsampling for aggregation, while the spatial processing includes a fully-connected layer to aggregate the global token, followed by a 2D convolution and bilinear upsampling, as mentioned in the main paper. For our multiple query sampling, we uniformly sample 10 random pixels within the crop during training.

Training time is approximately 6 days in the aforementioned setup, while inference time is 150ms per image in a single A100-40GB GPU, enabling quick user interaction with the model.

S1.2 Other Image Encoders

As mentioned in the main paper, we rely on DINOv2 as the backbone ViT of our model because we find that it yields higher quality features for the material selection task. Nevertheless, our architecture (including our multi-resolution processing) can be easily adapted to the chosen ViT and we experiment with alternative options, namely DINO [Caron et al. 2021] and Hiera [Ryali et al. 2023], described in this section.

DINO [Caron et al. 2021] is a ViT aiming to maximize the similarity between image embeddings under different augmentations, leveraging knowledge distillation techniques. Its general representations have shown high robustness for many downstream tasks, such as image segmentation, even without task-specific fine-tuning. Internally, it splits the image into non-overlapping square patches of size $ps \times ps$, sequentially projected with transformer blocks. This produces local tokens of size $H/p_s \times W/p_s \times d$, along with an additional global token of size d, where H and W denote the original image resolution, and d represents the feature dimension. We use DINO's pre-trained embeddings with $ps = 8$ and $d = 768$. As with DINOv2, we follow prior work [Sharma et al. 2023], and extract four intermediate features from attention blocks at indices (2, 5, 8, and 11). Consequently, the output of our image encoder with DINO, for the single resolution configuration, consists of four tensors, each at 1/8 of the original image resolution. For multi-resolution aggregation, we aggregate at 1/4 resolution for blocks 1 and 2, and at 1/8 for blocks 3 and 4, respectively. Despite its lower patch size that leads to higher resolution than DINOv2 features, we observe in our experiments that DINO features are less discriminative, particularly

Authors' addresses: Julia Guerrero-Viu, Universidad de Zaragoza - I3A, Spain and Adobe Research, United Kingdom, juliagviu@unizar.es; Michael Fischer, Adobe Research, United Kingdom, mifischer@adobe.com; Iliyan Georgiev, Adobe Research, United Kingdom, igeorgiev@adobe.com; Elena Garces, Adobe Research, France, elenag@adobe.com; Diego Gutierrez, Universidad de Zaragoza - I3A, Spain, diegog@unizar.es; Belen Masia, Universidad de Zaragoza - I3A, Spain, bmasia@unizar.es; Valentin Deschaintre, Adobe Research, United Kingdom, deschain@adobe.com.

for disentangling appearance from color variations, as shown in Figure 14.

Hiera [Ryali et al. 2023] image encoder follows a different architecture by employing a hierarchical patching strategy. It is a simplified version of earlier hierarchical ViTs [Dong et al. 2022; Li et al. 2022] that achieves higher accuracy and faster performance through implementation optimizations and self-supervised training for the masked autoencoder (MAE) pretext task. Its hierarchical design allows to retain spatial information and capture global context without keeping an explicit global token, while supporting a smaller patch size ($ps = 4$). As a result, Hiera's intermediate embeddings span multiple resolutions and produce sharper feature boundaries that improve accuracy in dense downstream tasks. Following previous work, we use the Hiera-L variant initialized from the MAE pre-trained model and finetuned for object selection [Ravi et al. 2024], using windowed absolute positional embeddings [Bolya et al. 2023]. Unlike for DINO-based encoders, we observe better performance when unfreezing the Hiera encoder while training our material selection pipeline, so we report results with this configuration. The output of our image encoder with Hiera, for the single resolution configuration, consists of four tensors with resolutions $\frac{1}{4}$, $\frac{1}{8}$, $\frac{1}{16}$, and $\frac{1}{32}$ of the original image one, and a feature dimension of $d = 256$. For multi-resolution aggregation, we aggregate at the maximum resolution per block. In our experiments, despite its higher feature resolution which produces sharper prediction edges, Hiera encoder leads to reduced performance in material selection compared to DINOv2 (see Table 3 in the main paper, and qualitative results in Figure 14).

S1.3 SAM2 Fine-Tuning

We fine-tune the state-of-the-art model in object selection SAM2 [Ravi et al. 2024] for the material selection task, using our DuMaS dataset. For all our experiments, we use the large configuration of the model, which includes the Hiera-L [Ryali et al. 2023] image encoder. To fine-tune the model, we encode our DuMaS dataset as MP4 videos with 1024×1024 resolution and the annotations in CoCoRLE encoding for efficient storage, as in their SA-V dataset [Ravi et al. 2024]. We cut our 1-minute videos at 30fps into short clips of 40 frames, and then sub-sample them by skipping every 6 frames, to increase the intra-frame distance. We follow previous work [Fischer et al. 2024] and fine-tune all modules of the model except the image encoder that we keep frozen. We include their negative sampling during training, by sampling a positive click with 80% probability, and a negative click on a random other material with 20% and reverse the temporal order of the frame sequence with a chance of 50%, avoiding sampling clicks on the edge of the material masks. We use the AdamW optimizer with weight decay 0.01 and learning rate 1e-5 until convergence, which takes 1,200K steps. Due to memory constraints, we use a batch size of 1 with aggregated gradients every 4 steps. Fine-tuning takes approximately 10 days with the aforementioned setup, while inference time is 180ms per image in a single A100-40GB GPU.

S2 DATASETS

S2.1 Training Dataset: DuMaS

We show representative examples from our large-scale Dual-level Material Selection dataset (DuMaS) in Figure 1. Please refer to the main paper for more information about the creation of this dataset.

S2.2 Test Datasets

S2.2.1 Two-Level Test. We show all images from our Two-Level Test dataset and their corresponding (sub)texture annotations in Figure 2. The background and unannotated regions share the ID zero, which denotes un-clickable regions. For the few images that do not have two-level annotations (e.g., the meeting room image), the subtexture and texture annotations are identical. All 20 images were manually annotated using LabelMe and come from royalty-free sources like Pixabay and Unsplash. We will release the dataset and annotations upon publication.

S2.2.2 Challenging Subset. Our Challenging Subset includes 30 test cases (image and input click) that showcase difficult scenarios from our test datasets (10 from Materialistic Test dataset and 20 from Two-Level Test dataset). We show all test cases in Figure 3, each classified among the following scenarios:

- Fine structures: The input click targets a fine structure (e.g., a basketball net), or the selection mask includes thin details at the texture or subtexture level (e.g., lettering on the Christmas sack).
- Albedo entanglement: The scene contains different materials with similar albedo to that of the input click, which need to be distinguished.
- Strong light variations: There are areas of the scene with the same material as the input click but very different appearance due to strong shadows or lighting reflections.

S3 ADDITIONAL RESULTS

In this section, we include additional results to those reported in the main paper, specifically: quantitative results on synthetic data (S3.1), qualitative results of our material selection method (S3.2), qualitative comparisons to previous works trained on our DuMaS dataset (S3.3), quantitative assessment of the frame duplication trick for SAM2 (S3.4), additional robustness metrics and qualitative comparisons (S3.5), qualitative ablations (S3.6), and illustrative comparisons to SAM2's interactive setting (S3.7).

S3.1 Quantitative Evaluation on Synthetic Data

In the main paper, we extensively evaluate our method on real-world in-the-wild photographs (see Sec.4.1.). We include in Table 1 additional evaluation of our method and all the baselines in *synthetic* scenes from our DuMaS dataset. In particular, we use 4,325 synthetic images from the six test scenes not seen during training, which we will publicly release. Our method significantly outperforms all the baselines, including those trained with our DuMaS dataset, and Materialistic using DINOv2 as a backbone.

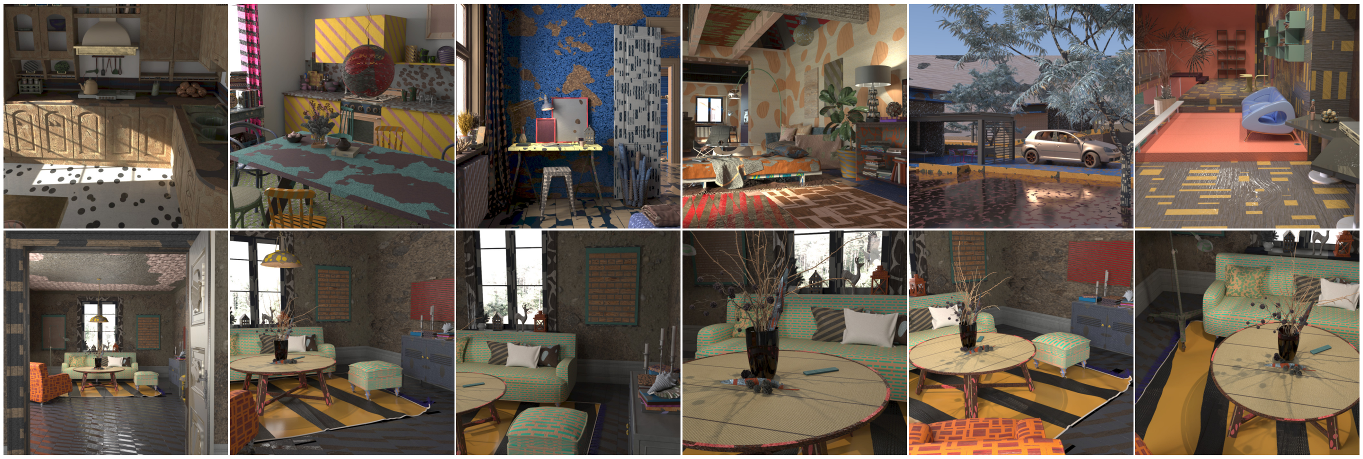

Fig. 1. Our DuMaS training dataset. First row shows representative examples from the scenes in our synthetic DuMaS training dataset, including spatially varying materials with annotations at two levels. As our dataset includes videos, second row shows examples of different frames along a single video of the same scene.

Table 1. Mean results of various methods (rows) across different metrics (subcolumns) on the synthetic DuMaS test dataset. For single-level methods trained on our DuMaS dataset (+DuMaS), we train two separate models (one per level) and report the result of the relevant one per column. Gray text indicates cases where the model is evaluated on a different task to the one it is trained for, and boldface marks the best results.

	DuMaS Test dataset					
	Subtexture Level			**Texture Level**		
	L1 ↓	**IoU ↑**	**F1 ↑**	**L1 ↓**	**IoU ↑**	**F1 ↑**
Materialistic	0.135	0.481	0.587	0.130	0.556	0.649
Materialistic + DINOv2	0.161	0.449	0.561	0.133	0.584	0.674
Materialistic + DuMaS	0.096	0.524	0.610	0.100	0.612	0.695
SAM2 *obj. selection	0.106	0.354	0.455	0.132	0.387	0.476
SAM2 + DuMaS	0.104	0.490	0.597	0.091	0.615	0.703
Ours	**0.060**	**0.626**	**0.703**	**0.062**	**0.707**	**0.776**

S3.2 Additional Qualitative Results

We show additional qualitative results of our method in Figure 4, for real-world images from our test datasets. Results demonstrate the accuracy and confidence of our method at both texture and subtexture selection levels, even in challenging scenarios. Please refer to the main paper for more results.

Despite improvement over previous work, we acknowledge failure cases for a few very challenging examples included in our test datasets, like the tiger or the fishes in Figure 8.

We conduct an additional qualitative analysis regarding sensitivity of our method to roughness differences. As shown in Figure 7, our method can successfully differentiate materials with the same albedo and different roughness (e.g., Christmas balls, left). We observe that if the roughness differences occur within a single object and are too subtle (e.g., climbing holds, right), our method may not differentiate them. This could be addressed by explicitly including these materials in the training data.

Table 2. Mean results of the SAM2 model fine-tuned on our DuMaS dataset (rows) across different metrics (subcolumns) on our two real-world test datasets (columns). Results show the benefit of using the frame duplication trick (FDT) that we apply as default for inference (see text for details).

	Materialistic Test			**Two-Level Test**					
	Texture Level			**Subtexture Level**			**Texture Level**		
	L1 ↓	**IoU ↑**	**F1 ↑**	**L1 ↓**	**IoU ↑**	**F1 ↑**	**L1 ↓**	**IoU ↑**	**F1 ↑**
SAM2+DuMaS w/o FDT	0.085	0.761	0.831	0.138	0.525	0.642	0.096	0.695	0.778
SAM2+DuMaS	**0.060**	**0.784**	**0.847**	**0.103**	**0.576**	**0.681**	**0.071**	**0.730**	**0.799**

S3.3 Additional Qualitative Comparisons

We show additional qualitative comparisons to previous works fine-tuned with our DuMaS dataset in Figures 5 (subtexture level) and 6 (texture level). As shown in Table 1 of the main paper, our method outperforms both Materialistic [Sharma et al. 2023] and SAM2 [Ravi et al. 2024] methods, also when trained on our data, both in terms of accuracy of the final selection (binary prediction after thresholding) and confidence of the predictions.

S3.4 SAM2 Frame Duplication Trick

Following previous work [Fischer et al. 2024], for SAM2 model evaluations, we duplicate the first frame and use the output of the second frame in inference. We show in Table 2 that this trick significantly improves its overall confidence and accuracy, by forcing the network to use its memory module, fine-tuned for the material selection task. We always report results using this frame duplication trick (FDT) in the main paper and supplemental document, unless stated otherwise.

S3.5 Additional Robustness Results

We include the robustness evaluation in terms of prediction confidence in Table 3.

For each method, we report the average mean and standard deviation of the IoU using 1,000 different thresholds in the range

[0.001, 0.999]. As it can be seen, our method is more robust to the selected threshold than previous work (upper part of Table 3), with higher mean and lower std, demonstrating higher accuracy and lower variability across thresholds. Being less sensitive to the selection threshold eliminates the need for users to manually tune the threshold each time, making the process more automatic and user-friendly. SAM2 fine-tuned on our DuMaS dataset exhibits lower sensitivity to the threshold than Materialistic, being more robust despite having lower overall accuracy, as shown in Table 1 of the main paper.

Regarding ablations of our method (lower part of Table 3), we observe higher sensitivity to the threshold for DINO and Hiera variants, mainly due to their less confident predictions overall. Interestingly, our multiple query sampling module shows highly beneficial to improve robustness to the threshold, achieving 1.3× lower variability across thresholds than the variant without including it (w/o Multi-Samp.). Additionally, we include more qualitative examples of robustness for pixel consistency, zoom consistency, and illumination consistency in Figures 9, 10 and 12, respectively. For every example, we also include the comparison to previous works Materialistic and SAM2 + DuMaS (see main paper for further details).

Finally, we evaluate consistency across frames, including an example in Figure 11. We implement this as cross-image selection, selecting a query pixel in the first frame, and using its embedding as a query across the next frames of the video. Despite not being explicitly trained for it, our method shows reasonable consistency in real-world videos, effectively grouping the cushions made of the same material (even the ones not originally visible in the first frame).

S3.6 Additional Qualitative Ablations

We include additional qualitative ablations in Figure 13, illustrating the main improvements due to our multi-resolution and multi-sampling modules in challenging scenarios with fine structures and albedo entanglement (see also Figure 10 of the main paper).

Figure 14 shows qualitative results of our ablations using different image encoders, namely DINO [Caron et al. 2021] and Hiera [Ryali et al. 2023], for images in our test dataset, at texture level. We observe lower accuracy and confidence in the predictions when using these alternative ViTs compared to DINOv2, especially in examples when it is more challenging to disentangle appearance from color (rows 1 and 2).

S3.7 Comparison to SAM2 Interactive Setting

Our material selection model is highly effective and accurate to select pixels sharing the same texture and subtexture components within an image, using a single user click. In contrast, SAM2's interactive setting requires the user to provide several positive and negative clicks to refine their selection. While this adds certain flexibility to the selection modality, it might also be very tedious and time consuming for selecting materials at texture and subtexture levels, requiring prior knowledge of areas sharing the same material, and several clicks, as illustrated in Figure 15.

Table 3. Results of sensitivity to the selection threshold across different methods (rows) on our two real-world test datasets (columns). We show average mean and std IoU using 1000 different thresholds in the range [0.001, 0.999] (higher mean and lower std is better).

	Materialistic Test	**Two-Level Test**	
	Texture Level	**Subtexture Level**	**Texture Level**
	Sens.TH IoU	**Sens.TH IoU**	**Sens.TH IoU**
Materialistic	0.860 ± 0.122	0.451 ± 0.107	0.572 ± 0.151
Materialistic + DuMaS	0.880 ± 0.073	0.516 ± 0.147	0.649 ± 0.113
SAM2 + DuMaS	0.881 ± 0.061	0.591 ± 0.064	0.722 ± 0.054
Ours, DINO	0.875 ± 0.076	0.585 ± 0.127	0.677 ± 0.107
Ours, Hiera	0.859 ± 0.093	0.527 ± 0.129	0.736 ± 0.113
Ours, w/o Multi-Res.	0.898 ± 0.076	0.607 ± 0.131	0.763 ± 0.113
Ours, w/o Multi-Sampl.	0.925 ± 0.061	0.573 ± 0.147	0.705 ± 0.118
Ours, Single Level	0.925 ± 0.046	0.606 ± 0.144	0.756 ± 0.099
Ours Full	**0.930 ± 0.042**	**0.670 ± 0.111**	**0.793 ± 0.089**

S3.8 Multi-selection Combination

One idea to mitigate potential failure cases in practical scenarios could be to combine multiple selection masks. We have evaluated this by, given an initial input pixel, randomly sampling N additional query pixels from the initial similarity score map; we then combine the N+1 resulting similarity score maps by computing the per-pixel median (and, as with all the results in the paper, binarize the resulting similarity to produce the final mask using a threshold of 0.5). We include the results of our experiments (with N = 2, 4 and 10) in Table 4. As it can be seen, this combination tends to slightly improve the original selection in most of the results, although we qualitatively observe it is counter-productive in a few cases. Note that this experiment has been designed to *automatically* evaluate multi-selection and thus extra query pixels are randomly selected, whereas in practice a user would be selecting extra query pixels manually; thus, this experiment is a conservative evaluation of the potential of multi-selection in our method, also discussed in the main paper.

Table 4. Quantitative evaluation of multi-selection combination, with N additional query pixels (N=0 means without multi-selection, our default setting), on our two real-world test datasets.

	Materialistic Test			**Two-Level Test**					
	Texture Level			**Subtexture Level**			**Texture Level**		
	L1 ↓	**IoU ↑**	**F1 ↑**	**L1 ↓**	**IoU ↑**	**F1 ↑**	**L1 ↓**	**IoU ↑**	**F1 ↑**
N=0	0.030	0.896	0.935	0.071	0.673	0.766	0.069	**0.750**	0.823
N=2	**0.027**	0.894	0.933	**0.070**	**0.675**	**0.767**	0.067	0.730	0.823
N=4	0.028	**0.897**	**0.936**	0.075	0.671	0.760	**0.065**	0.732	**0.825**
N=10	**0.027**	0.896	0.935	0.074	0.664	0.753	0.073	0.738	0.811

REFERENCES

Daniel Bolya, Chaitanya Ryali, Judy Hoffman, and Christoph Feichtenhofer. 2023. Window Attention is Bugged: How not to Interpolate Position Embeddings. *arXiv preprint arXiv:2311.05613* (2023).

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. 2021. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*. 9650–9660.

Xiaoyi Dong, Jianmin Bao, Dongdong Chen, Weiming Zhang, Nenghai Yu, Lu Yuan, Dong Chen, and Baining Guo. 2022. Cswin transformer: A general vision transformer backbone with cross-shaped windows. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 12124–12134.

Michael Fischer, Iliyan Georgiev, Thibault Groueix, Vladimir G Kim, Tobias Ritschel, and Valentin Deschaintre. 2024. SAMa: Material-aware 3D Selection and Segmentation. *arXiv preprint arXiv:2411.19322* (2024).

Yanghao Li, Chao-Yuan Wu, Haoqi Fan, Karttikeya Mangalam, Bo Xiong, Jitendra Malik, and Christoph Feichtenhofer. 2022. Mvitv2: Improved multiscale vision transformers for classification and detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 4804–4814.

Lukas Murmann, Michael Gharbi, Miika Aittala, and Fredo Durand. 2019. A Multi-Illumination Dataset of Indoor Object Appearance. In *2019 IEEE International Conference on Computer Vision (ICCV)*.

Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. 2024. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714* (2024).

Chaitanya Ryali, Yuan-Ting Hu, Daniel Bolya, Chen Wei, Haoqi Fan, Po-Yao Huang, Vaibhav Aggarwal, Arkabandhu Chowdhury, Omid Poursaeed, Judy Hoffman, et al. 2023. Hiera: A hierarchical vision transformer without the bells-and-whistles. In *International Conference on Machine Learning*. PMLR, 29441–29454.

Prafull Sharma, Julien Philip, Michaël Gharbi, Bill Freeman, Fredo Durand, and Valentin Deschaintre. 2023. Materialistic: Selecting similar materials in images. *ACM Transactions on Graphics* 42, 4 (2023).

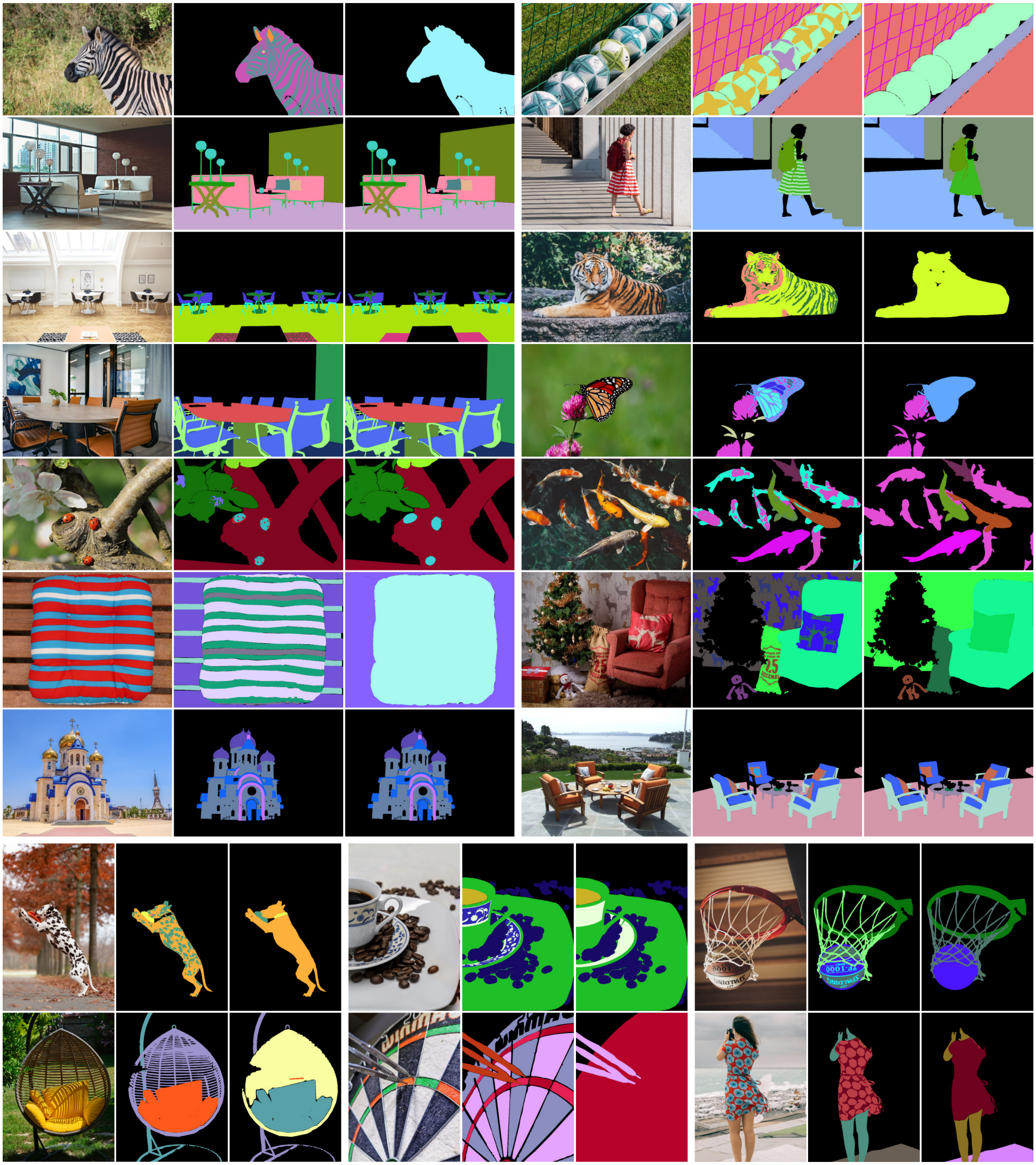

Fig. 2. Our manually annotated Two-Level Test evaluation dataset of 20 real-world images, with subtexture- and texture annotations per image, respectively. For images where subtexture- and texture-level coincide, the annotations are identical. Subtexture- and texture-levels are mapped to random colors here for visibility; background- and unannotated regions have ID zero and are colored black.

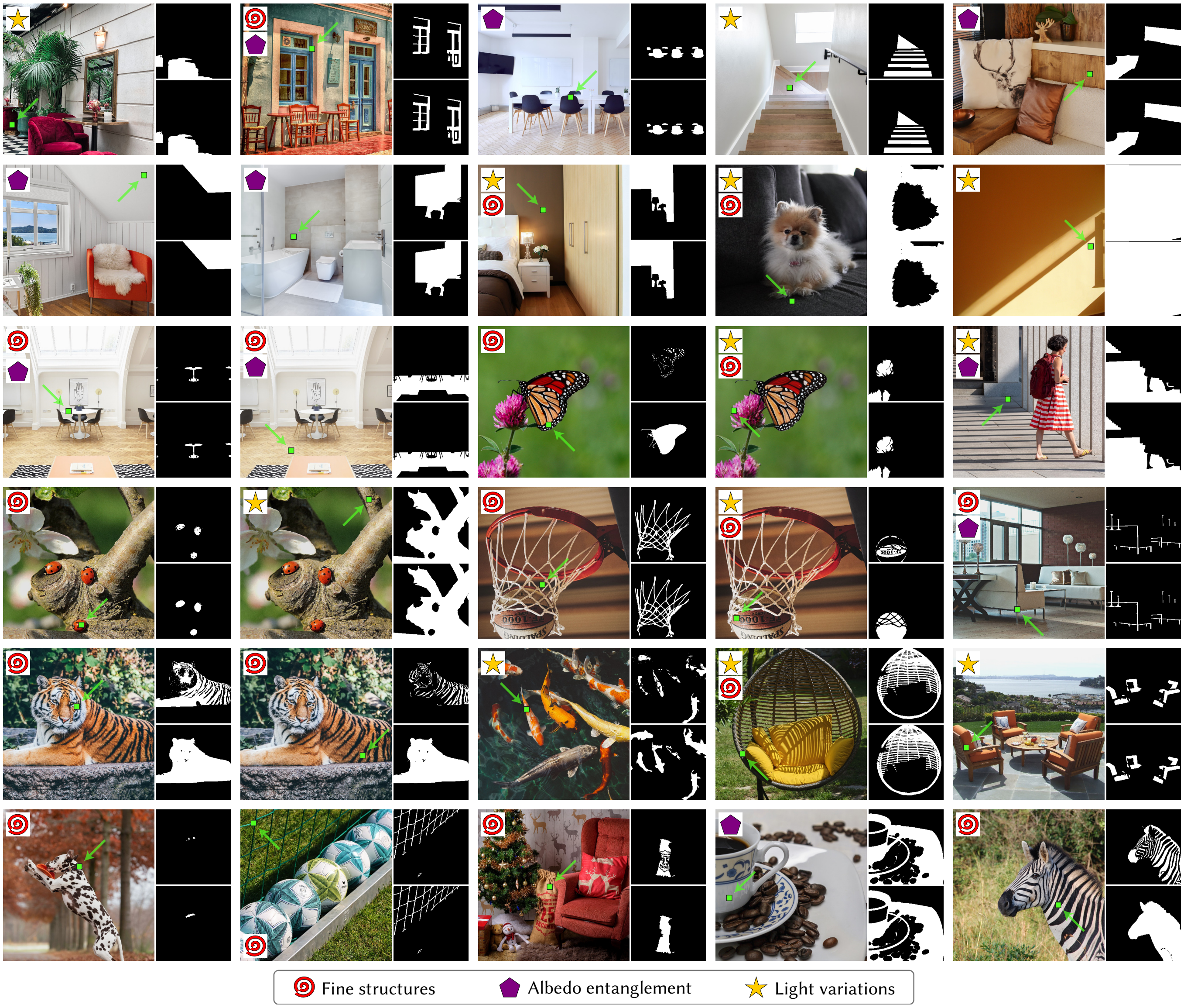

Fig. 3. Our Challenging Subset test dataset contains 30 clicks on challenging material selections, with the first two rows from the Materialistic Test dataset, and the bottom four rows from our Two-level Test dataset. We show the ground-truth selection binary masks for each click in black and white (for better visibility) next to each image, with subtexture and texture in the top and bottom mask, respectively. For the first two rows, subtexture- and texture-annotations are identical, since the images are from the (single-level) Materialistic Test dataset.

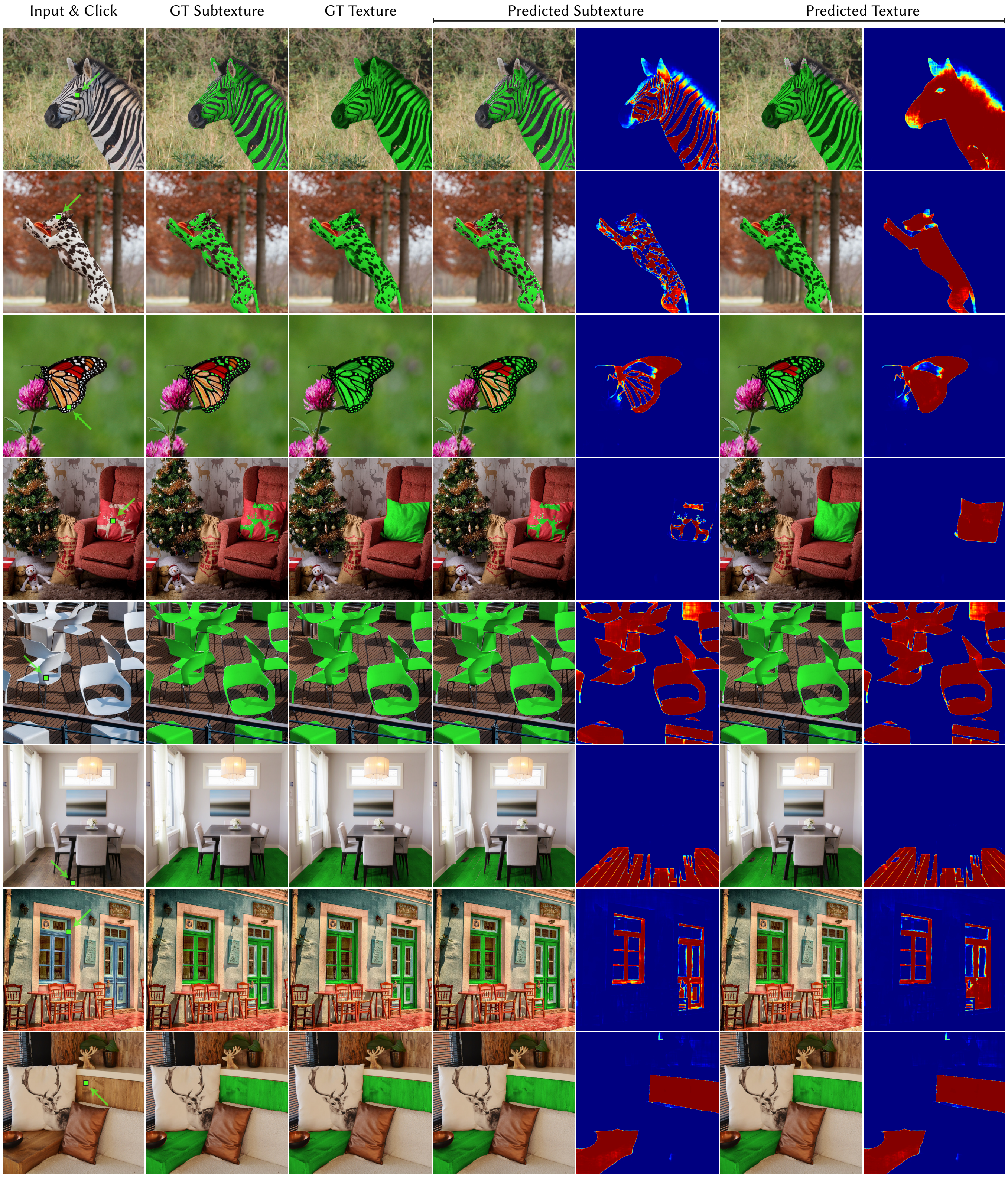

Fig. 4. Additional qualitative results of our method on our Two-Level Test dataset (first four rows) and the Materialistic Test dataset (bottom four rows). Note that, at the texture level, we accurately identify the full animal pattern while, at the subtexture level, we discern different subtextures within the pattern.

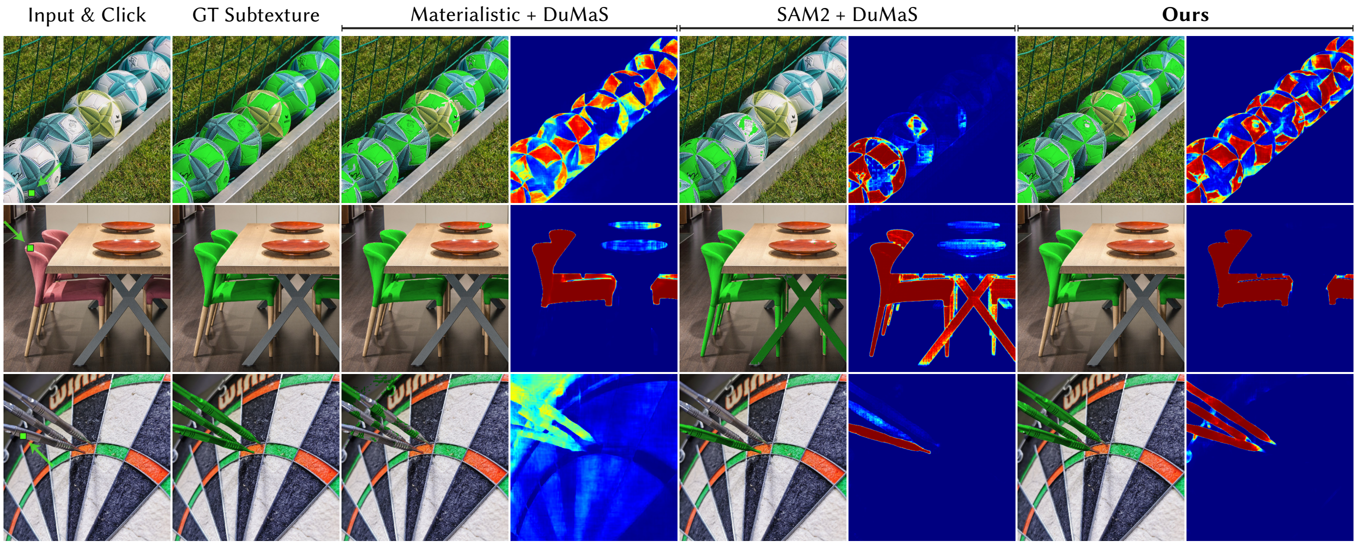

Fig. 5. Additional comparisons between our method, and Materialistic and SAM2 trained on our DuMaS dataset at the subtexture level. Only our method is able to obtain clean subtexture selections.

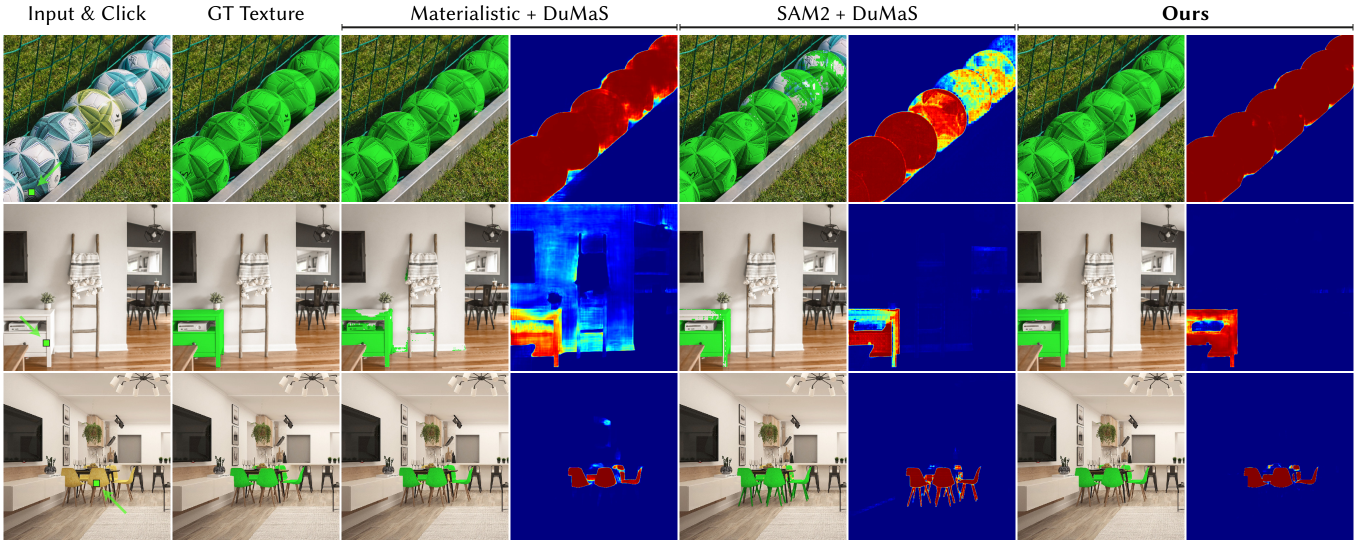

Fig. 6. Additional comparisons between our method, and Materialistic and SAM2 trained on our DuMaS dataset at the texture level. Our results are more accurate and less biased by color.

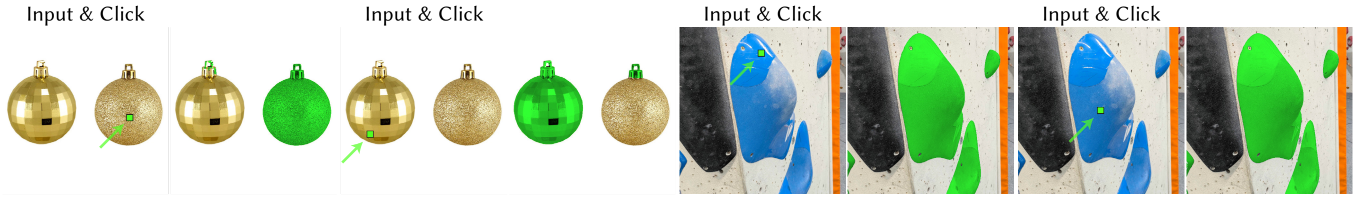

Fig. 7. Our method can differentiate between materials with the same albedo and different roughness (left columns), but sometimes struggles when the differences are too subtle and appear within an object (right columns), presumably since these cases are under-represented in our training dataset.

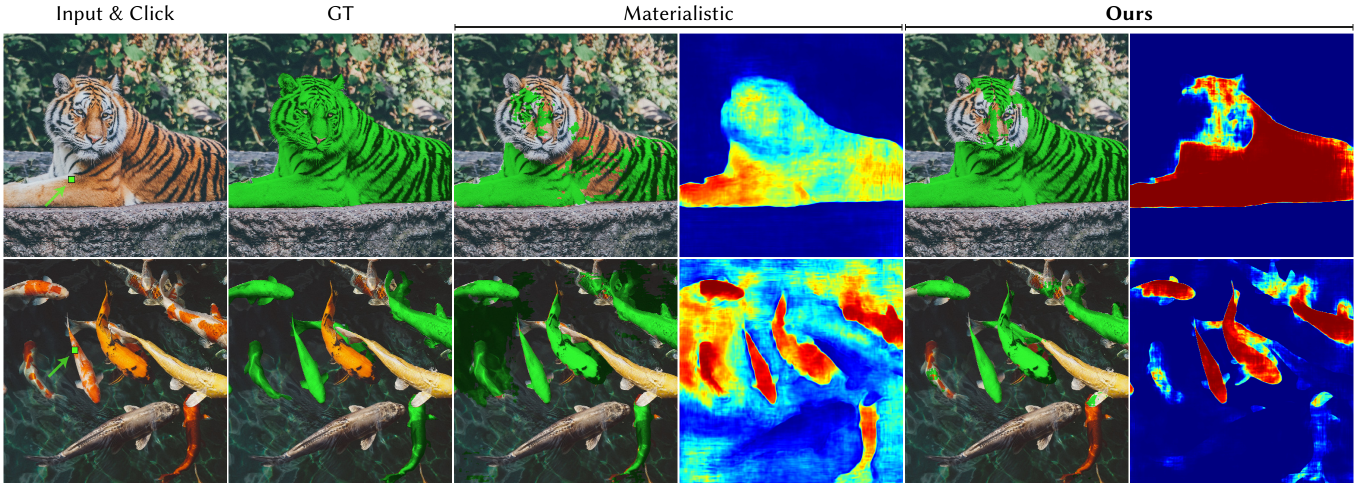

Fig. 8. Challenging examples where our method cannot accurately recover the selected material. Our results show a slight improvement over previous work in terms of confidence, but are inaccurate for selecting the material of the tiger (maybe due to the very high frequency details of the fur) or the fishes (probably affected by the underwater reflections).

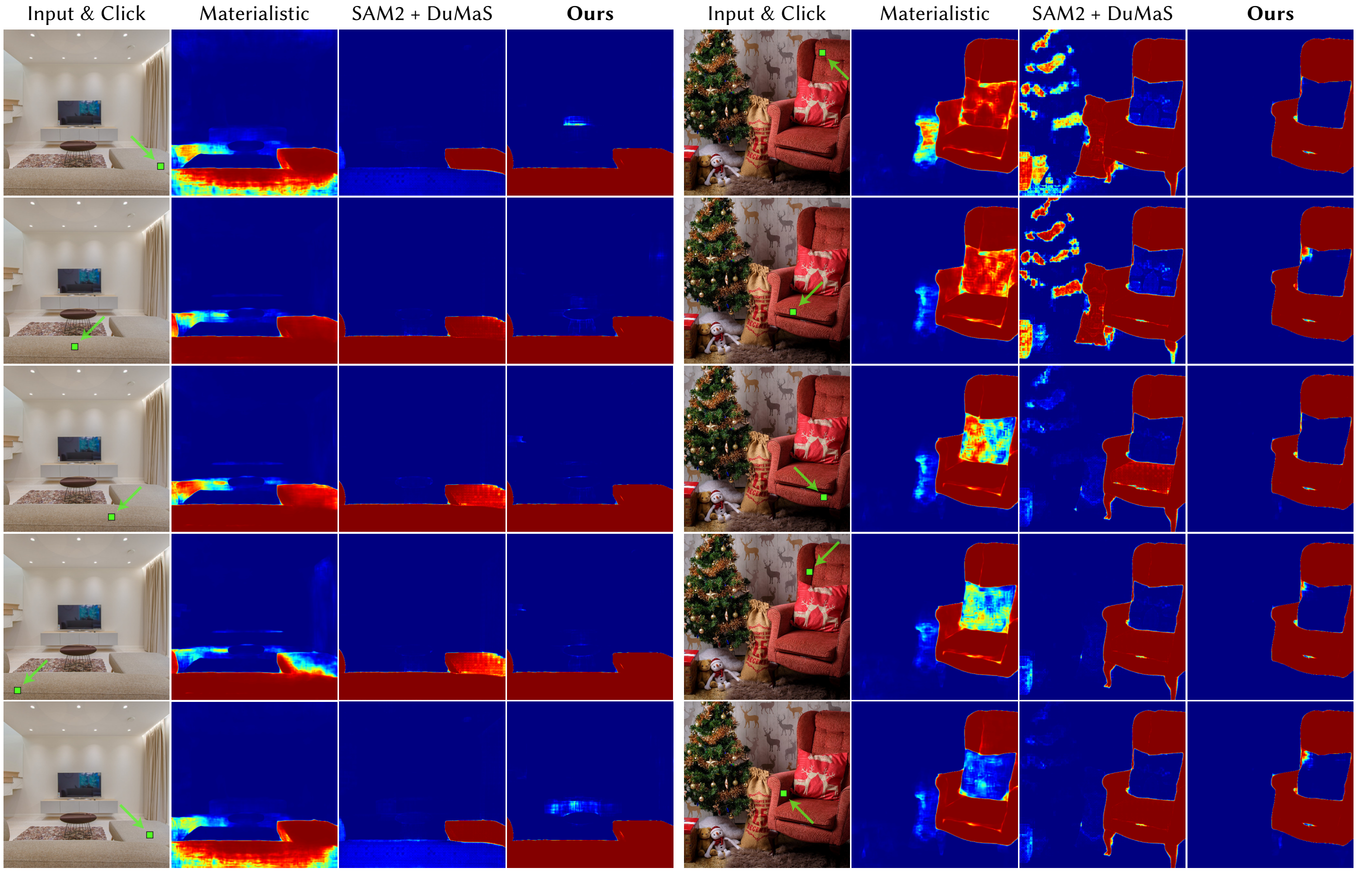

Fig. 9. Pixel consistency. We evaluate the consistency of the selection by clicking on different query pixels of the same material. Our method produces the best overall results –invariant to the input clicks– while Materialistic, and SAM2 trained with our dataset, produce more unpredictable outcomes depending on the selected region.

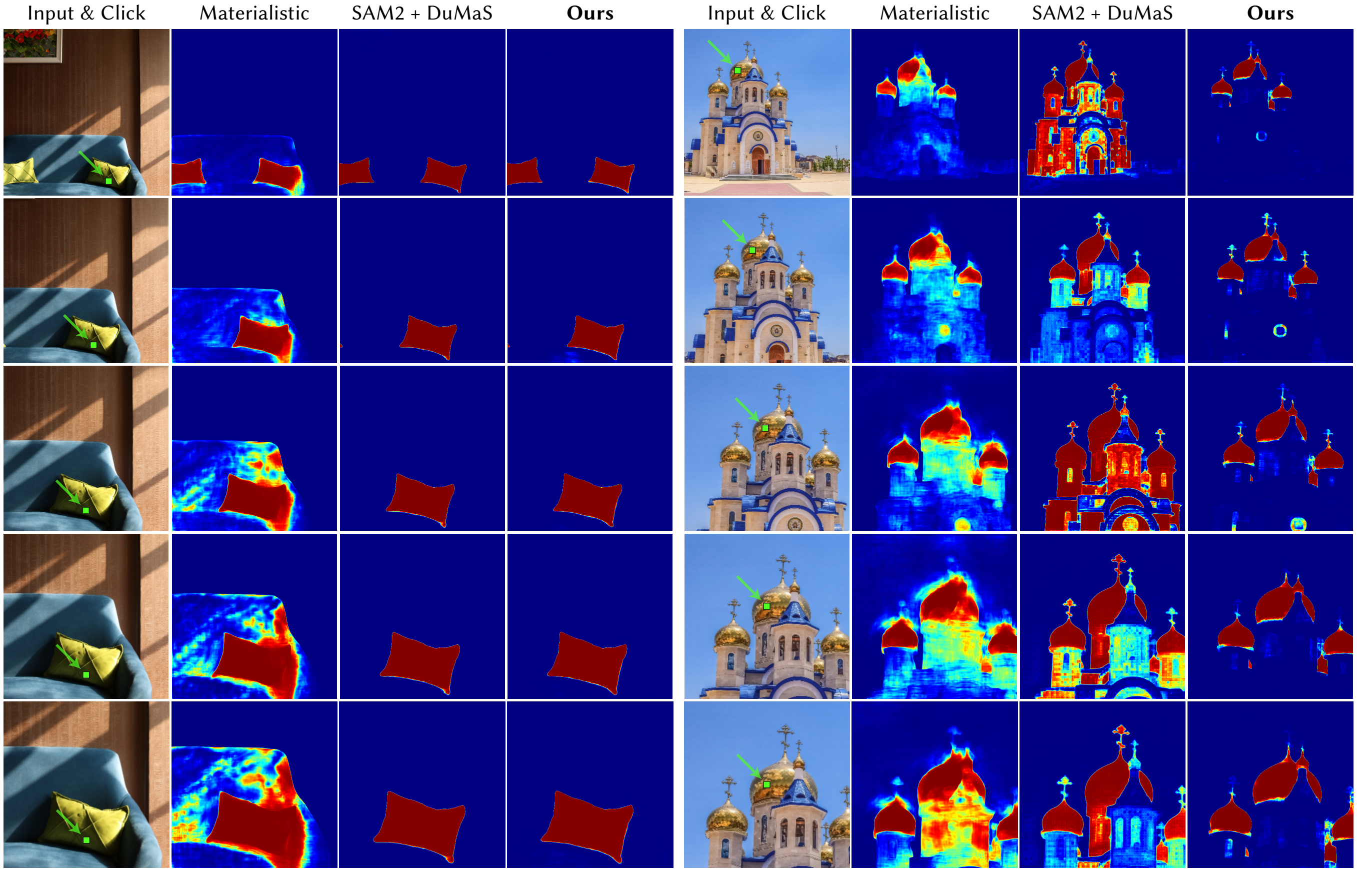

Fig. 10. Zoom consistency. We evaluate the consistency of the selection at different zoom levels. As shown in both examples, our output is consistent even in complex scenes, like the church. On the contrary, Materialistic's outputs are highly dependent on the zoom. SAM2 fine-tuned with our dataset works well for the cushions but fails drastically, and is clearly inconsistent across zoom levels, on the church example.

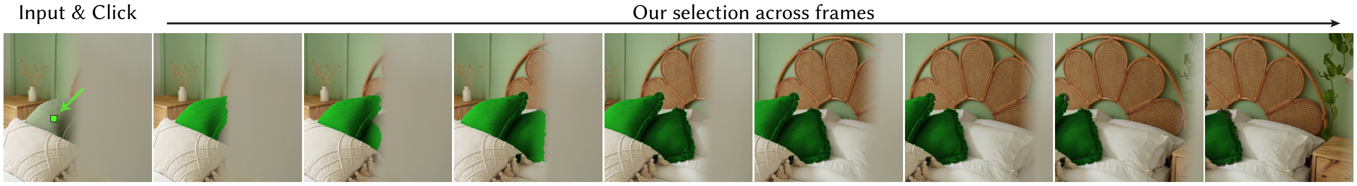

Fig. 11. Our selection on a clicked frame (left), propagated to the remaining frames of a video, progressing towards the right. Our results are consistent, successfully grouping the cushions made of the same material, even when not initially visible in the first frame.

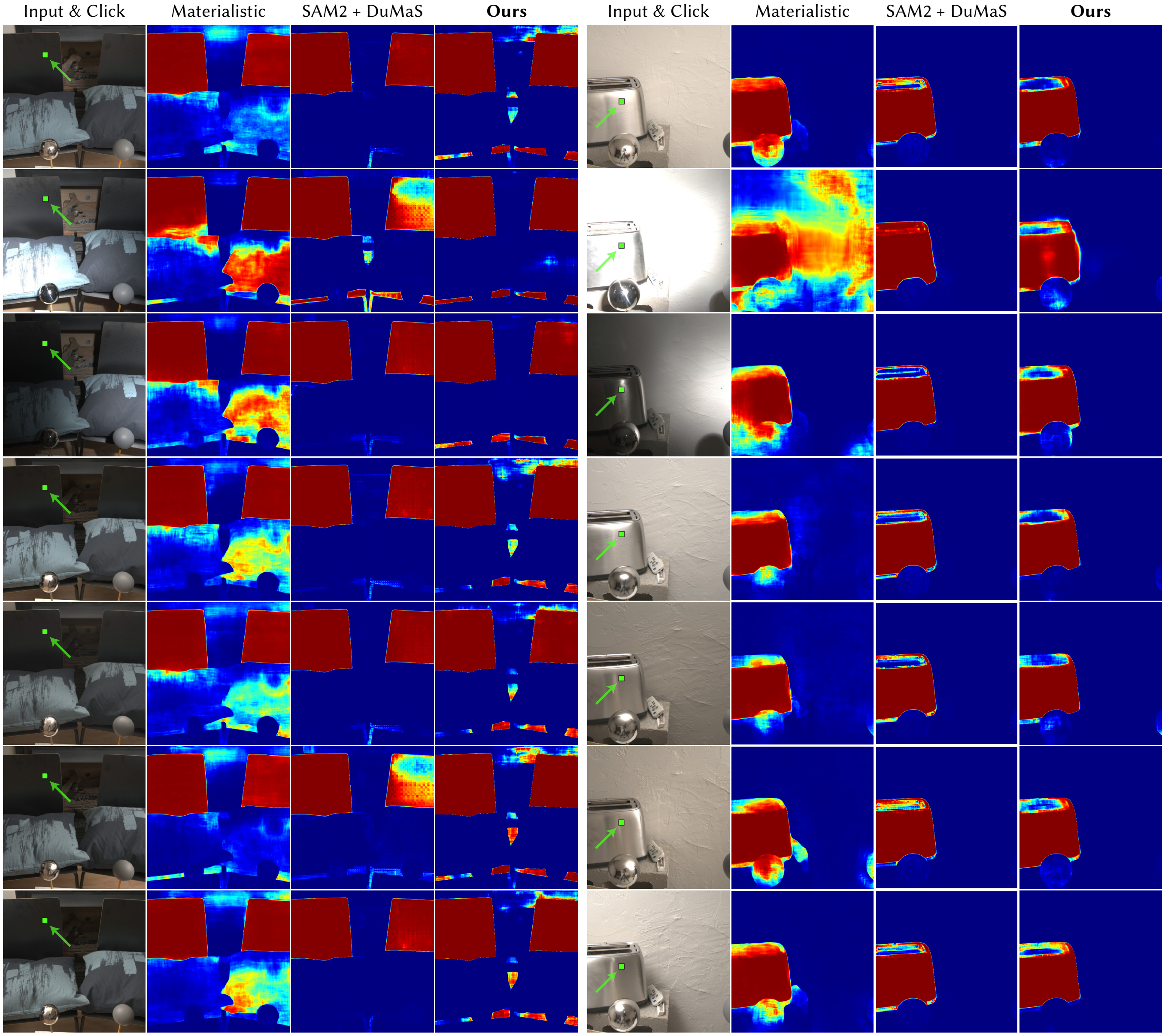

Fig. 12. Illumination consistency. We evaluate the consistency of our method under varying illumination conditions using the Multi-Illumination dataset [Murmann et al. 2019]. We click on the same pixel across all examples. While strong lighting variations (second row) have a slight impact on our estimation, our method demonstrates high consistency overall. This is particularly notable in the challenging scene containing metallic materials, whose appearance changes significantly under varying illumination. In contrast, Materialistic fails to produce consistent outputs. SAM2 fine-tuned with our dataset exhibits quite consistent results, despite failing in some challenging examples (e.g., second row, left).

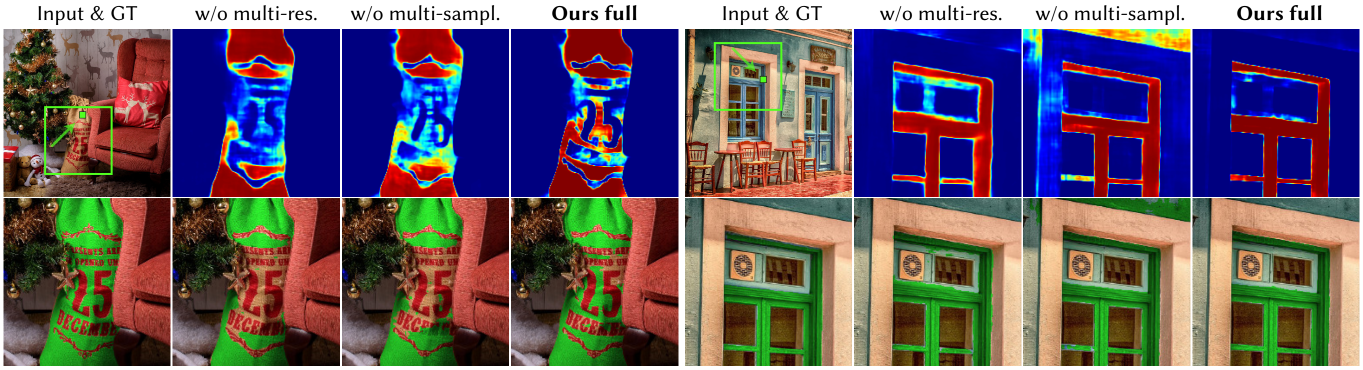

Fig. 13. Additional qualitative ablations for our multi-resolution and multi-sampling components, in difficult scenarios from our Challenging Subset. When removing our multi-resolution pipeline, performance is especially degraded on edges and fine structures, such as the text on the sack (left) and the edges of the window (right). Removing our multi-sampling strategy leads to less confident predictions overall (left) and selection of incorrect areas with similar albedo like the blue wall over the window (right).

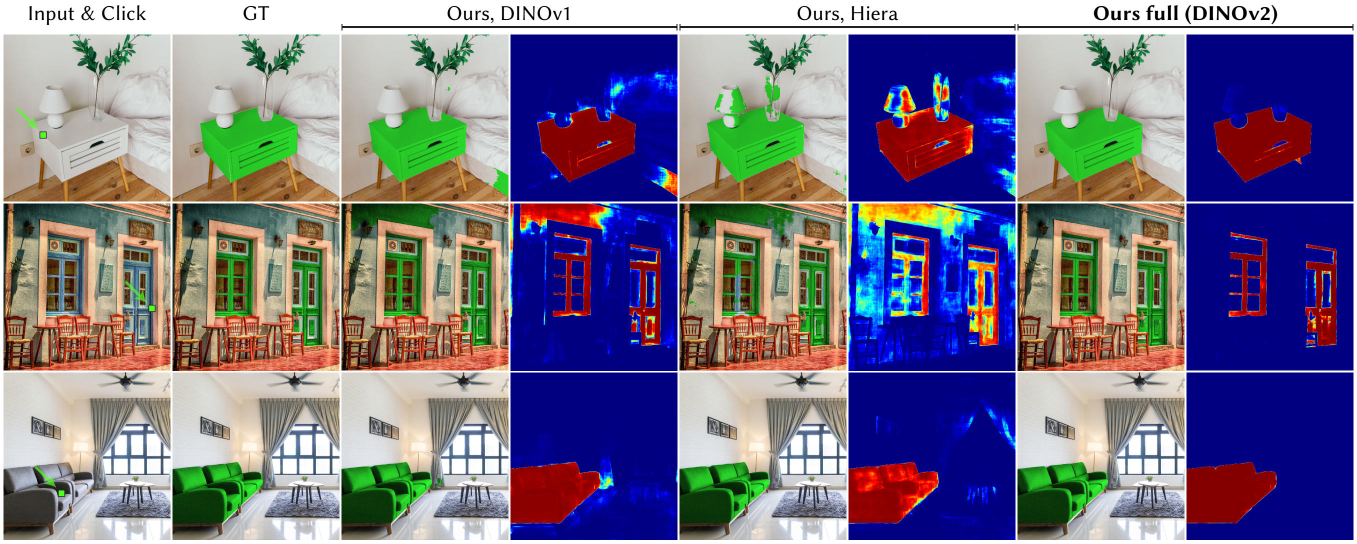

Fig. 14. Qualitative results of ablations changing the image encoder. We tested DINO and Hiera, and both performed worse than our final choice, DINOv2.

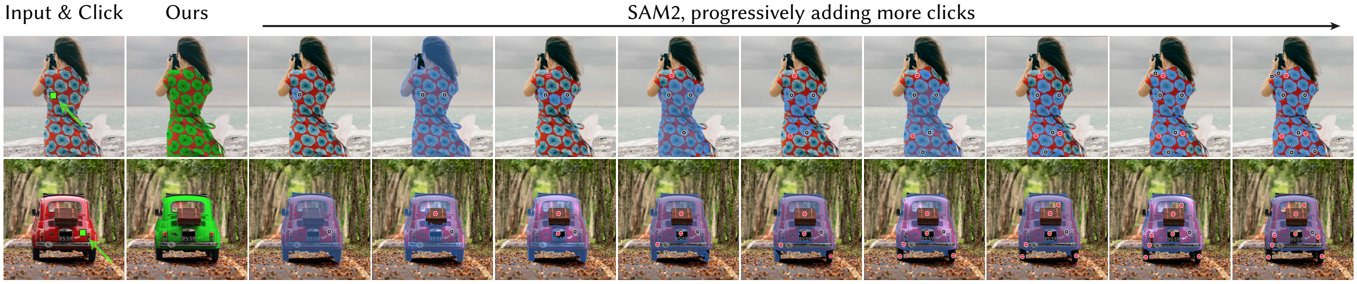

Fig. 15. Comparison of our method vs. SAM2 material selection capabilities. SAM2 requires nine clicks, both positive and negative, to select the flowers on the dress in the upper row, and still does not achieve a satisfactory outcome on the car's metal material in the bottom row. This process also assumes that there is a user in the loop that can iteratively let the selection know what has been missed/overselected. Note that the click and mask visualization from SAM2 is different here (i.e., blue overlay for selection masks) since we use the official SAM2 demo at https://sam2.metademolab.com.