SUPPLEMENTARY

A MORE IMPLEMENTATION DETAILS

Extracting Stable Diffusion Features. Following DIFT (Tang et al., 2023), when we extract Stable Diffusion features, we add a different random noise 8 times and then take the average of the generated features. We use an empty prompt '' as the text prompt.

Train/Val Partition. For the partition of train/val split, we select the train & val images from different scenes for the NYUv2 (Silberman et al., 2012) and ScanNetv2 (Dai et al., 2017) dataset.

Sampling of Images. For the train/val/test splits, if the number of images used is less than the original number of images in the datasets, we randomly sample our train/val/test images from the original datasets.

Sampling of Positive/Negative Pairs. For each property, we try to obtain as many positive/negative region pairs as possible in every image. For each image, if the number of possible negative pairs is larger than the number of possible positive pairs, we randomly sample from the negative pairs to obtain an equal number of negative and positive pairs, and vice versa. In this way, we keep a balanced sampling of positive and negative pairs for the binary linear classifier. As can be observed in Table 1, the number of train/val pairs for different properties are different, although we keep the same number of train/val images for different properties. This is because for different properties the availability of positive/negative pairs are different. For *depth*, we select a pair only if the average depth of one region is 1.2 times greater than the other because it is even challenging for humans to judge the depth order of two regions below this threshold. For *perpendicular plane*, taking the potential annotation errors into account, we select a pair as perpendicular if the angle between their normal vectors is greater than 85°and smaller than 95°, and select a pair as not perpendicular if the angle between their normal vectors is smaller than 60°or greater than 120°.

Region Filtering. When selecting the regions, we filter out the small regions, *e.g.,* regions smaller than 1000 pixels, because regions that are too small are challenging even for humans to annotate.

Image Filtering. As there are some noisy annotations in the (Liu et al., 2019) dataset, we manually filter the images whose annotations are inaccurate.

Linear SVM. The feature vectors are L2-normalised before inputting into the linear SVM. The binary decision of the SVM is given by $sign(w^T v + b)$, where v is the input vector to SVM:

$$v = |v_A - v_B| \tag{4}$$

for the *Same Plane*, *Perpendicular Plane*, *Material*, *Shadow* and *Occlusion* questions, and

$$v = v_A - v_B \tag{5}$$

for the *Support Relation* and *Depth* questions.

Extension of Separated COCO. To study the occlusion problem, we utilise the Separated COCO dataset (Zhan et al., 2022). The original dataset only collects separated objects due to occlusion in the COCO 2017 val split, we further extend it to the COCO 2017 train split for more data using the same method as in (Zhan et al., 2022).

B Analysis of Stable Diffusion Generated Images

As Figure 1 shows, our motivation for the paper is that we observe that Stable Diffusion correctly predicts different physical properties of the scene. The reason why we do not study the generated images directly is that there are no annotations available on different properties for these synthetic images, so it is expensive to get quantitative results. But in this section, we provide more qualitative examples and analysis of Stable Diffusion generated images in terms of different physical properties. The observations match our findings in the main paper – Stable Diffusion 'knows' about a number of physical properties including scene geometry, material, support relations, shadows, occlusion and depth, but may fail in some cases in terms of material and occlusion.

We show examples for: **Scene Geometry** in Figure 6; **Material**, **Support Relations**, and **Shadows** in Figure 7; and **Occlusion** and **Depth** in Figure 8.

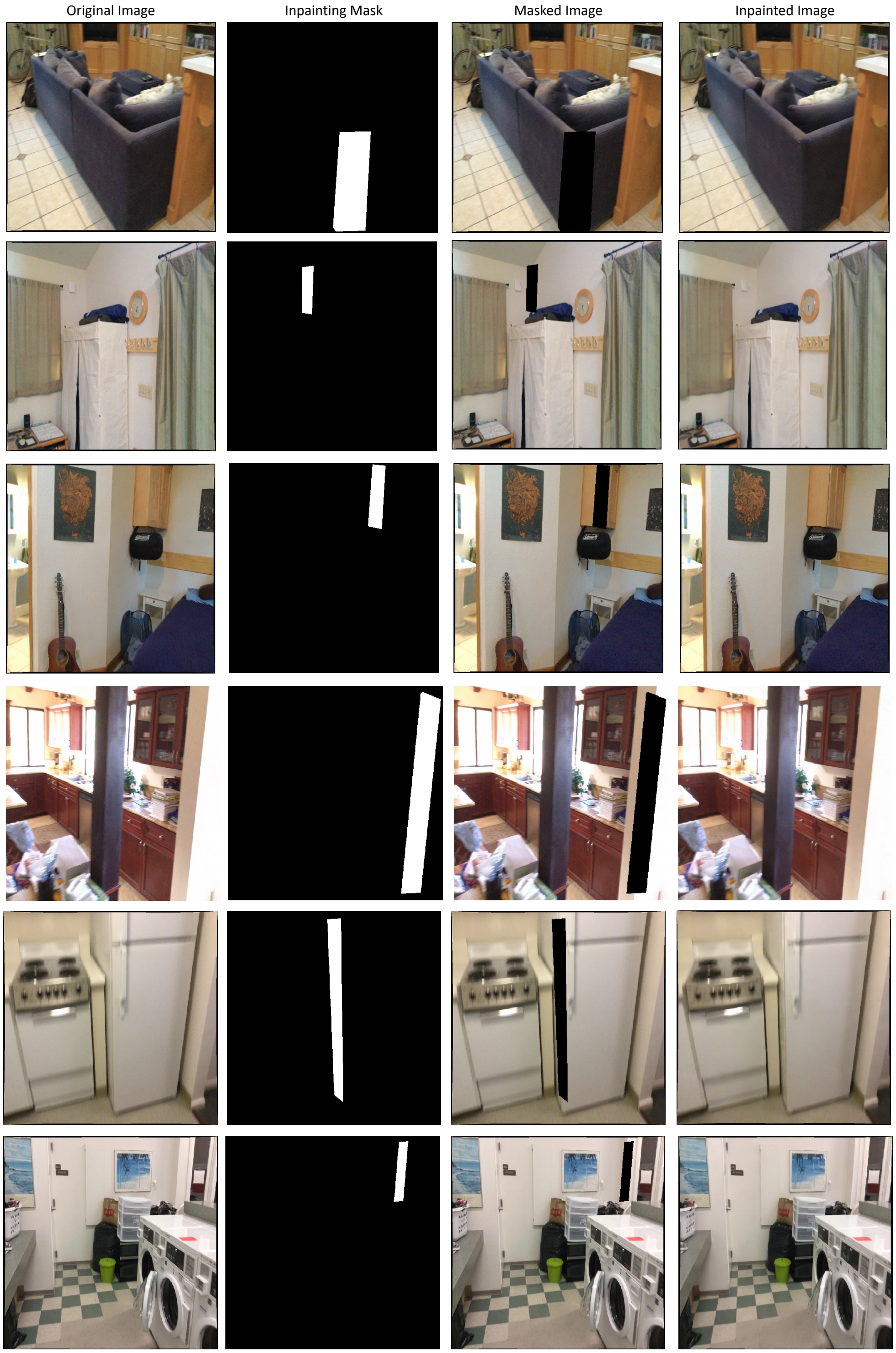

Figure 6: **Stable Diffusion generated images testing *scene geometry* prediction.** Here and for the following figures, the model is tasked with inpainting the masked region of the real images. Stable Diffusion 'knows' about *same plane* and *perpendicular plane* relations in the generation. When the intersection of two sofa planes (first row), two walls (second and sixth row), two cabinet planes (third row), two pillar planes (fourth row) or two fridge planes (fifth row) is masked out, Stable Diffusion is able to generate the two perpendicular planes at the corner based on the unmasked parts of the planes.

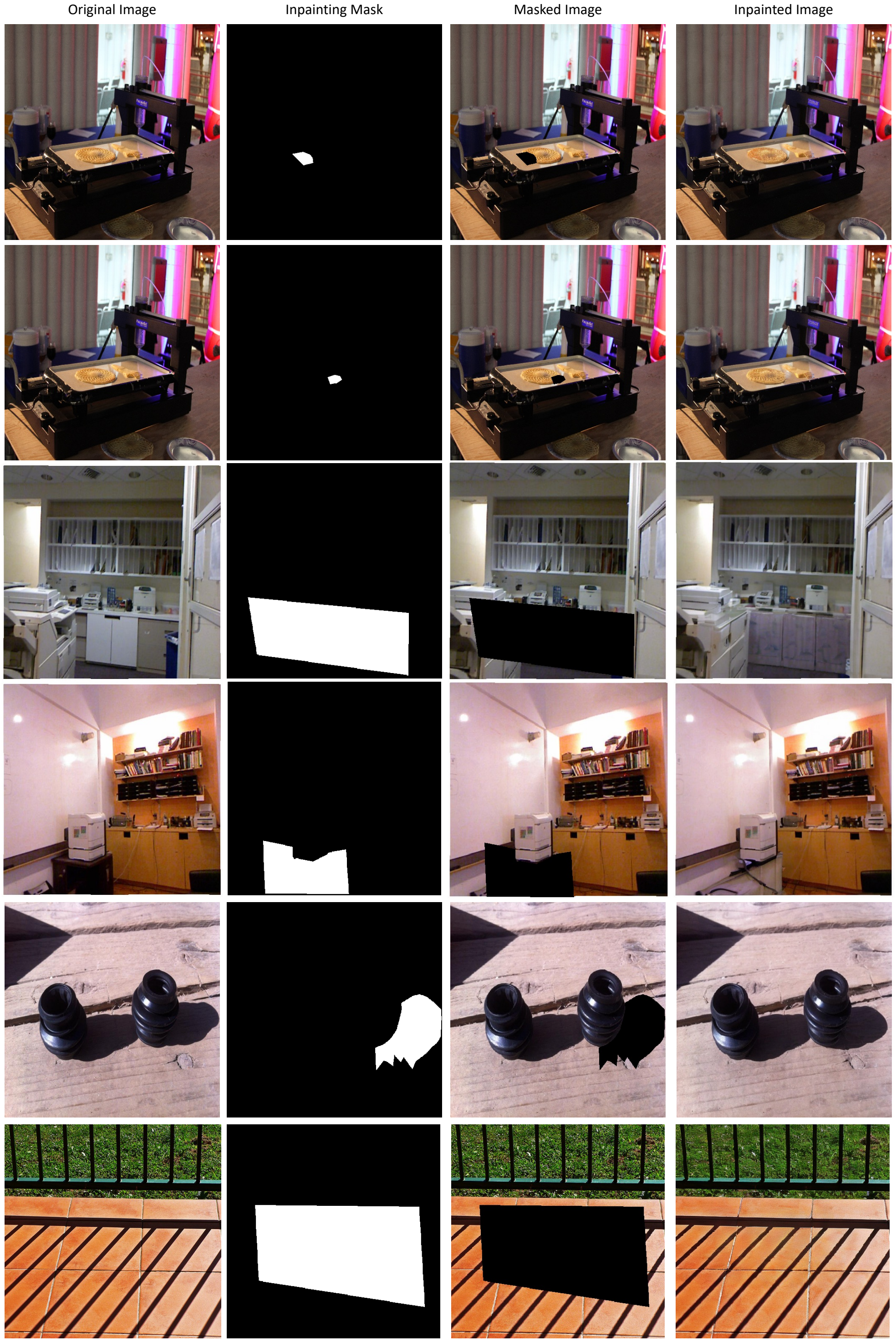

Figure 7: **Stable Diffusion generated images testing *material*, *support relation* and *shadow* prediction.** Stable Diffusion 'knows' about *support relations* and *shadows* in the generation, but may fail sometimes for *material*. Rows 1-2: Material; Rows 3-4: Support Relation; Rows 5-6: Shadow. In the first row, the model distinguishes the two different materials clearly and there is clear boundary between the generated pancake and plate; while in the second row, the model fails to distinguish the two different materials clearly, generating a mixed boundary. In the third row and fourth rows, the model does inpaint the supporting object for the stuff on the table and the machine. In the fifth and sixth rows, the model manages to inpaint the shadow correctly. Better to zoom in for more details.

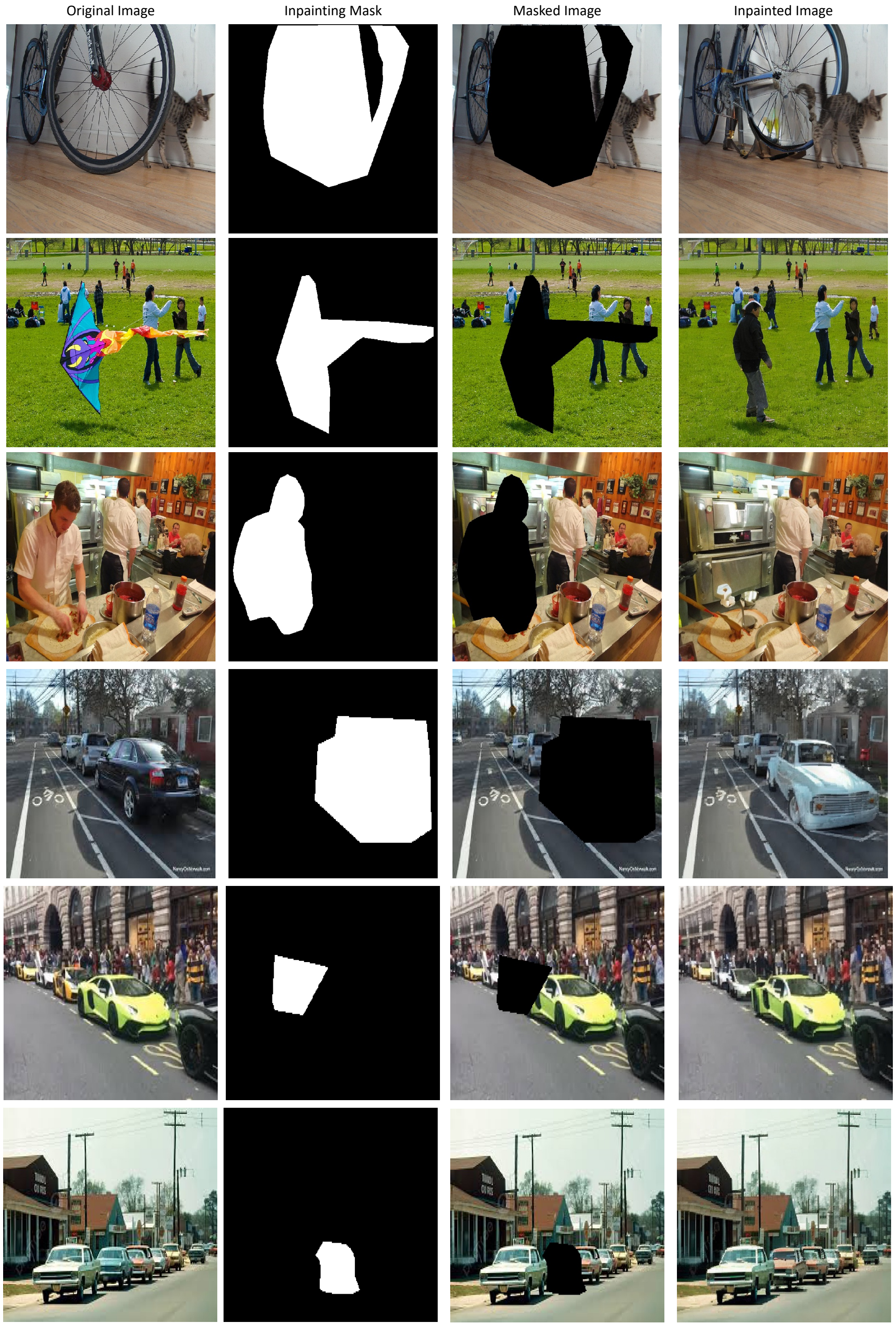

Figure 8: **Stable Diffusion generated images testing *occlusion* and *depth* prediction.** Stable Diffusion 'knows' about *depth* in the generation, but may fail sometimes for *occlusion*. Rows 1-3: Occlusion; Rows 4-6: Depth. In Row 1, the model fails to connect the tail with the cat body and generates a new tail for the cat, while in Row 2, the model successfully connects the separated people and generates their whole body, and in Row 3, the separated parts of oven are connected to generate the entire oven. In Rows 4-6, the model correctly generates a car of the proper size based on depth. The generated car is larger if it is closer, and smaller if it is farther away.

C Additional Results for Other Features Trained at Large Scale

As mentioned in Section 4.3 of the main paper, we have conducted grid search for all the other large pre-trained models, including OpenCLIP, DINOv1, DINOv2 and VQGAN for all tasks. Tables in this section provide results for the Same Plane (Table 5), Perpendicular Plane (Table 6), Shadow (Table 7), Occlusion (Table 8) and Depth (Table 9) tasks for these models. It can be observed that for all tasks, the test performance of each model is improved if we take the best combination of layer and C on the val split, but the performance is still lower than Stable Diffusion.

Table 5: **Performance of different layers for state-of-the-art pre-trained models for the Same Plane task.**

Layer	Split	Same Plane				
		OpenCLIP	DINO v1	DINO v2	VQGAN	Stable Diffusion
Last	Val	72.7	74.9	80.9	65.2	-
Best	Val	84.4	81.7	82.1	77.5	97.2
Last	Test	74.6	79.3	86.0	65.4	-
Best	Test	84.3	82.9	84.5	78.4	95.0

Table 6: **Performance of different layers for state-of-the-art pre-trained models for the Perpendicular Plane task.**

Layer	Split	Perpendicular Plane				
		OpenCLIP	DINO v1	DINO v2	VQGAN	Stable Diffusion
Last	Val	54.9	54.1	62.8	54.6	-
Best	Val	62.6	58.9	68.5	61.3	85.4
Last	Test	55.5	59.8	63.4	50.2	-
Best	Test	61.1	58.6	66.2	54.9	83.9

Table 7: **Performance of different layers for state-of-the-art pre-trained models for the Shadow task.**

Layer	Split	Shadow				
		OpenCLIP	DINO v1	DINO v2	VQGAN	Stable Diffusion
Last	Val	78.1	85.4	88.5	50.0	-
Best	Val	93.9	88.8	90.2	86.0	96.1
Last	Test	75.5	84.3	86.8	50.8	-
Best	Test	92.0	86.9	87.0	85.9	94.5

Table 8: **Performance of different layers for state-of-the-art pre-trained models for the Occlusion task.**

Layer	Split	Occlusion				
		OpenCLIP	DINO v1	DINO v2	VQGAN	Stable Diffusion
Last	Val	61.5	65.3	65.8	49.7	-
Best	Val	74.0	71.3	70.3	72.5	83.2
Last	Test	63.8	60.0	67.9	53.9	-
Best	Test	65.6	62.0	67.1	60.4	75.6

Table 9: **Performance of different layers for state-of-the-art pre-trained models for the Depth task.**

Layer	Split	Depth				
		OpenCLIP	DINO v1	DINO v2	VQGAN	Stable Diffusion
Last	Val	96.8	94.4	97.5	79.4	-
Best	Val	98.4	95.5	98.4	90.9	99.5
Last	Test	95.5	93.7	98.0	73.8	-
Best	Test	97.7	94.4	98.4	90.5	99.3

D Visualisation of Stable Diffusion Feature Representations

In Figure 9 we visualise the vectors representing the positive/negative pairs in the Depth and Material tasks using t-SNE. It is obvious that the vectors are easier to be separated for the Depth task than the Material task, which confirms to the observation that we get a higher AUC when we apply linear SVM to the depth task but lower AUC when we apply it to the material task. In the future, more efforts should be put into training the Stable Diffusion model to have a better understanding of Material and Occlusion, *e.g.*, explicitly incorporate these tasks into training.

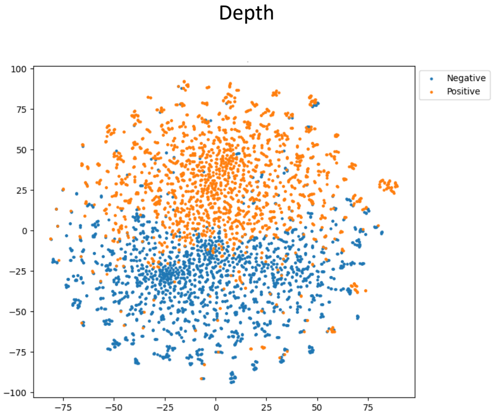

Figure 9: **t-SNE Visualisation of Stable Diffusion feature space for the *Depth* and *Material* tasks.** It can be observed that the vectors for the depth task are more easy to separate than the material, which confirms to the observation that we get a higher AUC when we apply linear SVM to the depth task but lower AUC when we apply it to the material task.