VideoNeuMat: Neural Material Extraction from Generative Video Models
Supplementary Material

Fig. 1. **Tilable texture results**. We include a few materials examples obtained from our tilable variant. The first image shows the location of the seams. In many cases, the patterns seem to match at the edges, however, the reflectance does seem to differ, causing visible seams. Further architectural changes can improve tilability such as circular convolutions in the LRM, etc.

1 Checkpoint Selection and Prior Preservation

We use the RPC metric to identify the optimal stopping point that balances trajectory compliance with prior preservation. Here we qualitatively validate this selection using out-of-distribution prompts.

We test "cat molded silver" and "dragon molded copper"—concepts absent from MatSynth—to verify that our selected checkpoint indeed preserves semantic knowledge from the pretrained prior. Supplementary Figure 2 confirms our selection: at iteration 2000, the model generates recognizable cat sculptures and intricate dragon reliefs while following the learned trajectory.

The figure also illustrates what would happen with suboptimal selection. Training beyond our chosen checkpoint leads to progressive semantic collapse: the cat dissolves into featureless planes by 5k, while dragon details degrade to generic MatSynth-like textures by 9k.

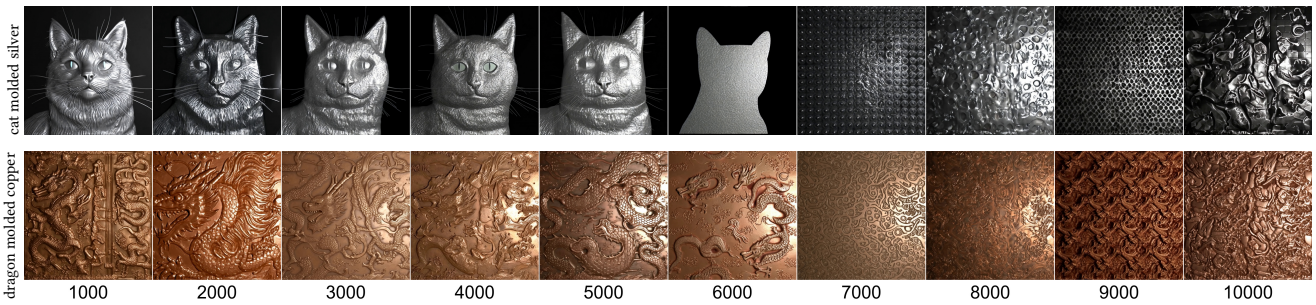

Fig. 2. **Effect of training iterations on out-of-distribution generalization.** Top: recognizable cat sculpture (1k) collapses to featureless planes (5k) then unrelated patterns (6k+). Bottom: detailed dragon reliefs (1k–2k) degrade to generic textures (9k–10k).

2 Tileable Material Generation

For applications that require periodic textures, we apply a lightweight post-processing step to the generated latent material texture. The goal is to remove boundary discontinuities without changing the generator, retraining the LRM, or running an iterative optimization. Given a latent texture $\mathbf{z} \in \mathbb{R}^{H \times W \times C}$ decoded by our material generator, we modify only a narrow band along the four image boundaries and leave the interior unchanged.

We use an edge band of width $b = 32$ in latent space. For every pixel p in a left or right edge band, let $\pi(p)$ denote the horizontally flip-paired pixel on the opposite edge, i.e., the pixel that becomes adjacent to p when the texture is tiled. Top and bottom bands are paired

analogously. A constant averaging of the entire band can suppress seams, but it also over-smooths material structure near the border. We therefore use a distance-dependent weight

$$w(d) = w_{\max} - (w_{\max} - w_{\min})\frac{d}{b-1}, \quad w_{\max} = 1.0,\ w_{\min} = 0.25, \tag{1}$$

where d is the distance from the boundary within the band. Each paired latent value is moved toward the pair average by

$$\mathbf{z}'_p = \mathbf{z}_p + w(d_p)\,\frac{1}{2}\left(\mathbf{z}_{\pi(p)} - \mathbf{z}_p\right). \tag{2}$$

The same closed-form projection is applied once to all four edge bands. At corners, where horizontal and vertical bands overlap, we average the two projection proposals before writing the updated latent value. This enforces continuity at the tile boundary while keeping the perturbation localized to the border region.

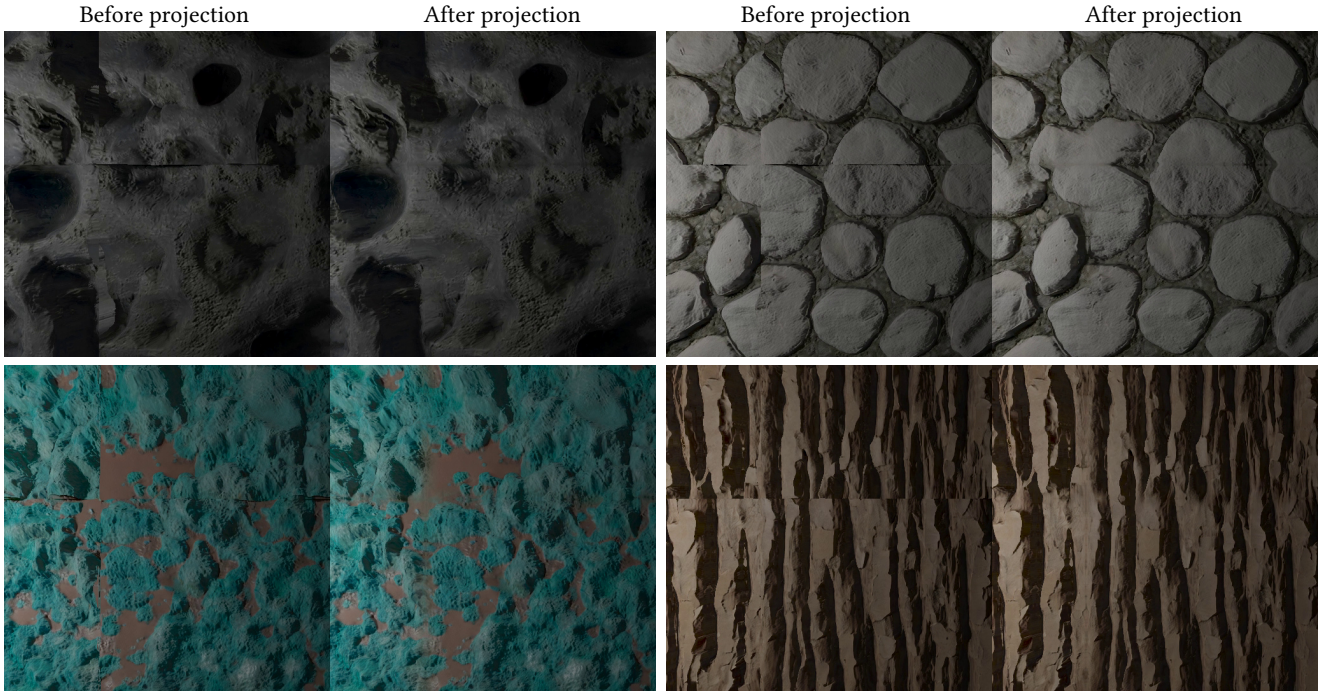

Fig. 3. Representative tileable material examples selected from challenging frames where the unprocessed baseline exhibits visible tiling seams. In each panel, the left half shows the decoded material before latent edge projection, and the right half shows the result after projection at the same video timestamp. The operation reduces visible tiling seams while preserving the material appearance away from the boundaries.

The projection is deterministic and requires only direct tensor operations on the latent texture. It adds negligible runtime compared to video generation or LRM inference, and the same parameters are used for all examples in Fig. 3.

3 Nearest-Neighbor Analysis Against MatSynth

We perform a nearest-neighbor analysis to examine whether the generated materials are simple reproductions of the MatSynth training set. For each generated sample, we compare its first generated frame against the complete MatSynth training set, using the same canonical rendering setup for all candidates, and retrieve the top-ranked neighbors over this entire candidate pool. The retrieval score combines semantic similarity from CLIP with a weaker perceptual appearance term from LPIPS:

$$S(x, y) = 0.9\,\mathrm{sim}_{\mathrm{CLIP}}(x, y) + 0.1\left(1 - d_{\mathrm{LPIPS}}(x, y)\right), \tag{3}$$

where $\mathrm{sim}_{\mathrm{CLIP}}$ is the cosine similarity between CLIP image embeddings and d_{LPIPS} is the LPIPS perceptual distance.

Figure 6 shows the nearest neighbors for representative generated materials. Since the search is performed over the complete MatSynth training set, these are the closest MatSynth matches available under our retrieval metric, rather than examples selected from a small subset. Many of our generated materials have no close counterpart in MatSynth: for instance, a croissant-like material retrieves wood and stone as nearest neighbors, and a dragon fabric retrieves only plain cloth. This confirms that our method generates materials well beyond the coverage of existing datasets.

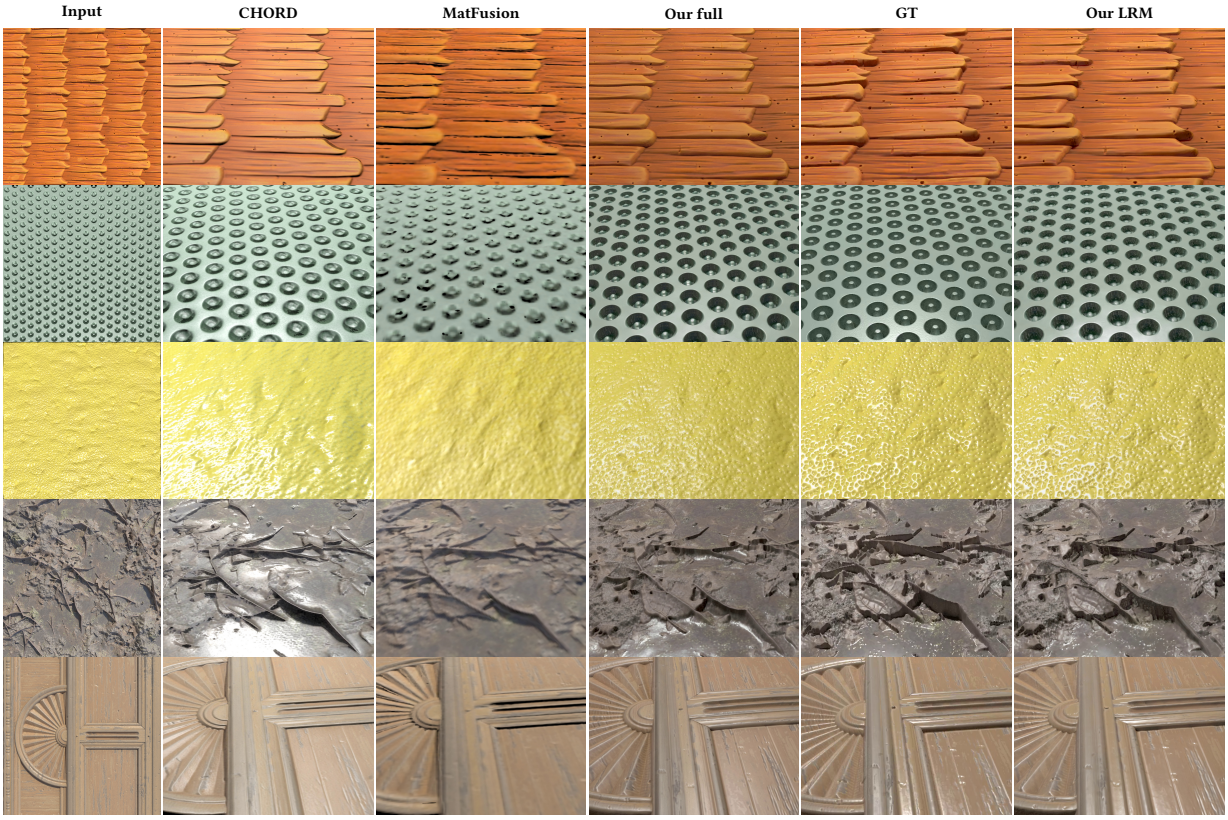

Fig. 4. Additional image-to-texture comparison examples. Columns and method labels follow the same convention as the main-text figure.

4 Prompt Augmentation Instructions

We use the following prompt to instruct an LLM (Claude Opus 4.5) to expand concise material descriptors into detailed captions:

> "Help me modify the following prompts one by one, adding some descriptions to each prompt so that they naturally possess a certain degree of displacement. These descriptions should relate to the inherent attributes of the material itself, not changes in lighting or environment. These descriptions will be used for generation with a video diffusion model. They should still align with the original material descriptions, just with added detail descriptions to give these prompts more displacement."

For example, Given a concise material descriptor ("green moss"), the LLM automatically augments it into a detailed caption following the training format ("The video clip features a close-up view of cushion moss mound with dense velvety surface, thousands of tiny upright stems creating unified texture, rich emerald green with golden new growth tips, dewdrops caught in fibrous canopy"). This expansion incorporates surface micro-geometry, weathering characteristics, color variations, and material-specific details.

5 Increased resolution to 2K using the LRM

We include a variant of our method that reconstructs 2K material textures from 1k input generated frames. Visualization and more details in Figure 10.

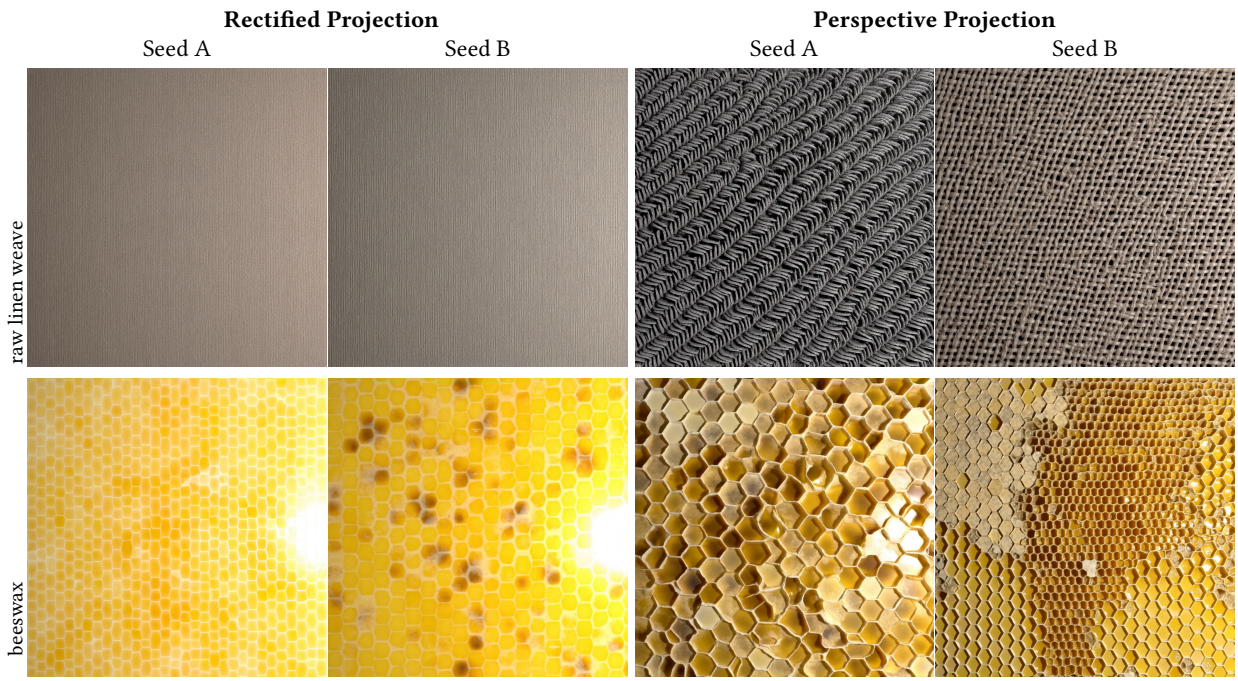

Fig. 5. Ablation on projection mode. For each text prompt, we generate material videos using two different random seeds. **Rectified projection** (left): The model suffers from mode collapse—different seeds produce nearly identical outputs with flat, low-detail textures that lack the diversity of the pretrained prior. **Perspective projection** (right, ours): Different seeds yield visually distinct materials with rich geometric detail and varied appearances, demonstrating that perspective projection preserves the generative diversity of the pretrained video model.

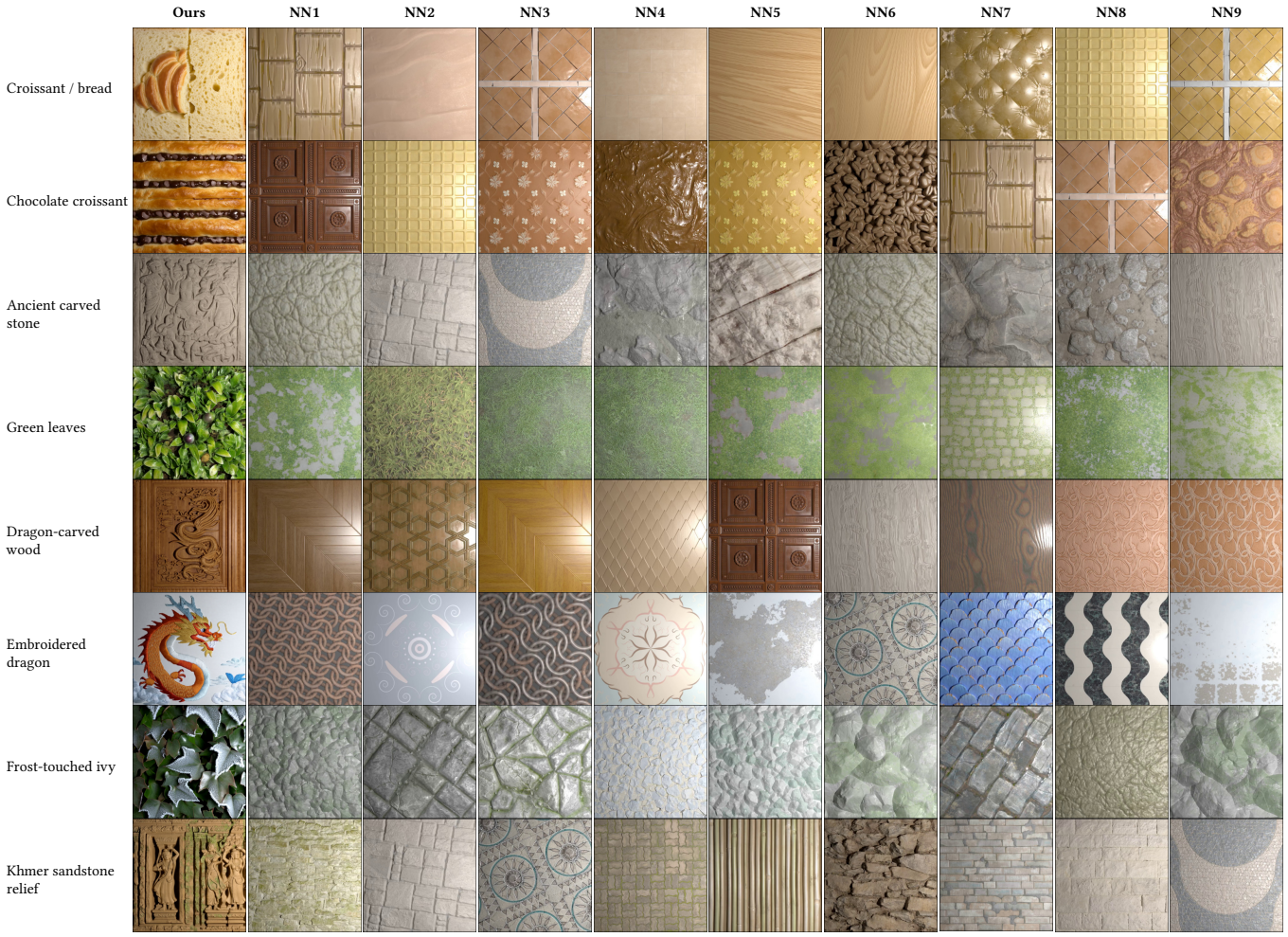

Fig. 6. Nearest-neighbor analysis against the complete MatSynth set.Each row shows one generated sample and its top nine nearest MatSynth neighbors over the entire candidate pool. Nearest-neighbor results in MatSynth indicate that our generated materials lie outside the dataset's distribution.

Fig. 7. Qualitative results at 512×512 using our model. For each material, the left image shows the inferred texture rendered on a cloth drape, while the right image is a zoomed-in crop highlighting high-quality displacement and parallax.

Fig. 8. **Reconstruction results.** A selection of materials generated by our pipeline from text prompts, shown on a flat and curved surface under different illuminations. Note the realism of the results and the ability to handle non-trivial geometry (leaves, fur, fabric) that cannot be represented by heightfields. Please see the extensive supplementary materials for animated results, showing parallax effects.

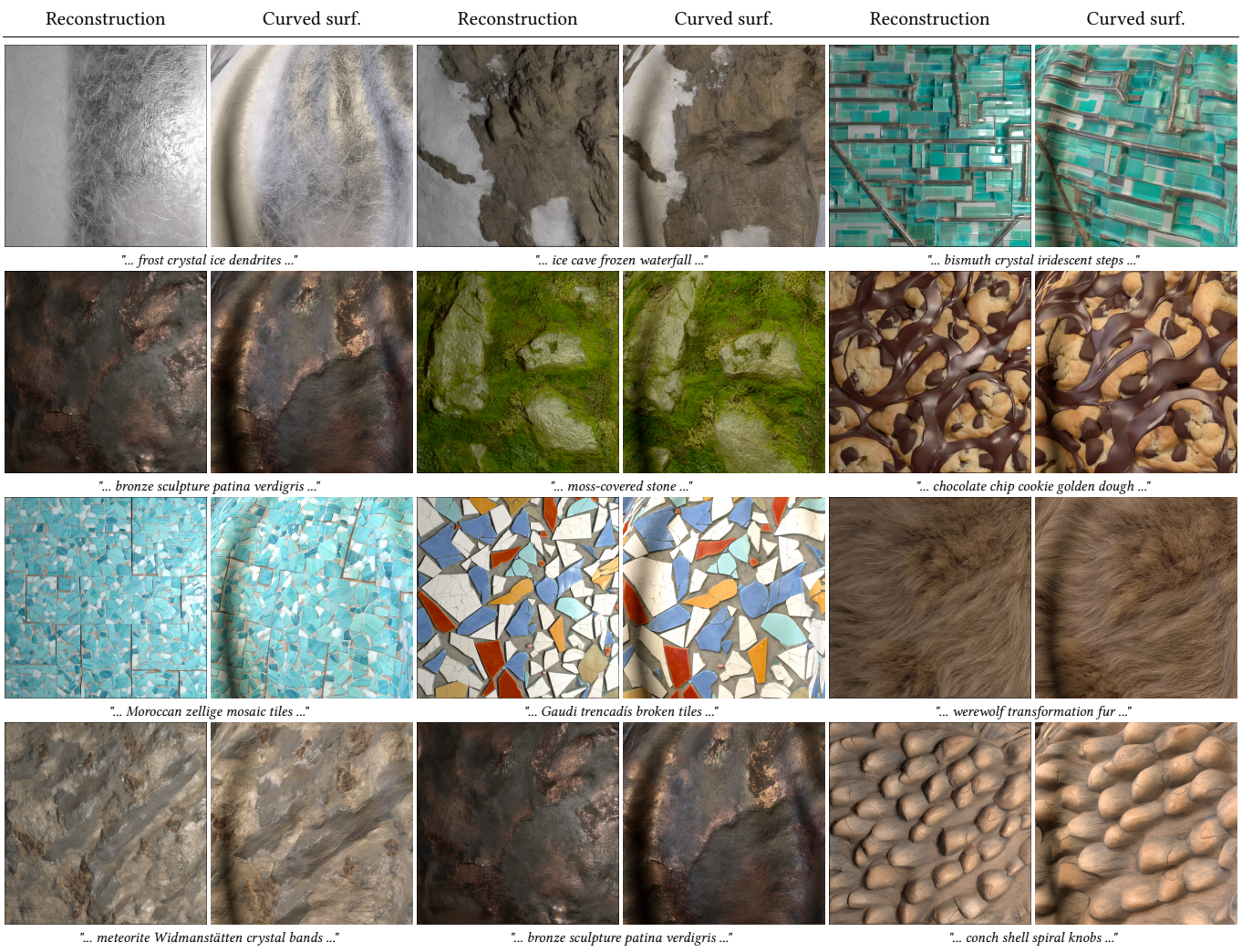

Fig. 9. **Reconstruction results.** A selection of materials generated by our pipeline from text prompts, shown on a flat and curved surface under different illuminations. Note the realism of the results and the ability to handle non-trivial geometry (leaves, fur, fabric) that cannot be represented by heightfields. Please see the extensive supplementary materials for animated results, showing parallax effects.

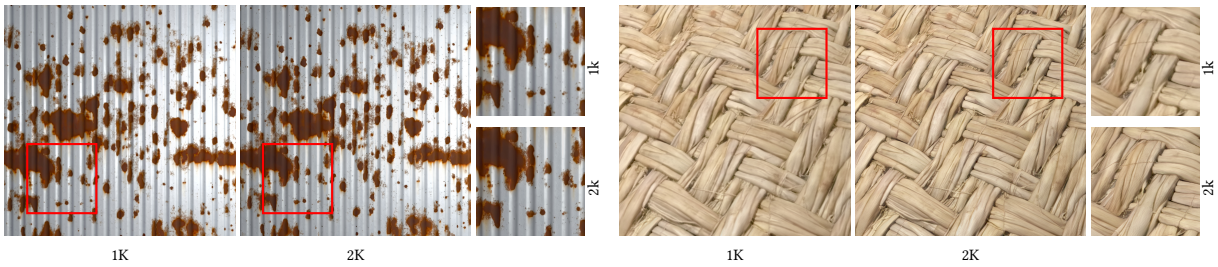

Fig. 10. Comparison of 1K and 2K results using our approach. To obtain 2K results, we still input 1K diffusion frames to the LRM, but the LRM reconstruction is in 2K. In order to do the upsampling in the LRM, we add new upsampling layers in the VAE decoder that are trained from scratch. Red boxes indicate the zoomed area.