MatE: Material Extraction from Single-Image via Geometric Prior

Supplementary Material

6. More Details on Tileability

6.1. Tileable Material Extraction

Although our proposed method does not explicitly impose tileability constraints during training, we demonstrate that it can effectively generate seamless PBR materials. By incorporating the noise rolling strategy [41] during the inference sampling stage, we achieve tileability without the need for fine-tuning. Fig. 10 presents qualitative results demonstrating this capability.

6.2. Seam Analysis

However, while the noise rolling strategy in ControlMat [41] significantly facilitates boundary continuity, it does not strictly guarantee seamlessness; perceptible seams often persist when the generated materials are tiled. Furthermore, our empirical experiments reveal a notable performance discrepancy between tiling in the latent space versus tiling the final RGB outputs. We hypothesize that this variation stems from the inherent smoothing properties of the pre-trained VAE decoder, which affects how boundary discontinuities are reconstructed.

We perform a comprehensive analysis of tiling artifacts by comparing four experimental settings in Fig. 9: (1) The original tileable material (tiled in RGB space); (2) The latent-space tiled result, where latent features are tiled prior to decoding (without subsequent RGB tiling); (3) The VAE reconstruction of the original material (tiled in RGB space); (4) The top-left crop of the decoded RGB image from Setting 2, which is then tiled. As shown in Fig. 9, while the latent-space Tiled result (Col 2) is seamless, the re-tiled crop (Col 4) exhibits perceptible seams. This observation isolates the critical limitation: the VAE decoder introduces artifacts at the absolute image boundaries. While internal transitions in the latent-tiled image (Col 2) are smoothed by the decoder, the absolute boundaries are corrupted by reconstruction artifacts. This results in visible seams in the re-tiled crop (Col 4) despite the maintained structural tileability. This limitation is further evidenced by the VAE reconstruction (Col 3), where the decoder inherently introduces boundary seams into the originally seamless material.

Figure 9. Comparison of tiling consistency between latent and RGB spaces.

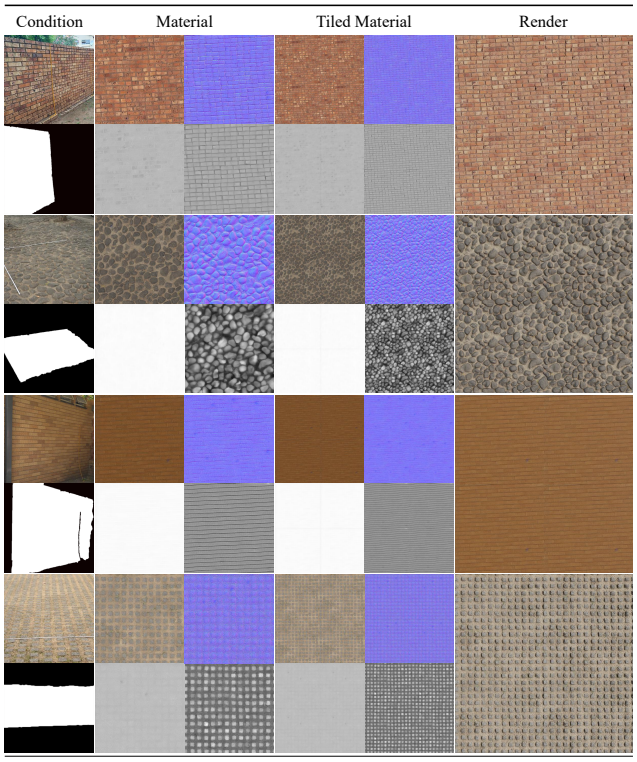

Figure 10. Visualization of generated tileable materials. The figure displays tileable materials generated by noise rolling [41].

7. More Details Details on Implementation

7.1. Dataset Implementation

Dataset Generation. We derive our training data from the high-quality PBR materials provided in the MatSynth dataset [40]. To generate paired data, we utilize Blender to render corresponding images. Given the limited number of unique materials, we employ a robust data augmentation strategy to scale up the dataset. Specifically, for each material, we render 20 distinct views by randomly sampling camera poses and applying random Thin-Plate Spline (TPS) transformations to the underlying planar mesh, thereby simulating diverse geometric surface variations. To further ensure diverse illumination conditions, we utilize 100 HDRI environment maps collected from PolyHaven [2] during the rendering process. This combination of geometric and pho-

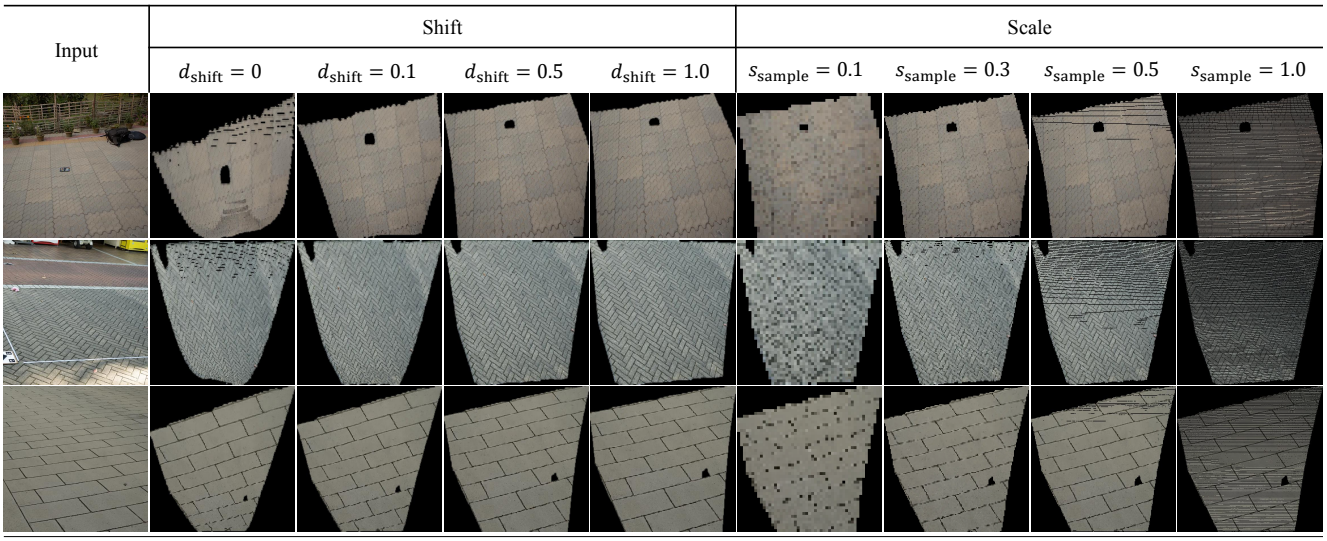

Figure 11. Sensitivity analysis of hyperparameters. We demonstrate that a large depth shift d_{shift} diminishes perspective correction strength by flattening depth, while a small scale factor s_{sample} fills projection holes at the cost of image sharpness.

tometric augmentation ultimately results in a comprehensive dataset of 100k image-material pairs.

Rotational Alignment Mechanism. During the synthetic data generation pipeline in Blender, we explicitly record the camera extrinsic parameters. Utilizing this pose information allows us to bypass the need for the model to implicitly learn the canonical orientation of the materials. As illustrated in Fig. 12, without this explicit rotational alignment, the model suffers from orientation ambiguity, being unable to uniquely determine the canonical orientation of the extracted material. By removing this degree of freedom, we simplify the learning objective, allowing the network to focus solely on material reconstruction rather than inferring the canonical orientation.

Comparison of Data Generation Pipelines. We further compare the synthetic datasets generated by our strategy against those from MaterialPicker [26]. As visualized in Fig. 13, our pipeline demonstrates superior capability in preserving the structural integrity of the textures. In contrast, MaterialPicker applies textures onto complex geometries, which inevitably introduces discontinuities and severe distortions, thereby disrupting the structural coherence of the texture patterns.

7.2. Model Implementation

Our method is implemented using PyTorch. We empirically observed that initializing our model with pre-trained Stable Diffusion weights did not noticeably accelerate convergence. We attribute this to the significant domain gap between the natural image distribution used in Stable Diffusion pre-training and the specific PBR material data required for our task. Consequently, we train our Denoising UNet from scratch. Our Denoising UNet follows the standard architecture of Stable Diffusion. However, to optimize GPU memory efficiency during training, we reduce channel dimensions to $[256, 512, 1024, 1024]$. Our experiments confirm that this lightweight configuration yields negligible impact on convergence performance while significantly reducing computational overhead. Regarding the dual-branch design, we removed the cross-attention modules from the Reference UNet. We extract the Key and Value features from the Reference UNet and inject them into the cross-attention layers of the Main UNet to guide the generation. For the Variational Autoencoder (VAE), we adopt the pre-trained checkpoint from Stable Diffusion v1.4 (kept frozen). We train our model using the AdamW optimizer with a fixed learning rate of 6.4×10^{-5}. The training process spans ap-

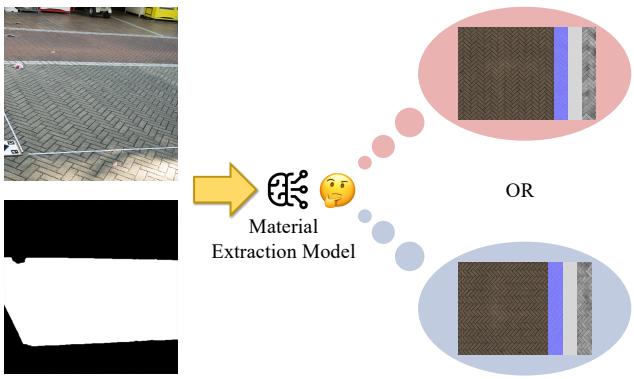

Figure 12. Illustration of orientation ambiguity. Given an input image and mask, the model will face inherent uncertainty regarding the canonical orientation of the material.

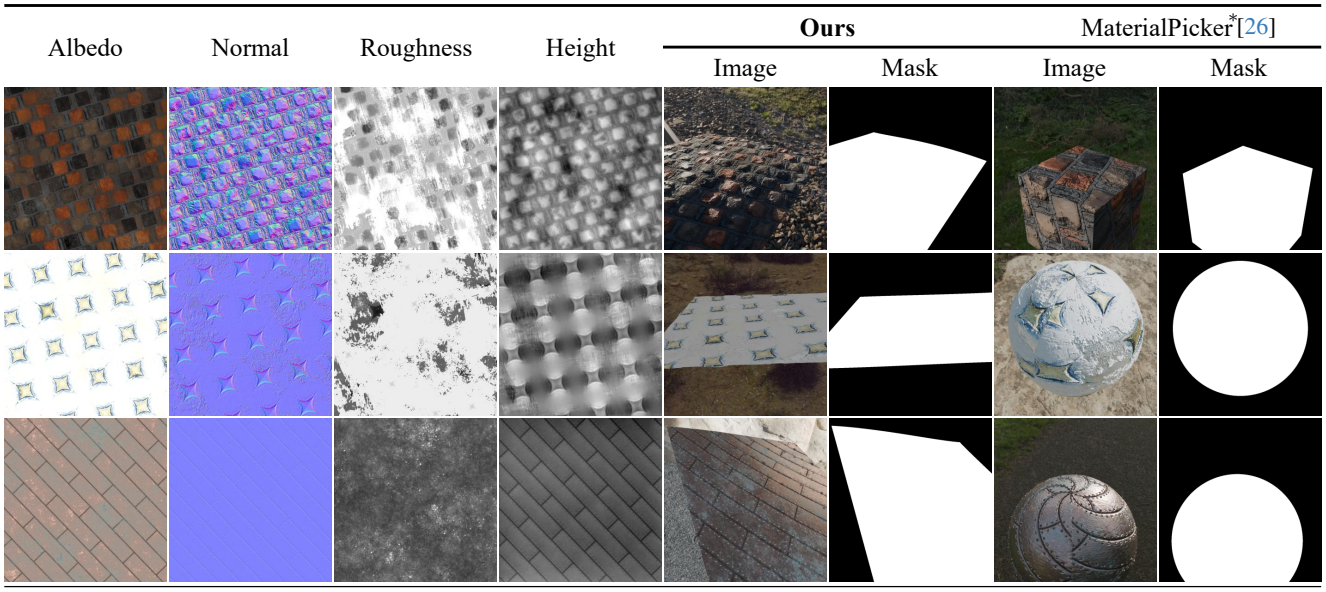

Figure 13. Quality comparison of synthetic datasets. We compare materials generated by our pipeline against MaterialPicker [26]. MaterialPicker utilizes complex underlying geometries, which inevitably lead to severe distortions and discontinuities, breaking the structure of the texture. In contrast, our approach effectively preserves the structural integrity and coherence of the patterns.

proximately $400,000$ iterations with a total batch size of 32. The entire training requires roughly 5 days on a workstation equipped with 4 NVIDIA A6000 GPUs.

8. More Experimental Results

8.1. Temporal Dependencies in MaterialPicker

To quantify causality, we analyzed cross-frame attention maps at $t = 300$ (avg. 226 samples). We extracted scores from the last DiT block and applied min-max normalization along the last dimension before computing the spatial sub-block ($H \times W$) means. As illustrated in Fig. 14, the resulting dependency matrix exhibits a distinct diagonal pattern, confirming that the failure in Fig. 7 is structural and deterministic, driven by a strong dependency on the preceding Normal channel.

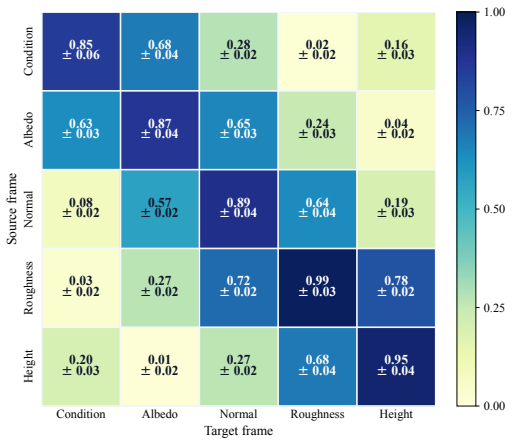

Figure 14. Causality analysis of cross-frame attention. The distinct diagonal pattern reveals the erroneous temporal dependencies introduced by MaterialPicker [26].

8.2. Geometry-based Rectification

Impact of Hyperparameters. As detailed in Sec. 3.3, we introduce the hyperparameters d_{shift} and $[s_x, s_y]$ to constrain the projection process. Specifically, d_{shift} is designed to mitigate the ambiguity of monocular relative depth estimation. Without this shift, points with near-zero depth values would be projected erroneously close to the principal point (c_x, c_y), causing severe geometric distortion. By enforcing the depth range to $[d_{\text{shift}}, 1+d_{\text{shift}}]$, we ensure numerical stability.However, we observe a critical trade-off: excessively increasing d_{shift} dampens the relative depth variations, effectively flattening the geometry and reducing the strength of the perspective correction. Consequently, a large d_{shift} compromises the rectification performance by diminishing effective depth variance relative to the absolute depth. We visualize the sensitivity of our rectification results to varying d_{shift} values in Fig. 11. Simultaneously, the hyperparameter s_{sample} constraints $[s_x, s_y]$ is governed by the aspect ratio changes between the source and target views. During the reprojection of the 3D point cloud back onto the 2D image plane, sparse sampling can occur, where gaps between projected points manifest as holes (empty pixels).Our experiments show that employing a smaller s_{sample} effectively mitigates these artifacts by densifying the projection on a

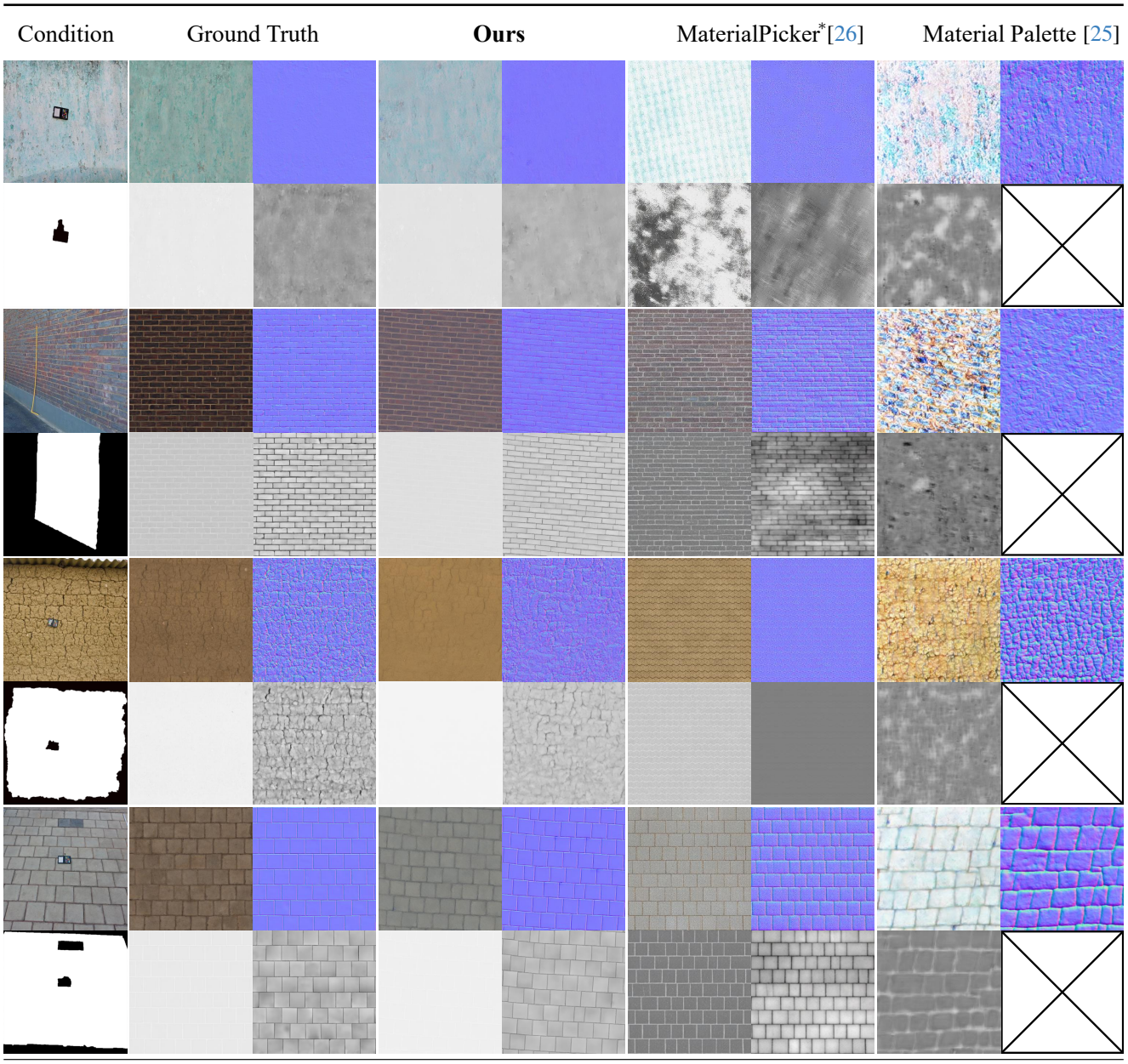

Figure 15. More qualitative results on the evaluation set.

lower-resolution grid. However, to restore the output to its original resolution, we apply bilinear interpolation for upsampling. While this strategy successfully fills the holes, it inevitably introduces blurring and leads to a loss of high-frequency details. We visualize this trade-off between hole reduction and detail preservation in Fig. 11. Note that for this visualization, we intentionally omit the interpolation step defined in Eq. (8) to clearly exhibit the raw projection results and sampling artifacts.

Limitations on Near-Orthogonal Views. One limitation arises when the camera viewpoint is near-orthogonal to the surface and the scene lacks surrounding depth cues (e.g., background objects). In such cases, the absolute depth distribution of the target surface theoretically exhibits negligible variance. However, monocular relative depth estimators inherently normalize predictions to a fixed range of $[0, 1]$. This forced normalization artificially amplifies the depth variance, introducing spurious geometric distortions into an undistorted texture, as illustrated in Fig. 17. Despite this, the induced distortion is generally within a manageable range. Furthermore, thanks to our coarse-to-fine strategy, the subsequent generation stages are not strictly dependent

Depth Model	LPIPS ↓	SSIM ↑	CLIP ↑
Omnidata [10]	0.418	0.377	0.723
Marigold [20]	0.415	0.381	0.719
Depth Anything V2 [49]	0.422	0.374	0.722

Table 4. Quantitative results on depth estimation models.

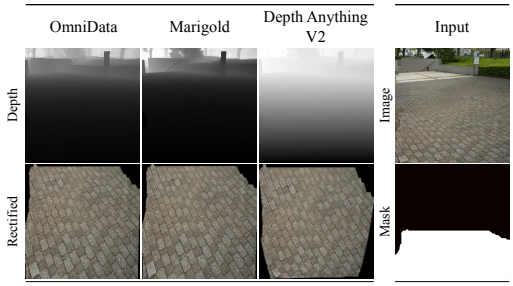

Figure 16. Qualitative results on depth estimation models.

on perfectly rectified inputs. Our pipeline demonstrates robustness to these minor geometric imperfections, effectively correcting or tolerating them during the refinement process.

8.3. More Qualitative Results

Evaluation Set Examples. We provide additional qualitative results evaluated on the evaluation set to further demonstrate the effectiveness of our approach. As illustrated in Fig. 15, our method consistently synthesizes high-fidelity PBR material maps across a diverse range of scenes. For more qualitative results, please visit https://tiptoehigherz.github.io/Material-Extraction/.

Real-World Examples. To further demonstrate the generalization capability of our method, we present additional qualitative results on real-world images in Fig. 18. These samples were captured using mobile devices in unconstrained environments, featuring diverse lighting conditions and viewing angles. Since ground-truth PBR maps are unavailable for such in-the-wild data, we focus on visual plausibility and quality.

8.4. Ablation Study on Depth Estimators

To validate the robustness of our method across different depth estimation priors, we conduct comparative experiments using three monocular depth estimation models: OmniData [10], Depth Anything v2 [49] and Marigold [20]. As demonstrated by the qualitative and quantitative results in Fig. 16 and Tab. 4, we observe that while the inherent performance of these depth estimators varies, their influence on the intermediate rectification results remains marginal. Consequently, the downstream PBR material extraction is largely unaffected.

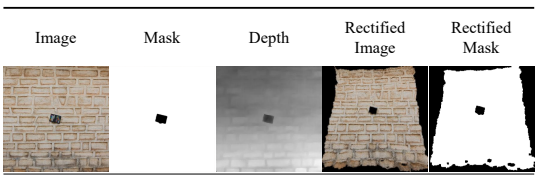

Figure 17. Limitation of geometry-based rectification on near-orthogonal views.

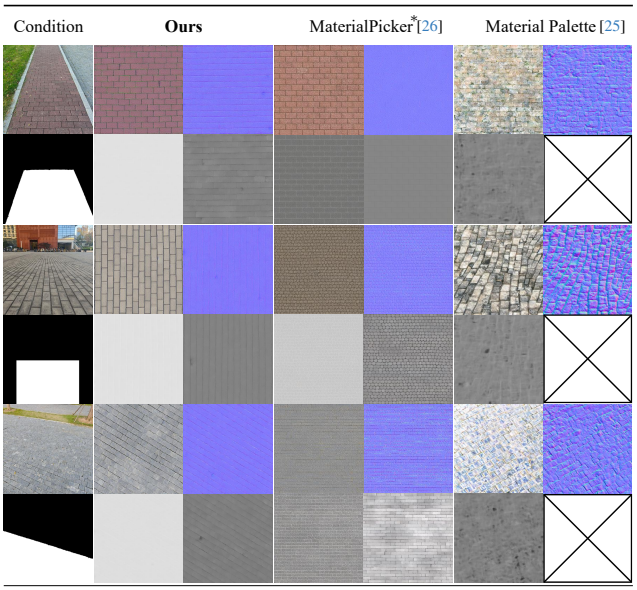

Figure 18. Qualitative results on real-world images. We present material extraction results from inputs captured by mobile devices in unconstrained environments.

9. Discussion

Potential of Textual Control. While textual prompts serve as a powerful control signal in many generative tasks, we deliberately exclude them from our current pipeline. This design choice stems from the observation that high-level semantic descriptions often lack the granularity required to precisely articulate high-frequency texture details and spatial structures. Language is inherently abstract, whereas material synthesis demands pixel-level fidelity. However, we believe that integrating Vision-Language Models (VLMs) offers a promising avenue for future exploration. Specifically, leveraging the multi-turn conversational capabilities of VLMs to iteratively refine materials aligns well with user intent, allowing for precise, step-by-step adjustments to the generated output.

Leveraging Pretrained Priors. Currently, employing image-based diffusion architectures for material generation often necessitates training from scratch, a process that is both time-consuming and computationally intensive. We argue that a crucial direction for future research is to develop methodologies that can effectively leverage pretrained diffusion models directly. Unlocking the potential of these off-the-shelf priors for material synthesis would significantly reduce training costs and democratize high-quality generation.