A Supplementary Material

A.1 Legacy Architecture: GAN-Based System

Before adopting a diffusion-based pipeline, DESPINA was implemented using a GAN architecture—specifically Pix2PixHD trained on segmented lunar terrain images. While this approach produced coarse outputs that roughly matched terrain structure, it failed to preserve fine detail and often introduced artifacts. The GAN system also required manually crafted label maps, which limited scalability and generalization. These limitations motivated a transition to a diffusion-based architecture, which demonstrated superior texture realism and structural coherence when paired with depth and soft-edge constraints.

A.2 Dataset Preparation

To establish a reliable dataset for DESPINA's architecture, we exclusively utilized Apollo mission imagery, which is publicly available and readily accessible. The decision to rely solely on Apollo data, rather than incorporating additional datasets like those from more recent missions like Chang'e 5, was driven by the need for image consistency in format and structure. Although other sources, such as Chang'e 5 and Yutu rover images, as seen in, offered potentially valuable data, they introduced significant challenges: either the datasets were fragmented and lacked cohesive organization, or access was heavily restricted by governing agencies, limiting their practical use. Specifically, data from Chinese lunar missions is tightly controlled, and usage typically requires affiliation with a Chinese academic institution or research organization. Despite inquiries, permissions to use these datasets could not be secured. However, small samples of the Chang'e dataset were utilized in the paper as a validation set, which proved useful.

To obtain the Apollo mission imagery, we developed a script to download all publicly available images from NASA's online archives. The script utilized Python's requests library to automate the retrieval of image files. Using a recursive script, images were downloaded sequentially from mission subdirectories. Metadata files accompanying the images were also retrieved where available. The resulting dataset comprised approximately 10,000 high-resolution images, which had to be sorted to determine project relevance. See **??** the incorporation of this sorting model into DESPINA's system.

A.2.1 Segmentation Process. We manually segmented 150 images using CVAT to fine-tune a DINOV2 instance segmentation model. This model segmented objects in scenes rather than every pixel, requiring post-processing with inpainting techniques to fill unclassified areas at class boundaries. While this automated approach saved time and resources, it introduced quality limitations that could be addressed in future work.

Most Apollo mission images include fiducial markers, which are small crosshair-like artifacts introduced for photogrammetric purposes. These markers interfere with existing computer vision techniques by introducing unnatural geometry that disrupts segmentation models. To address this, we developed a preprocessing pipeline for detecting and removing fiducial markers. The process involves identifying markers based on their geometric properties and applying inpainting techniques to remove them while preserving surrounding image features.

A.2.2 Data Augmentation. To enhance dataset diversity and mitigate overfitting, we applied controlled augmentation techniques:

- Brightness adjustments (±20%) to simulate varied lunar lighting conditions
- Contrast alterations (±15%) to adapt to different surface reflectance levels
- Horizontal flips (50% probability) to double training data for non-directional features

Small random rotations were initially implemented but later removed due to overfitting concerns.

A.2.3 BLIP Prompt Generation Example. We used BLIP [?] to generate structured text prompts from binary annotations for Apollo imagery. Below is an example showing the image and corresponding prompt:

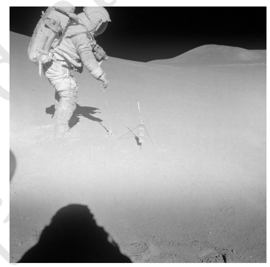


Figure 2: Apollo image used in BLIP prompt generation.

Generated Prompt: *A realistic image of the Moon's surface, from the perspective of a rover. A lunar landscape with hills in the background. A lunar crater is in view. A rover is present in the image. Dark shadows cast by the terrain. An astronaut is visible in the image.*

A.3 Hardware and Training Configuration

The SDXL LoRA and the DPT Fine-Tuning was trained using:

- Hardware: Six NVIDIA A100 GPUs (40GB memory) with CUDA 12.1.1
- Software: Python 3.11, PyTorch 2.1.2 (foss-2023a toolchain), TorchVision 0.16.0
- Dataset: 3,500 preprocessed images (2,900 training, 600 validation)
- Key parameters: Batch size 6, three discriminators, dropout enabled

A.4 Code Availability

The DEM-to-Depth pipeline, processed datasets, and supporting tools for dataset preparation will be made available at:

https://github.com/UH-DAIS/TIF-to-image
https://github.com/UH-DAIS/DESPINA_dataset

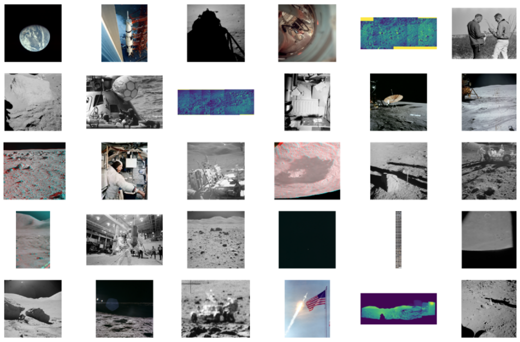


Figure 3: Several Examples of Images Easily Identified as *Not Horizon*. See ??.

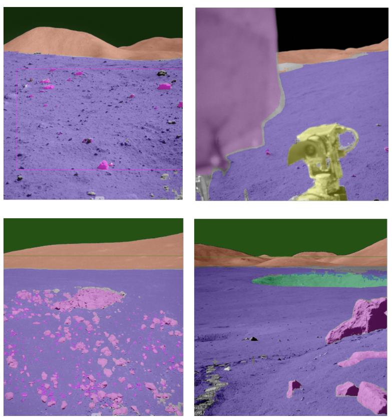


Figure 4: Examples of Manual Segmentation Through CVAT.

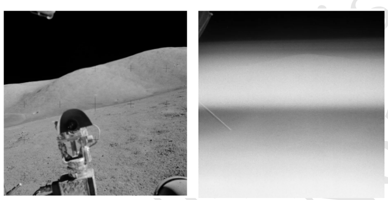
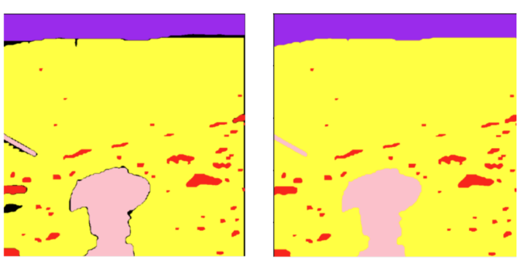
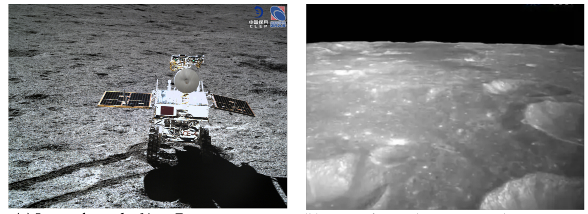


Figure 5: Examples of usable and unusable images from the Apollo dataset.



(a) Raw Segmented Image by DINOv2 (b) Image After Inpainting Process

Figure 6: Before and After the Inpainting Process on Raw DINOV2 Generated Segmentation Maps.



(a) Image from the Yutu Rover (b) Image from Chang'e 5 Landing Camera

Figure 7: Images of the Lunar Horizon from Non-Apollo Datasets. Sourced from the Yutu Rover (a) and Chang'e 5 (b).