

AIAXR

AI-Aided XR

Final Presentation

ESA AO/1-11249/22/D/AH

12/06/2024

ESOC, Robert-Bosch-Strase 5, 64293 Darmstadt,
Germany

Agenda

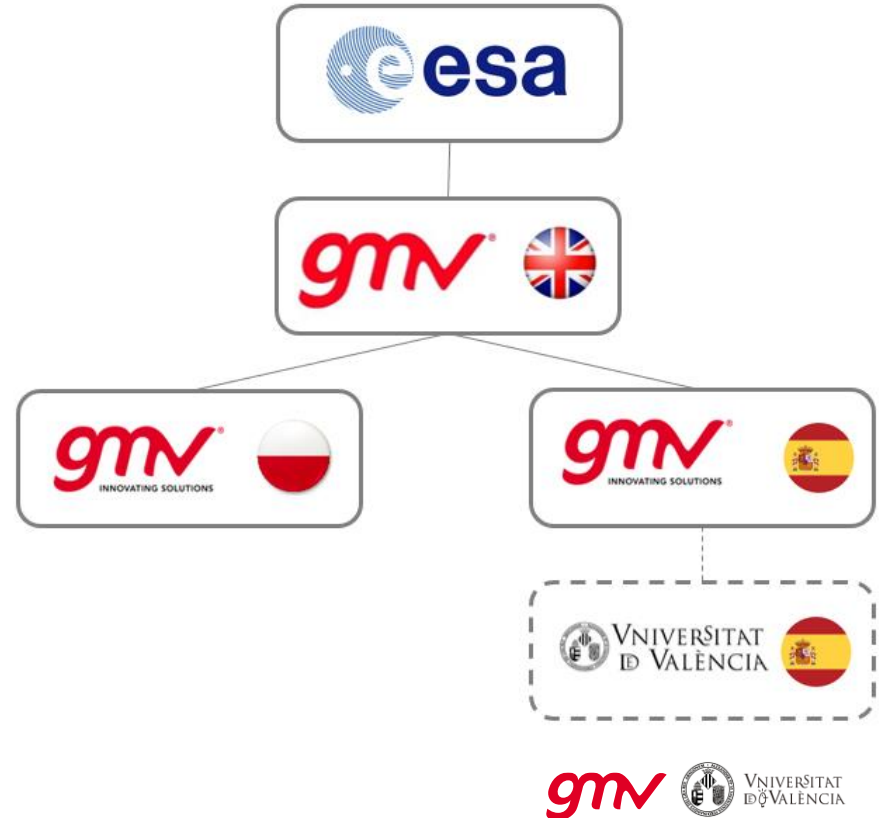
- **Project Overview**
- **Use Case Overview**
- **Framework Architecture and Design**
- **Use Case Design**
- **Demo**
- **Next Steps and Recommendations**
- **Summary**
- **Q&A**

Project Overview

Consortium

AI-Aided-XR: AI Aided VR/AR Applications and VR Aided ML

- **ESA TO: Evridiki Ntagiou**
- **GMV UK: Project Lead**
 - Project management
 - AI component
- **GMV SES: Subcontractor**
 - Support to XR prototype development
- **Univ. of Valencia, ARTEC: Expert Support**
 - XR component
 - Background and expertise in XR
- **GMV POL: Subcontractor**
 - Instrumental in providing and adapting the LHDAC dataset for training



Project Overview

Activity rationale

- **Artificial intelligence (AI), and especially Deep Learning (DL), are transformational technologies. However, their voracious appetite for data requires feeding them with huge volumes of high-quality data.**
- **Especially true for the space domain, where datasets are difficult to curate.**
- **The AI-Aided-XR activity is conceived to address the symbiotic relationship between Artificial Intelligence (AI) and eXtended Reality (XR) technologies, and how they can support space operations missions.**
- **The activity asks the question: how can AI and XR mutually benefit each other in a cyclical relationship? And how can this support space missions?**



Project overview

Scope & Objectives

- A state-of-the-art analysis was prepared on AI and classical Computer Vision and terrain generation technologies.
- User requirements were provided and collected through:
 - data from existing robotic mission simulations
 - analogue experiments
 - and workshops (**AR-VR**) with end users.
- A combined architecture of both the XR and AI applications were developed, allowing both to be triggered manually or simultaneously.
- The developed AI solutions were integrated with the **existing AINabler platform** to allow operators to easily trigger both use cases.

Framework Architecture and Design

Framework Architecture

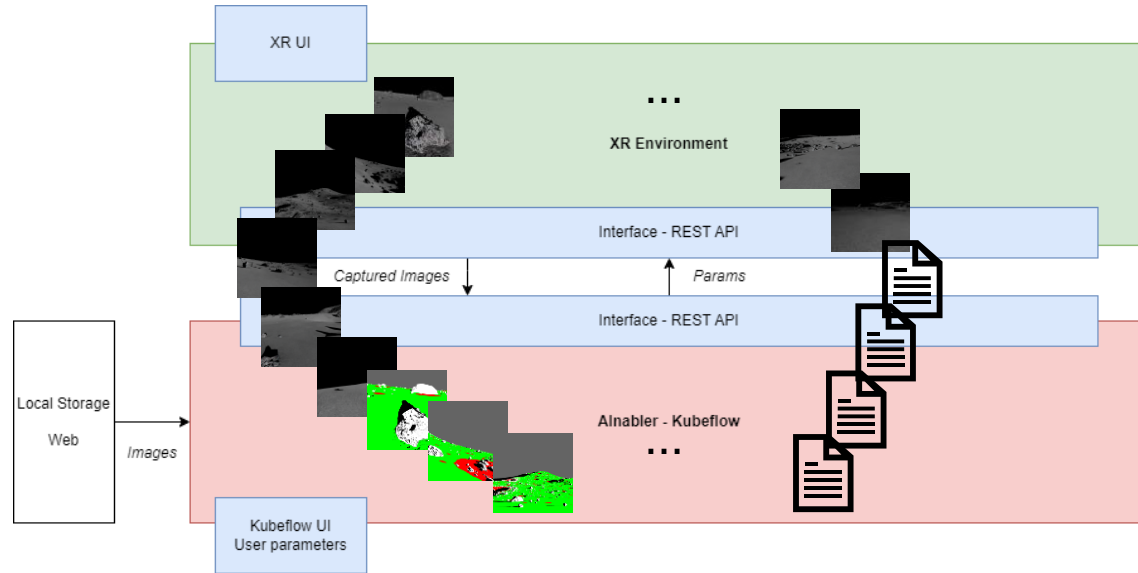
Objectives

1. The primary objective of the framework is to address the symbiotic relationship between AI and XR.
2. The framework explores the identified use cases in the space domain and how they serve each other for ESA applications.
3. The framework must be demonstrated on use cases which were previously selected.
4. A single framework should be established, integrating two smaller frameworks – the AI framework and the XR framework – that combine and interact seamlessly.

System architecture

Workflow

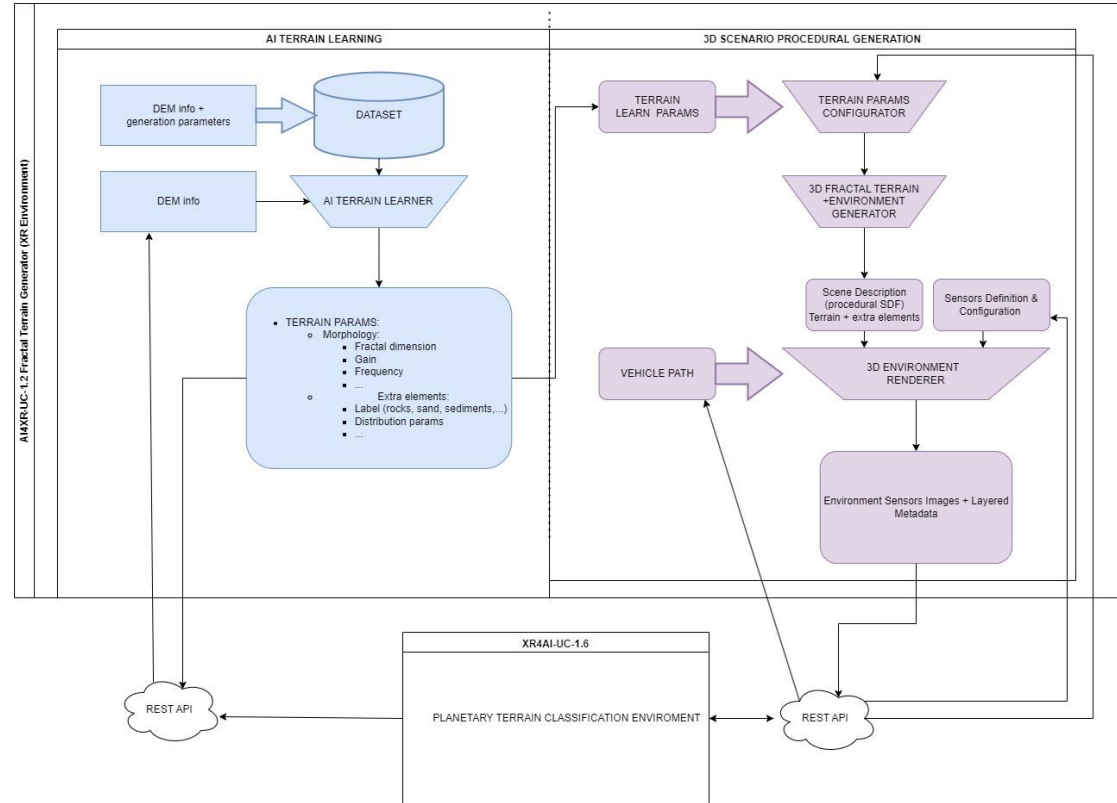
- Both AI and XR modules are connected seamlessly through a REST API.
- Each module will be stored on ESA hardware (AINabler) and communicate locally.
- A typical workflow is:
 1. The user accesses AINabler.
 2. The user triggers a pipeline with their desired parameters.
 3. The pipeline receives images from the XR component and performs segmentation and fine tuning if masks are included.
- Or:
 - The user accesses the XR environment.
 - The user uses the XR UI to generate an environment in real-time.



AI4XR - Framework architecture

AI4XR Framework Overview

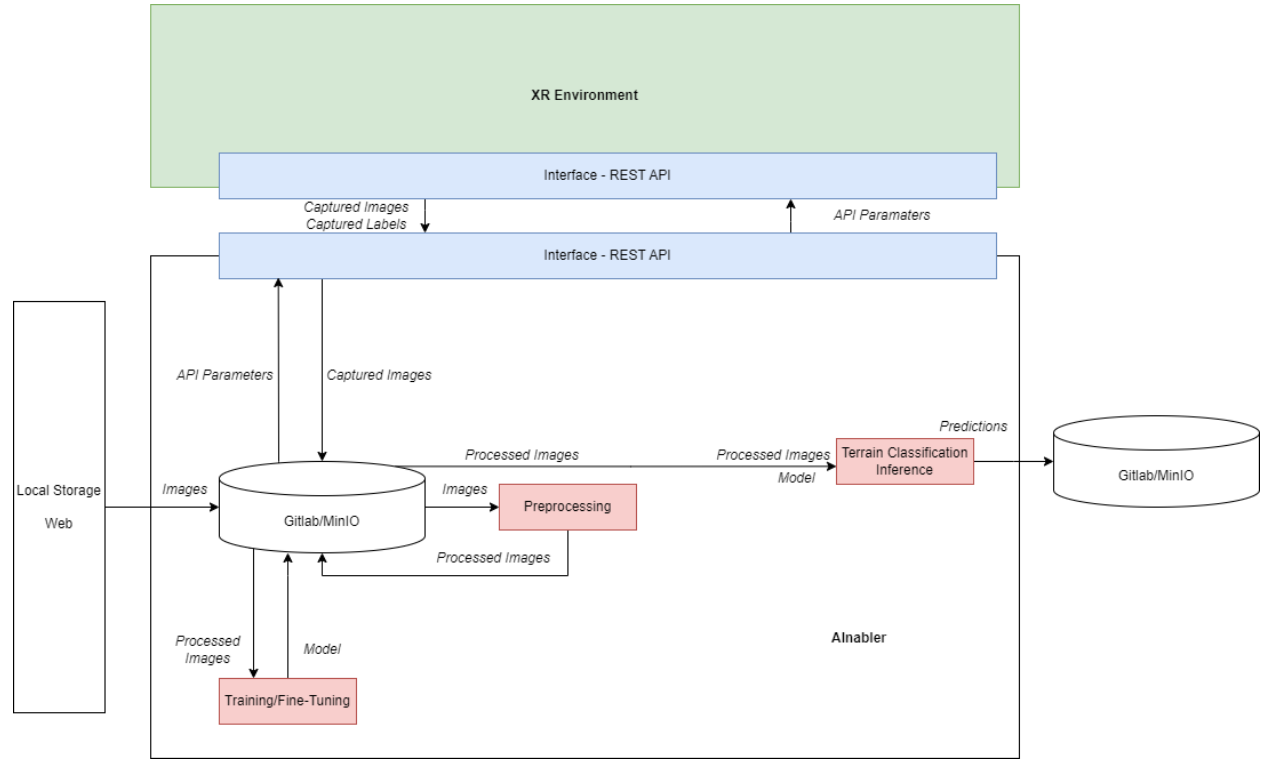
- 2 Main modules:
- **AI Terrain learning:** estimation of parameters from heightmaps to generate similar environments.
- **Terrain generator:** generation of realistic environments based on a set of input parameters.



XR4AI - Framework architecture

XR4AI Framework Overview

- A change from the ViBEKO framework – **GitLab now stores important objects like models for better permanence**
- Parameters set by the user are sent to the XR component
- Images and/or labels are sent from the XR environment



Use Case: AI4XR

AI4XR

Parameters

Terrain description

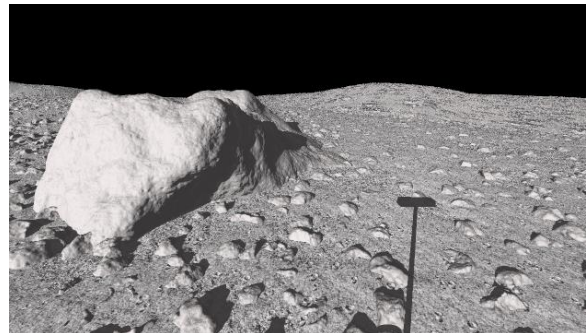
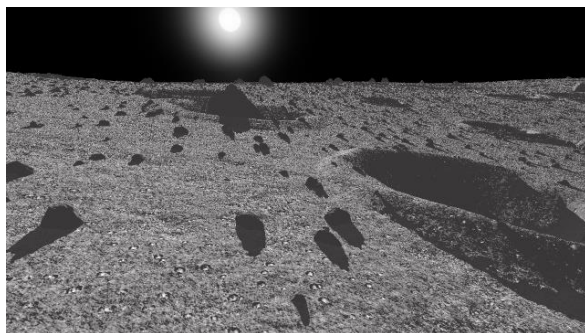
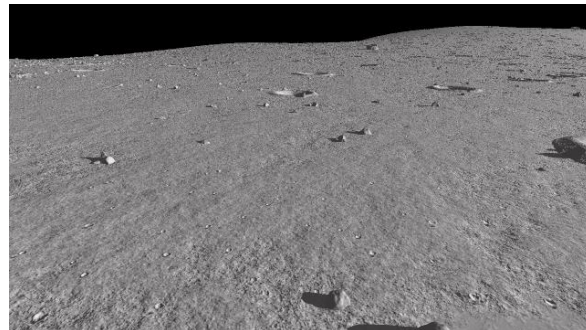
- Mountains
- Ground
- Craters (big and small)
- Rocks (big and small)

Light description

- Direction
- Elevation
- Intensity

Vehicle description

- Position, Rotation
- Number of cameras
- Model
- Description of each camera
- Position, rotation, resolution and distortion



AI4XR

Parameters estimation

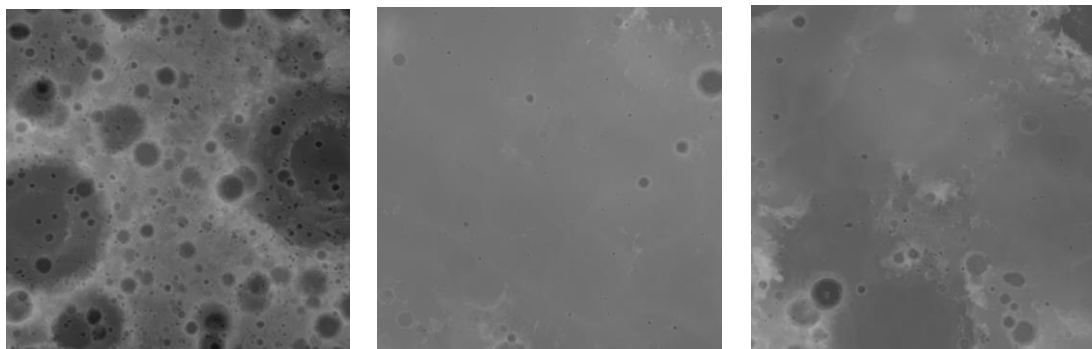
Terrain Learner

- Parameter estimation from heightmaps
 - Mountains
 - Big rocks
 - Big craters
- Dataset of synthetic terrain elevation maps for training
- 2 approaches:
 - Direct parameter values estimation
 - Terrain classification

Synthetic digital elevation maps:



Real digital elevation maps:



AI4XR

Parameters editor

User interface for parameter definition

- Terrain
- Light
- Vehicle
- Camera

Vehicle navigation

- Position and rotation
- Navigation with keys

Rendering type

- Color image
- Label image
- Distance image
- Digital elevation map
- Digital elevation map labels

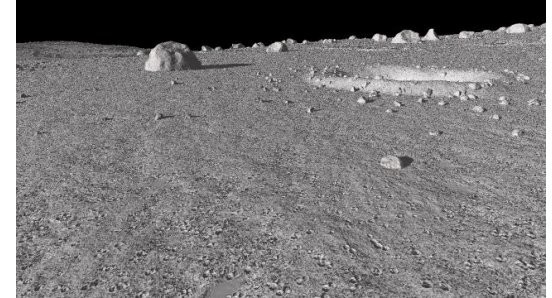
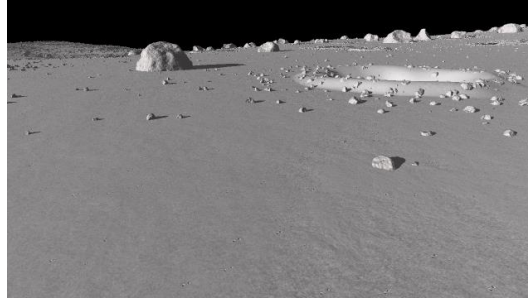


AI4XR

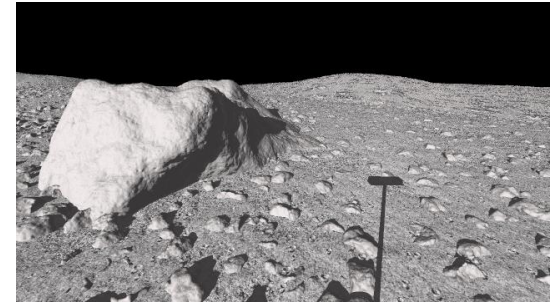
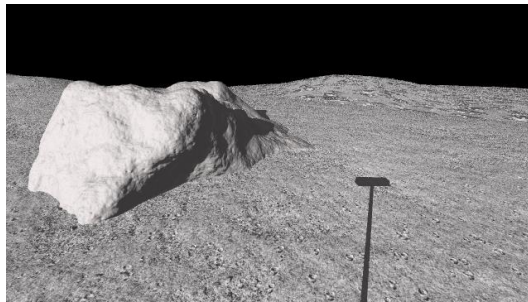
Terrain renderer

- Near realtime rendering
- **Based on parameters and random distributions**
- Easy to generate big datasets
- 5 rendering types:
 - Color image
 - Label image
 - Distance image
 - Digital elevation map
 - Digital elevation map labels
- Ray tracing and fractal functions, no 3D meshes
- Accurate lighting and shadows including regolith scattering

Different grounds



Different little rocks

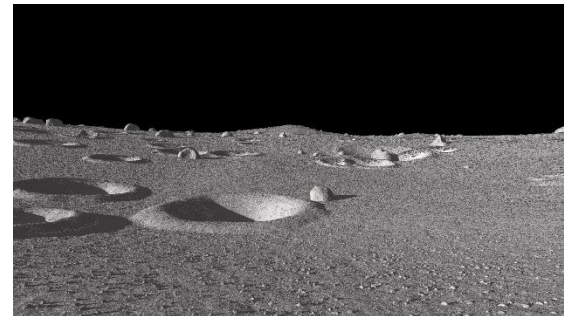
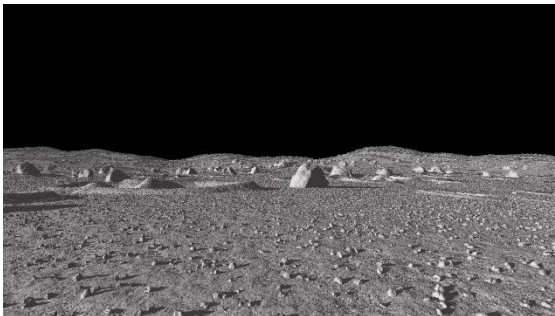
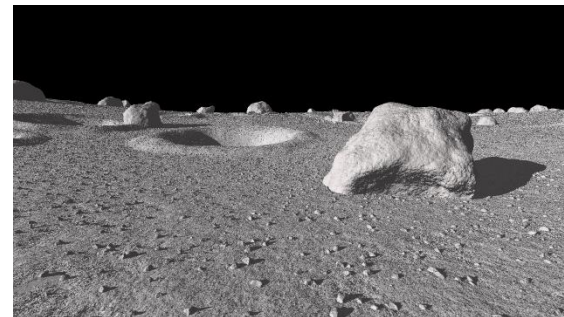
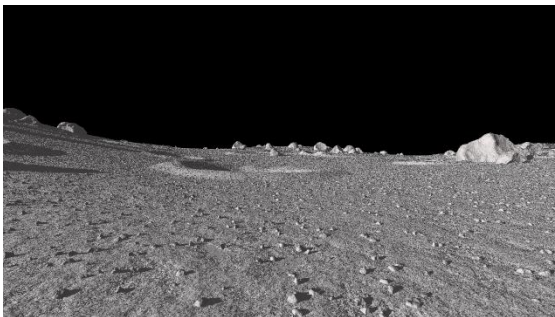


AI4XR

Terrain renderer

- Near realtime rendering
- Based on parameters and random distributions
- **Easy to generate big datasets**
- 5 rendering types:
 - Color image
 - Label image
 - Distance image
 - Digital elevation map
 - Digital elevation map labels
- Ray tracing and fractal functions, no 3D meshes
- Accurate lighting and shadows including regolith scattering

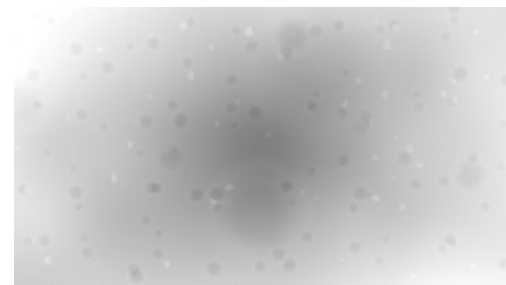
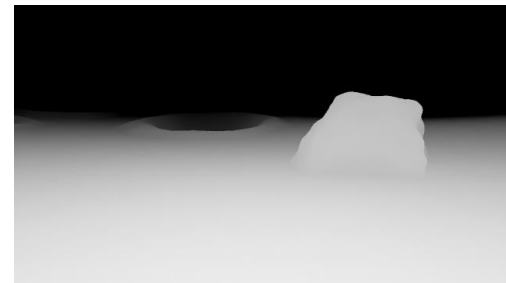
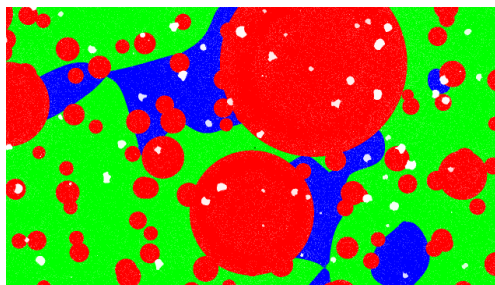
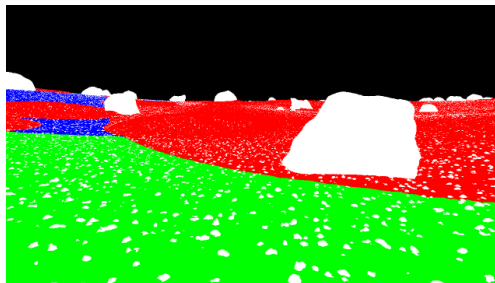
Different random seeds



AI4XR

Terrain renderer

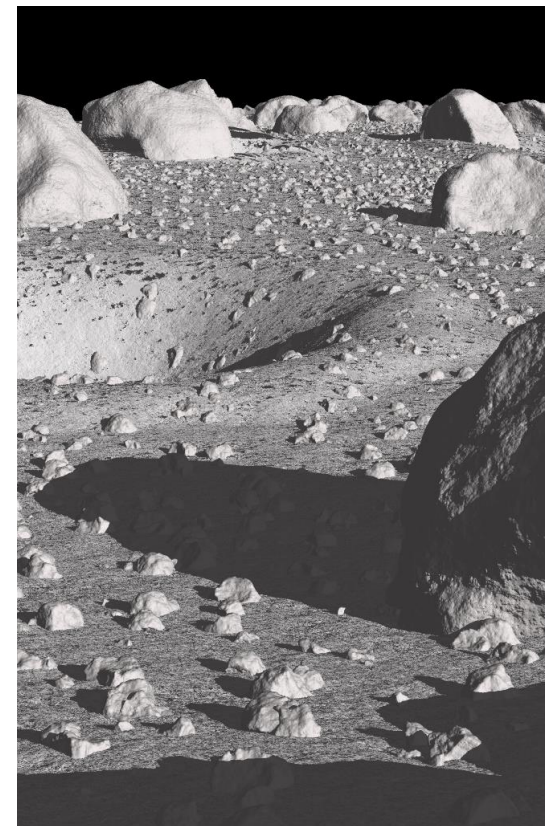
- Near realtime rendering
- Based on parameters and random distributions
- Easy to generate big datasets
- **5 rendering types:**
 - **Color image**
 - **Label image**
 - **Distance image**
 - **Digital elevation map**
 - **Digital elevation map labels**
- Ray tracing and fractal functions, no 3D meshes
- Accurate lighting and shadows including regolith scattering



AI4XR

Terrain renderer

- Near realtime rendering
- Based on parameters and random distributions
- Easy to generate big datasets
- 5 rendering types:
 - Color image
 - Label image
 - Distance image
 - Digital elevation map
 - Digital elevation map labels
- **Ray tracing and fractal functions, no 3D meshes**
- **Accurate lighting and shadows including regolith scattering**



Use Case: XR4AI

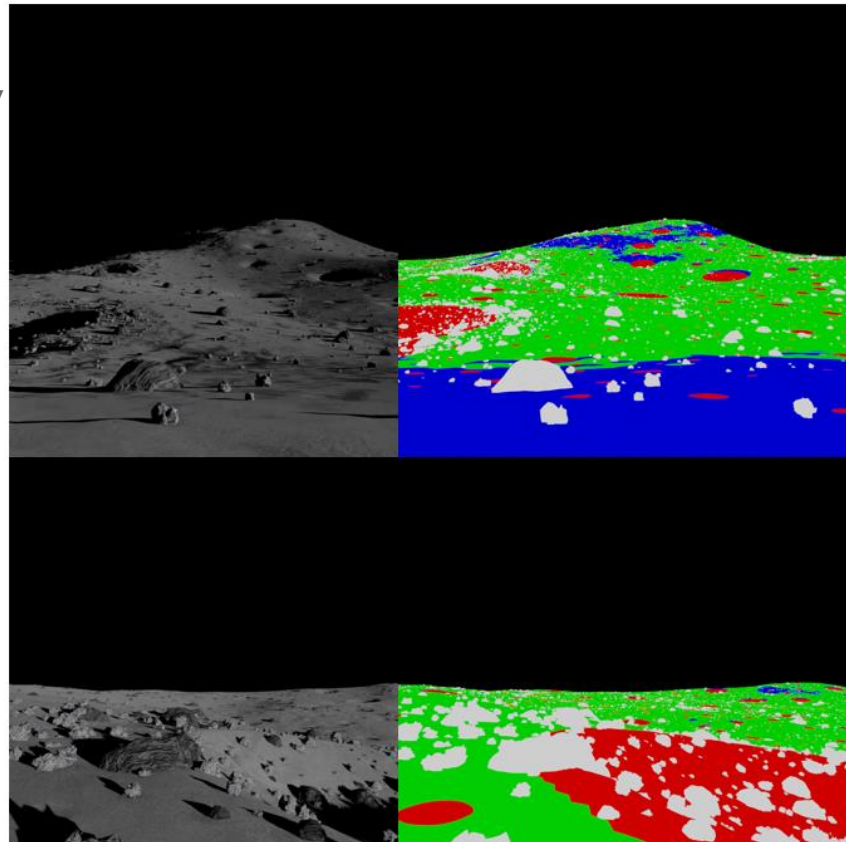
XR4AI

Datasets

- Data availability of rover-perspective lunar imagery is scarce → **even less with labels/ground truth!!**
- This training dataset is provided by **GMV POL** as an extension to their ESA LHDAC activity
- 3000 images and masks
- This dataset is procedurally generated in Blender
- Provides operators with automatically-labelled segmentation masks and the ability to create new classes (slopes, caves)

Flat area
Slope (10 degrees)

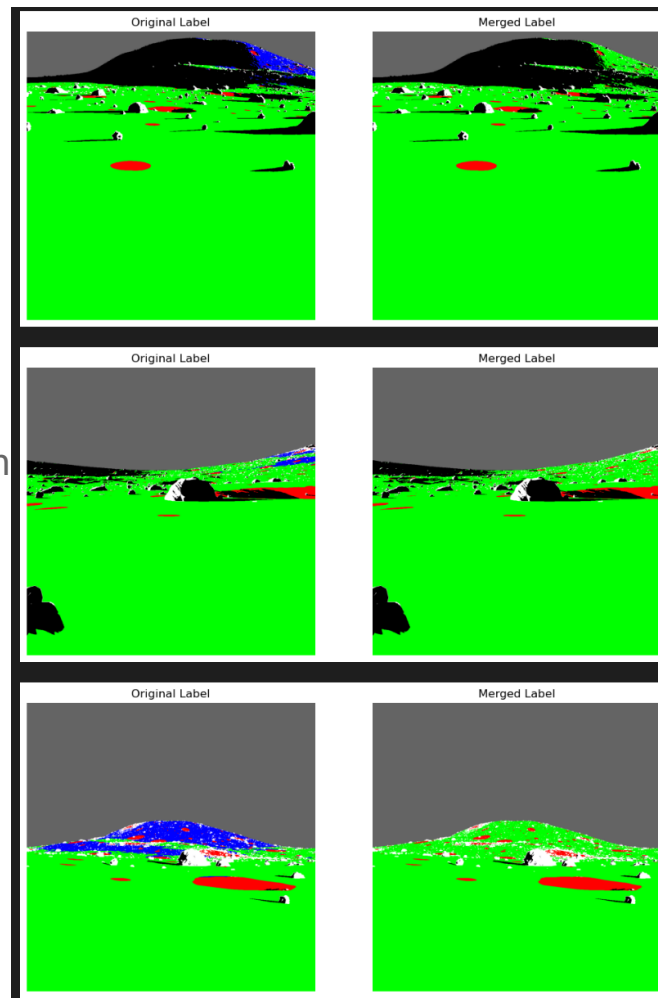
Crater
Sky



XR4AI

Pre-Processing

- Training on the base LHDAC resulted in a high accuracy of 90% (model selection and hyperparameter tuning is discussed later).
- However, slopes were far too ambiguous. The DeepLabV3+ model was unable to discern between slopes and flat areas due to the visual characteristics of the terrain.
- **Solution:** Merge the slope and flat area into a single class, called: **Terrain**

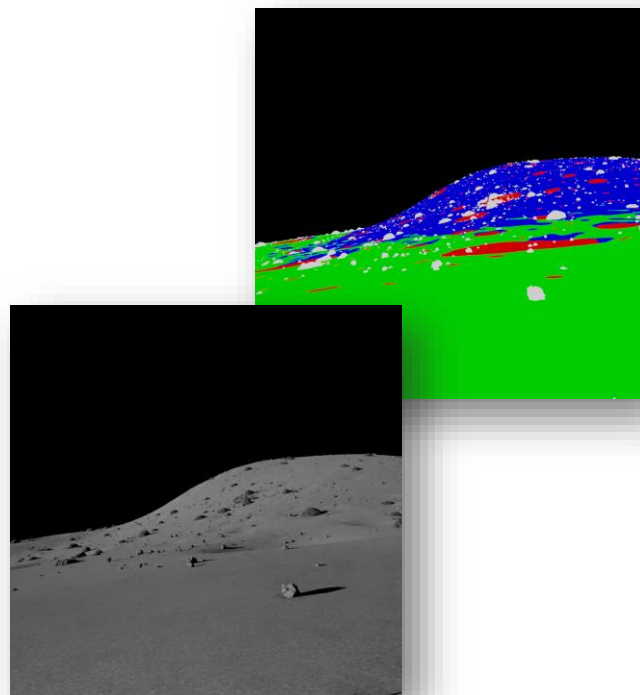


XR4AI

Pre-Processing

- With the classes merged, we pre-process the input images and masks.
- The following techniques are used for preparing the input images:
 - **Normalization**
 - **Centralizing**
- Masks are assigned indices so that they can be **one-hot encoded**.

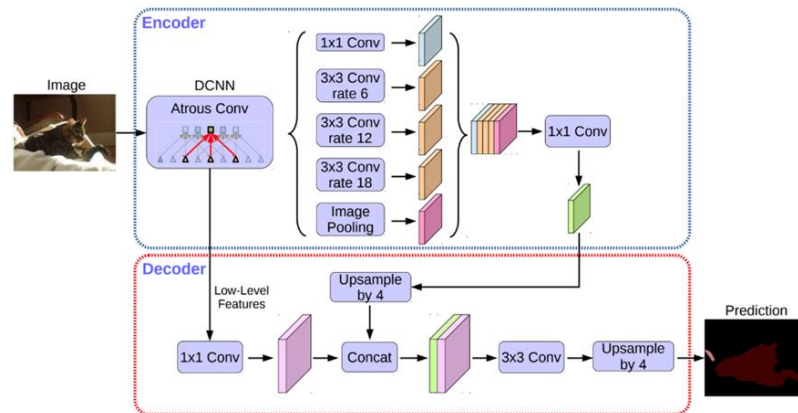
- **Note: data augmentation was experimented with but led to negative results.**



XR4AI

Model Overview

- DeepLabv3+ is a state-of-the-art model for semantic image segmentation, which classifies each pixel in an image into a category.
- DeepLabv3+ builds on the successes of its predecessors, DeepLabv1, DeepLabv2, and DeepLabv3, introducing several enhancements for more accurate and efficient segmentation.
- **Key features:**
 - **Atrous Convolution**
 - Increases the field of view without increasing the number of parameters.
 - **Encoder-Decoder**
 - The encoder captures multi-scale contextual information using atrous convolutions, while the decoder refines the segmentation to produce high-resolution, detailed maps.



XR4AI

Hyperparameter tuning

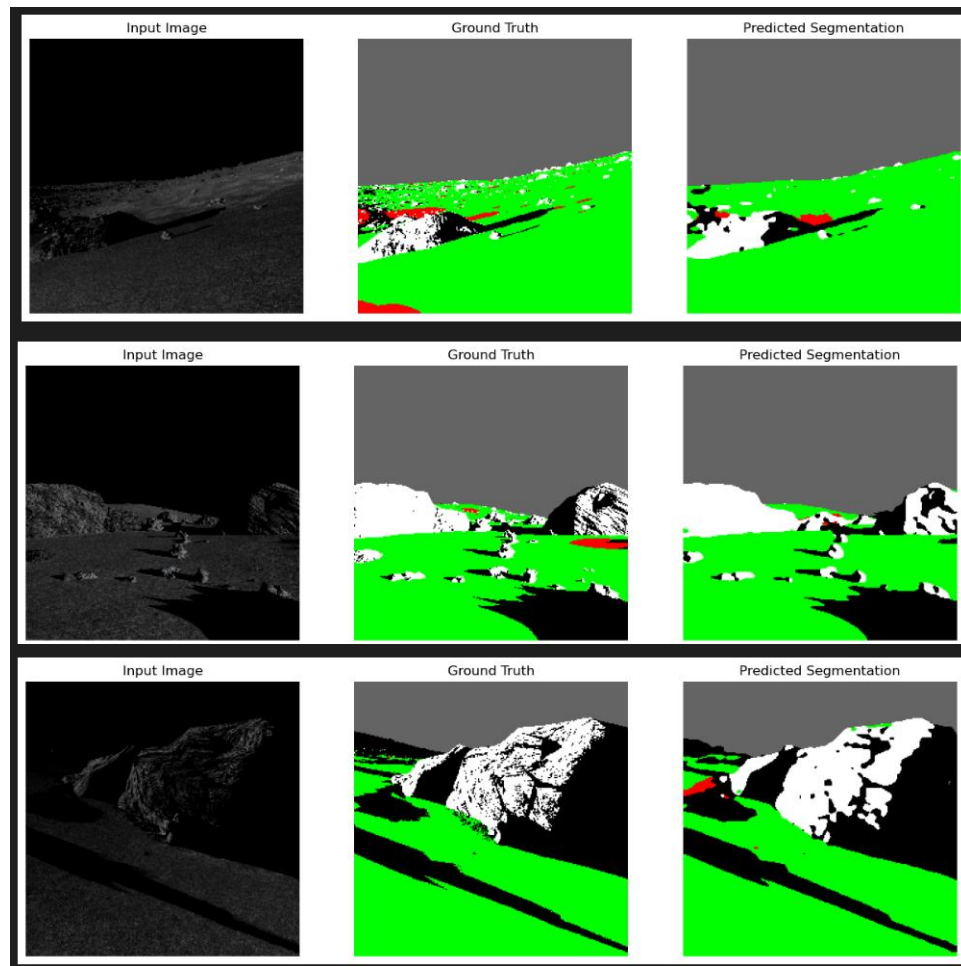
- Training on the base LHDAC dataset was done with hyperparameter tuning using a matrix consisting of:
 - *Resnet backend sizes*
 - *Epochs*
 - *Early stopping*
 - *Learning rates*
- 9 models were trained using this matrix.
- **model_epochs200_patience12_lr0.0001_resnet101** is the best performing. While it doesn't have the lowest loss, the high accuracy and mean IoU suggest it performs better.

Mode Name	Test Loss	Test Accuracy	Test Mean IoU
model_epochs200_patience12_lr0.0001_resnet101	0.206139	0.942582	0.900285
model_epochs150_patience10_lr0.001_resnet50	0.205519	0.935451	0.887122
model_epochs250_patience15_lr0.0001_resnet101	0.220385	0.934568	0.888755
model_epochs200_patience15_lr0.0005_resnet50	0.188931	0.937943	0.893502
model_epochs150_patience12_lr0.0005_resnet101	0.196071	0.939475	0.895178
model_epochs250_patience10_lr0.0005_resnet101	0.191520	0.941333	0.899186
model_epochs200_patience12_lr0.001_resnet101	0.189564	0.937698	0.891672
model_epochs150_patience15_lr0.0001_resnet50	0.202210	0.939824	0.897328
model_epochs200_patience10_lr0.0005_resnet101	0.185155	0.939713	0.895702
model_epochs250_patience12_lr0.001_resnet101	0.200027	0.936459	0.890046

XR4AI

Results

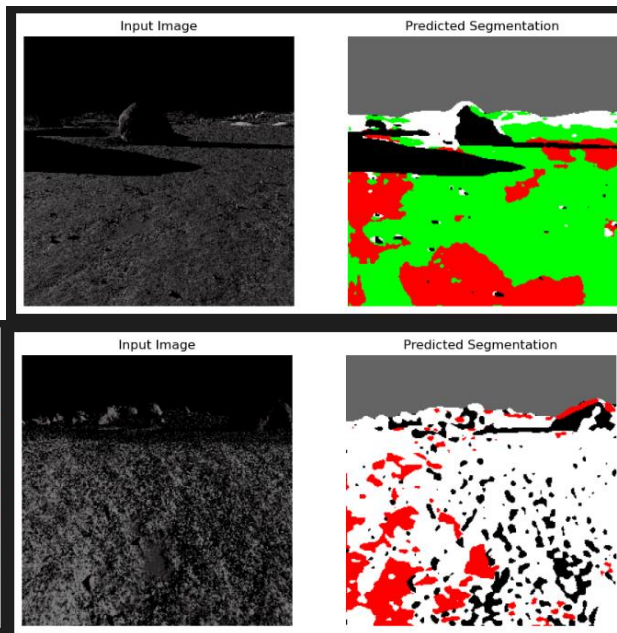
- Inference against images in the test dataset returned good predictions.
- Merging slopes and flat area into one terrain class increased the training scope for the lunar terrain.
- **Craters** can be problematic due to the characteristics of the blender LHAC dataset – some are flat or too close to the texture of the terrain class



XR4AI

Results

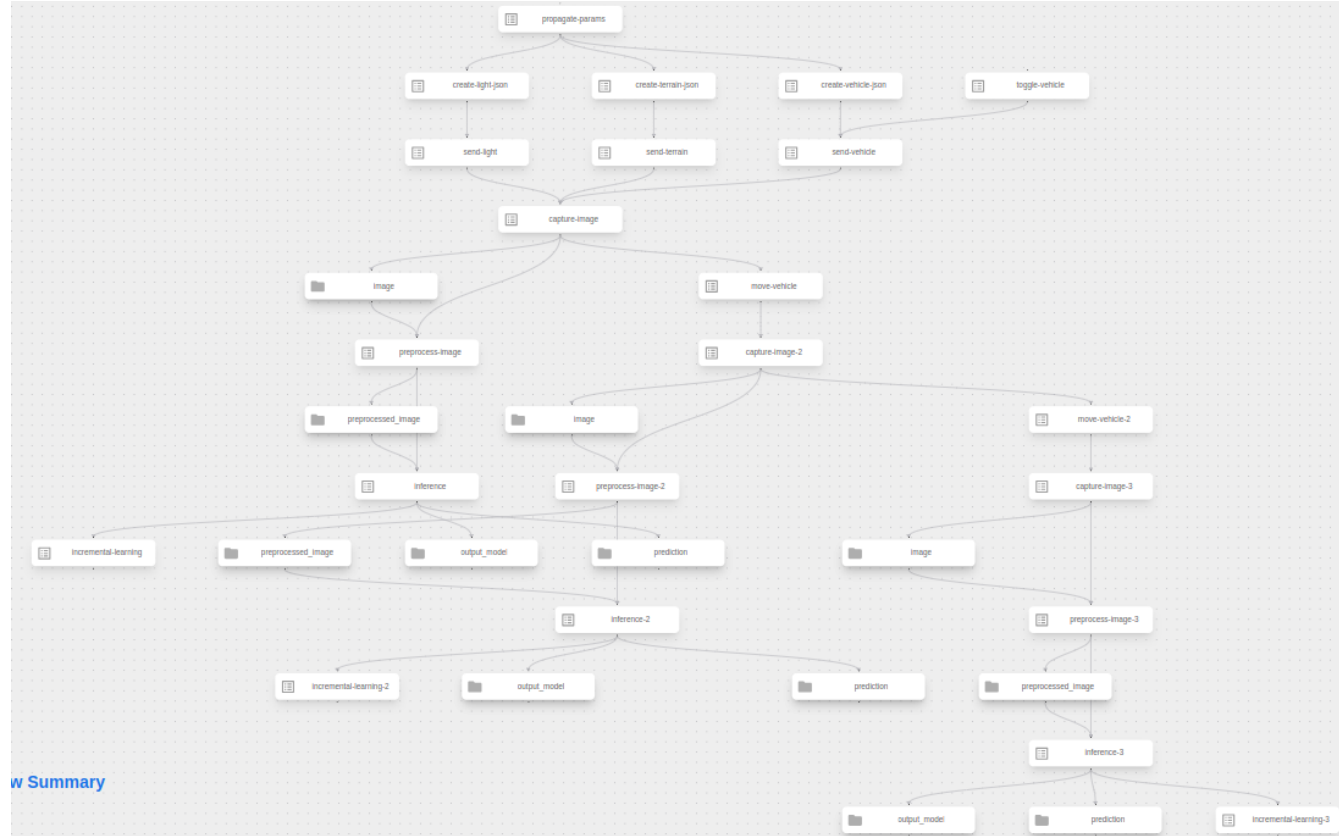
- However, when introduced to totally new, unseen data from the XR environment, predictions were very poor.



XR4AI

AINabler - Kubeflow

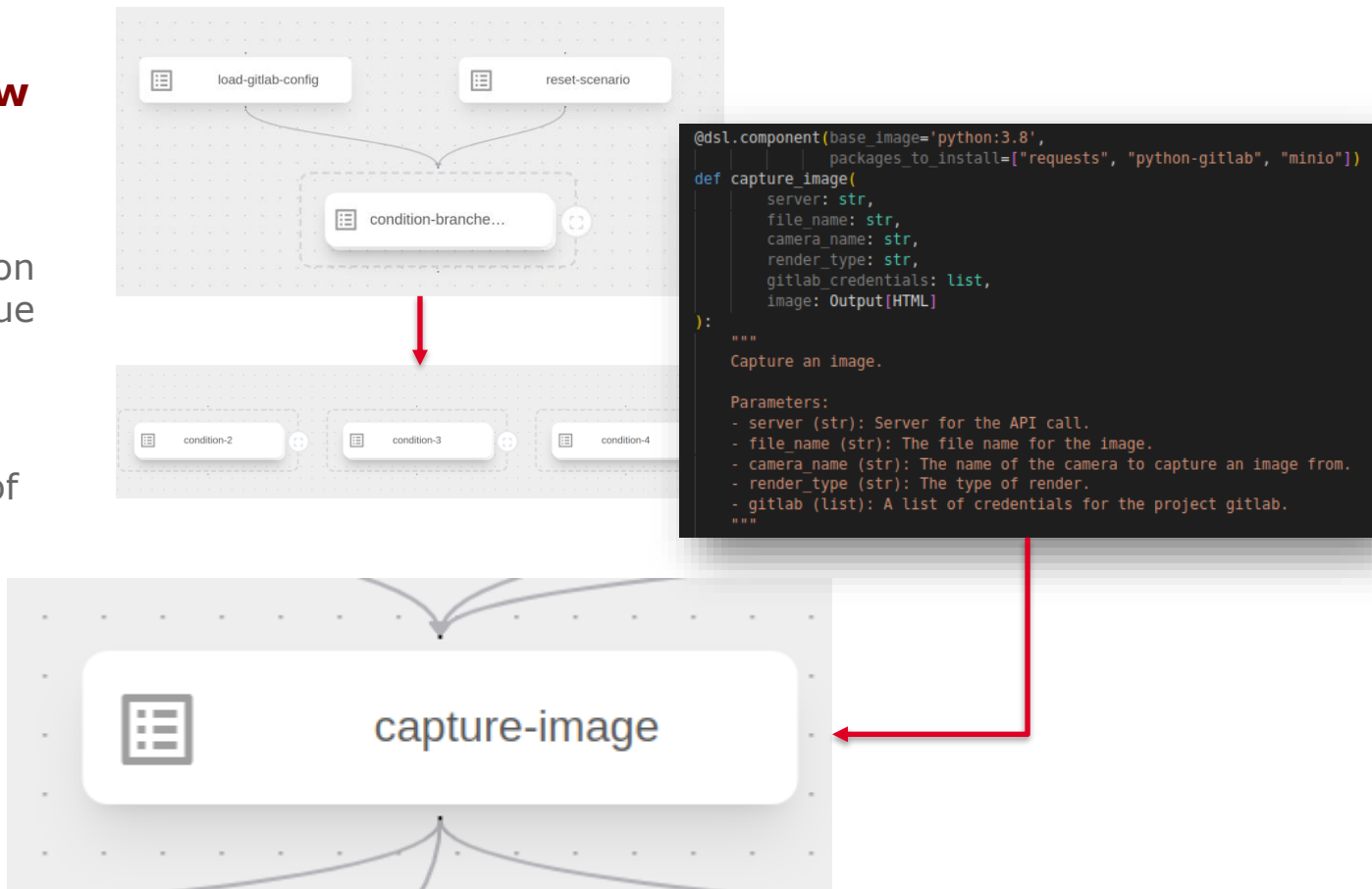
- AINabler is ESA's MLOps platform.
- GMV have worked extensively with ESA with this AINabler platform and have successfully deployed one previous project on it: ViBEKO.
- AINabler is built on top of Kubeflow, a Kubernetes platform for machine learning.



XR4AI

AINabler - Kubeflow

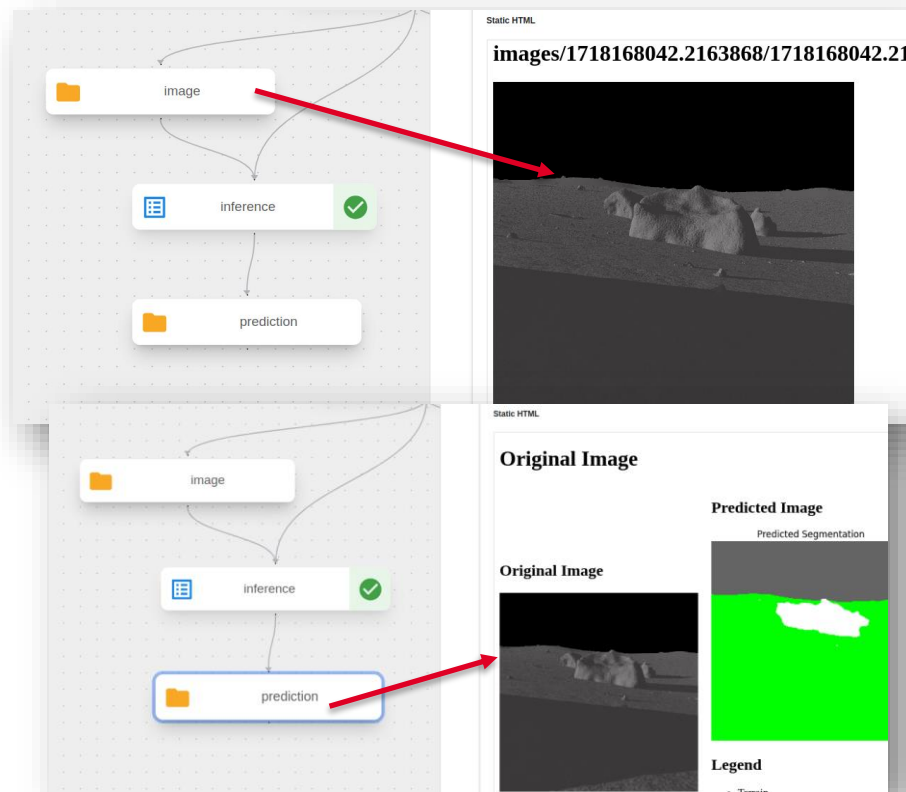
- **Conditionals**
 - Allow pipelines to take routes based on conditions being true or false.
- **Steps**
 - Essentially blocks of reusable Python code, to be used several times in pipelines.



XR4AI

AINabler - Kubeflow

- **User input**
 - Input variables on each pipeline run.
- **Visualization**
 - Visualize your predictions and images in the graph!

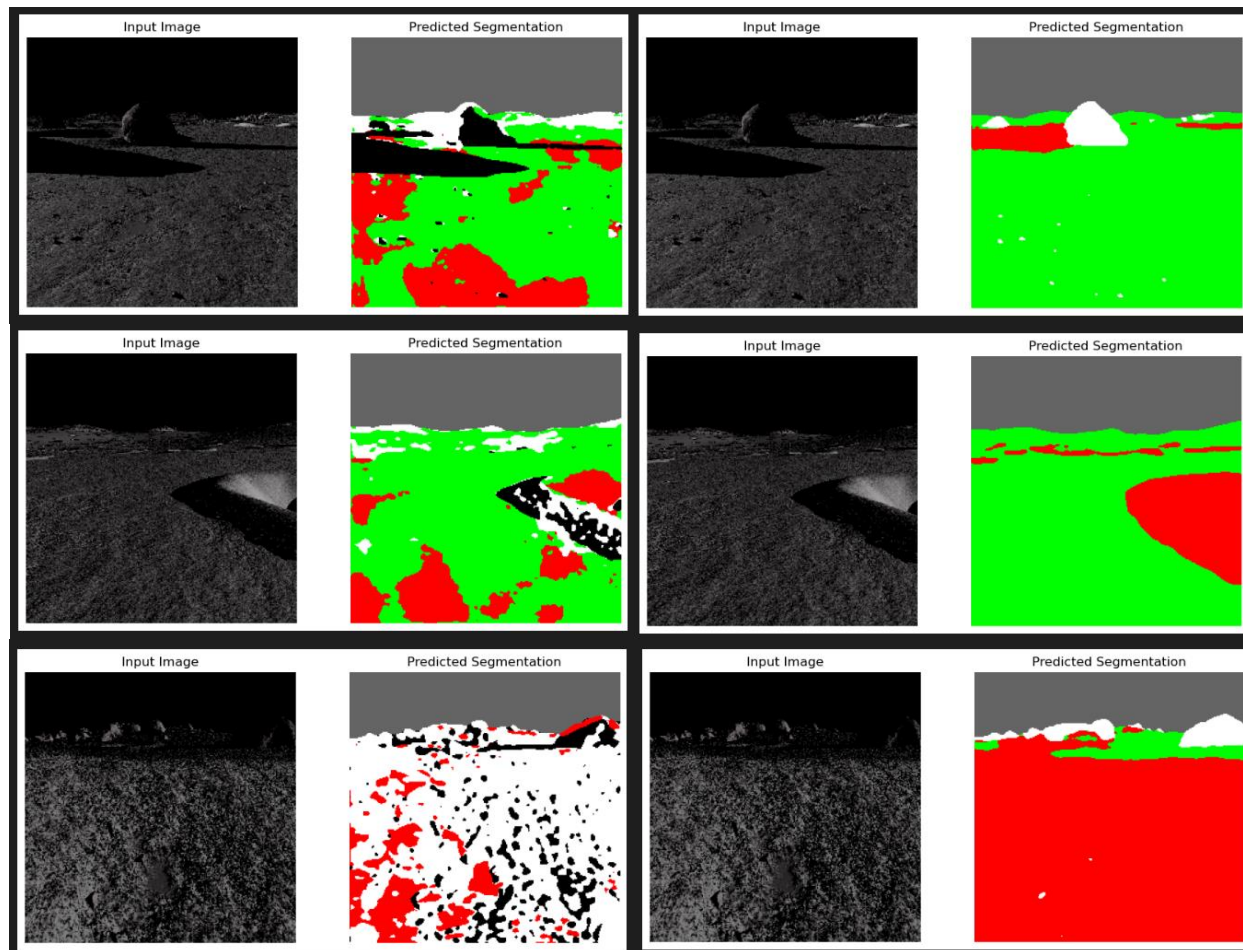


XR4AI

Results

- With these AINabler pipelines, we now have n new images and labels to synthetically generate and train our base LHDAC model on.

For this experiment, I only used 200 images!



Demo



admin (Owner) ▾



Dashboard

Activity

Quick shortcuts

- Upload a pipeline**
Pipelines
- View all pipeline runs**
Pipelines
- View Katib Experiments**

Recent Notebooks

No Notebooks in namespace admin

Recent Pipelines

- [Tutorial] DSL - Control structures**
Created 20/02/2024, 02:19:13
- [Tutorial] Data passing in python components**
Created 20/02/2024, 02:19:12

Recent Pipeline Runs

- Run of aixr_version_at_2024-06-12T02:04:44.275...**
Created 12/06/2024, 03:05:05
- Run of aixr_version_at_2024-06-12T01:37:12.583...**
Created 12/06/2024, 02:43:15
- Run of aixr_version_at_2024-06-12T01:15:26.869...**
Created 12/06/2024, 02:15:49
- Run of aixr_version_at_2024-06-12T00:50:51.194...**
Created 12/06/2024, 01:51:30
- Run of aixr_version_at_2024-06-12T00:01:04.455...**
Created 12/06/2024, 01:01:39

Documentation

- Getting started with Charmed Kubeflow**
Learn how to get started with Kubeflow
- Microk8s for Kubeflow**
Quickly get Kubeflow running locally on native hypervisors
- Requirements for Kubeflow**
Get more detailed information about using Kubeflow and its components

Home

Notebooks

Experiments (AutoML)

Experiments (KFP)

Pipelines

Runs

Recurring Runs

Volumes

Tensorboards

Manage Contributors

GitHub

Documentation

XR Demo

Next Steps

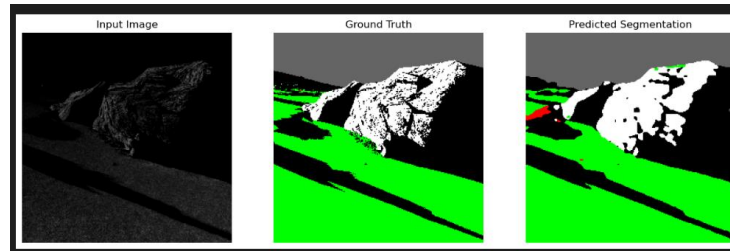
Next Steps Recommendations

- **GMV are hosting a SPINtern who will experiment with SPARTAN VO and the AIAXR tool**
- **Further extension to the AINabler platform – lack of GPU prevents operators triggering training pipelines**
 - Additional use cases for other space operations using AI – not explicitly computer vision-based
- **A deeper look into incremental learning, which was not implemented and would allow operators to introduce new, unseen classes**
- **To improve the large terrain elements in the XR environment such as very big craters**
- **Rendering time for the XR environment could be improved, and another parameter estimation algorithm could be investigated for the terrain learner**
- **ESA thoughts?**

Summary

Summary

- Two use cases were implemented on one combined framework (AIAXR).
- AI4XR, an XR environment hosted on ESA hardware which can render environments using fractals and parameters sent by the AI component.
- XR4AI, a planetary terrain classification use case built on ESA's AINabler platform for space operation missions. Demonstrates the communication capabilities between the XR framework, and the visualization capabilities for AI using computer vision.
- Further demonstrated AINabler's capabilities for space operations. However, lack of GPU prevented the team from training on the platform.
- Scope for follow-on activities making use of either AI4XR or XR4AI: synthetic data generation is sorely needed, and more AI and operations demonstrators on AINabler.



Questions



gmv.com

Thank you