CloudShadingAI: the technical details

Training and the Cloud Shadow Service use Python (https://www.python.org/).

1. Training (model development)

The training is used exclusively to develop a model for predicting movement vectors (cloud motion).

Input data

- Cloud images of Europe from the MSG-3 satellite, prepared by the project partner MetGIS
- Meteorological data, prepared by MetGIS (primarily wind data) (15-minute resolution)
- The cloud images contain encoded information on: (15-minute resolution)
    - CTP (Cloud Top Pressure) → from which cloud height can be derived
    - COT (Cloud Optical Thickness) → measure of optical depth / light transmissivity

Data processing

- The input data are split into 18 layers
- For training, the layers are merged, which improves the robustness of motion detection
- The wind data are physically transformed:
    - Conversion from m/s to pixels per 15 minutes
    - Accounting for the curvature of the Earth

Generating training labels (movement vectors)

- To determine the actual cloud motion between two points in time, OpenCV is used (https://opencv.org).
- Using SIFT (https://docs.opencv.org/4.x/da/df5/tutorial_py_sift_intro.html) keypoint descriptors, visual features are extracted from successive cloud images
- Descriptor distance serves as a measure of visual similarity between two cloud pixels
- Wind data constrain the search space of possible matches so that only physically plausible movements are considered
- Matching is evaluated based on:
    - Distance to the expected wind-driven movement
    - Descriptor distance (visual similarity)
- Result: movement vectors (MVs) as ground truth for training

Model training

- The data are organised into grids and grouped according to similar topographical properties
- A separate model is trained for each group
    - Input: Input: transformed wind data
    - Target: movement vectors
- Training is performed using PyTorch (https://pytorch.org/)
- Model architecture:
    - Linear neural network with 4 layers
- Training is carried out on the Leonardo supercomputer
    - 1650 core hours used → approx. 250 hours with 8 cores
- Training over five epochs (no significant improvement thereafter)
- Result: models that can predict movement vectors based on meteorological data (wind)

2. Cloud prediction & shadow service

The service uses the trained model for future prediction and shadow calculation. Forecasts up to six hours into the future are possible.

Input data in the service

- Current cloud data:
    - CTP (Cloud Top Pressure)
    - COT (Cloud Optical Thickness)
- Forecast future weather data (in particular wind fields)
- The actual cloud thickness is not directly known
- It is estimated based on surrounding cloud structures

Preprocessing in the service

- The data are preprocessed again (especially wind data → conversion into pixel coordinates)
- The data are divided into groups – analogous to the structure used in training

Motion ector prediction

- The trained model calculates movement vectors for future time steps
- This is followed by movement aggregation
- The vectors are not simply summed, but combined in a physically consistent way to produce accurate multi-step predictions
- The aggregated vectors are applied to the cloud image

Cloud shadow calculation

- Cloud height (in metres) is derived from CTP
- Shadow calculation is performed using ray tracing
    - COT/light transmissivity is taken into account for clouds
- Numba (https://numba.pydata.org) is used for performance optimisation
- The output consists of GeoTIFF files (cloud height and shadows)
- Delivery to the frontend is handled via MapServer (https://mapserver.org)

3. Terrain Shadows Service

This service calculates terrain shadows based on global elevation data.

Data

- Terrain data from ASTER Global Digital Elevation Model (GDEM) Version 3 are used
- The global DEM data were downloaded and split into tiles, then stored in a custom optimised database format
- Tiles are loaded on demand via mmap

Implementation

- The service is implemented in Zig (https://ziglang.org)
- Calculations are performed using ray tracing
- Only tiles along the solar direction are loaded; their number is limited for performance reasons
- Execution runs on the GPU via Vulkan compute shaders
- The system is optimised for UMA architectures, allowing CPU and iGPU to share the same memory
- Shadow calculations are performed on the server and the shadows are delivered to the frontend
- The simulation is extremely precise (significantly more accurate than comparable systems on the market), calculating shadows per pixel and taking into account many factors such as the curvature of the Earth and atmospheric light refraction

For questions about the model or product enquiries, please contact office@luxactive.com