Imaging simulation tool for small solar system objects

Antti Penttilä
University of Helsinki

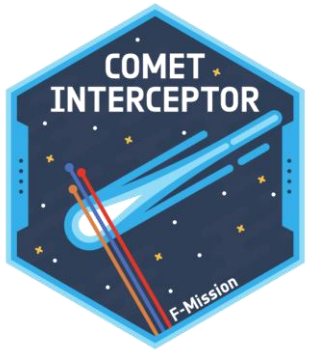
With contributions by Meri Kolehmainen, Anne Keski-Vakkuri, Leonardo Negri, Noora Halla-aho, Joel Jääskeläinen, Mario F. Palos, Tomas Kohout (UTU), Antti Näsilä (VTT), Richard Cole (VTT)

Nordic-Baltic Astronomy Days 2026, Turku

Motivation

- Space mission operations need to be planned beforehand
- This invokes questions like:
  - What do we see with camera *XXX* in mission phase *YYY* when observing geometry is *ZZZ*?
  - What is the quality of the images with detector *XXX* in geometry *YYY* and with integration time *ZZZ*
  - What is the quality of scientific products from the image with certain detector/geometry/integration etc.

- We are especially interested in these questions in relation with VTT Fabry-Perót interferometric hyperspectral camera on ESA space missions
  - **Hera**/MILANI smallsat, ASPECT instrument targeting Didymos binary asteroid system
  - **Comet Interceptor** MIRMIS instrument with yet undecided target, hopefully new long-period comet



Goals for the tool on what the instrument is seeing

- Physically “correct” and traceable imaging simulations of surfaces (asteroid, comet nucleus) and particle and gas volumes (cometary coma)
- Correct physical units such as radiance
- Description of surface scattering with photometric functions and their parameters typical for planetary science
  - Disk functions: Lommel-Seeliger, Lambert
  - Phase functions: polynomial (ROLO), exponential, linear-magnitude
- Description of particle/gas volumes by particle or molecule density, particle size, particle refractive index or molecule absorption spectra
- 3D shape models for the targets
- (Simple) coma density models
- Accurate observing geometries from SPICE kernels

Goals for the tool on what the instrument is outputting

- Physically correct models for instrument optics transmissions and detector noise components
- Correct units in noise modeling
  - Irradiance goes in
  - Photons transform into electric charges in the detector
  - Detector noise components in unit charges
  - Detector transforms charge counts into digital numbers with some bit depth
- Instrument output is binary stream of digital numbers

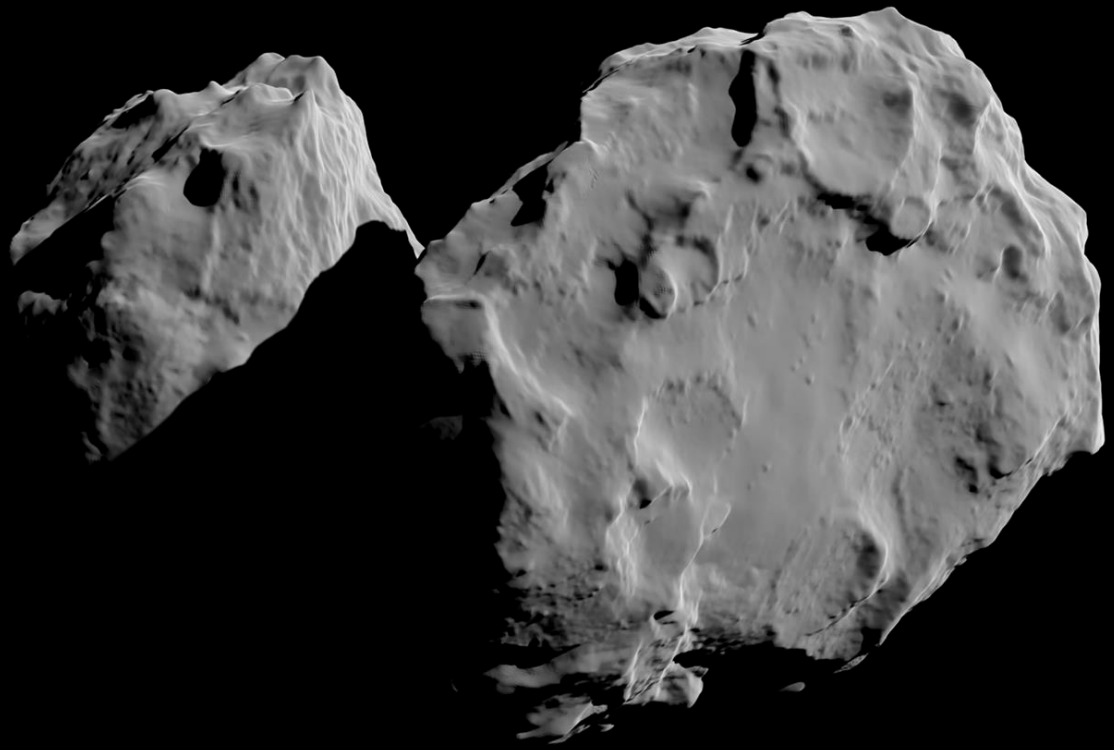
Design choices for the tool

- The 3D imaging simulation at one wavelength is done in Blender open-source 3D modeling software
- Image simulation part implemented as a Python module working inside Blender's Python console
- Disk and phase functions implemented as Blender shader nodes
- The GUI tools from Blender available, as well as Python command line using the Python module functionality
- Shape models as Blender targets
- SPICE kernel functionality embedded in the Blender Python console for observation geometry

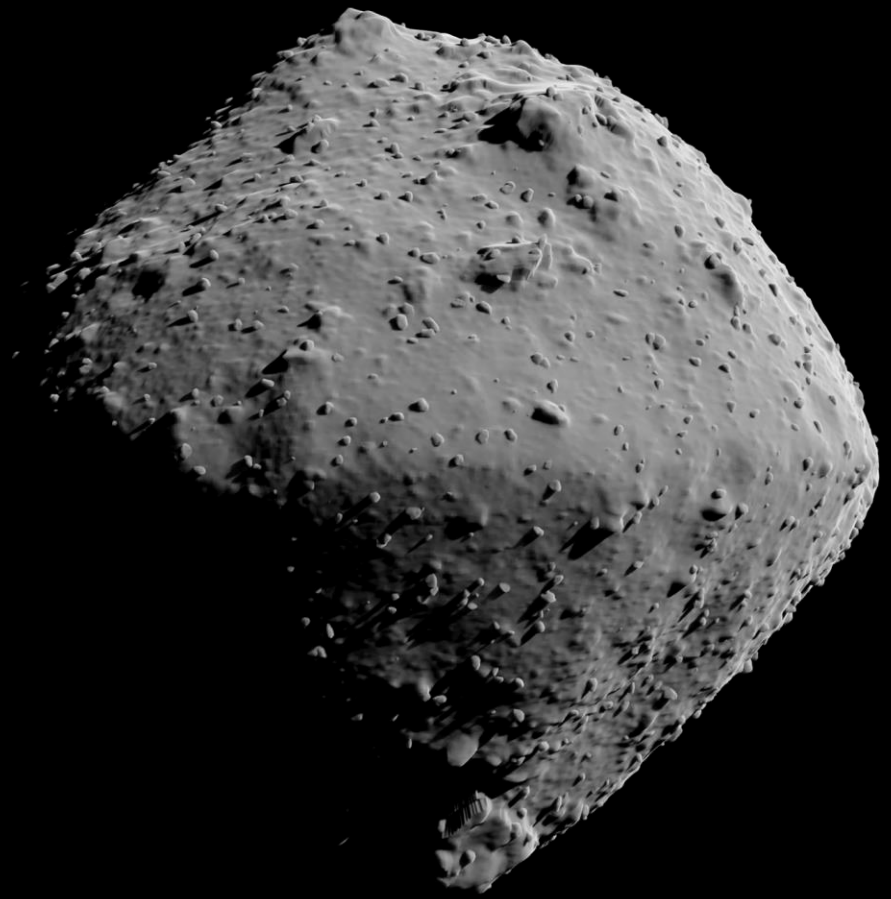
...design choices for the tool

- Instrument simulation as stand-alone Python module outside Blender
- Uses generated images from Blender with additional information to convert image digital numbers into
  - →radiance factor, then
  - →irradiance at instrument lens
  - →flux to detector pixel
  - →charges at detector pixel
  - →pixel readout digital number
  - →instrument binary output data stream
- With spectral instruments, if the phase function of the surface photometric model can be assumed wavelength-independent, then single image can be expanded into spectral datacube by multiplying by target reflectance spectrum
- For wavelength-dependent cases (comet with gas and dust coma), different wavelengths need to be individually rendered in Blender

Showcase of results with current version of the tool



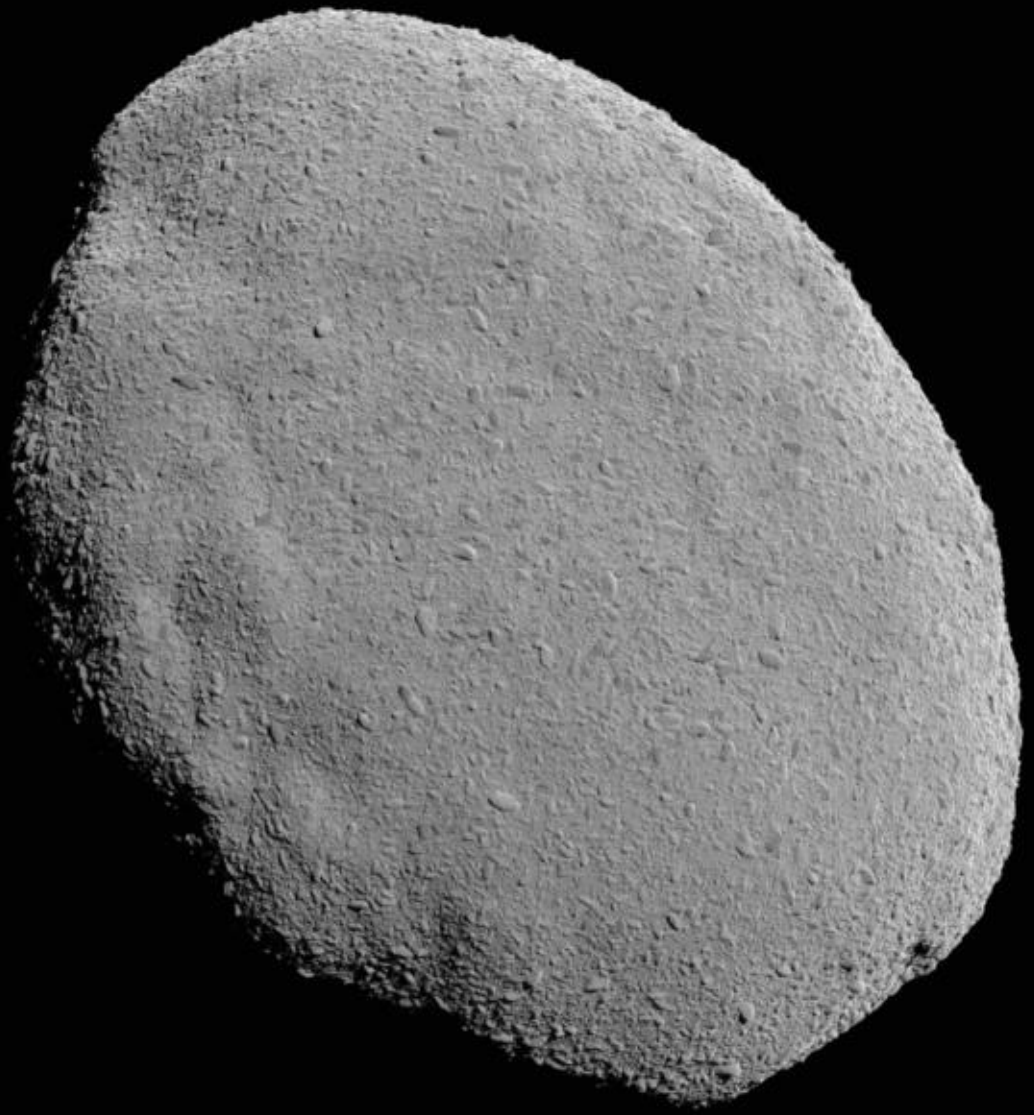
67P/C-G, McEwen photometric function

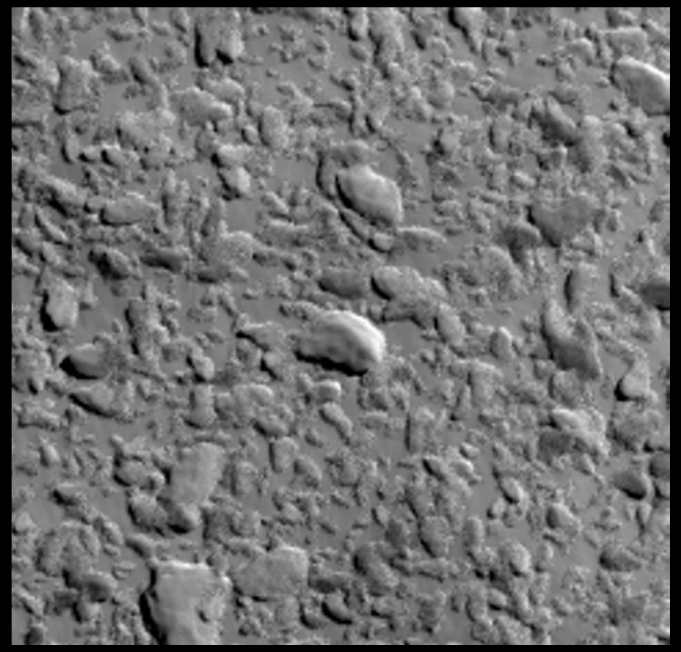


Ryugu with added boulders,
McEwen photometric function

Instrument orbiting an asteroid

...result showcase

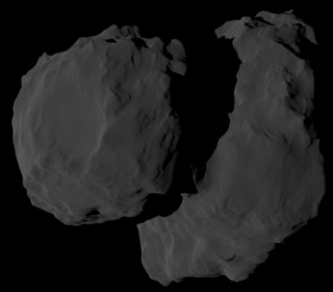


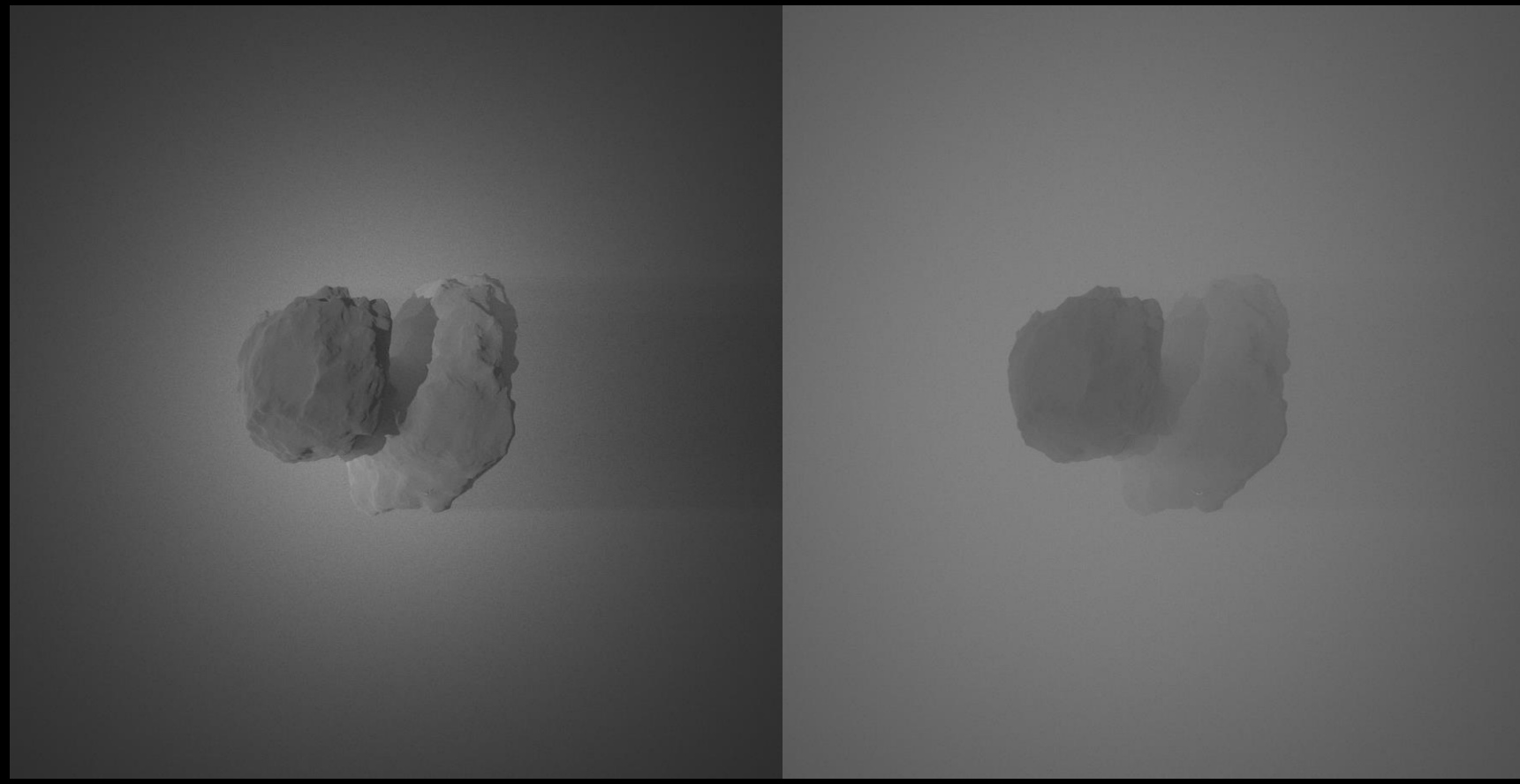


Didymos imaged with ASPECT camera from 5 km distance (left), and zooms on one surface patch (right) either without noise (up), or with added instrument noise (down)

...result showcase

Different levels of dust around a comet nucleus





Current state