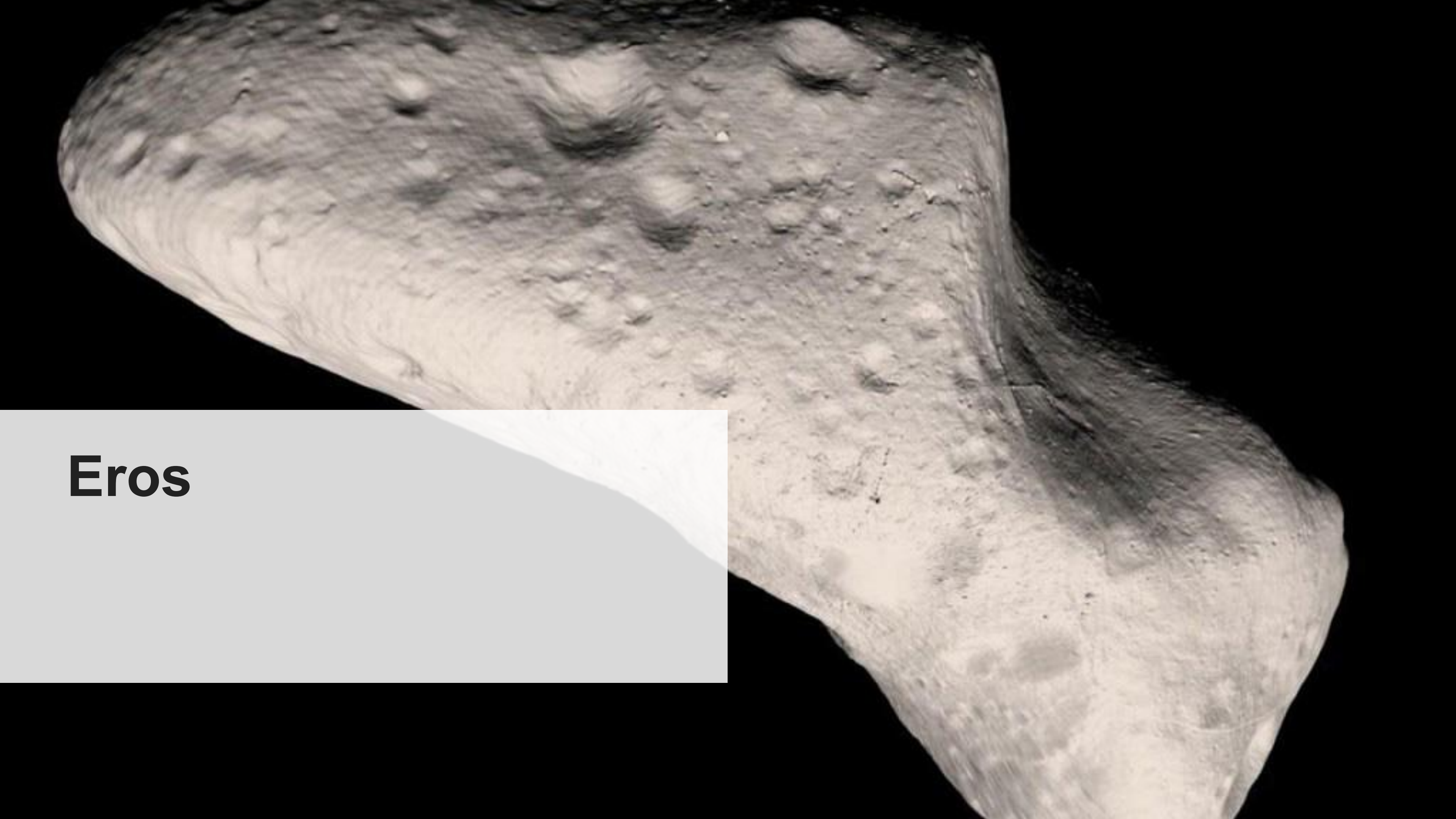
Neural Density Fields

by Dario Izzo
ESA's Advanced Concepts Team

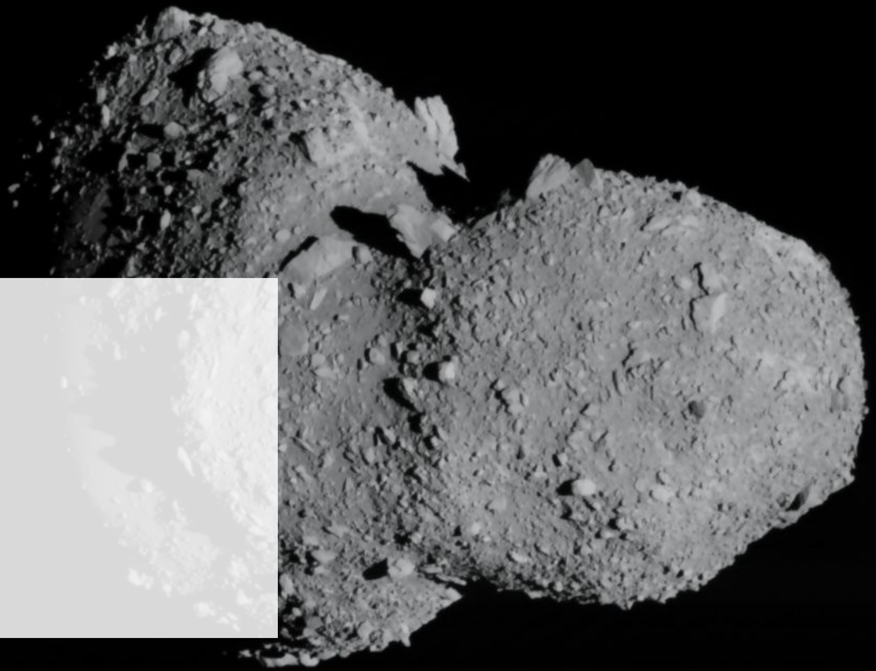
2022 seminar in Mathematics, Physics & Machine Learning,
https://mpml.tecnico.ulisboa.pt/seminars?action=show&id=6712

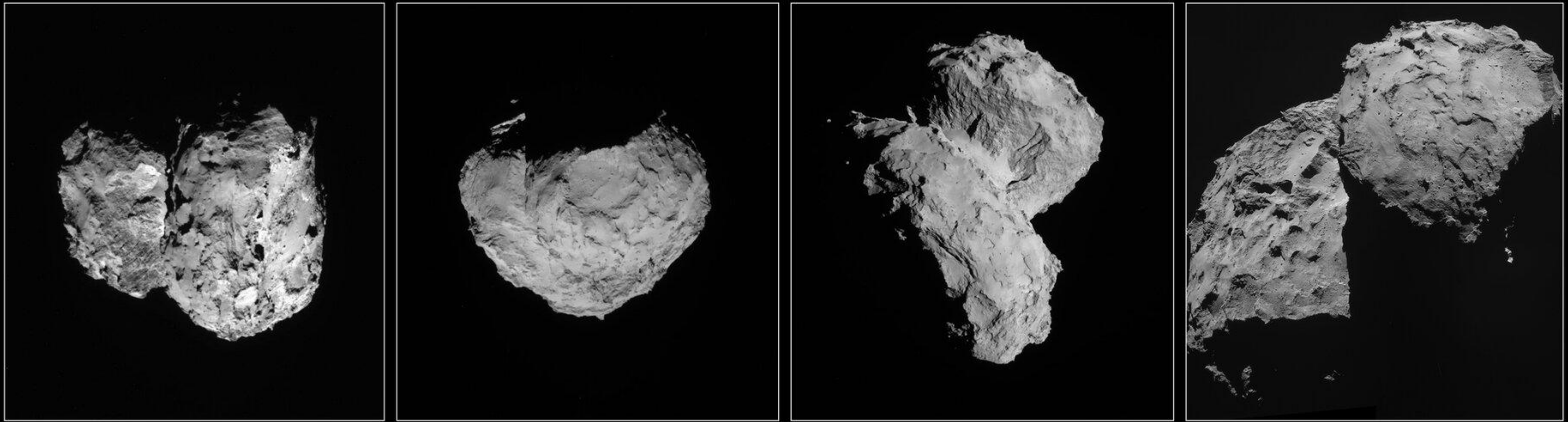
Irregular bodies in the solar system

Eros

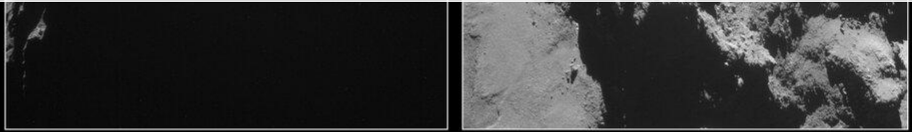


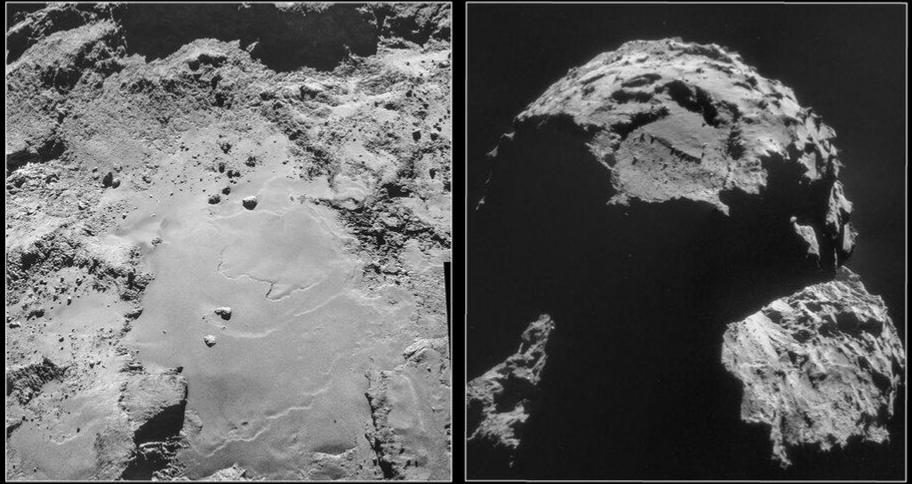
Itokawa



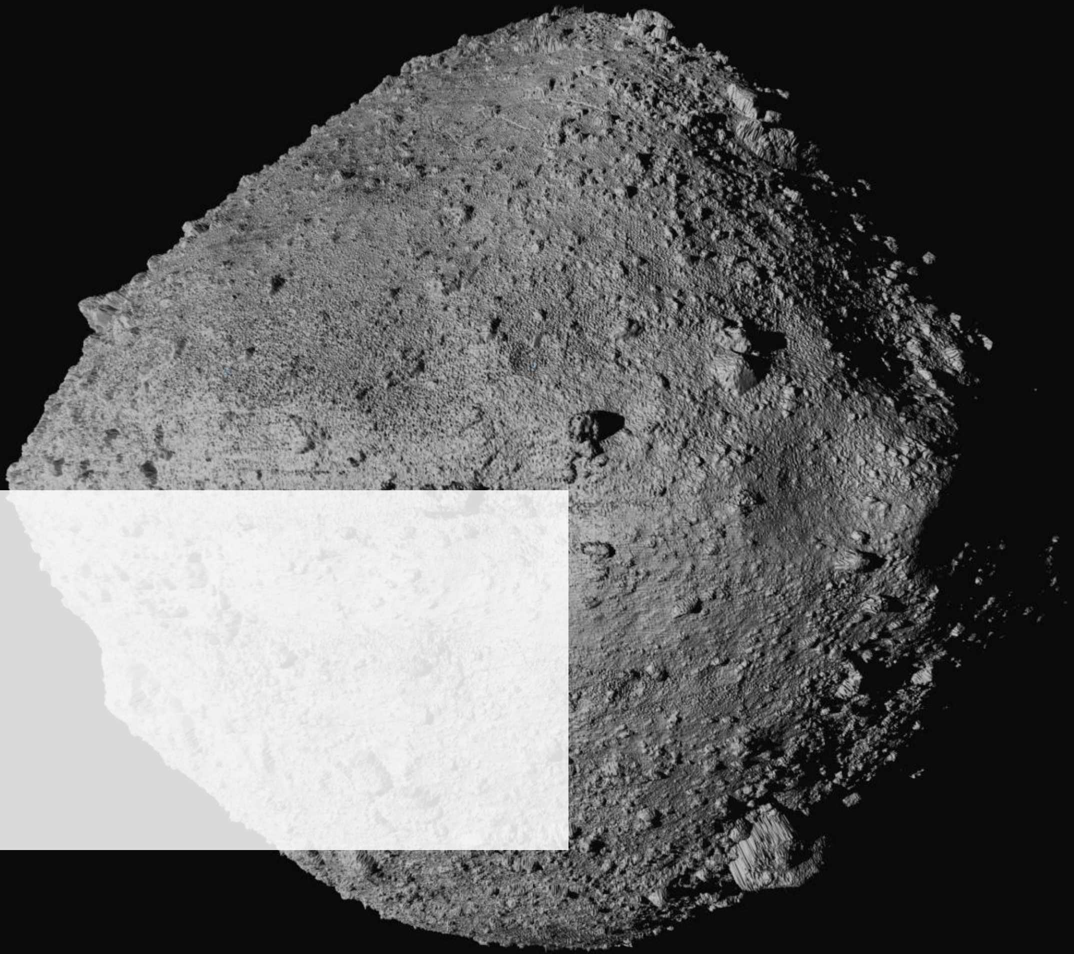


67P/Churyumov Gerasimenko

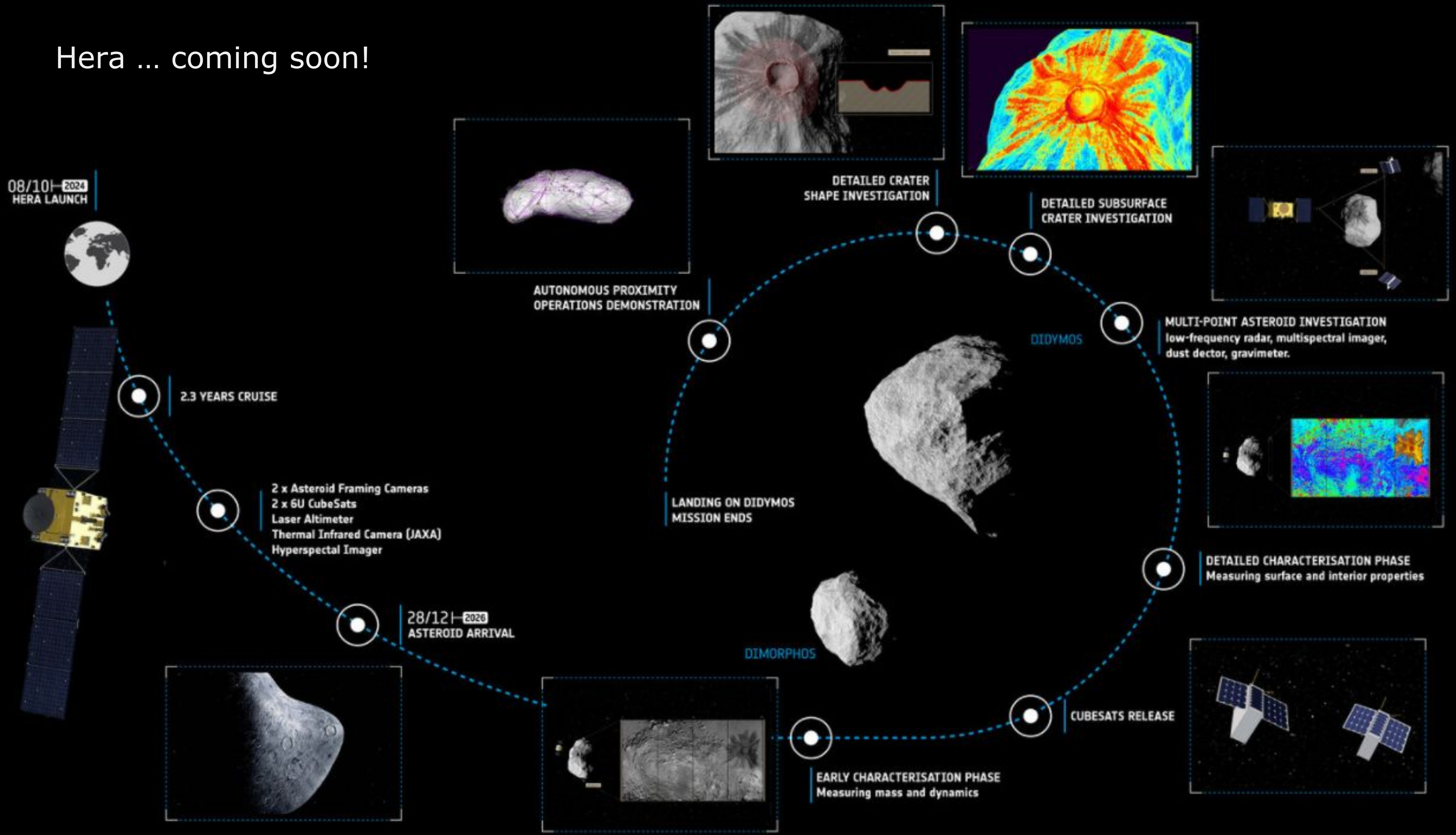




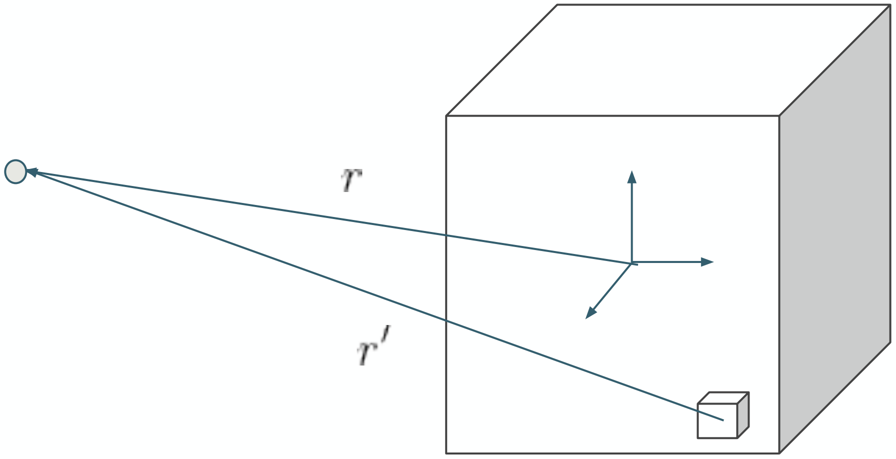
Bennu



Hera … coming soon!



Representing the gravity field (state-of-the-art)



$$U(r) = \int_{V'} \frac{\rho}{r'} dV' = U(x,y,z) = \int_{V'} \frac{\rho(x',y',z')}{\sqrt{(x-x')^2+(y-y')^2+(z-z')^2}} dV'$$

1. Spherical harmonics - (⅓)

Associated Legendre polynomials

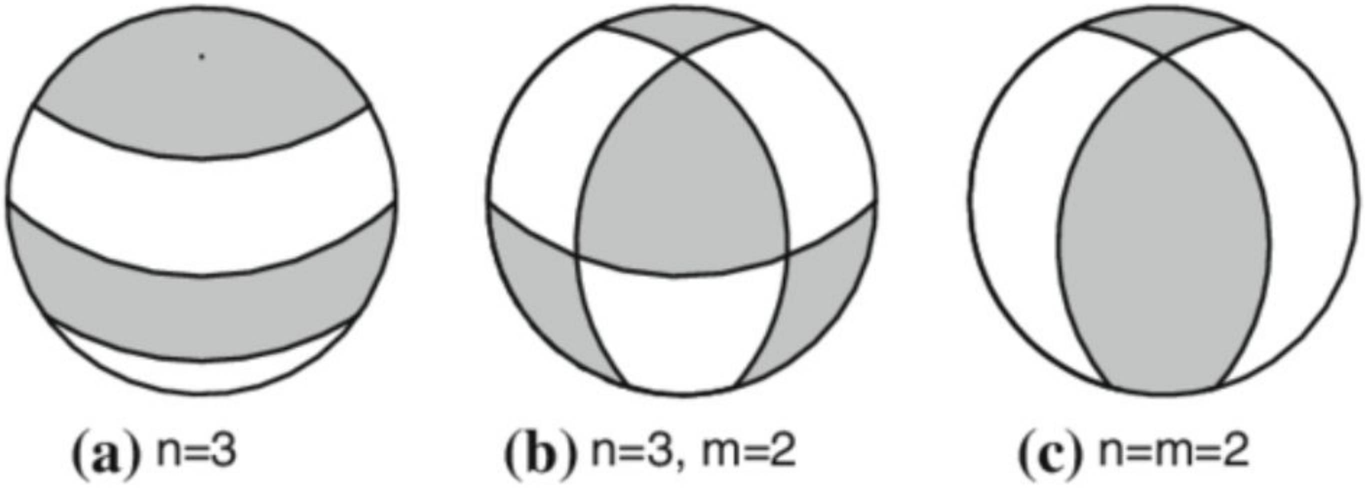
$$U(r,\theta,\phi) = \frac{\mu}{r}\sum_{l=0}^{l=\infty}\sum_{m=0}^{m=l}\left(\frac{r_0}{r}\right) P_{lm}(\cos\theta)\cdot(C_{lm}\cos m\phi + S_{lm}\sin m\phi)$$

$$C_{lm} = \frac{(2-\delta_{m,0})}{M}\frac{(l-m)!}{(l+m)!}\int_V \rho\left(\frac{r}{r_0}\right)^l \cdot P_{lm}(\cos\theta)\cos m\phi dV$$

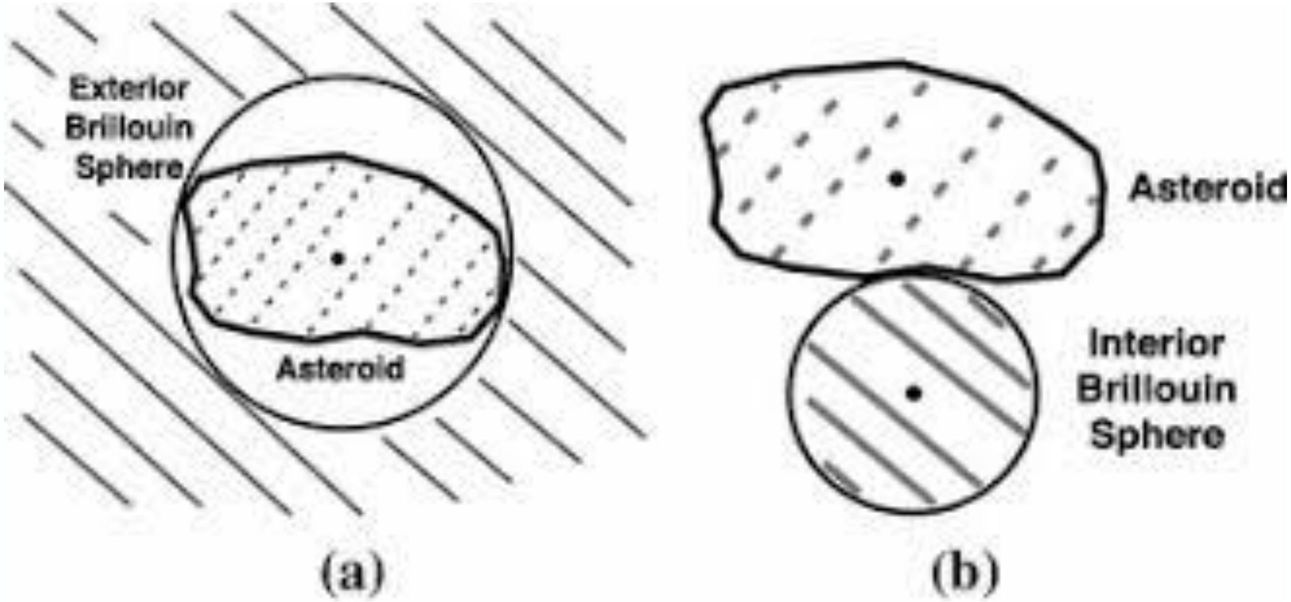
$$S_{lm} = \frac{(2-\delta_{m,0})}{M}\frac{(l-m)!}{(l+m)!}\int_V \rho\left(\frac{r}{r_0}\right)^l \cdot P_{lm}(\cos\theta)\sin m\phi dV$$

Stokes Coefficients

Spherical harmonics - (⅔)

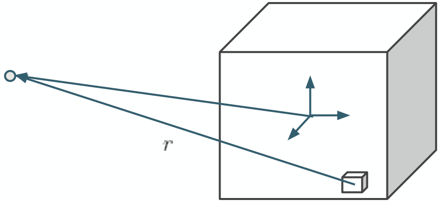


Spherical harmonics - (3/3)



Poor convergence properties next to irregular surfaces.

2.Polyhedral gravity (½)

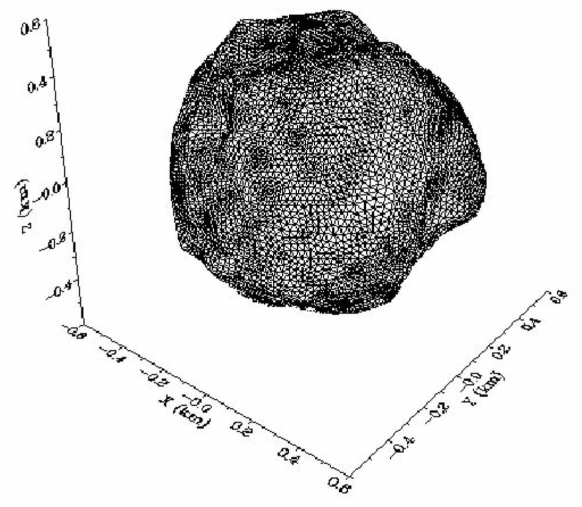


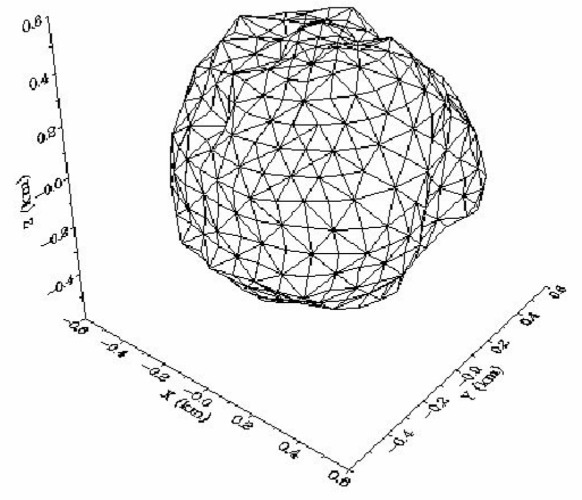
divergence theorem

$$\iiint_V \frac{\rho}{r} dV = \rho \iiint_V \frac{1}{r} dV = \rho \iiint_V \operatorname{div}\left(\frac{\mathbf{r}}{r}\right) dV = \rho \oiint_{\partial V} \frac{\mathbf{r} \cdot \mathbf{n}}{r} dV$$

constant

Polyhedral gravity (2/2)

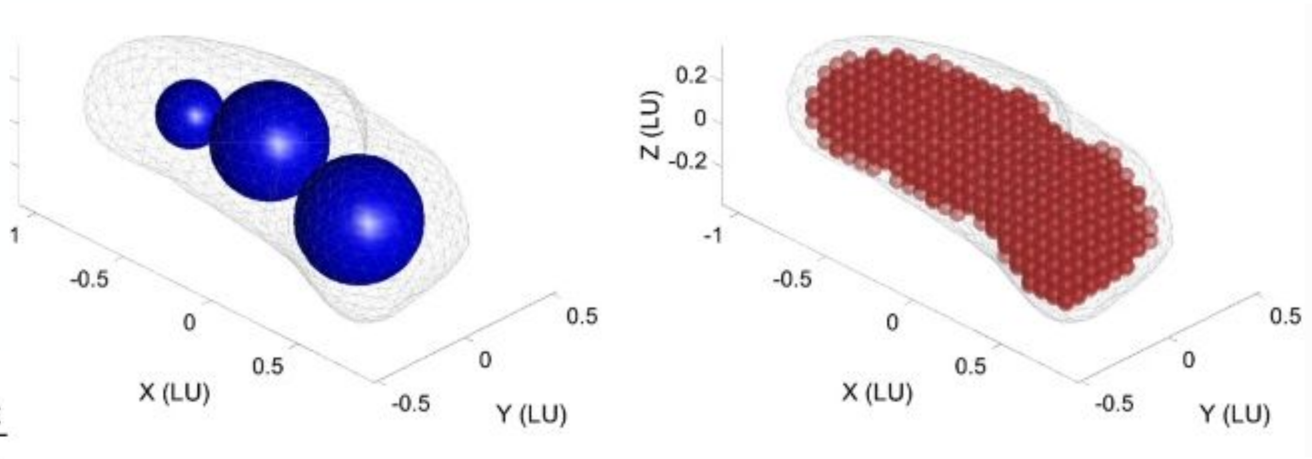




code: https://github.com/esa/polyhedral-gravity-model

Relies and needs on the asteroid shape, unable to see inside.

3.Mascon models



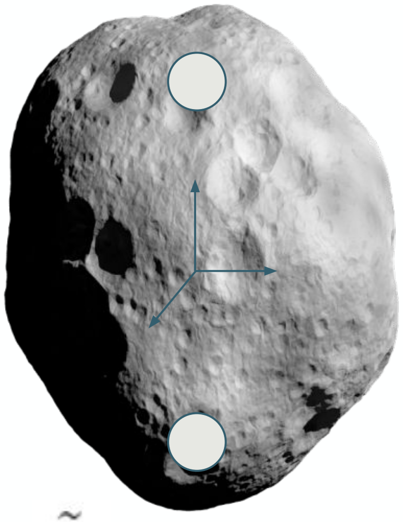
$$U(r) = \sum_{i=0}^{N} \frac{m_i}{r_i}$$

Great flexibility but poor precision next to the surface and needs shape information.

An ill-posed problem!

(gravity inversion)

m



$\tilde{m}$

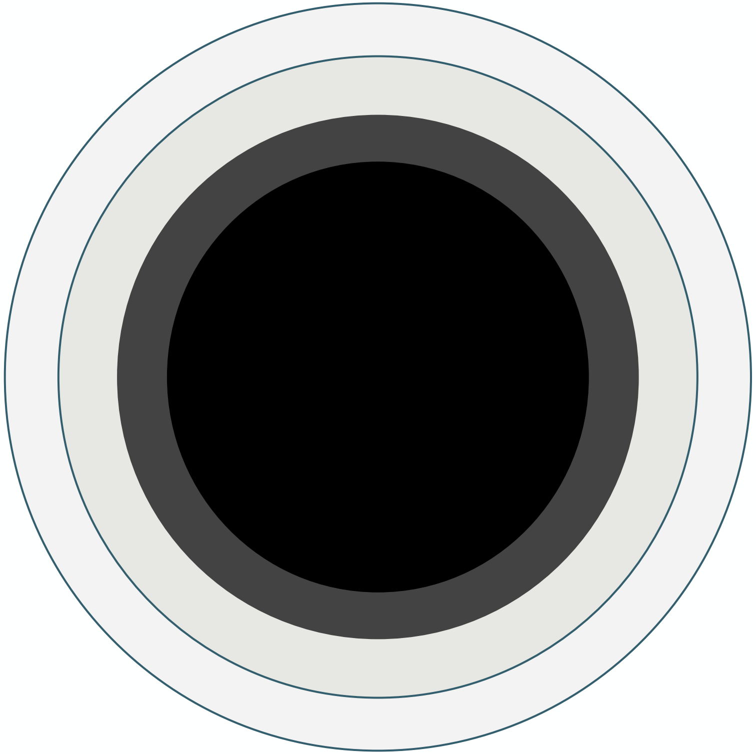
model

observations

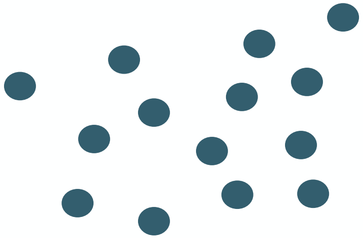
U_1 U_2

$$\begin{cases} \dfrac{m}{r_1} + \dfrac{\tilde{m}}{r_1} = U_1 \\ \dfrac{m}{r_2} + \dfrac{\tilde{m}}{r_2} = U_2 \end{cases}$$

model



observations



Newton’s Shell theorem :(

a fourth way …
Neural Density Fields

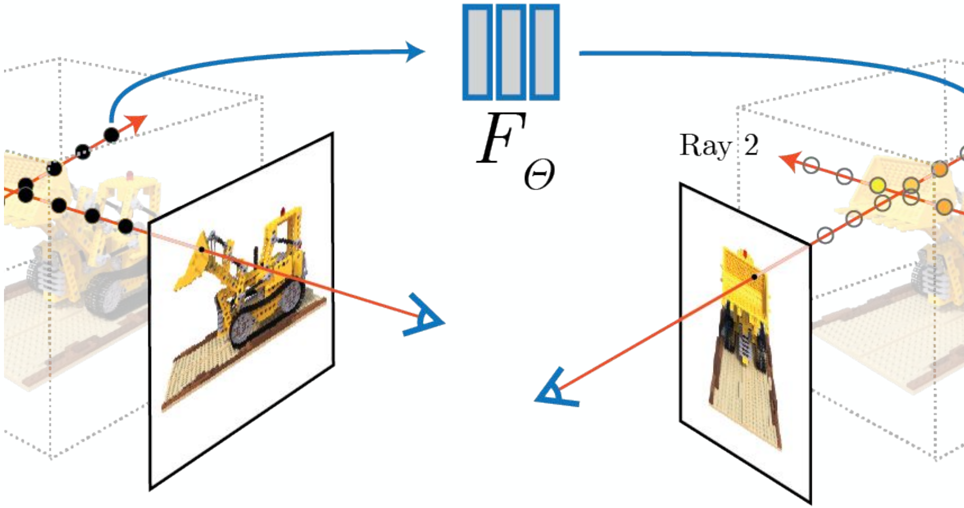
Izzo, Dario, and Pablo Gómez. "Geodesy of irregular small bodies via neural density fields: geodesyNets." *arXiv preprint arXiv:2105.13031* (2021).

von Looz, Moritz, Pablo Gomez, and Dario Izzo. "Study of the asteroid Bennu using geodesyANNs and Osiris-Rex data." *arXiv preprint arXiv:2109.14427* (2021).

Inspired from NeRF:
(neural radiance fields)

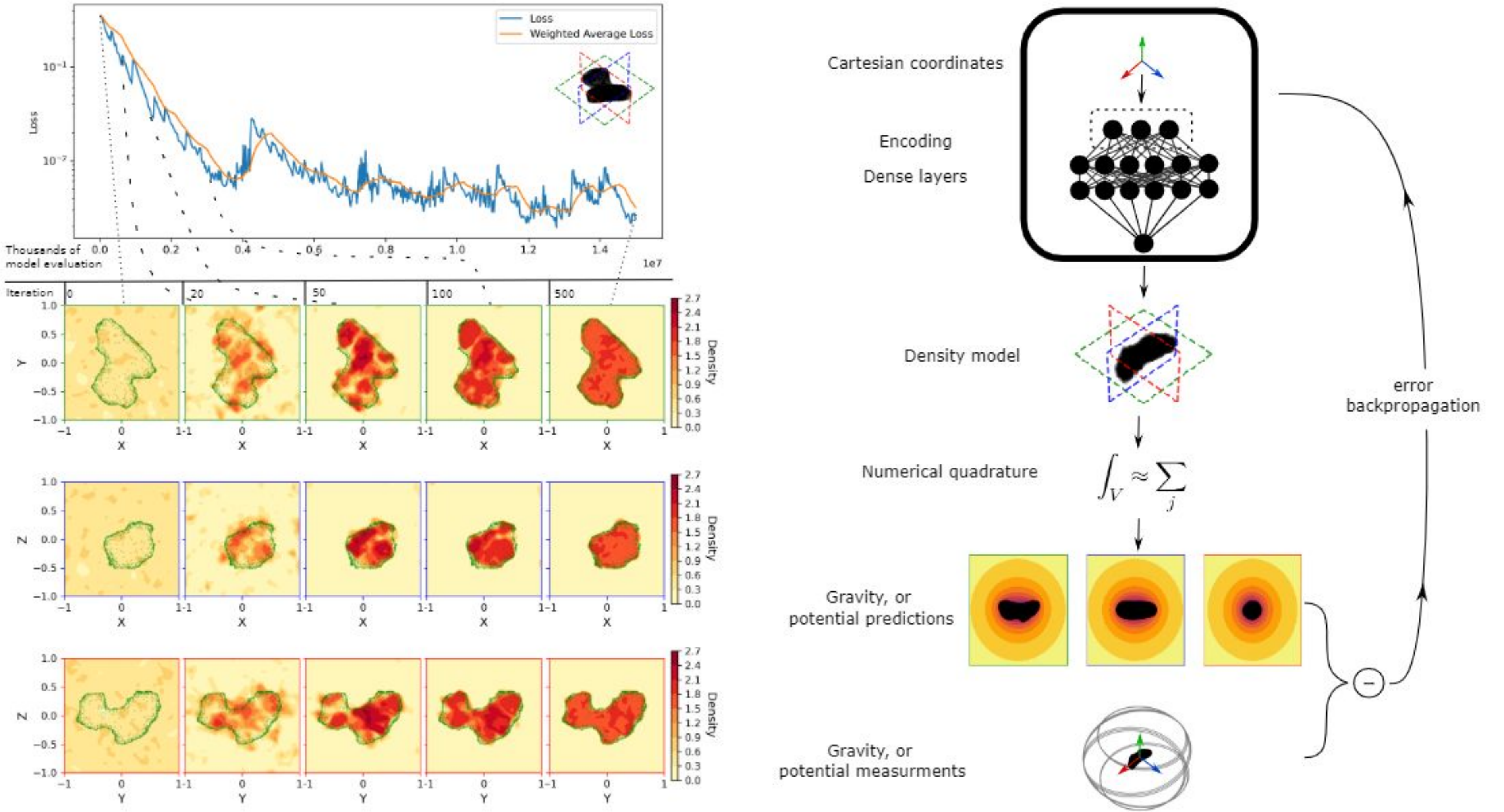
The weights of a neural network are able to store highly detailed information on complex 3D scene

Mildenhall, Ben, et al. "Nerf: Representing scenes as neural radiance fields for view synthesis." *European conference on computer vision*. Springer, Cham, 2020.



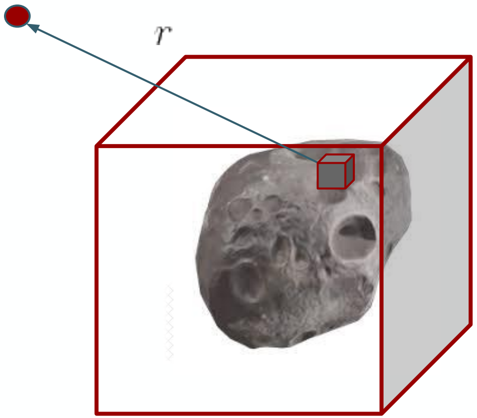
“With four parameters I can fit an elephant, with five I can make him wiggle his trunk”

John von Neumann


Loss
Weighted Average Loss
Loss
Thousands of model evaluation
1e7
Iteration
0
20
50
100
500
Density
Cartesian coordinates
Encoding
Dense layers
Density model
Numerical quadrature
Gravity, or potential predictions
Gravity, or potential measurments
error backpropagation

	Approach			
	Masc.	Harm.	Poly.	geodesyNets
Differentiable	✗	✓	✓	✓
Inside Brillouin sphere	✓	✗	✓	✓
Heterogeneous densities	✓	✓	✗	✓
Shape model not needed	✓	✓	✗	✓
Can utilize shape model	✓	✗	✓	✓
Accurate in the near field	✗	✓	✓	✓

A volume integral: $\longrightarrow \int_V \frac{\rho}{r} dV \longrightarrow$?



$$\iiint_V \frac{\mathcal{N}(x,y,z)}{r} dV \approx \sum_i w_i \frac{\mathcal{N}(x_i,y_i,z_i)}{r_i}$$

Numerical quadrature

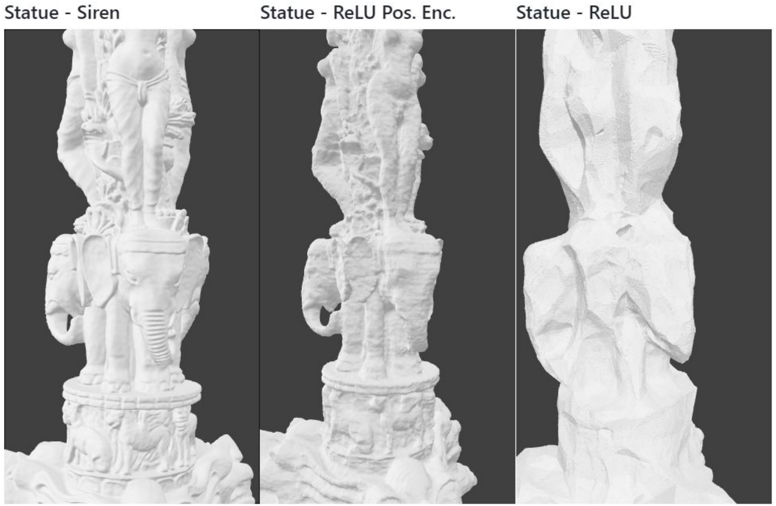
$$\iiint_V \frac{\mathcal{N}(x,y,z)}{r} dV = E\left[\frac{1}{n}\sum_i \frac{\mathcal{N}(x_i,y_i,z_i)}{r_i p(x_i,y_i,z_i)}\right]$$

Monte Carlo methods

- Importance sampling
- Stratified sampling

Network architecture: SIREN

$$\Phi(\mathbf{x}) = \mathbf{W}_n(\phi_{n-1} \circ \phi_{n-2} \circ \ldots \circ \phi_0)(\mathbf{x}) + \mathbf{b}_n, \quad \mathbf{x}_i \mapsto \phi_i(\mathbf{x}_i) = \sin(\mathbf{W}_i \mathbf{x}_i + \mathbf{b}_i).$$



Sitzmann, Vincent, et al. "Implicit neural representations with periodic activation functions." *Advances in Neural Information Processing Systems* 33 (2020): 7462-7473.

The loss

constraint: $$\iiint_V \rho(x,y,z)\,dV = M$$

solution: $$\rho(x,y,z) = c\mathcal{N}(x,y,z)$$

Its as if we added one more parameter (weight) after the output neurons!

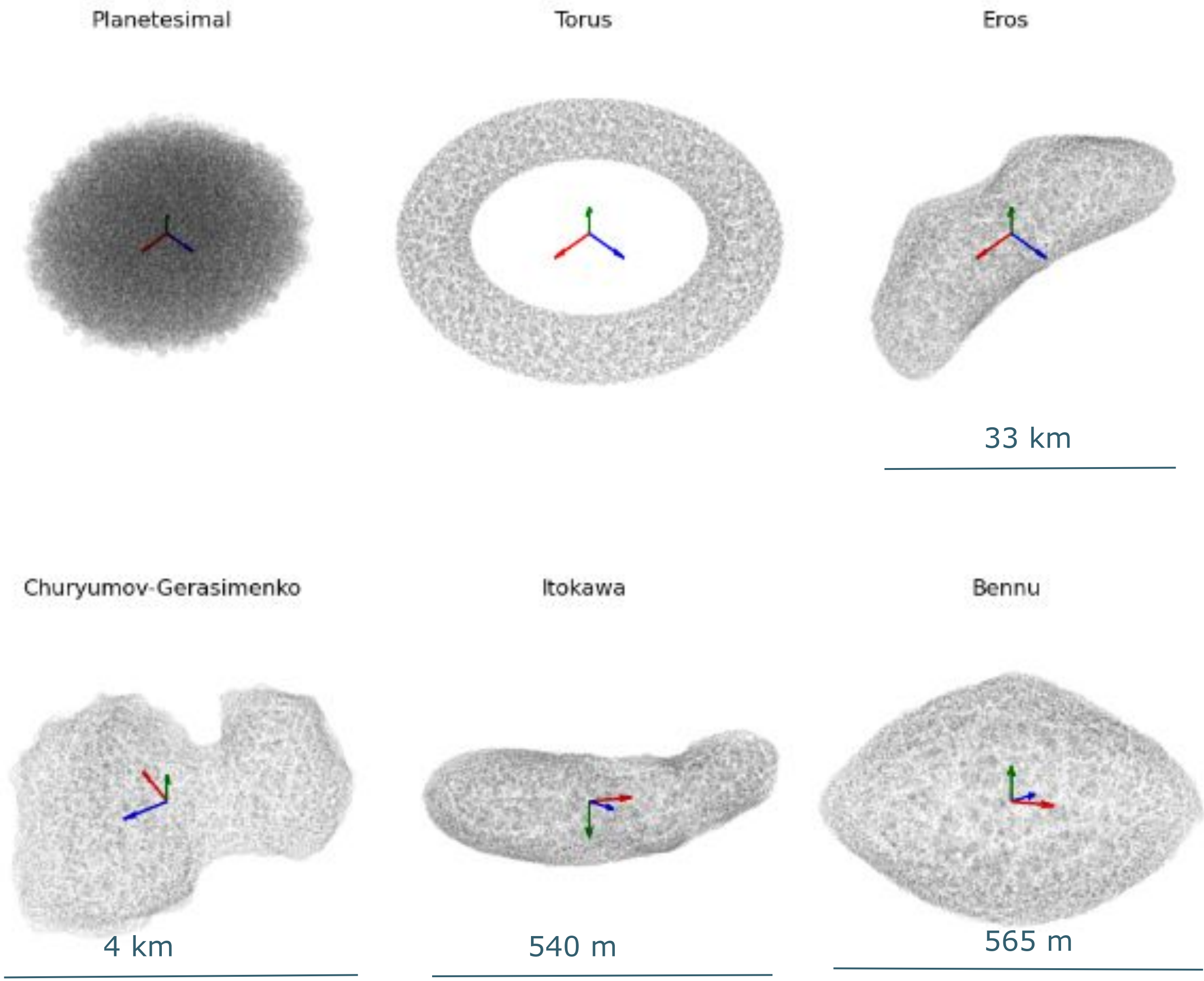
$$\sum_i (y_i - c\hat{y}_i)^2 = c^2 \sum_i \hat{y}_i^2 - 2c \sum_i y_i \hat{y}_i + \sum_i y_i^2$$

$$c = \frac{\sum y_i \hat{y}_i}{\sum \hat{y}_i^2}$$

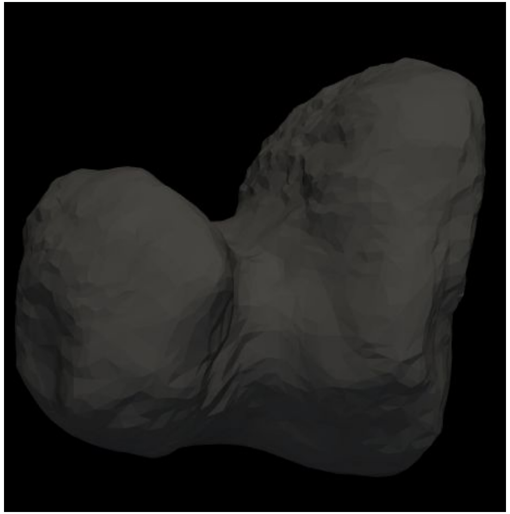
... can also be used in generic ML tasks

Experiments

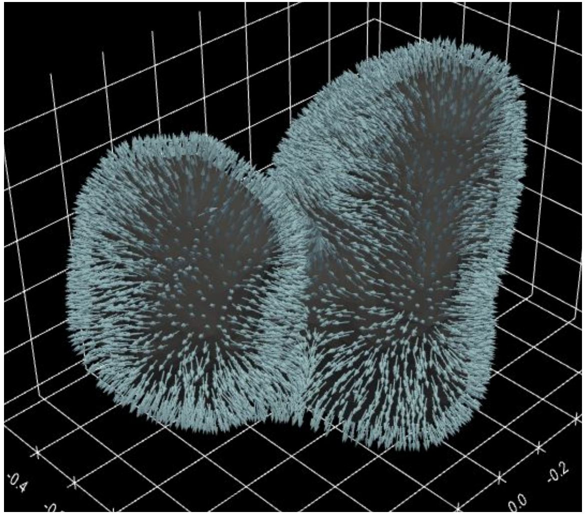
Test cases


Planetesimal
Torus
Eros
33 km
Churyumov-Gerasimenko
4 km
Itokawa
540 m
Bennu
565 m

Sampling points



Model



Surface normals

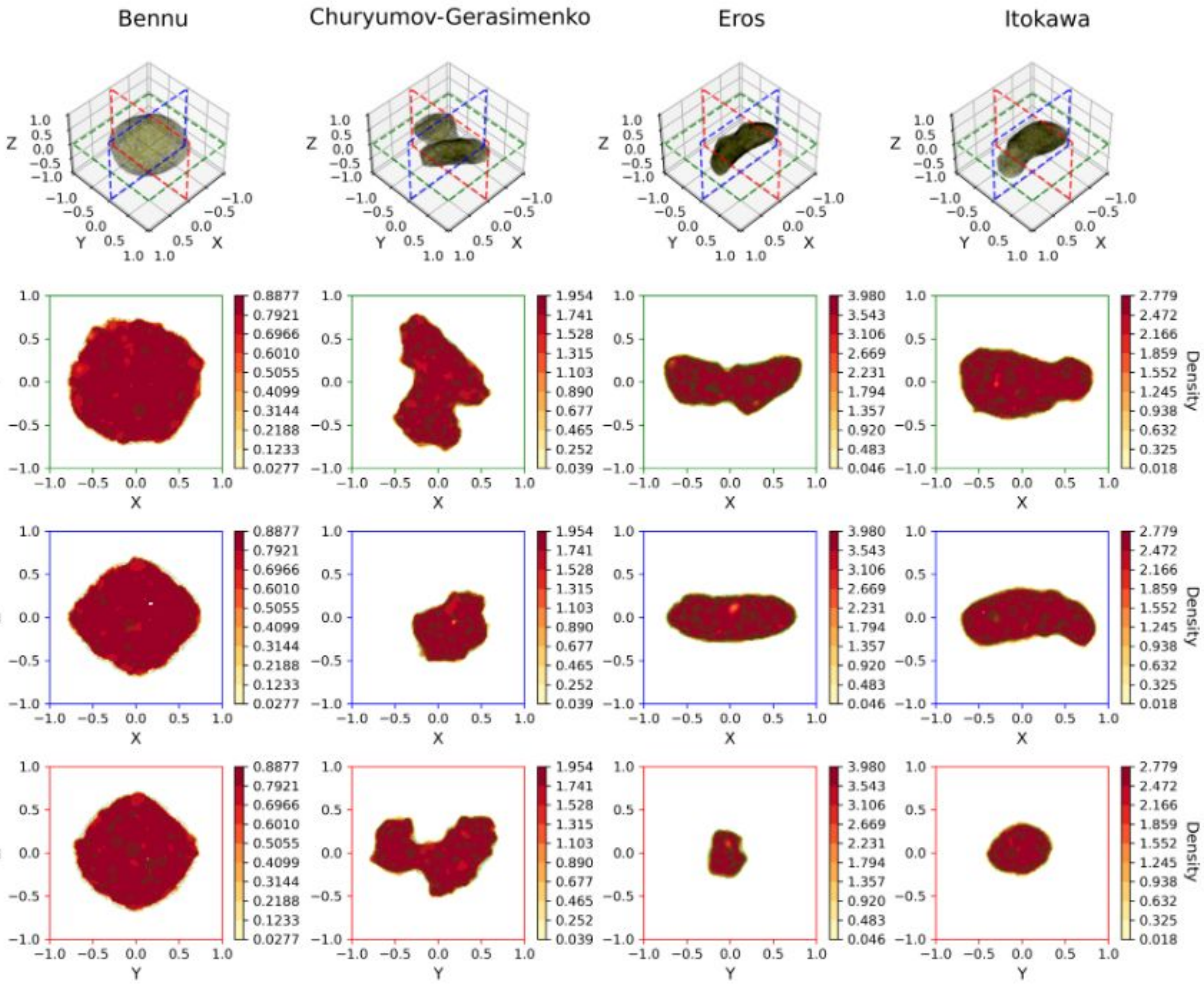
Samples taken at three altitudes low, mid and high

Nominal Learning
only gravity

Results

Nominal learning

		Sampling Altitudes			Absolute Errors			Relative Errors		
	Body	$h_{low}[m]$	$h_{med}[m]$	$h_{hi}[m]$	$\epsilon_{low}[m/s^2]$	$\epsilon_{med}[m/s^2]$	$\epsilon_{hi}[m/s^2]$	$\epsilon_{low}[\%]$	$\epsilon_{med}[\%]$	$\epsilon_{hi}[\%]$
HMG	Bennu	14.1	28.2	70.4	2.63e-08	4.75e-09	6.89e-10	0.11	0.02	0.005
	Churyumov-Gerasimenko	125	250	625	1.13e-07	2.02e-08	2.20e-09	0.19	0.04	0.006
	Eros	817	1630	4080	2.24e-06	4.45e-07	5.52e-08	0.16	0.04	0.01
	Itokawa	14	28	70.1	3.15e-08	6.35e-09	1.06e-09	0.15	0.04	0.01
	Planetesimal	125	250	625	5.69e-08	1.31e-08	3.43e-09	0.11	0.03	0.011
	Torus	125	250	625	1.41e-07	3.74e-08	8.49e-09	0.28	0.09	0.034
HTG	Bennu	14.1	28.2	70.4	4.70e-08	9.57e-09	1.57e-09	0.20	0.05	0.011
	Itokawa	14	28	70.1	4.27e-08	9.36e-09	9.33e-10	0.20	0.05	0.009
	Planetesimal	125	250	625	9.90e-08	2.53e-08	4.22e-09	0.20	0.06	0.014



Bennu
Churyumov-Gerasimenko
Eros
Itokawa
Density

Visualizing the Neural Density field

Torus

Visualizing the Neural Density field

67P

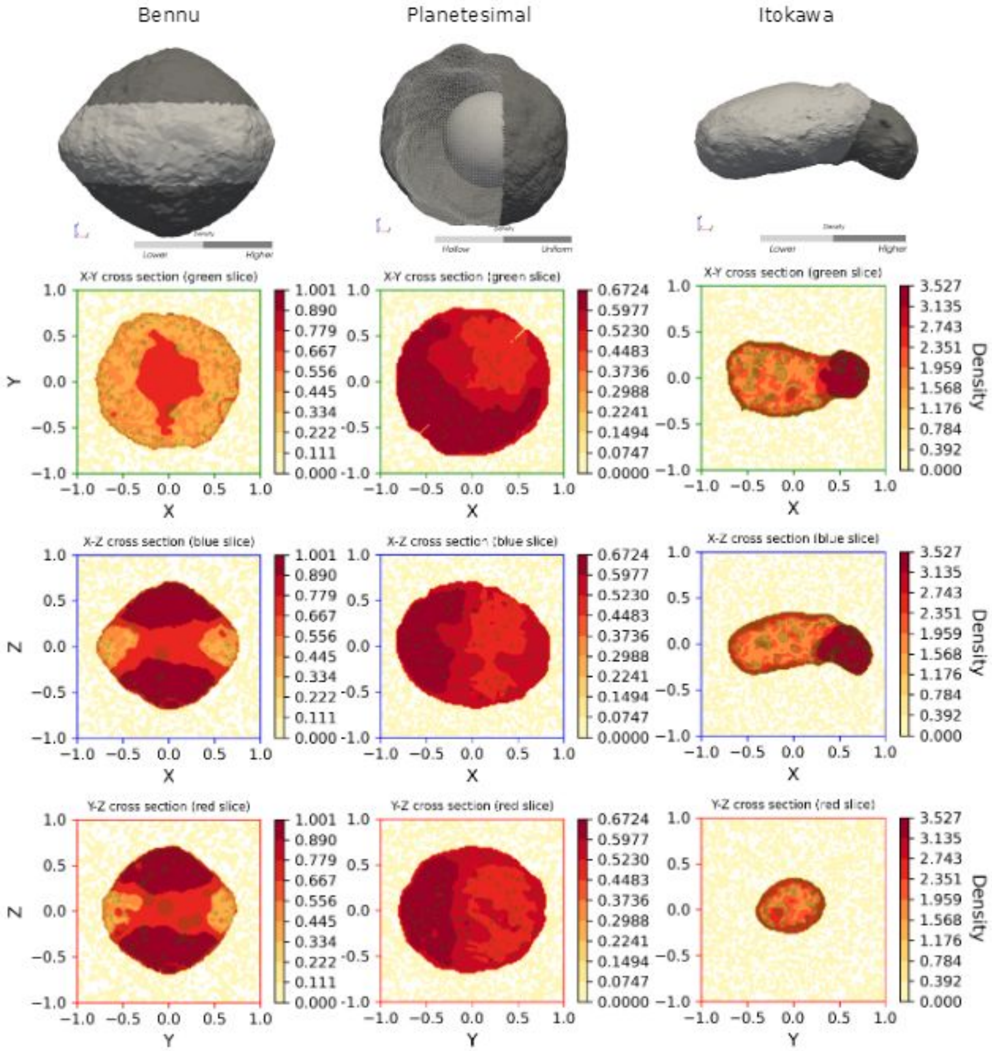
Differential Learning

fusing in camera information

Results

differential learning

	Sampling Altitudes			Absolute Errors			Relative Errors		
Heterogeneous body	$h_{low}[m]$	$h_{med}[m]$	$h_{hi}[m]$	$\epsilon_{low}[m/s^2]$	$\epsilon_{med}[m/s^2]$	$\epsilon_{hi}[m/s^2]$	$\epsilon_{low}[\%]$	$\epsilon_{med}[\%]$	$\epsilon_{hi}[\%]$
Bennu	14.1	28.2	70.4	4.07e-08	1.19e-08	7.67e-09	0.10	0.03	0.031
Itokawa	14	28	70.1	2.49e-08	1.45e-08	1.01e-08	0.12	0.08	0.091
Planetesimal	125	250	625	3.55e-08	2.29e-08	1.84e-08	0.08	0.06	0.071



Bennu
Planetesimal
Itokawa
Lower
Higher
Hollow
Uniform
Lower
Higher
X-Y cross section (green slice)
X-Z cross section (blue slice)
Y-Z cross section (red slice)
X
Y
Z
Density
1.001
0.890
0.779
0.667
0.556
0.445
0.334
0.222
0.111
0.000
0.6724
0.5977
0.5230
0.4483
0.3736
0.2988
0.2241
0.1494
0.0747
0.0000
3.527
3.135
2.743
2.351
1.959
1.568
1.176
0.784
0.392
0.000

Do we need a neural model for the density field?

		Bennu	Churyumov-Gerasimenko	Eros	Itokawa
GeodesyNet	low	**0.72**	**2.30**	**1.82**	**2.13**
	hi	0.02	**1.75**	0.17	**0.38**
masconCUBE	low	1.00	2.87	2.39	2.47
	hi	**0.01**	2.03	**0.13**	0.70

Open questions:

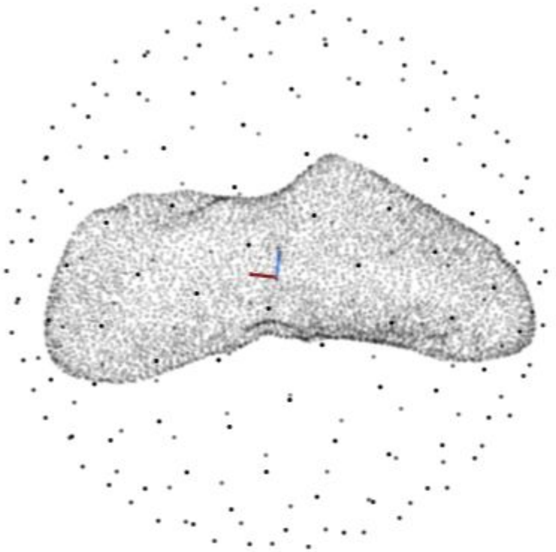
- Pumping up the card memory (points used limited by NVIDIA 2080 RTX capability).
- Numerical quadrature vs Monte Carlo methods.
- Sensitivity to data noise (random and non gravitational).
- Sensitivity to data availability (spacecraft orbit design).
- On-board training effectiveness.
- Thorough comparison with masconCUBE and spherical harmonics using the same training.

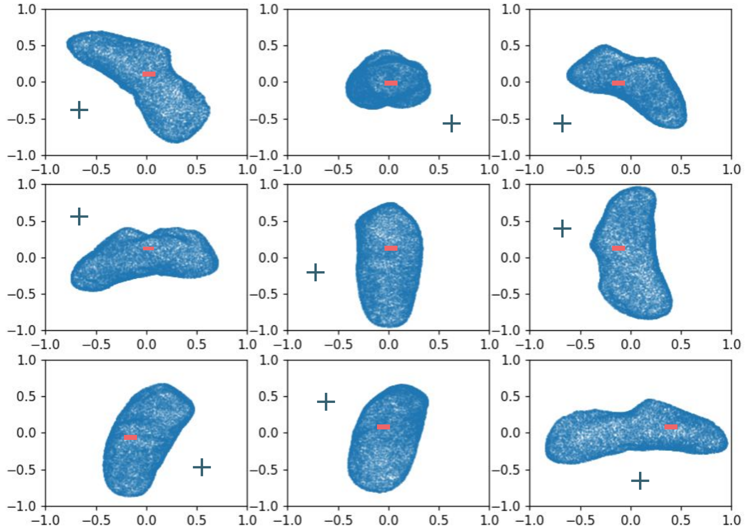
Eclipse Nets

… also an implicit neural representation

Biscani, Francesco, and Dario Izzo. "Reliable event detection for Taylor methods in astrodynamics." *Monthly Notices of the Royal Astronomical Society* 513.4 (2022): 4833-4844.

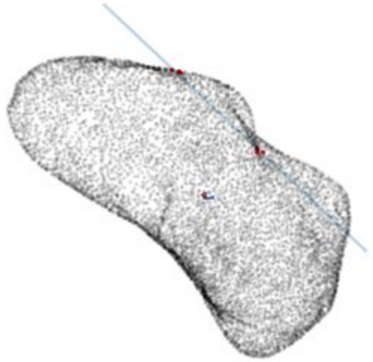
The eclipse function: $F(\mathbf{r}, \hat{\mathbf{i}}_S)$



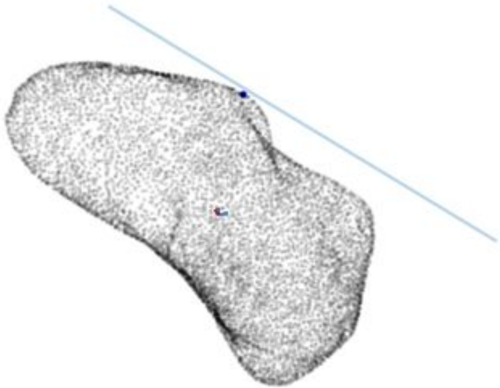


The eclipse function can be used to determine the presence or absence of solar radiation pressure.

The eclipse function: $F(\mathbf{r}, \hat{\mathbf{i}}_S)$

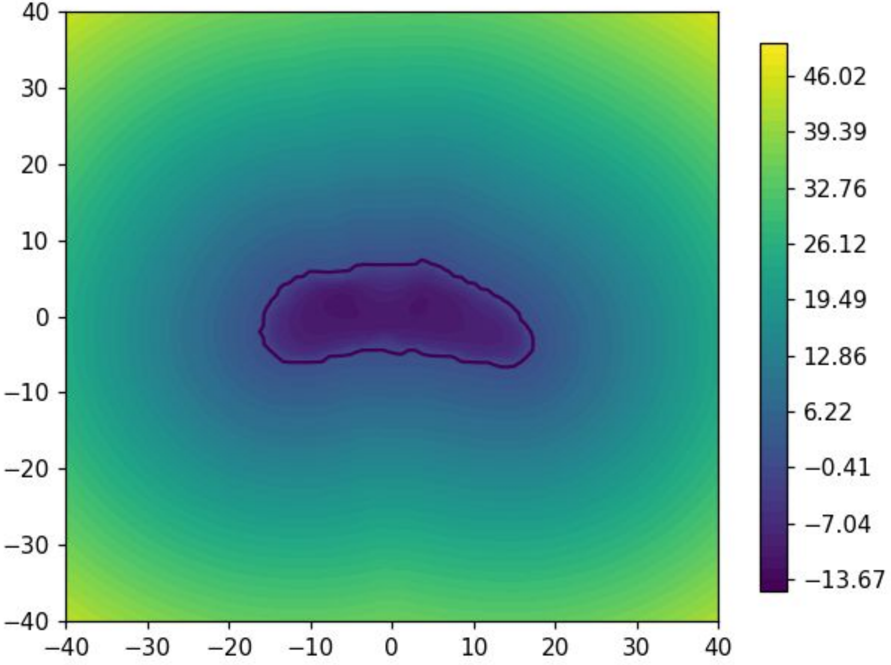


Even intersections -> EF is the length of the ray inside the asteroid

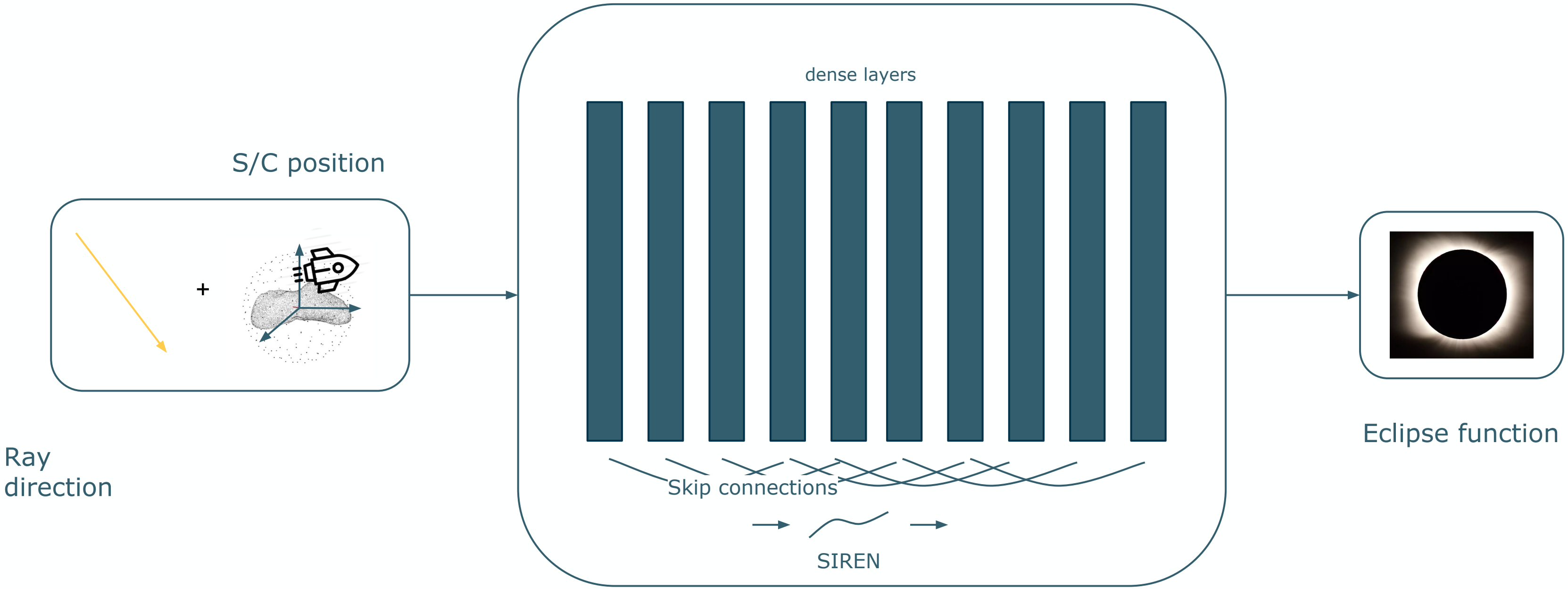


No intersections -> EF is the distance of the point to the shadow cone.

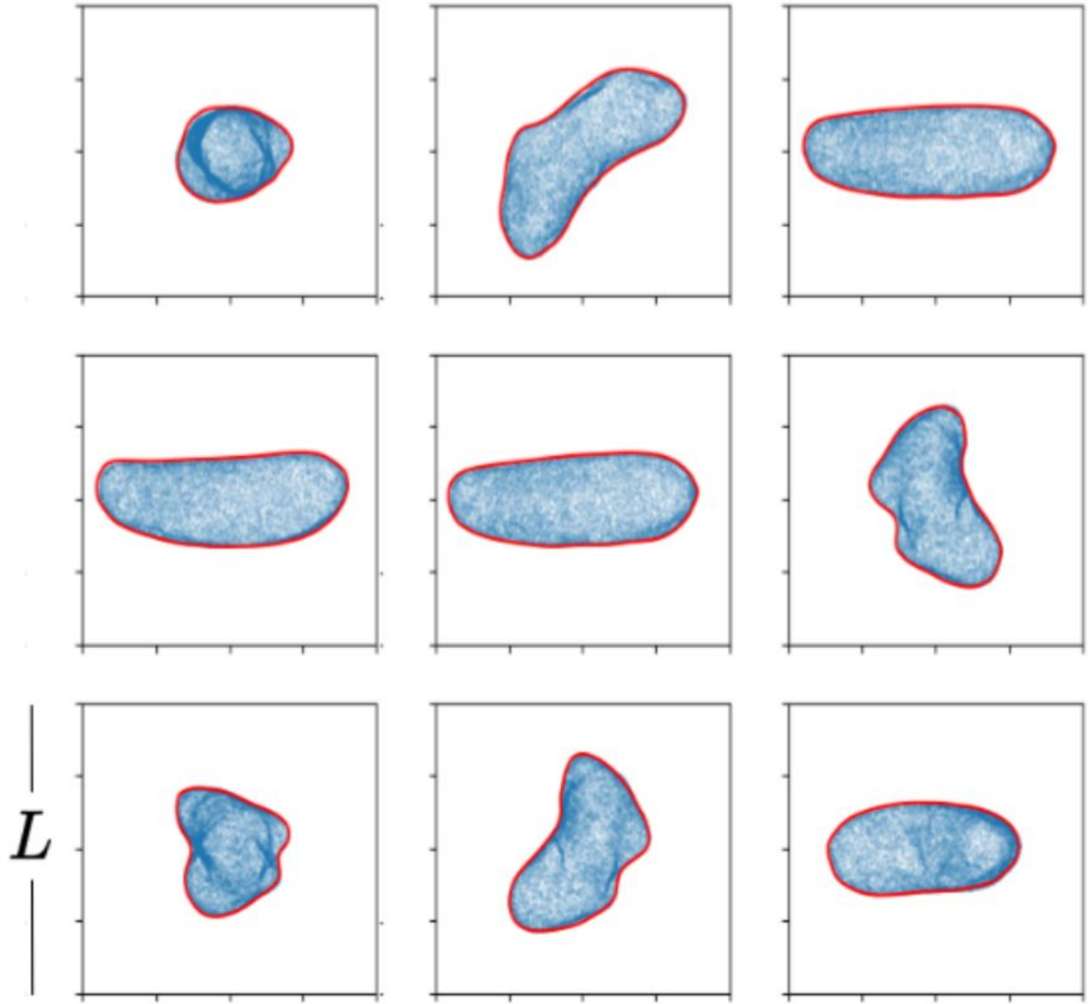
The eclipse function: $F(\mathbf{r}, \hat{\mathbf{i}}_S)$



eclipseNet



S/C position
Ray direction
dense layers
Skip connections
SIREN
Eclipse function


L

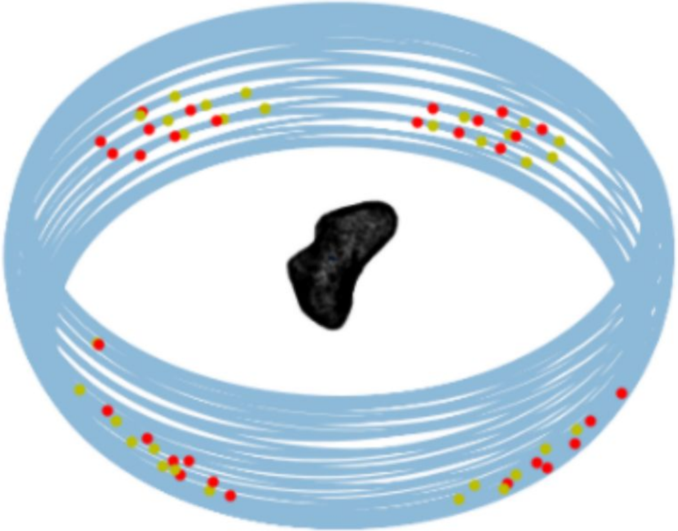
$$\ddot{\mathbf{r}} = -G\sum_{j=0}^{N}\frac{m_j}{|\mathbf{r}-\mathbf{r}_j|^3}(\mathbf{r}-\mathbf{r}_j) - 2\omega\times\mathbf{v} - \omega\times\omega\times\mathbf{r} - \eta\nu(\mathbf{r})\hat{\mathbf{i}}_S(t),$$

$$\nu(\mathbf{r}) = 1$$

no penubra

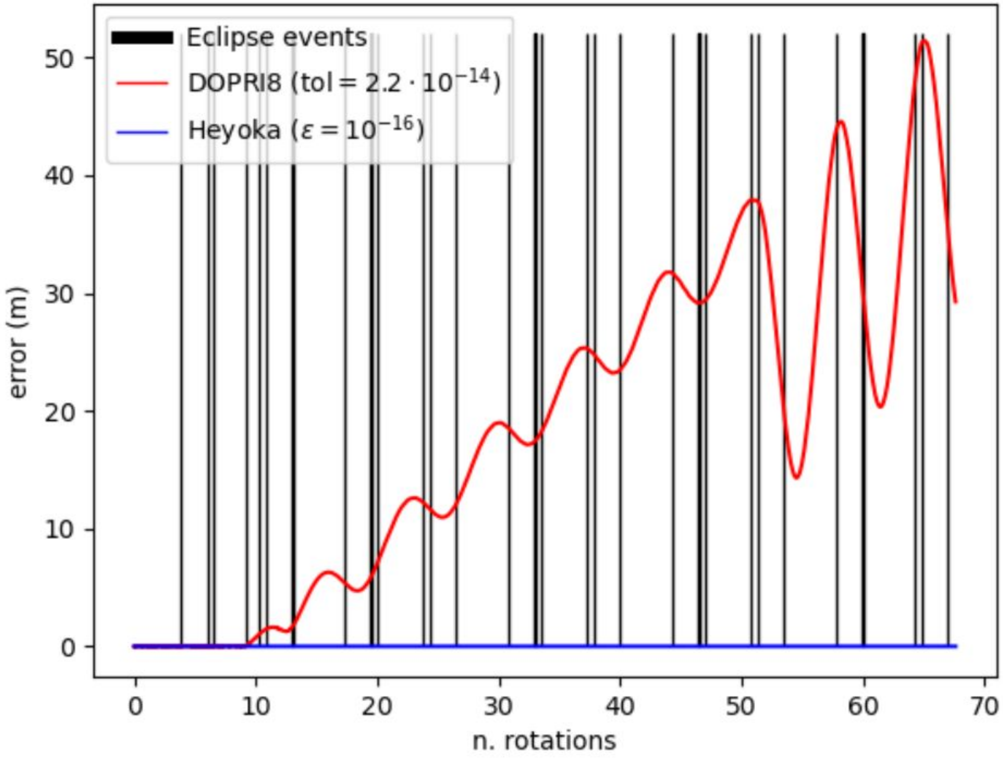
$$\eta(\mathbf{r}, \mathbf{i}_S) = H(F(\mathbf{r}, \mathbf{i}_S))$$

Heaviside function



We obtain a "neural" ODE on top of which to perform event detection -> heyoka!

Biscani, Francesco, and Dario Izzo. "Reliable event detection for Taylor methods in astrodynamics." *Monthly Notices of the Royal Astronomical Society* 513.4 (2022): 4833-4844.


Eclipse events
DOPRI8 (tol = 2.2 · 10^−14)
Heyoka (ε = 10^−16)
error (m)
n. rotations
0
10
20
30
40
50
60
70

Thank you for listening!