Detecting dark matter in strong gravitational lenses with deep learning

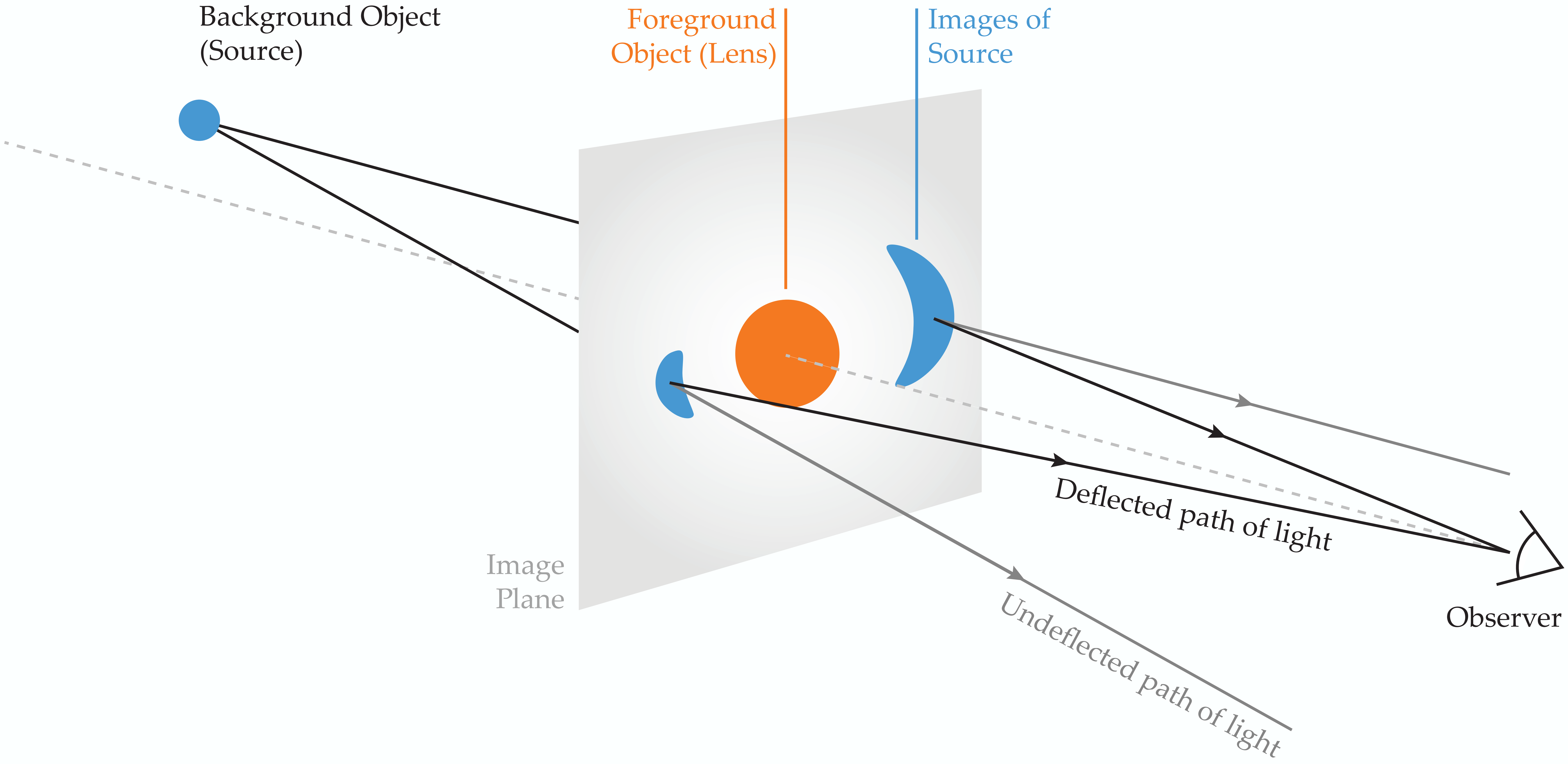
Conor O'Riordan, Simona Vegetti, Giulia Despali

AI goes MAD | IFT UAM | 15.06.22

MAX PLANCK INSTITUTE
FOR ASTROPHYSICS

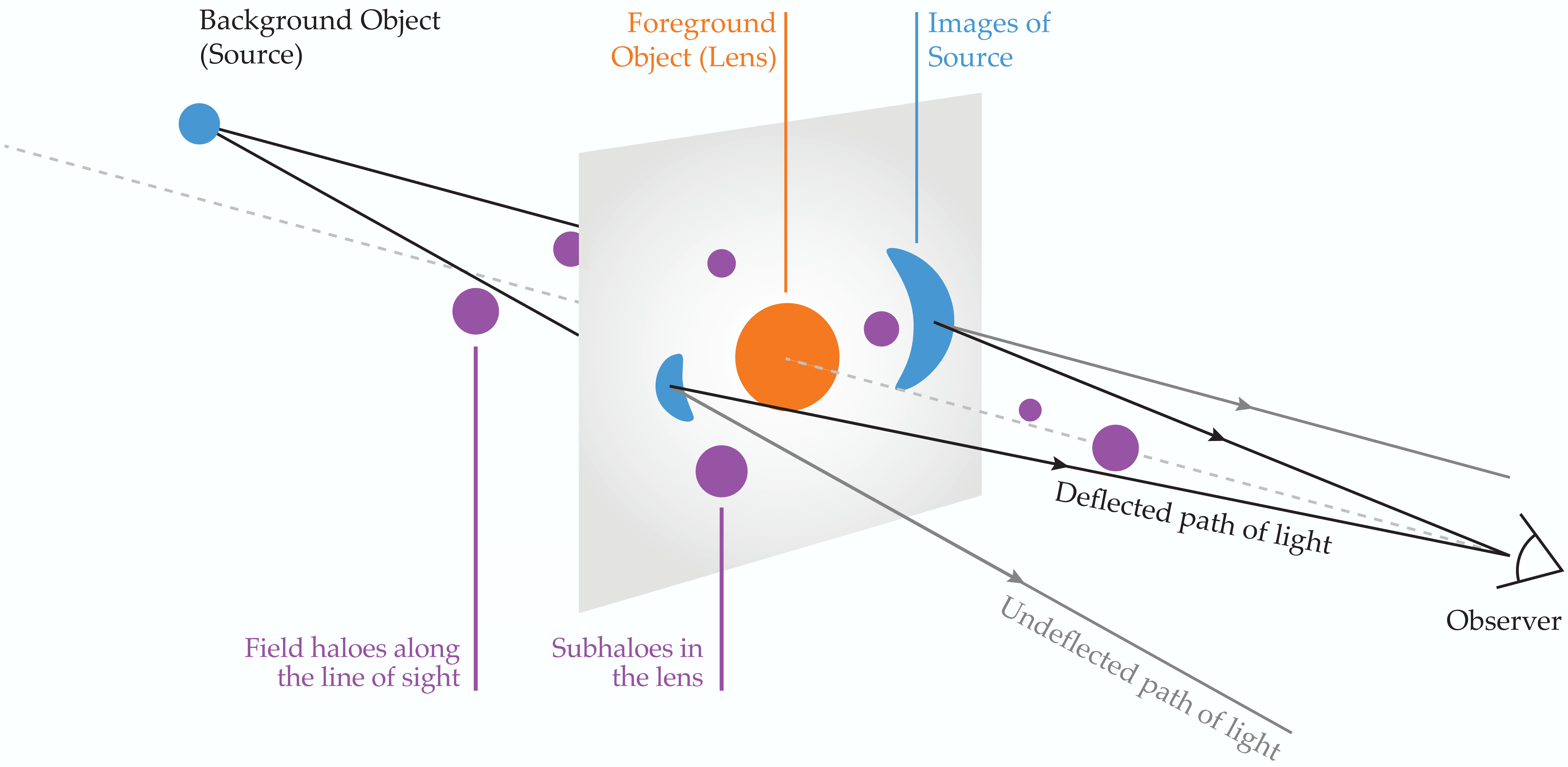
Background:

Strong gravitational lensing



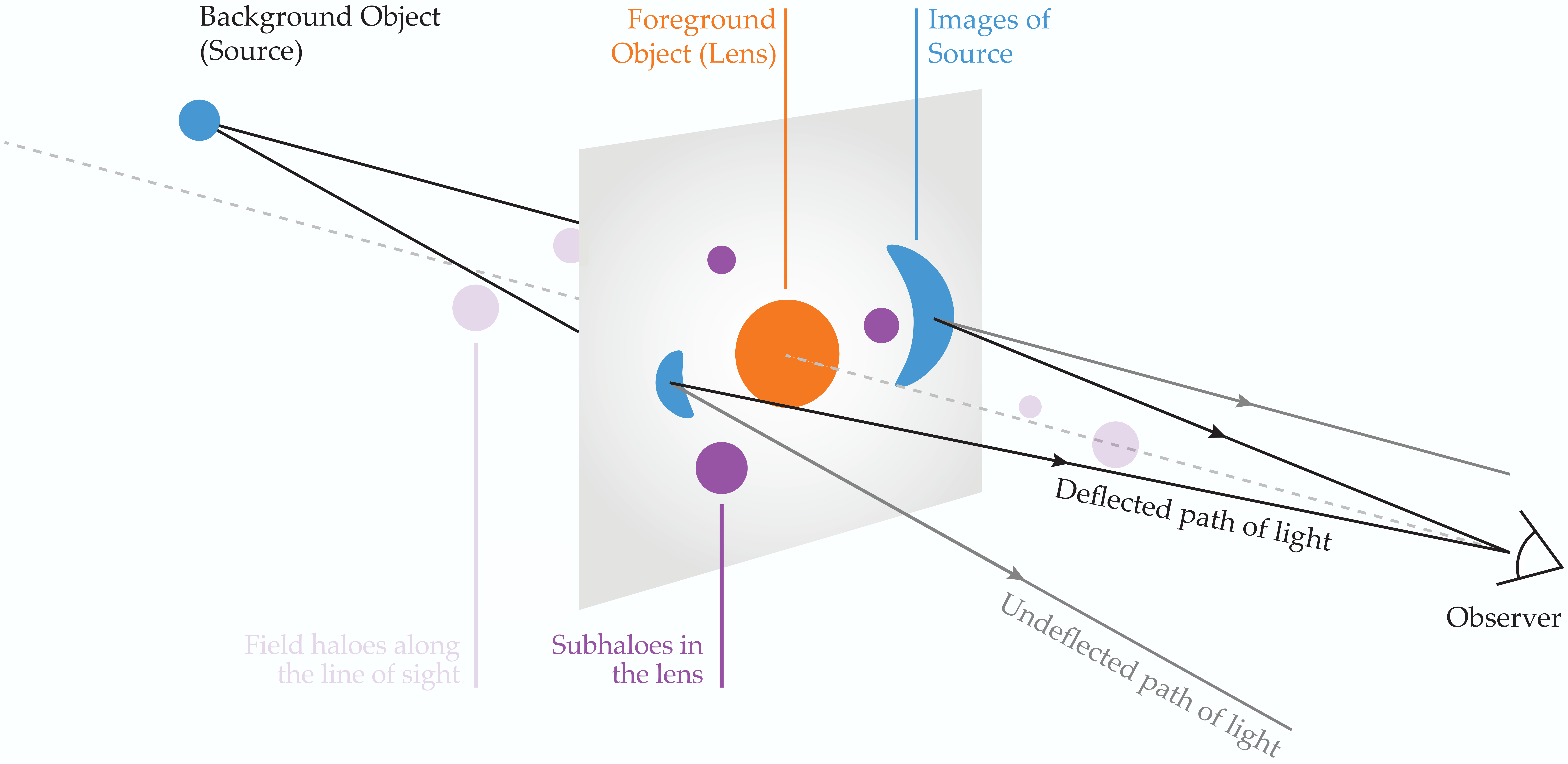
Background:

Strong gravitational lensing



Background:

Strong gravitational lensing



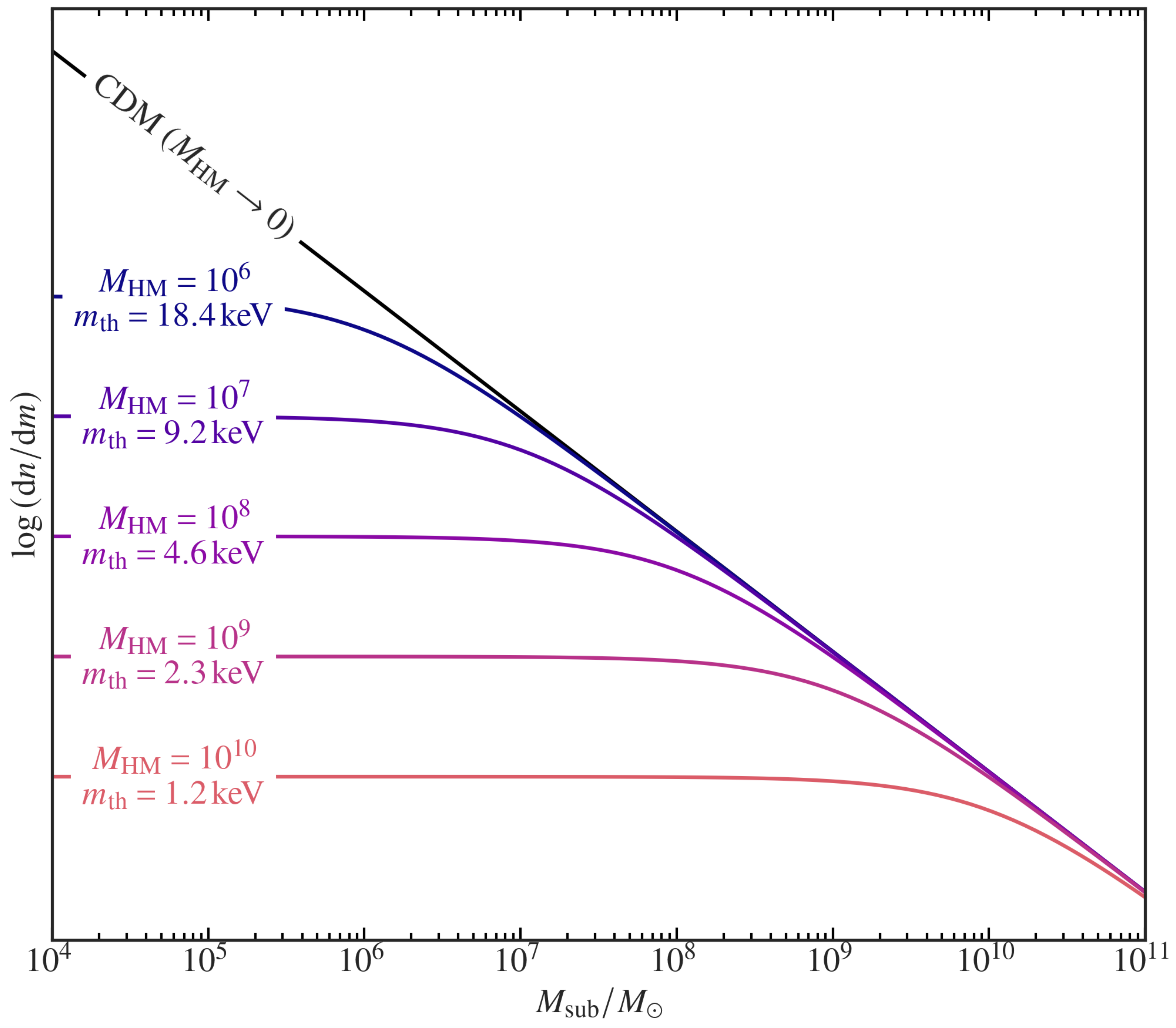
Background:

Dark matter substructure

DM models warmer than CDM predict a suppression in the formation of structure below a certain mass called the *half mode mass* or M_{HM}.

M_{HM} parametrises the DM model via the subhalo mass function.

Measuring this mass function (counting subhaloes) in the universe constrains the dark matter model.



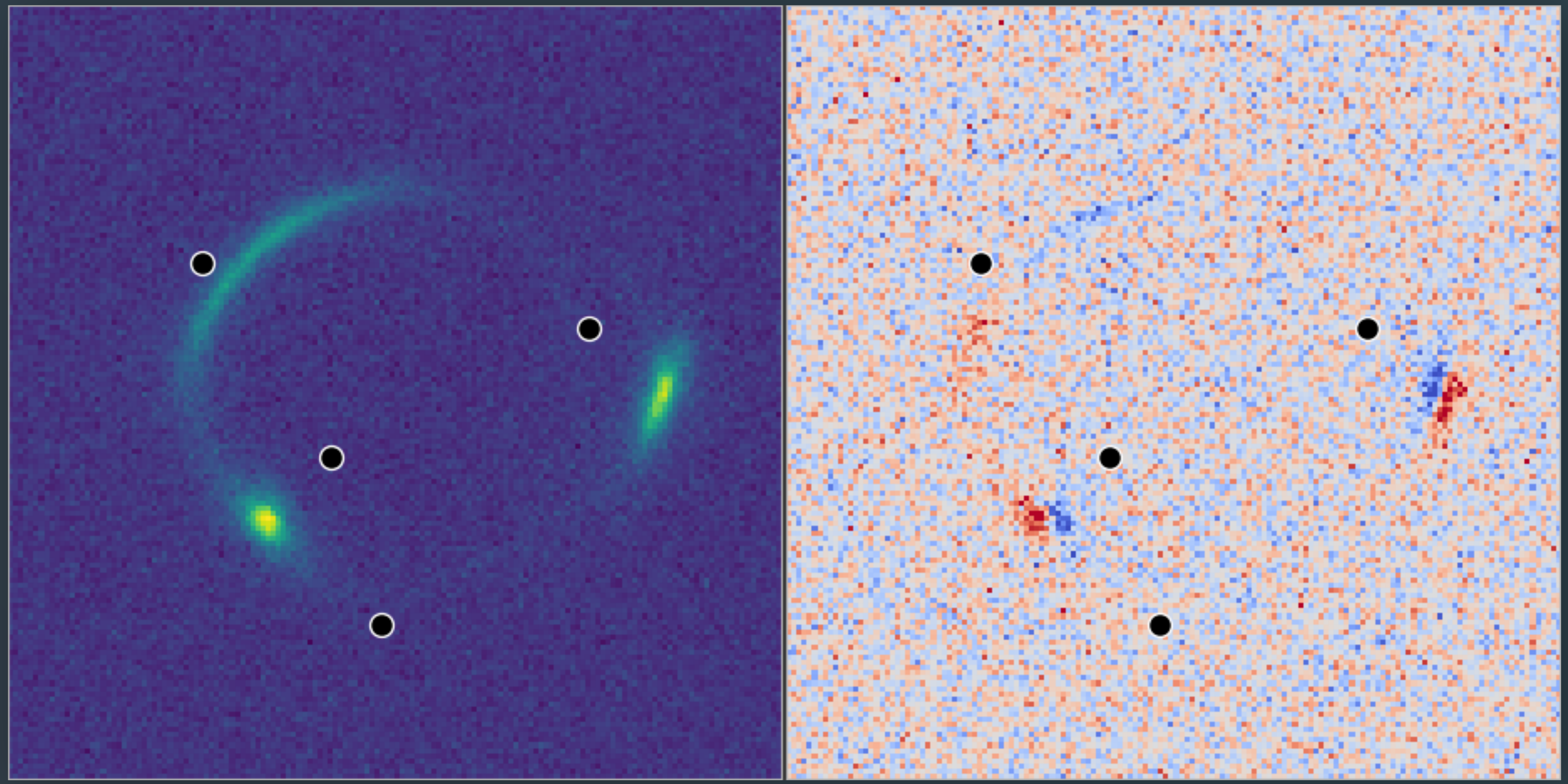
Background:

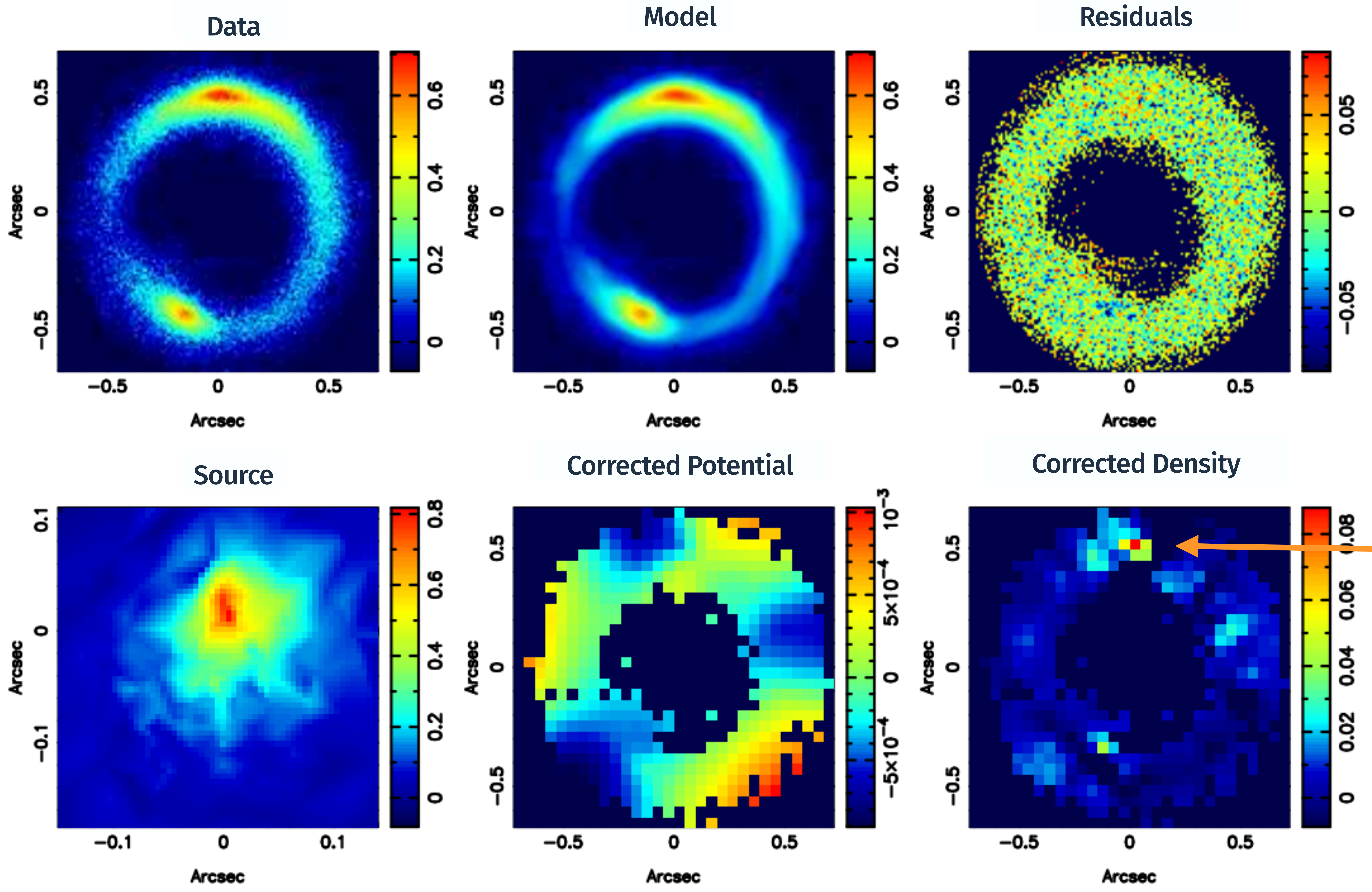
Gravitational imaging

Find the best *smooth model*

↓

Make corrections to the potential → Yields detections of subhaloes





$10^9 M_\odot$subhalo in the lens

Vegetti+ (2012)

Despali+ (2021)

Background:

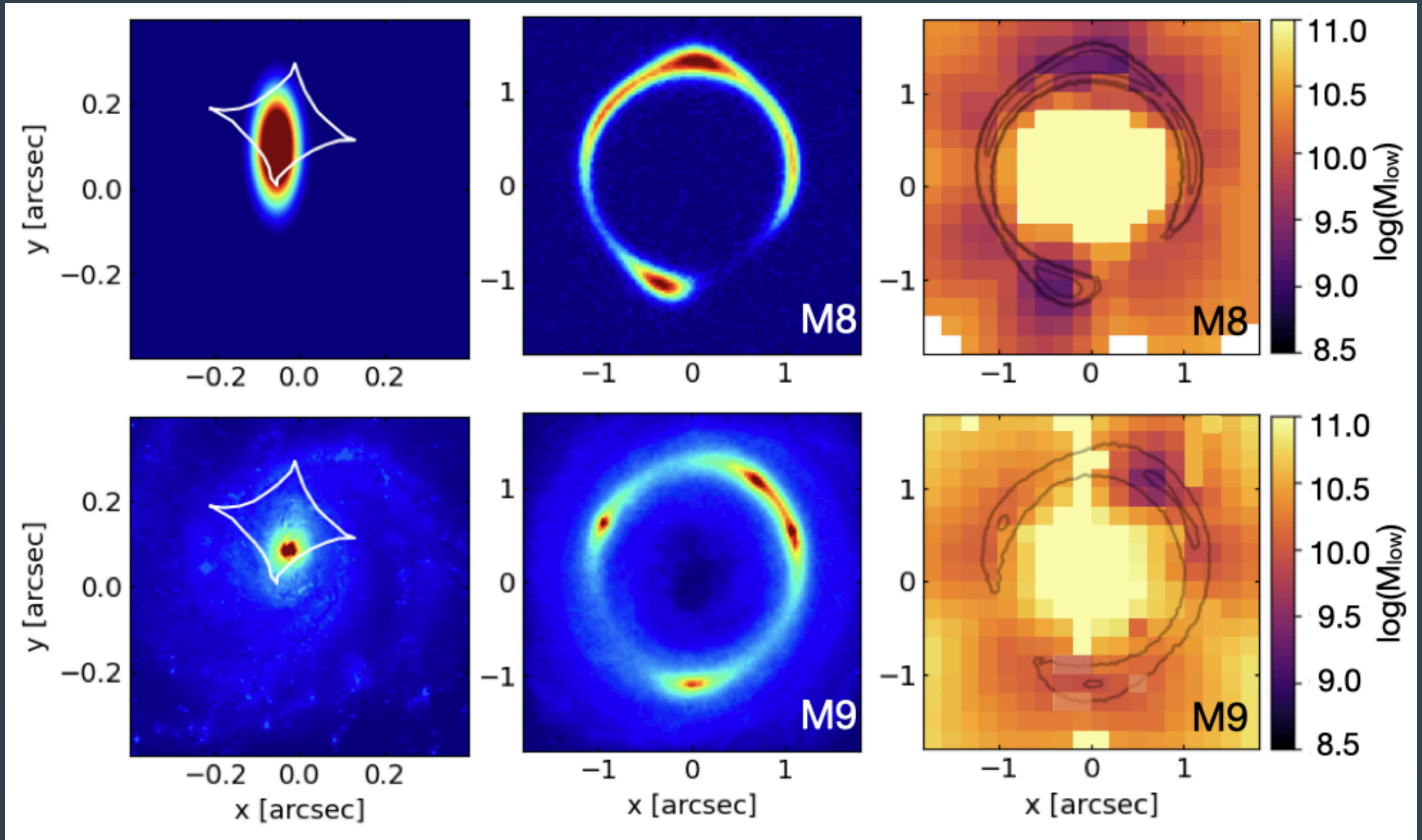
Gravitational imaging

Find the best *smooth model*

Make corrections to the potential → Yields detections of subhaloes

Quantify the *sensitivity* of each pixel → Places limits on non-detections

*Traditionally we compare the change in evidence for a subhalo in every pixel, for every mass → Very expensive to calculate → Replace/approximate with a machine learning method



Background:

Upcoming surveys

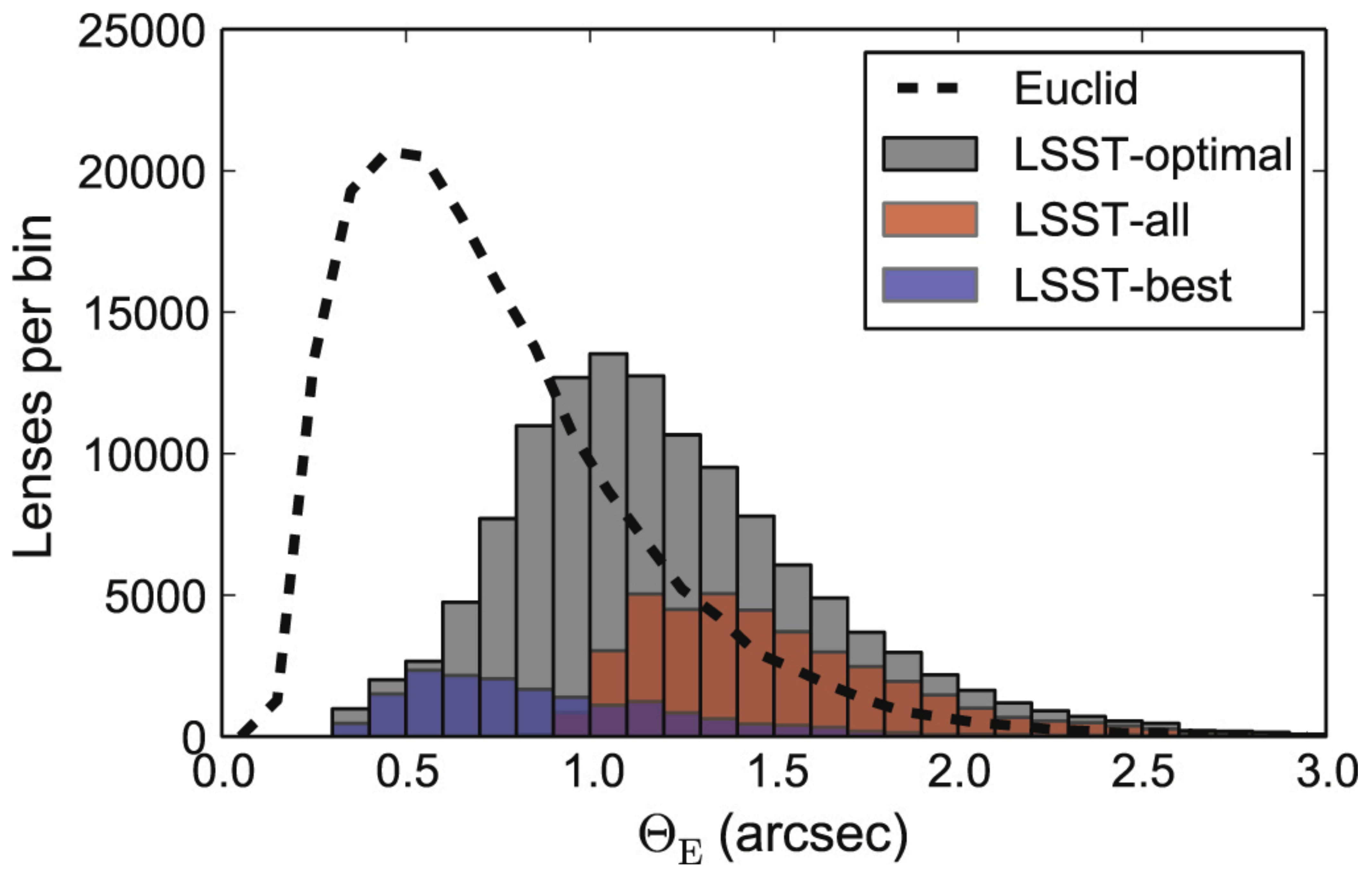
Currently known strong lenses

~10^2

Euclid, *DES* and *Vera Rubin* will increase this to

~10^5

Collett (2015)



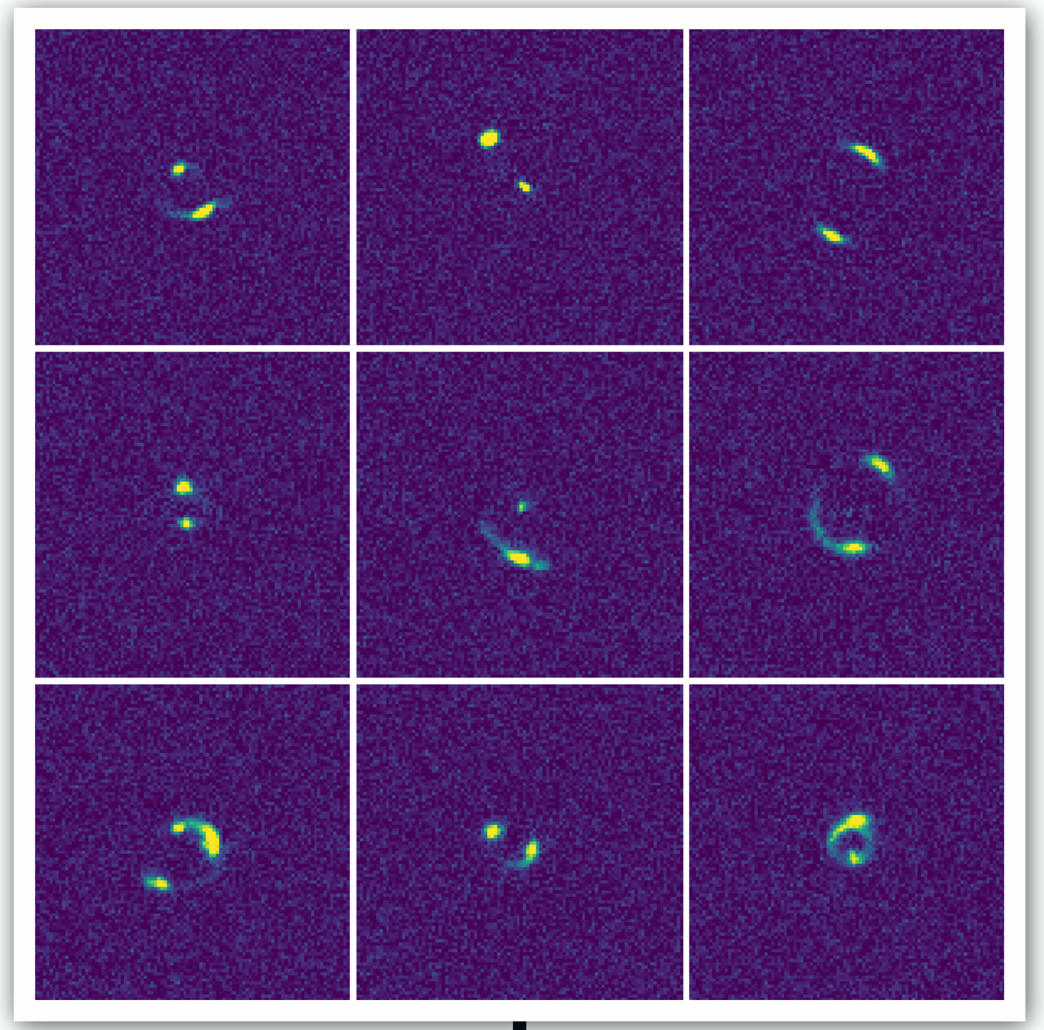
Method:

Architecture and data

Network gives a simple binary probability of subhalo/no subhalo for each image

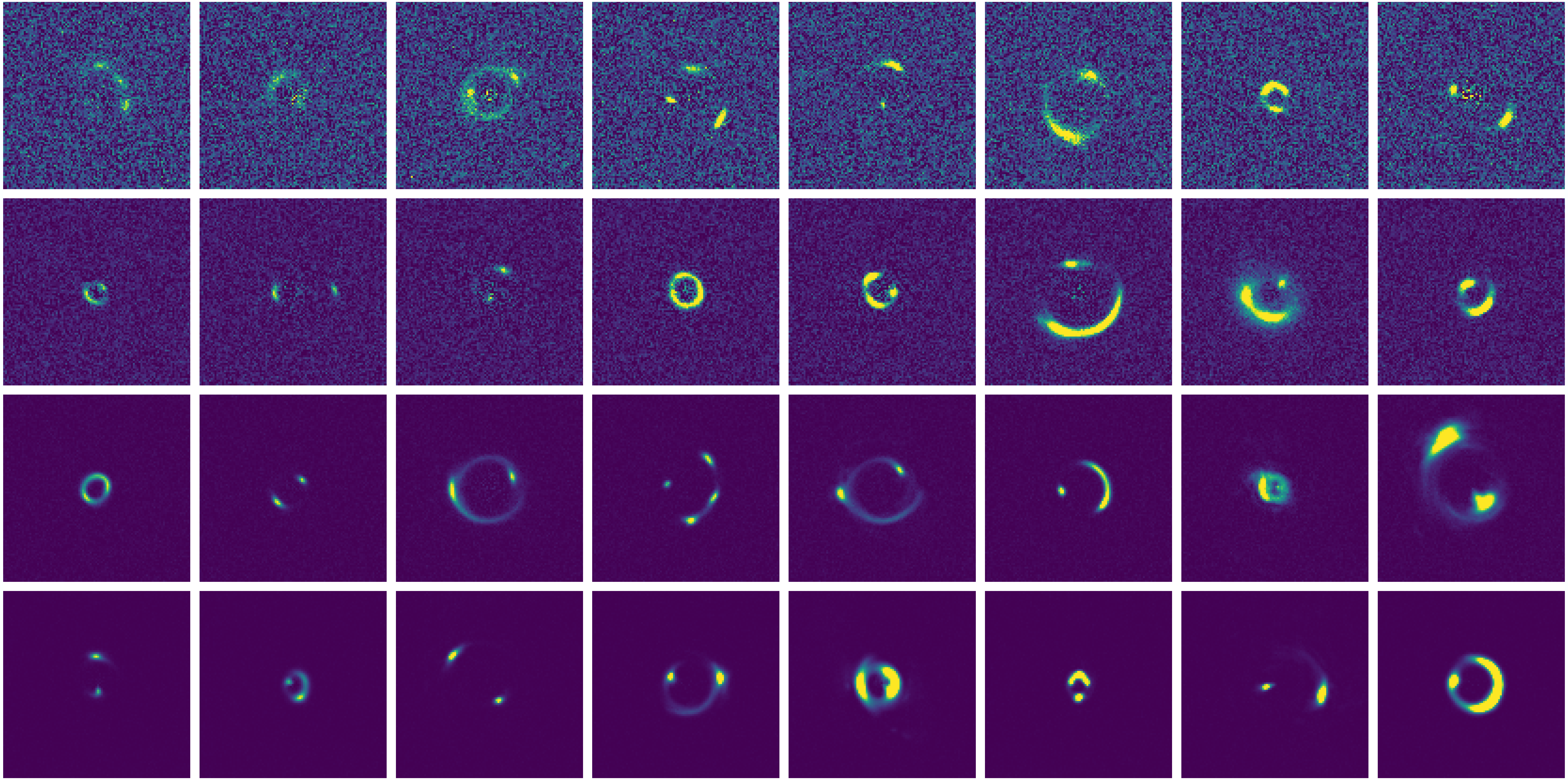
Training data has:

- Hubble deep field sources (and redshifts)
- Elliptical power-law lens
- Euclid pixel size, noise, PSF
- Range of subhalo masses and concentrations
- Range of source and lens magnitudes
- External shear
- (Poisson-limited) Lens subtraction
- Either one or “some” (1-4) subhaloes, randomly placed



ResNet50

$$\frac{\Pr\left(\text{Subhalo} \,|\, D\right)}{\Pr\left(\text{No subhalo} \,|\, D\right)}$$



A sample of training data, ordered by S/N

	S/N		$M_{\rm sub}/M_\odot$	
	Min.	Max.	Min.	Max.
1	10^2	10^3	10^{11}	-
2	10^2	10^3	10^9	10^{11}
3	20	10^3	10^9	10^{11}
4*	20	10^3	10^9	10^{11}
5*	20	10^3	$10^{8.6}$	10^{11}

*Stages 4 and 5 add external shear. This can produce a very similar magnification effect to substructure

Method:

Training procedure

We train in five stages, making the data more complex, and the classification problem harder at each stage.

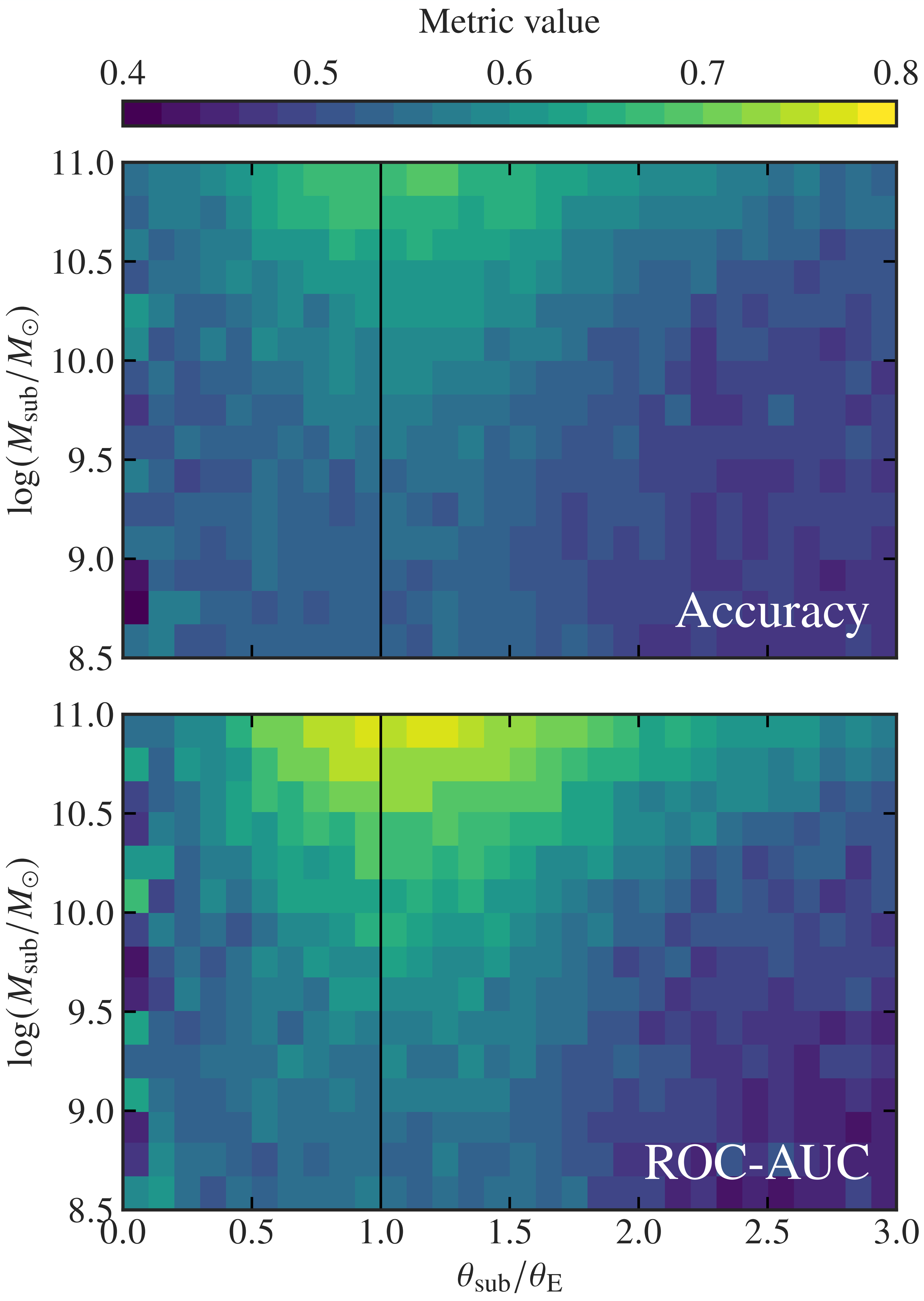
Each stage has 2M images and starts from the weights of the previous stage, trained until convergence.

~1000 total epochs, ~200 GPU hours total

Method:

Model performance

Is, at a glance, terrible...

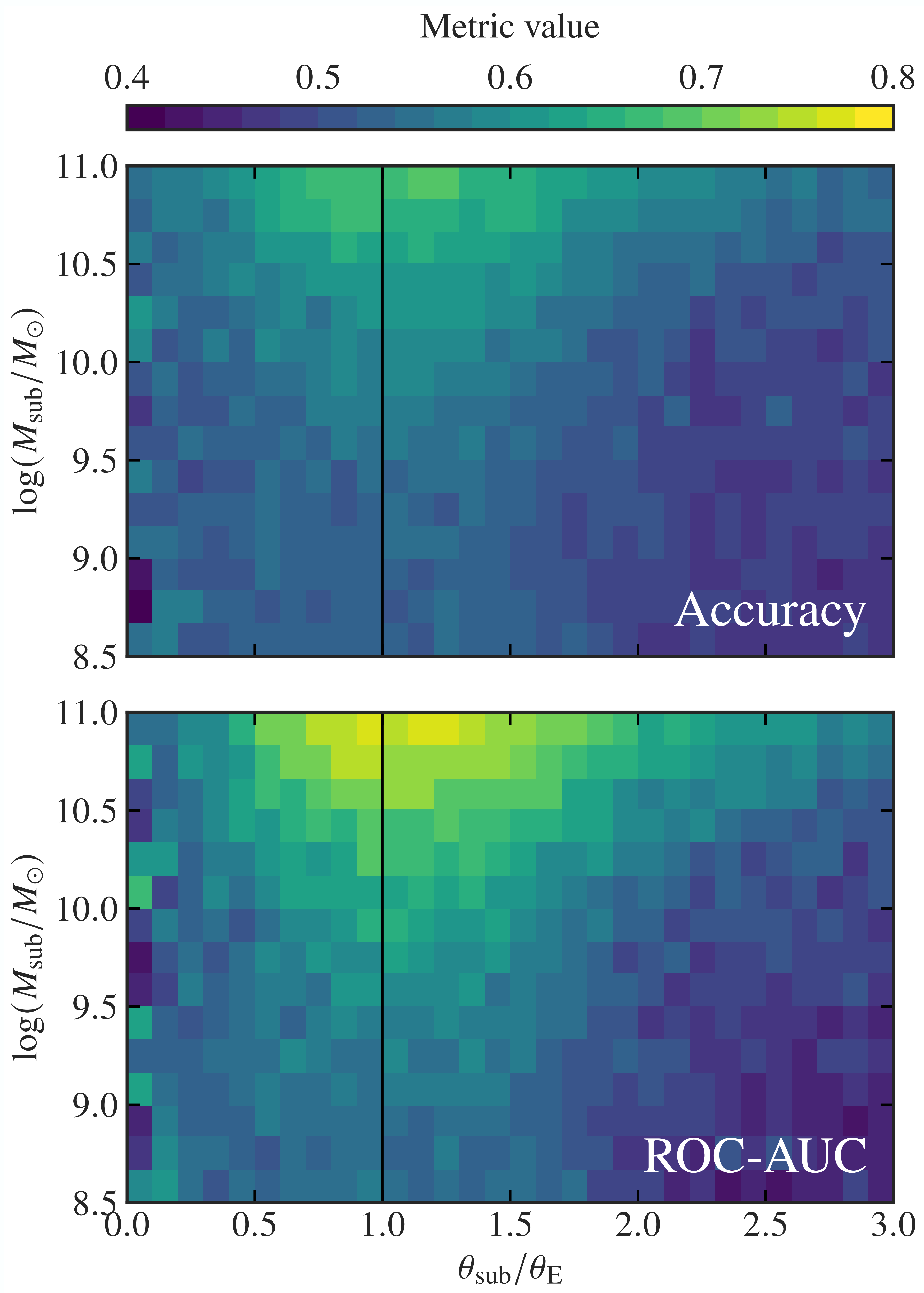


Method:

Model performance

Is, at a glance, terrible…

But most of the data the network sees are simply not sensitive to the subhaloes shown.



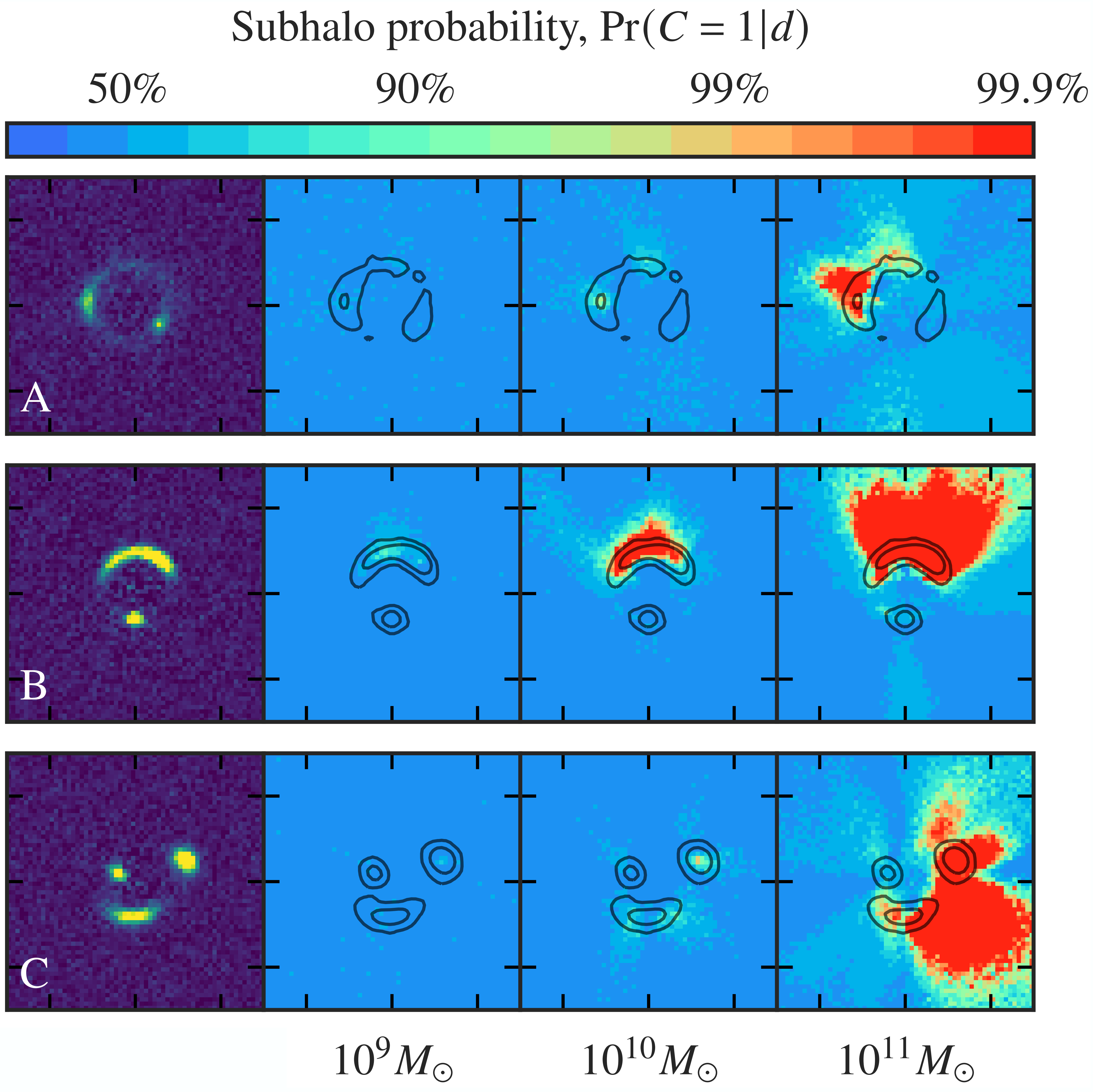
Method:

Model performance

Is, at a glance, terrible...

But most of the data the network sees are simply not sensitive to the subhaloes shown.

The network learns to be incredibly cautious about making detections, but is able to do so accurately in the situations we expect.



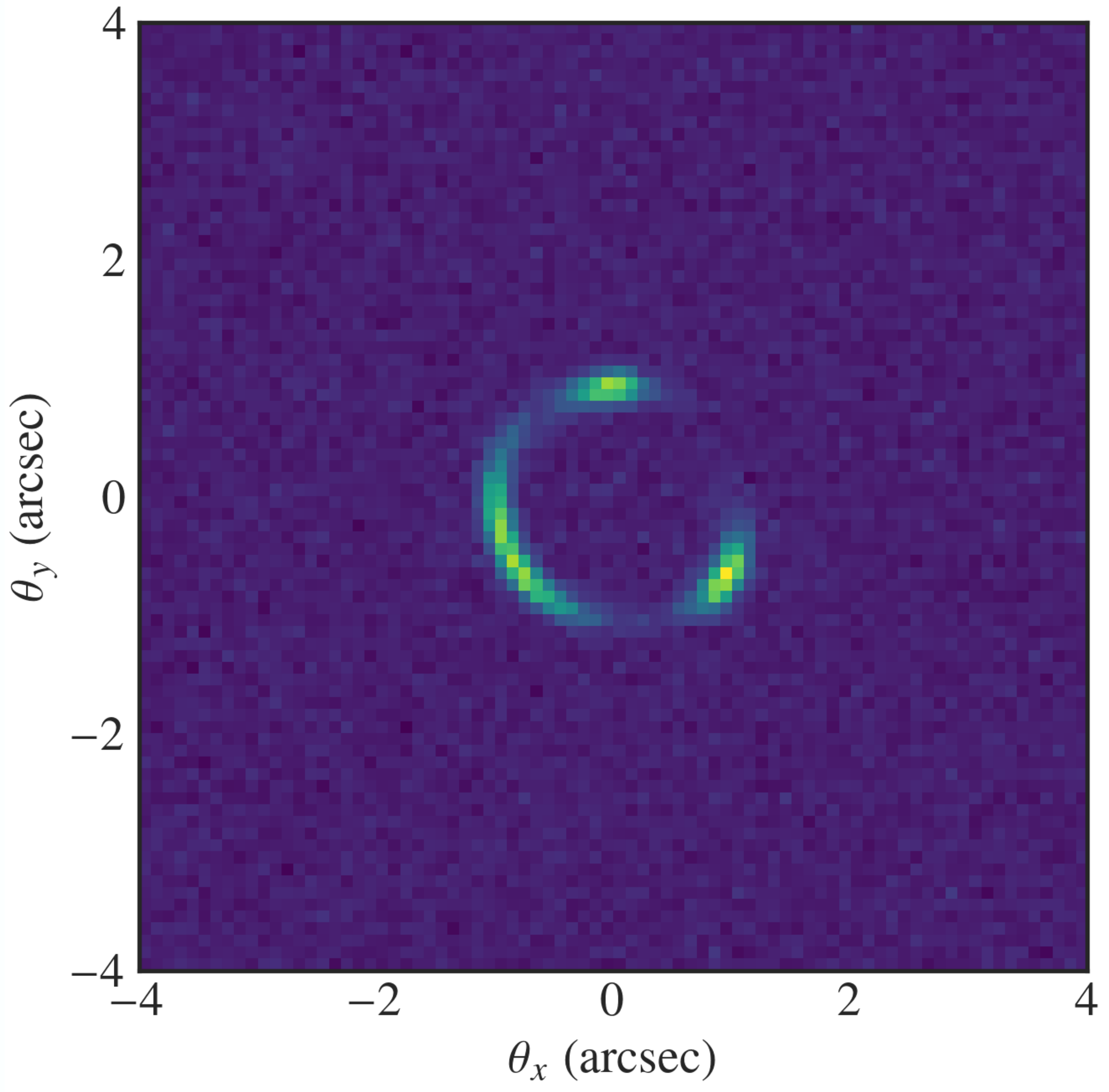
Method:

Estimating sensitivity

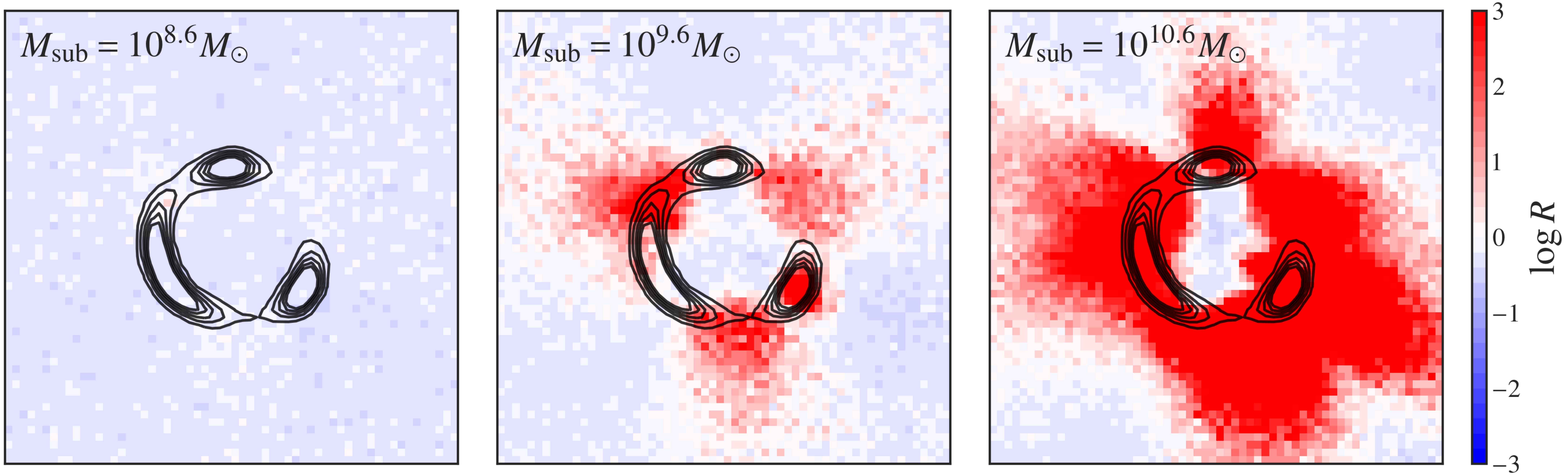
The network as trained can give us the sensitivity for an individual system

We create realisations of the same system with a sub-halo in each pixel, over a range of subhalo masses

Lens model, source, noise realisation etc stay the same

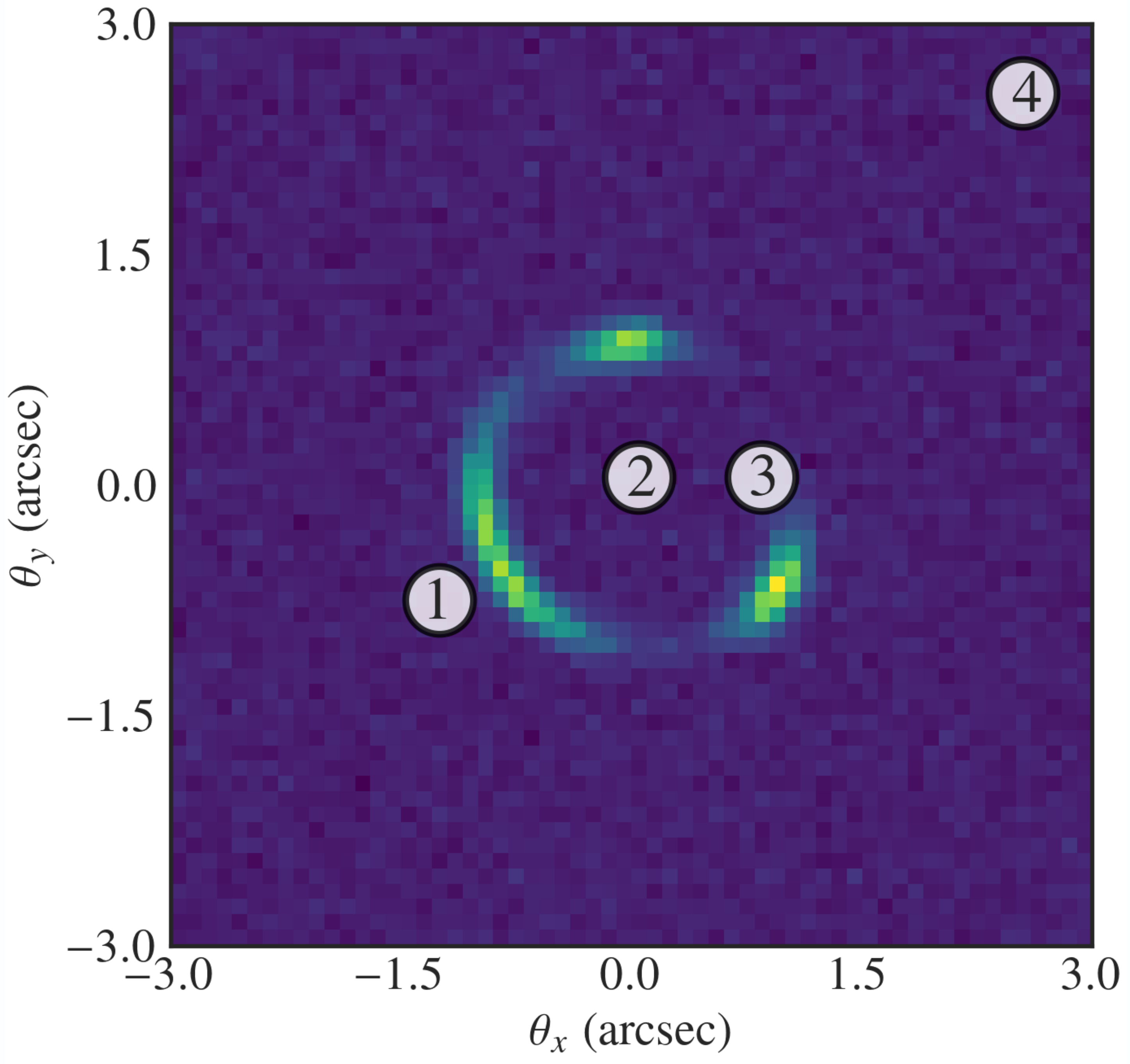


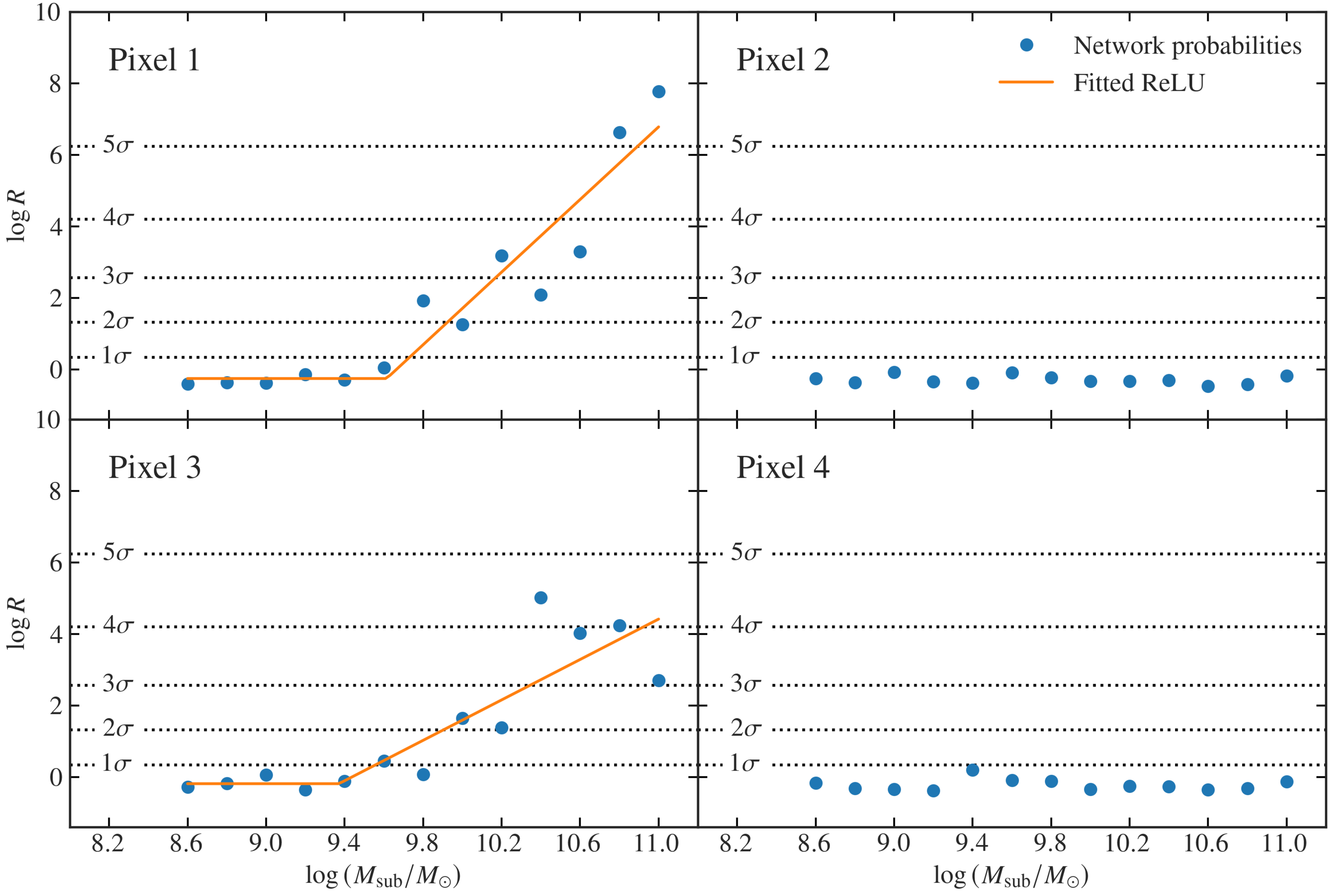
We run every realisation through the trained network and produce a map of the detection odds for each subhalo mass



Plotted quantity is $\log R = \log \left[\frac{\Pr(\text{Subhalo})}{\Pr(\text{No subhalo})} \right] \Rightarrow$ negative values predict nothing
positive values predict substructure

We obtain the odds R of detecting a subhalo as a function of mass in every pixel





We fit a rectified linear unit (ReLU) function:

$$\log R = \max\left[\log R_0, a\log\left(M_{\mathrm{sub}} - M_0\right) + \log R_0\right]$$

and find the mass at which a given odds threshold is reached

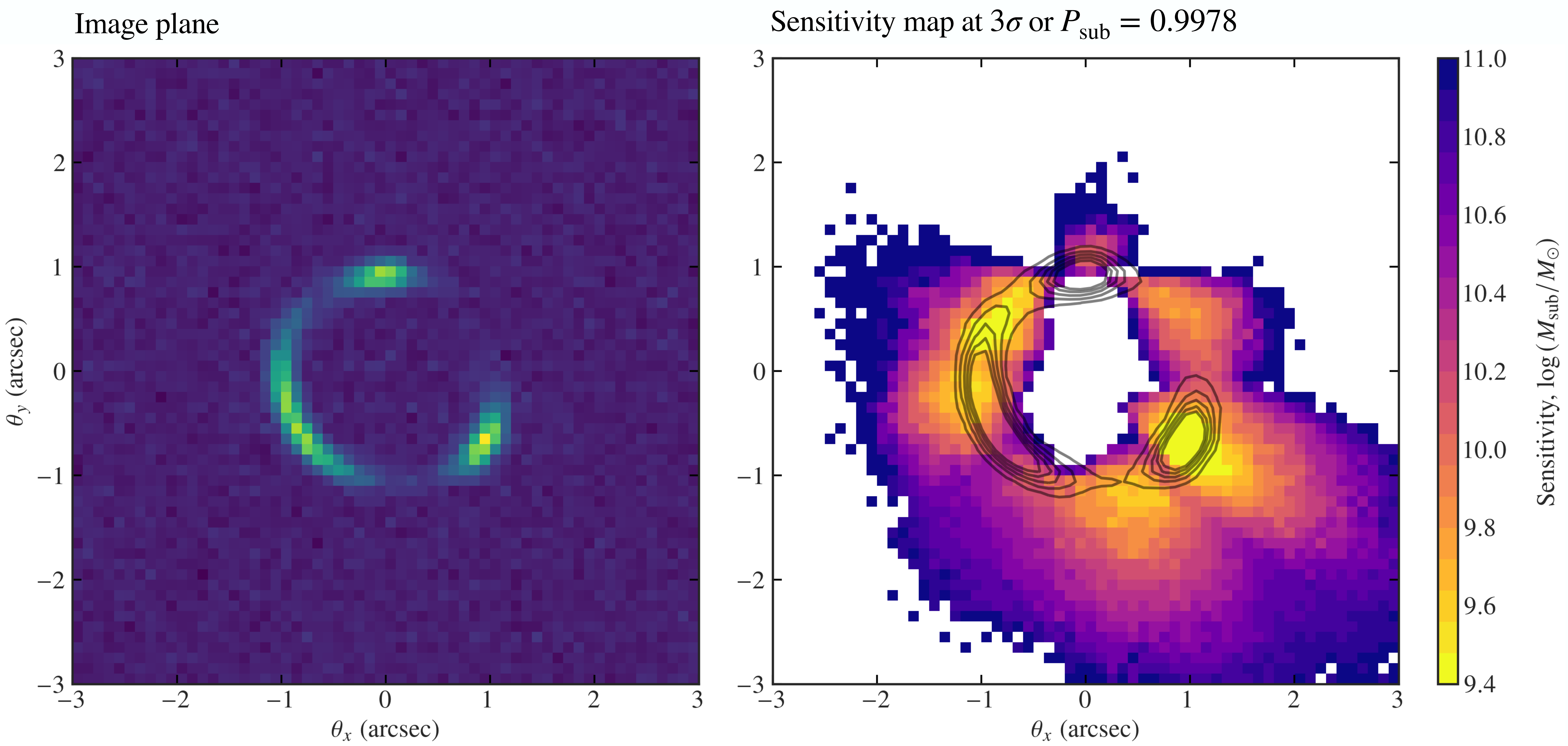
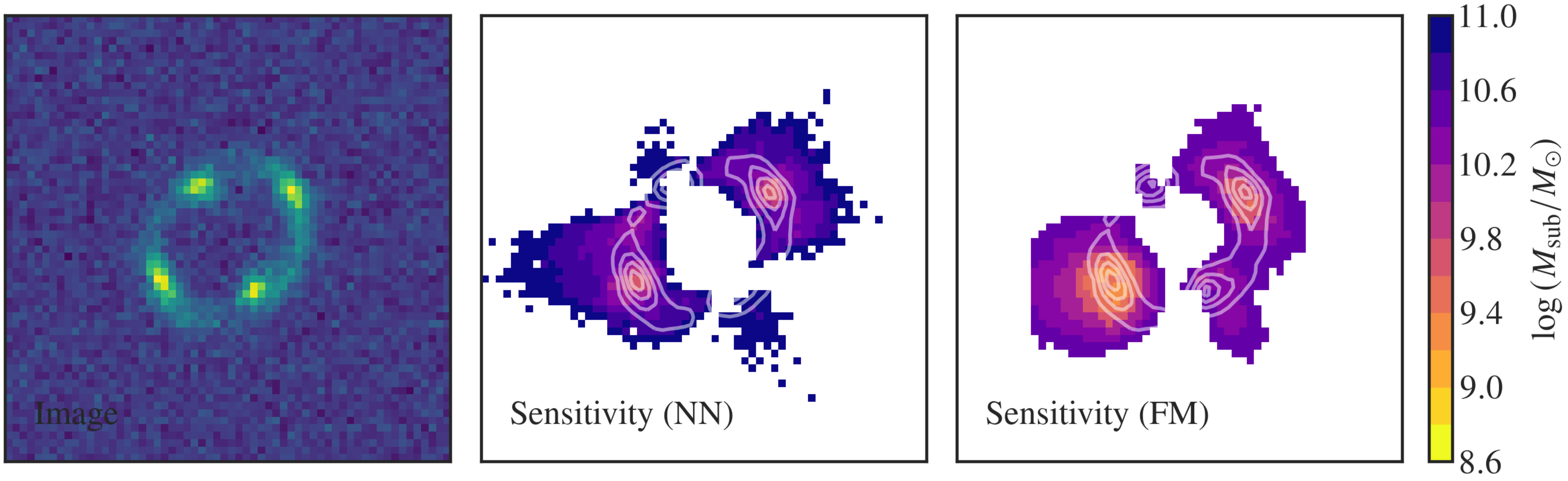


Image plane
Sensitivity map at 3σ or $P_{\text{sub}} = 0.9978$
θ_y (arcsec)
θ_x (arcsec)
θ_x (arcsec)
Sensitivity, $\log(M_{\text{sub}}/M_{\odot})$
11.0
10.8
10.6
10.4
10.2
10.0
9.8
9.6
9.4

Method:

Comparison of methods



Expected number of subhaloes detectable at 3σ in CDM

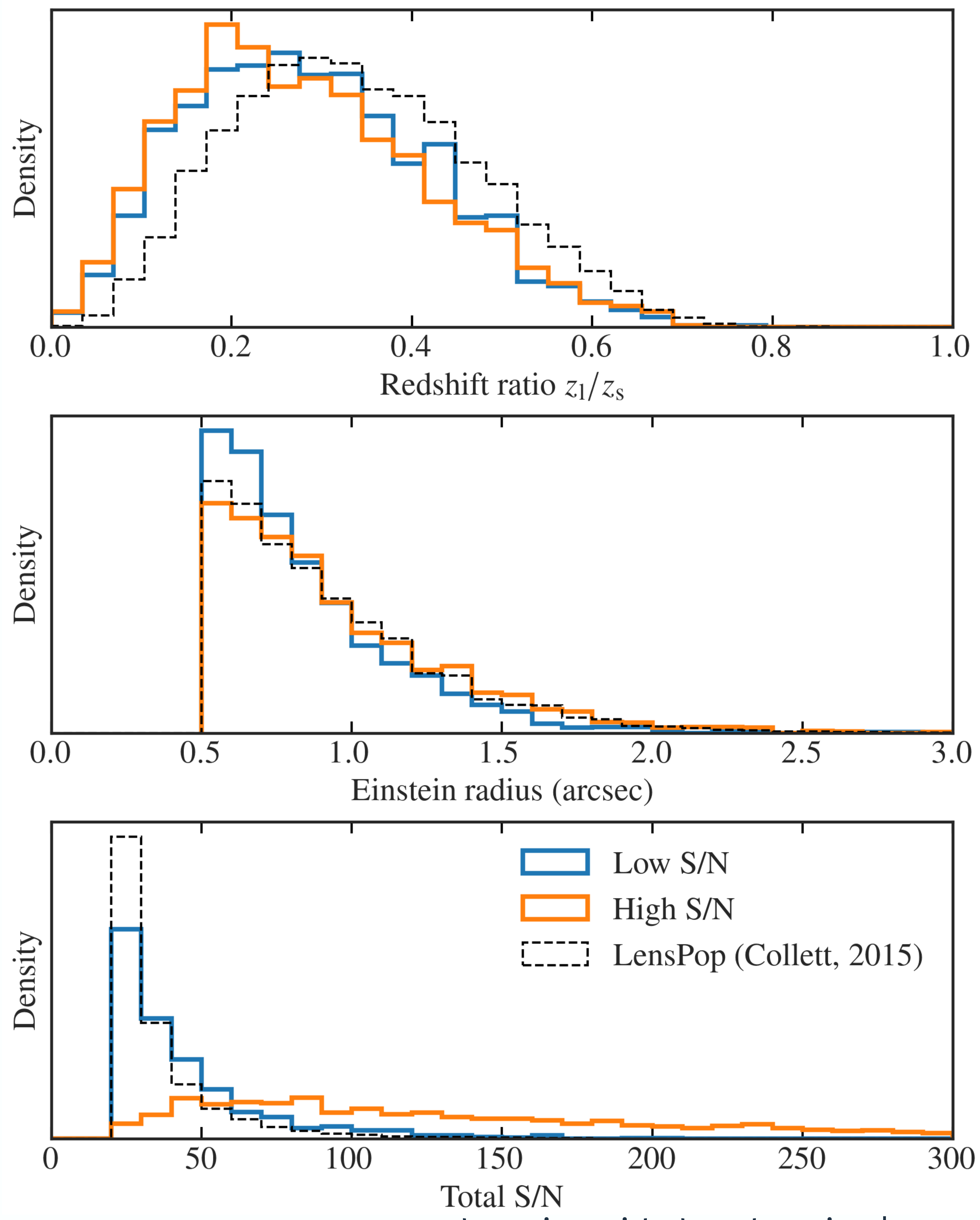
$$\mu_{\rm sub} = 44.1 \qquad \mu_{\rm sub} = 42.5$$

Results:

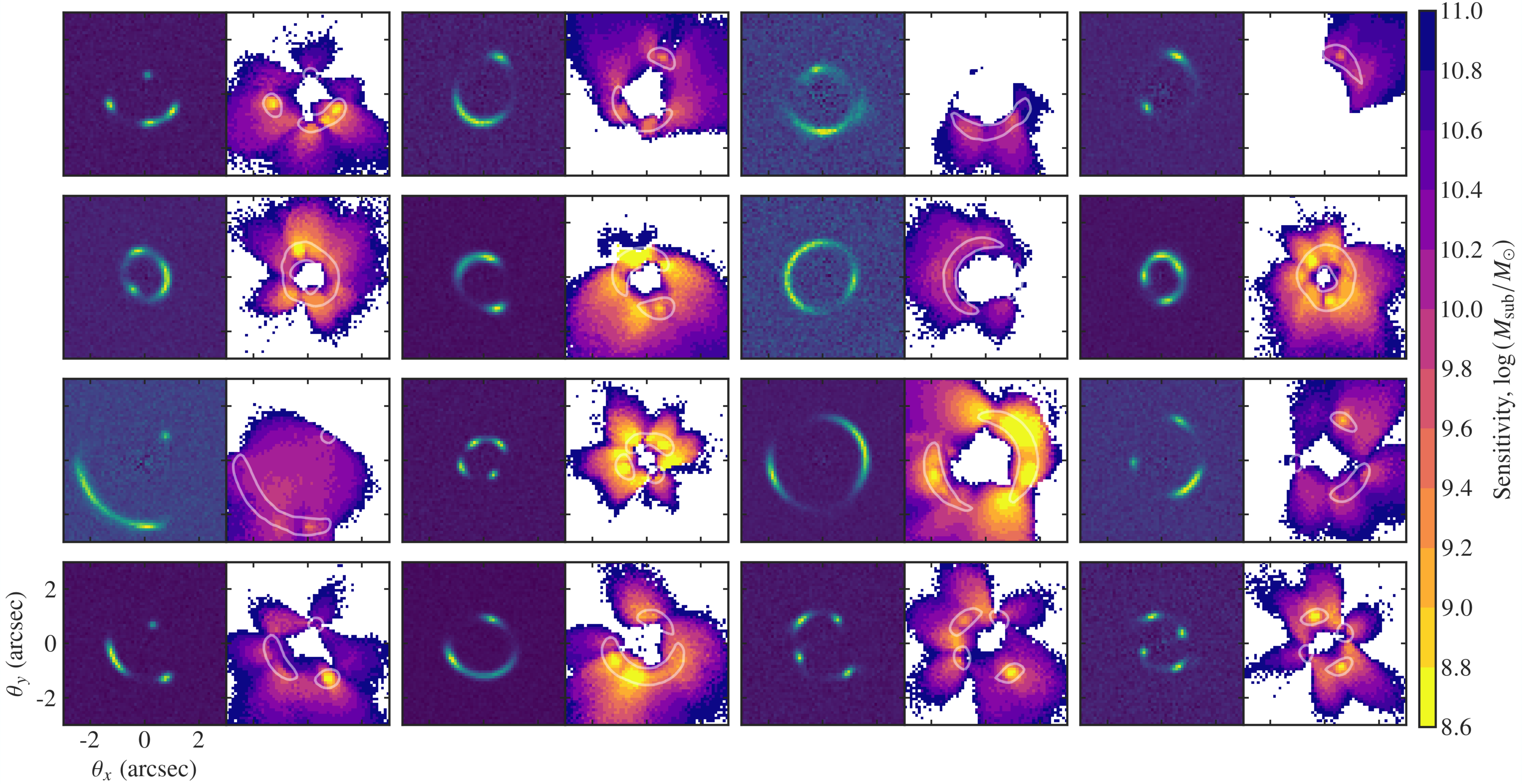
Sensitivity in Euclid

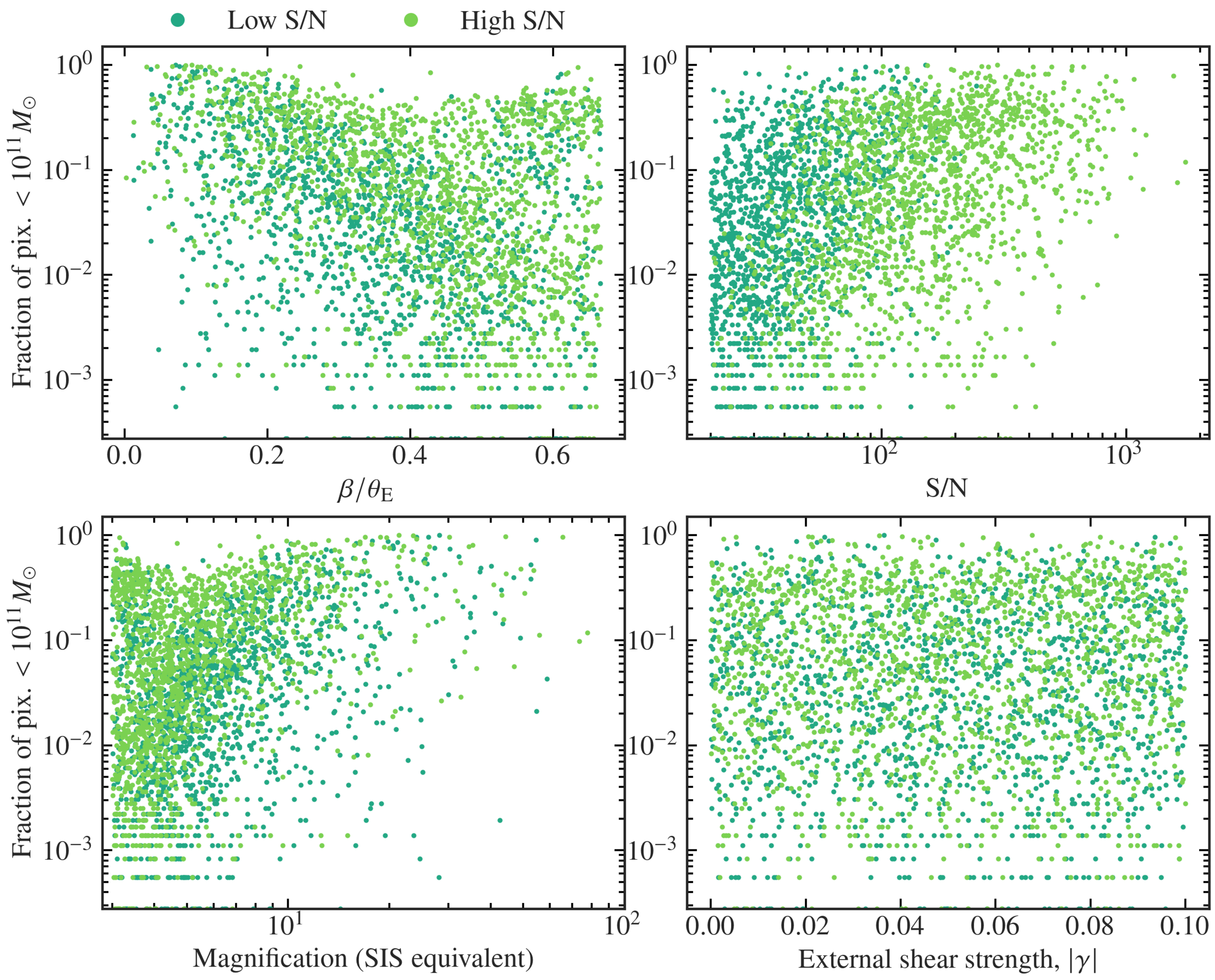
Following *LensPop* (Collett, 2015) we simulate a sample of 20k Euclid strong lenses and find the sensitivity in detail.

Where a sensitivity map used to take weeks to compute, we can now run one in 30 mins on one GPU.

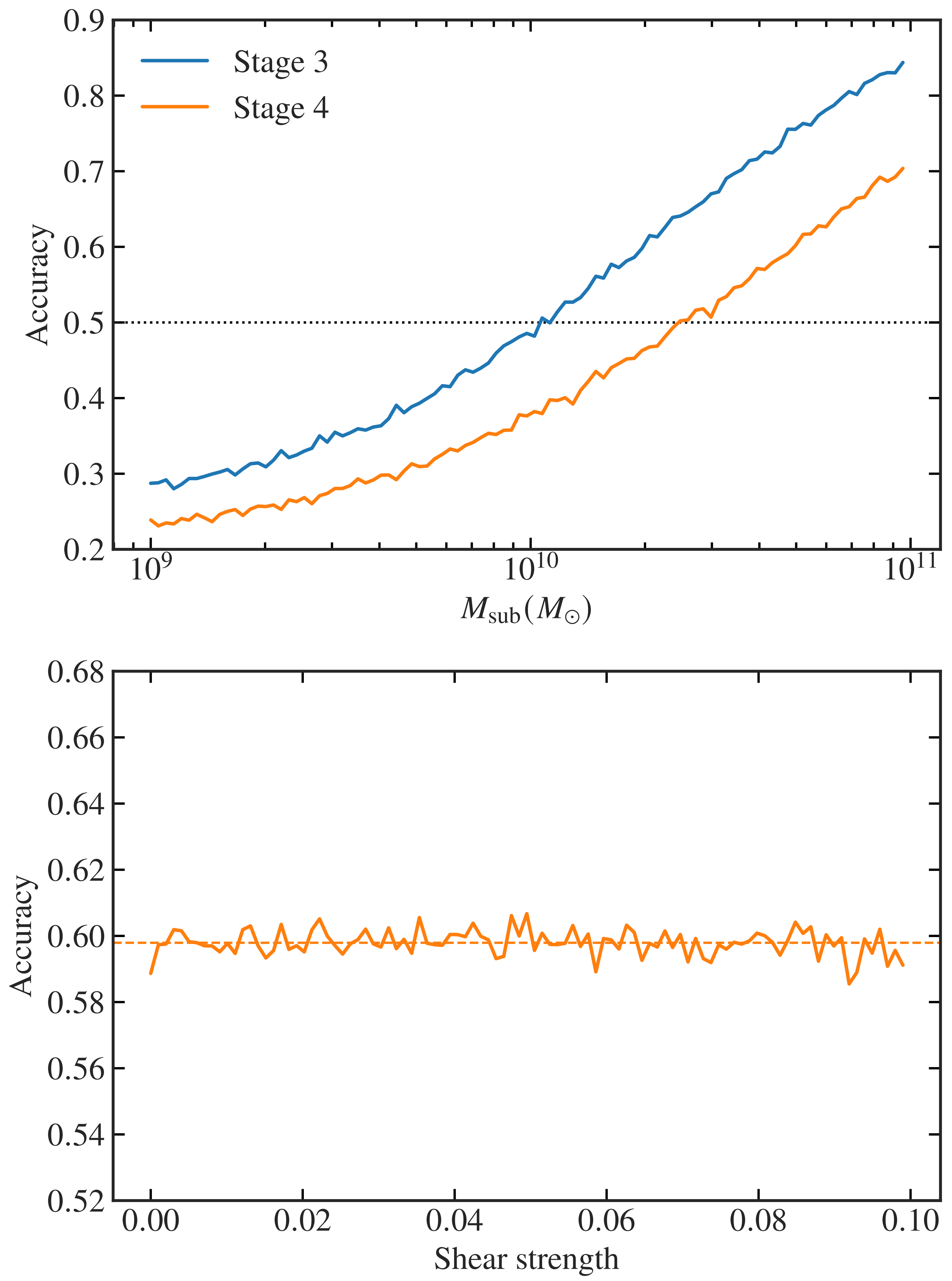


A sample of the most sensitive observations





A large number of sensitivity maps allows us to mine for correlations with the lensing parameters.



Results:

Learning degeneracies

Once external shear is added to the model, the confidence of the network's predictions drops, no matter the specific shear strength.

The accuracy does not correlate with shear strength - implying the network is not confusing shear for substructure but has learnt to account for it.

Results:

Detection statistics

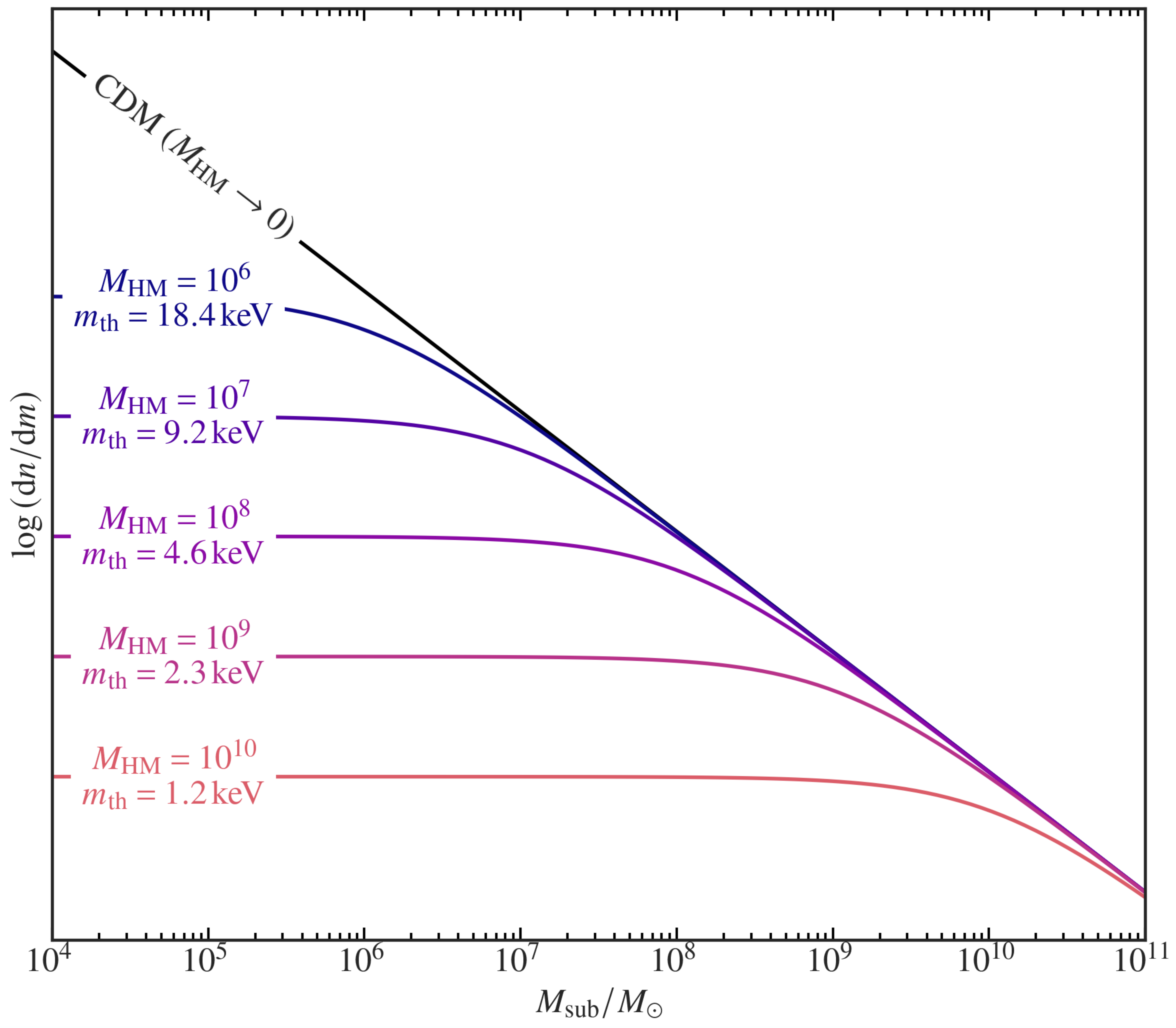
The sensitivity in each pixel tells us the minimum mass subhalo we could detect. We can integrate the mass function,

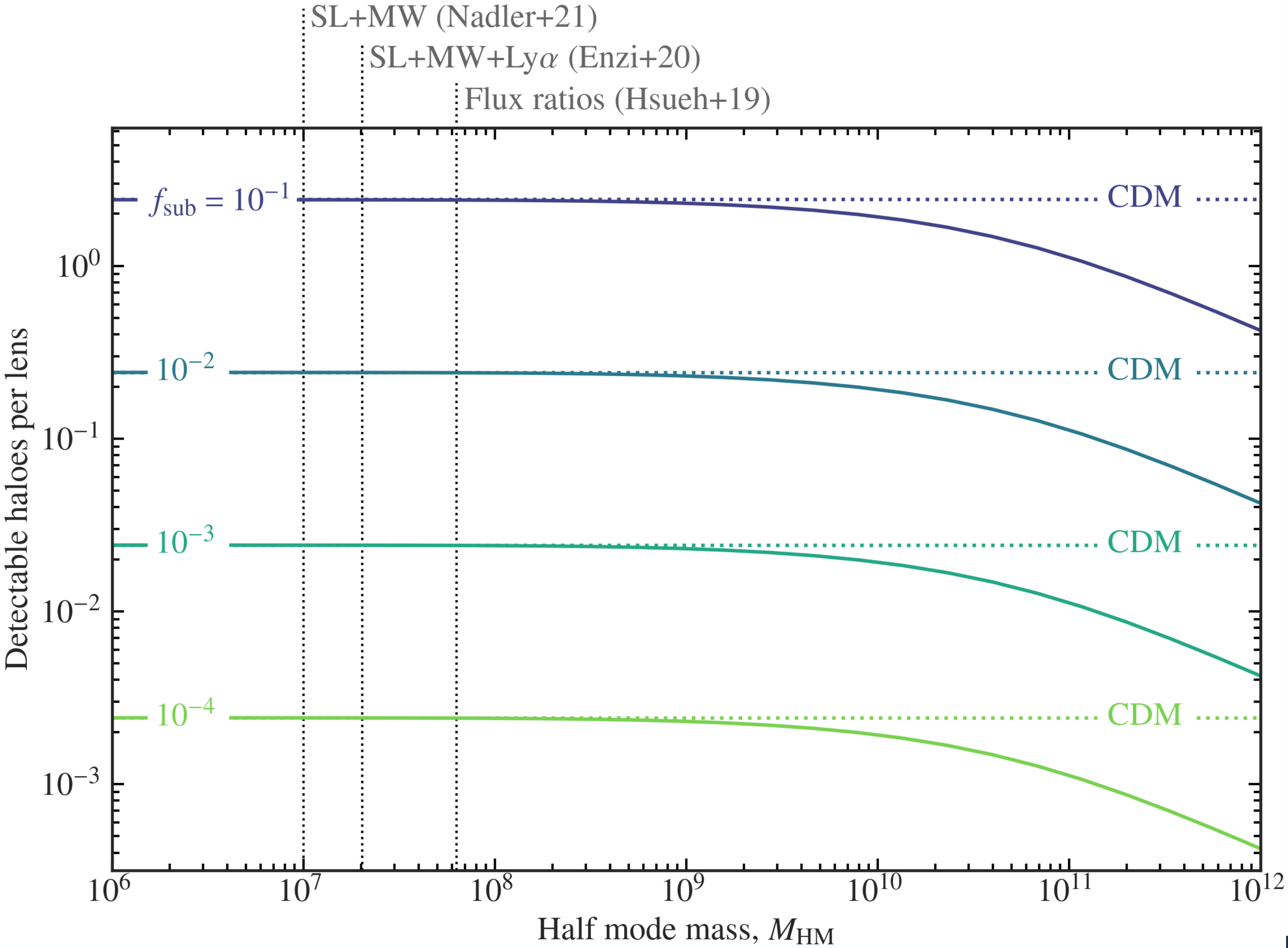
$$\frac{\mathrm{d}n}{\mathrm{d}m} \propto m^{-\alpha_1}\left[1+\alpha_2\frac{M_{\mathrm{HM}}}{m}\right]^{\gamma}$$

Lovell (2020)

to find the expected number, μ_{sub}, of *detectable* subhaloes per pixel (and by summing up, per system)

The mass function is normalised by f_{sub}, the fraction of mass in substructure within $2\theta_{\mathrm{E}}$



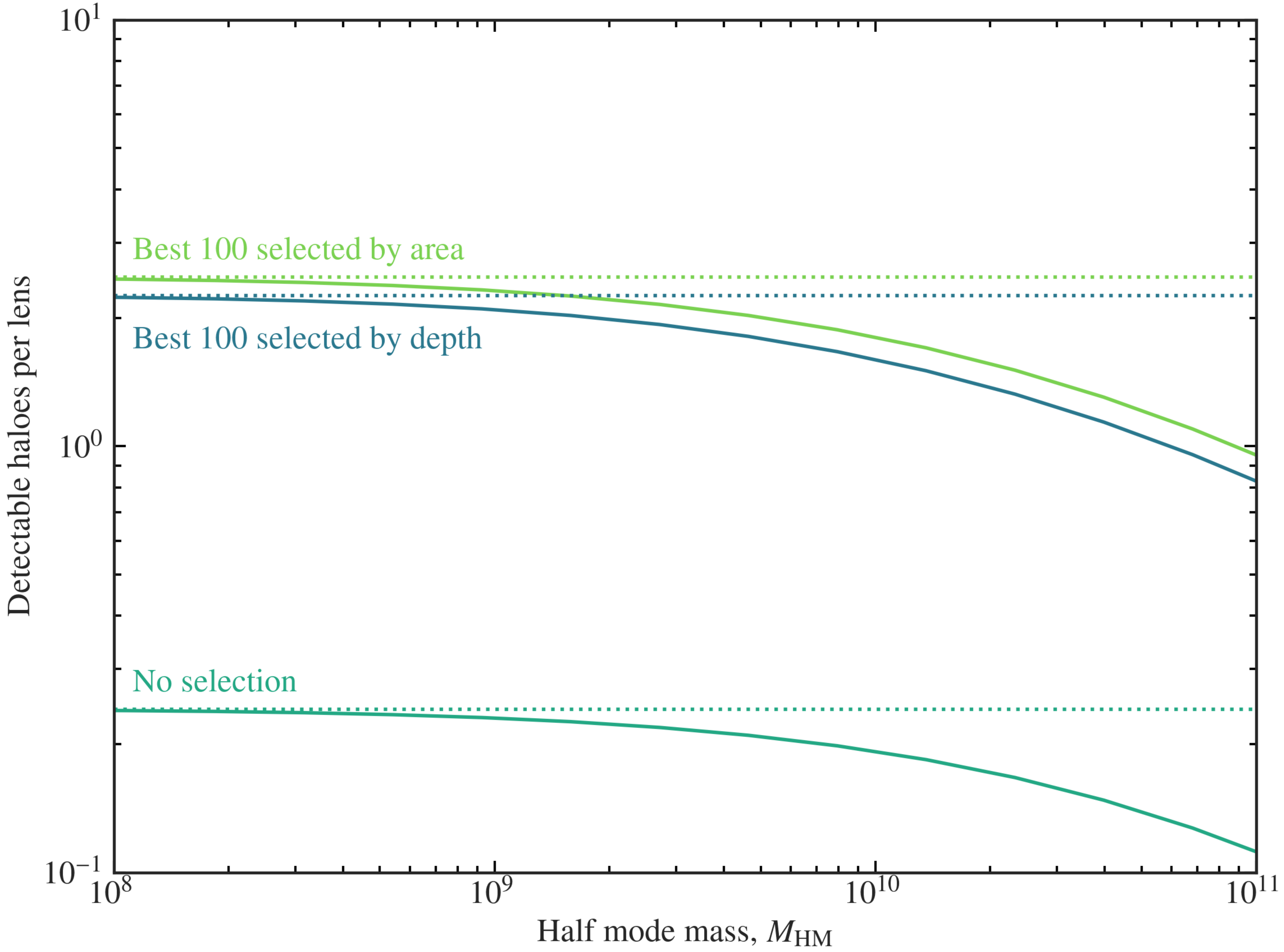


The expected number of detectable haloes does not change with respect to CDM for $M_{\rm HM} \lesssim 10^8 M_\odot$

Other studies have already ruled out models warmer than this limit (various 95% CLs shown)

Sensitivity depth: minimum sensitivity in a system

Sensitivity area: total area sensitive within mass range



Selecting by area or depth greatly improves the possible constraints on $f_{\rm sub}$ (fixed here to 10^{-2})

Either selection can (very) marginally improve the constraints on $M_{\rm HM}$

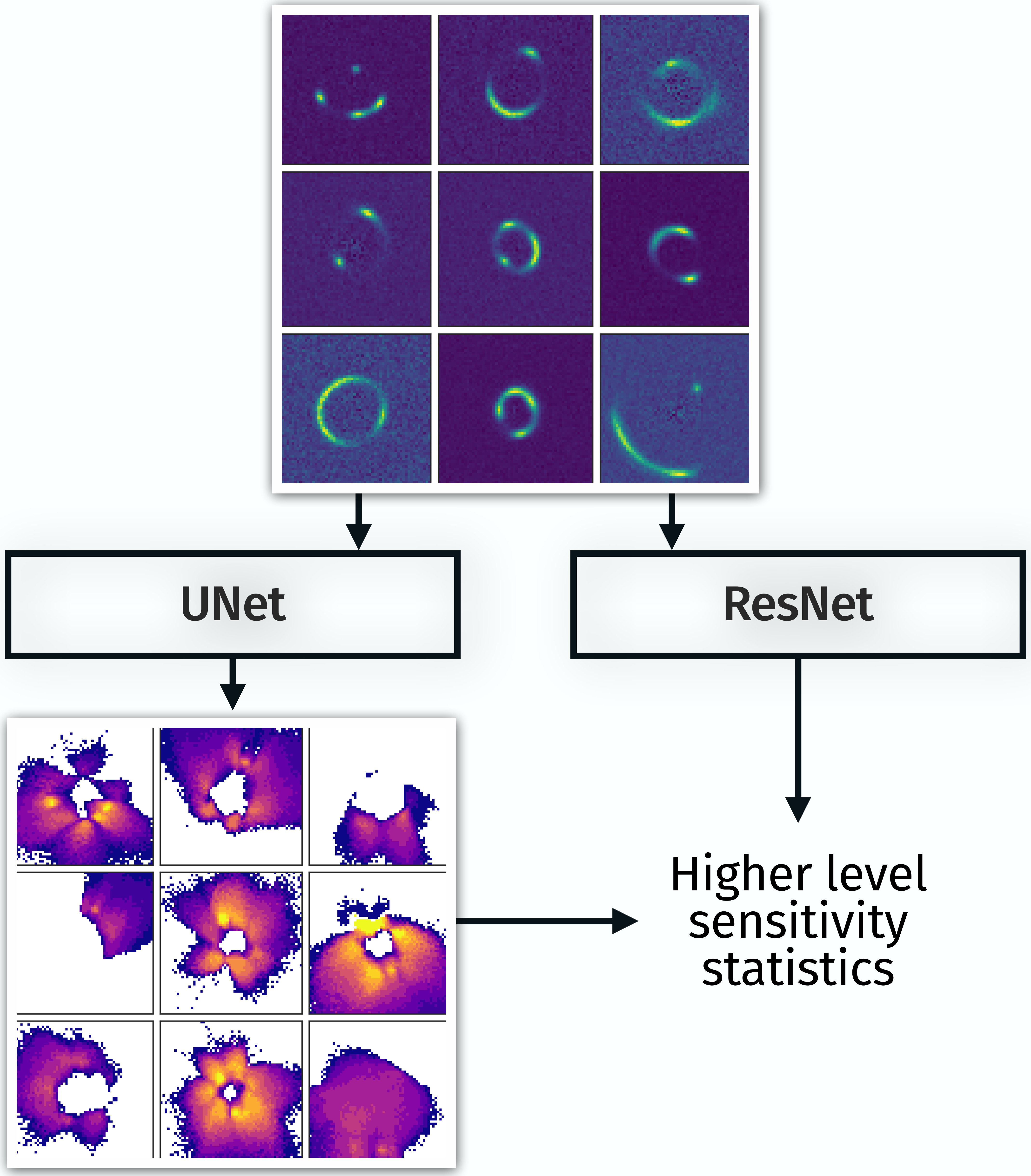
But how do we a priori select for the most sensitive lenses?

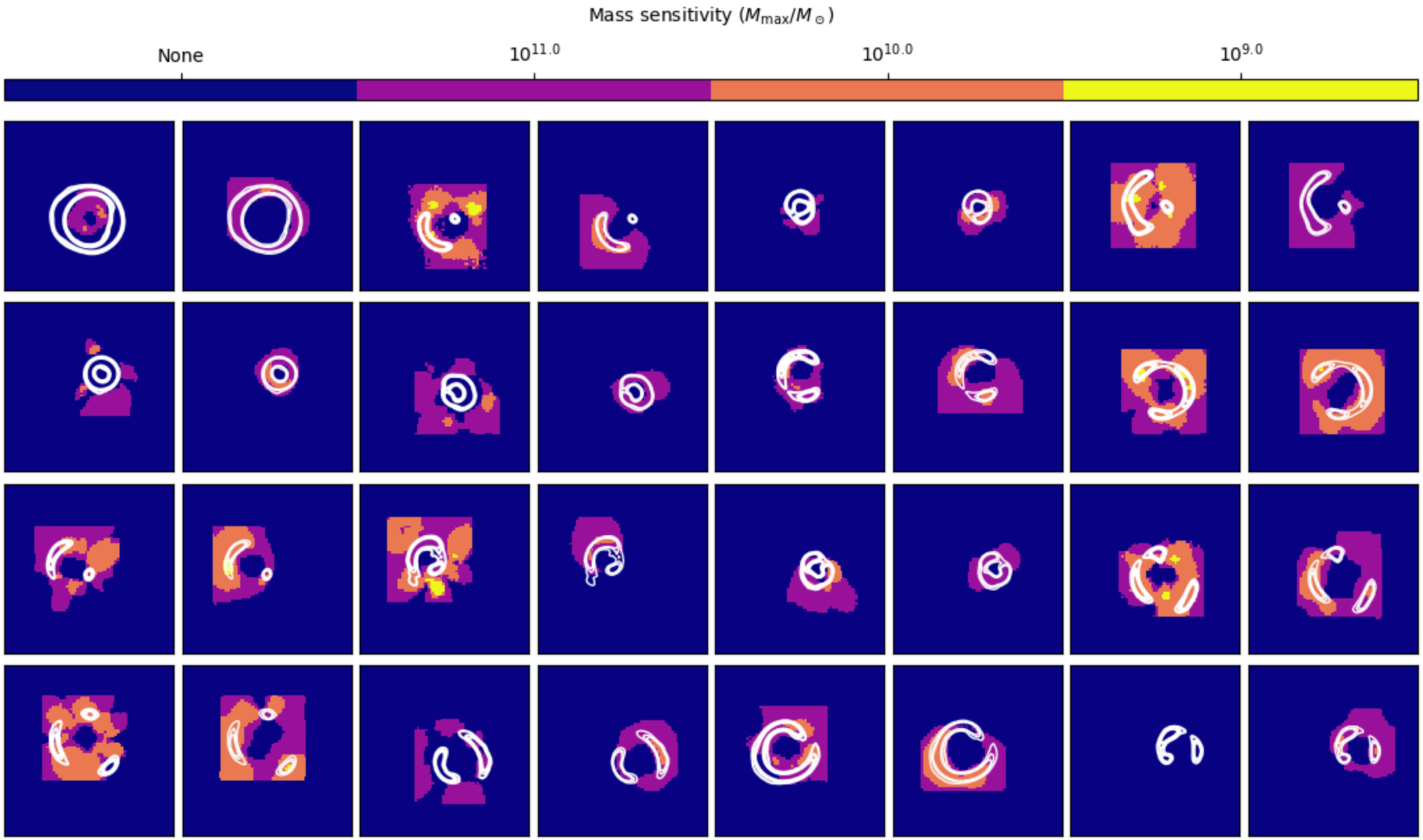
Next steps:

Predicting sensitivity directly

We can train new neural networks using the existing sensitivity maps to predict maps and statistics directly from the images.

A trained network could assess new lenses very cheaply, giving sensitivity predictions for found lens candidates for follow-up.





Next steps:

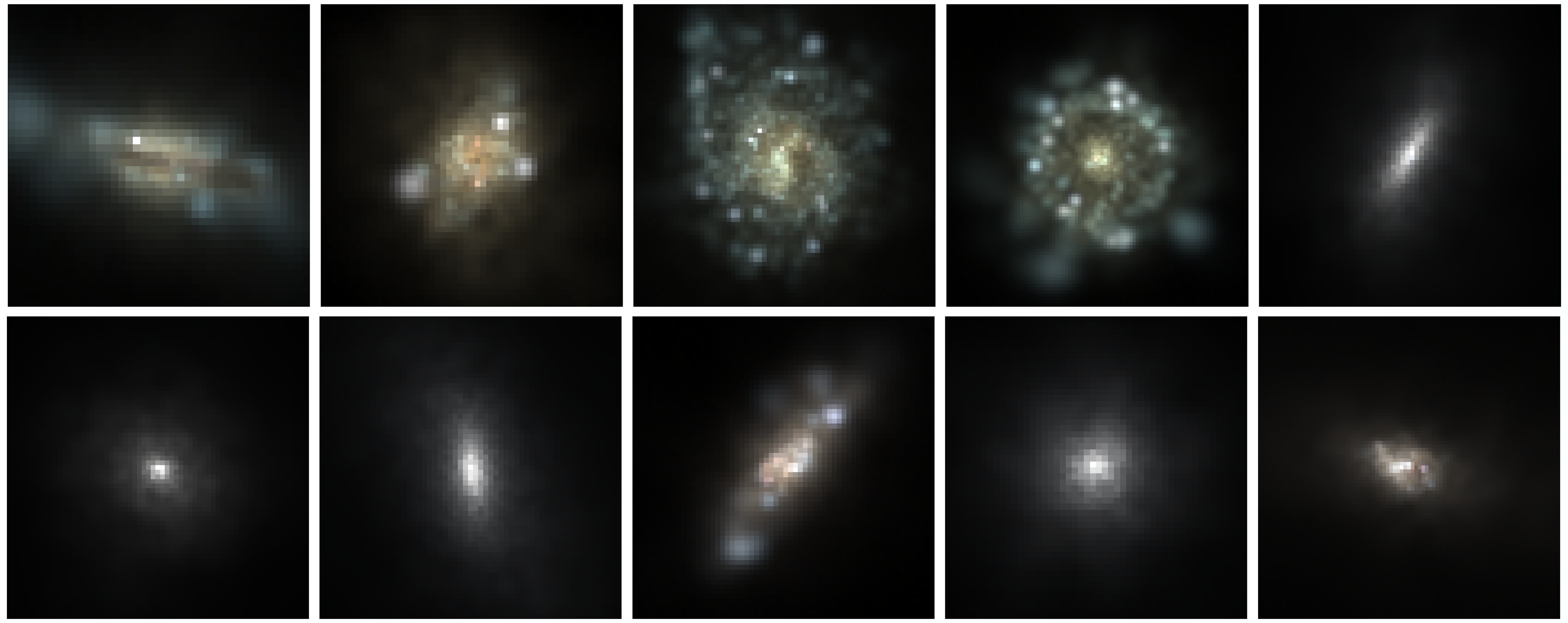
Predicting sensitivity directly

Other work:

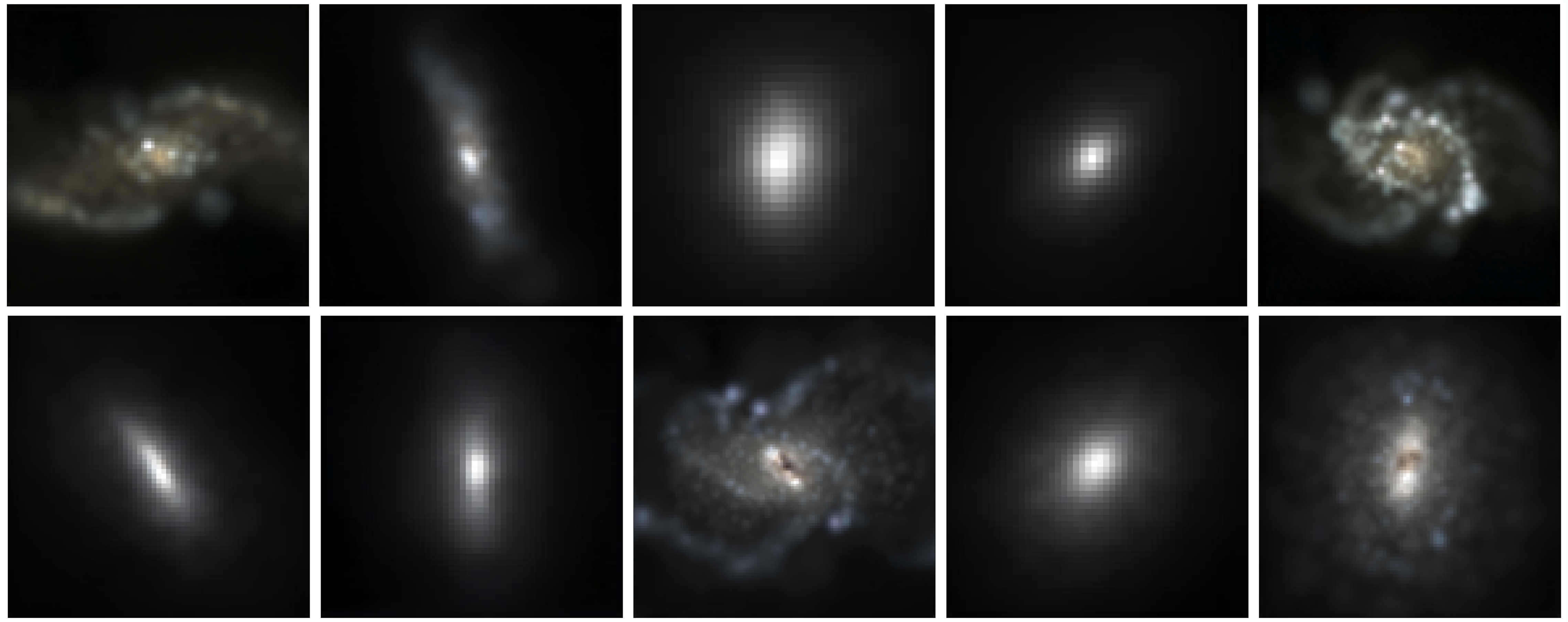
Generating realistic galaxy images

Hozlschuh+ (2022, arXiv:2203.11956)

Training Data



Generated Galaxies



We generated high resolution images of galaxies using a variety of generative models and input data.

Our best model was able to accurately reproduce all the tested physical properties of the input galaxy population

We also found that mixing the generated data with the original data in a separate de-noising problem significantly improved model robustness.

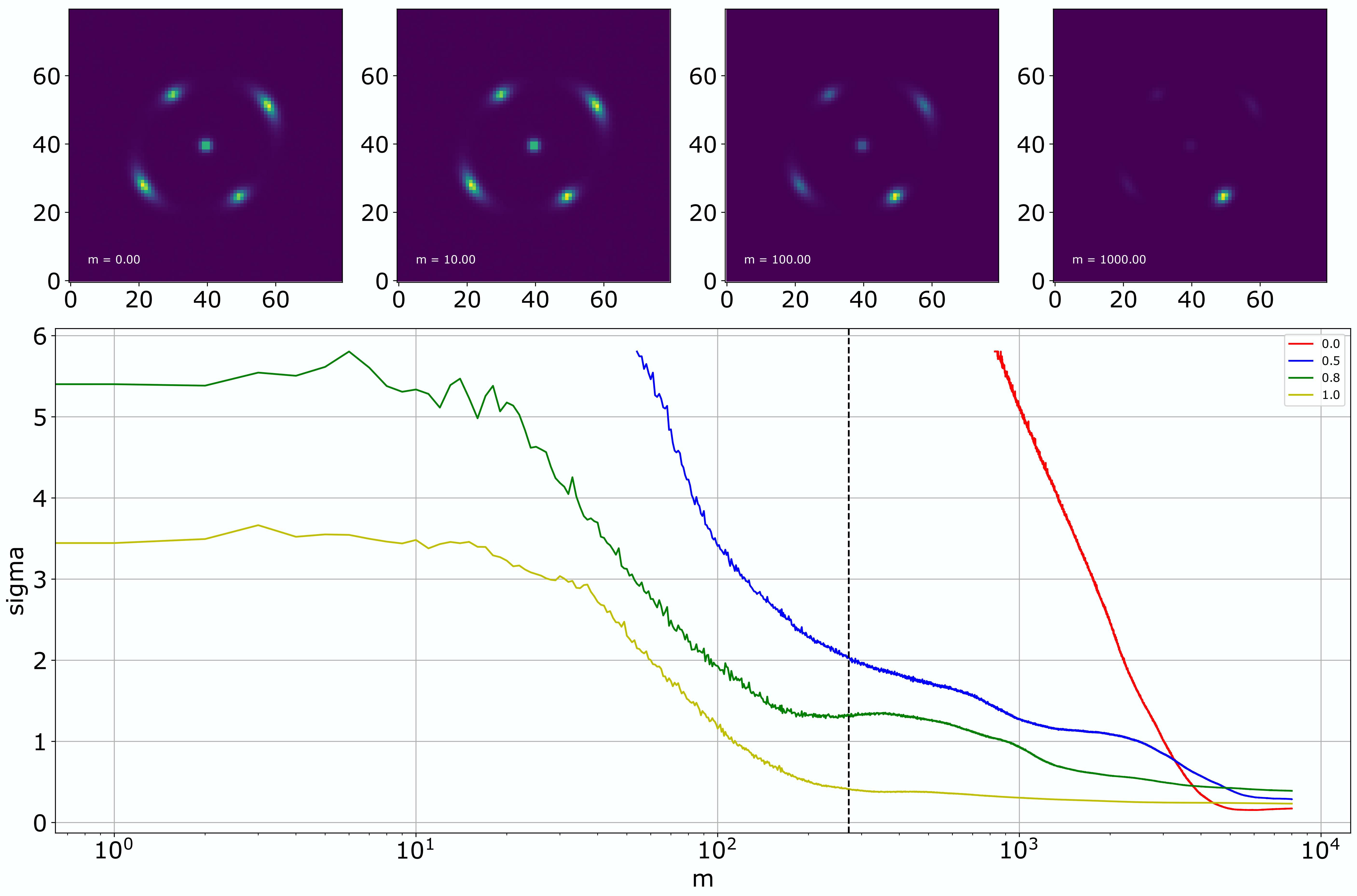
Interpreting CNN lens finders

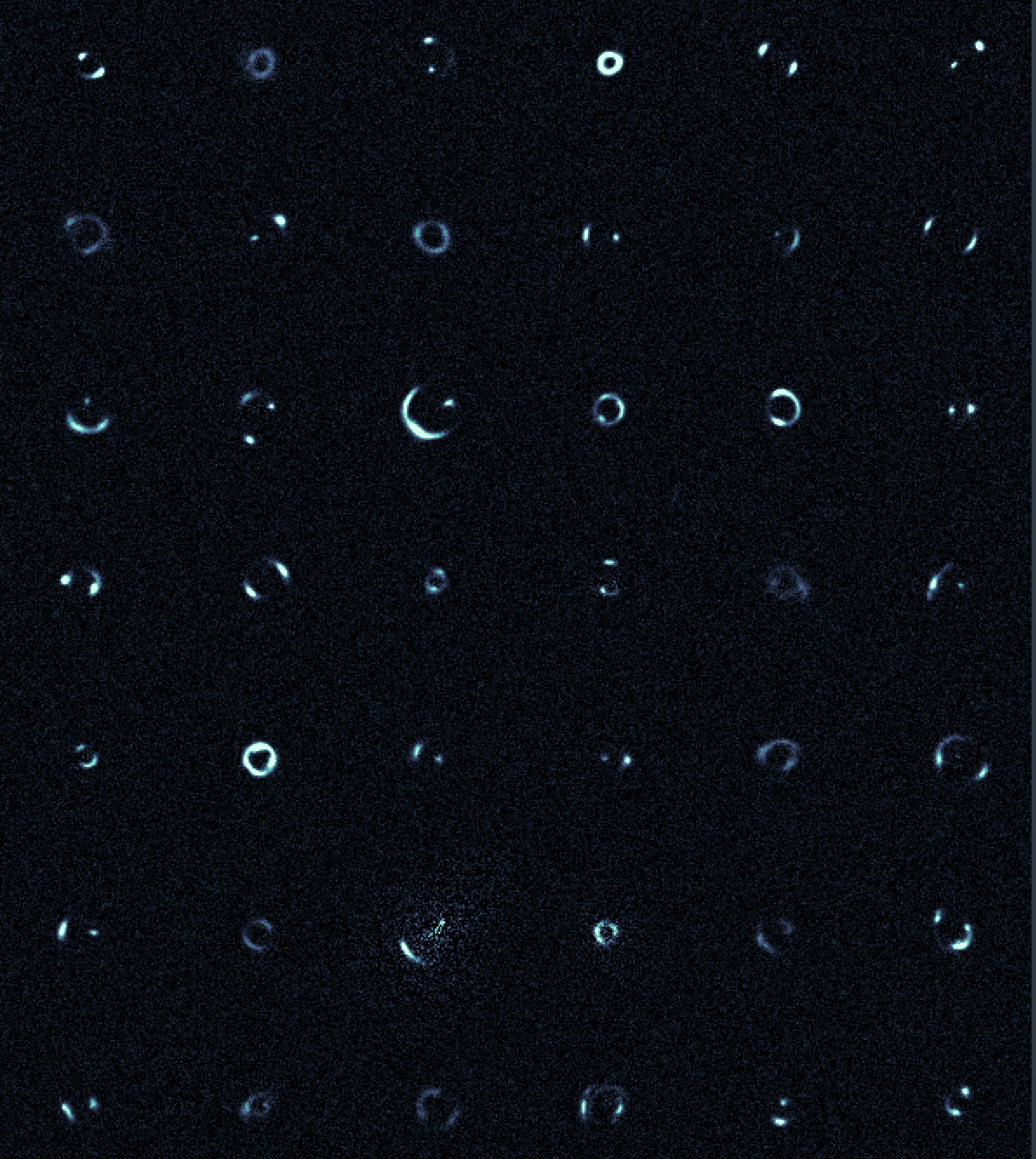
Herle+ (in prep.)

Testing lens finding neural networks to see what (if any) physics they have learned.

When we adjust the brightness of images to create non-physical flux ratios, the network quickly loses confidence that the object is a lens

This is very sensitive to how many "real" galaxies the network has seen in training





Conclusions

- We can estimate subhalo sensitivity orders of magnitude faster than before.
- For Euclid, we expect 1 detectable subhalo per ~10 lenses
- This improves with some pre-selection
- WDM cannot be distinguished from CDM below the point where WDM has already been ruled out.
- A small number of lenses could give strong constraints on $f_{\rm sub}$.
- With enough data from the current method, a new neural network could predict sensitivities directly from images.