# Machine Learning in Euclid Strong Lensing Working Group

Eric Jullo

Based on slides taken from the ECSL-WG

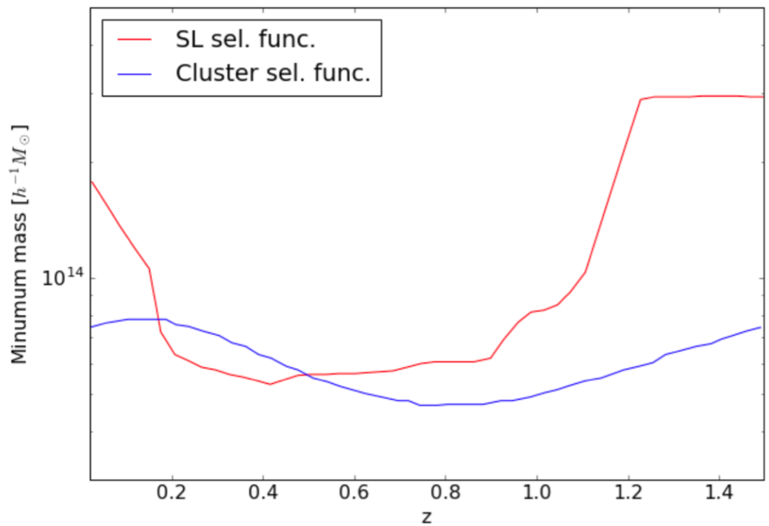
Thanks to the support of ISSI @ Bern

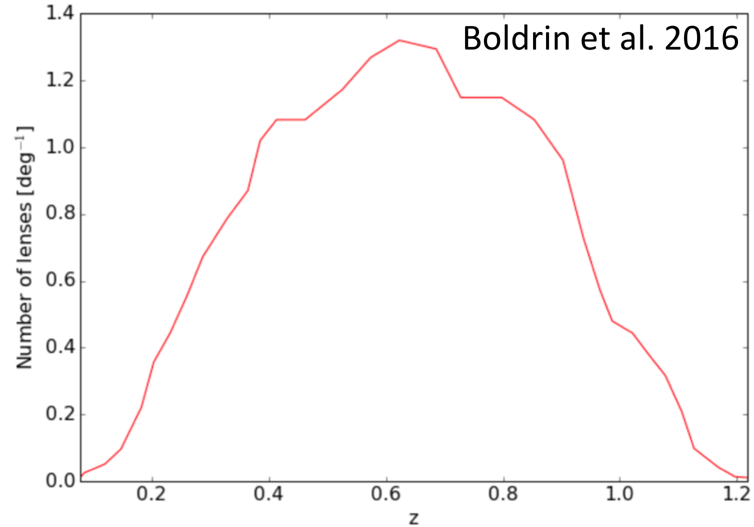
# Context

- Today, the number of known lenses is ~1000.
- Euclid will observe about 100 000 lenses *(LensPop, Collett 2015)*
  - in a single uniform survey
  - with higher resolutions compared to ground based survey (DES, KiDS, LSST…)
- Only SKA2-mid could yield similar number of lenses

- Euclid will observe 15,000 deg$^2$ to depth 24.5 in VIS filter, FWHM=0.1"

# Many Science cases



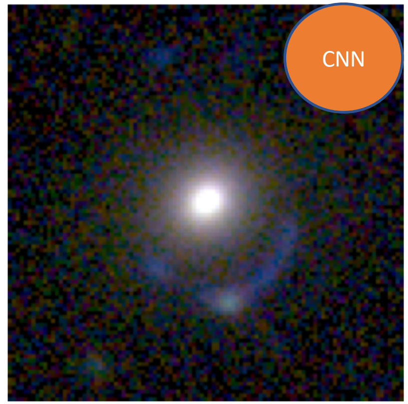

- Study of galaxy clusters, structure and galaxy formation
- Study of high redshift galaxies, reionization
- High resolution study of AGN
- Compound lensing cosmography

# Existing lens samples

BOSS-SLACS


Bolton et al. 2006

KIDS



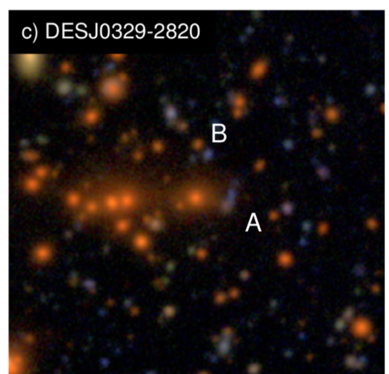

Petrillo et al. 2016

CFHT-SL2S

Cabanac et al. 2007

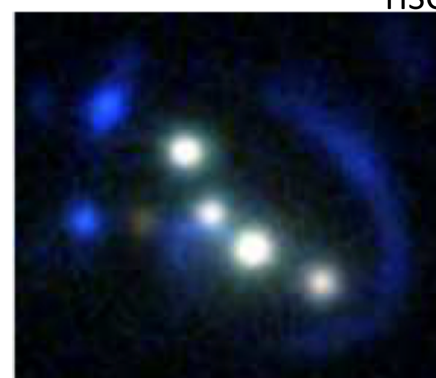
COSMOS

Faure et al. 2008




Nord et al. 2016

HSC



Sonnenfeld et al. 2017

# Existing search methods

- Visual inspection (Jackson et al. 2008)
- Detect arc-like features (Alard et al. 2006, Seidel & Bartelmann 2007, Bom et al. 2007)
- Detect rings (Gavazzi et al. 2014, Joseph et al. 2014)
- Lens model fitting (Sonnenfeld et al. 2017)
- Artificial Neural Networks & Convolutional Neural Networks
- Support Vector Machines

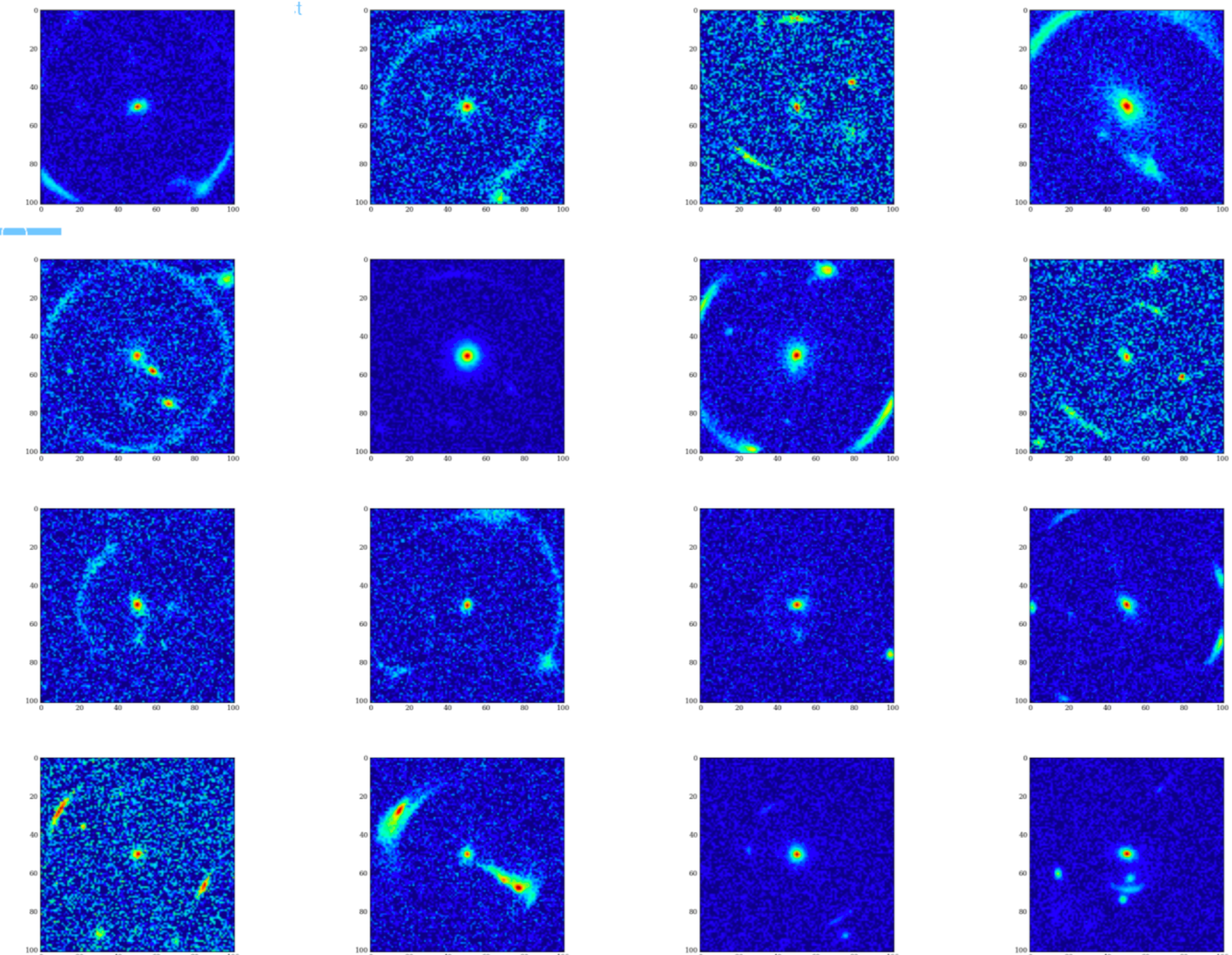
# Strong Lensing Simulation challenge

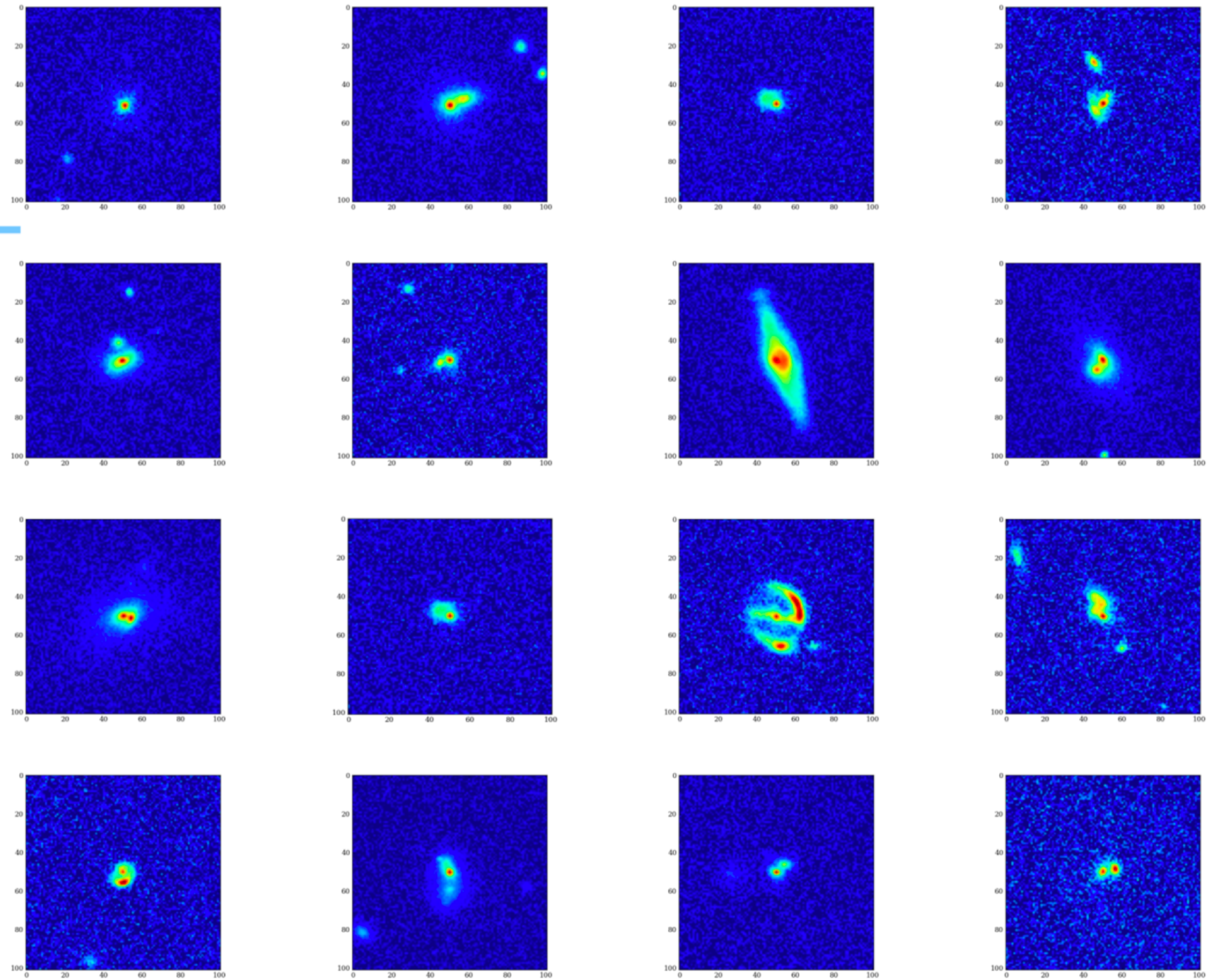
- Mock lenses taken from Millenium Simulations
    - Lightcone 1.6 $deg^2$, up to z=6
    - Projection in lens planes and ray-tracing with GLAMER (Metcalf & Petkova 2014
    - Extrapolation beyond N-body resolution with NFW analytical profiles
- Background sources taken from Hubble UDF
    - Decomposed into shapelets to remove noise (Meneghetti et al. 2008)
- Visible galaxies taken from Millenium semi-analytic model (Guo et al. 2011)
    - Bulge and disk simulated with Sersic profiles
    - Spiral arms model from Metcalf et al. 2018

# Strong Lensing Simulation challenge

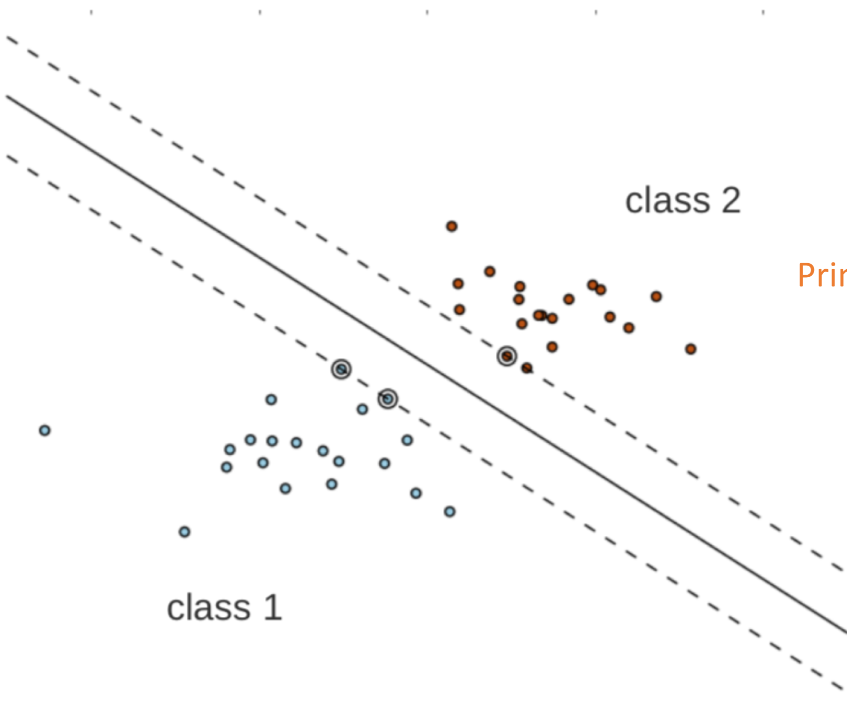
- Production of 20,000 images in *u,g,r,i* KiDS and VIS band for Euclid
  ➔100,000 images of 101x101 pixels for training
- The test sample contains 100,000 images VIS and 400,000 images KiDS
- The participants have 48h to rank each image between 0 (Non-Lens) and 1 (Lens)
- The challenge was open btw Nov 25$^{th}$, 2016 and Feb 5$^{th}$, 2017
- 24 entries in the challenge (people from Euclid, KiDS, HSC, … )

  *Jackson & Togore examined 70,000 and 30,000 images in 48-hours*





# Support Vector Machine – Supervised




labels

$$\{\mathbf{x}_i, y_i\}, \quad i = 1, \cdots, l, \quad y_i \in \{-1, 1\}, \quad \mathbf{x}_i \in \mathbf{R}^d$$

$$\mathbf{x}_i \cdot \mathbf{w} + b \geq +1 \quad \text{for } y_i = +1$$
$$\mathbf{x}_i \cdot \mathbf{w} + b \leq -1 \quad \text{for } y_i = -1$$

Primal likelihood

$$L_P \equiv \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^{l} \alpha_i y_i (\mathbf{x}_i \cdot \mathbf{w} + b) + \sum_{i=1}^{l} \alpha_i$$

$$\mathbf{w} = \sum_i \alpha_i y_i \mathbf{x}_i \qquad \sum_i \alpha_i y_i = 0.$$

Dual likelihood

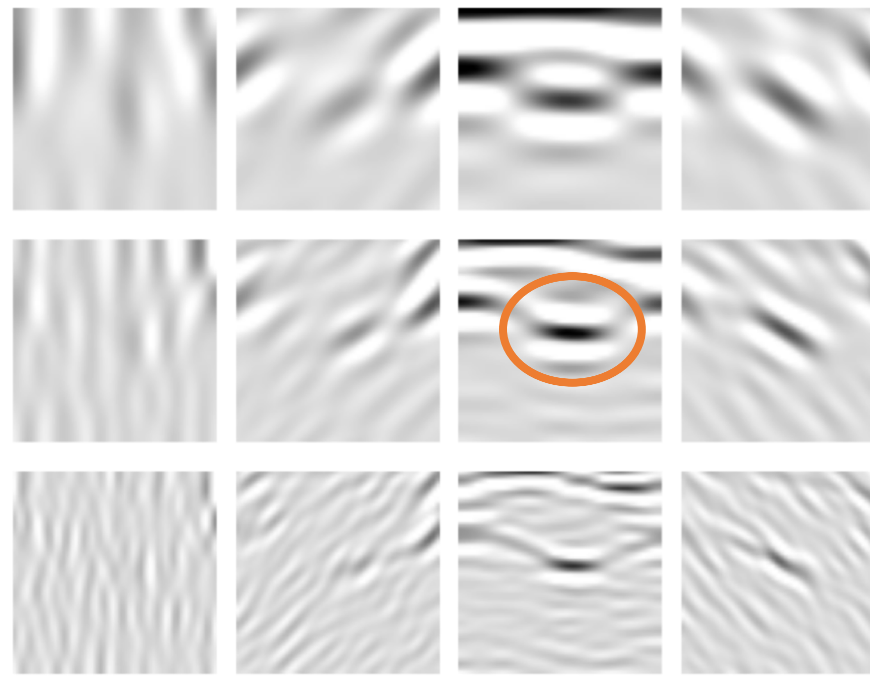
$$L_D = \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j \mathbf{x}_i \cdot \mathbf{x}_j$$

Vapnik et al. 1992

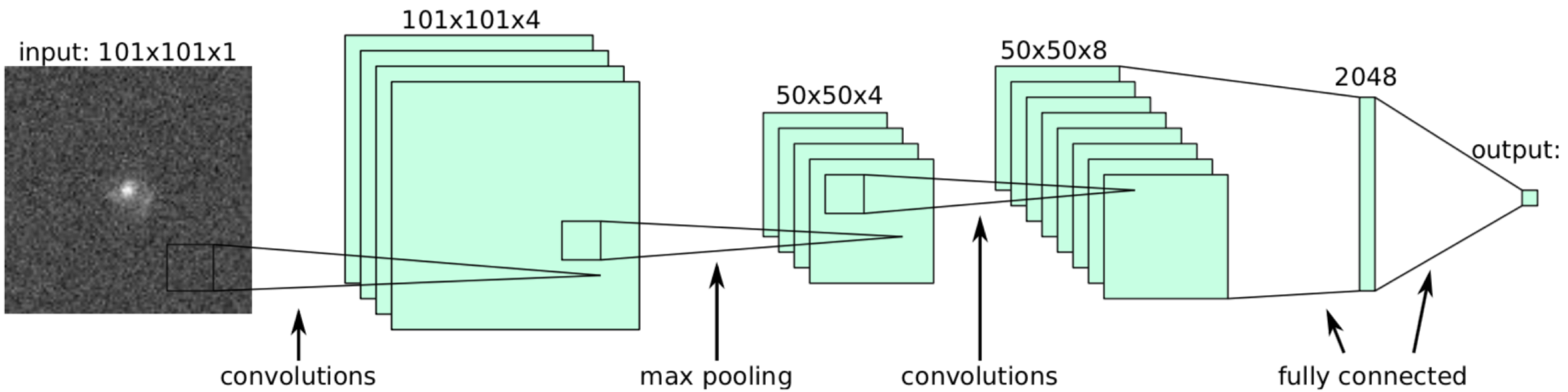
# Gabor-SVM (Hartley et al. 2017)

- Extraction of features from images using
  - Sextractor & GALFIT
  - Gabor filter bank  & their statistical moments
- Use Scikit to run SVM

$$G_c[i, j] = Be^{-\frac{(i^2+j^2)}{2\sigma^2}} \cos\left(\frac{2\pi}{\lambda}(i \cos\theta + j \sin\theta)\right)$$



# Convolutional Neural Networks




Example of CNN (Schaefer et al. 2017)

# Glossary

from Schaefer et al. 2017

- Fully connected layer: the classic ANN neuron layer. Every input is connected to every neuron of the layers. They are used as the final CNN layers to merge the information contained in the feature maps into the desired out- put form.

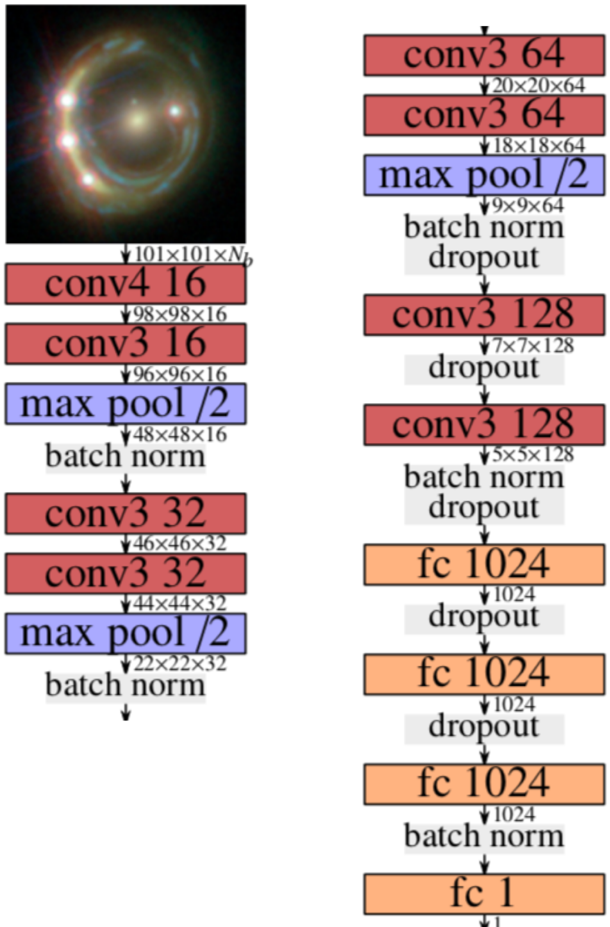
- Dropout layer: only active during training. They randomly sever half the connections between the two layers they separate. Reduce coadaption of the neurons (learning the same features) and reduce overfitting.

- Batch normalization layers : normalize and shift the output along a small input sample $B = \{x1...m\}$ following the equation

$$y_i = \gamma \frac{x_i - \mu_B}{\sigma_B^2} + \beta,$$

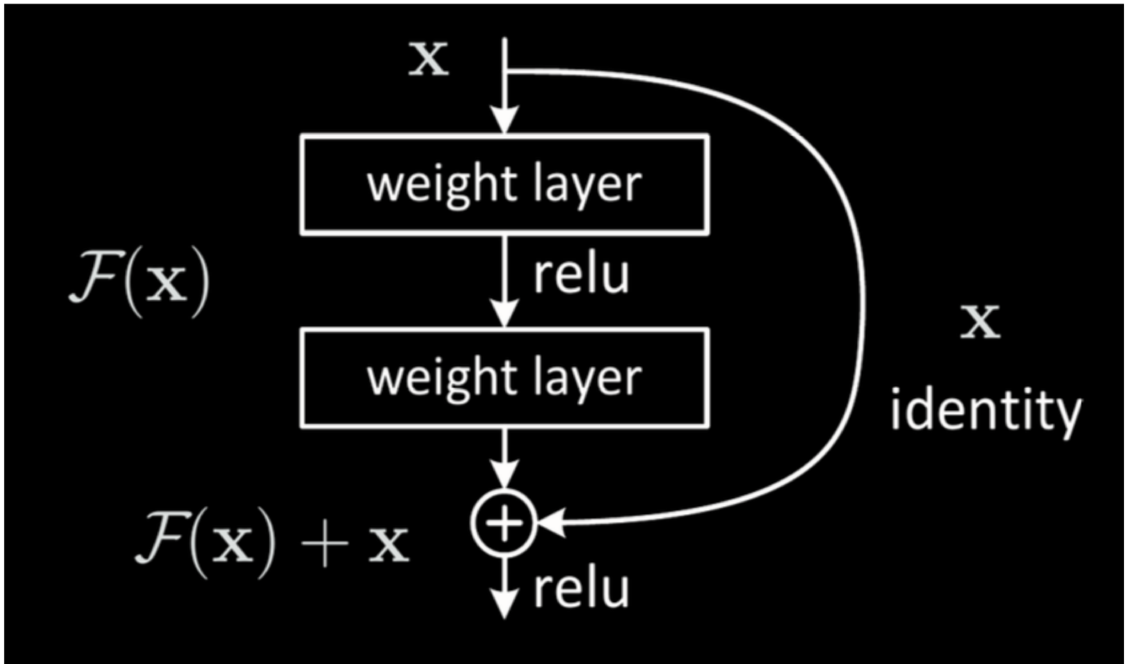
where $\mu_B$ and $\sigma_B$ are the mean and the variance over $B$. $\gamma$ and $\beta$ are two model parameters of the layer. Batch normalization is used to increase the training speed of the CNN

# Schaefer et al. 2017



# Residual blocks

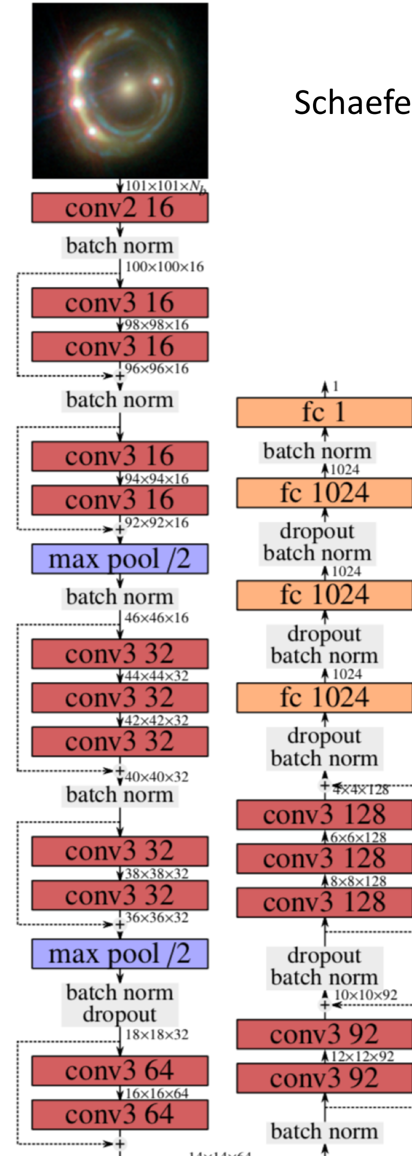
He et al. 2015




ReLU (rectified layer unit, Nair & Hinton 2010) $f(x) = \max(x,0)$

Residual blocks: Learning the difference to the identity. Easier to initialize and to train in deep architectures

Schaefer et al. 2017



# BASELINE NETWORK ARCHITECTURE

Lanusse et al. 2018

Same baseline architecture for space and ground using pre-activated bottleneck ResNet units:

- Conv1: 101x101x32
- ResNet1: (101x101x32) x 3
- ResNet2: (50x50x64) x 3
- ResNet3: (25x25x128) x 3
- ResNet4: (12x12x256) x 3
- ResNet5: (6x6x512) x 3
- AvePool: 512

$\Longrightarrow$ A total of 46 convolution layers

Training: $\sim$ 6 hours, Classification (space): $\leq$ 1 minute

[Nvidia Titan X (Pascal) GPU]

# TRAINING DETAILS

Lanusse et al. 2018

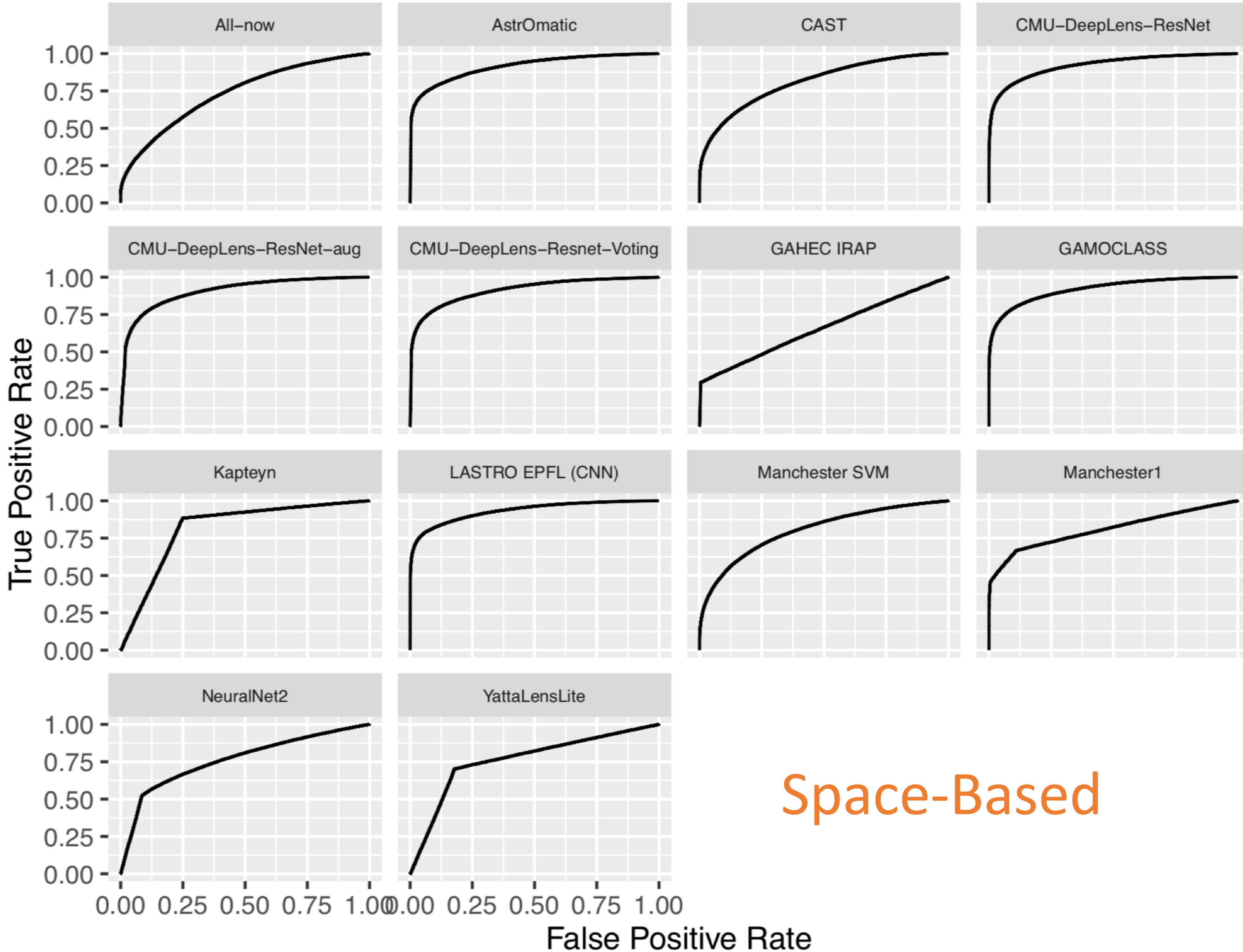
## Preprocessing

- Clipping of extreme values
- Subtraction of mean image
- Standardisation by noise variance
- Masked areas set to 0

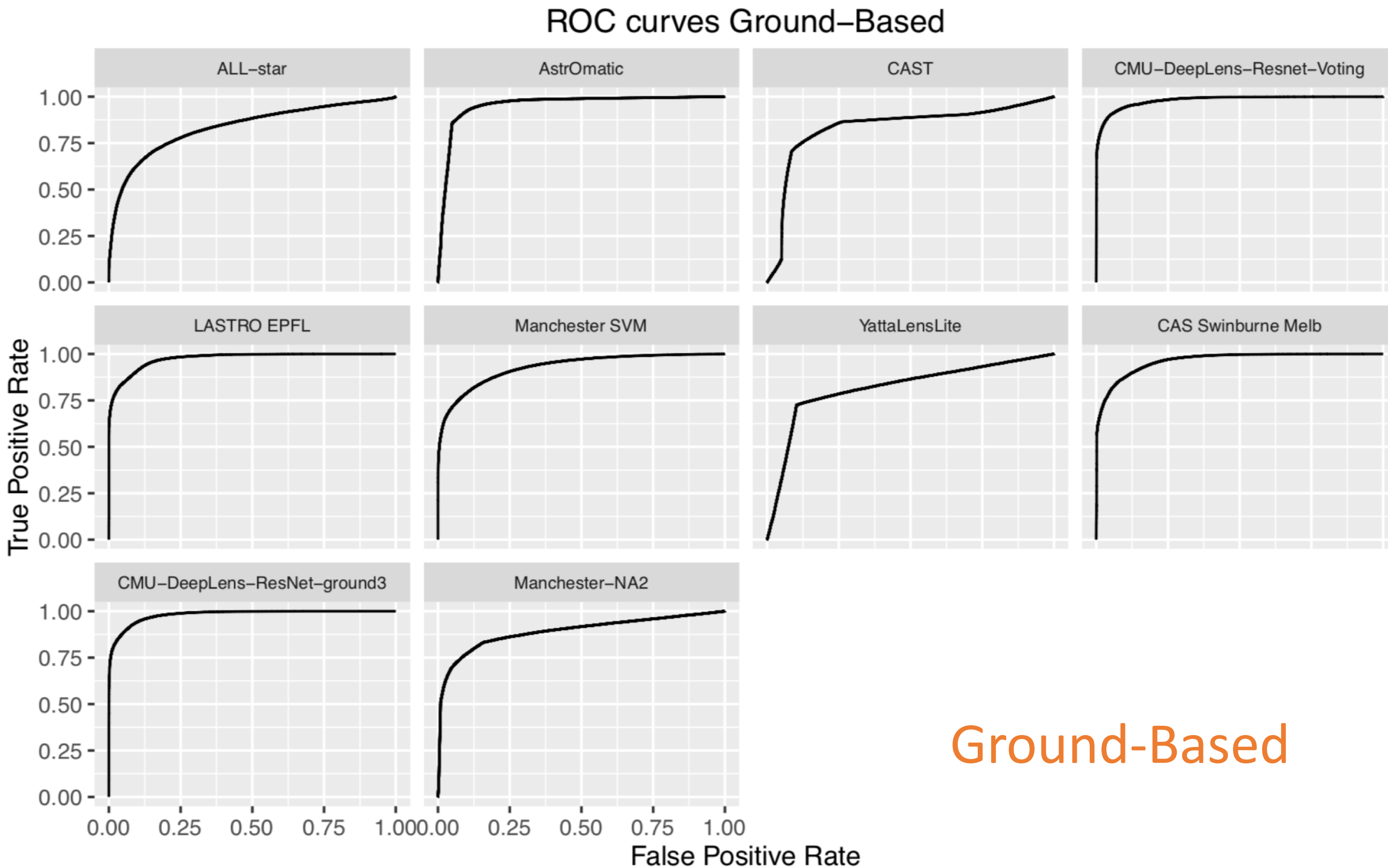
## Data augmentation

- Random rotations
- Random isotropic stretching (zooming)
- Random mirror flipping in x and y
- Additional Gaussian noise

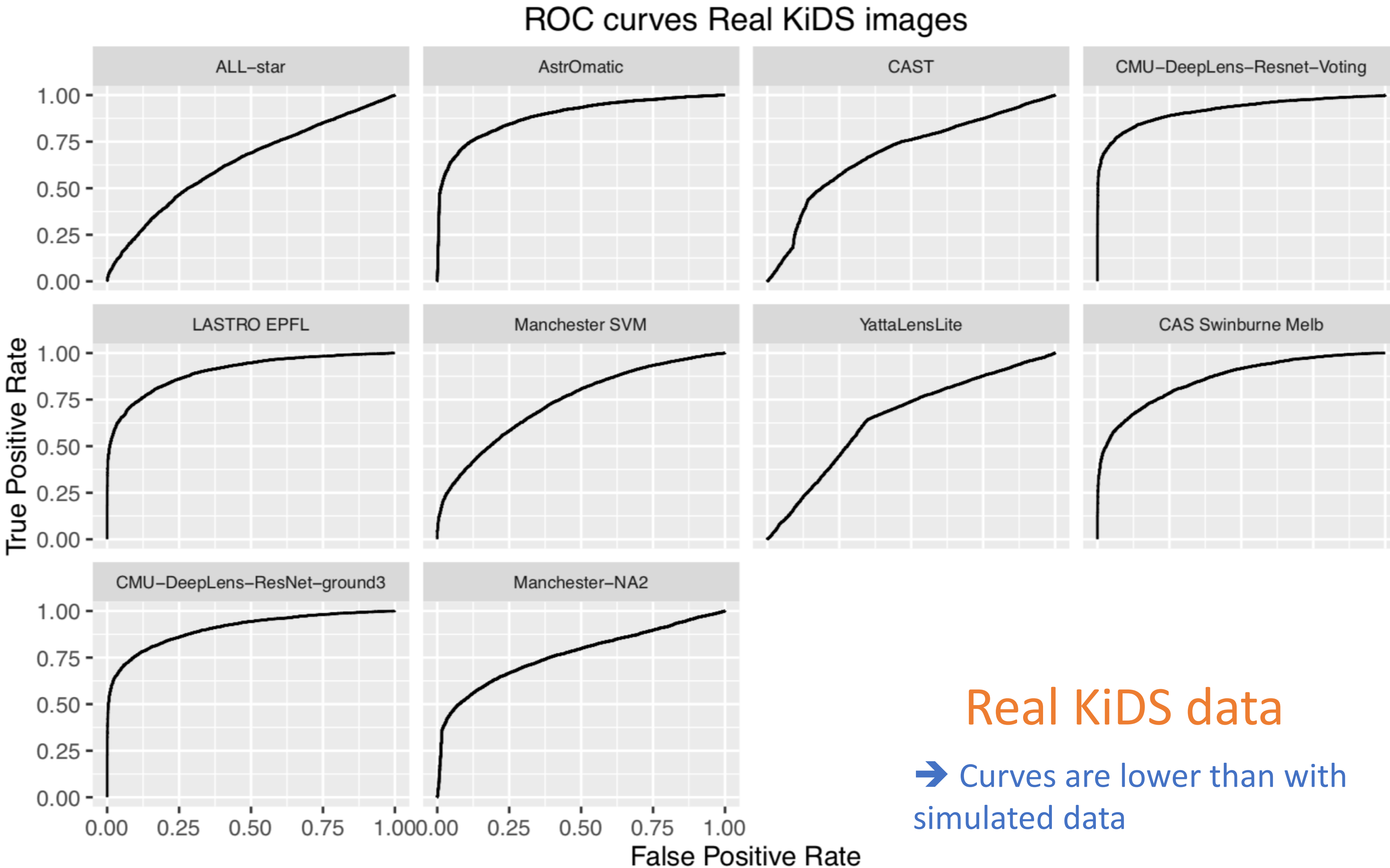
# Results




Space-Based




Ground-Based




Real KiDS data

➔ Curves are lower than with simulated data

# Some false positive cases

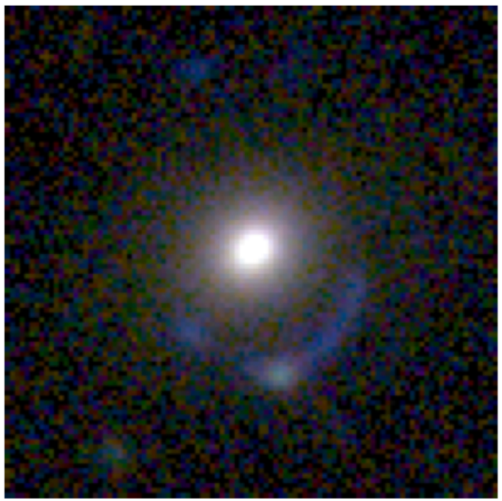
Petrillo et al. 2017





# Difficult true lenses

**✓ True positive**



J085446-012137

**✓ True positive**

J114330-014427

**✗ True negative**

J1403+0006

| Name | type | AUROC | $TPR_0$ | $TPR_{10}$ | short description |
|---|---|---|---|---|---|
| CMU-DeepLens-ResNet-ground3 | Ground-Based | 0.98 | 0.09 | 0.45 | CNN |
| CMU-DeepLens-Resnet-Voting | Ground-Based | 0.98 | 0.02 | 0.10 | CNN |
| LASTRO EPFL | Ground-Based | 0.97 | 0.07 | 0.11 | CNN |
| CAS Swinburne Melb | Ground-Based | 0.96 | 0.02 | 0.08 | CNN |
| AstrOmatic | Ground-Based | 0.96 | 0.00 | 0.01 | CNN |
| Manchester SVM | Ground-Based | 0.93 | 0.22 | 0.35 | SVM / Gabor |
| Manchester-NA2 | Ground-Based | 0.89 | 0.00 | 0.01 | Human Inspection |
| ALL-star | Ground-Based | 0.84 | 0.01 | 0.02 | edges/gradiants and Logistic Reg. |
| CAST | Ground-Based | 0.83 | 0.00 | 0.00 | CNN / SVM |
| YattaLensLite | Ground-Based | 0.82 | 0.00 | 0.00 | SExtractor |
| LASTRO EPFL | Space-Based | 0.93 | 0.00 | 0.08 | CNN |
| CMU-DeepLens-ResNet | Space-Based | 0.92 | 0.22 | 0.29 | CNN |
| GAMOCLASS | Space-Based | 0.92 | 0.07 | 0.36 | CNN |
| CMU-DeepLens-Resnet-Voting | Space-Based | 0.91 | 0.00 | 0.01 | CNN |
| AstrOmatic | Space-Based | 0.91 | 0.00 | 0.01 | CNN |
| CMU-DeepLens-ResNet-aug | Space-Based | 0.91 | 0.00 | 0.00 | CNN |
| Kapteyn Resnet | Space-Based | 0.82 | 0.00 | 0.00 | CNN |
| CAST | Space-Based | 0.81 | 0.07 | 0.12 | CNN |
| Manchester1 | Space-Based | 0.81 | 0.01 | 0.17 | Human Inspection |
| Manchester SVM | Space-Based | 0.81 | 0.03 | 0.08 | SVM / Gabor |
| NeuralNet2 | Space-Based | 0.76 | 0.00 | 0.00 | CNN / wavelets |
| YattaLensLite | Space-Based | 0.76 | 0.00 | 0.00 | Arcs / SExtractor |
| All-now | Space-Based | 0.73 | 0.05 | 0.07 | edges/gradiants and Logistic Reg. |
| GAHEC IRAP | Space-Based | 0.66 | 0.00 | 0.01 | arc finder |

**Table 3.** The AUROC, $TPR_0$ and $TPR_{10}$ for the entries in order of AUROC.

# Next challenge in preparation

- Results on real data are deceiving because simulations to too simple

- Next challenge in preparation for the Euclid Bonn meeting in June
  - More realistic galaxies with better colors
  - More complex noise taken from OU-SIM
    - Simulation of lenses with OU-SIM/OU-VIS/OU-NIR pipelines
    - Bypass simulations with noise properties taken from OU-SIM/VIS/NIR pipelines

# Conclusion

- Forthcoming imaging surveys cannot be processed by eye anymore

- Machine learning techniques appear as good solutions
  - Drawback: They require large simulated and realistic samples
  - They might learn from real data as more lenses are discovered?