Galaxy cluster detection on LSST simulated images using convolutional neural networks

Kirill Grishin (APC)
Simona Mei (APC)

Galaxy cluster detection on composite color images: Motivation

- We do not need to make source catalogs – we eliminate one (middle) step
- ML algorithms can be much faster than usual methods
- For the use of CNN we do not need to internally scale data – images are already scaled to some reasonable range (0…255) when preparing composite color images.
- We can store the information about galaxy colors in three bands in one images.

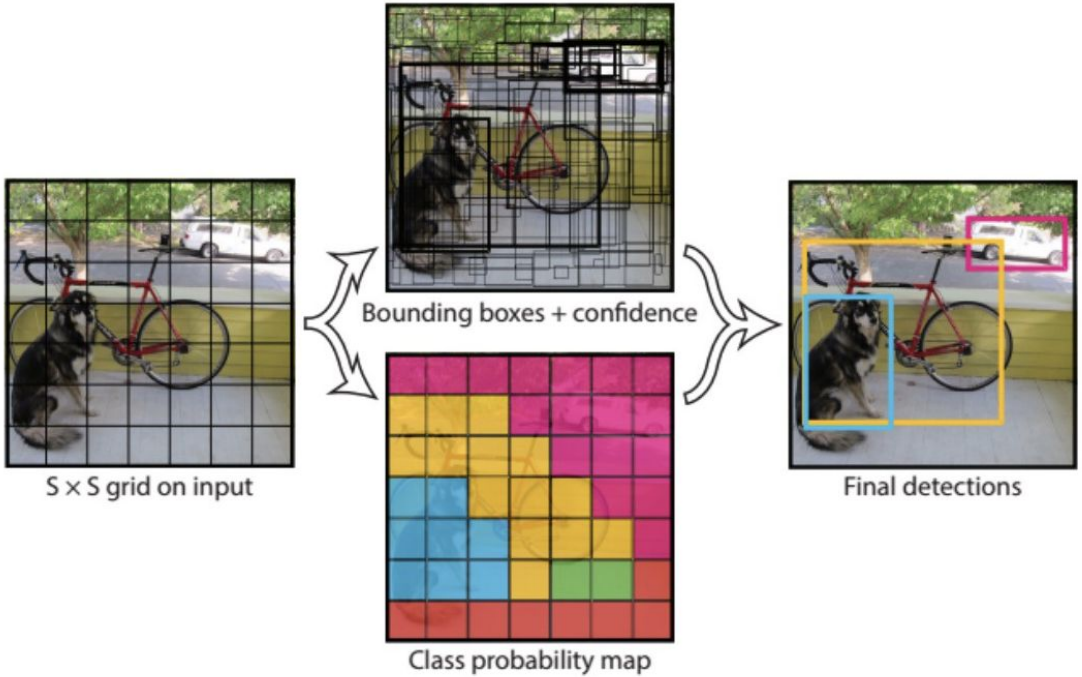
DESC project : Project: [226] Cluster Detection with Deep Learning Networks

- We use Deep Learning Networks to detect optical galaxy clusters and explore different network architectures to derive cluster position and parameters in DC2 and publicly available DES Y1 data. In particular, we will start with the Yolo software, which has been previously applied to SDSS data, then extend to Bayesian Neural Networks. We will compare results with basic CNN and conventional cluster finding methods using the DESC tool, CLEVAR, and will characterize the selection function of our cluster finding algorithms.
- Jim Annis, Camille Avestruz, Dominique Boutigny, Mariano Dominguez, Kirill Grishin, Simona Mei, John Stott + Michel Aguena, Céline Combet, Marie Paturel, Florian Ruppin, ++15 people on Slack channel

  #desc-cl-clfinder-yolo

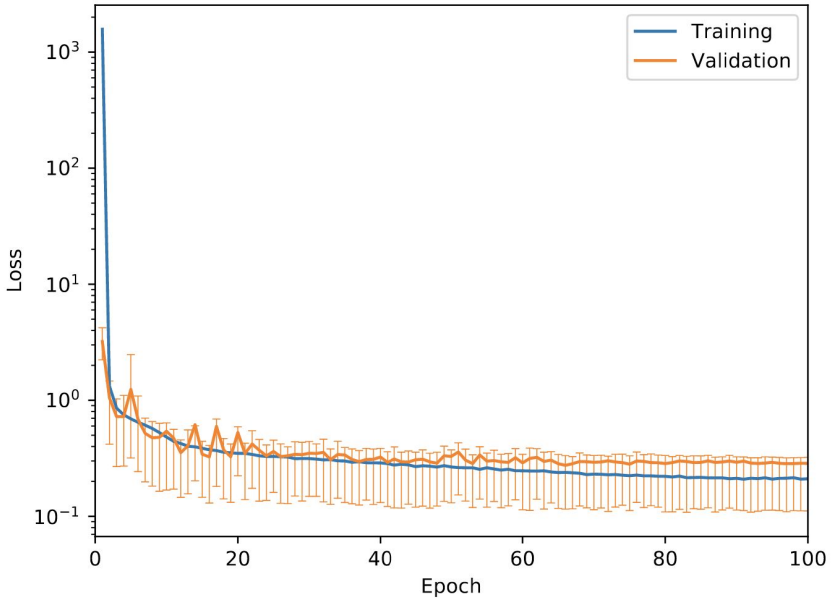
YOLO (Redmon et al. 2016, 2018)

- The network is based on a Deep Convolutional network (Darknet53) and it is used for fast object detection in images (e.g., animals, cars, people, etc.)



Yolo v3 training on SDSS images for redMaPPer clusters

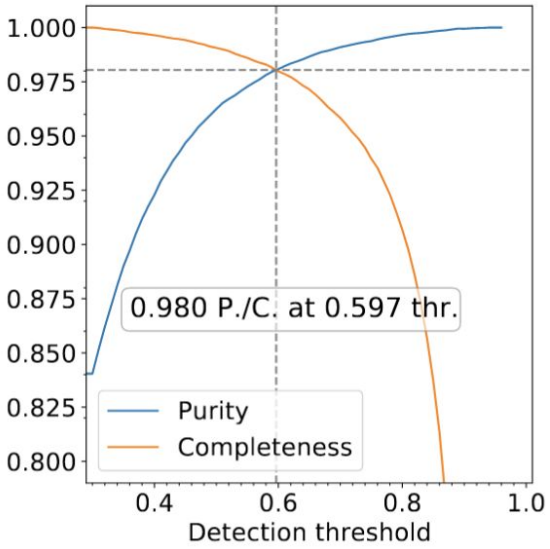
- Our Yolo network was trained on SDSS images for redMaPPer clusters
- The training dataset contained ~12000 galaxy clusters + 12000 random field images
- Validation dataset contained ~12000 galaxy cluster + 12000 random field images
- We run training for 512x512 and 1024x1024 images

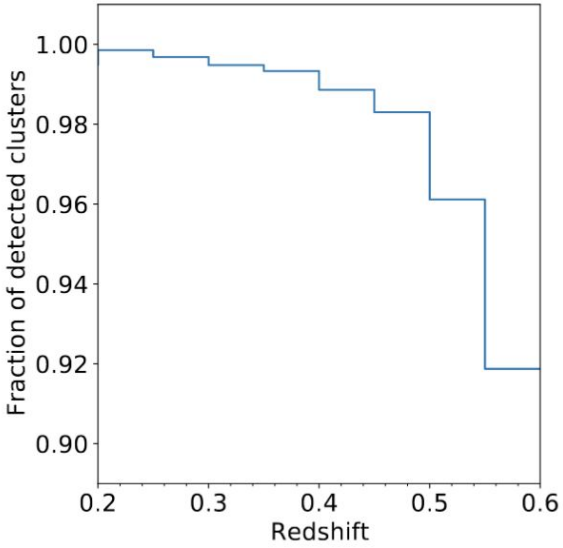


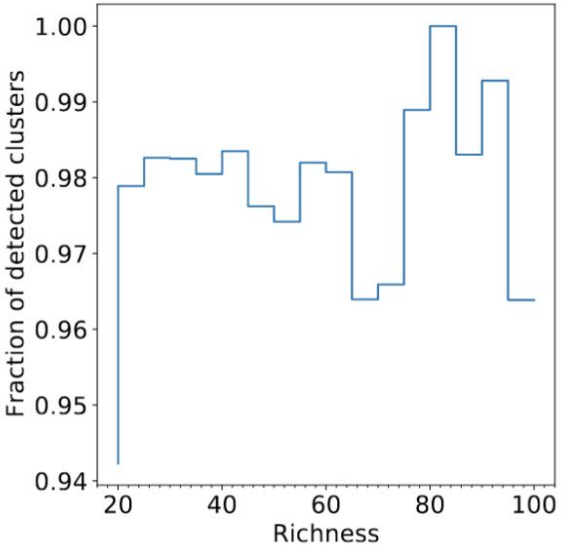
Grishin, Mei, Ilic, in preparation

Yolo v3 performance on redMaPPer clusters

We were able to reach 98% of purity and completeness at a probability threshold of 0.6.







Grishin, Mei, Ilic, in preparation

Application of Yolo to simulated LSST images

We applied Yolo convolutional neural network trained on SDSS images to the simulated LSST DC2 images.

These images were generated for halos with masses $M_{Halo} > 10^{14} M_{\odot}$ from the skysim v 1.1.1 catalog

In total we have produced around 5000 color composite images for clusters and random fields.

Grishin, Mei, et al DESC paper, in preparation

Composite color images for DC2 clusters

We have created composite color images for ~4500 clusters from the “full” version skysim 1.1.2 and for ~5000 images for random fields.

DC2 full resolution images were rescaled to SDSS imaging scale (0.39”/pix). Composite color images were generated using the same filter set as in SDSS: g, r and i

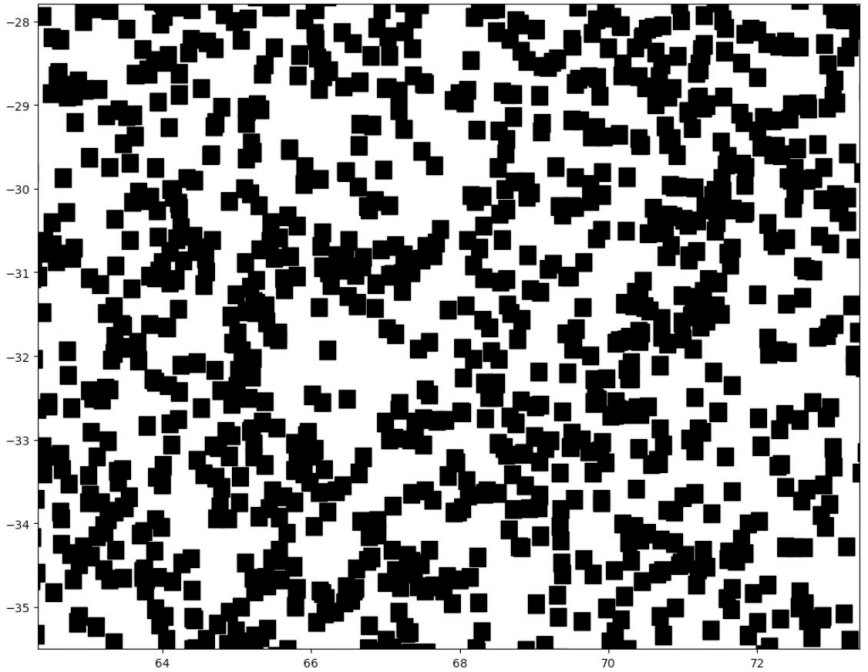
We used the same flux scaling as in SDSS. We used Lupton algorithm to prepare color images.

DC2: random field selection

Given the depth of the DC2 simulation (and LSST), the chance of one or even two cluster for a randomly selected area in a sky is much higher than for SDSS and all randomly selected areas should be checked against the presence of galaxy clusters.

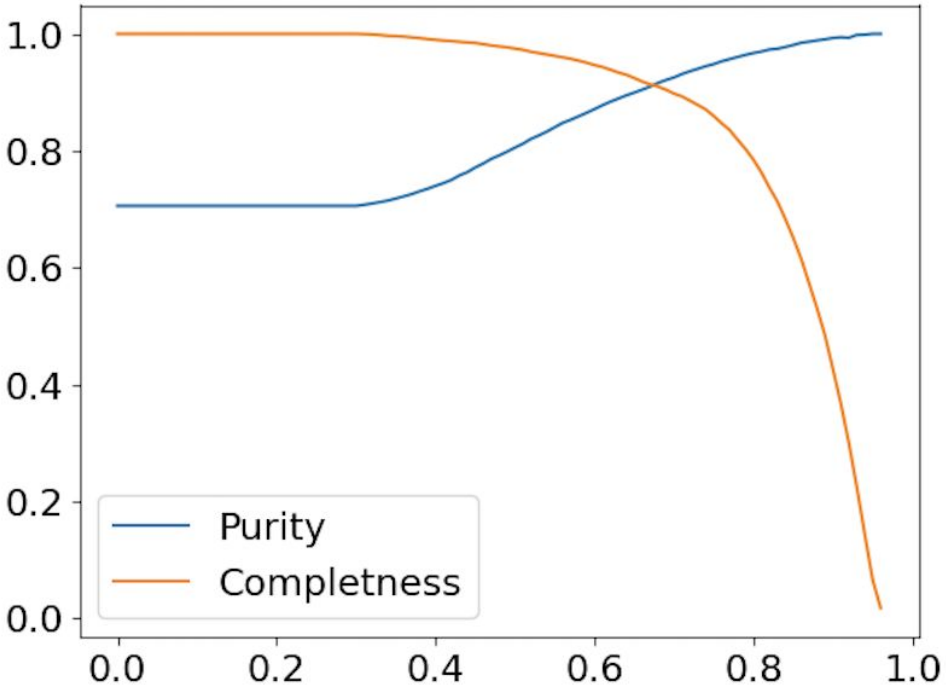
Around ~20% of sky is filled by 13.5x13.5 arcmin cutouts for galaxy clusters.

We have developed an iterative procedure to select random fields far away from clusters



Galaxy cluster detection on DC2 images using Yolo v3

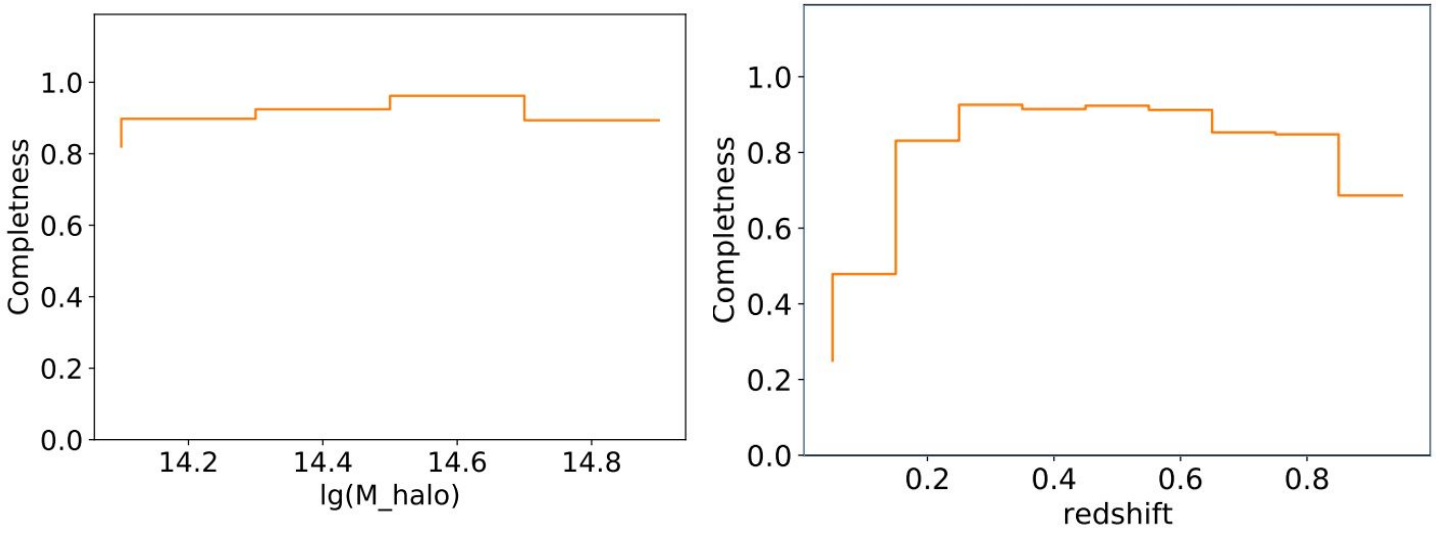
By applying Yolo v3 to the LSST simulated images we can reach ~90% completeness and purity.



Galaxy cluster detection on DC2 images using Yolo v3

The completeness does not depend on mass.

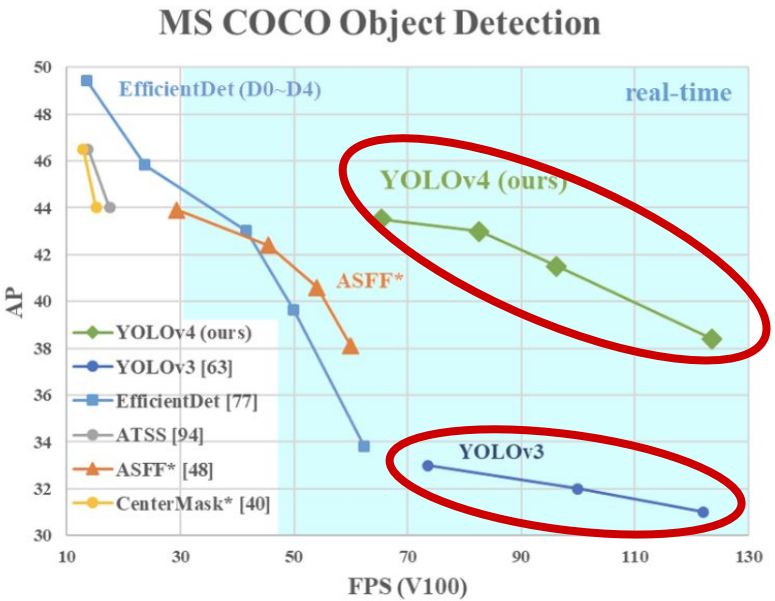
For low-redshift clusters (z< 0.2) the completeness is much lower than at higher redshift. In the redmapper redshift range (z~0.2-0.3), we obtain the same completeness as for redmapper.



Galaxy cluster detection on DC2 images using Yolo v5

We used Yolo v5 to compare its performance with Yolo v3. It has been reported to be faster at training.

Use of Yolo v5 allows to train network on 2048x2048 images in a reasonable timescales: one training epoch for 2048x2048 using Yolo v3 took ~6 hours, using Yolo v5 ~30 min.



Larger images and more recent Yolo versions

- Work from Dominique Boutigny and Marie Paturel: Yolo v4; DC2 full resolution g, r and i.
- 6000 images ready
- Allocation of 50k hours on the Jean Zay supercomputer at IDRIS, latest A100 GPU generation (8x80 GB GPU/server).
- Extend to 6 bands in future

Conclusions

- Weights of Yolo CNN are well transferable: you can train it on the imaging dataset different from the dataset you are applying it to afterwords.
- Color schemes of both datasets should be the same
- Cutout sizes may affect the goodness of cluster detection: on smaller cutouts cluster can be splitted
- We plan to study the purity of the clusters selected by Yolo