Universidad de La Laguna

MINISTERIO DE CIENCIA, INNOVACIÓN Y UNIVERSIDADES

Gobierno de Canarias
Consejería de Economía, Conocimiento y Empleo

Unión Europea
Fondo Europeo de desarrollo Regional
"Una manera de hacer Europa"

Pushing the limits of source detection tools towards LSB light

Nushkia Chamba[1], Caroline Haigh[2], Michael Wilkinson[2], Aku Venhola[3] & Reynier Peletier[2]

LSST PCW, 10-14 August 2020

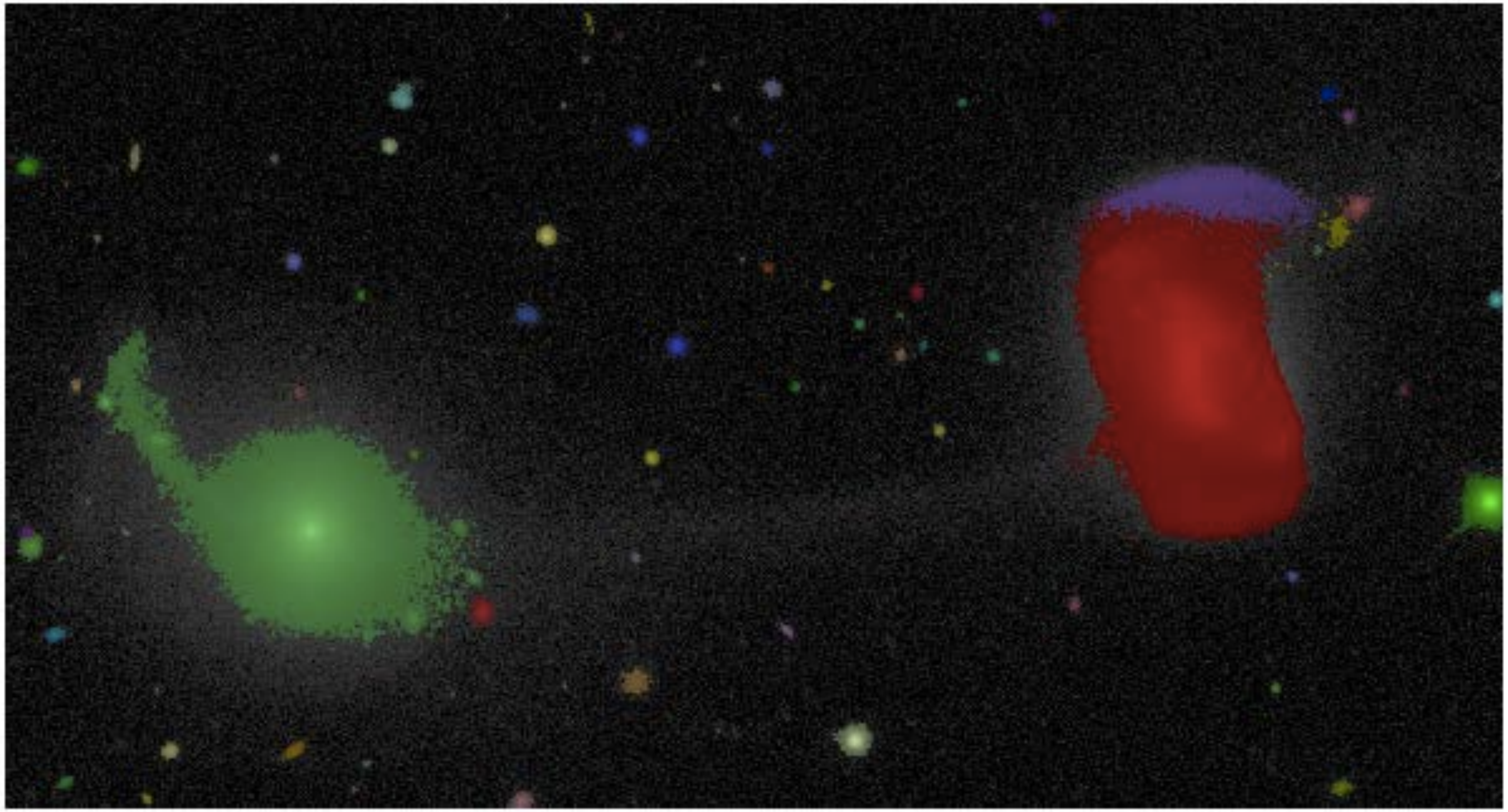
[1]Instituto de Astrofisica de Canarias, Tenerife, Spain
[2]University of Groningen, Groningen, The Netherlands
[3]University of Oulu, Oulu, Finland

SUNDIAL
H2020 Innovative Training Network

This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie grant agreement No 721463

Everybody knows Source Extractor (SE)

Bertin & Arnouts (1996), but it has its limits



SDSS DR7 image,
SE default settings

Teeninga, Moschini, Trager & Wilkinson (2016)

Many other tools exist

Can they be automatically optimised to detect LSB light?

Source Extractor (SE)

- Bertin & Arnouts (1996)
- General purpose

NoiseChisel (NC)

- Akhlaghi & Ishikawa (2015)
- Faint object specialised

Profound (PF)

- Robotham et al. (2018)
- General purpose

Max-Tree Objects (MT)

- Teeninga et al. (2016)
- Faint object specialised

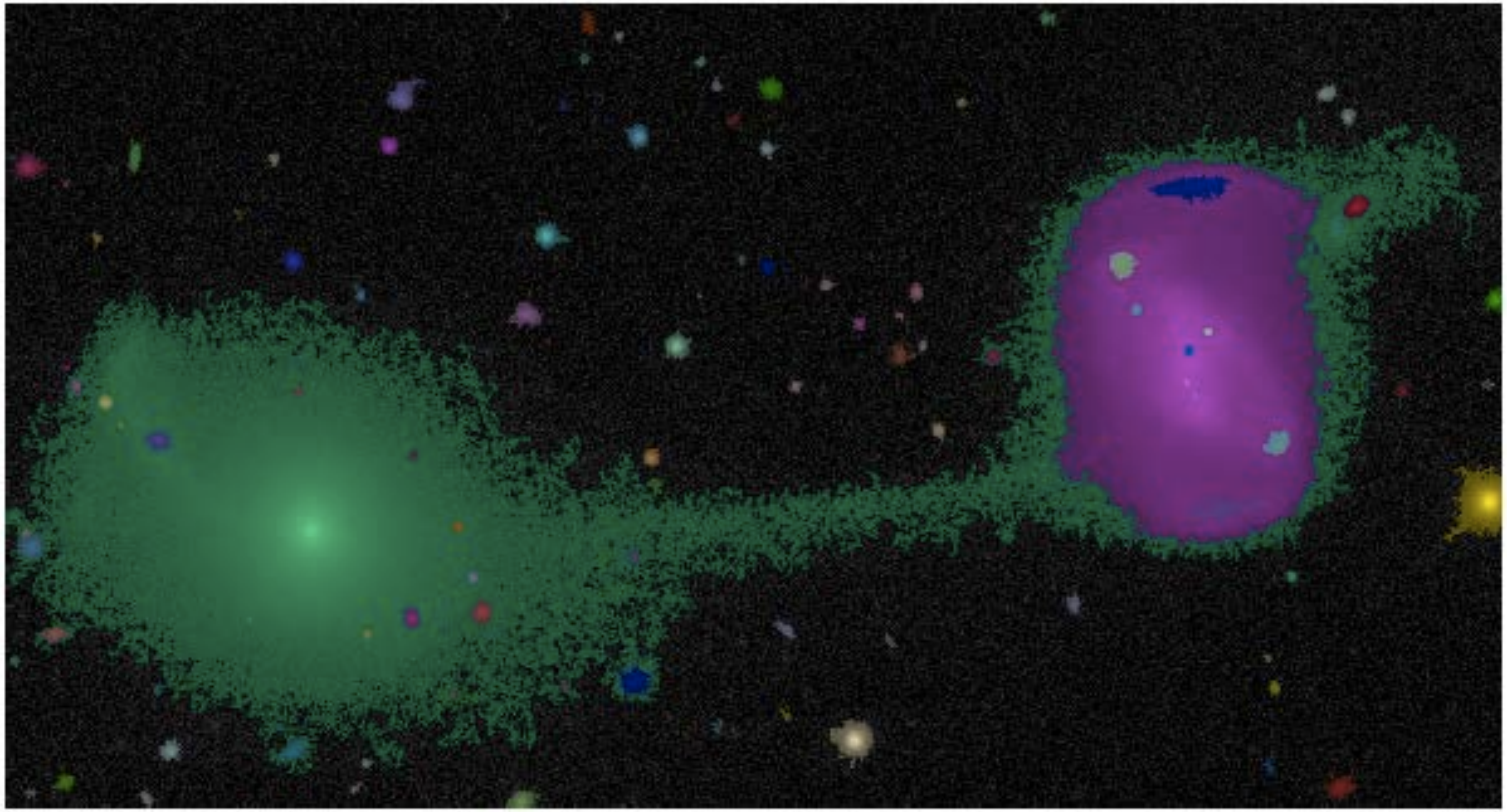
Basic characteristics

- Measure background
- Threshold image w.r.t background
- Locate sources
- Catalogue and measure properties

	SE	PF	NC	MT
core method	nested thresholds	watershed	watershed	max-tree
initial threshold	$\lambda\sigma$	$\lambda\sigma$	percentile	0
nested objects	-	-	-	+
# thresholds	discrete	NA	NA	∞
detection by statistical test	-	-	-	+
parallel	+	?	+	-
# parameters	12	8	25	2

Source finding using Trees

Max-Tree Objects (MT)



SDSS DR7 image, MT, 2 relevant parameters

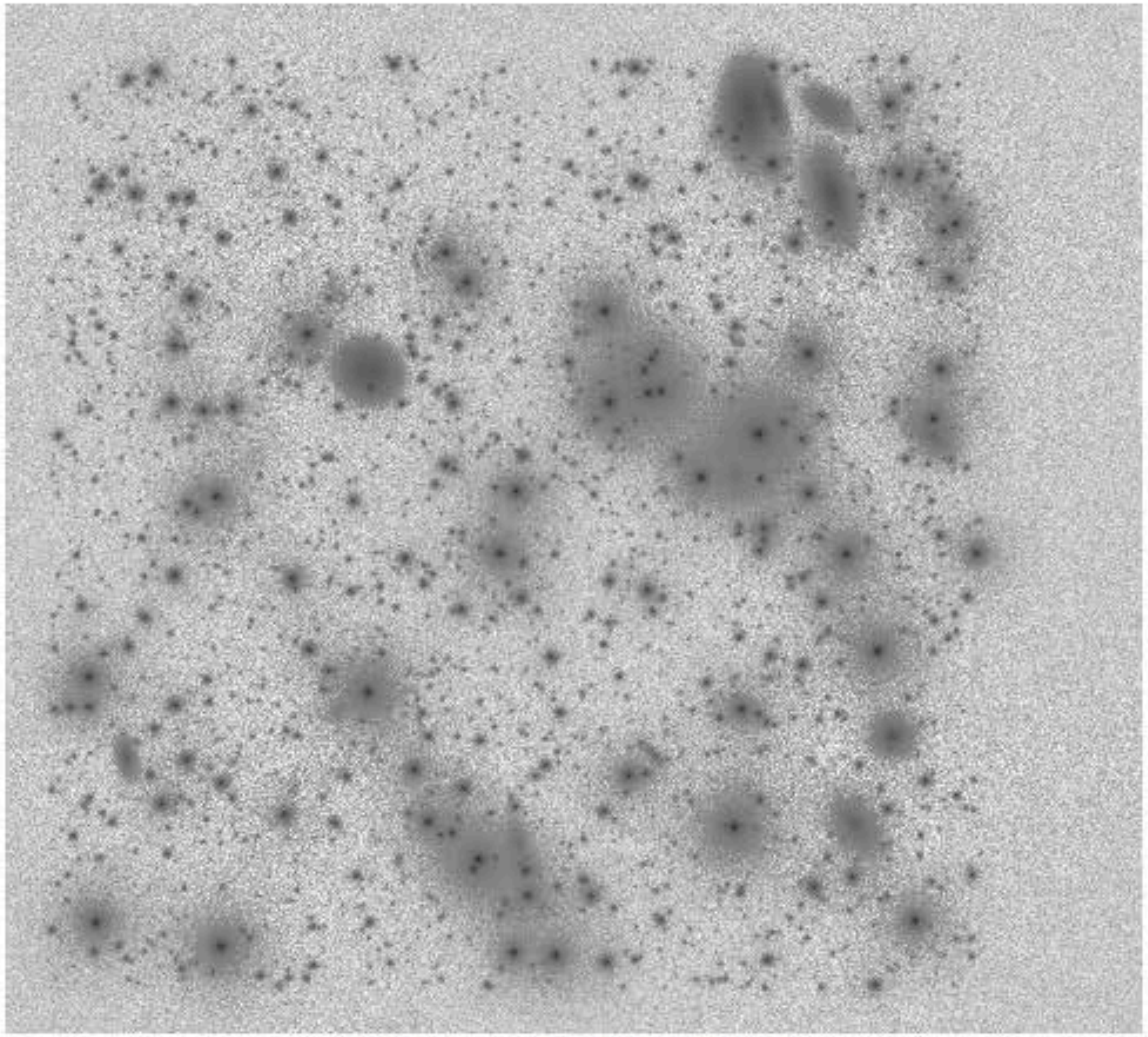
Teeninga, Moschini, Trager & Wilkinson (2016)

A comparison of detection tools

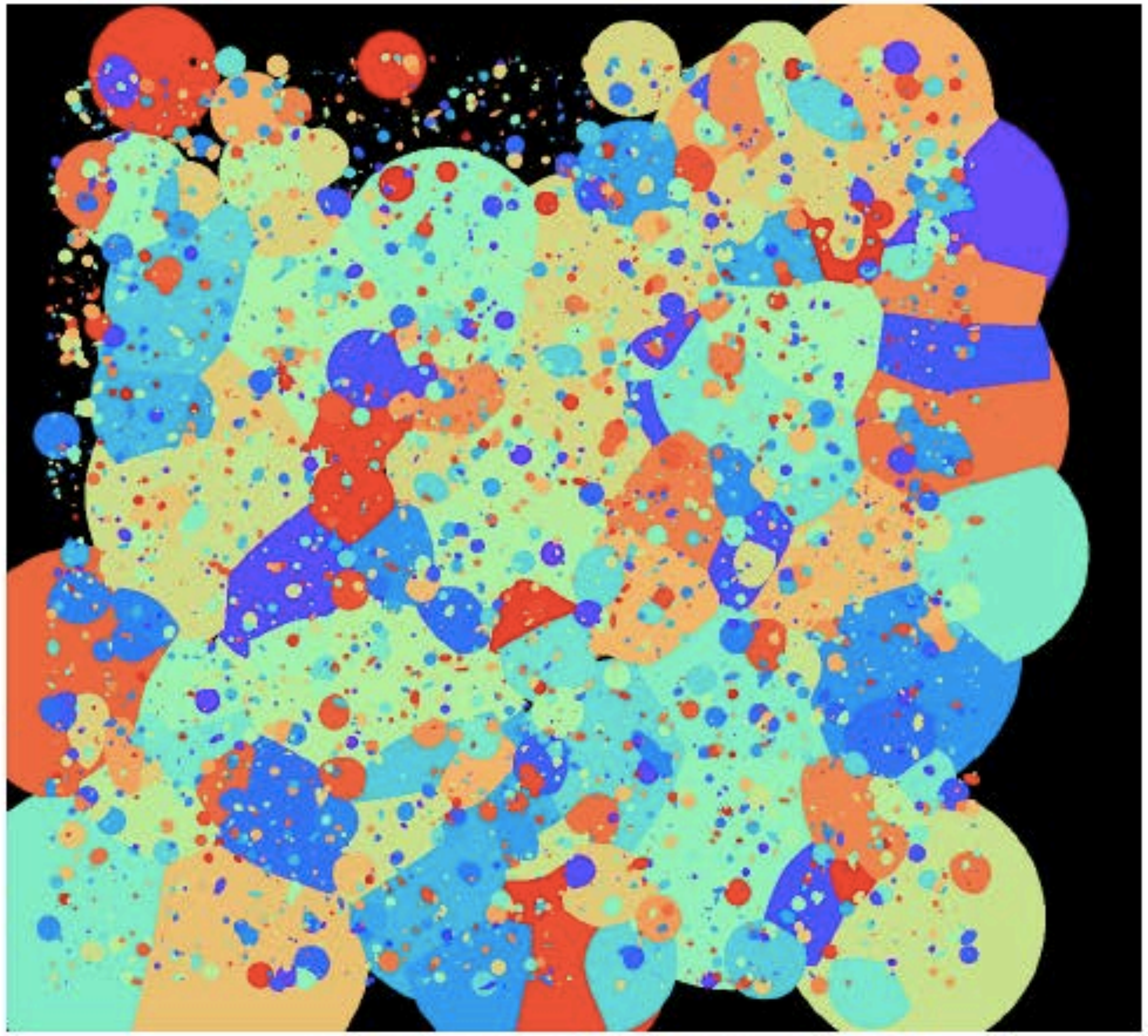
Overview

- In this work: SE, NC, PF and MTO
- Simulated deep data [Fornax Deep Survey, μ_{lim} ~ 30 mag/arcsec^2 (3σ; 100 arcsec^2)]
- Automatic parameter optimisation
- Four different quality measures
- Tests on real images (FDS, IAC Stripe 82, Hubble Ultra Deep Field)

Ground truth for faint light



Simulated FDS image



Ground truth at 0.1σ

Evaluation

Quality criteria

F_1 score: Combines precision (purity) and recall (completeness) in pure detection task

$$F_1 = \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}$$

Area score: Optimizes segmentation quality, combining under-merging error (UM) and over-merging error (OM)

$$\text{Area-score} = 1 - \sqrt{OM^2 + UM^2}$$

Combined score A: $\sqrt{\text{Area-score}^2 + F_1^2}$

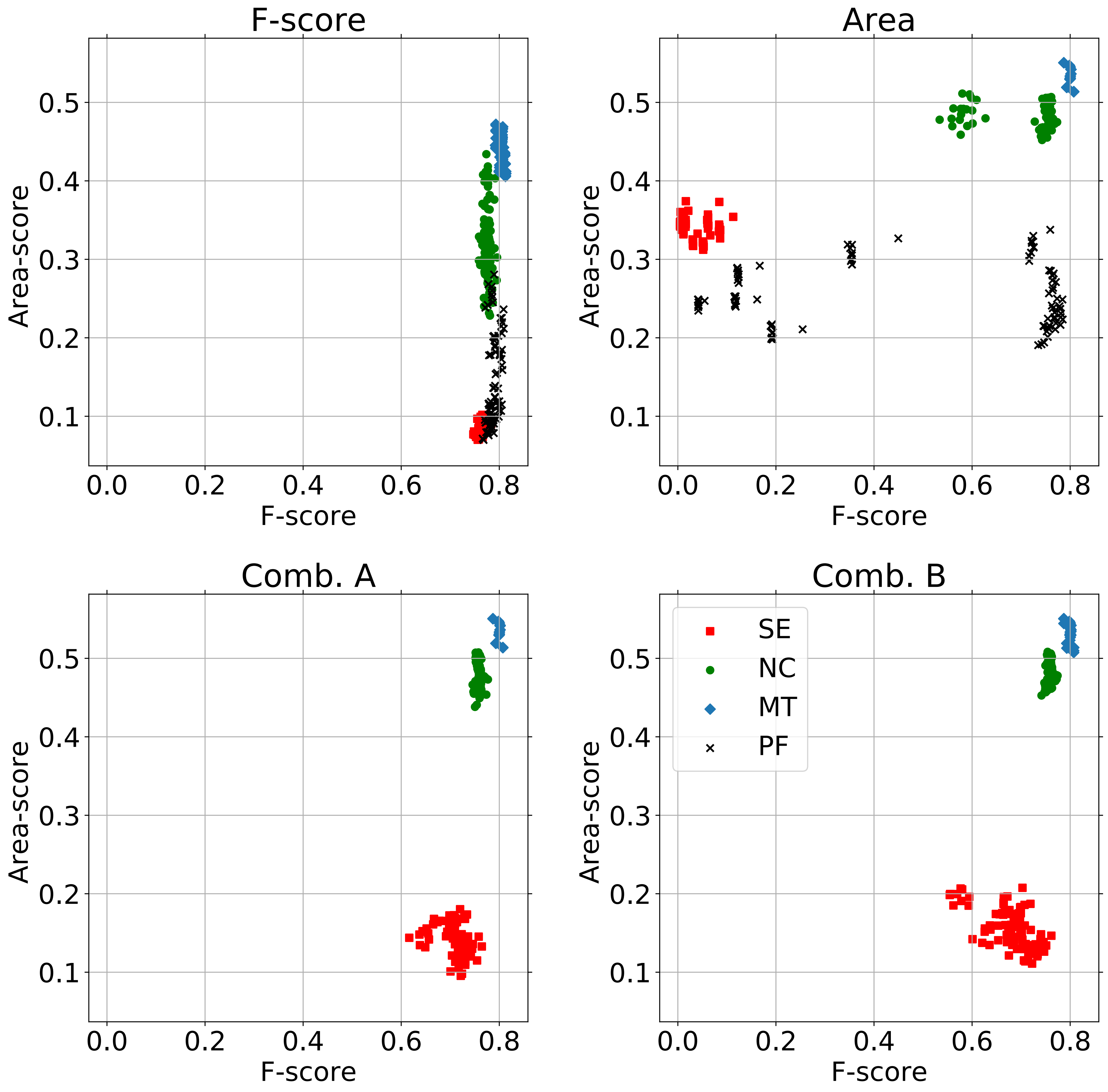
Combined score B: $\sqrt[3]{(1 - OM)(1 - UM)F_1}$

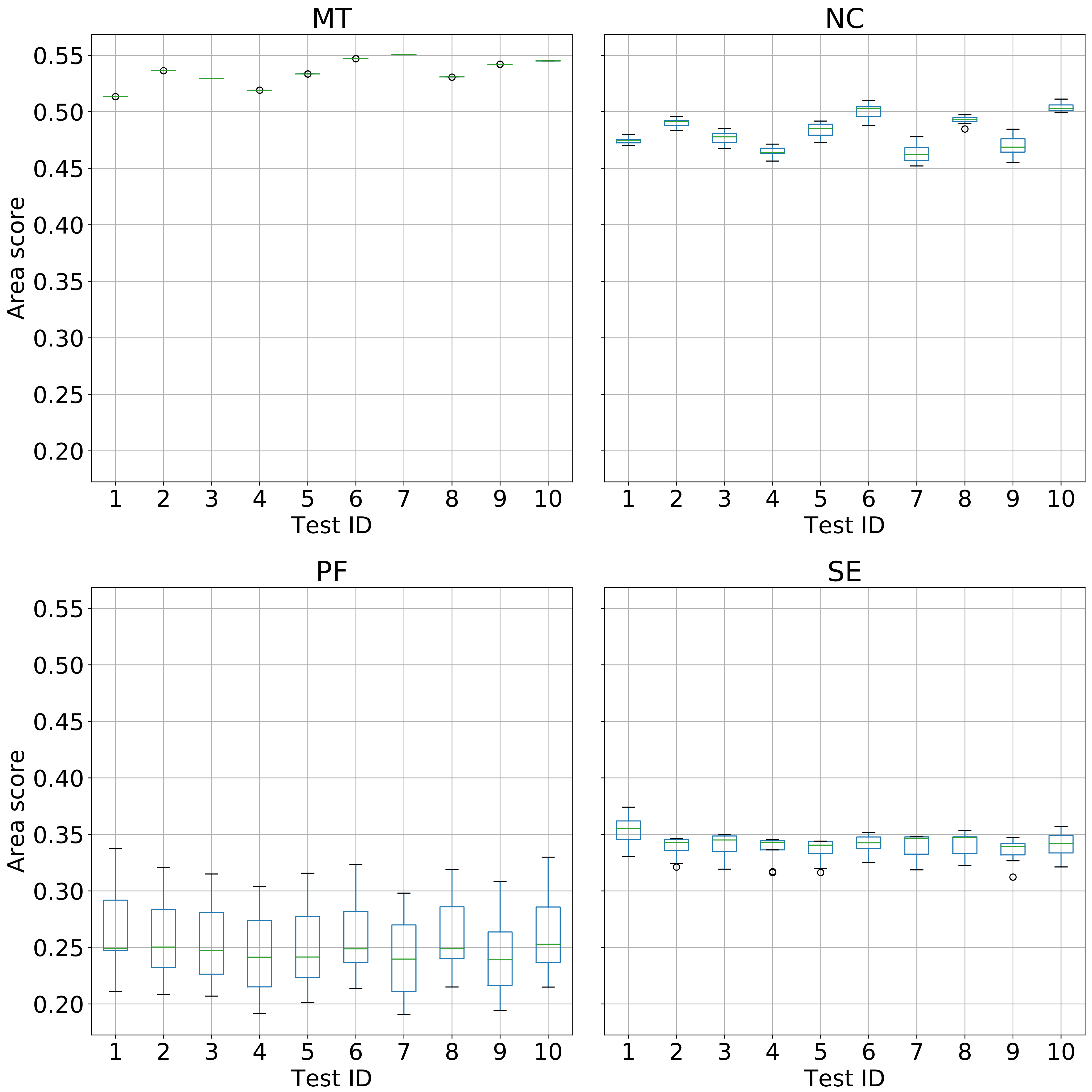
Evaluation

Parameter Optimisation

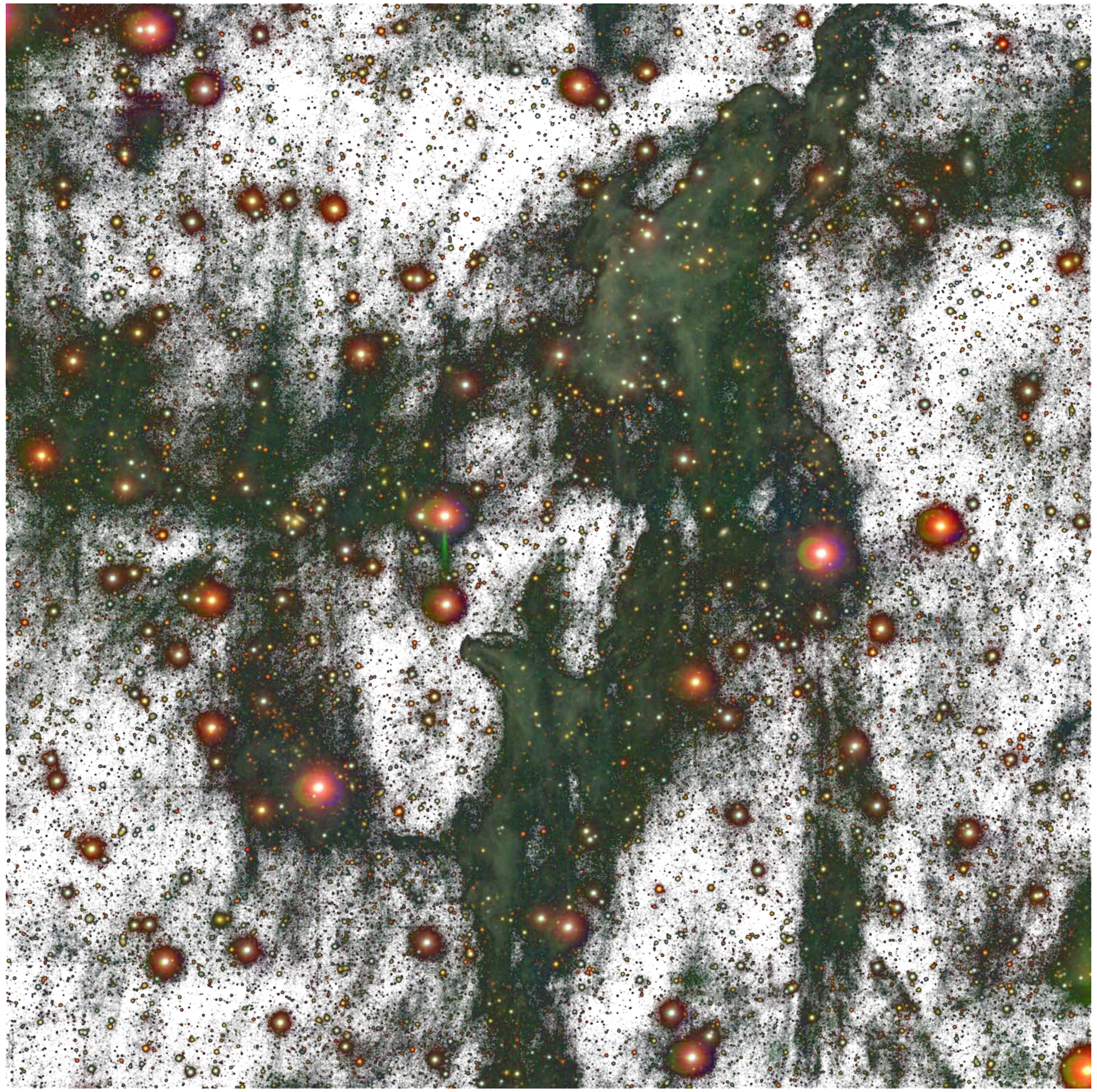
- Ten simulated images are used
- Bayesian optimisation is performed on each image for each quality measure
- Each of the settings is tested on the remaining 9 images

Results - Summary





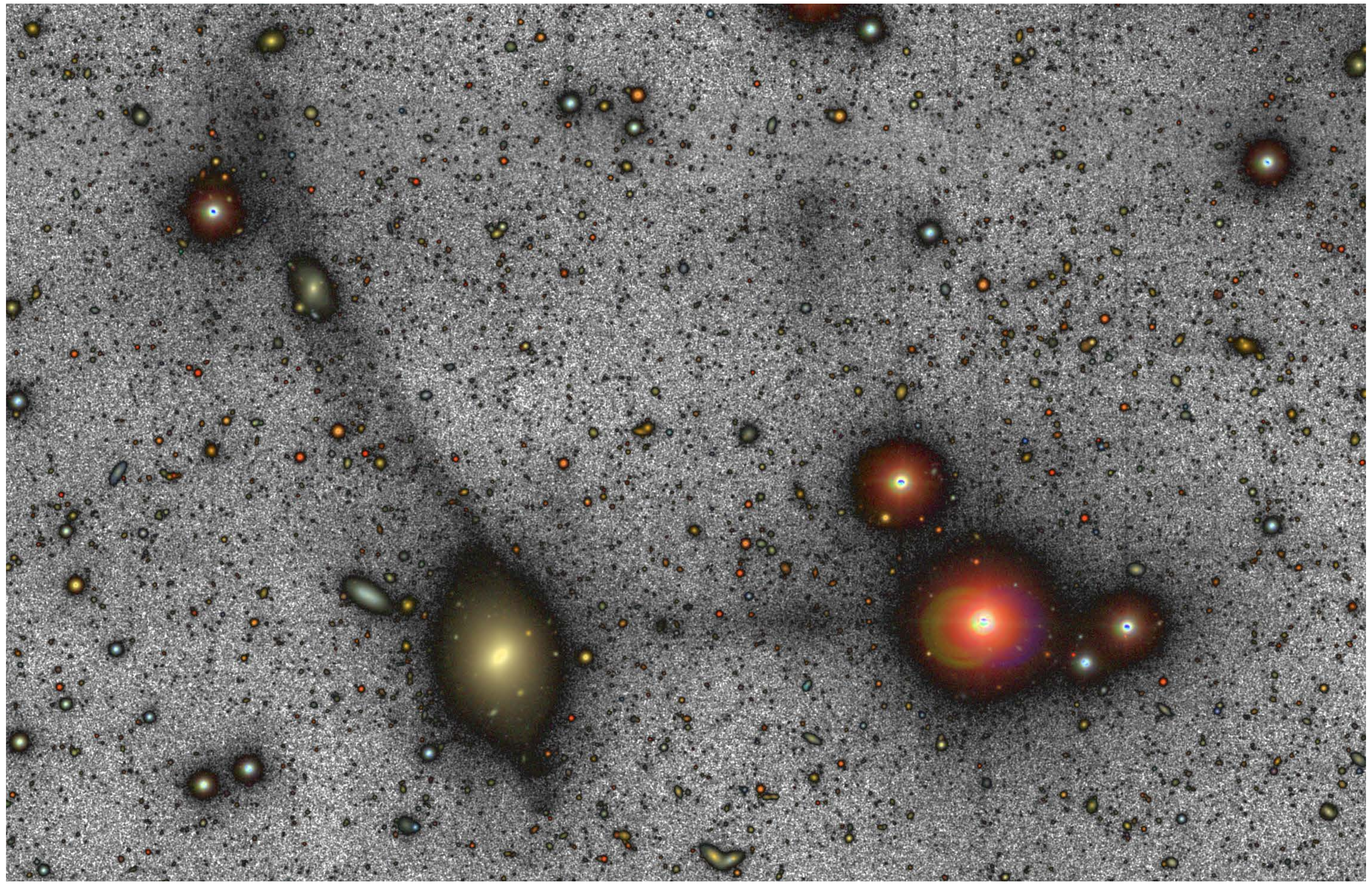
Real images - Two IAC Stripe 82 examples



Galactic cirrus

$\mu_{g,lim}$ = 29.1 mag/arcsec^2 (3σ, 100 arcsec^2)

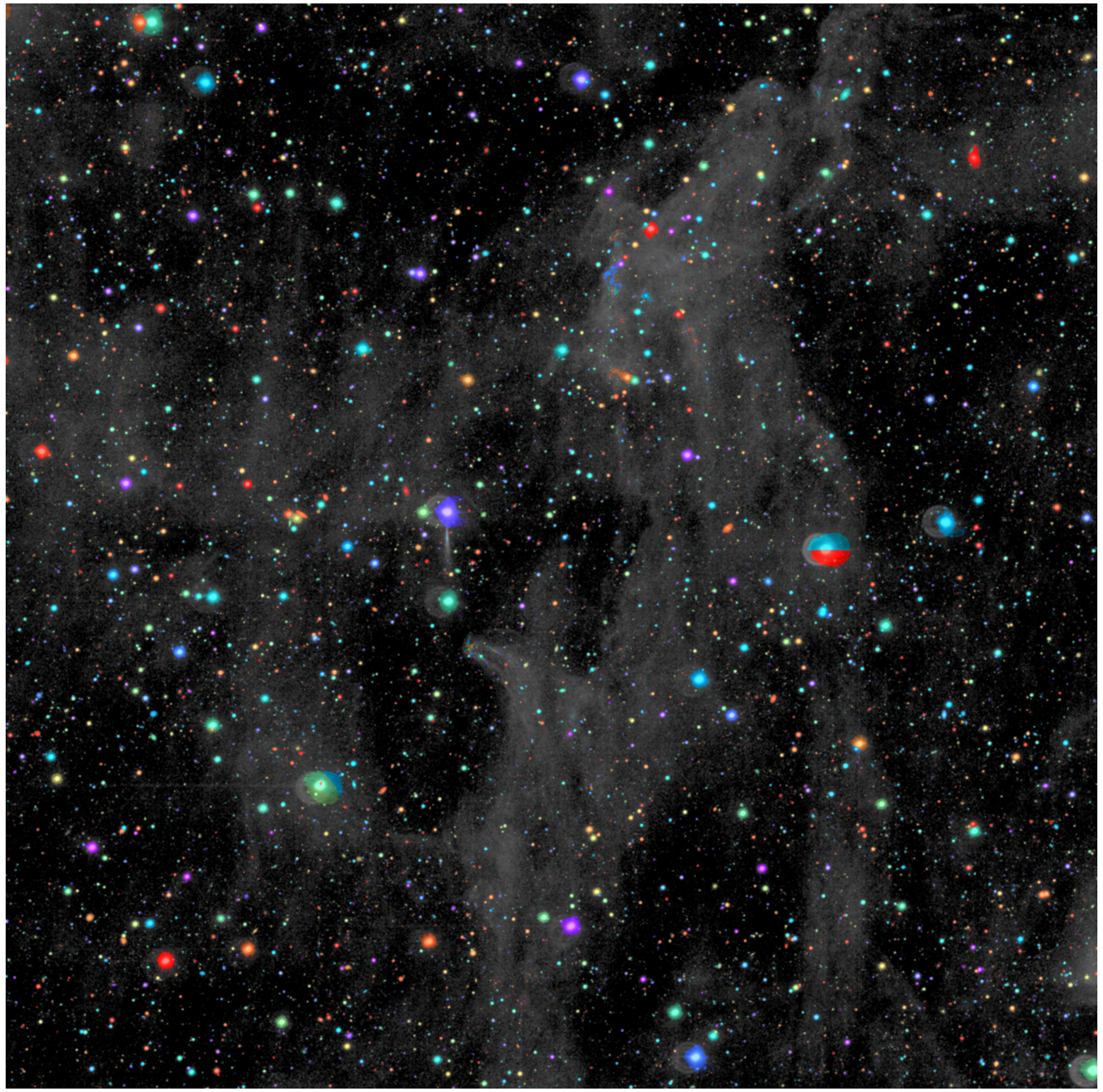
http://research.iac.es/proyecto/stripe82/



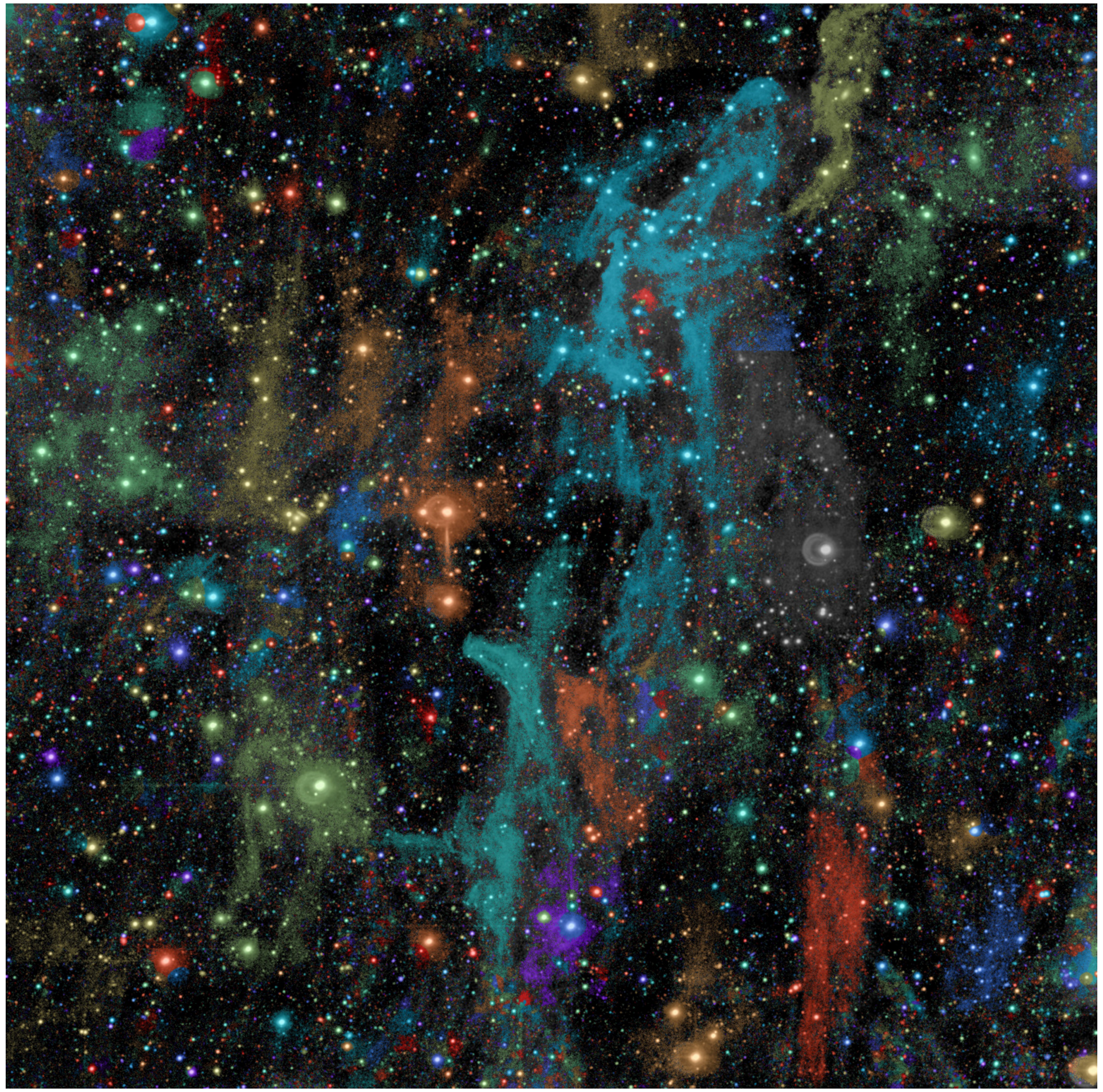
Tidal streams, bright sources

Results - Galactic cirrus

SExtractor



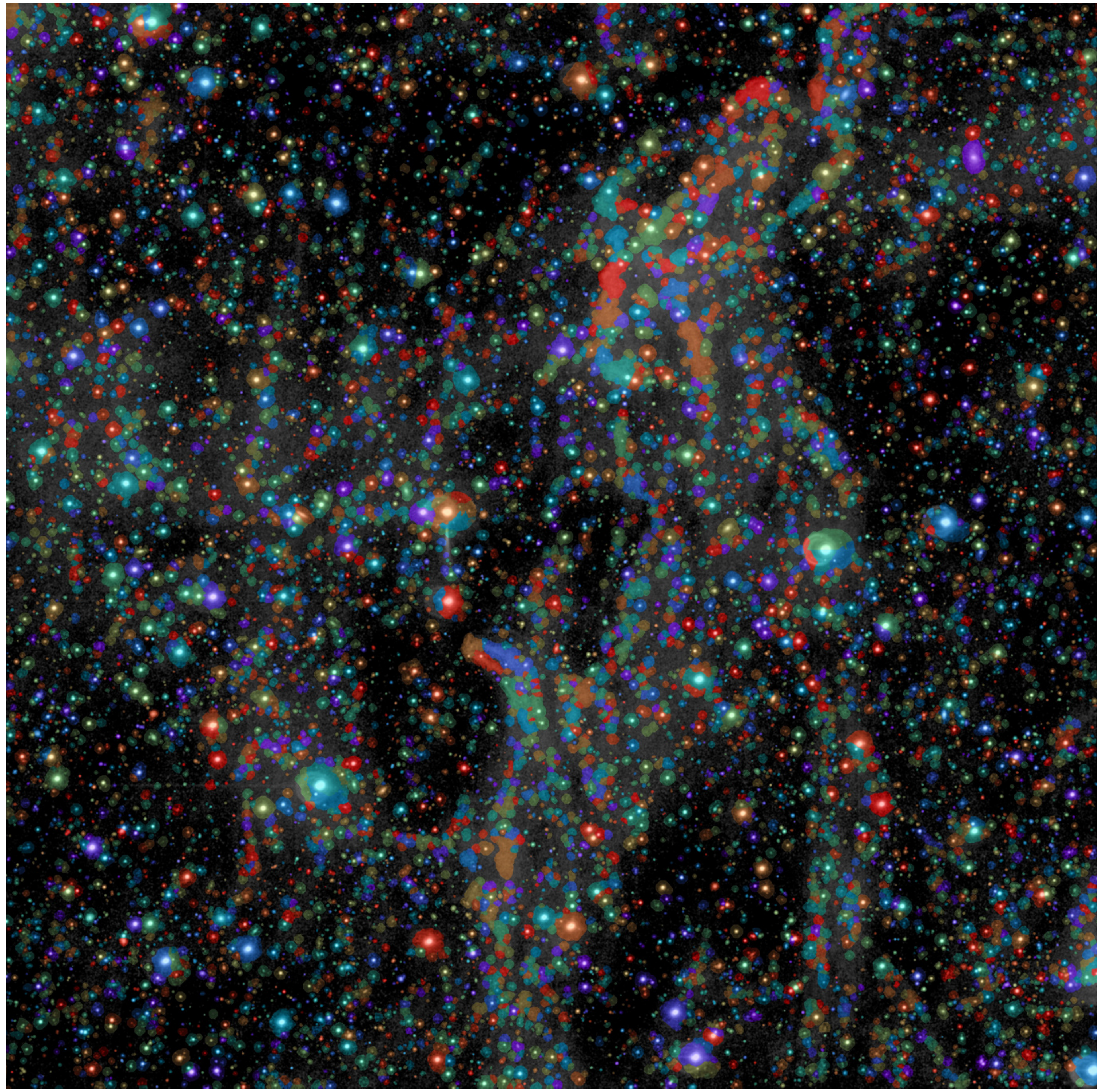
Optimised for F-score



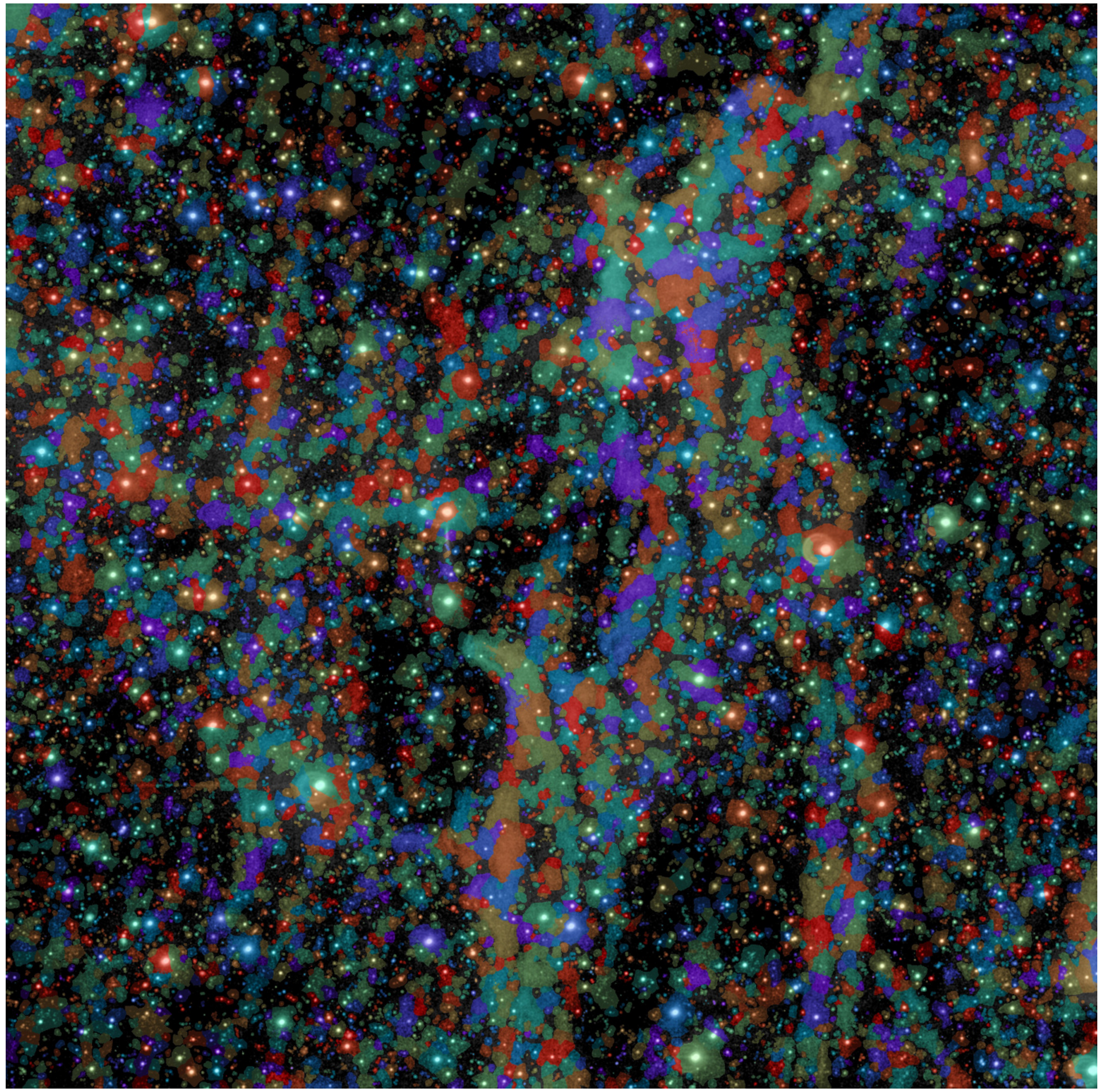
Optimised for Area score

Results - Galactic cirrus

ProFound



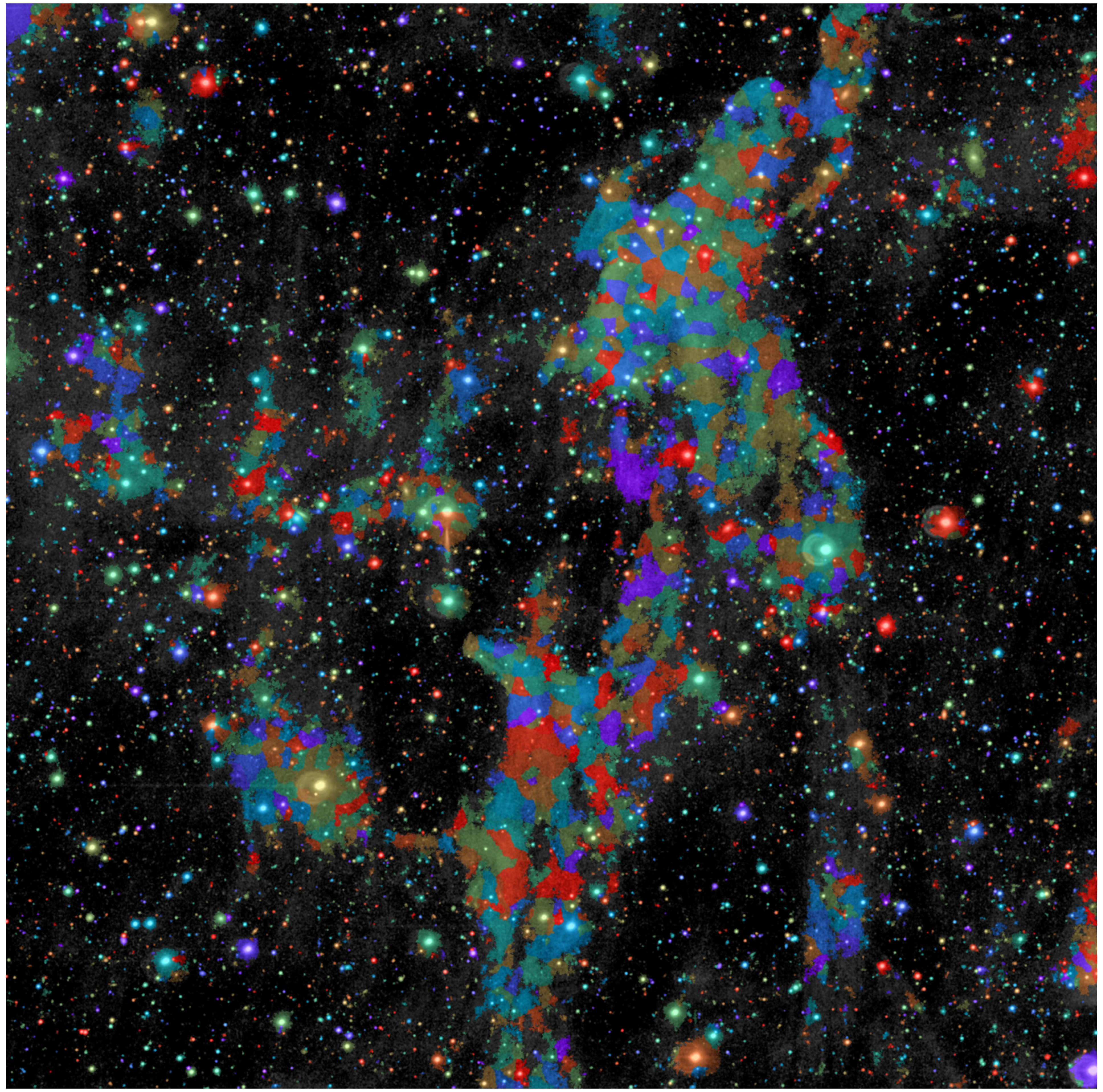
Optimised for F-score



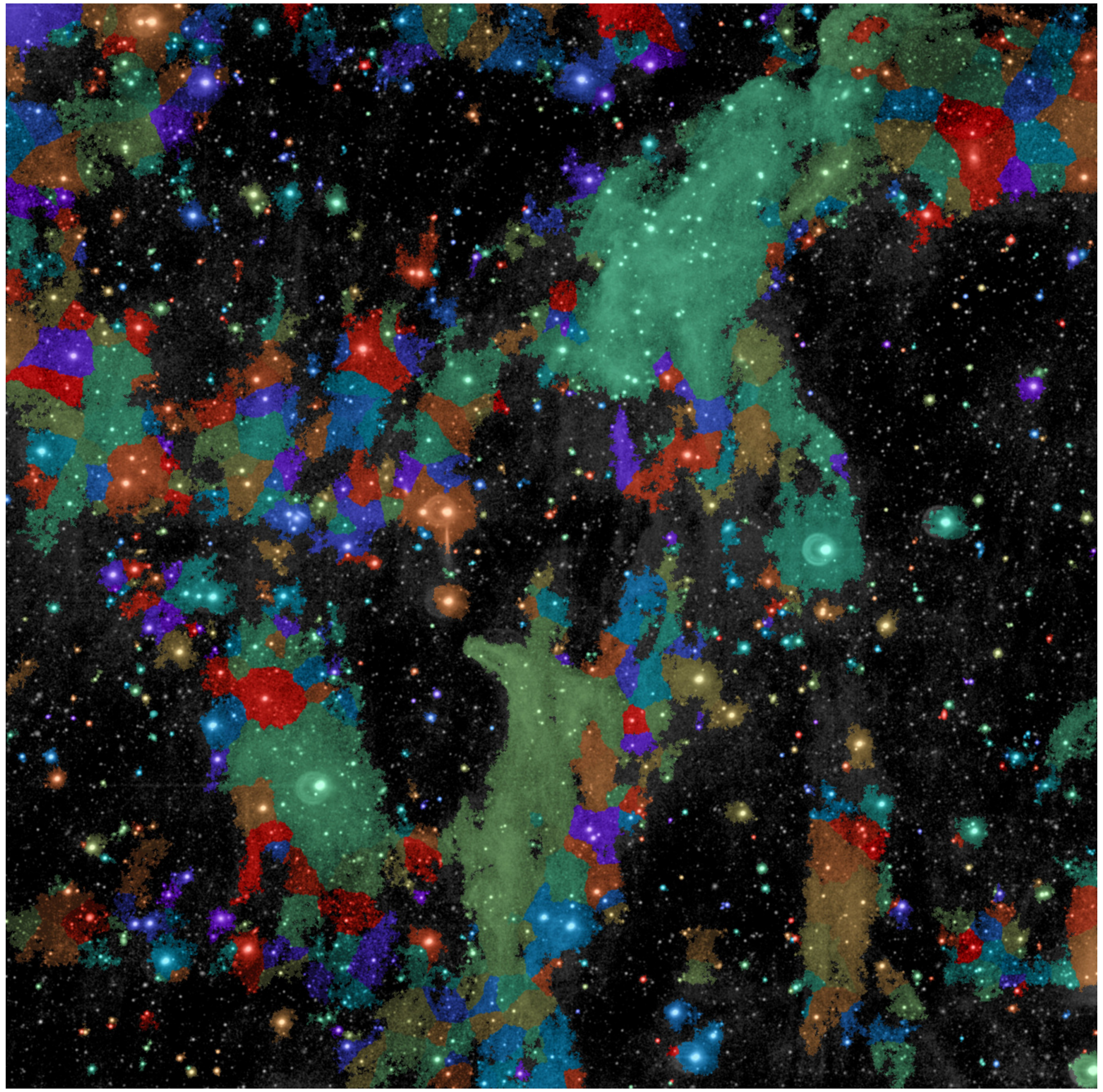
Optimised for Area score

Results - Galactic cirrus

NoiseChisel



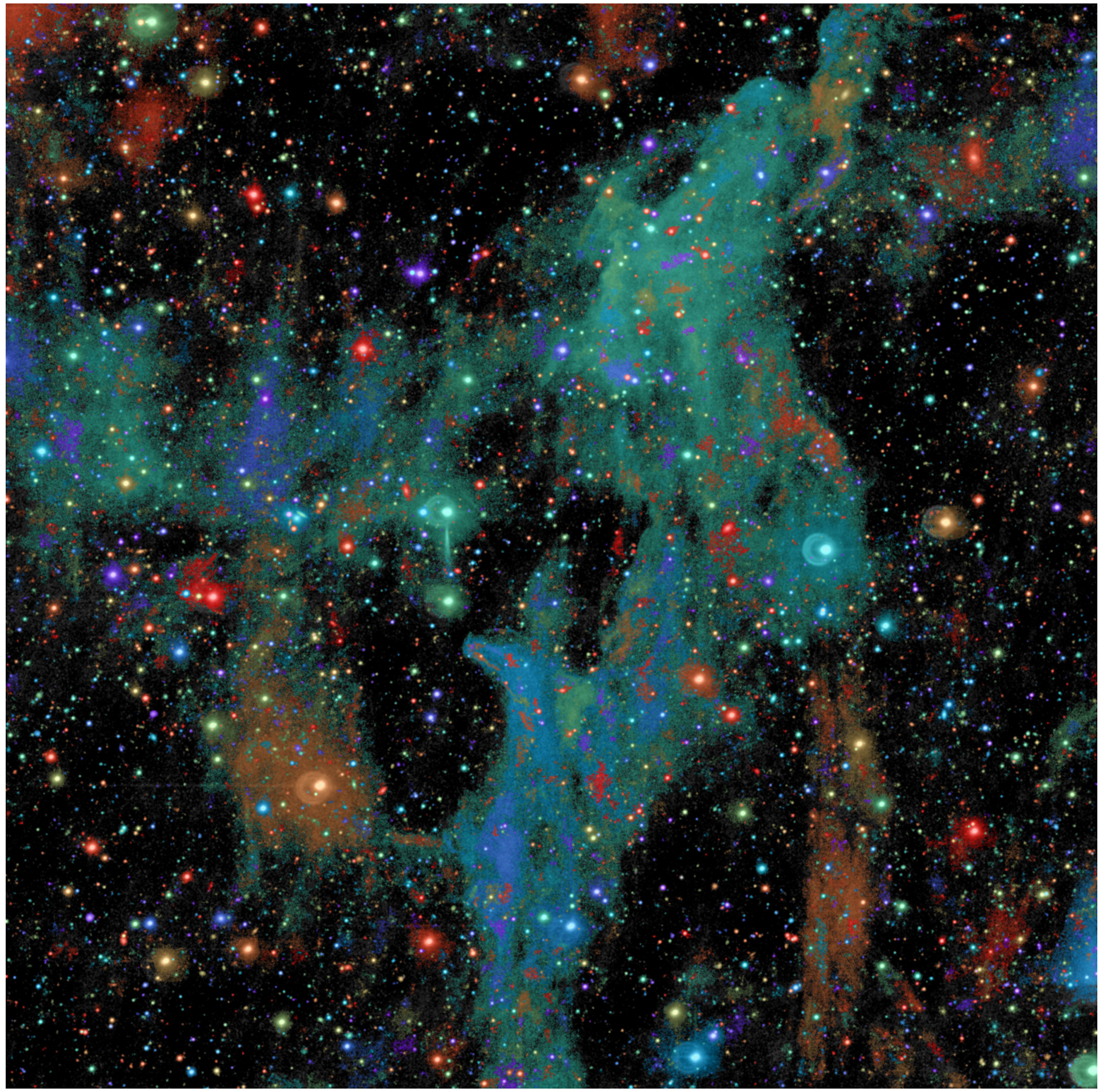
Optimised for F-score



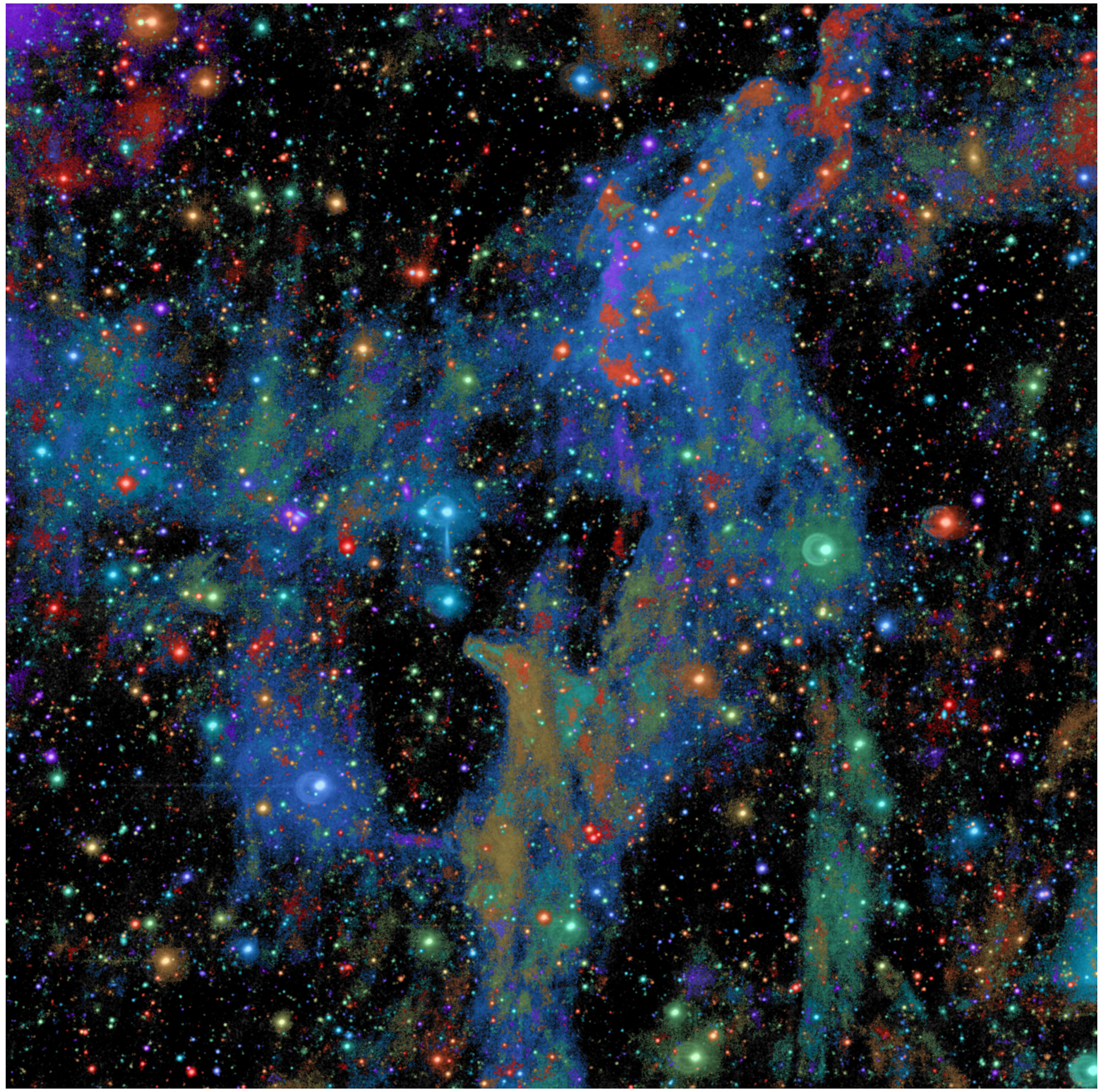
Optimised for Area score

Results - Galactic cirrus

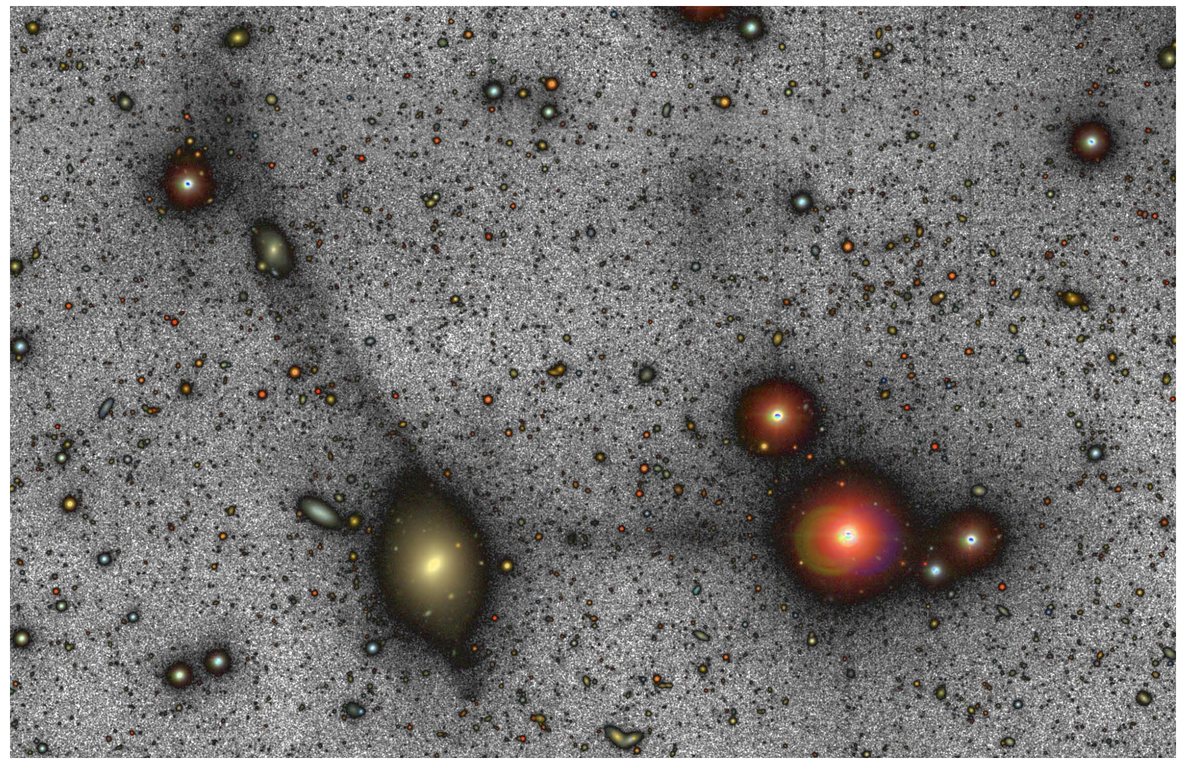
Max-Tree Objects



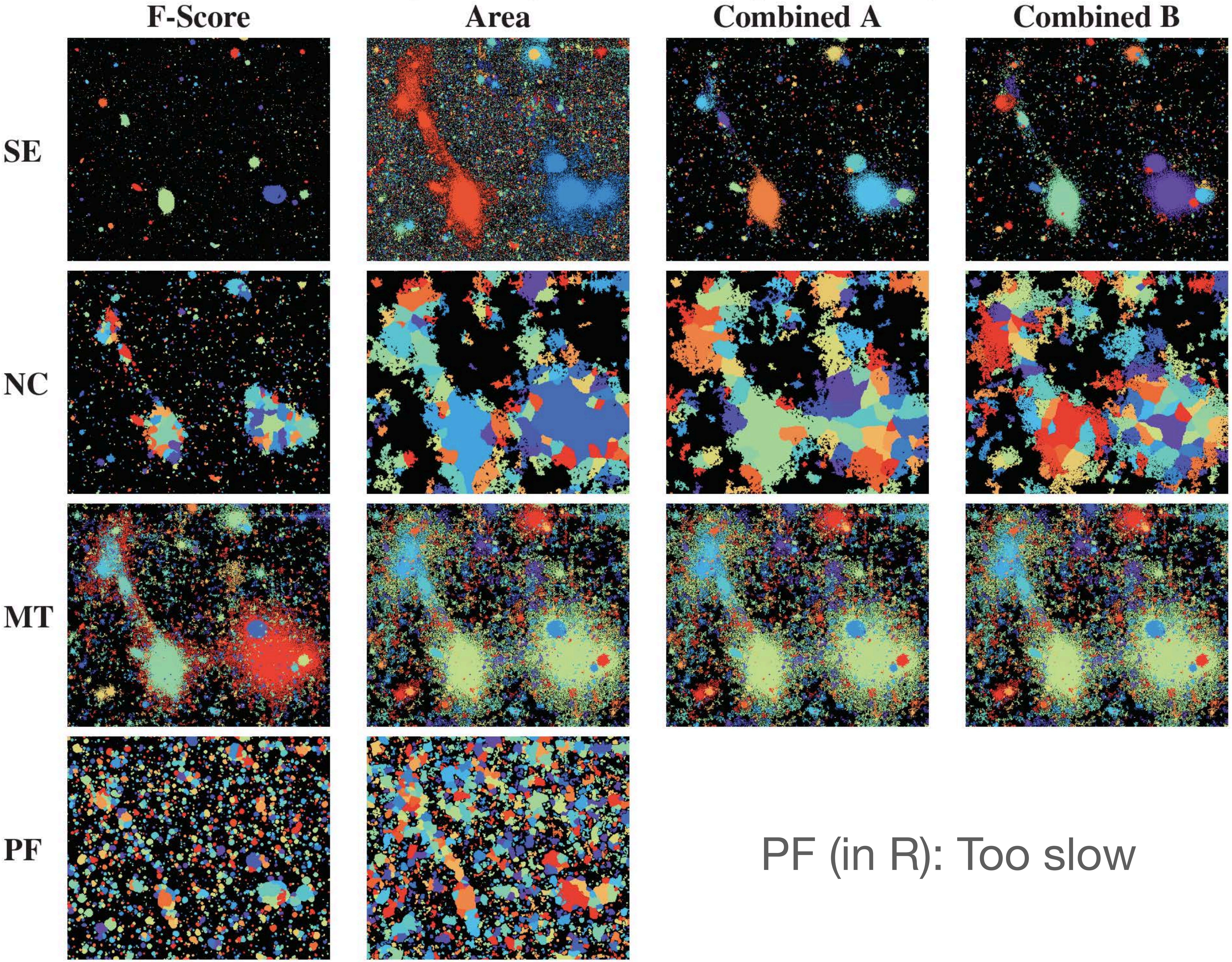
Optimised for F-score



Optimised for Area score

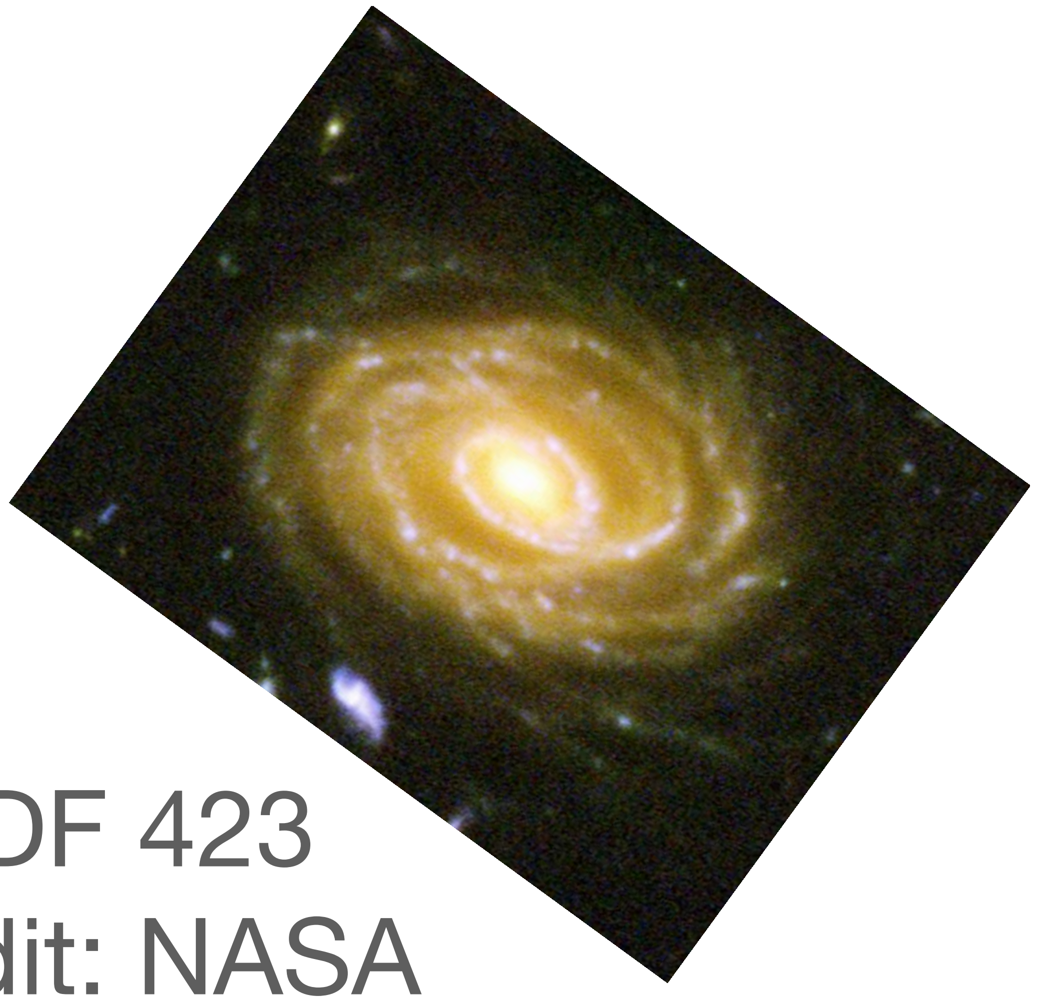


Tidal streams, bright sources

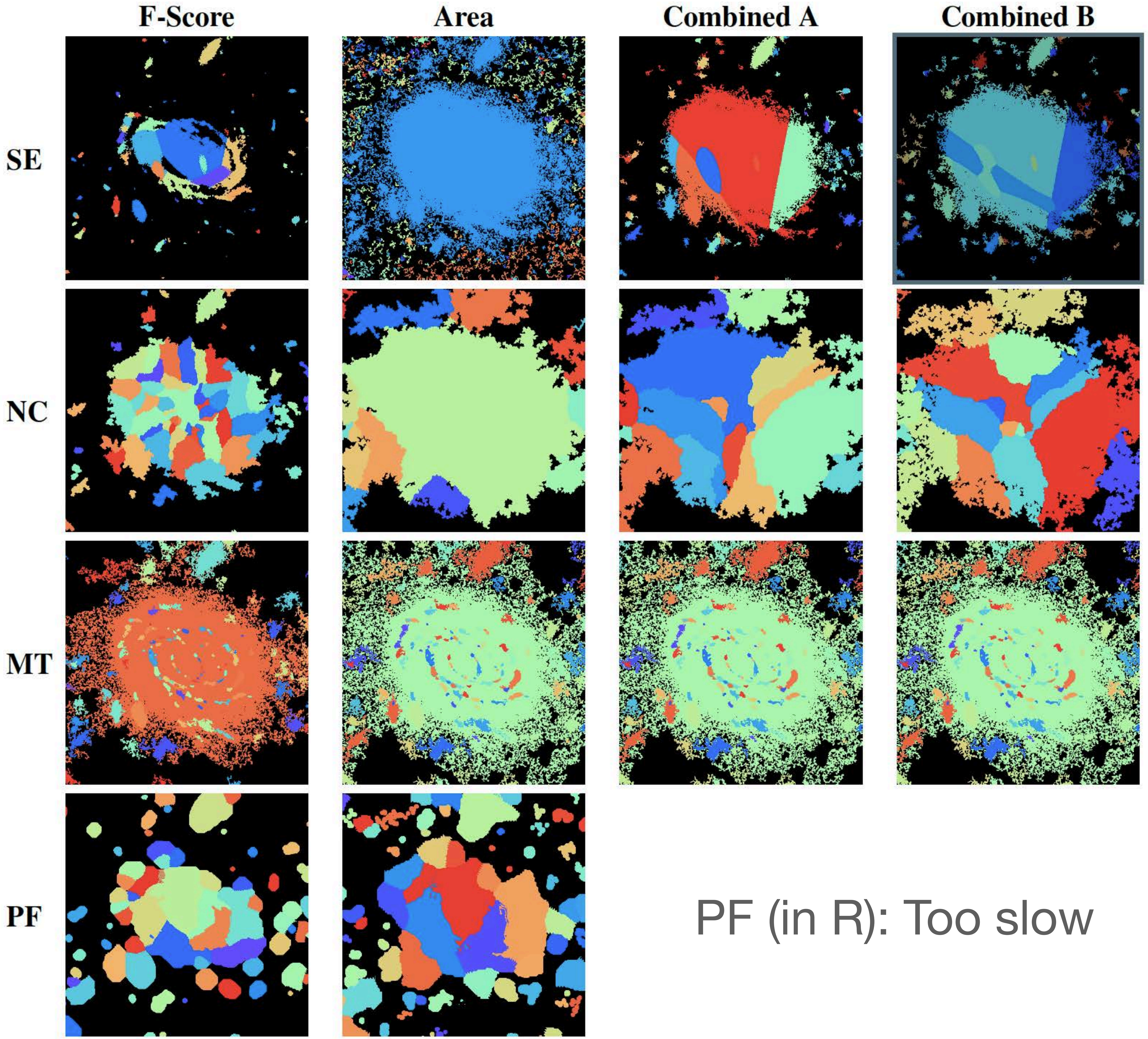


PF (in R): Too slow

HUDF



UDF 423
Credit: NASA



PF (in R): Too slow

Background values

Talk to me for details!

- Mean background value of simulated image is zero
- Each algorithm has its internal estimator (can be imrpov)

- Both PF and SE consistently overestimated the background: $O(10^{-1}\sigma)$

- MT underestimated the value: area score $O(-10^{-1}\sigma)$ and F-score $O(-10^{-2}\sigma)$

- NC showed the strongest performance: $O(\pm 10^{-3}\sigma)$

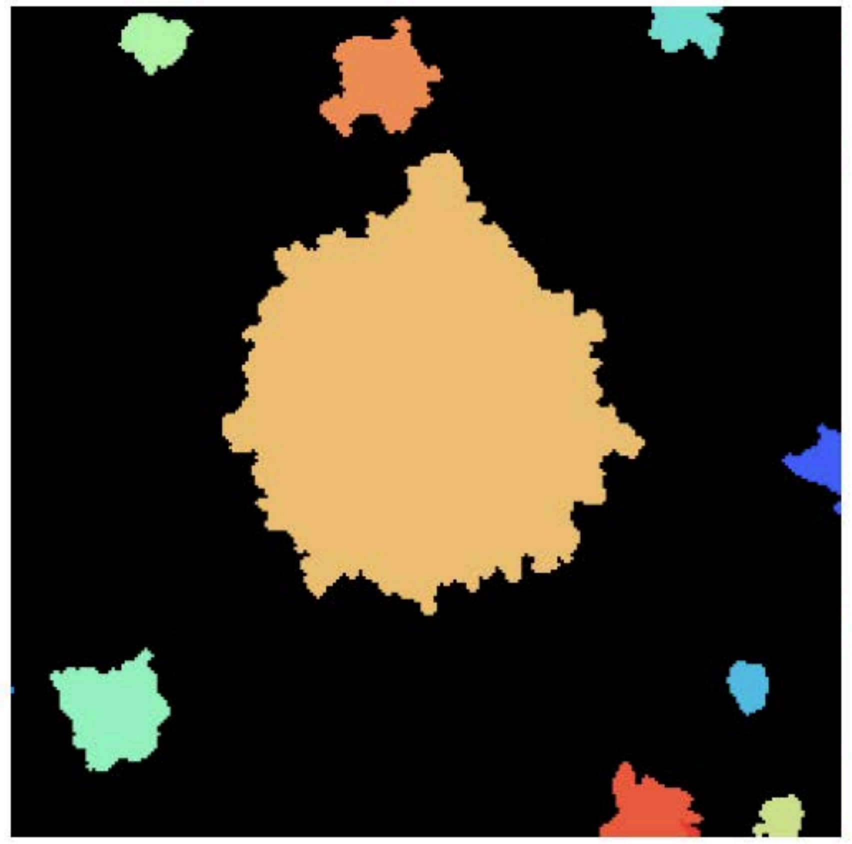
Concluding remarks

How can these results help you?

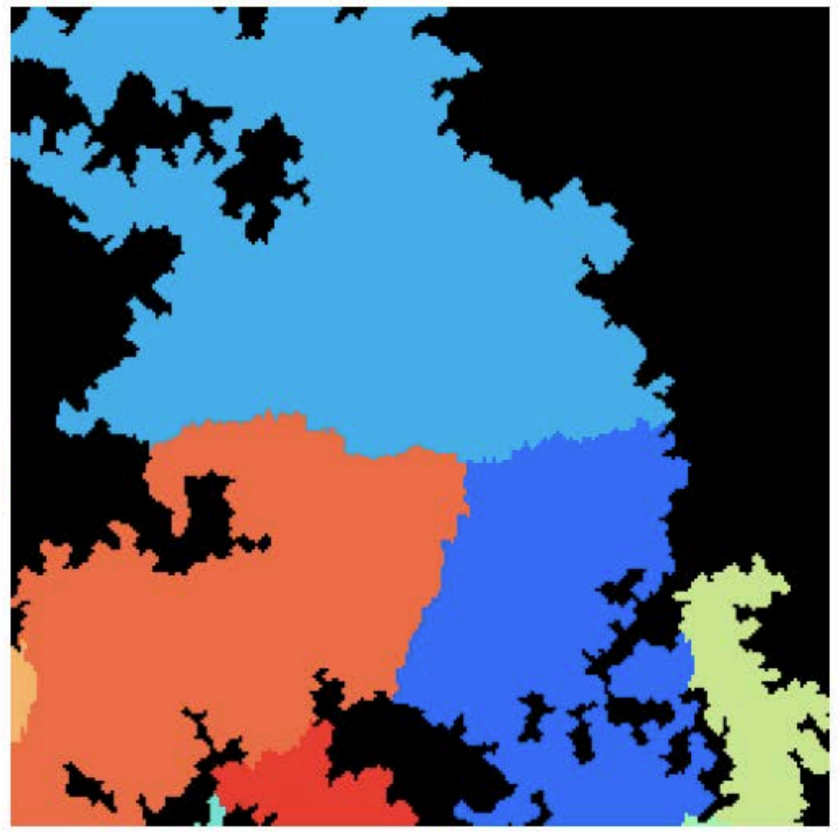
- Robust, optimised parameters for detection algorithms. SCARLET?
- Evaluation: MT overall most stable and consistent performance (C. Haigh et al. re-submitted to A&A)

	MTObjects	NoiseChisel	ProFound	SExtractor
Optimised parameters	2	20	8	6
Language	Python/C	C	R	C
Clean edges of detected objects	-	✓	✓	Sometimes
Detects galaxy close to star (Stripe 82)	✓	Fragmented	-	Fragmented
Detects cirrus (Stripe 82)	✓	✓	-	Sometimes
Isolates spiral substructures (HUDF)	✓	-	-	-

- Be aware (beware) of each algorithms limits and failures: do you care about nested objects or only faint outskirts? de-blending?



(a) A ‘whole’ detected galaxy.

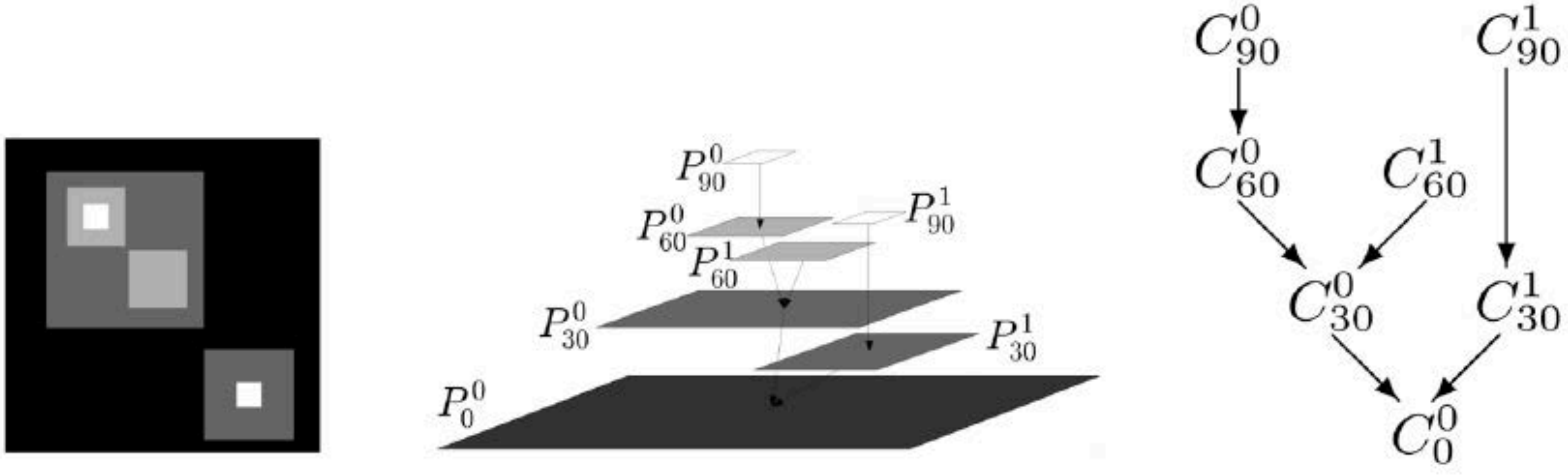


(b) A fragmented galaxy.

Max Tree Objects: Concept

Component Trees

- Based on decomposition of image into connected components



(a) 2D image (b) Peak components (c) Max-Tree

Teeninga, Moschini, Trager & Wilkinson (2016)