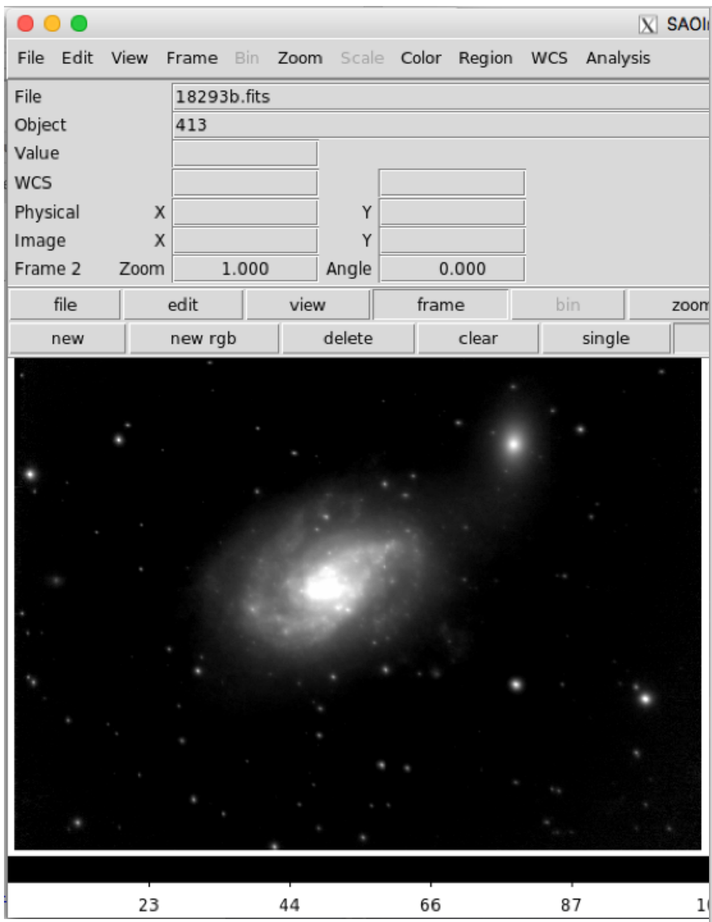
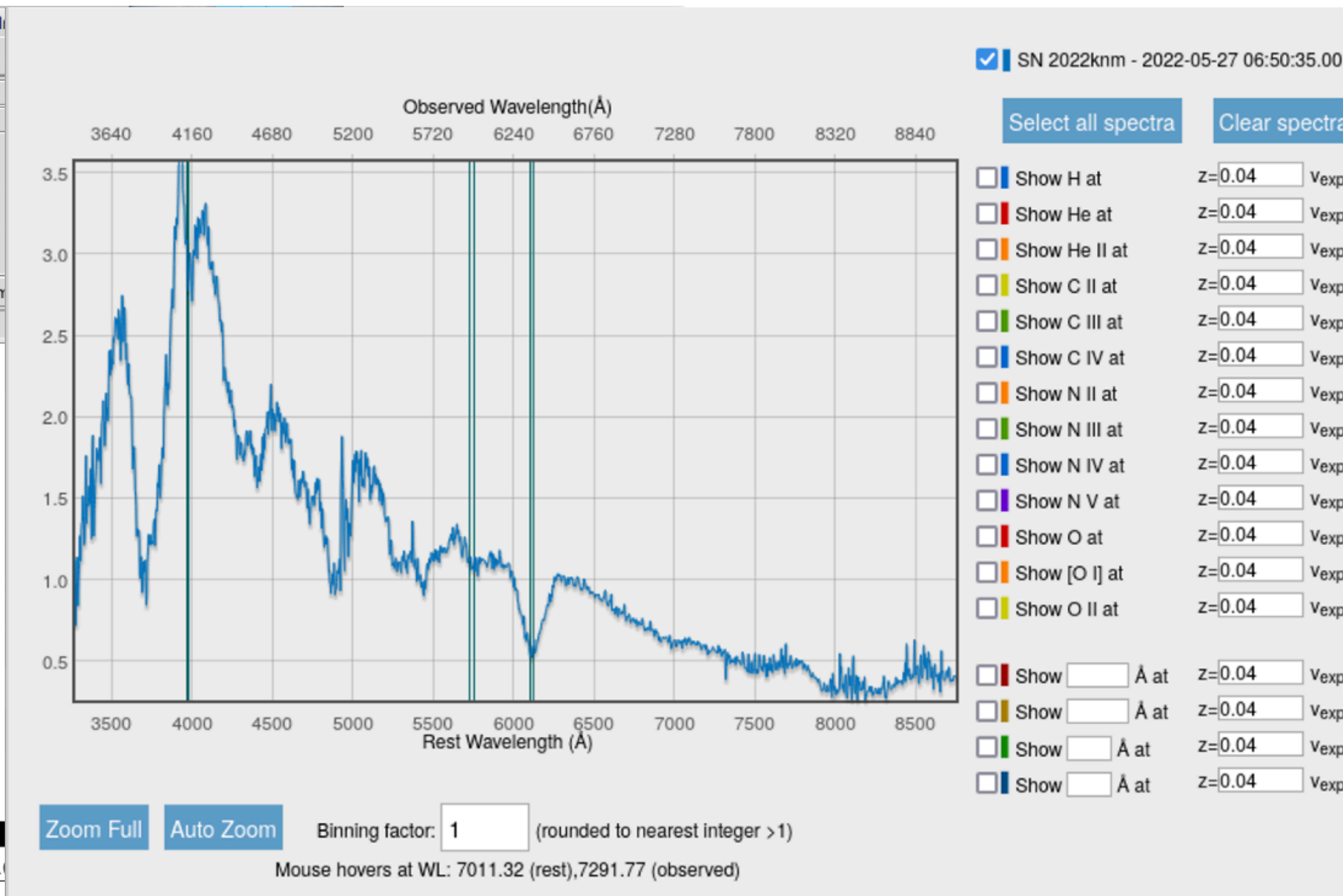
Astronomical imaging & spectroscopy

Seppo Mattila





Course report

- The minimum length is 9 pages (12 pt font) of text (2 pages based on each of the four practical sessions + 0.5 page introduction + 0.5 page summary) + figures, tables, references
- For reporting the work done in each of the sessions please follow the advice of the teachers
- Keep in mind the learning outcomes (slide 4) when preparing your report
- For writing the report you can use any word processing software that you are familiar with. Please, save the report as PDF
- **Deadline for handing-in the reports to sepmat@utu.fi on 1st August**

Learning outcomes

After completing the course the students should be able to:

(1) Describe the principles behind some advanced astronomical imaging techniques and identify suitable topics in astrophysics that can be studies with them;

(2) Understand the physics behind some of the most important medical imaging modalities and describe their value in clinical applications;

(3) Identify and discuss the differences and similarities in the challenges faced when analyzing data in these two different disciplines;

(4) Describe the theoretical basis and suitability of several image/signal processing and analysis methods commonly used in astronomy and medical imaging;

(5) Identify suitable algorithms and apply them to astronomical and/or medical imaging datasets to enhance their scientific and/or clinical value;

(6) Produce a written course report

Exercise 1: Convolution

Convolution

"real" signal

additive noise

$$b(\vec{x}) = f(\vec{x}) * p(\vec{x}) + n(\vec{x})$$

observed signal

PSF

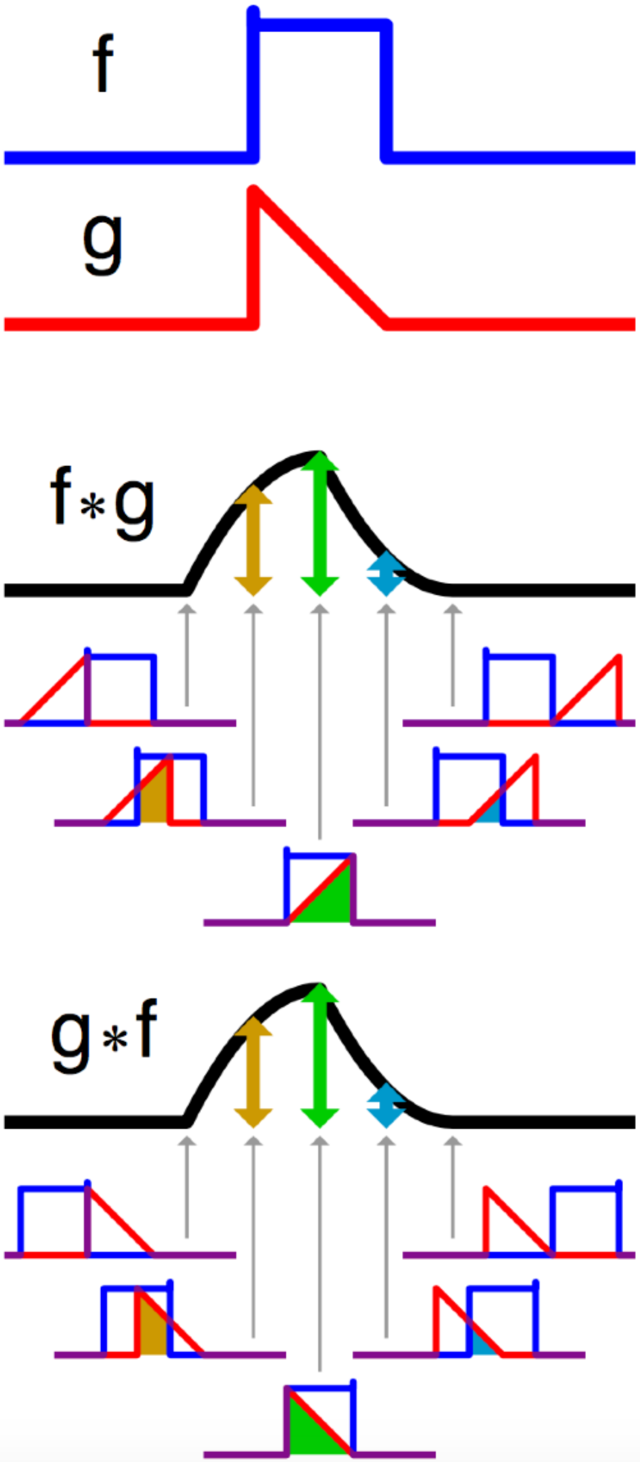
In the case of 1-D functions

$$(f * g)(x) = \int_{-\infty}^{\infty} f(\tau) g(x - \tau) \ \ d\tau$$

In the case of discrete 1-D functions

$$(f * g)_j = \sum_{k=-m/2+1}^{m/2} f_k g_{j-k}$$

Convolution



$$(f * g)(x) = \int_{-\infty}^{\infty} f(\tau) g(x - \tau) \; d\tau$$

$$(f * g)_j = \sum_{k=-m/2+1}^{m/2} f_k g_{j-k}$$

astropy

A Community Python Library for Astronomy

Available Kernels

`AiryDisk2DKernel`(radius, **kwargs)	2D Airy disk kernel.
`Box1DKernel`(width, **kwargs)	1D Box filter kernel.
`Box2DKernel`(width, **kwargs)	2D Box filter kernel.
`CustomKernel`(array)	Create filter kernel from list or array.
`Gaussian1DKernel`(stddev, **kwargs)	1D Gaussian filter kernel.
`Gaussian2DKernel`(x_stddev[, y_stddev, theta])	2D Gaussian filter kernel.
`RickerWavelet1DKernel`(width, **kwargs)	1D Ricker wavelet filter kernel (sometimes known as a “Mexican Hat” kernel).
`RickerWavelet2DKernel`(width, **kwargs)	2D Ricker wavelet filter kernel (sometimes known as a “Mexican Hat” kernel).
`Model1DKernel`(model, **kwargs)	Create kernel from 1D model.
`Model2DKernel`(model, **kwargs)	Create kernel from 2D model.
`Moffat2DKernel`(gamma, alpha, **kwargs)	2D Moffat kernel.
`Ring2DKernel`(radius_in, width, **kwargs)	2D Ring filter kernel.
`Tophat2DKernel`(radius, **kwargs)	2D Tophat filter kernel.
`Trapezoid1DKernel`(width[, slope])	1D trapezoid kernel.
`TrapezoidDisk2DKernel`(radius[, slope])	2D trapezoid kernel.

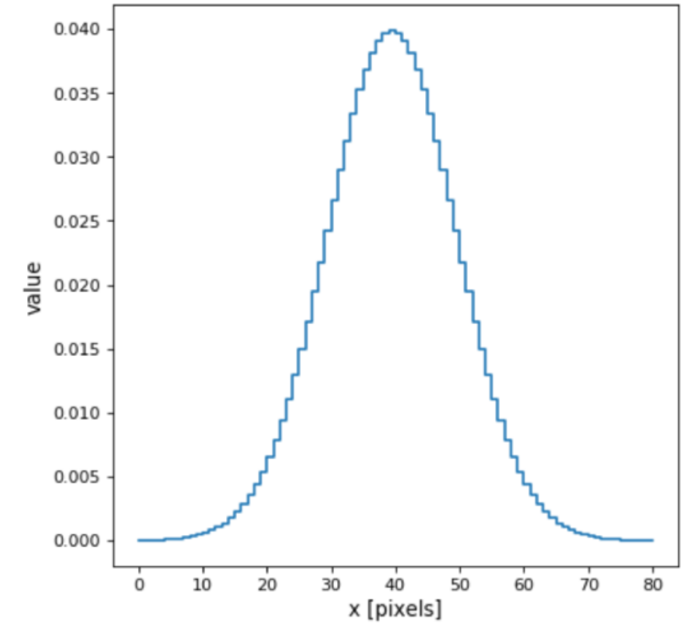
Gaussian1DKernel ¶

class `astropy.convolution.` **`Gaussian1DKernel`** (*stddev*, ***kwargs*) [edit on github][source]

Bases: `astropy.convolution.Kernel1D`

1D Gaussian filter kernel.

The Gaussian filter is a filter with great smoothing properties. It is isotropic and does not produce artifacts.



Parameters:

stddev : number

Standard deviation of the Gaussian kernel.

x_size : odd int, optional

Size of the kernel array. Default = 8 * stddev

mode : str, optional

One of the following discretization modes:

- 'center' (default)
  Discretize model by taking the value at the center of the bin.
- 'linear_interp'
  Discretize model by linearly interpolating between the values at the corners of the bin.
- 'oversample'
  Discretize model by taking the average on an oversampled grid.
- 'integrate'
  Discretize model by integrating the model over the bin. Very slow.

factor : number, optional

Factor of oversampling. Default factor = 10. If the factor is too large, evaluation can be very slow.

Gaussian2DKernel

class `astropy.convolution.` **`Gaussian2DKernel`** (*x_stddev*, *y_stddev=None*, *theta=0.0*, ***kwargs*) ¶ [source] [edit on github]

Bases: `astropy.convolution.Kernel2D`

2D Gaussian filter kernel.

The Gaussian filter is a filter with great smoothing properties. It is isotropic and does not produce artifacts.

Parameters: **x_stddev** : float

Standard deviation of the Gaussian in x before rotating by theta.

y_stddev : float

Standard deviation of the Gaussian in y before rotating by theta.

theta : float

Rotation angle in radians. The rotation angle increases counterclockwise.

x_size : odd int, optional

Size in x direction of the kernel array. Default = 8 * stddev.

y_size : odd int, optional

Size in y direction of the kernel array. Default = 8 * stddev.

mode : str, optional

One of the following discretization modes:

- 'center' (default)

  Discretize model by taking the value at the center of the bin.

- 'linear_interp'

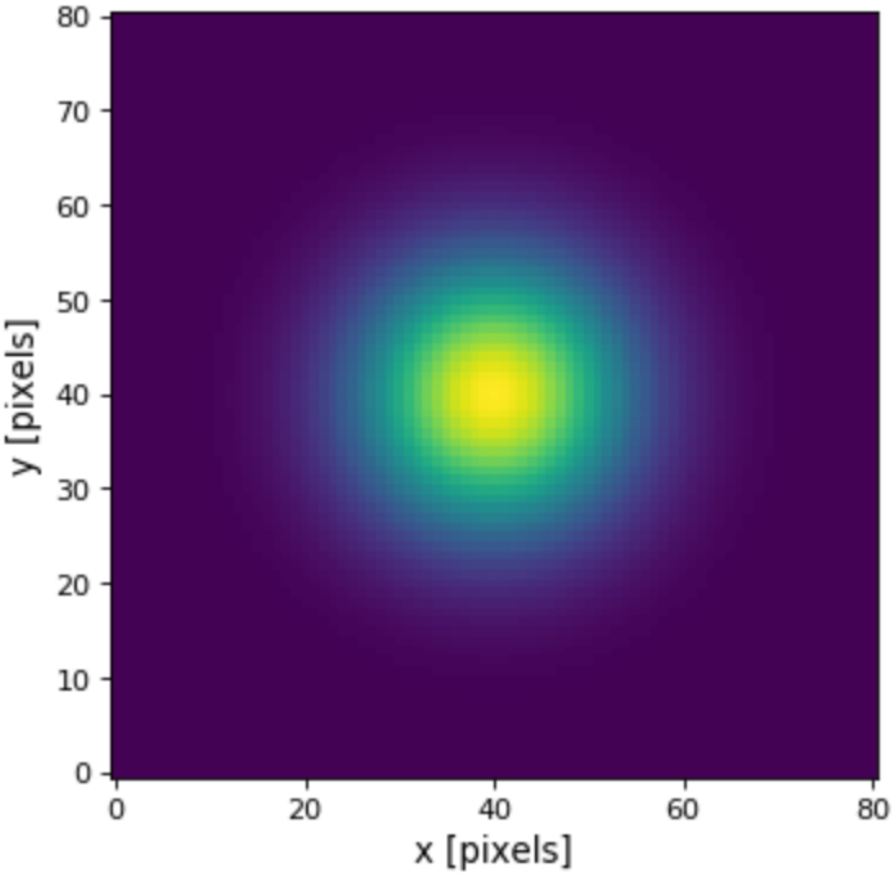
  Discretize model by performing a bilinear interpolation between the values at the corners of the bin.

- 'oversample'

  Discretize model by taking the average on an oversampled grid.

- 'integrate'

  Discretize model by integrating the model over the bin.



Exercise 1: practical work

Experiment using different types of convolution kernels for 1D or 2D data, present your experiment in the report

Gaussian1DKernel¶

class `astropy.convolution.` **Gauss**

Bases: **astropy.convolution.Kernel1**

1D Gaussian filter kernel.

The Gaussian filter is a filter with great smo

Parameters: **stddev** : number

Standard deviation

x_size : odd int, optional

Size of the kernel ar

mode : str, optional

One of the following

- 'center' (default)

  Discretize mo

- 'linear_interp'

  Discretize mo
  the bin.

- 'oversample'

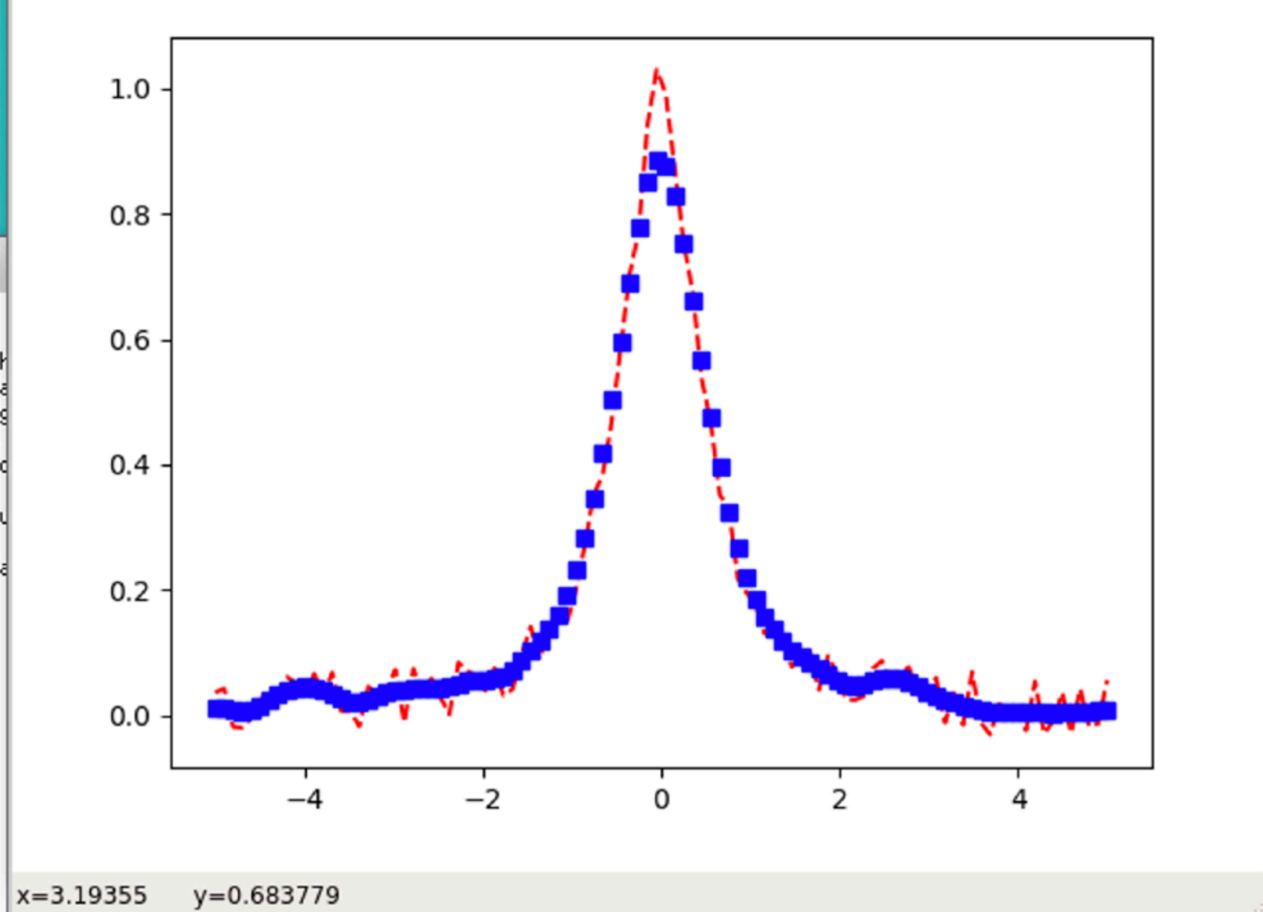
  Discretize mo

- 'integrate'

  Discretize mo

factor : number, optional

Factor of oversamp
be very slow.



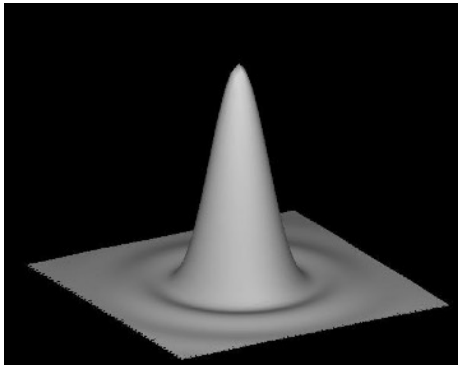
not3@course2021: ~

not3@course2021: ~ 80x15

```
>>>
>>> import numpy as np
>>> from astropy.modeling.functional_models import Lorentz1D
>>> from astropy.convolution import convolve, Gaussian1DKernel, Box1DKernel
>>> lorentz = Lorentz1D(1, 0, 1)
>>> x = np.linspace(-5, 5, 100)
>>> data_1D = lorentz(x) + 0.1 * (np.random.rand(100) - 0.5)
>>> gauss_kernel = Gaussian1DKernel(2)
>>> smoothed_data_gauss = convolve(data_1D, gauss_kernel)
>>> import matplotlib.pyplot as plt
>>> plt.plot(x, data_1D, 'r--', x, smoothed_data_gauss, 'bs')
[<matplotlib.lines.Line2D object at 0x7f975c025ad0>, <matplotlib.lines.Line2D ob
ject at 0x7f975c025b90>]
>>> plt.show()
```

Exercise 2: image matching and subtraction of astronomical images to detect variability over time

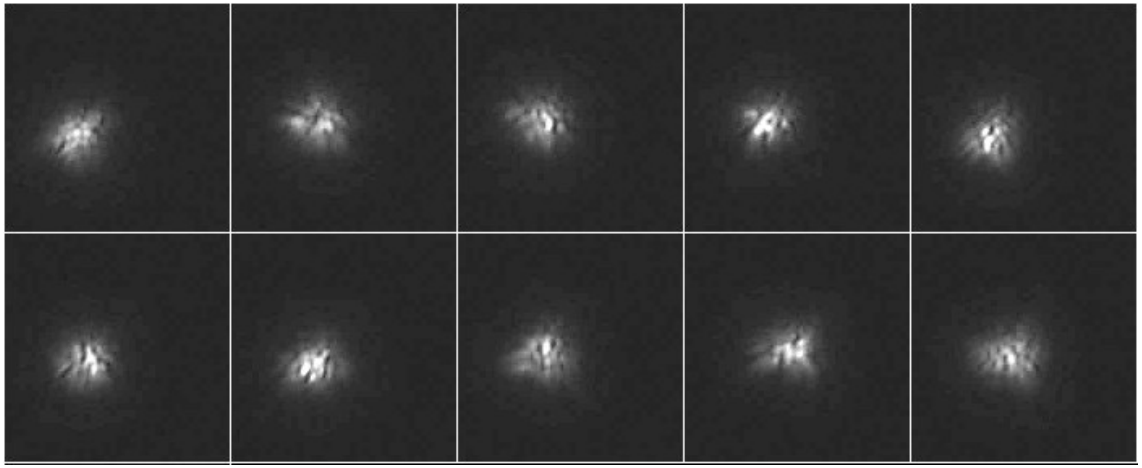
Point spread function (PSF)



Ideal (diffraction limited) PSF if no atmosphere

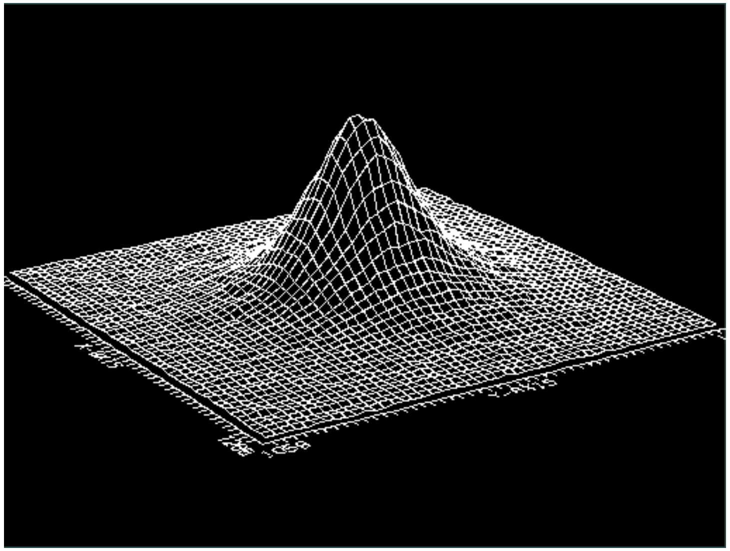
$\theta \sim 1.22 \times \lambda / D$

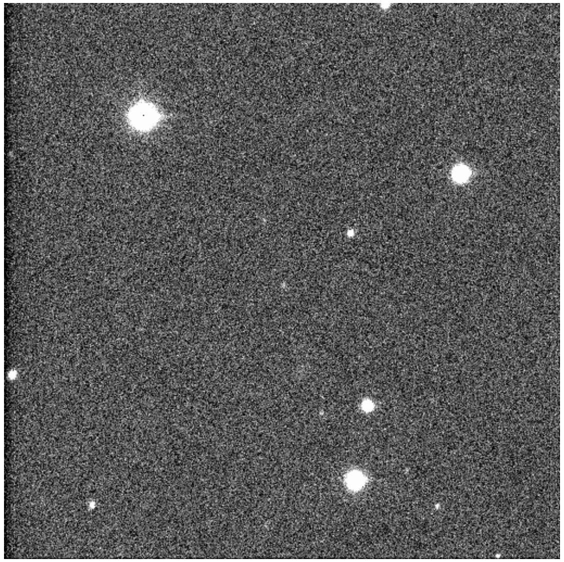
(where λ is wavelength, D the diameter of the telescope and θ is in radians)

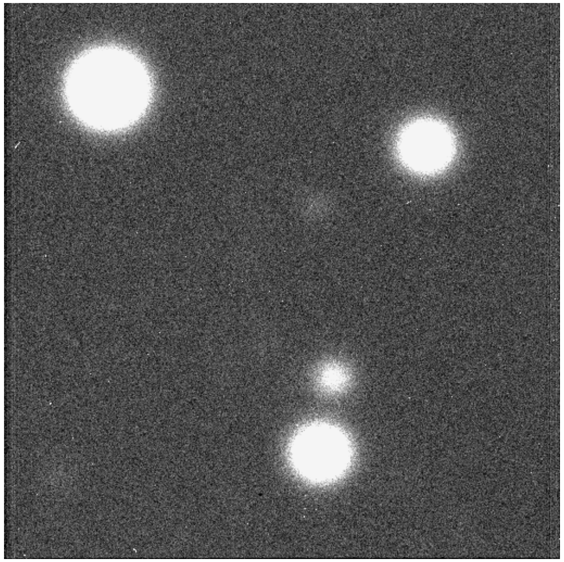


Atmospheric turbulence broadens the PSF resulting in a Gaussian PSF

$$I(r) = I(0)\exp(-r^2/2\sigma^2)$$







Adaptive Optics (AO) imaging

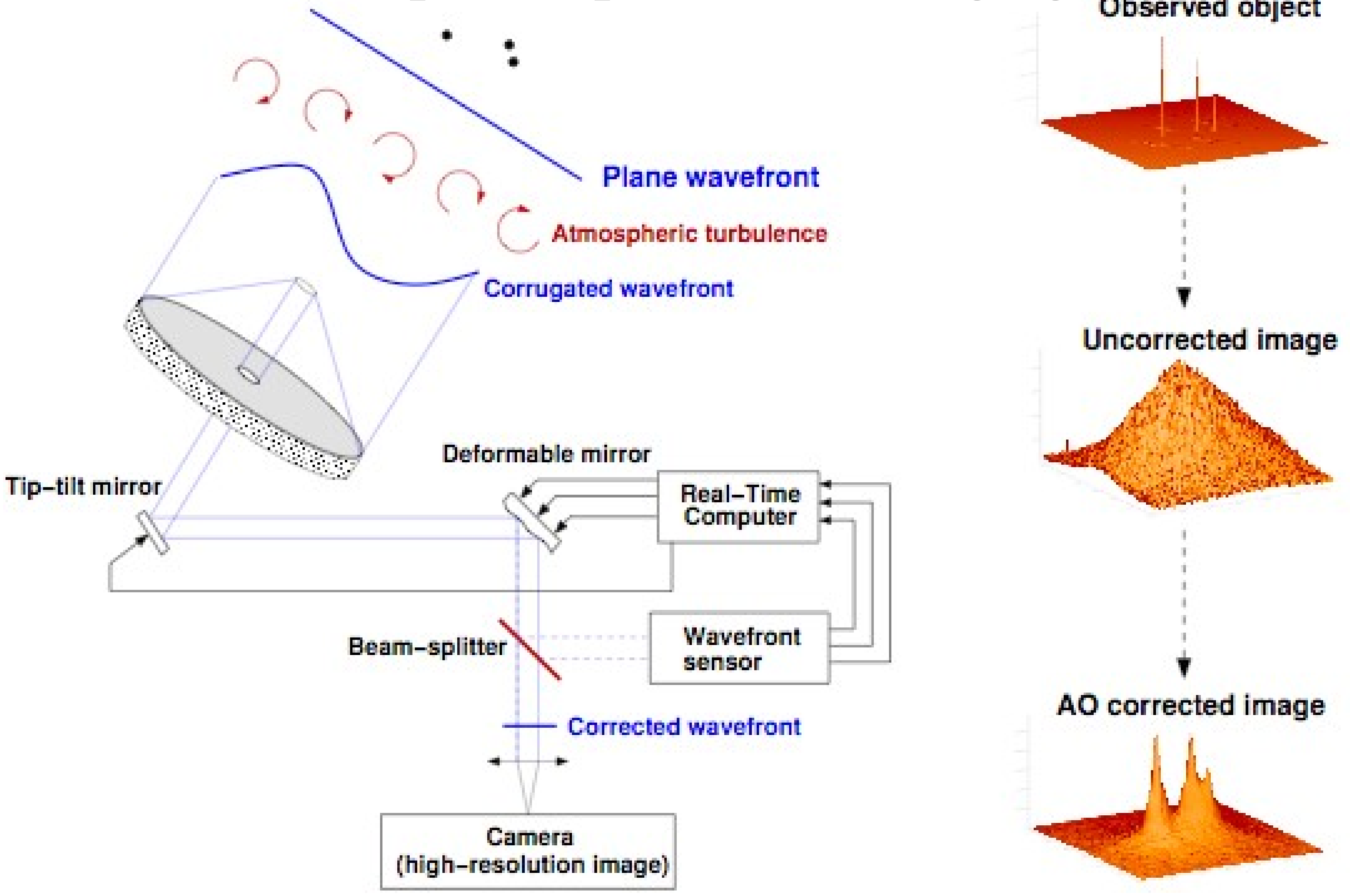
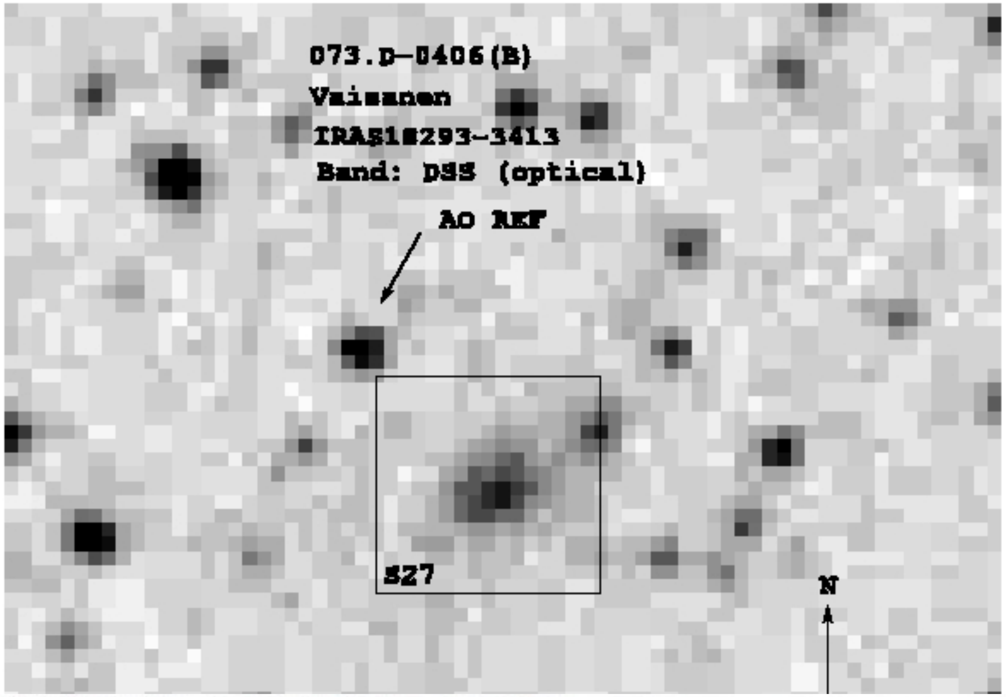
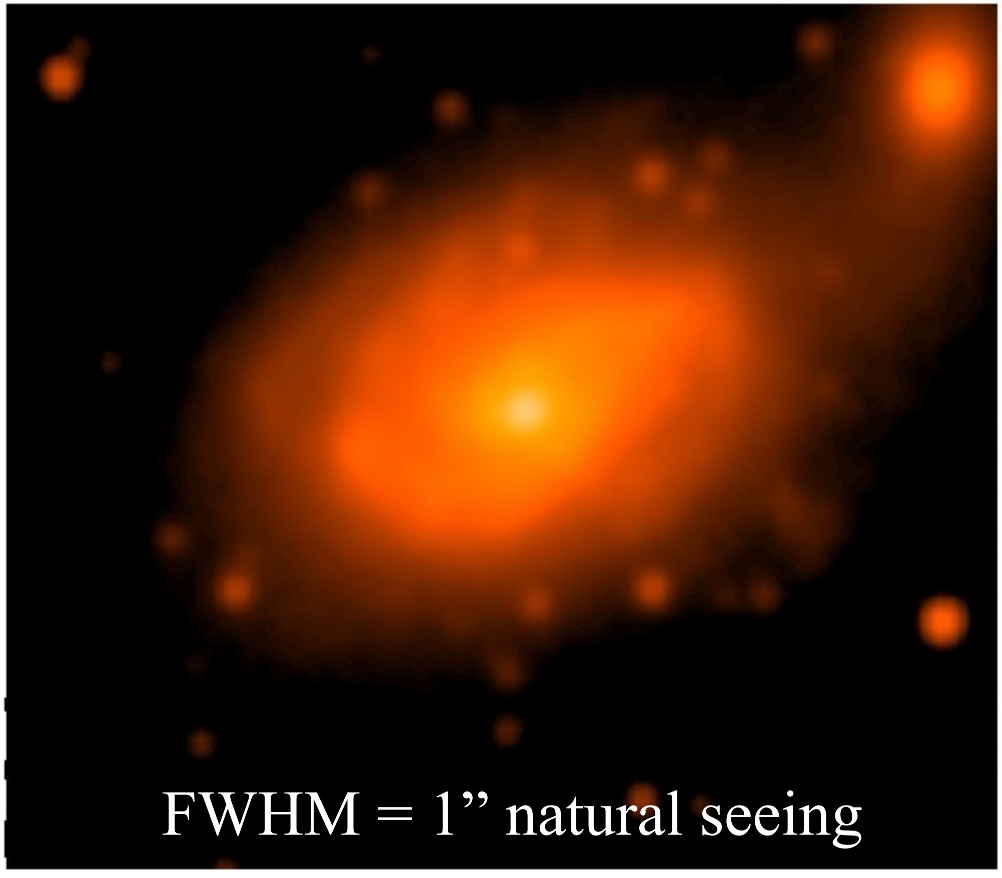
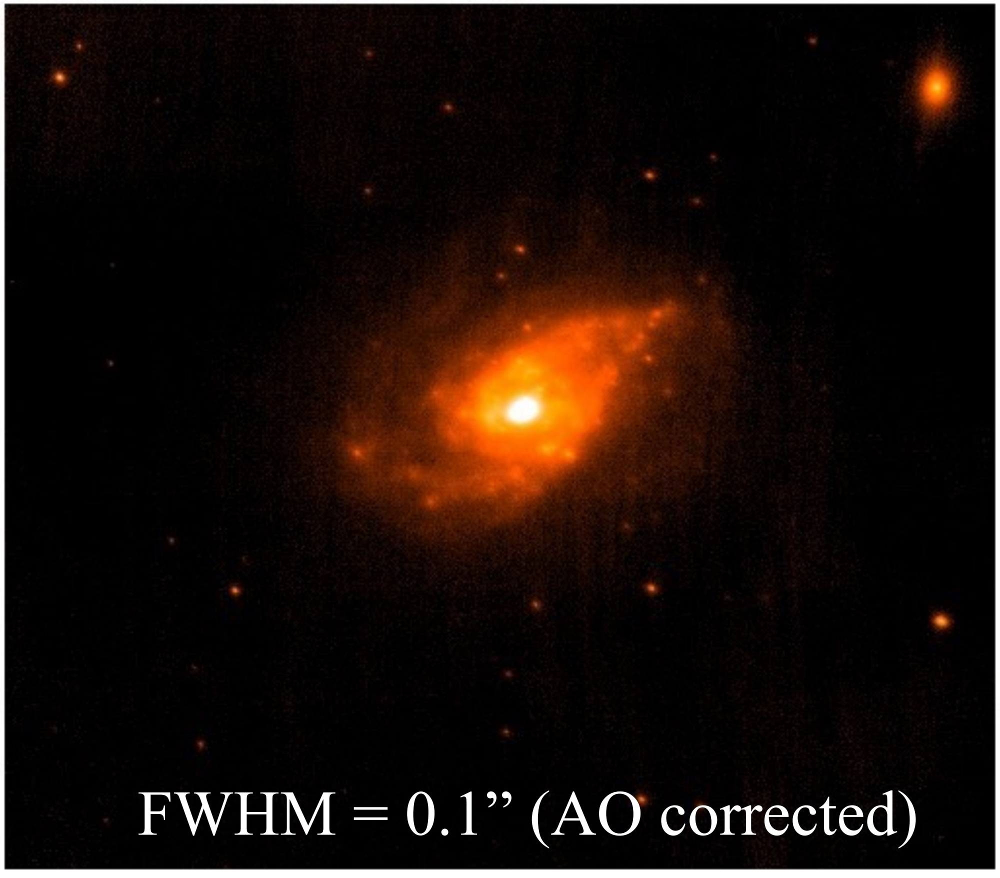


Figure 3-1: Principle of Adaptive Optics

VLT/NACO User manual


073.D-0406(B)
Vaisanen
IRAS18293-3413
Band: DSS (optical)
AO REF
S27
N


FWHM = 1” natural seeing

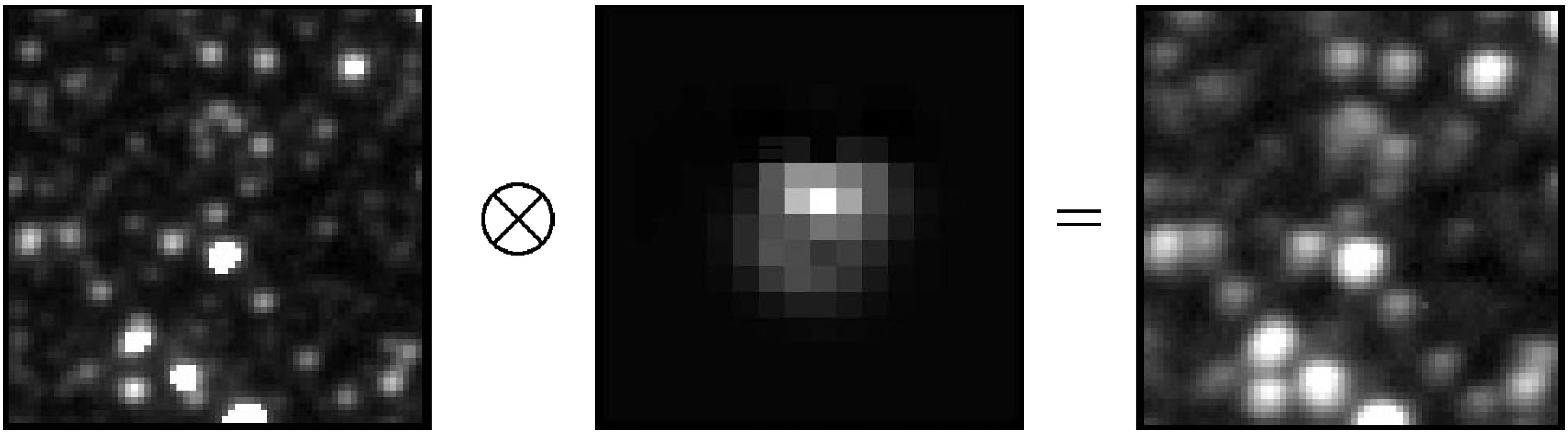

FWHM = 0.1” (AO corrected)

Convolution of images

“real” signal **additive noise**

$$b(\vec{x}) = f(\vec{x}) * p(\vec{x}) + n(\vec{x})$$

observed signal **PSF**



$$ref(x,y) \otimes kernel(x,y,u,v) = im(x,y)$$

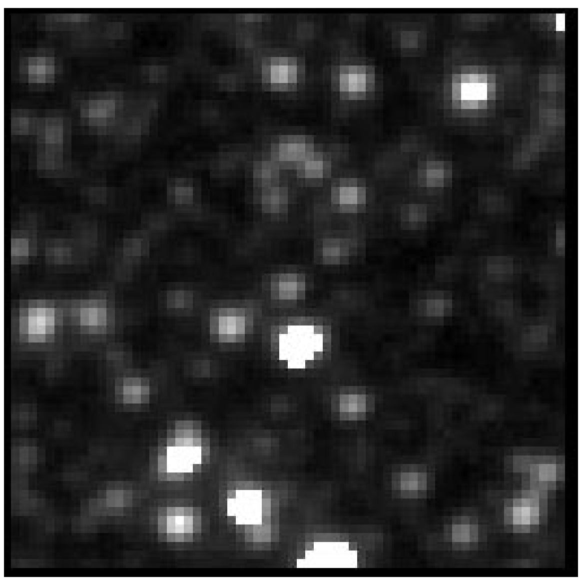
Aland & Lupton 1998: A method for optimal image subtraction, arXiv:astro-ph/9712287

Optimal Image Subtraction

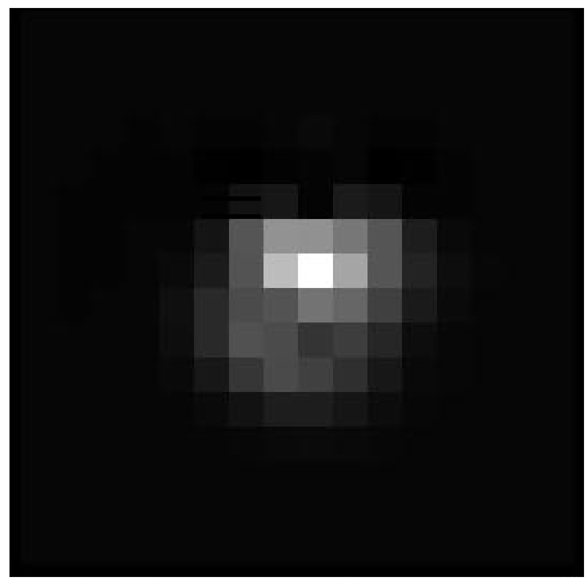
$$ref(x,y) \otimes kernel(x,y,u,v) = im(x,y) + bg(x,y)$$

$$kernel(x,y,u,v) = \sum_{n} \sum_{d_n^x} \sum_{d_n^y} \sum_{\delta^x} \sum_{\delta^y} [a_n \underbrace{x^{\delta^x} y^{\delta^y}}_{3} \underbrace{e^{-(u^2+v^2)/2\sigma_n^2}}_{1} \underbrace{u^{d_n^x} v^{d_n^y}}_{2}]$$

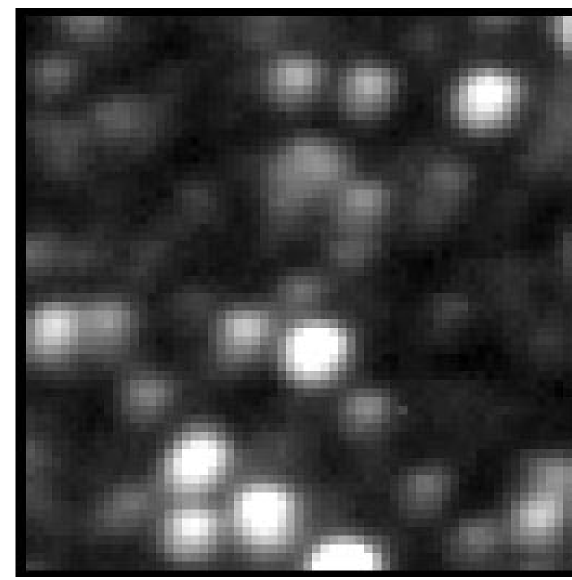
The convolution kernel consists of a set of Gaussian functions (1) which are modified by polynomials (2) and a model for the spatial variations of the kernel (3) where $0 < d_n^y + d_n^x \leq D_n$, and $0 < \delta^y + \delta^x \leq D^k$.



$\otimes$



$=$



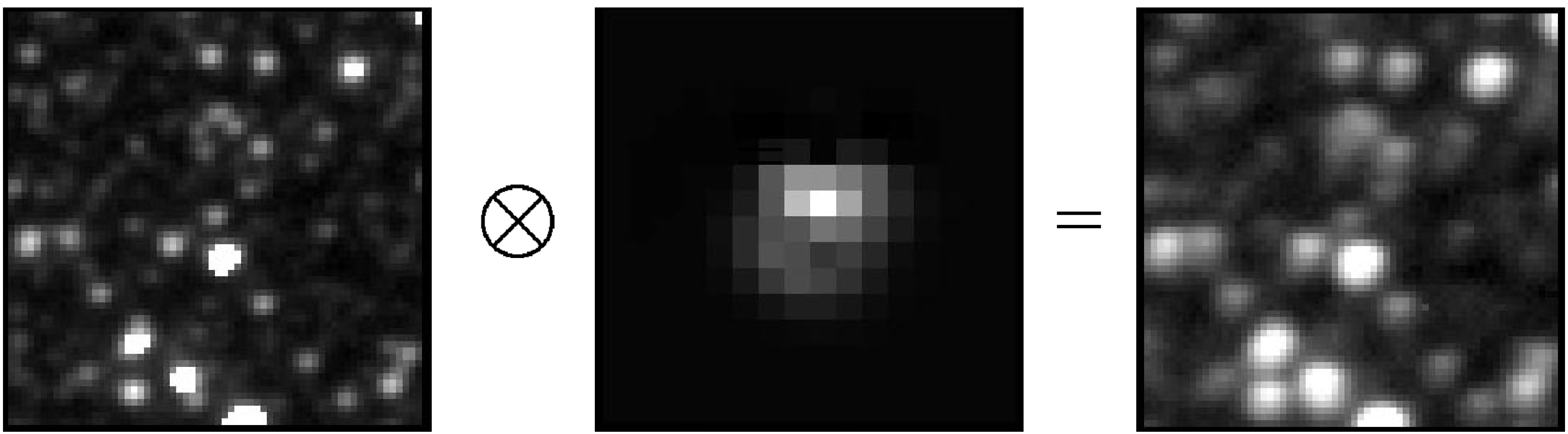
Aland & Lupton 1998: A method for optimal image subtraction, arXiv:astro-ph/9712287

Optimal Image Subtraction

$$ref(x, y) \otimes kernel(x, y, u, v) = im(x, y) + bg(x, y)$$

$$kernel(x, y, u, v) = \sum_{n} \sum_{d_n^x} \sum_{d_n^y} \sum_{\delta^x} \sum_{\delta^y} [a_n \underbrace{x^{\delta^x} y^{\delta^y}}_{3} \underbrace{e^{-(u^2+v^2)/2\sigma_n^2}}_{1} \underbrace{u^{d_n^x} v^{d_n^y}}_{2}]$$

$$bg(x, y) = \sum_{i} \sum_{j} a_i x^i y^j$$



Aland & Lupton 1998: A method for optimal image subtraction, arXiv:astro-ph/9712287

Convolution: Optimal Image Subtraction

$$kernel(x,y,u,v) = \sum_{n}\sum_{d_n^x}\sum_{d_n^y}\sum_{\delta^x}\sum_{\delta^y}[a_n \underbrace{x^{\delta^x} y^{\delta^y}}_{3} \underbrace{e^{-(u^2+v^2)/2\sigma_n^2}}_{1} \underbrace{u^{d_n^x} v^{d_n^y}}_{2}]$$

$$bg(x,y) = \sum_{i}\sum_{j} a_i x^i y^j$$

n	number of Gaussian functions in the kernel
σ_n	sigmas of the Gaussians
D_n	polynomial degrees associated with each of the n gaussians
D^k	degree of the polynomial transform for the spatial variations of the kernel
D^{bg}	degree of the polynomial used to model the background variations
N_x	number of stamps along x-axis
N_y	number of stamps along y-axis
S_k	width of the convolution kernel
S_s	width of the region used for fitting the background
N_c	minimum number of counts in the middle of a stamp
N_{min}	minimum value of a pixel to be included in the fit
N_{sat}	maximum value of a pixel to be included in the fit

Aland & Lupton 1998: A method for optimal image subtraction, arXiv:astro-ph/9712287

Exercise 2: practical work

Experiment with matching and subtraction of AO images from two different dates present your results in the report

Can you spot the supernova? Maybe, image subtraction would help ...

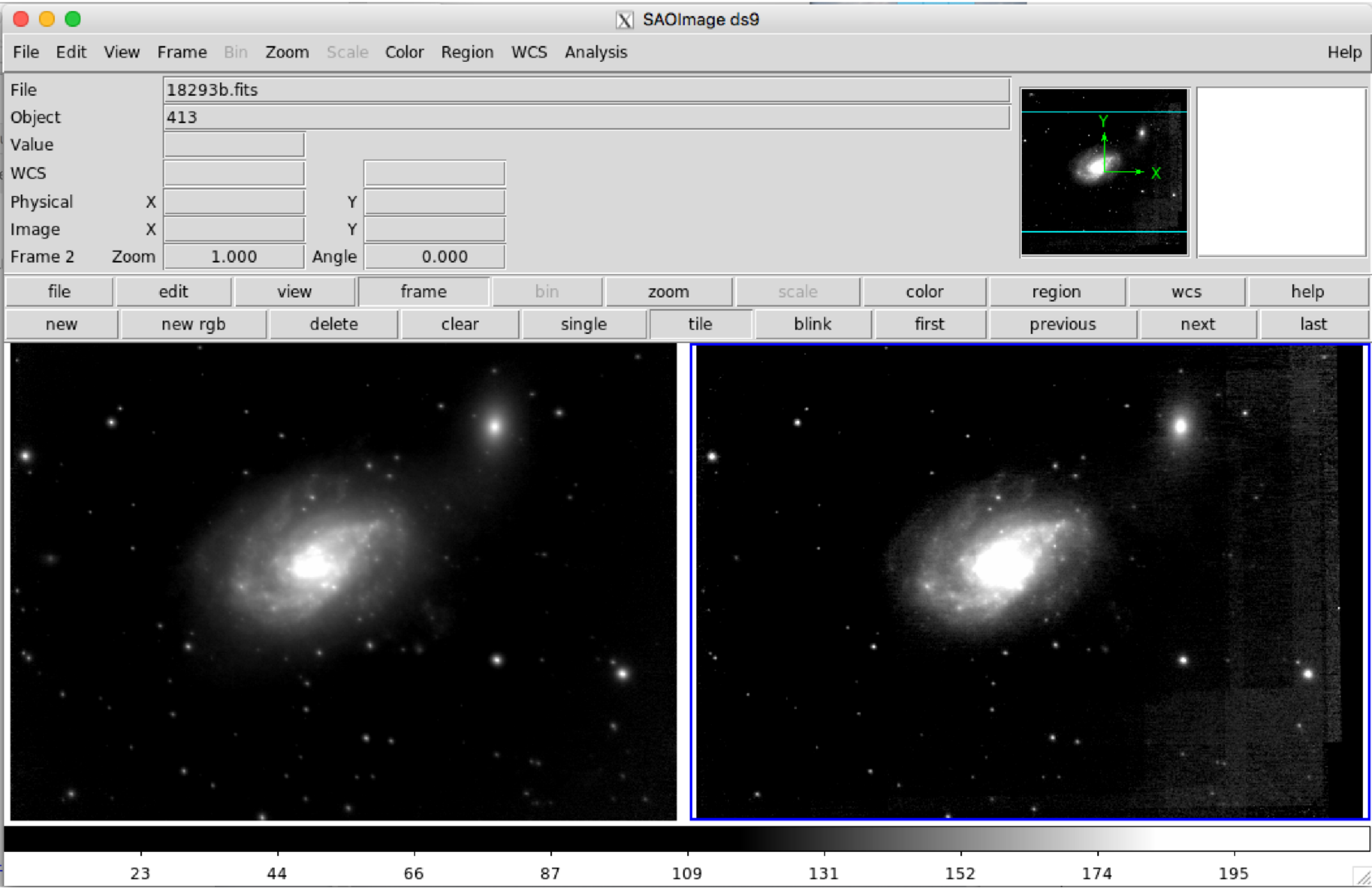


image matching and subtraction of astronomical images

not3@course2021: ~/imagesubtraction

not3@course2021: ~/imagesubtraction 80x7

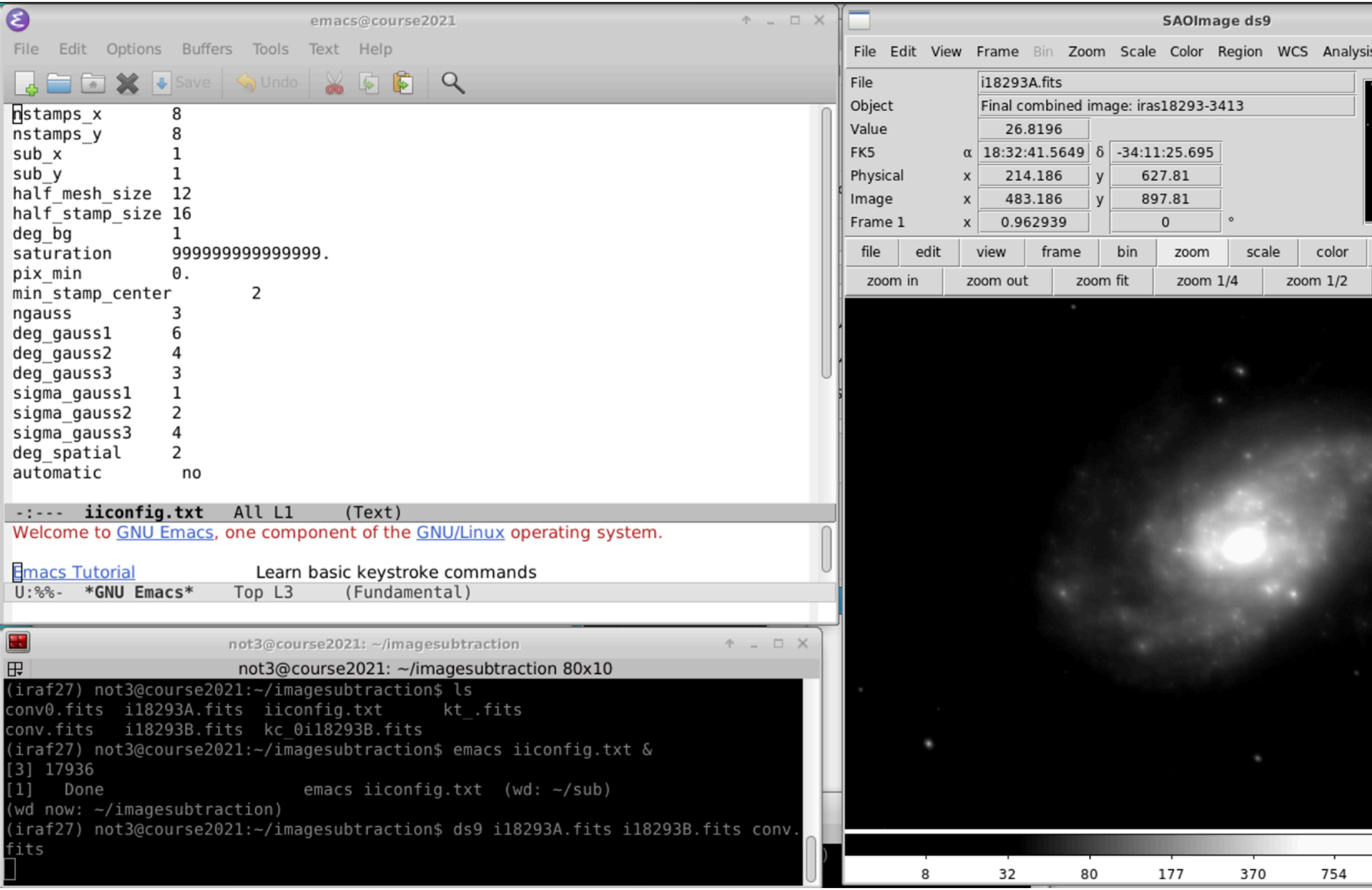
```
(iraf27) not3@course2021:~$ mkdir imagesubtraction
(iraf27) not3@course2021:~$ cd imagesubtraction
(iraf27) not3@course2021:~/imagesubtraction$ cp /home/not3/sub/* .
(iraf27) not3@course2021:~/imagesubtraction$ ls
i18293A.fits  i18293B.fits  iiconfig.txt
(iraf27) not3@course2021:~/imagesubtraction$ mrj_phot i18293A.fits i18293B.fits
-c iiconfig.txt
```

not3@course2021: ~/imagesubtraction

not3@course2021: ~/imagesubtraction 80x56

```
Reading Image1
name: i18293A.fits
width: 1562 height: 1562 bitpix: -32 offset_header: 46080
name: i18293B.fits
width: 1562 height: 1562 bitpix: -32 offset_header: 46080
offsets: 46080 46080
ss: 1 1 1562 1562
ss: 1562 1562 1562 1562
nsx: 8 nsy: 8
mesh_size: 25 stam_size: 33
ng: 3 6
sg: 0.500000
sat: 999999999999999.000000 pix_min: 0.000000 max_stamp 2
deg_spatial: 2 deg_bg: 1
subs: 1562 1562
Allocate: 64
Getting Stamps
bitpix: -32 4
width: 1562 height: 1562 1562 1562
bitpix: -32 4
width: 1562 height: 1562 1562 1562
sub_ref: 32.391235 10.000000
Making defect map
Looking for Stamps
nb: 64
Building Zero Order Matrix
Checking Stamps
x: 71 y: 125 -1.375231
```

image matching and subtraction of astronomical images



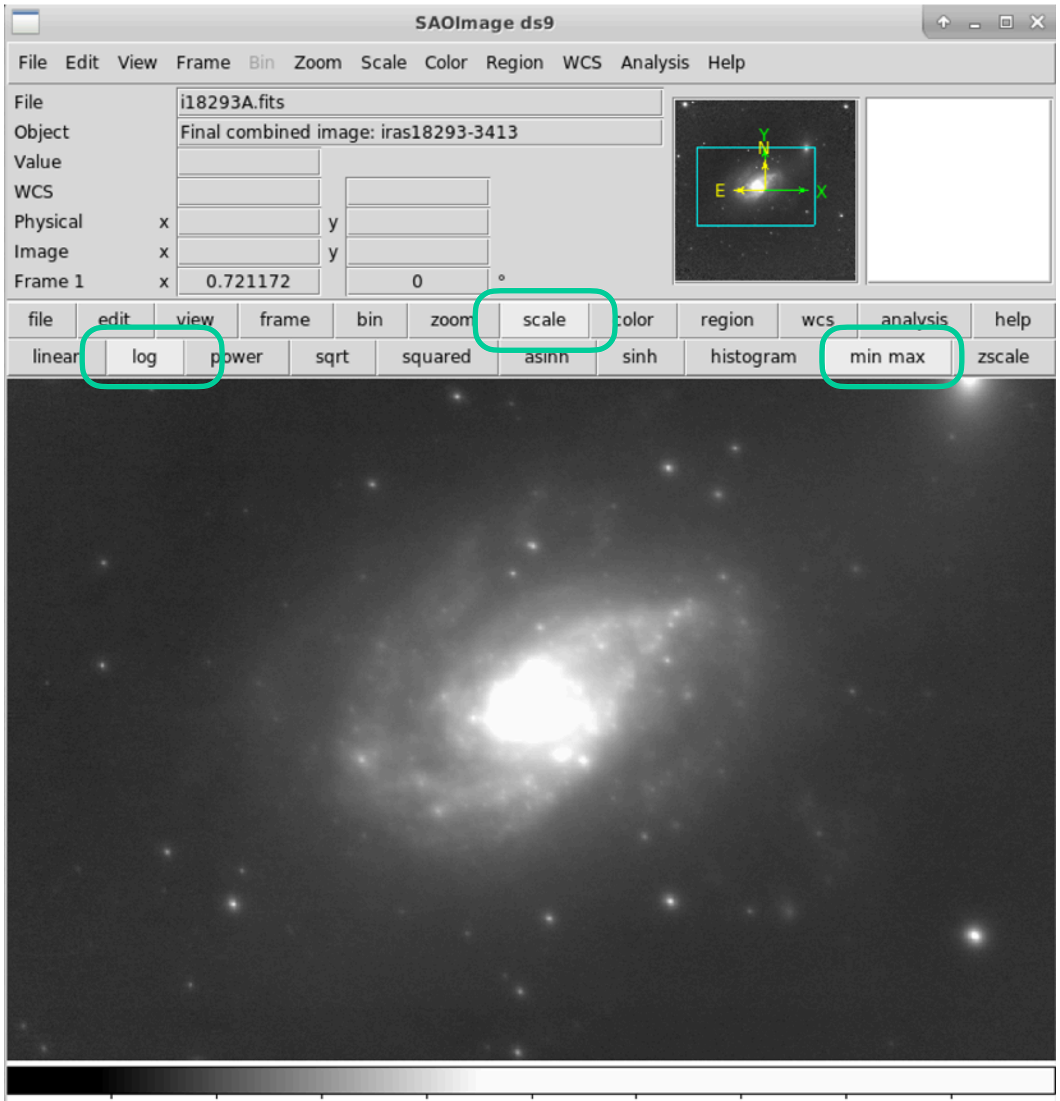
extract relevant information from the fits headers incl. dates of the observations, telescope and instrument used, wavelength of the observation, atmospheric conditions?

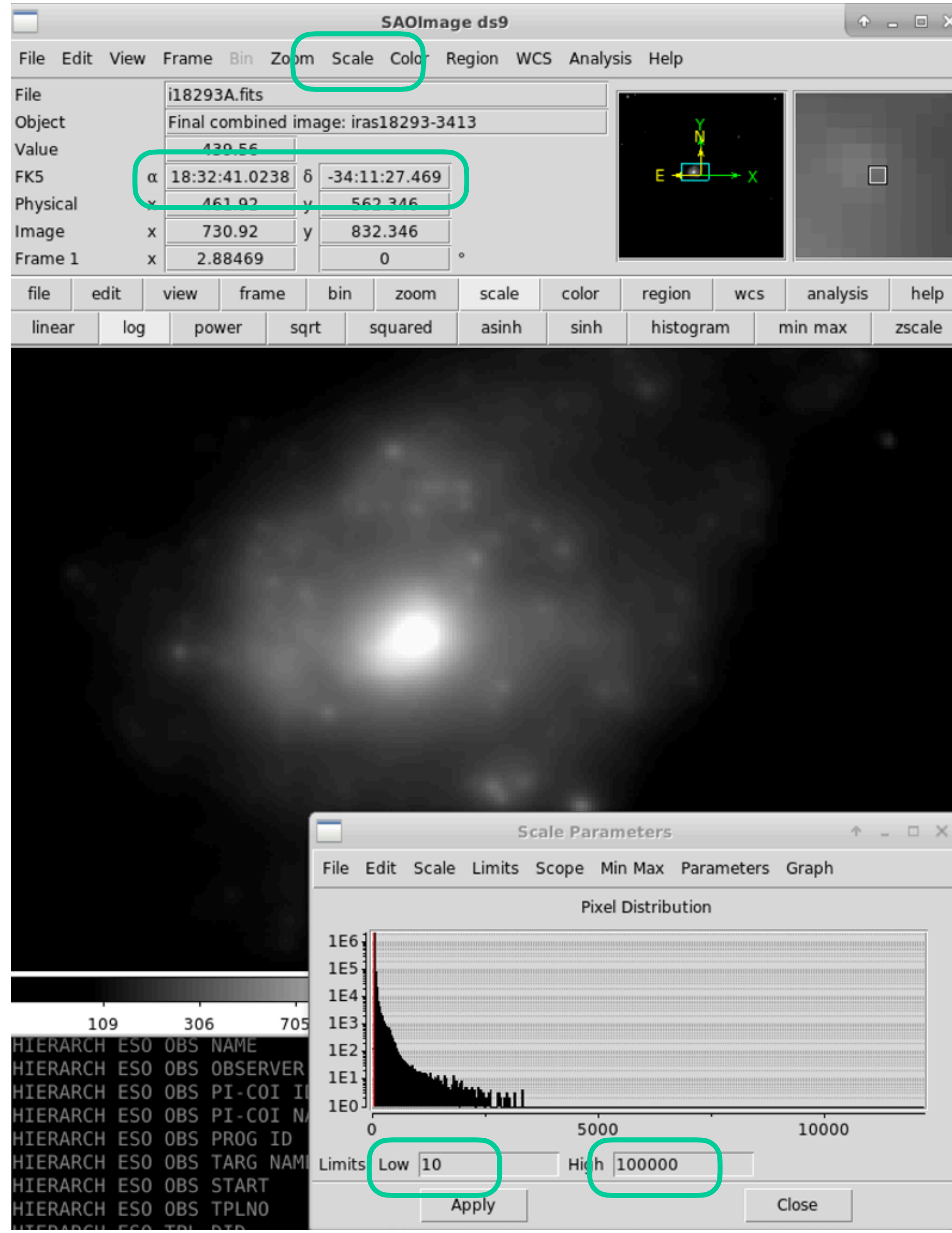
not3@course2021: ~/imagesubtraction

not3@course2021: ~/imagesubtraction 80x53

```
(iraf27) not3@course2021:~/imagesubtraction$ more i18293A.fits
SIMPLE  =                    T / Fits standard
BITPIX  =                  -32 / Bits per pixel
NAXIS   =                    2 / Number of axes
NAXIS1  =                 1562 / Axis length
NAXIS2  =                 1562 / Axis length
EXTEND  =                    T / File may contain extensions
IRAF-TLM= '2022-05-30T15:10:15' / Time of last modification
OBJECT  = 'Final combined image: iras18293-3413'
DATE    = '2022-05-30T15:10:15'
IRAF-MAX=           0.000000E0  /  DATA MAX
IRAF-MIN=           0.000000E0  /  DATA MIN
ORIGIN  = 'NOAO-IRAF FITS Image Kernel July 2003' / FITS file originator
DATE    = '2005-08-03T11:53:46' / Date this file was written
EXPTIME =                    1 / Total integration time
MJD-OBS =       53262.98191961          / MJD start (2004-09-14T23:33:57.854)
DATE-OBS= '2004-09-14T23:33:57.8545'    / Observing date
ORIGFILE= 'NACO_0016.fits'              / Original File Name
INSTRUME= 'NAOS+CONICA'                 / Instrument used
TELESCOP= 'ESO-VLT-U4'                  / ESO Telescope Name
RA      =           278.170930          / 18:32:41.0 RA (J2000) pointing
DEC     =            -34.19111          / -34:11:27.9 DEC (J2000) pointing
EQUINOX =                2000.          / Standard FK5
RADECSYS= 'FK5     '                    / FK5
LST     =            66541.046          / 18:29:01.046 LST
UTC     =            84812.000          / 23:33:32.000 UTC
OBSERVER= 'UNKNOWN '                    / Name of observer
PI-COI  = 'UNKNOWN '                    / Name(s) of proposer(s)
ALARM   = '       '                     / Active alarm(s), if any.
AIRMASS =              1.01400          / Averaged air mass
CRVAL1  =            278.17093          / Coordinate at reference pixel in <axis
CRVAL2  =            -34.19111          / Coordinate at reference pixel in <axis
CRPIX1  =                731.2 / Ref pixel in <axis direction>
CRPIX2  =                812.9 / Ref pixel in <axis direction>
CDELT1  =  -7.5277800000000E-6 / Increment in <axis direction>
CDELT2  =   7.5277800000000E-6 / Increment in <axis direction>
CTYPE1  = 'RA---TAN'                    / Coordinate system of <axis direction>
CTYPE2  = 'DEC--TAN'                    / Coordinate system of <axis direction>
CD1_1   =          -7.52778E-06         / Translation matrix element
CD2_2   =           7.52778E-06         / Translation matrix element
ARCFILE = 'NACO.2004-09-14T23:33:57.854.fits' / Archive File Name
UT      = '23:33:32.000'                / UT at start
ST      = '18:29:01.046'                / ST at start
IMAGETYP= 'OBJECT  '                    / Observation type
HIERARCH ESO OBS DID          = 'ESO-VLT-DIC.OBS-1.10' / OBS Dictionary
HIERARCH ESO OBS EXECTIME     =           3620 / Expected execution time
HIERARCH ESO OBS GRP          = '0       '  / linked blocks
HIERARCH ESO OBS ID           =      200141584 / Observation block ID
HIERARCH ESO OBS NAME         = 'TOO_I18293-3413Ks' / OB name
HIERARCH ESO OBS OBSERVER     = 'UNKNOWN '  / Observer Name
HIERARCH ESO OBS PI-COI ID    =           5889 / ESO internal PI-COI ID
HIERARCH ESO OBS PI-COI NAME  = 'UNKNOWN '  / PI-COI name
HIERARCH ESO OBS PROG ID      = '073.D-0406(B)' / ESO program identification
HIERARCH ESO OBS TARG NAME    = 'iras18293-3413' / OB target name
```

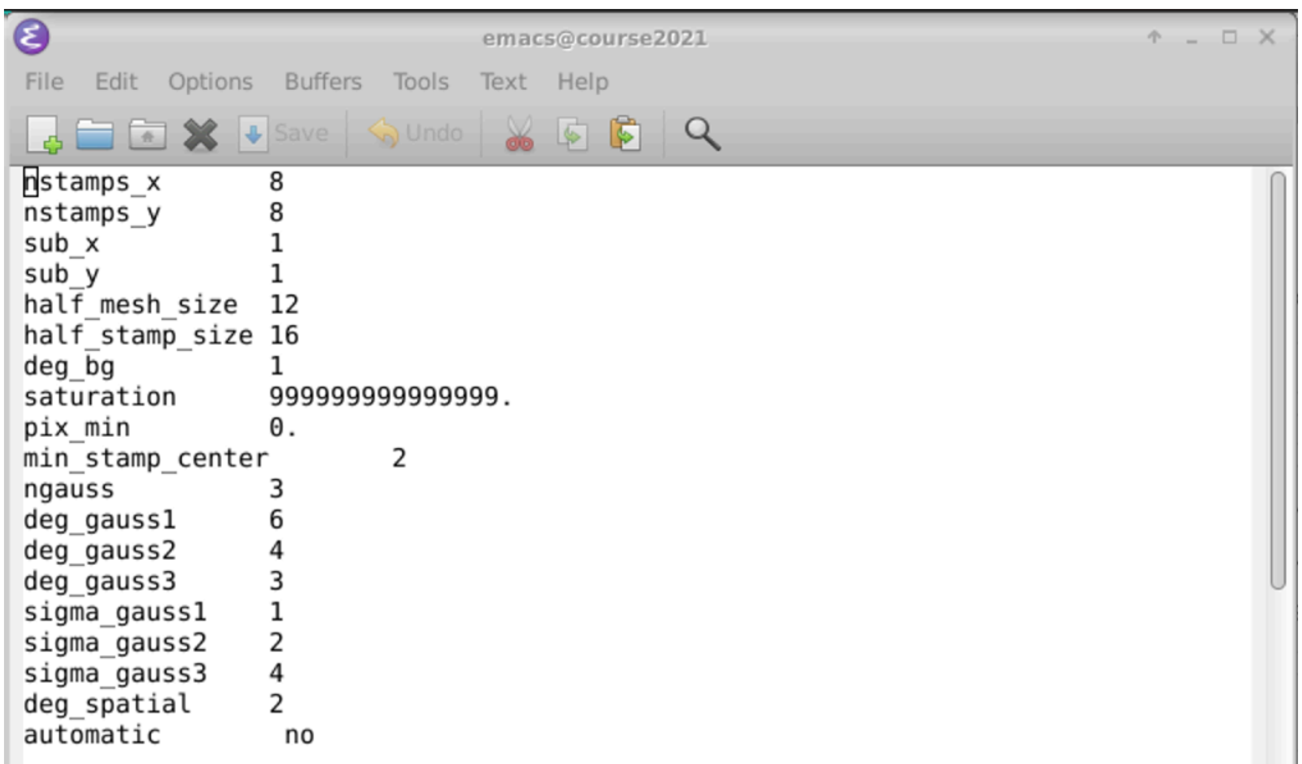
inspect the images using the ds9 tool: different contrast settings, comparison between different images, identifying the subtraction residuals with image sources, astronomical coordinates, estimate the spatial resolution of the images



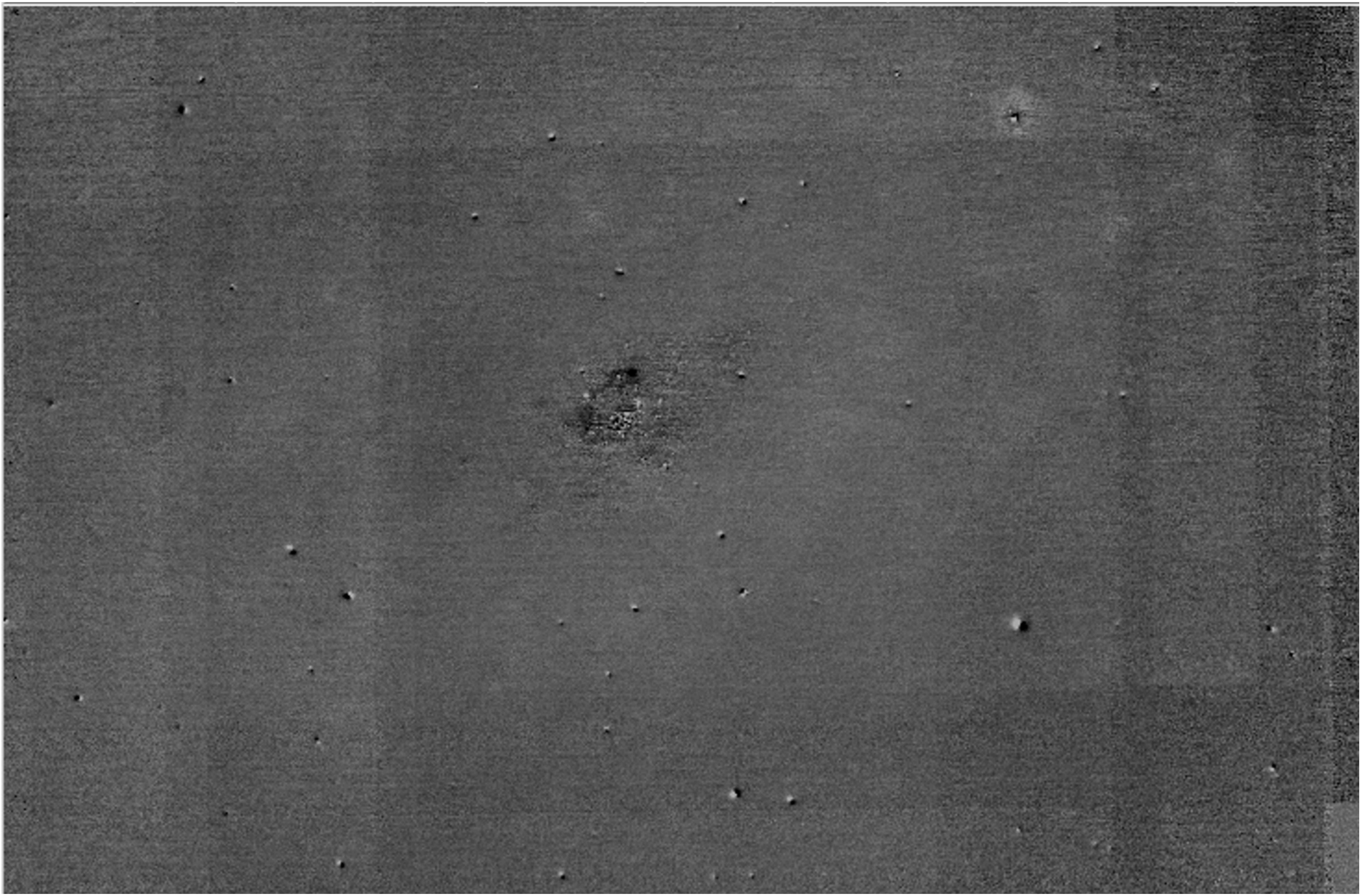


experiment what happens if we change the subtraction parameters?

n	number of Gaussian functions in the kernel
σ_n	sigmas of the Gaussians
D_n	polynomial degrees associated with each of the n gaussians
D^k	degree of the polynomial transform for the spatial variations of the kernel
D^{bg}	degree of the polynomial used to model the background variations
N_x	number of stamps along x-axis
N_y	number of stamps along y-axis
S_k	width of the convolution kernel
S_s	width of the region used for fitting the background
N_c	minimum number of counts in the middle of a stamp
N_{min}	minimum value of a pixel to be included in the fit
N_{sat}	maximum value of a pixel to be included in the fit



how does the subtracted image look like? what is causing the different residuals? any real variability between the two images?



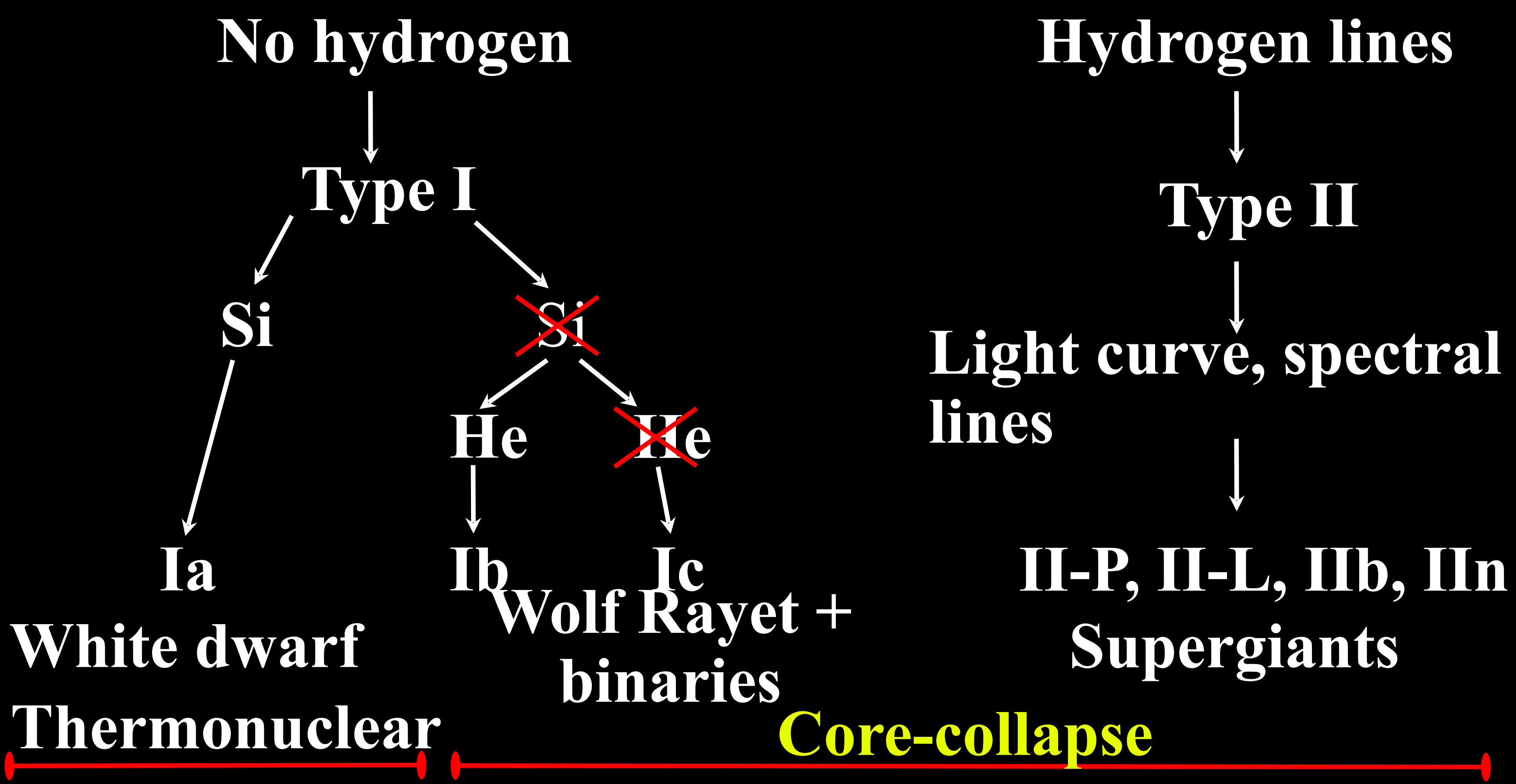
Exercise 3: Deep learning for the automated spectral classification of supernovae

https://github.com/daniel-muthukrishna/DASH

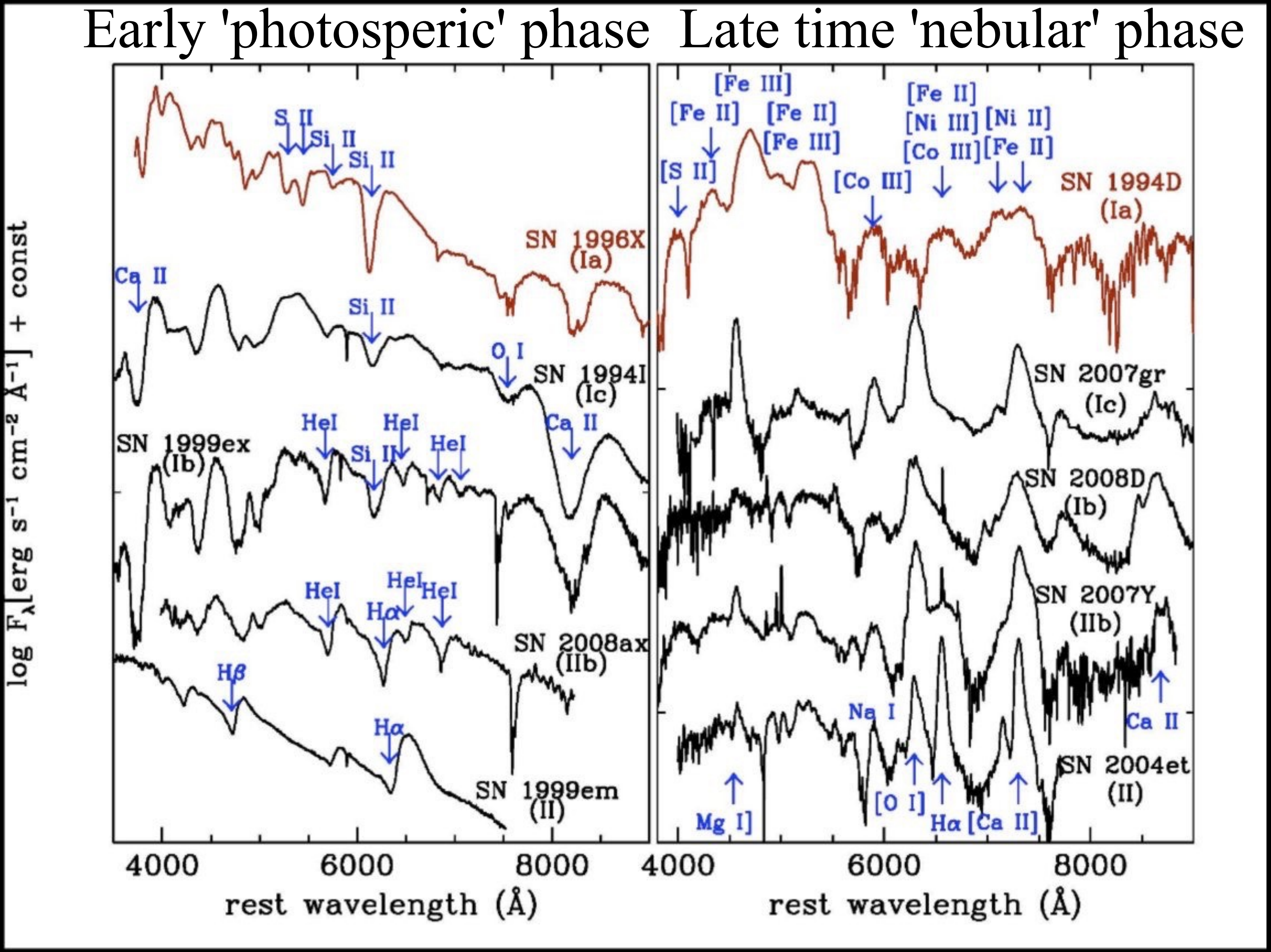
Muthukrishna et al. 2019, ApJ, 885, 18

https://iopscience.iop.org/article/10.3847/1538-4357/ab48f4/pdf

Supernova types

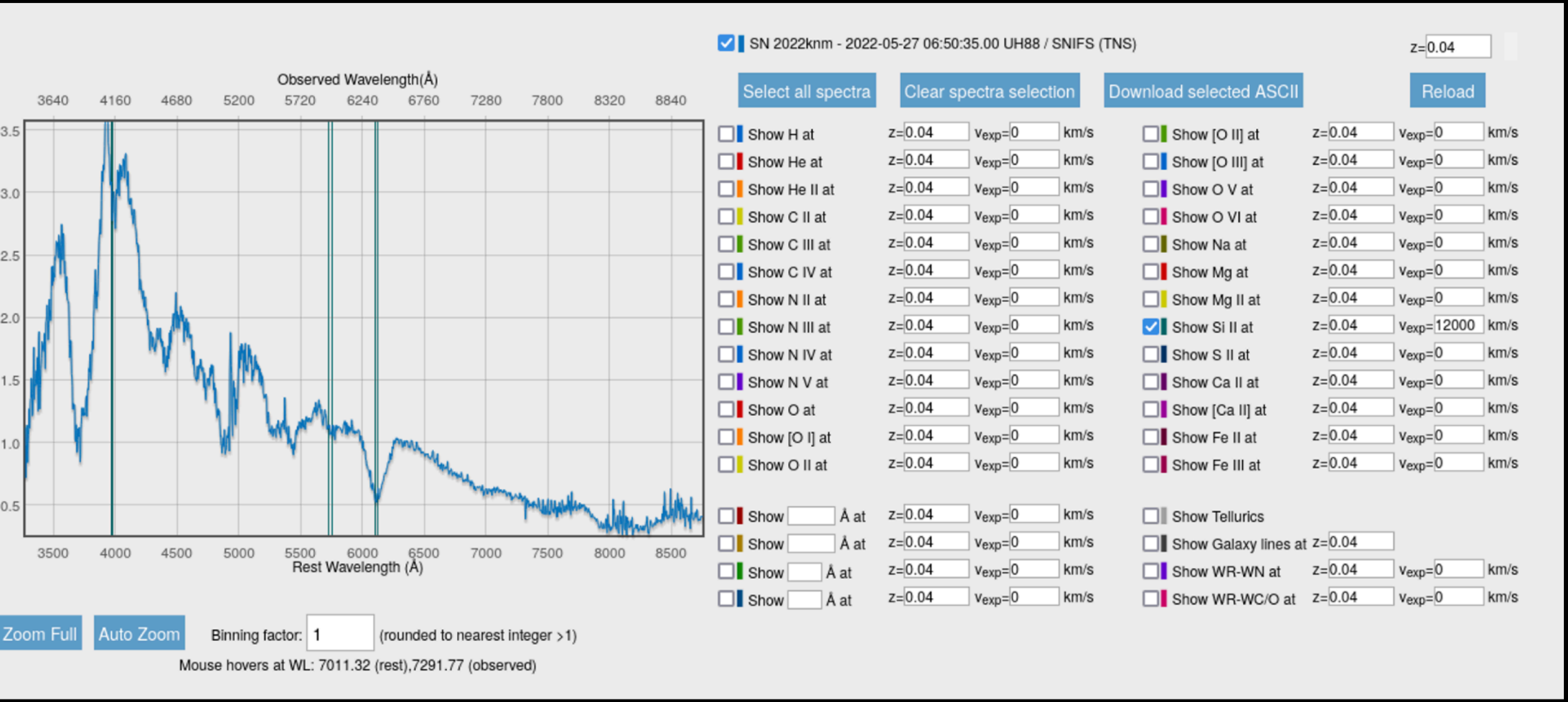


Supernova types

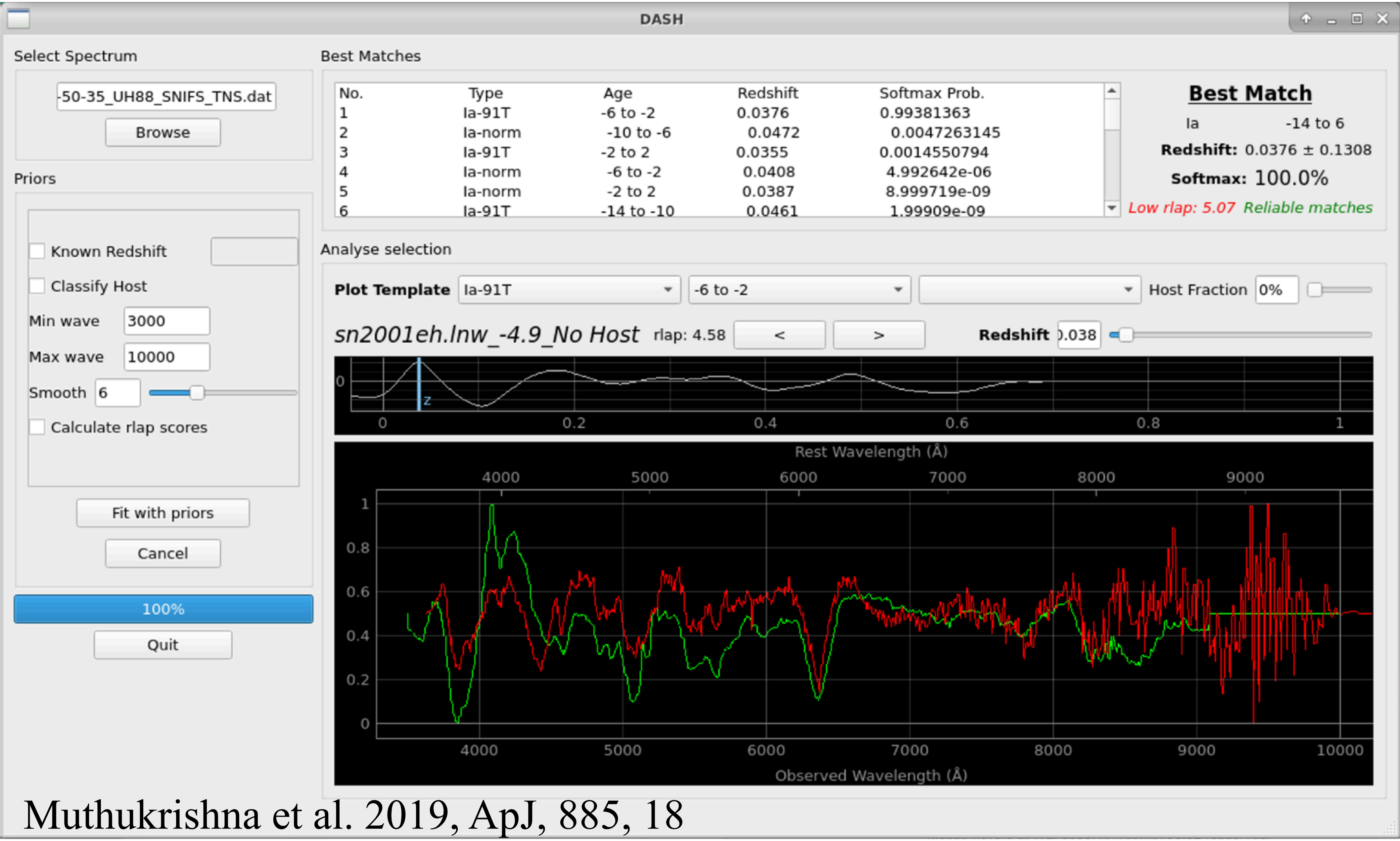


Pastorello+ 2007

Supernova spectral classification by identifying characteristic spectral lines (and elements)

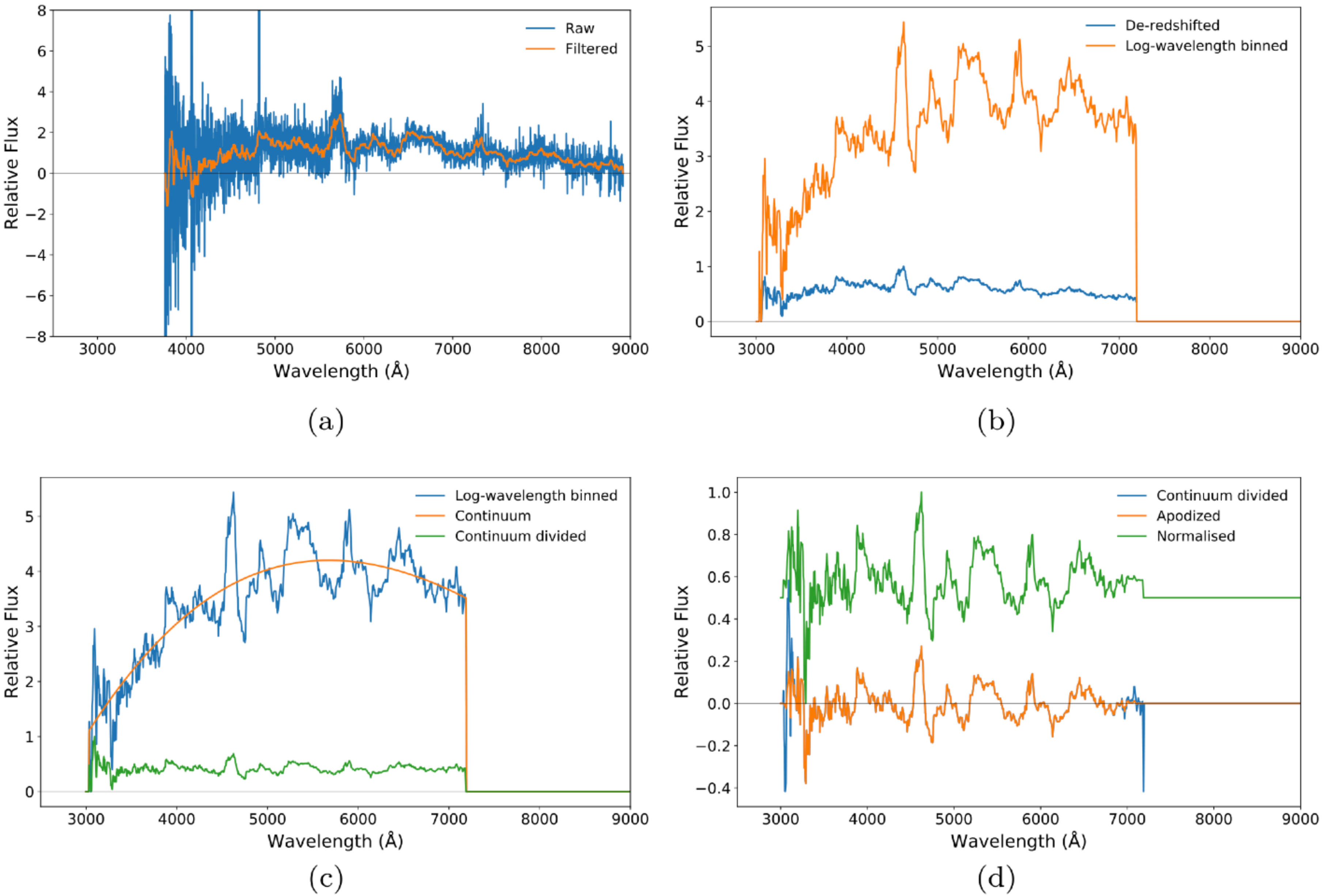


Automatic supernova classification based on a deep learning approach



Muthukrishna et al. 2019, ApJ, 885, 18

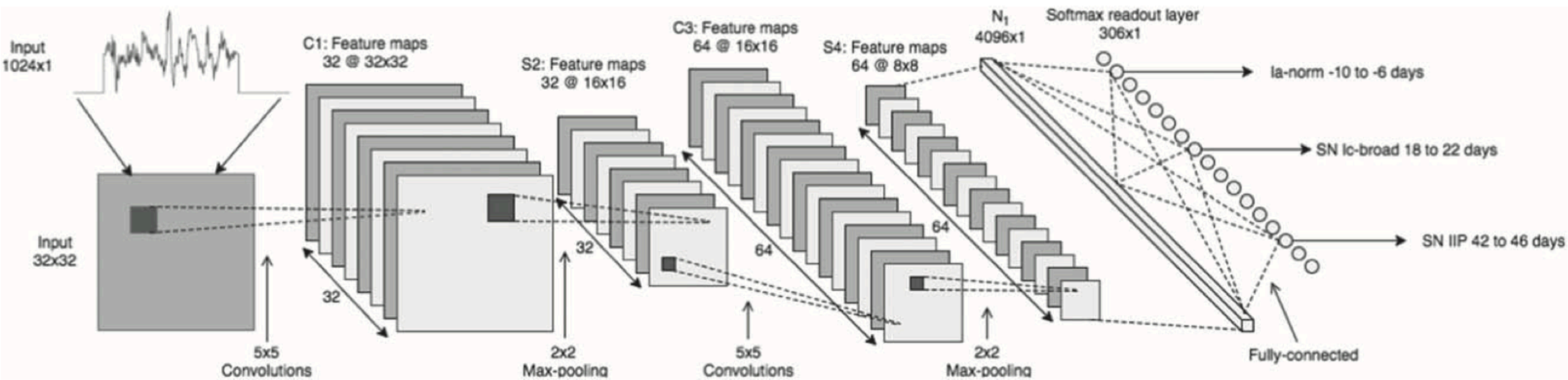
Automatic supernova classification based on a deep learning approach



Muthukrishna et al. 2019, ApJ, 885, 18

Automatic supernova classification based on a deep learning approach

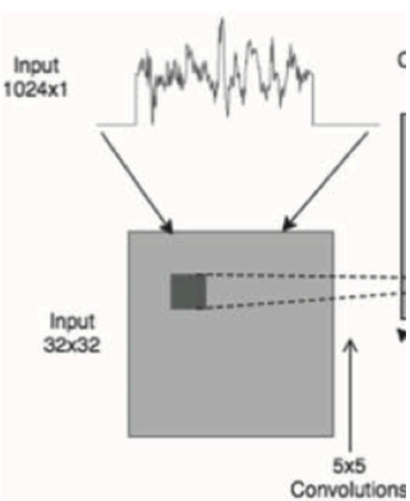
- Developed a convolutional neural network with TensorFlow's Python library
- 1st input layer is made up of 1024 neurons representing fluxes of an input spectrum
- Additional layers captures a more abstract representation of the original input layer
  - DASH classifies based on spectral features (different SN types & ages)
- Final layer is a 306-point vector, with a score for each SN type and age bin
  - Training + validation using different datasets

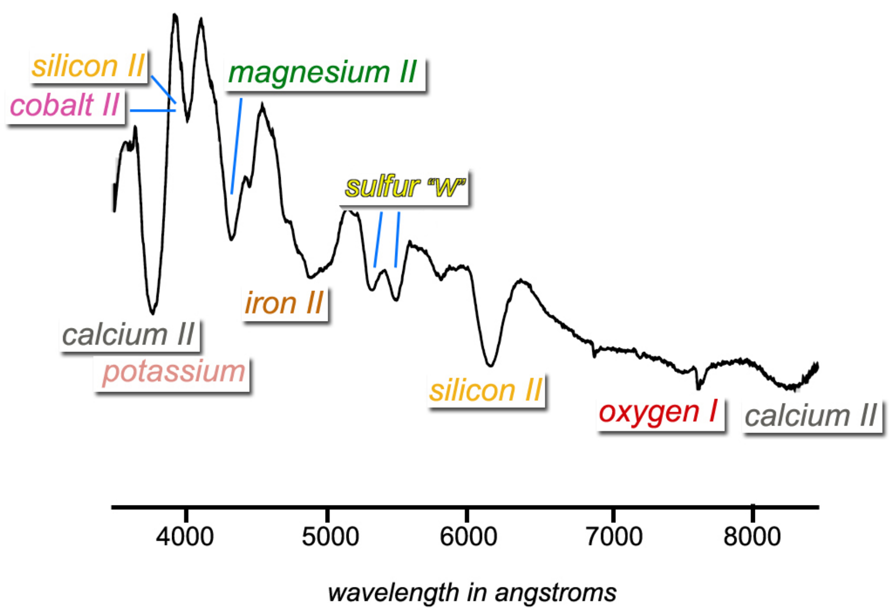


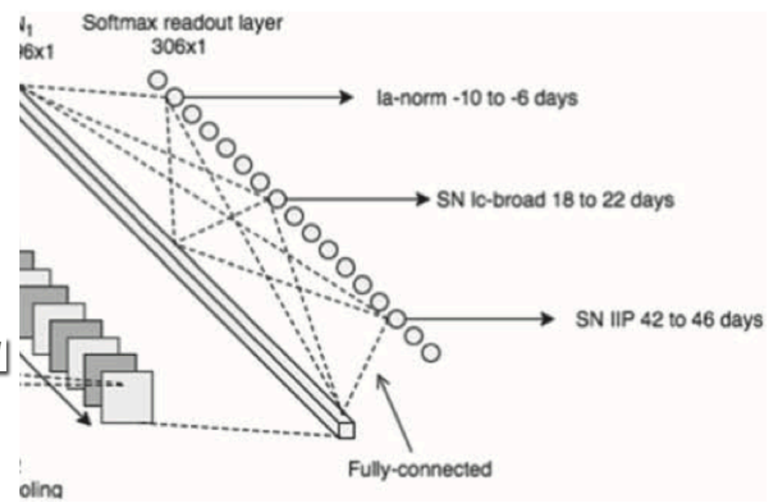
Muthukrishna et al. 2019, ApJ, 885, 18

Automatic supernova classification based on a deep learning approach

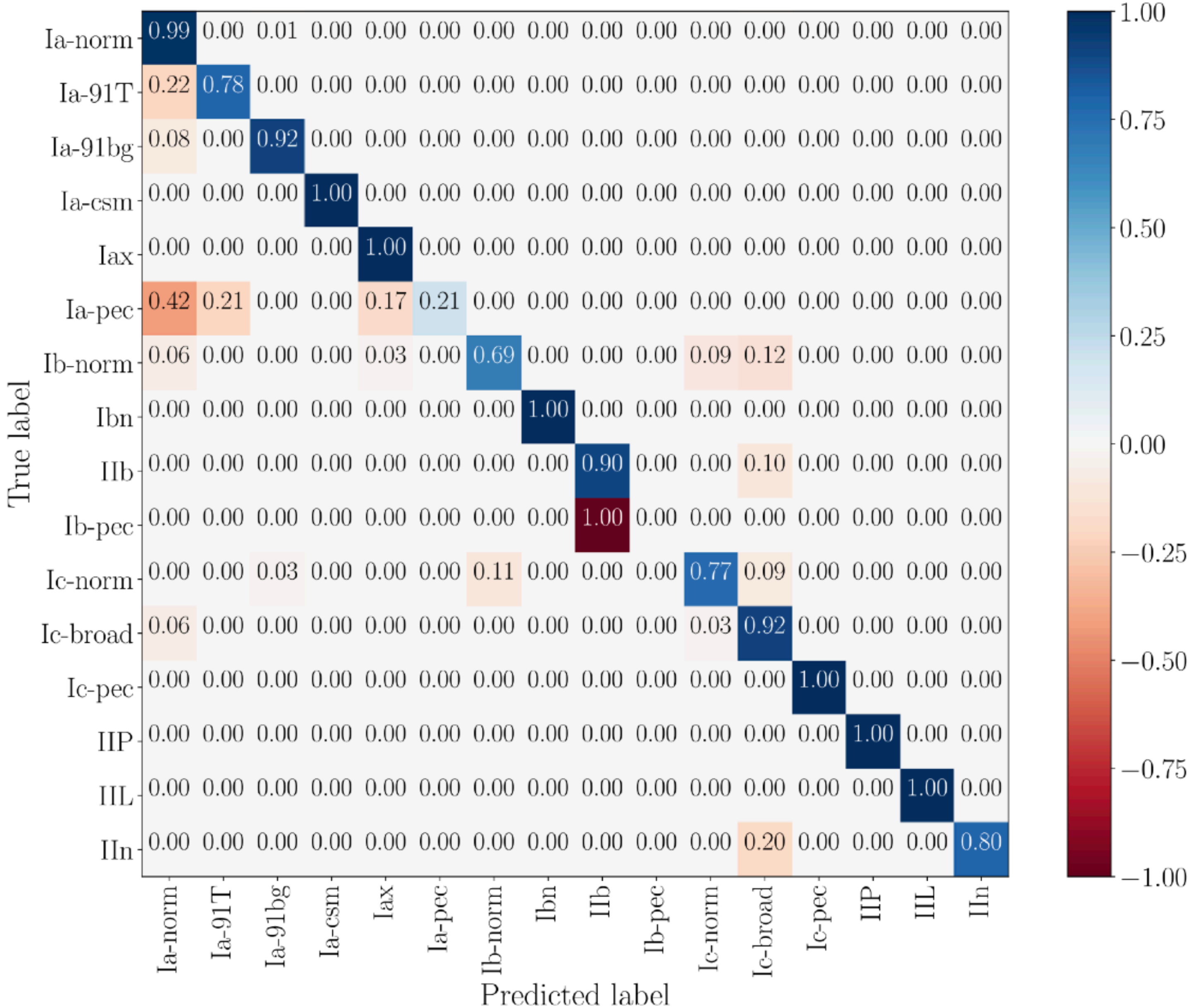
- Developed a convolutional neural network with TensorFlow's Python library
- 1st input layer is made up of 1024 neurons representing fluxes of an input spectrum
- Additional layers captures a more abstract representation of the original input layer
    - DASH classifies based on spectral features (different SN types & ages)
- Final layer is a 306-point vector, with a score for each SN type and age bin
    - Training + validation using different datasets







Automatic supernova classification based on a deep learning approach



Muthukrishna et al. 2019, ApJ, 885, 18

Exercise 3: practical work

Experiment with a deep learning based method for supernova classification and present your findings in the report; discuss the pros and cons of the use of machine learning in supernova classification

Home | WISeREP

https://www.wiserep.org

WISeREP

MENU

- SEARCH
- AUX
- BOTS
- CONVERSIONS
- ATELS
- ?
- ✉
- LOGIN

Search by name or coordinates

Coordinates, e.g.: 23:19:44.664 +10:11:04.56 OR: 349.9361 +10.1846

Submit

Please make sure to update your WISeREP URL to: www.wiserep.org

Getting started: If new to WISeREP v2, begin by acquiring a personal account here.
Then you may request to join an existing group (survey / program) or either initiate a new group if relevant.

In the new WISeREP v2, everyone can contribute and upload data (spectra / photometry / related-files) directly, either via the Report webpage or via the Bulk APIs.

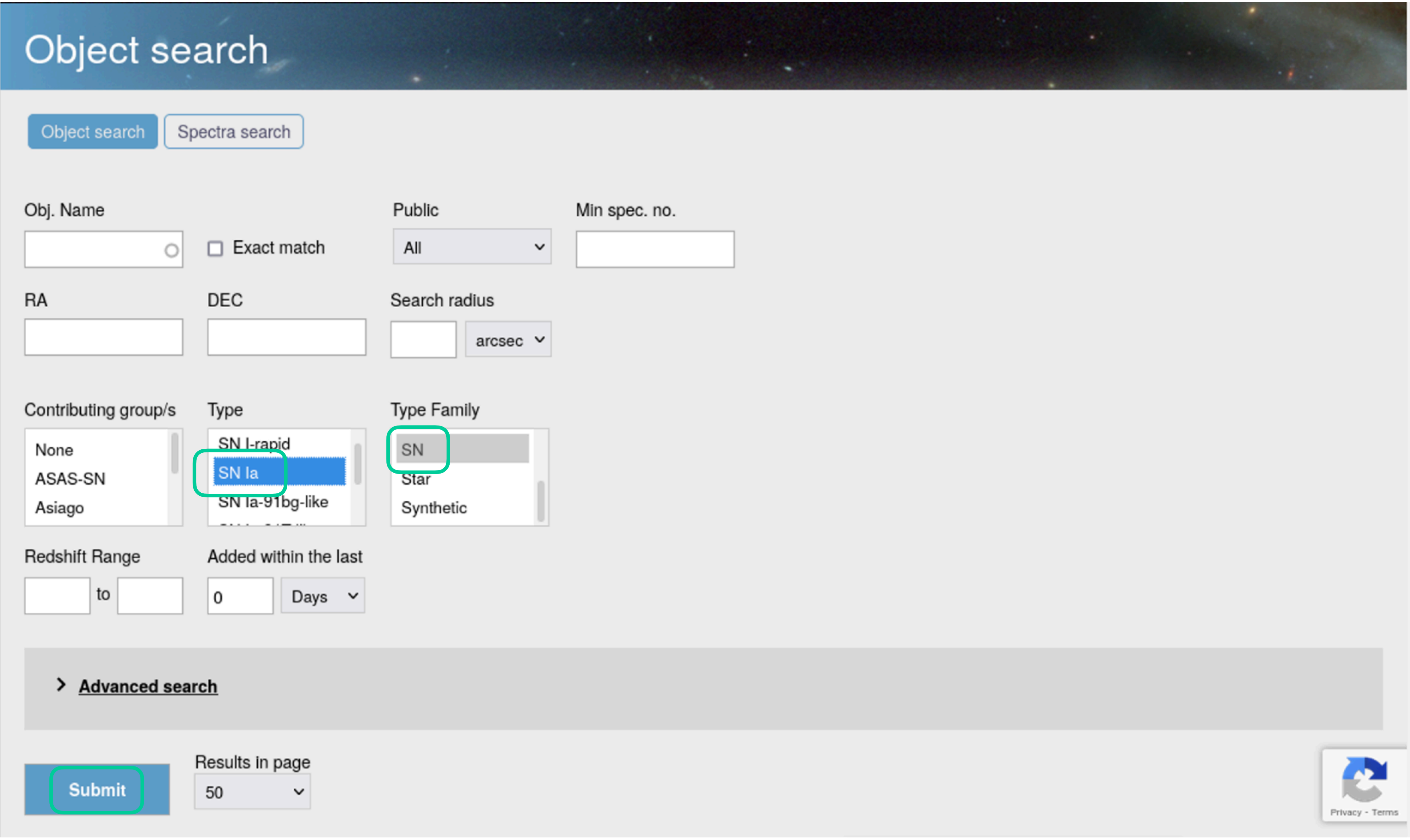
Please contact us if you encounter any problems or if you have any questions.

No. of Spectra: 48215

https://www.wiserep.org/search/object

Privacy - Terms

select suitable observed supernova spectra from the WISeREP archive, download the spectra, make notes of their types and redshifts



select suitable observed supernova spectra from the WISeREP archive, download the spectra, make notes of their types and redshifts

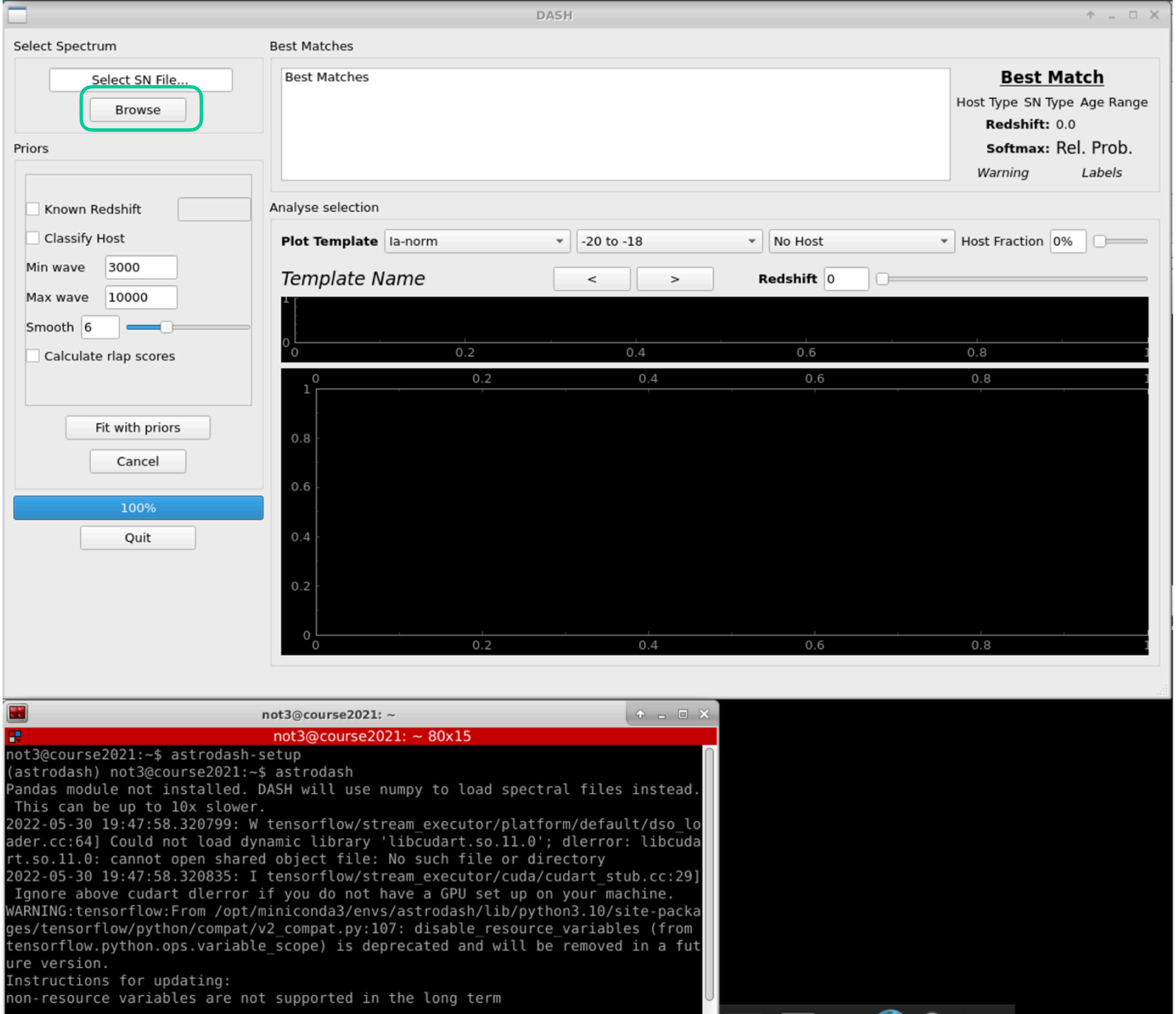
Showing results 1 to 50 out of 11031

1 2 3 4 5 6 7 ... > >>

ID	IAU Name	Internal / Alt. Name/s	Obj. Reps	Coords X-Refs	RA	DEC	Type	Redshift	Host Redshift	Host Name	Contributing Group/s	Public	Obj. Spectra	Obj. Phases	Light Curves	Related Files	Created by
20836	SN 2022lax	ZTF22aaldrem	2	Coords X-refs	14:24:50.686	-08:45:10.60	SN Ia	0.058			TNS	Y	1				TNS_Bot1
20842	SN 2022kwf	ATLAS22ozp, ZTF22aalfezf	3	Coords X-refs	17:36:17.968	+40:08:16.10	SN Ia	0.0387500003			TNS	Y	1				TNS_Bot1
20825	SN 2022kpy	ZTF22aalaajn	2	Coords X-refs	00:36:32.281	+19:23:54.61	SN Ia	0.062			TNS	Y	1				TNS_Bot1
20818	SN 2022kpr	ATLAS22oqv	2	Coords X-refs	16:26:57.750	+51:07:42.95	SN Ia	0.08			TNS	Y	1				TNS_Bot1
20828	SN 2022knm	ZTF22aakkmri, ATLAS22orp, PS22ekk	4	Coords X-refs	13:25:04.348	-24:39:24.80	SN Ia	0.04			TNS	Y	1				TNS_Bot1
20821	SN 2022kmp	ZTF22aakpwqf, ATLAS22okr	3	Coords X-refs	11:50:24.455	+22:42:48.92	SN Ia	0.067			TNS	Y	1				TNS_Bot1
20839	SN 2022klm	ATLAS22oht	2	Coords X-refs	17:33:57.181	+50:33:11.43	SN Ia	0.07			TNS	Y	1				TNS_Bot1
20806	SN 2022klj	ZTF22aajykxi, PS22eiq	3	Coords X-refs	16:18:24.720	+12:28:13.86	SN Ia	0.085			ePESSTO+, TNS	Y	1				WIS_Bot1
20786	SN 2022klh	ATLAS22off	2	Coords X-refs	15:21:19.265	+34:17:47.13	SN Ia	0.031492			TNS	Y	1				TN
20816	SN 2022klg	ATLAS22ofd	2	Coords	13:52:43.849	+59:16:42.91	SN Ia	0.03			TNS	Y	1				TN

Privacy - Terms

select suitable observed supernova spectra from the WISeREP archive, download the spectra, make notes of their types and redshifts


DASH
Select Spectrum
Select SN File...
Browse
Priors
Known Redshift
Classify Host
Min wave 3000
Max wave 10000
Smooth 6
Calculate rlap scores
Fit with priors
Cancel
100%
Quit
Best Matches
Best Matches
Best Match
Host Type SN Type Age Range
Redshift: 0.0
Softmax: Rel. Prob.
Warning Labels
Analyse selection
Plot Template Ia-norm -20 to -18 No Host Host Fraction 0%
Template Name
< >
Redshift 0
not3@course2021: ~
not3@course2021: ~ 80x15
not3@course2021:~$ astrodash-setup
(astrodash) not3@course2021:~$ astrodash
Pandas module not installed. DASH will use numpy to load spectral files instead.
 This can be up to 10x slower.
2022-05-30 19:47:58.320799: W tensorflow/stream_executor/platform/default/dso_lo
ader.cc:64] Could not load dynamic library 'libcudart.so.11.0'; dlerror: libcuda
rt.so.11.0: cannot open shared object file: No such file or directory
2022-05-30 19:47:58.320835: I tensorflow/stream_executor/cuda/cudart_stub.cc:29]
 Ignore above cudart dlerror if you do not have a GPU set up on your machine.
WARNING:tensorflow:From /opt/miniconda3/envs/astrodash/lib/python3.10/site-packa
ges/tensorflow/python/compat/v2_compat.py:107: disable_resource_variables (from
tensorflow.python.ops.variable_scope) is deprecated and will be removed in a fut
ure version.
Instructions for updating:
non-resource variables are not supported in the long term

make experiments with DASH for the spectra that you downloaded, do the spectral types agree with previous classifications, are priors needed?