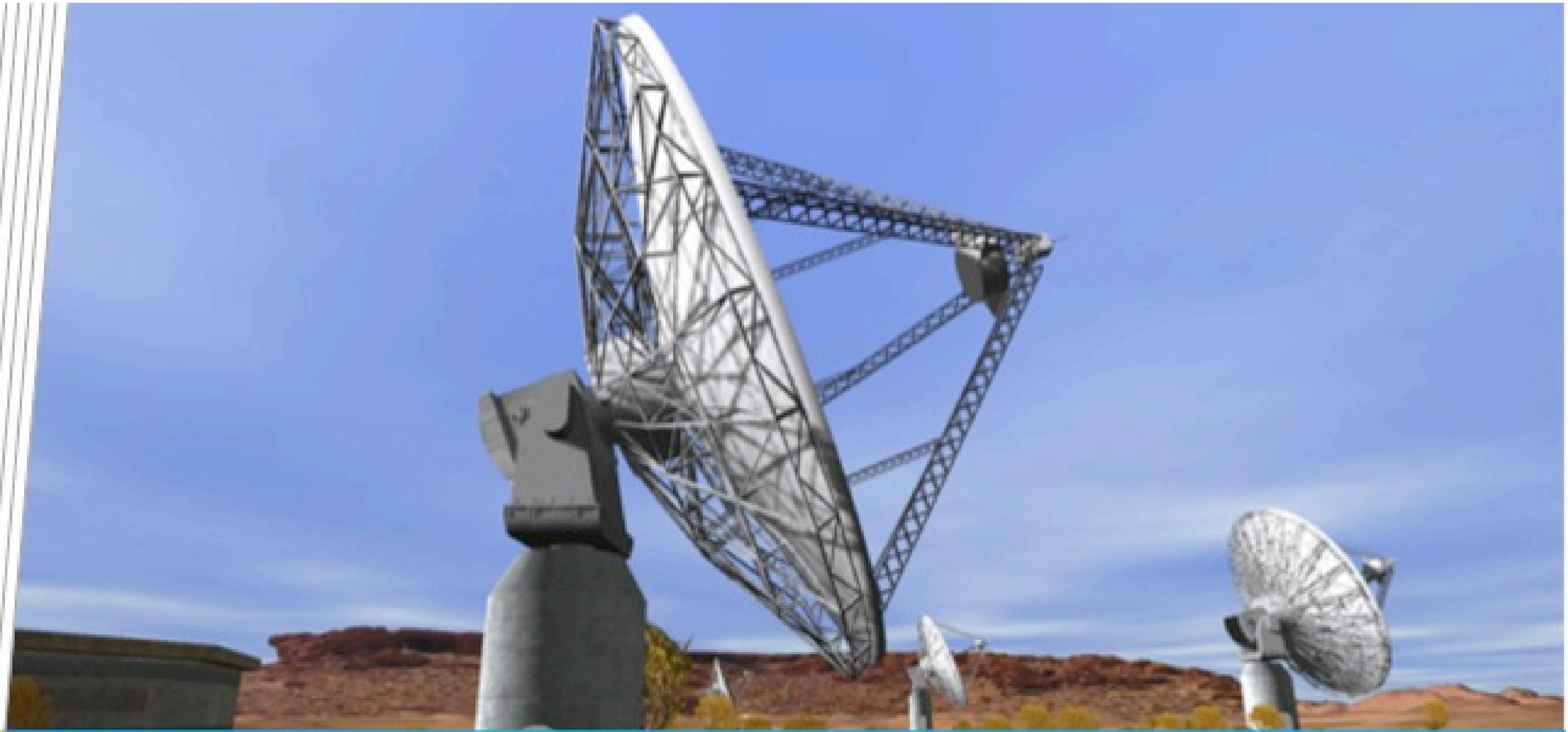


www.csiro.au

Source Detection and Cataloguing

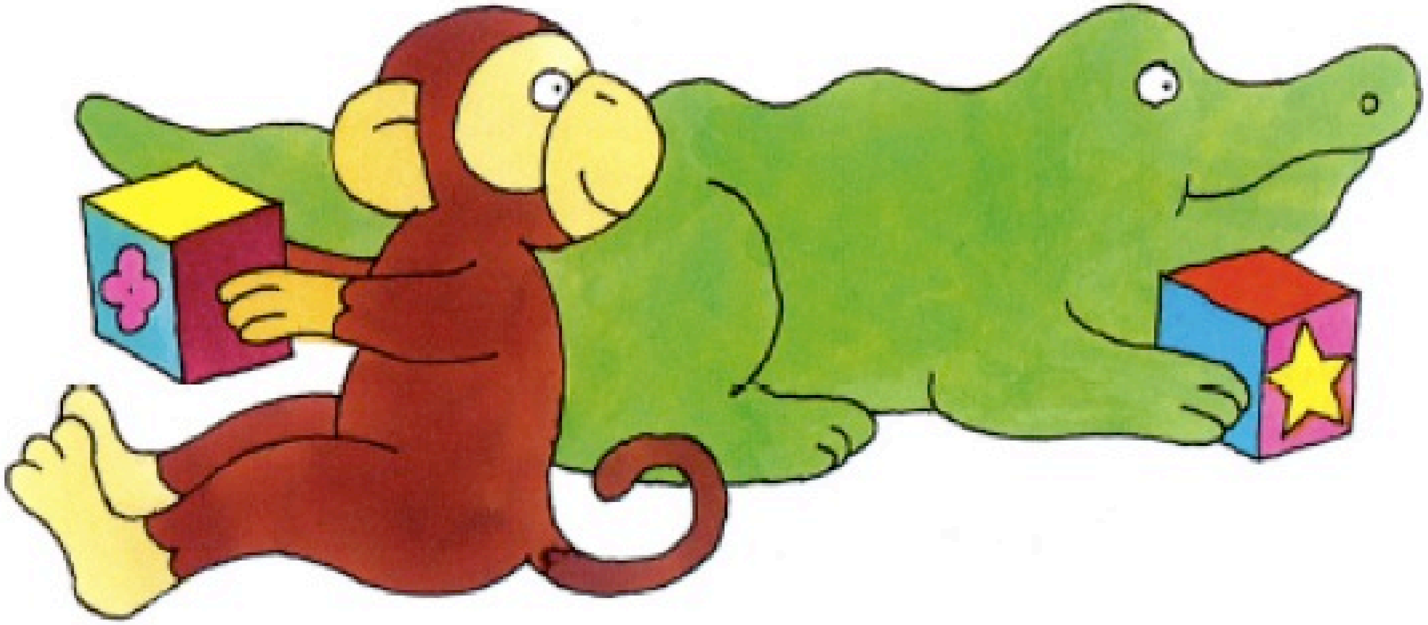
Dr Matthew Whiting
Australia Telescope National Facility

CSIRO

Outline

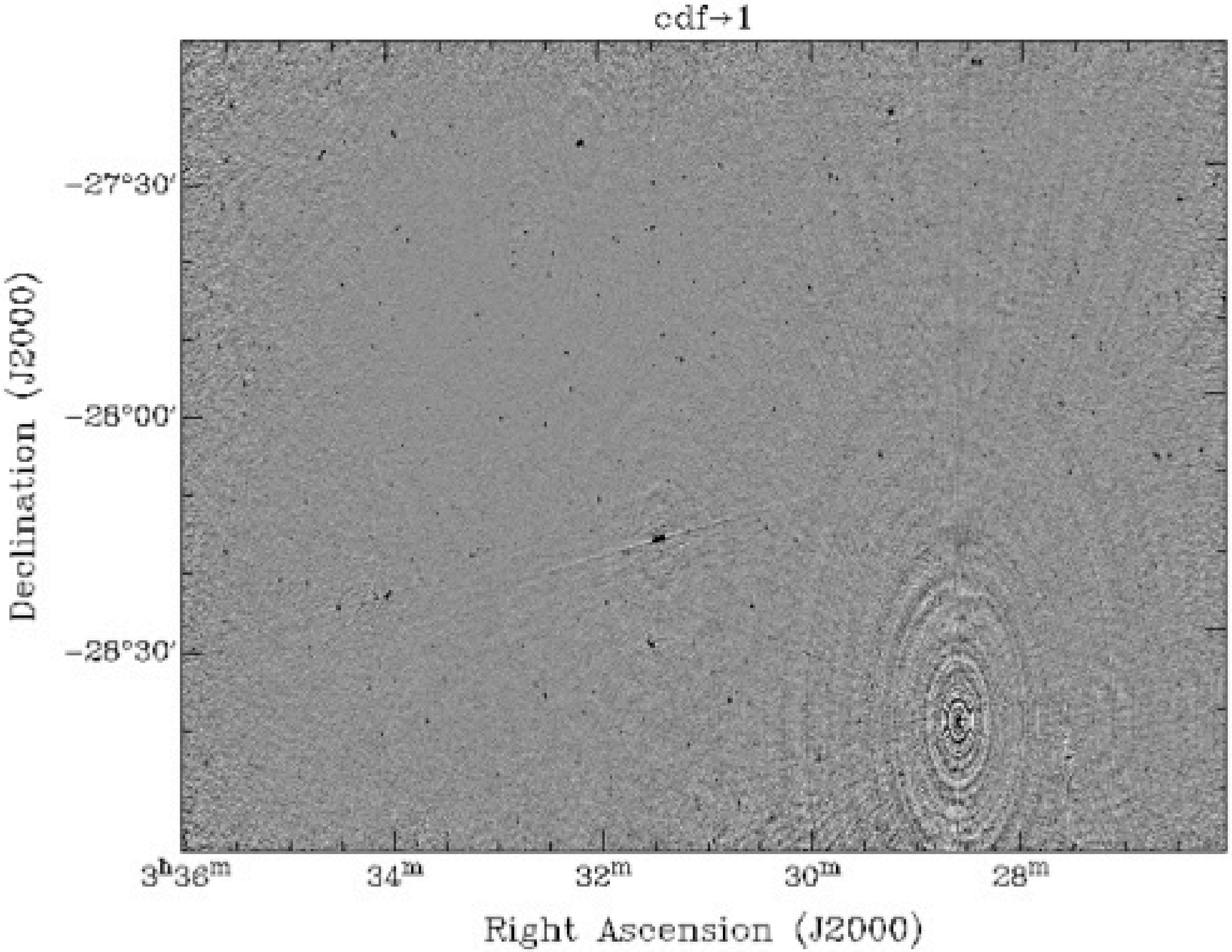
- Aims of source detection
- What is a source?
- Detections and noise
- How you measure the noise
- How you deal with the noise
- How you find sources
- What do you do with sources once you find them
- Software options you can use

Where's the star?



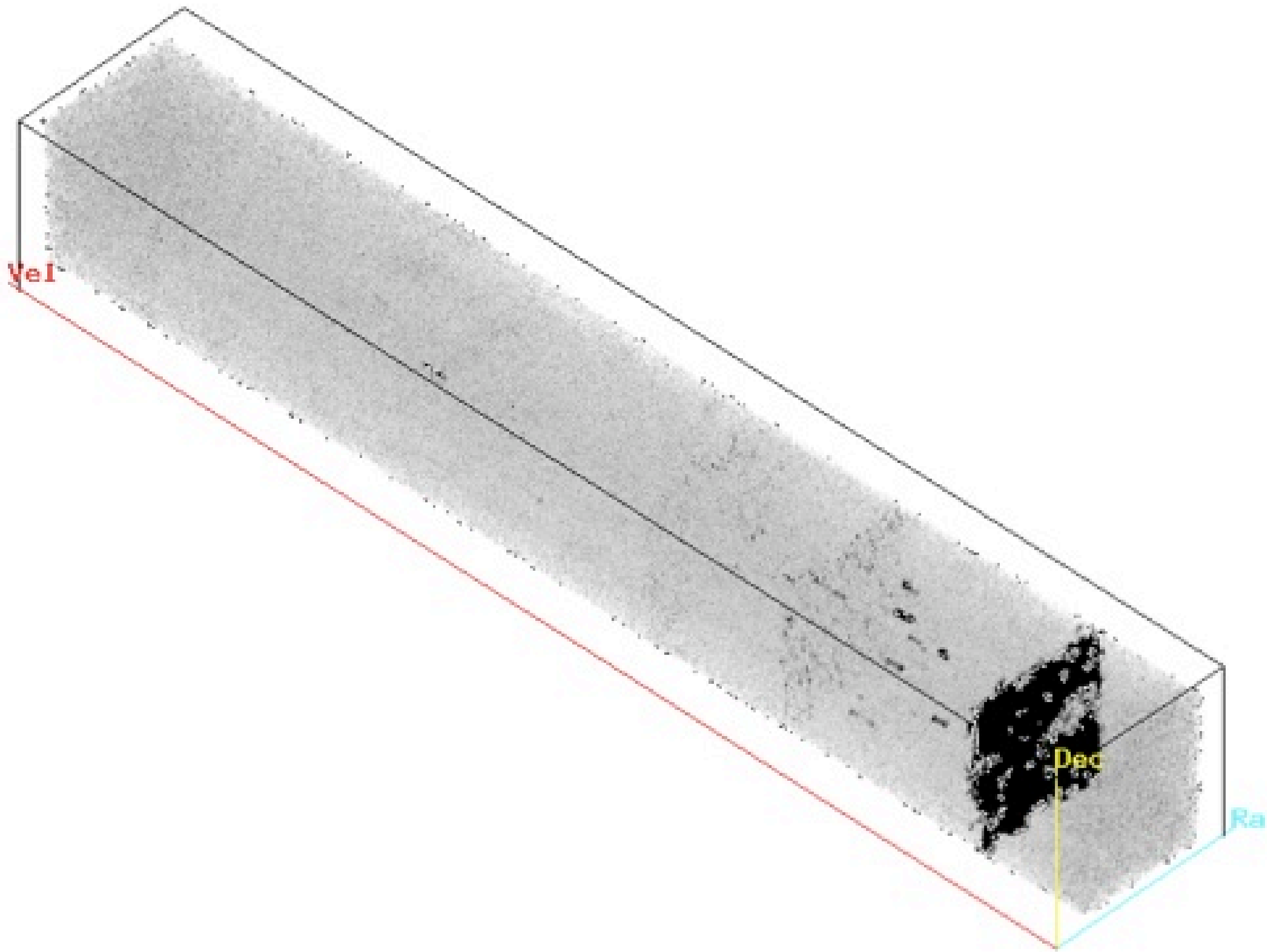
E.Hill (2005)

Examples

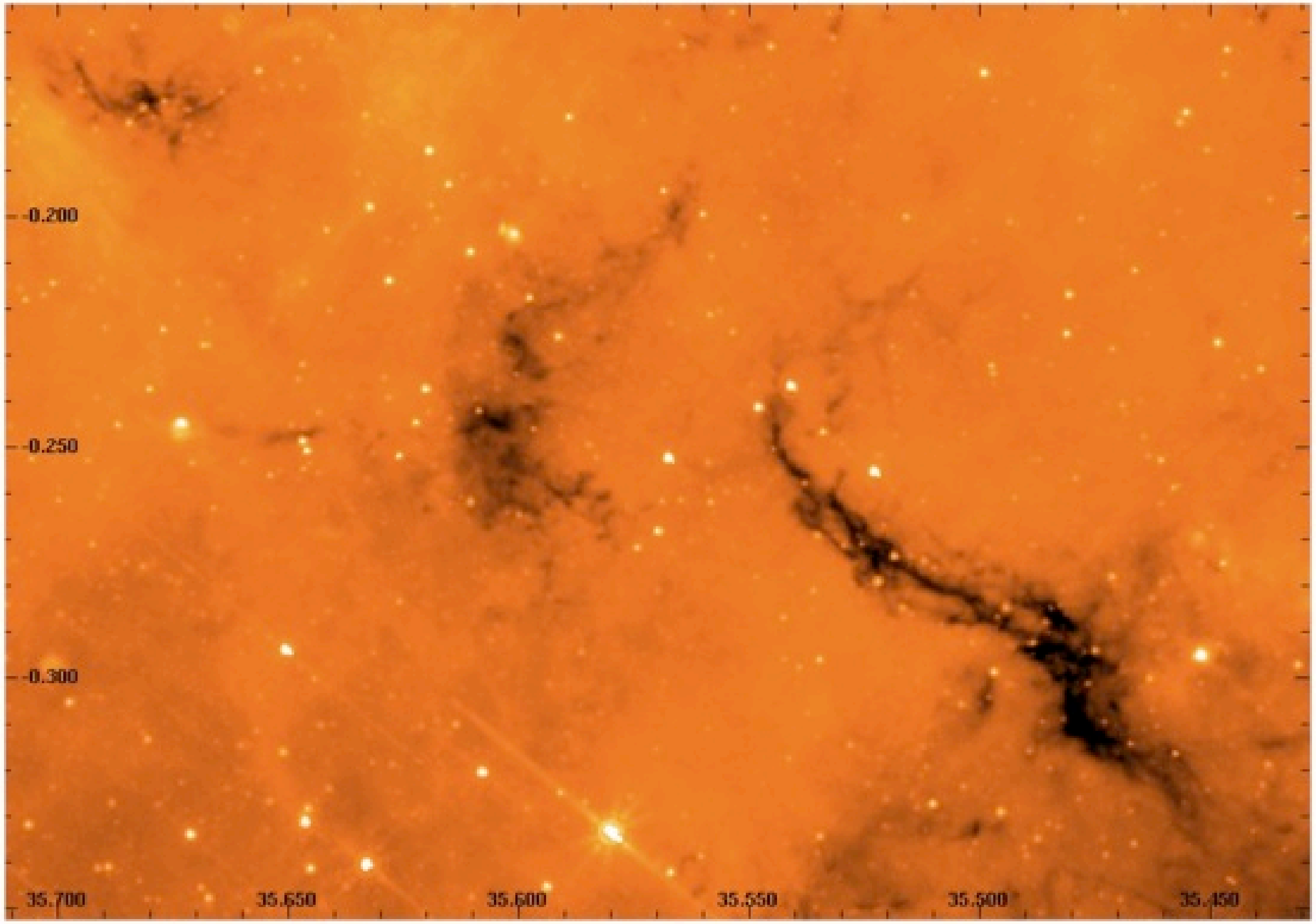


ATLAS CDFS field, Norris et al

Examples



Examples



Spitzer/GLIMPSE/U.Wisconsin

What do we mean by “source detection”?

- Location and cataloguing of objects of interest within your data

- Find all objects brighter than X in your image.
- Find all galaxies brighter than Y extending over Z km/s in your HI cube.
- Find all emission line peaks with S/N > Q

- Fit 2D Gaussian to each continuum source and record shape & flux
- Fit Gaussian components to each emission line
- Measure shape, extent and flux of extended emission in a continuum map

Detection and Noise

What is Source Detection?

- Key question:

Is this pixel value part of the background noise, or is it a "source"?

- Resolve via hypothesis testing
- H_0: Pixel value is due to the background noise
- H_1: Pixel value is due to something else

- Use statistical testing to reject (or not) H_0

Noise and source detection

- Background pixel values randomly distributed with a particular probability density function
  - Gaussian (Normal) distribution, $N(\mu,\sigma^2)$:

$$f(x|\mu,\sigma^2) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left[\frac{-(x-\mu)^2}{2\sigma^2}\right]$$

- Probability of a given pixel value governed by this function
- Use to test hypotheses.
- Example:
  - Assume the standard normal distribution, N(0,1)
  - Probability of x>3.2 is

$$\int_{3.2}^{\infty} f(x)dx = 1 - \int_{-\infty}^{3.2} \frac{1}{\sqrt{2\pi}} e^{-x^2/2} dx = 1 - 0.9993 = 7 \times 10^{-4}$$

- There is a 1 in 1429 chance that a 3.2σ "detection" is simply noise
  - This will occur about 11.5 times in a 16K-channel spectrum, or about 734 times in a 1024x1024 pixel image.

What do Gaussian errors mean?

$n\sigma$	Single-tail probability	# detections per ASKAP image (4096x4096)	# detections per ASKAP cube (4096x4096x 16384)
3	1.35e-3	22649	371 million
5	2.865e-7	4.5	78.7 thousand
6	9.87e-10	0.0166 (1 in 60)	271
7	1.28e-12	2.1e-5	0.35
10	7.62e-24	Small!	2.e-12

Errors, Reliability and Completeness

- No source detector will be perfect in the presence of noise
- There will always be errors, due to misidentified or missed sources

- False detection
  - False-detection rate = prob(data > S_{lim} | no source)

- Reliability
  - Fraction of your sample that are real sources
  - 1 - FDR

- Completeness
  - Chance that a real source is measured to be above the flux limit
  - Prob(data > S_{lim} | source)

What is the noise level?

- Key to implementing these sort of statistical tests is parametrising the noise:
  - How is the noise distributed?
  - What is the standard deviation and mean?
- But how do we measure the noise properties?
  - Separate measurements
  - From the data set we are searching
    - Noise is the background signal away from sources of interest
  - Construct a noise map
    - Important for imaging, as noise may vary with position

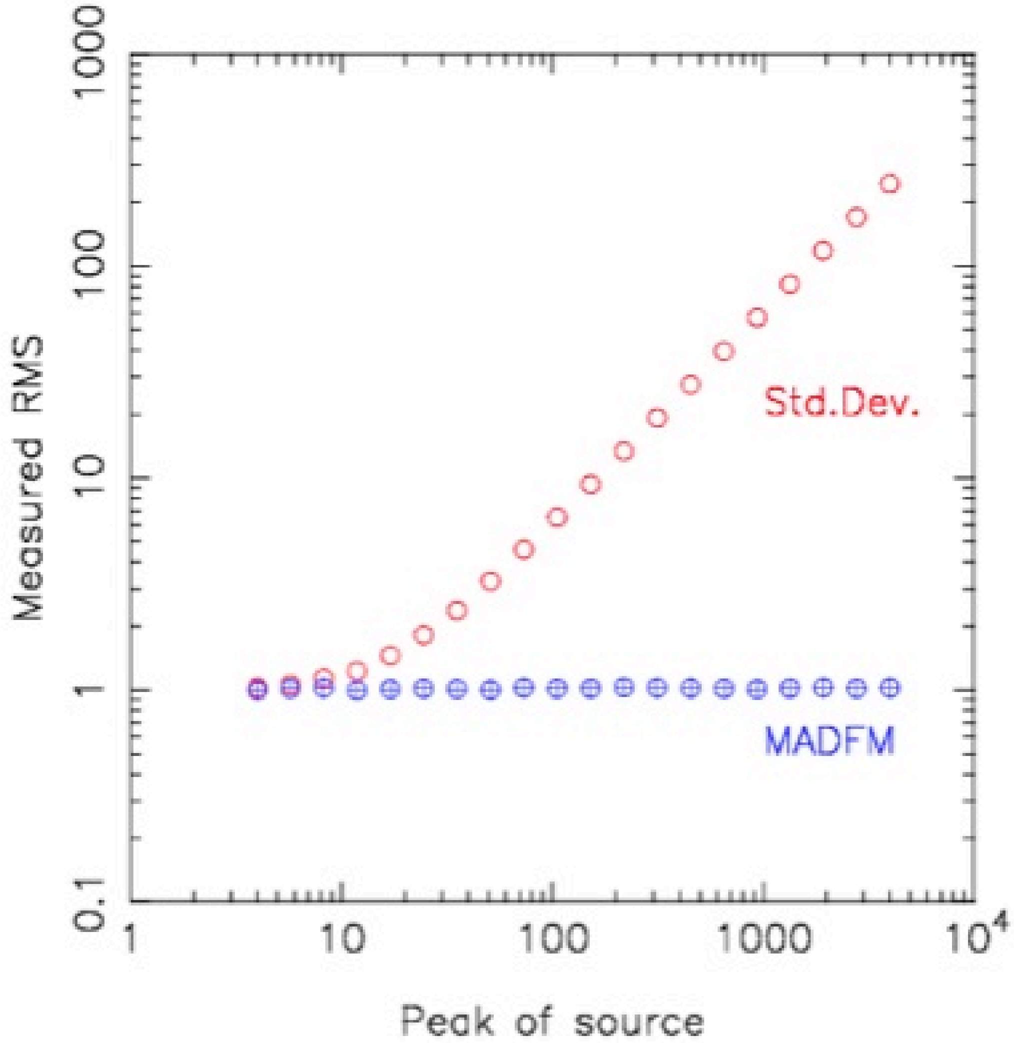
Robust techniques for noise estimation

- Suppose we want to estimate the noise from our data
- If there are bright pixels from sources present, this will bias the calculation of the mean and standard deviation:

$$\bar{x} = \frac{1}{N}\sum_{n=1}^{N} x \qquad s^2 = \frac{1}{N}\sum_{n=1}^{N}(x - \bar{x})^2$$

- Would like to not include those pixels, but that is part of the source-finding problem!
- Robust methods are those that are not affected by strong outliers:
  - Median rather than mean
  - Median absolute deviation from the median instead of the RMS.
  - Inter-quartile and inter-hexile ranges

Standard deviation vs. MADFM



Robust techniques

- Median = mid point of the ordered data set
  - Take set of data points
  - Rank them by value
  - Take middle point, or average of two middle points if even number
- Median Absolute Deviation from Median:
  - Find median
  - Find absolute value of the difference of each data point and median
  - Rank these values then take middle point
  - If assume Normal statistics, convert to standard deviation by

$$s = m/0.6744888$$

- Inter-hexile range
  - Hexile: divide a ranked list into six equal groupings
  - Inter-hexile range is the difference between the first and fifth hexiles
  - Semi-interhexile range is very close to standard deviation for a Normal distribution

Complications with noise

- Noise will not, in general, be nicely Normally distributed
- Central Limit Theorem will provide a good Normal distribution in most cases, although beware of the tails!
- However, other influences will confuse things:
  - Interference - localised in frequency or space
  - Sidelobes from bright sources
  - Artifacts from bright sources (e.g. CLEANing residuals)
  - T_{sys} variations across the field of view
- Have to be careful about extrapolating noise estimates from one part of an image/spectrum to other parts.
- One solution can be to make a "noise map"
  - Will lead to a varying detection threshold across your data
  - Affects completeness etc of the final catalogue

Enhancing detectability

Circumventing the noise

- We can use the fact that the noise has different properties to the sources to try and reduce its effect
- Key observation: the scale of noise fluctuations is often different to the scale of the sources in your data
  - Spectral-lines: HI galaxies many channels wide, but channel noise largely independent
- Use pre-processing to enhance structure on the scale of your sources and suppress the random signal
  - Smoothing
  - Wavelet reconstruction
- Process your raw data and then run your source detection algorithm over the processed data

Simple smoothing

- Average neighbouring pixels together in some way by using some sort of filter
  - Can use some form of weighting: e.g. Hanning smoothing
- Choose some width/scale, and noise on smaller scales will be smoothed out.
- Ideally, want Source scale > Filter scale > Noise scale
- Optimal approach is matched filtering, where your sources have a particular scale size, and you match the filter to that scale
  - Need to know this *a priori* which is not always possible
  - Needs to be a single scale, or it loses effectiveness
- Effect on noise: standard deviation of background will reduce according to the filter

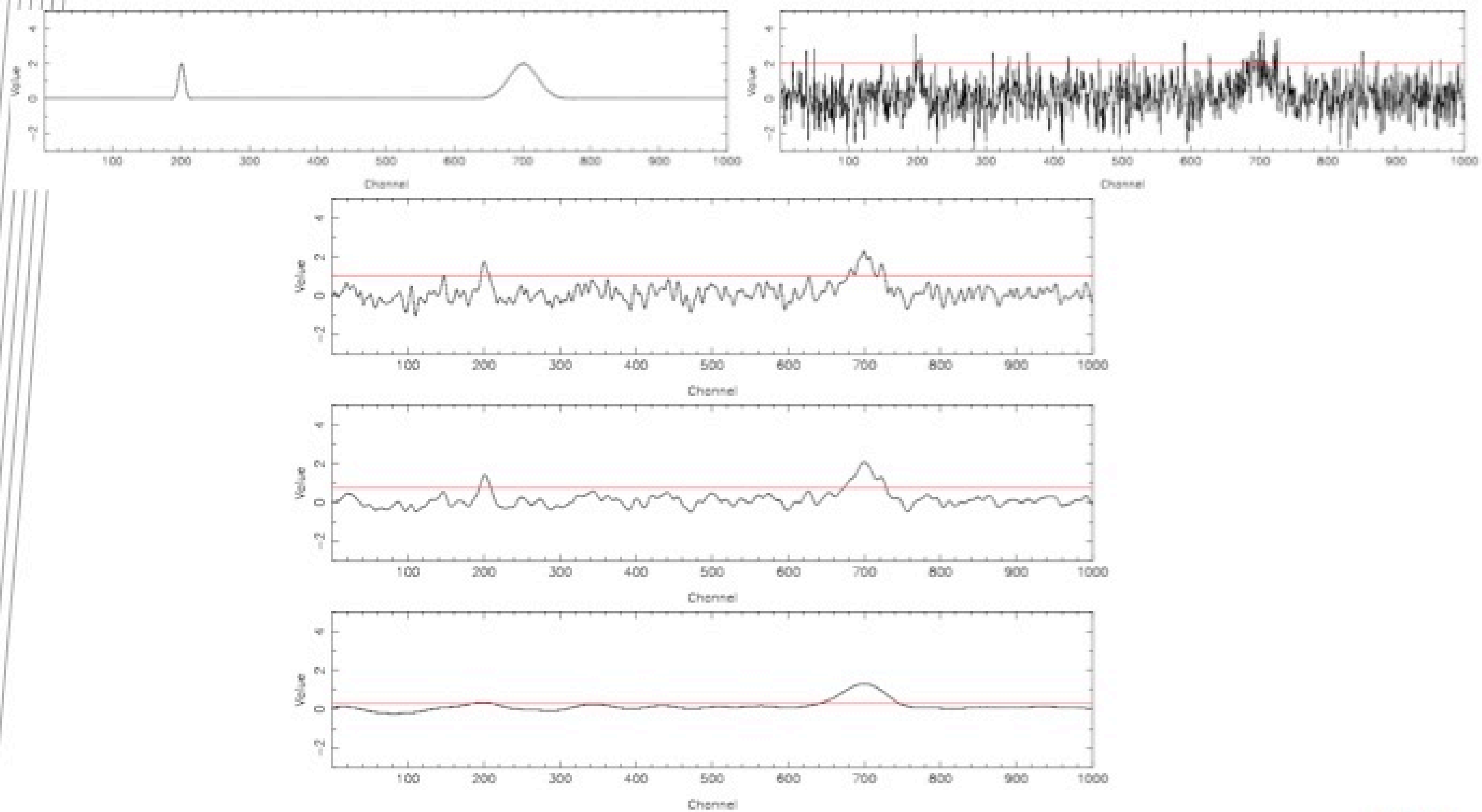
Filtering and noise

- Define filter by discrete components $\{w_j\}, j \in [1, 2n+1]$
- Have input spectrum $\{F_i\}, i \in [1, N]$
- Calculate new spectrum by filtering: $F_i' = \sum_{j=0}^{2n+1} w_j F_{i+j-n}$
- If the noise on all points in the original spectrum has the same standard deviation $\sigma_i = \sigma$
- Then the noise in the filtered spectrum will scale as:

$$\sigma_i' = \sigma \sqrt{\sum_{j=0}^{2n+1} w_j^2}$$

- E.g.: B_3-spline filter: $\{\frac{1}{16}, \frac{1}{4}, \frac{3}{8}, \frac{1}{4}, \frac{1}{16}\} \rightarrow \sigma_i = 0.5229\ \sigma$

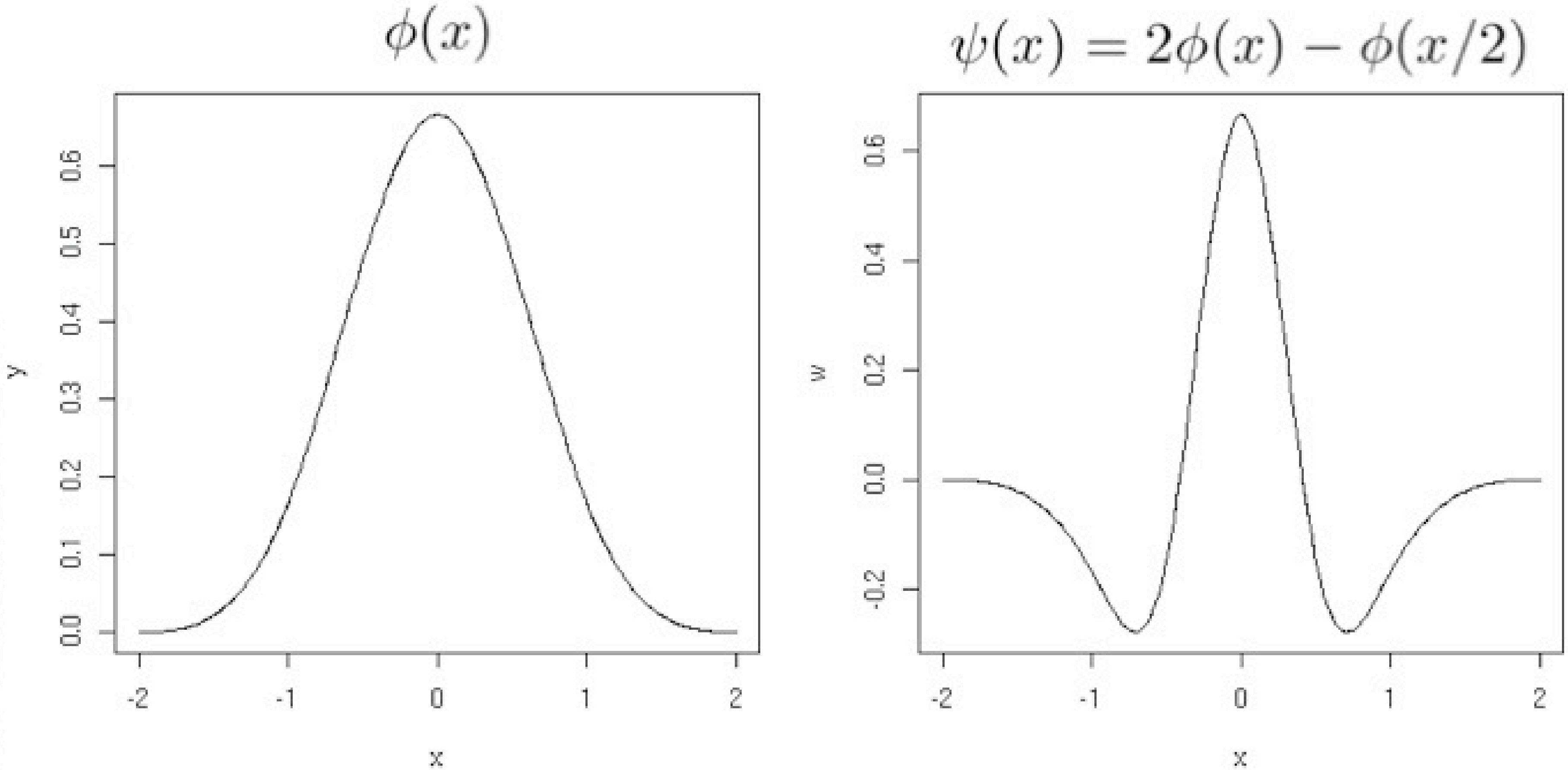
Filtering example



Wavelet reconstruction

- You may not know the typical source scale *a priori* or there may not be one unique scale
- It is possible to filter at a range of scales and use that information to reconstruct a noise-free spectrum/image
  - Highlight a logarithmically-increasing range of scales to cover the full range of possibilities
- One such technique is the *à trous* wavelet reconstruction algorithm, which can be used to remove unwanted noise.

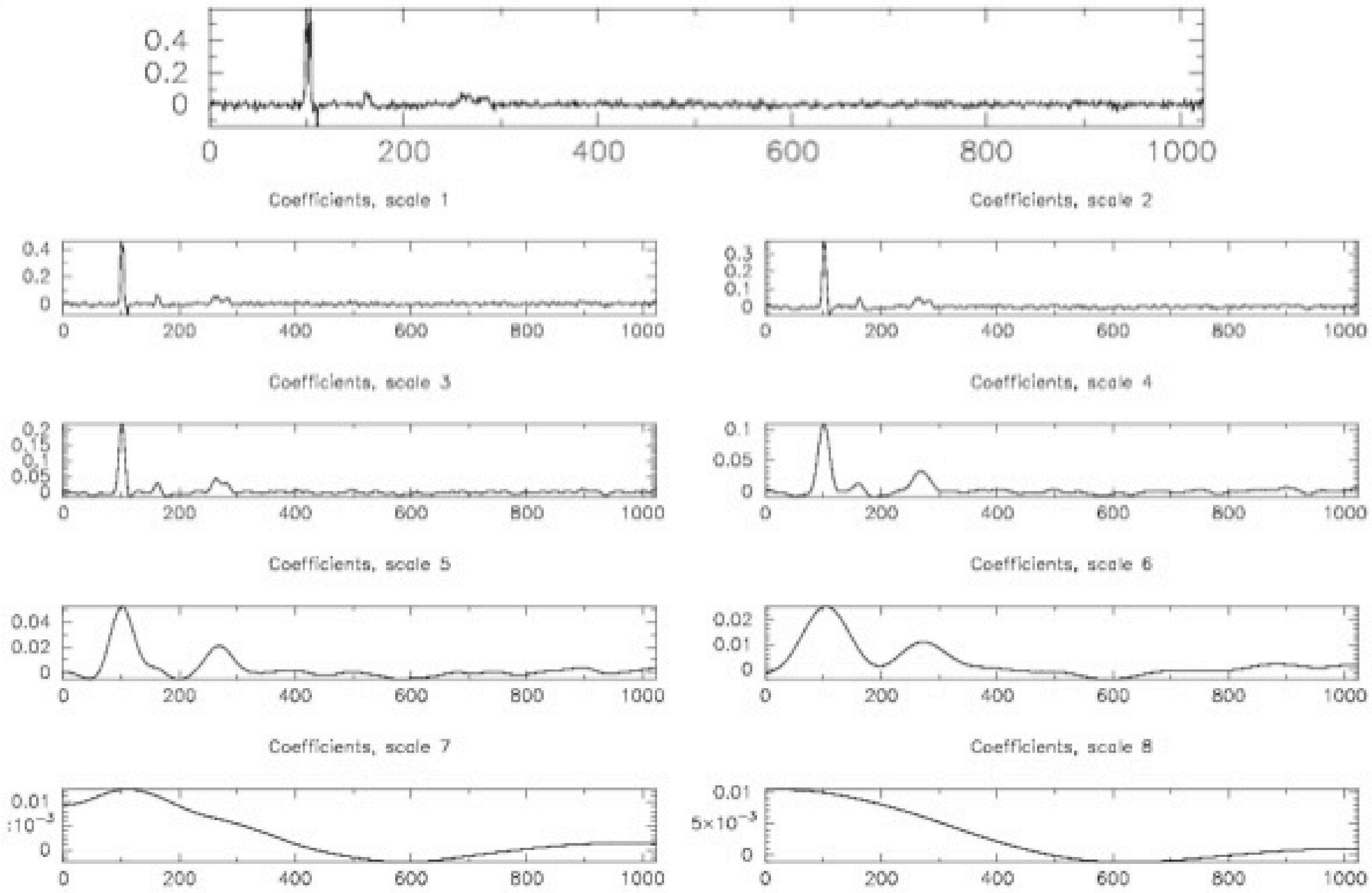
What are wavelets?



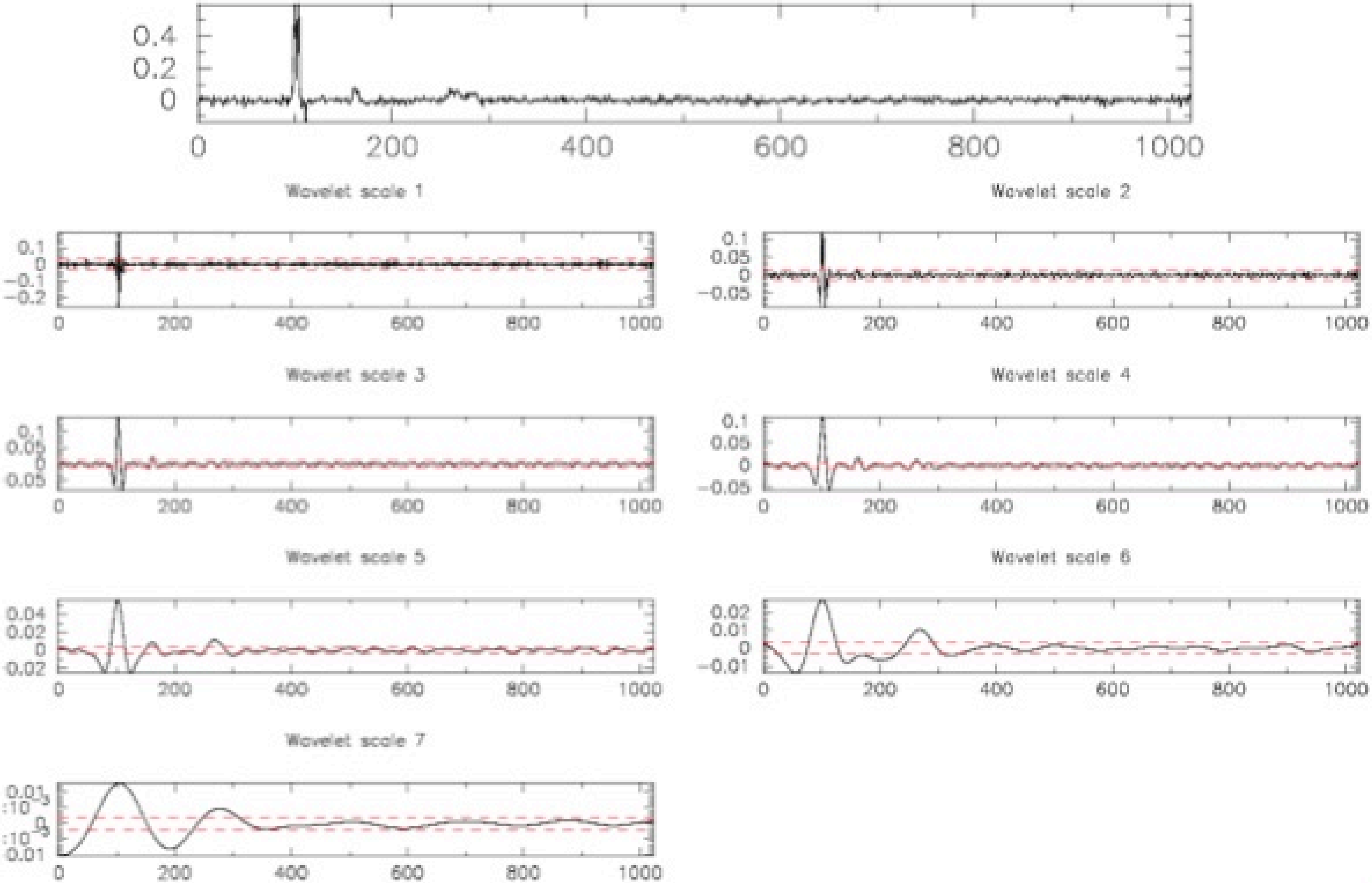
À trous algorithm

- Start with a spectrum (the input data): $S^0 = \{S_i^0\}, \forall i \in [1, N]$
- Also have a filter, used to smooth the data: $F^1 = \{F_j^1\}, \forall j \in [1, f]$
- Convolve the spectrum with the filter to produce first smoothed array $S^1 = \{S_i^1\} = S^0 \otimes F^1$
- Subtract the coefficients from the spectrum to produce the wavelet array $W_i^1 = S_i^0 - S_i^1$
- Apply some threshold to the wavelet array, so that only pixels with signal are kept. $\hat{W}_i^1 = \begin{cases} W_i^1 & |W_i^1| \geq T^1 \\ 0 & |W_i^1| < T^1 \end{cases}$
- Double the spacing between the filter coefficients
- Convolve the smoothed array with the filter
- Produce the wavelet array and apply threshold.
- Continue until size of filter ~ size of spectrum
- Reconstruction: add thresholded wavelet arrays, plus final smoothed spectrum. $R_i = \sum \hat{W}_i^k + S_i^n$

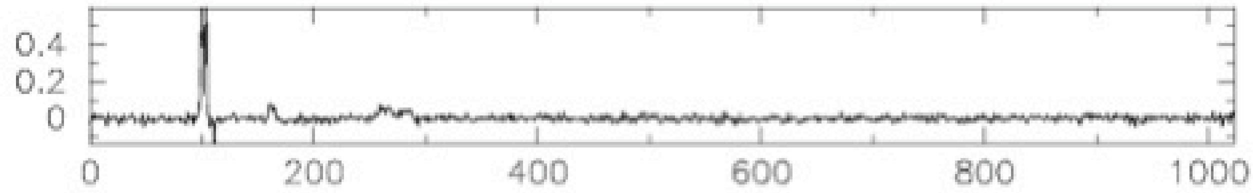
À trous example



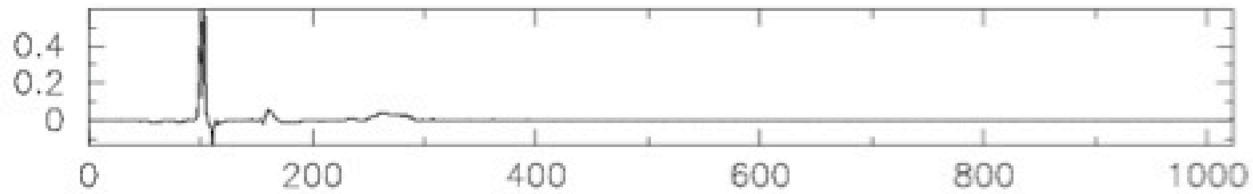
À trous example



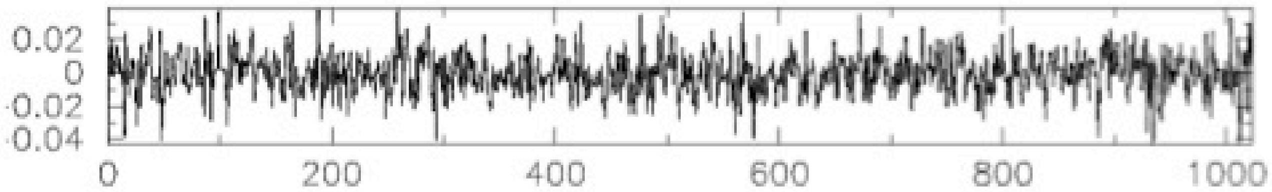
À trous example



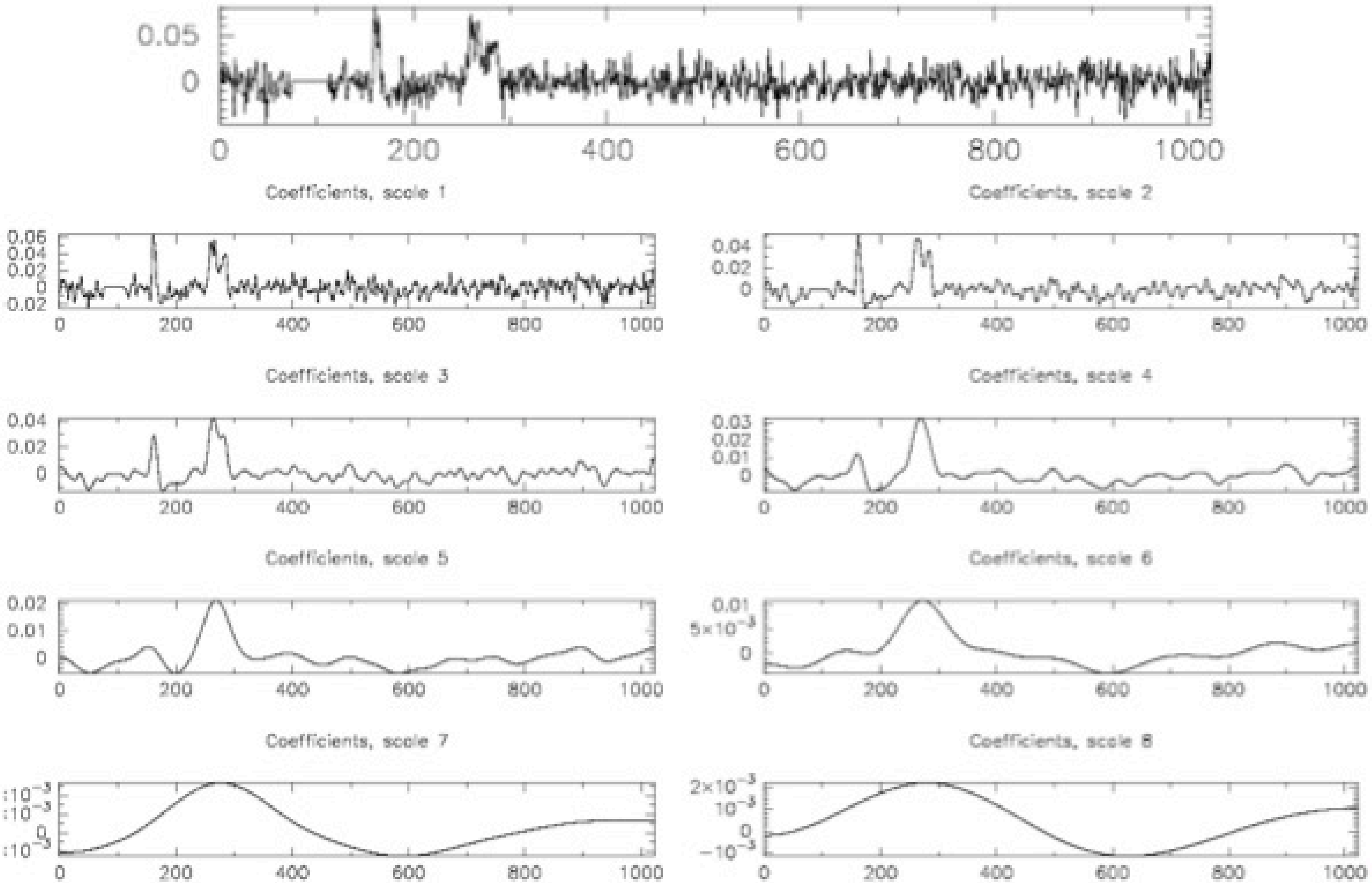
Reconstructed spectrum after one iteration



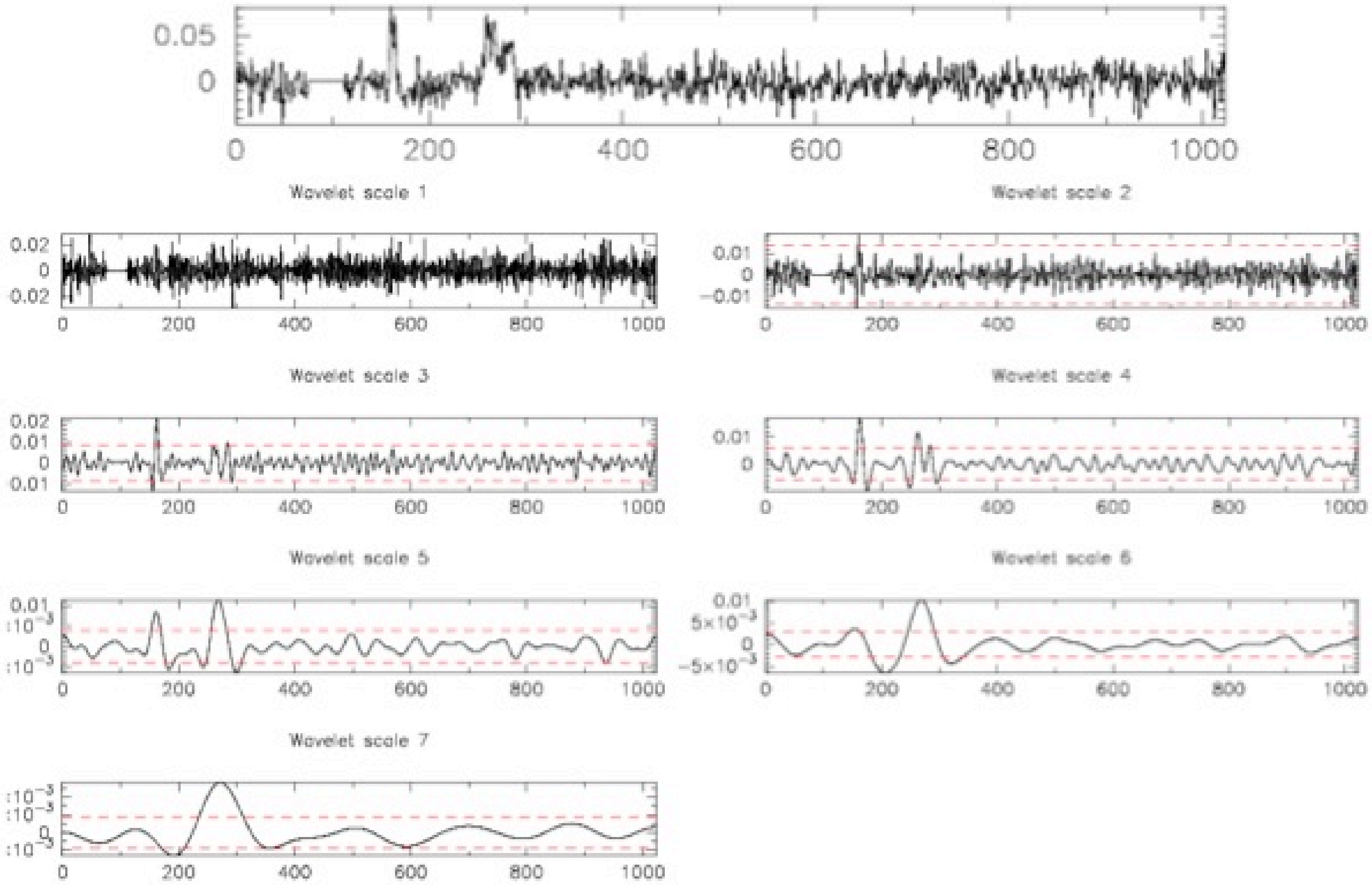
Residual spectrum after one iteration



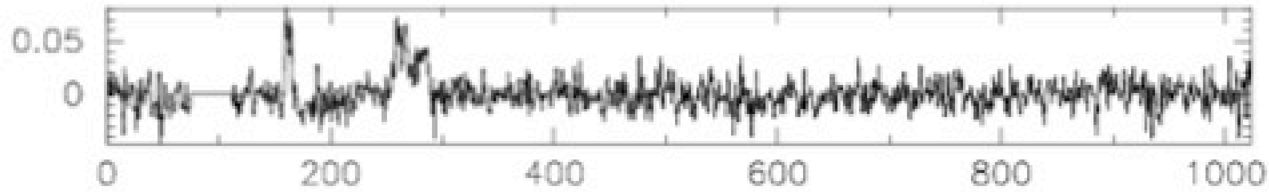
À trous example



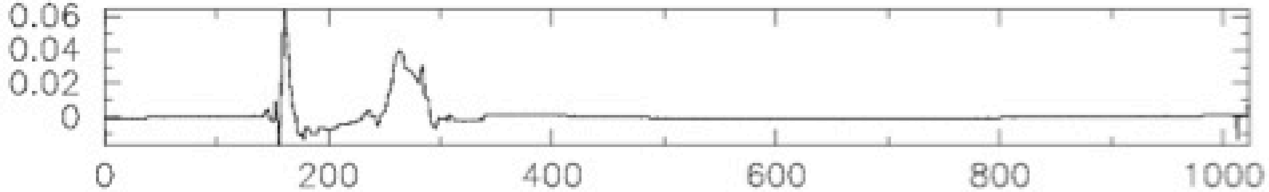
À trous example



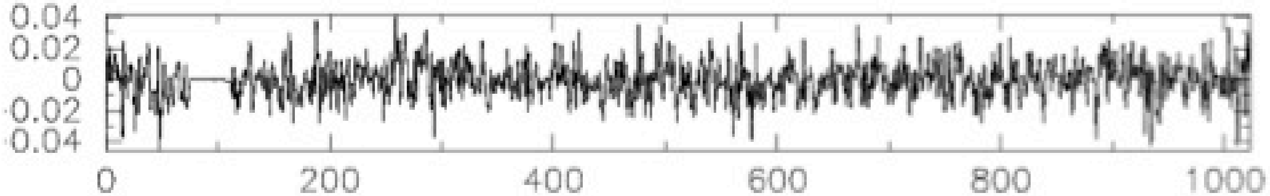
À trous example



Reconstructed spectrum after one iteration



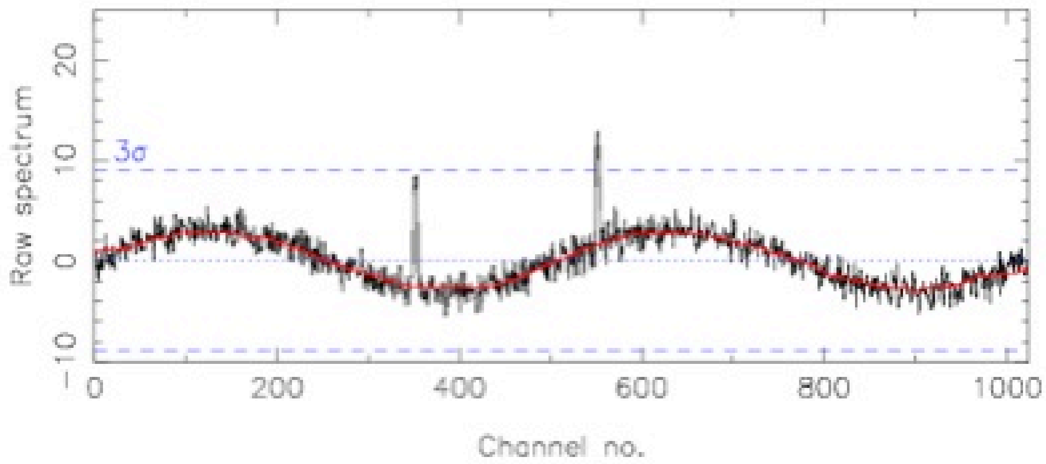
Residual spectrum after one iteration

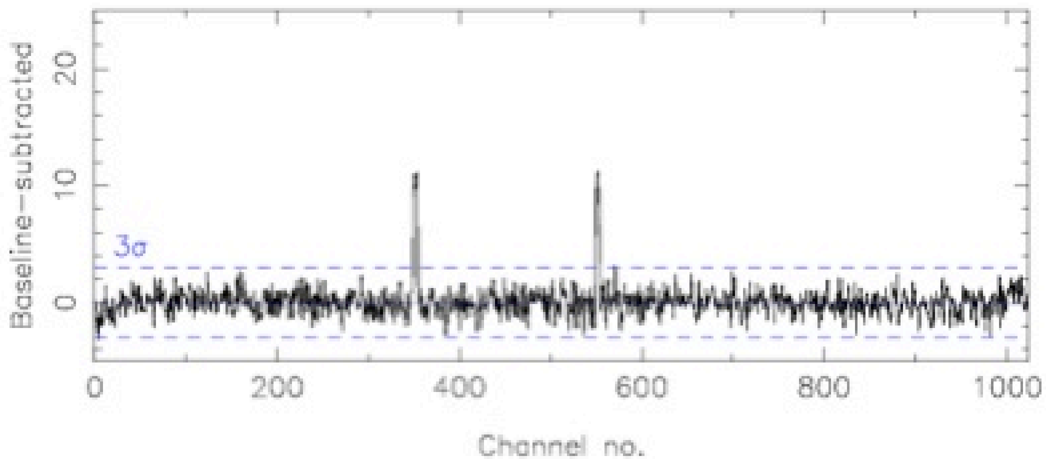


Baseline/background variations

- All examples shown here have mean(noise) = 0
- This need not be the case, however:
  - Baseline ripple in single-dish spectra
  - Solar interference
  - Errors in preconditioning
- Need to accurately account for the changing baseline before searching for sources
- Variety of ways to estimate the baseline:
  - Polynomial fitting
  - Median filtering
  - *À trous* reconstruction, keeping largest scale(s).

Baseline example





Source Extraction

How is *source* detection performed?

- Move from image segmentation to object detection
- Emphasis here on automated detection of objects
  - Automated operation necessary for modern data sets
  - Provides objectivity and reproducibility of results, and easily scalable
  - Needs to be well designed

- *Detected* pixels are those for which the null hypothesis is rejected.
- An *object* is a set of detected pixels that are connected in some way.
- *Connected* can be directly touching or within some separation threshold

- 1D is relatively straightforward
  - Look for connected pixels above the threshold
- Various algorithms for joining them up
  - Can scan along spectrum, starting a new object at a detected pixel and stopping it at a non-detected one.
  - Can start at the maximum point and grow out to non-detected pixels and continue to next maximum not part of an object.

2D source detection

- Two dimensions means an extra degree of freedom in which to connect pixels
- Still well behaved, with two important features:
  - Objects do not overlap within a given row of pixels
  - Objects are *well-nested*
    - Consider a row from an image. If a section of Object B lies between two sections of Object A, then *all* of Object B lies between those two sections.
    - Objects cannot cross each other and remain distinct
- This allows simple raster-scanning algorithms can be applied that examine each pixel in the image once only to pick out all connected objects.
  - Lutz (1980) is a good example: used in Duchamp & SExtractor

3D source detection

- The extra dimension breaks the simple arrangement seen in 2D
- The well-nested criterion no longer applies
  - Objects can be intertwined while still remaining distinct
  - Makes a simple raster-scanning algorithm not possible
- Need to use a two-stage approach
  - Search individual 2D channels or 1D spectra separately
  - Have a merging algorithm to combine objects that are connected
  - Needs to be carefully designed to not be too time-intensive
- Use knowledge about your dataset
  - Are most of the sources unresolved?
  - Is the emission extended & diffuse?

Spurious sources and their rejection

- Automatic source detection is great, but **you need to understand your data**
- Some apparent sources that will be picked up will not be the sort that you want
  - Interference often shows up spectrally as narrow bright spikes
  - Gridded data may show bright spatial pixels due to RFI in certain scans
  - Grating rings & spikes in interferometric data can resemble sources
- Basic requirements such as minimum number of pixels or channels can exclude a large fraction of RFI "sources".
- Awareness of where your sources are appearing is crucial to understanding the results of source detection.

Post-detection Analysis

- What do you want to do with your sources once you've found them?
  - Measure source parameters
    - Location, size, shape, flux, …
  - Fit standard functions to each source
    - 1D Gaussian profile (or other type of function) in frequency/velocity space
    - 2D Gaussian spatial profile
      - Standard approach for large continuum surveys: NVSS (Condon+ 1998), FIRST (Becker+ 1995), SUMSS (Mauch+ 2003)
  - 3D sources: create moment maps
    - 0^{th} moment: integrated flux
    - 1^{st} moment: mean velocity
    - 2^{nd} moment: velocity dispersion

Source detection tools: 3D

- Duchamp
  - An ATNF development (by me :)
    - http://www.atnf.csiro.au/computing/software/duchamp
  - Designed for sparse 3D spectral-line source detection
    - Isolated sources embedded in noise
    - HI surveys a good example
  - Provides wavelet reconstruction & smoothing options
  - Good graphical output
  - Continuing to be maintained and used in ASKAP development
  - Available for download. Runs as a standalone package.
- Clumpfind
  - Williams et al (1994), ApJ 428, 693
  - Designed with molecular-line surveys in mind
  - Decomposes clouds into 3D clumps via contouring
    - Finds peaks in the 3D contour map and follows them down to lower levels
  - Widely used in the literature
  - Available as part of miriad, also as stand-alone package.

Source Detection tools: 2D

- Many data-reduction packages will have a source-extraction tool
  - Sfind in miriad, SAD in AIPS
- SExtractor developed for optical data, considered state-of-the-art for 2D source extraction
  - Able to be used on radio data
- Duchamp able to examine 2D data
  - Source extraction algorithms being used for ASKAP development

- These all have their pros & cons
  - Depends on starting assumptions about sources
  - Treatment of sources varies

References

- *Practical Statistics for Astronomers*, Wall & Jenkins, CUP
  - Good round-up of statistical ideas for astronomy
- Lutz (1980), Computer Journal, **23**, 262
  - Extraction of objects from a 2D image
- Dixon & Kraus (1968), AJ, **73**, 381
  - A 1415 MHz continuum survey, with a good discussion of reliability & completeness
- Condon et al (1998), AJ, **115**, 1693
- Becker et al (1995), ApJ, **450**, 559
- Mauch et al (2000), MNRAS, **342**, 1117
  - Survey/catalogue papers for NVSS, FIRST & SUMSS
- Meyer et al (2004), MNRAS, **350**, 1195
  - HIPASS Survey catalogue
- Hobson & McLachlan (2003), MNRAS, **338**, 765
  - Bayesian approach to object detection. Applied to microwave background
- Williams et al (1994), ApJ, **428**, 693
  - Describes Clumpfind
- Starck & Murtagh (1994), A&A, **288**, 342
  - Description of the *à trous* transform as a noise suppression technique.
- Duchamp documentation
  - At http://www.atnf.csiro.au/computing/software/duchamp
  - Journal paper to be submitted shortly!

Australia Telescope National Facility
Matthew Whiting
ASKAP Computing, Science Applications

Phone: 02 9372 4683
Email: matthew.whiting@csiro.au
Web: http://www.atnf.csiro.au/projects/askap/
http://www.atnf.csiro.au/people/Matthew.Whiting

www.csiro.au

Thank you