Source Extractor for Dummies,
Everything you wanted to know about Source Extractor
and I was forced to find out.

DON'T PANIC!

Benne W. Holwerda
Space Telescope Science Institute, Baltimore, MD 21210

Contents

Chapter 1

Foreword

Sextractor for dummies is not a book in the famous series, I just liked the title. Sextractor is not a toy for grownups, it is a extremely usefull and versatile astronomical (and perhaps other disiplines) softwaretool.

In the course of my PhD I learned a great deal about it by trial and error and from Ed Smith and others. To have somewhere to explain every knob and dial is the motivation behind this manual. [1] I hope to make it as accurate and complete as possible. I hope it helps with whatever project you are working on.

---

[1] yes my motto is "ever read a manual as long there are still buttons and levers left..." but in sextractor's case you might want to make an exception. Besides I'm perfectly happy if you just skim it.

Chapter 2

Introduction

Source Extractor [1] is used for the automated detection and photometry of sources in fits files. SE works on scans of photographic plates as well as CCDs. However I am assuming you are working with ccd data.

If you goal is to get catalogs of all the detected objects with reasonably good photometry from a *FITS* file of processed astronomical imaging data, then SE is the instrument you need.

If you want REALLY good photometry on only a few objects and you know where they are, do not use SE. I'll discuss how SE works and subsequently how to start it and which parameters does what and what the different parameters in the catalogs mean. Some, but by no means all strategies for its use are discussed and to illustrate some examples from my own experiences will be given.

This started as a bunch of notes of mine and Ed Smith and is evolving slowly beyond that. I hope to provide for slightly more insight in SE for two reasons; first all the parameters will be discussed in one document, making it slightly more complete then either of the manuals and second it has DON'T PANIC written in large friendly letters on the cover.

---

[1] Sometimes abbreviated to 'sex', like the executable. If this funny to you a second time you might want to get out more.

Chapter 3

Pros and Cons of SE

As I tried to point out in the introduction and will emphasize occasionally SE is not suited for every astronomy project that needs photometry of objects in a field. As everything, it has strong and weak points.

The pros of SE are listed in the manual but the most important ones are:

1. Speed. SE is made to go through data quickly. And if you're trying to beaver through several square degrees of data, speed is GOOD.

2. The capacity to handle large fits files. SE is coded up so that it'll take it a piece at a time. Again good for the skyeaters among us.

3. Works on CCD and scanned photographic plate data. Nice if you happen to have this kind of data.

4. Does decent photometry.

5. Robust, it'll run with idiotic input.

6. Controllable, most steps can be influenced by user.

7. The possibility to accept user specified flag images of weight images.

8. Output parameters and the order in which these are listed, are specified by the user.

8. Output files depicting apertures, detections and more.

9. The possibility to detect in one image and do the photometry in another.

However SE has some drawbacks. It was made for speedy use and in some case accuracy has been sacrificed for speed on purpose. So here is the other side of the coin:

1. Only as good as its settings. SE is dependent on some setting and these are crucial for the detection and photometry. It will run on just about any set of input parameters but give back output that may be total bogus.

2. Manuals are outdated and incomplete. This handbook is written as a remedy for that but by a user and not the person who wrote the code. [1]

3. Limited accuracy. You'll see this with the geometrical output parameters. These are computed (from moments), NOT fitted (which would be more accurate).

4. Classification of objects is of *very* limited use.

5. Breaks down in crowded fields eventually.

6. Corrections of photometry for the 'wings' of object profiles is very rudimentary.

---

[1]So I'm NOT claiming completeness or correctness.

Chapter 4

How to install SE

First you get the most recent version of SE from *http://terapix.iap.fr/soft/sextractor/index.html*. Then you unzip and tar the file (UNIX):

```
gzip -dc sex_2.2.2.tar.gz | tar xv
```

That should leave you with a directory in the directory where you did this called *sextractor2.2.2/* with instructions on how to install in 'INSTALL'. Basically you go to the 'sextractor2.2.2/source' directory and type:

```
make SEXMACHINE= 'machine type'
```

where the 'machine type' can be any of the following possibilities:

```
aix     (for IBMs RS6000 running AIX)
alpha   (for DEC-ALPHAs with Digital UNIX)
hpux    (for HP/UX systems)
linuxpc (for PCs running LINUX, using gcc)
linuxp2 (for Pentium2/3/4 PCs running LINUX)
linuxk7 (for Athlon PCs running LINUX)
sgi     (for SGI platforms)
solaris (for SUN-Solaris machines)
sunos   (for SUN-OS platforms)
ultrix  (for DEC stations running ULTRIX)
```

The SE manual is available in postscript format to you in the 'sextractor2.2.2/doc' directory. Congrats, you have Source Extractor available as an executable 'sex' in the 'sextractor2.2.2/source' directory. The 'make' file

tries to make a shortcut to this executable in you 'home' directory. If this fails, try making an alias for the command or simply type the whole path *./sextractor2.2.2/source/sex* .

Chapter 5

How SE works

The source extractor package works is a series of steps; it determines the background and whether pixels belong to background or objects. Then it splits up the area that is not background into separate objects and determines the properties of each object, writing them to a catalog.

The background determination is treated in the manual and in section 6.2 All the pixels above a certain threshold are taken to belong to an object. If there is a saddle point in the intensity distribution (there are two peaks in the light distribution distinct enough), the object is split in different entries in the catalogs. Photometry is done on these by dividing up the intensity of the shared pixels. There is an option to "clean " the catalog in order to eliminate artifacts caused by bright objects. After there is a list of objects with a series of parameters measured (ellipticity, size etc.). These are classified into stars and galaxies (everything non-star) by a neural network.

The first steps are controlled by a number of parameters. How to estimate the threshold? How much contrast should there be to split an object? However the classification by the neural network is only depend indirectly on the parameters controlling the first steps. As the network has been trained on ground based data, there might be some doubts on the reliability of this classification as one switches to other passbands or instruments. So check up on this classification in the case of faint of blended objects.

Some of the steps in SE have maps associated with them and these can be written to fits files, the 'check' images.

Aside from the inputimage, SE can handle weightimages and flag-images

to mark the relative importance of pixels or to flag bad ones.

The fitting of the Point-Spread-Function is not yet operational but it was already indicated in the second flow diagram in manual version 2.1.3.

Steps of SE:

1. Measure the background and its RMS noise (background and RMS maps). (6.2)

2. Subtract background.

3. Filter (convolve with specified profile). (6.3.2)

4. Find objects (thresholding). (6.3.1)

5. Deblend detections (break up detection into different objects. (6.3.3)

6. Measure shapes and positions.

7. Clean (reconsider detections, accounting for contributions from neighbors) (6.3.4)

8. Perform photometry.

9. Classify/index level of fuzziness –¿ more star-like or galactic?

10. Output Catalog and 'Check' Images
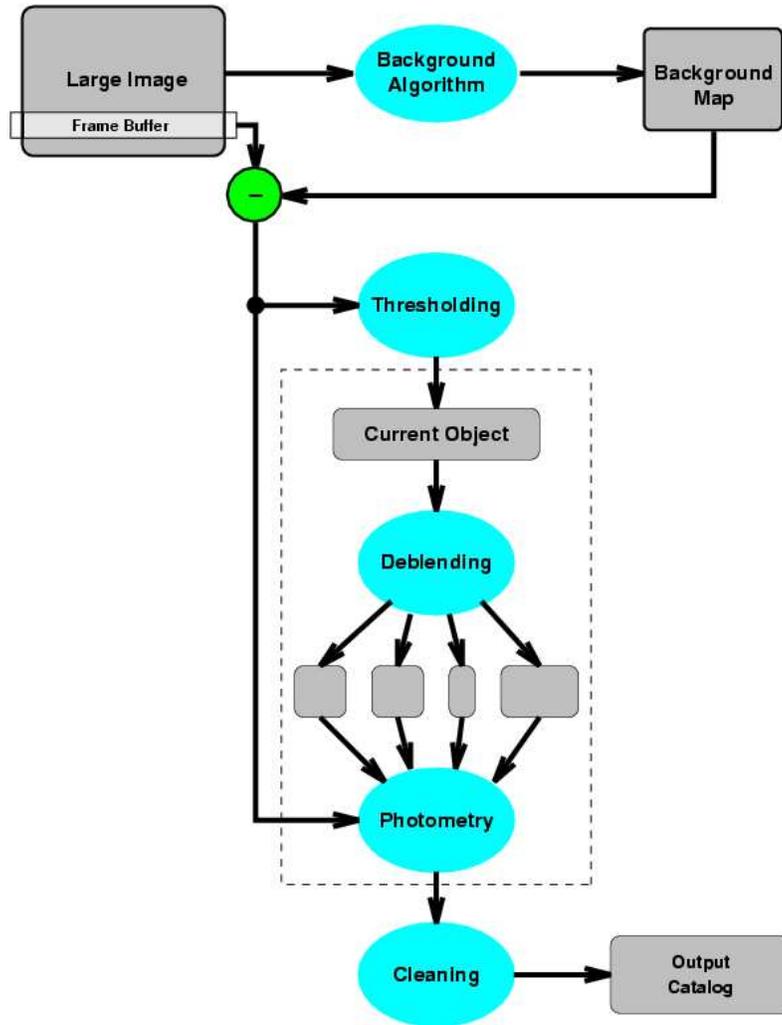
Figure 5.1: SE flow diagram from the first manual. This shows the principal steps done by SE.
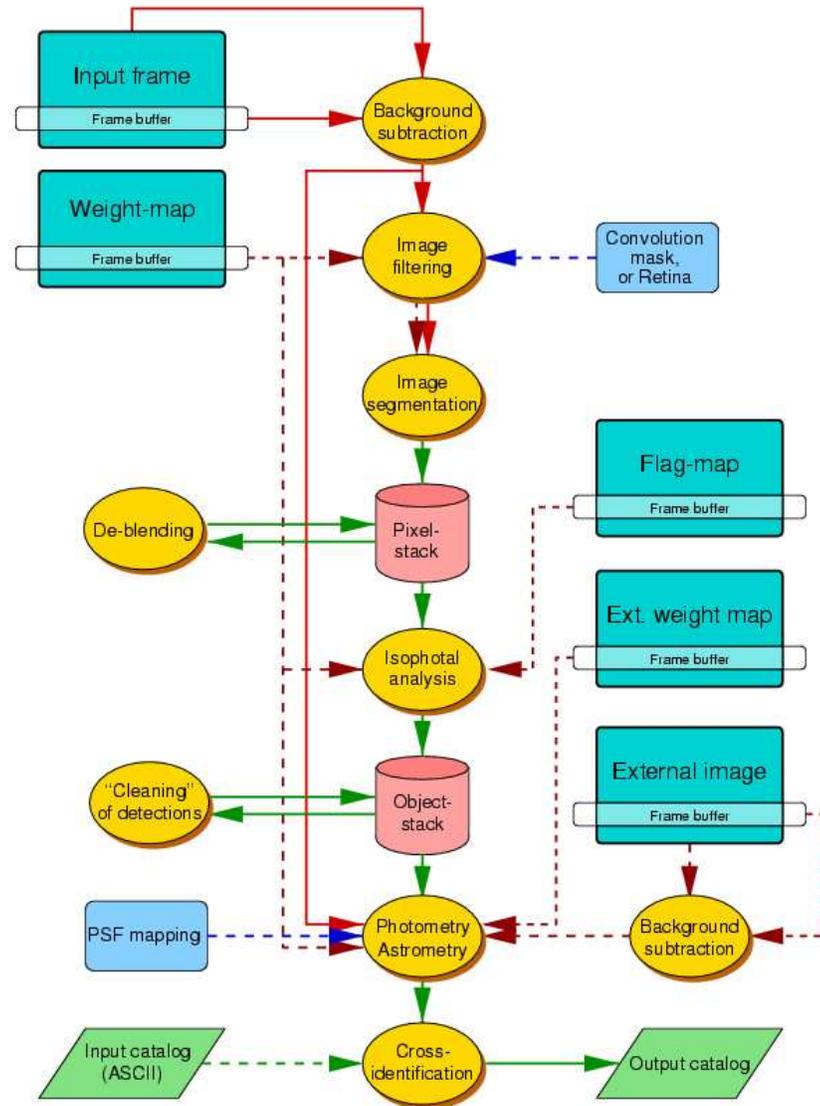
Figure 5.2: SE flow diagram from the second manual. This shows clearly the many extra options over a simple run of SE; the possibilities of weight maps, flags, crosscorellation catalogs and many more.

Chapter 6

Using SE

Source extractor can be used in basically three ways: on a single file for both detection and photometry, on two files, one for detection, the other for photometry and on two files with cross-identification in the catalogs.

6.1 Using SE on one image

Se needs a series of parameters in order to run and these can be given at the command line or in a configuration file. So to run SE on a single file with all the necessary parameters in the configuration file one types:

sex *image* -c configuration_file.txt

If there is no configuration file given, SE will try to read 'default.sex' from the local directory. However, the parameter values can be fed to SE on the command line as well:

sex *image* -c configuration_file.txt -PARAMETER1 *value1* -PARAMETER2 *value2*

The names of the parameters and their meanings and preferred values are discussed below.

NOTE: if you use both a configuration file and command line parameter input, the command line input takes precedence over

19

the configuration file value.

6.2   Using SE on separate images for detection and photometry

This may sound weird but the option in SE to find all the objects in one image and then apply the apertures and positions found on another image can be quite useful. To use *image1* for the detection of sources and *image2* for the photometry:

sex *image1,image2* -c configuration_file.txt

Suppose you want different colors of color information from the series of images you have in different filters. It is in that case very convenient to use the same apertures. So provided the images are well aligned, [1] the photometry done is essentially the same objects using the same apertures. [2] The nice part is that the numbering in the catalogs in this dual mode and the numbering in the single mode on the first image are the same. It should spare you a lot of rooting around in the catalogs if you want to compare fluxes in different bands for instance. This particular feature is further discussed in the strategy section.

6.3   Crosscorrellating catalogs. (ASSOC)

There is a third possibility to get information on objects that occur in separate images, for instance in overlapping fields in a survey or simulated objects in the SE detections. Basically you run SE on one field and take the X and Y positions (in pixels) from the catalogs and feed them to SE in a second run with a search radius and a priority (the brightest association or the nearest? etc.) And the matches are printed in a new catalog. See the ASSOC parameters in the catalog configuration section.

---

[1] The bright sources are on the same pixels, check by loading both in saotng or another display package and then 'blink' between them. If you're drizzling or stacking several exposures, use the same reference image.

[2] If you want awfully good photometry then it might be good te realize that a point spread function correction is dependent on the filter used. But keep in mind that SE doesn't do THAT good photometry to start with.

Chapter 7

SE Input; the Configuration File

"On two occasions, I have been asked [by members of Parliament], 'Pray, Mr. Babbage, if you put into the machine wrong figures, will the right answers come out?' I am not able to rightly apprehend the kind of confusion of ideas that could provoke such a question."
– Charles Babbage [1]

As stated in the previous section, SE tries to read the configuration file *default.sex* or a file can be given on the command line. [2] The *default.sex* can be found in the */sextractor2.2.2/config/* directory. It gives a good set of defaults for SE to use.

The configuration file is good way of remembering which parameters you used in running SE and you do not have to reset all the parameters when running SE over a series of files. The configuration file is an ASCII file (plain text) with the name of the parameters and the value on separate lines. A comment line begins with '#' and ends with the end-of-line.

The parameters are listed alphabetically in the manual. I discuss them here in topical order. Input parameters for SE can be roughly divided into the following categories: image information, background estimation, detection, photometry, catalogs and SE running parameters.

---

[1]Moral: input is inportant...

[2]If the idea of having *.sex* files littering your hard disk is a little too randy for you, SE will read in any ASCII text files it is given as a configuration file.

7.1 Image Information

SE gets the positional information from the FITS header but most of the following parameters must be specified.

GAIN is the ratio of the number of electrons to the number of ADU. The GAIN is dependent on the type of CCD you're using and the instrument in front of it (for instance the WF2 chip on the HST has a gain of 7).

How the number you use here, the effective gain, relates to the instrument gain is as follows:

Effective Gain	Magnitude zeropoint	Type of image
gain × total exposure time	zeropoint(1 sec)	input image is c/s
gain	zeropoint(1 sec) + 2.5 $\log_{10}$(exp. time)	sum of N frames
N×gain	zeropoint(1 sec) + 2.5 $\log_{10}$(av. exp. time)	average of N frames
2×N×gain/3	zeropoint(1 sec) + 2.5 $\log_{10}$(av. exp. time)	median of N frames

NOTE: There are different strategies you can follow using the GAIN and the zeropoints, see the SE strategies section.

The MAG_GAMMA is a relic of when this program was applied to photographic plates (scans thereof). SE did not RUN, in my case, without it specified, even though it does not use it while processing your nice CCD data. If you have the misfortune that your data is still scanned in photographic plate, then this is the slope of the response function of the emulsion used on the plates in question. Just put something in case of CCD data.

NOTE: SE will not run without MAG_GAMMA specified.

The DETECT_TYPE specifies what type of data SE is handling, scanned photo plates or CCD data. Even with DETECT_TYPE set to CCD, SE will still need that MAG_GAMMA.

MAG_ZEROPOINT is the zeropoint for the photometric measurements.

This is again different if you use counts-per-second images as opposed to total counts images. The counts-per-second images have the zeropoint specified by the instrument handbook (depends on filter, instrument and type of ccd used). But in the case of a total counts image is the handbook value plus the $2.5 log_{10}(exposuretime)$.

PIXEL_SCALE is again something you hopefully know before you started to run SE. Funny enough this is not read from the Fits header. So specify this! SE needs it *only* for the CLASS_STAR parameter (but still needs it).

SATUR_LEVEL is the limit for SE to start extrapolating to get the photometry. However as soon as you hit something as saturated as that you might want to see if there is another way to determine the flux from that object.

The SEEING_FWHM (Full Width at Half Maximum) is important for that separation between stars and galaxies. Again it is only used for the CLASS_STAR parameter. [3]

---

[3]Seeing is the blurring of the image as a result of atmospheric disturbances (cirrus clouds, turbulence etc.). An estimate of the seeing should be documented in either the header or the observation logs.

Parameter	Default	Type	Description
FLAG_IMAGE	flag.fits	strings ($n <= 4$)	File name(s) of the flag-image(s).
FLAG_TYPE	OR	keyword	Combination expression for the method for flags on the same object:
		OR	arithmetical OR,
		AND	arithmetical AND,
		MIN	minimum of all flag values,
		MAX	maximum of all flag values,
		MOST	most common flag value.
GAIN	-	float	Gain (conversion factor in e"=ADU) used for error estimates of CCD magnitudes.
DETECT_TYPE	CCD	keyword	Type of device that produced the image:
		CCD	linear detector like CCDs or NICMOS,
		PHOTO	photographic scan.
MAG_GAMMA	-	float	γ of the emulsion (slope of the response function). Takes effect in PHOTO mode only but **NEEDS** to be specified, even for CCD images.
MAG_ZEROPOINT	-	float	Zero-point offset to be applied to magnitudes.
PIXEL_SCALE	-	float	Pixel size in arcsec. (for surface brightness parameters, FWHM and star/ galaxy separation only).
SATUR_LEVEL	-	float	Pixel value above which it is considered saturated.
SEEING_FWHM	-	float	FWHM of stellar images in arcsec. This quantity is used only for the neural network star/galaxy separation as expressed in the CLASS_STAR output.

7.2 Background Estimation

SE estimated the background of the image as well as the RMS noise in that background, mapping both [4]. SE subtracts the estimated background from the photometry and uses the RMS to estimate errors. So the background is important in the rest of the SE run.

BACK_SIZE regulates the estimate. In an area of the BACK_SIZE, the mean and the σ of the distribution of pixel values is computed. Then the most deviant values are discarded and median and standard deviation σ are computed again. This is repeated until all the remaining pixel values are within mean $\pm 3\sigma$. If σ dropped with less then 20% per iteration, the field is considered not crowded.

The value for the background in the area is:

- the mean in the non-crowded case

- 2.5 × median - 1.5 × mean in the crowded case

Both the mean and the median are the ones computed in the last iteration. The mean is the average and the median is the average of all the values except the most extreme one.
The background map is an bi-cubic-spline interpolation over all the area's of size BACK_SIZE, after filtering. So obviously, the choice of BACK_SIZE is very important, too small and the background estimate will be partly object flux, too large and small scale variations cannot be taken into account.

NOTE: The BACK_SIZE parameter determines the background map. Estimate the average size of the objects in pixels and make sure the BACK_SIZE is larger then that.

But before the fit to the background values is done, the values can be smoothed: BACK_FILTERSIZE is the median filter for the background map.

---

[4]CHECKIMAGE_TYPE BACKGROUND and BACKGROUND_RMS if you want to inspect them. See also the checkimage section.

Effectively, you smooth the background image over this number of meshes to get rid of the deviations resulting from bright or extended objects.

BACK_TYPE is only used if you do not want SE to go off and estimate the background but use only one constant value supplied by you in BACK_VALUE.

The background computed above is first used in detection but the get accurate values for photometry, the background can be recomputed in an area centered around the object in question. To recompute, set BACK-PHOTO_TYPE to LOCAL and pick an BACKPHOTO_THICK to match your tastes.

NOTE: the RMS as determined from the BACKGROUND_RMS map will be used in more then just the photometry, the thresholds for detection and analysis can be dependent on it

NOTE: if you want to subtract the background and not have SE do this for you, set BACK_TYPE to MANUAL and BACK_VALUE to 0.0,0.0

Parameter	Default	Type	Description
BACK_SIZE	-	integers ($n <= 2$)	Size, or Width, Height (in pixels) of a background mesh.
BACK_FILTERSIZE	-	integers ($n <= 2$)	Size, or Width, Height (in background meshes) of the background-filtering mask.
BACK_TYPE	AUTO	keywords ($n <= 2$)	What background is subtracted from the images:
		AUTO	The internal interpolated background-map. In the manual it says "INTERNAL" here but the keyword is AUTO.
		MANUAL	A user-supplied constant value provided in BACK VALUE.
BACK_VALUE	0.0,0.0	floats ($n <= 2$)	in BACK TYPE MANUAL mode, the constant value to be subtracted from the images.
BACKPHOTO_THICK	24	integer	Thickness (in pixels) of the background LOCAL annulus.
BACKPHOTO_TYPE	GLOBAL	keyword	Background used to compute magnitudes:
		GLOBAL	taken directly from the background map,
		LOCAL	recomputed in a rectangular annulus around the object.

7.2.1 Weight Images

NOTE: I got this section wrong in earlier versions...

The individual pixels in the detection image can be given relative impor-

tance by using a weight for each of them. Different options are available: the background as determined by SE or an external weight image. If an external weight image is geven, it has to be specified what kind it is; a variance map or a rms map or a weight map, from which a variance map should be derived.

The weight for each pixel is derived as follows:

$$weight = \frac{1}{variance} = \frac{1}{rms^2}$$

WEIGHT_TYPE MAP_WEIGHT is directly taken as the weight, WEIGHT_TYPE MAP_VAR is inverted and WEIGHT_TYPE MAP_RMS is squared and inverted.

Reasons for using a weight image are various but to give you an idea: SE can ignore pieces of the image this way, use a flat-field in it's photometry or edited results from a previous run. [5]

Users of the stsdas *drizzle* package should check out the appendix on drizzle's weight image.

These are the parameters controlling it:

I can't really phantom why WEIGTH_GAIN isn't simply an option in WEIGHT_TYPE but it isn't. So there.

WEIGHT_IMAGE is the input parameter where the fits file is given which is to be used as weight map of the type defined by the WEIGHT_TYPE parameter.

With WEIGHT_TYPE set on BACKGROUND, the 'checkimage' (output image with CHECKIMAGE_TYPE set on BACKGROUND) of a previous run can be used. MAP_RMS can be for instance derived from known noise characteristics of the instrument or given by other programs used in data-reduction (for instance the 'drizzle' package for HST/WFPc data).

---

[5]Please note that I do not specify by what kind of black magic you obtained these weight images. They may be the result from a previous SE run but a ccd dark-frame might be used. Or perhaps your data-reduction scheme will produce a good estimate of the rms. To be used with caution!

Parameter	Default	Type	Description
WEIGHT_GAIN	Y	boolean	If true, weight maps are considered as gain maps.
WEIGHT_IMAGE	weight.fits	strings $(n <= 2)$	File name of the detection and measurement weightimage , respectively.
WEIGHT_TYPE	NONE	keywords $(n <= 2)$	Weighting scheme (for single image, or detection and measurement images):
		NONE	no weighting,
		BACKGROUND	variance-map derived from the image itself,
		MAP_RMS	variance-map derived from an external RMS-map,
		MAP_VAR	external variance-map,
		MAP_WEIGHT	variance-map derived from an external weight-map,

NOTE: WEIGHT_TYPE set to BACKGROUND does NOT mean that the weight image will be used for the background determination. (This is what I got wrong earlier on

7.3 Finding and Separating Objects

SE considers every pixel above a certain threshold (to be specified by YOU [6] , directly or indirectly) to be part of an object. The 'deblending' is the part where it figures out which pixels or parts of pixels belong to which objects.

7.3.1 Detection; Thresholds

The threshold parameters indicate the level from which SE should start treating pixels as if they were part of objects, determining parameters from them. There are three requirements for a candidate objects:

1. All the pixels are above the DETECT_THRESH.

---

[6]yes YOU!

2. All these pixels are adjacent to each other (either corners or sides in common).

3. There are more then the minimum a number of pixels (specified in DETECT_MINAREA).

ANALYSIS_THRESH is just the threshold for CLASS STAR and FWHM, all the other parameters are determined from the DETECT_THRESH.

NOTE: All the OTHER analysis (photometry and the like) is done with the DETECT_THRESH!

They can both be specified in three ways:

- In Surface Brightness: SBlimit,SBzeropoint i.e. DETECT_THRESH 23.5, 24

- In ADU's (set THRESH_TYPE ABSOLUTE) i.e. DETECT_THRESH 1.2

- Relative to background RMS (set THRESH_TYPE RELATIVE) i.e. DETECT_THRESH 1.2

The threshold in surface brightness mu ($mag/arcsec^2$) needs a calibration Zero-point ($mag/arcsec^2$ corresponding to 0 counts). Note that this can/will be different from the MAG_ZEROPOINT value. [7]

NOTE: This whole SB threshold stuff seems quite the thing until you realize SE just does this:

$$thresh = 10^{-\frac{SB_{limit} - SB_{zeropoint}}{2.5}}$$

With THRESH_TYPE set to ABSOLUTE, the threshold is set to the same number of ADUs across the image. If THRESH_TYPE set to RELATIVE (the default), the the threshold is that number of background RMS standard deviations above the background value. This is nice and flexible but sensitive to the background estimation!

The DETECT_MINAREA is the minimum number of pixels above the threshold required to be considered an object.

---

[7]For the HST instruments, the zeropoints can be obtained using CALCPHOT in the stsdas package in IRAF. Remember that these are dependent on instrument and filter.

Parameter	Default	Type	Description
THRESH_TYPE	RELATIVE	keywords ($n <= 2$)	Meaning of the DE-TECT THRESH and ANALYSIS_THRESH parameters :
		RELATIVE	scaling factor to the background RMS,
		ABSOLUTE	absolute level (in ADUs or in surface brightness).
ANALYSIS_THRESH	-	floats ($n <= 2$)	Threshold (in surface bright-ness) at which CLASS STAR and FWHM operate. 1 argument: relative to Background RMS. 2 arguments: mu ($mag/arcsec"^2$), Zero-point (mag).
DETECT_THRESH	-	floats ($n <= 2$)	Detection threshold. 1 argument: (ADUs or relative to Background RMS, see THRESH TYPE). 2 arguments: R (mag.arcsec" 2), Zero-point (mag).
DETECT_MINAREA	-	integer	Minimum number of pixels above threshold triggering detection.

7.3.2 Filtering

Before the detection of pixels above the threshold, there is the option of applying a filter. This filter essentially smooths the image. [8]

There are some advantages to applying a filter before detection. It may help detect faint, extended objects

However it may not be so helpful if your data is very crowded. There

---

[8]The photometry is still being done on the original image, don't worry.

are four types of filter to be fount in the *./sextractor2.2.2/config/* direc-
tory; Gaussian, Mexican hat, tophat and blokfunction of various sizes, all
normalized.

This is what the helpful README in this directory tells us:

Name	Description
default.conv	a small pyramidal function (fast)
gauss*.conv	a set of Gaussian functions, for seeing FWHMs between 1.5 and 5 pixels (best for faint object detection).
tophat*.conv	a set of "top-hat" functions. Use them to detect extended, low-surface brightness objects, with a very low THRESHOLD.
mexhat*.conv	"wavelets", producing a passband-filtering of the image, tuned to seeing FWHMs between 1.5 and 5 pixels. Useful in very crowded star fields, or in the vicinity of a nebula. WARNING: may need a high THRESHOLD!!
block_3x3.conv	a small "block" function (for rebinned images like those of the DeNIS survey).

The naming convention seems to be: name_seeingFWHM_size.conv. Both
the seeing FWHM and the size are in pixels. So depending on what you are
after, choose a filter and approximately your seeing.

NOTE: filter choice and threshold choice are interdependent!

Parameter	Default	Type	Description
FILTER	-	boolean	If true, filtering is applied to the data before extraction.
FILTER_NAME	-	string	Name and path of the file containing the filter definition.
FILTER_THRESH	-	floats ($n <= 2$)	Lower and higher thresholds (in back-ground standard deviations) for a pix-el to be considered in filtering (used for retina-filtering only).

7.3.3 Deblending; separating into different objects

Deblending is the part of SE where a decision is made whether or not a
group of adjacent pixels above DETECT_THRESH is a single object or not.
Suppose there is a little island of adjacent pixels above the threshold. It
is an object or maybe several really close next to each other. So how does

SE cut this up into different objects? First it defines a number of levels between the threshold and the maximum count in the object. This is set by the DEBLEND_NTHRESH parameter. The levels are spaced exponentially.

SE then constructs a 'tree' of the objects, branching every time there are pixels above a threshold separated by pixels below it (see figure). A branch is considered a different object provided:

1. The number of counts in the branch (A in the figure) is above a certain fraction of the total count in the entire 'island'.

2. There is at least one other branch (yep B!) above the same level that is also above this fraction.
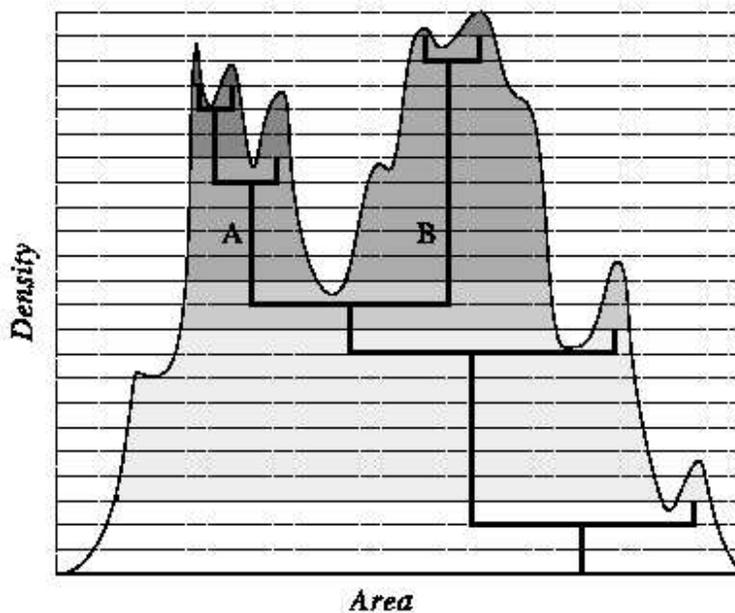
The fraction is defined in DEBLEND_MINCONT. [9]

Figure 7.1: Illustration of DEBLEND_NTHRESH and DEBLEND_MINCONT

---

[9]Note that 0.01 is 1% when defining this.

Parameter	Default	Type	Description
DEBLEND_MINCONT	-	float	Minimum contrast parameter for deblending.
DEBLEND_NTHRESH	-	integer	Number of deblending sub-thresholds.

7.3.4 Cleaning

There is the option to 'clean' the list of objects of artifacts due to bright objects (set CLEAN to YES). All the detections are checked to see if they would have been detected (i.e. exceeded the threshold etc.) if their neighbors were not there. To do this, the contributions of the neighboring objects has to be computed. An estimate is made from a moffat light profile. The Moffat profile is scaled and stretched to fit the neighbour's profile. The contribution to the object from the wings of the Moffat profile is then subtracted.

The Moffat profile looks like this:

$$\frac{I(r)}{I(0)} = \frac{1}{(1 + k \times r^2)^\beta}$$

The CLEAN_PARAM is the β parameter in the above formula.

NOTE: Decreasing CLEAN_PARAM yields brighter wings and more aggressive cleaning.

The value for the CLEAN_PARAM should be between 0.1 and 10.

NOTE: In earlier versions of SE, the Moffet profile was a Gaussian and the CLEAN_PARAM the stretch factor for the FWHM.

Cleaning would be more aggressive with a higher CLEAN_PARAM. This version of cleaning is explained in the original manual.

Parameter	Default	Type	Description
CLEAN	-	boolean	If true, a cleaning of the catalog is done before being written to disk.
CLEAN_PARAM	-	float	Efficiency of cleaning.

7.4   Influencing Photometry

After deblending the objects, SE performs astrometry (where stuff is), photometry (how bright stuff is) and geometric parameters (how stuff looks like). The astrometry cannot be influenced via input parameters (you just specify what kind of positions you want, easy). The photometry had a few input parameters associated with it; what to do with overlapping pixels, what is the zeropoint and how to apply apertures. To understand these, the AUTO photometry of SE has to be explained. The geometric parameters are mainly associated with the Kron (AUTO) photometry.

The GAIN and MAG_ZEROPOINT, have been discussed at the image characteristics section. Of course they are needed for the photometry. The GAIN to convert counts to flux and the MAG_ZEROPOINT for the calibration of the magnitude scale. Also the BACKPHOTO_THICK and BACK-PHOTO_TYPE give you influence on the way the background subtracted from the photometry is estimated (see the backgrond section).

There are five different approaches in SE's photometry; isophotal, isophotal-corrected, automatic, best estimate and aperture.

7.4.1   ISO

In the above you defined above what threshold SE should do it's photometry, with the estimated background as zeropoint. The pixels above this threshold constitute an isophotal area. The flux or magnitude determined from this (counts in pixels above threshold minus the background) is the *isophot* flux/magnitude. Apart from the threshold (DETECT_THRESH) and the background estimation, there is nothing to influence here.

7.4.2   ISOCOR

In real life however, objects rarely have all their flux within neat boundaries, some of the flux is in the "wings" of the profile. SE can do a crude correction for that, assuming a symmetric Gaussian profile for the object. This would be the *isophot-corrected* flux/magnitude. There is no parameter for you to influence this estimate.

7.4.3   AUTO

SE uses a flexible elliptical aperture around every detected object and measures all the flux inside that, described in (Kron, 1980). There are two

parameters regulating the elliptical apertures: PHOT_AUTOPARAMS and PHOT_AUTOAPERS.

The characteristic radius for the ellipse is:

$$r_1 = \frac{\Sigma r I(r)}{\Sigma I(r)}$$

From the objects second order moments, the ellipticity η and position angle θ are computed. The major and minor axes of the elliptical aperture are computed to be $k\dot{r}_1/\epsilon$ and $\epsilon k\dot{r}_1$ respectively. PHOT_AUTOPARAMS influences directly the estimation. The first is k factor mentioned above and the second is the minimum radius for a Kron ellipse. PHOT_AUTOAPERS are the minimum aperture diameters for the Kron photometry, estimation and measurement. These are circular! These are used in case the radius of the Kron aperture goes below the R_{min} specified in PHOT_AUTOPARAMS. The values in the default.sex are probably best.

7.4.4   BEST

With all this flexibility, you'd expect the Kron or AUTO photometry to be the best. However it can be influenced by nearby sources. Therefore there is a fourth option, MAG_BEST. This is usually equal to AUTO photometry but if the contribution of other sources exceeds 10%, it is ISOCOR.

NOTE: the characterization 'BEST' is a bit misleading. I was advised from several sides NOT to use this photometry. Both AUTO and ISOCOR are at least consistent.

7.4.5   APERTURES

Then there is the user specified apertures. The aperture se *diameters* are specified in pixels in the PHOT_APERTURES. The fluxes, magnitudes and errors in these are specified in FLUX_APER,MAG_APER, FLUXERR_APER and MAGERR_APER. Please note that upon inclusion of N apertures, these parameters are N columns each in the outputcatalog, corresponding to each specified aperture.

One of the geometric output-parameters is the half-light radius. The fraction of total light within this radius is specified in the PHOT_FLUXFRAC parameter.

ISO   **AUTO**
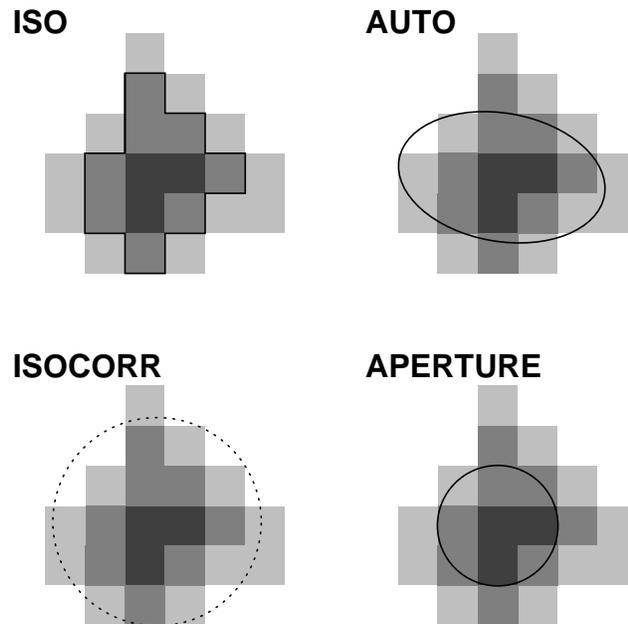
ISOCORR   **APERTURE**

Figure 7.2: Illustration of the different apertures possible; ISO, ISOCOR, AUTO and APER (user specified in PHOT_APERTURES

7.4.6   Masking Overlapping Objects

Now what if there are two objects overlapping each other? How to account for the overlapping pixels? This is handled by the MASK_TYPE parameter. NONE means that the counts in the overlap are simply added to the objects total. BLANK sets the overlapping pixels to zero. CORRECT, the default, replaces them with their counterparts symmetric to the objects' center. Best if you leave it at default. I'm just mentioning it out of completeness [10]

7.4.7   Fraction-of-light radii

SE has the option to put out radii containing a certain fraction of the light. (outputparameter FLUX_RADIUS) The default is 0.5 (the half light radius). PHOT_FLUXFRAC 0.2,0.5,0.9 will give three radii containing 20%, 50% and

---

[10]NONE might be useful to get the total flux from a large extended underlying structure with bright patches and no foreground stars.

90% of the light respectively.

Parameter	Default	Type	Description
GAIN	-	float	Gain (conversion factor in e"=ADU) used for error estimates of CCD magnitudes .
MAG_ZEROPOINT	-	float	Zero-point offset to be applied to magnitudes.
MASK_TYPE	CORRECT	keyword	Method of masking of neighbors for photometry:
		NONE	no masking,
		BLANK	put detected pixels belonging to neighbors to zero,
		CORRECT	replace by values of pixels symmetric with respect to the source center.
PHOT_APERTURES	-	floats (n ln 32)	Aperture diameters in pixels (used by MAG APER).
PHOT_AUTOPARAMS	-	floats (n = 2)	MAG AUTO controls: scaling parameter k of the 1st order moment, and minimum Rmin (in units of A and B).
PHOT_AUTOAPERS	0.0,0.0	floats (n = 2)	MAG AUTO minimum (circular) aperture diameters: estimation disk, and measurement disk.
PHOT_FLUXFRAC	0.5	floats (n <= 32)	Fraction of FLUX AUTO defining each element of the FLUX RADIUS vector.

7.5 SE Running

These inputparameter govern the way SE runs, if it should heed flags, how it should heed those, if and what to put in an outputimage, how much it should comment and how much memory it should use.

7.5.1 Flags

If pixels in your image should be flagged as unreliable or other, SE can use a flag image for this purpose.

Parameter	Default	Type	Description
FITS_UNSIGNED	N	boolean	Force 16-bit FITS input data to be interpreted as unsigned integers.
FLAG_IMAGE	flag.fits	strings ($n <= 4$)	File name(s) of the flagimage(s) .
FLAG_TYPE	OR	keyword	Combination method for flags on the same object:
		OR	arithmetical OR,
		AND	arithmetical AND,
		MIN	minimum of all flag values,
		MAX	maximum of all flag values,
		MOST	most common flag value.

7.5.2 Interpolation

If the data for pixels is missing, SE can interpolate. These parameters regulate the interpolation. Best kept at default. For some, the x and y gaps allowed are a bit wide (16 pixels after all, it is almost an entire object...). On the other hand, it allowd Sextractor to give you catalogs despite bad columns. Therefore do not set to zero.

Parameter	Default	Type	Description
INTERP_MAXXLAG	16	integers $(n <= 2)$	Maximum x gap (in pixels) allowed in interpolating the input image(s).
INTERP_MAXYLAG	16	integers $(n <= 2)$	Maximum y gap (in pixels) allowed in interpolating the input image(s).
INTERP_TYPE	ALL	keywords $(n <= 2)$	Interpolation method from the variance-map(s) (or weight-map(s)):
		NONE	no interpolation,
		VAR_ONLY	interpolate only the variance-map (detection threshold),
		ALL	interpolate both the variance-map and the image itself.

7.5.3   Memory, NNW and Comments

These are the parameters regulating the memory use of SE. To be honest, they are best kept at the values in the *./sextractor2.2.2/config/default.sex* file. [11]

There is to date only one neural network file and it's in the same directory (*default.nnw*) The VERBOSE_TYPE parameter regulates the amount of comments printed on the command line. It could possibly be instructive to run it with FULL once in a while.

---

[11]SE was programmed to do large images, even with limited memory and computing power. The only reason to change these defaults is when you get something of a stack overflow.

Parameter	Default	Type	Description
MEMORY_BUFSIZE	-	integer	Number of scanlines in the imagebuffer. Multiply by 4 the frame width to get equivalent memory space in bytes.
MEMORY_OBJSTACK	-	integer	Maximum number of objects that the objectstack can contain. Multiply by 300 to get equivalent memory space in bytes.
MEMORY_PIXSTACK	-	integer	Maximum number of pixels that the pixel-stack can contain. Multiply by 16 to 32 to get equivalent memory space in bytes.
STARNNW_NAME	-	string	Name of the file containing the neural network weights for star/galaxy separation.
VERBOSE_TYPE	NORMAL		
		keyword	How much SExtractor comments its operations:
		QUIET	run silently,
		NORMAL	display warnings and limited info concerning the work in progress,
		EXTRA_WARNINGS	
			like NORMAL, plus a few more warnings if necessary,
		FULL	display a more complete information and the principal parameters of all the objects extracted.

7.6 SE output settings

SE has two types of output. The catalogs with a whole range of characteristics of each of the detected objects and a outputs which allow you to compare SE estimates of background, apertures and objects with the real

data.

7.6.1   Catalog

The catalog is what you are running SE for! So in CATALOG_NAME ,
you specify the name of the output catalog. Again it's probably a good
idea to start straight away with a naming convention. (a .cat extension for
instance!)

The CATALOG_TYPE enables you to specify the type of outputcatalog.
Personally I prefer the ASCII_HEAD, as it allows me to read it in just about
anywhere and still tells me which parameters are listed. The nice thing
about the fits catalog is that all the inputparameter settings are saved in the
header.

**NOTE: the fits option can't handle array outputinformation
such as MAG_APER,FLUX_RADIUS if more then one value!** [12]

The ASCII_SKYCAT option for instance does not list all the parameters
before the actual catalog (like ASCII_HEAD) but puts the name of the
output parameter on top of the column in question.

And which parameters to list is specified in the file given to PARAME-
TERS_NAME.

---

[12]I have this on Ed's authority. I have NO idea how to display a fits table

Parameter	Default	Type	Description
CATALOG_NAME	-	string	Name of the output catalog. If the name "STDOUT" is given and CATALOG TYPE is set to ASCII, ASCII HEAD, or ASCII SKYCAT, the catalog will be piped to the standard output (stdout)
CATALOG_TYPE	-	keyword	Format of output catalog:
		ASCII	ASCII table; the simplest, but space and time consuming,
		ASCII_HEAD	as ASCII, preceded by a header containing information about the content,
		ASCII_SKYCAT	SkyCat ASCII format (WCS coordinates required),
		FITS_1.0	FITS format as in SExtractor 1,
		FITS_LDAC	FITS "LDAC" format (the original image header is copied).
PARAMETERS_NAME	-	string	The name of the file containing the list of parameters that will be computed and put in the catalog for each object.

7.6.2 ASSOC parameters

These are the parameters dealing with crosscorrolating two catalogs (one of whom is given in ASSOC_NAME, the other is created as SE runs). The cross correlation is controlled by two parameter: ASSOC_RADIUS and ASSOC_TYPE, the first governing the search radius and latter which objects gets selected if there are multiple candidates near the positions.

NOTE: ASSOC works only with pixelspositions (NOT RA and DEC!)

NOTE: Be aware of shifts between images when crosscorrellating two catalogs.

Find shifts beween images with imcentroid in IRAF for instance.

Parameter	Default	Type	Description
ASSOC_NAME	sky.list	string	Name of the AS-SOC ASCII file.
ASSOC_PARAMS	2,3,4	integers $(n \ <= \ 2,$ $n <= 3)$	Nos of the columns in the ASSOC file that will be used as coordinates and weight for cross-matching.
ASSOC_RADIUS	2.0	float	Search radius (in pixels) for ASSOC.
ASSOC_TYPE	MAG_SUM	keyword	Method for cross-matching in ASSOC:
		FIRST	keep values corresponding to the first match found,
		NEAREST	values corresponding to the nearest match found,
		MEAN	weighted-average values,
		MAG_MEAN	exponentially weighted average values,
		SUM	sum values,
		MAG_SUM	exponentially sum values,
		MIN	keep values corresponding to the match with minimum weight,
		MAX	keep values corresponding to the match with maximum weight.

Parameter	Default	Type	Description
ASSOCSELEC_TYPE	MATCHED	keyword	What sources are printed in the output catalog in case of ASSOC:
		ALL	all detections,
		MATCHED	only matched detections,
		- MATCHED	only detections that were not matched.
ASSOC_DATA	2,3,4	integers ($n <= 32$)	Numbers of the columns in the ASSOC file that will be copied to the catalog out-put.

7.6.3   The Checkimage

SE can output some of the maps used in intermediate steps. The names for the output fits files are specified in CHECKIMAGE_NAME. Keep a convention like your file_background.fits or your file_bgr.fits. A list of up to 16 can be given (separated by a comma).

The type of output files you want is defined in CHECKIMAGE_TYPE. As you can see, most of these have to do with the background estimation. Notable exceptions are the APERTURES and SEGMENTATION options. APERTURES is a good diagnostic on whether or not your threshold is right and the SEGMENTATION will show you if the objects are broken up too much or not. Load the original and this segmentation image into saotng and compare.

SEGMENTATION has another useful feature, the number in the catalog is given as the value to the isoarea in this image. Good for figuring out what is what from the catalogs.

Parameter	Default	Type	Description
CHECKIMAGE_NAME	check.fits	strings ($n <= 16$)	File name for each check-image .

| Parameter	Type	Description
CHECKIMAGE_TYPE	keywords ($n <= 16$)	Type of information to put in the check-images :
	NONE	no check-image,
	IDENTICAL	identical to input image (useful for converting formats),
	BACKGROUND	full-resolution interpolated background map,
	BACKGROUND_RMS	full-resolution interpolated background noise map,
	MINIBACKGROUND	low-resolution background map,
	MINIBACK_RMS	low-resolution background noise map,
	-BACKGROUND	background-subtracted image,
	FILTERED	background-subtracted filtered image (requires FILTER = Y),
	OBJECTS	detected objects,
	-OBJECTS	background-subtracted image with detected objects blanked,
	APERTURES	MAG APER and MAG AUTO integration limits,
	SEGMENTATION	display patches corresponding to pixels attributed to each object.
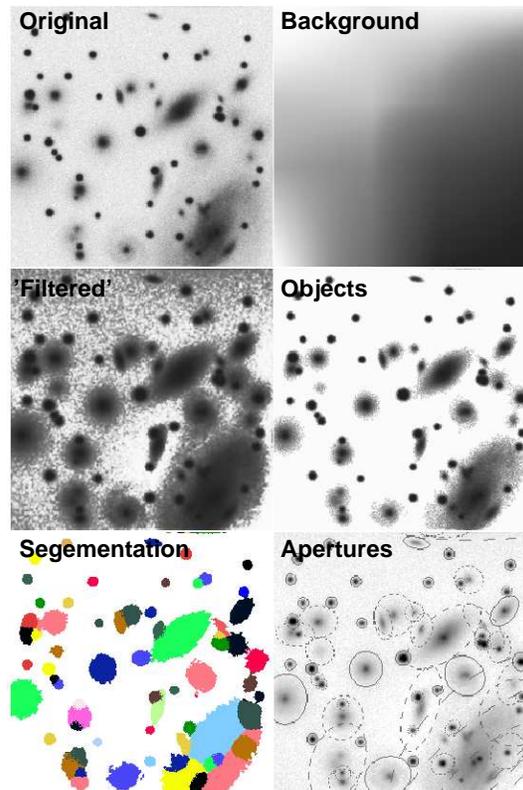
Figure 7.3: Illustration of the different checkimages possible. The original inputimage, the BACKGROUND image, the FILTERED image, the OB-JECTS image, the SEGMENTATION image and the APERTURES image. the contrast of the BACKGROUND image has been exaggerated.

Chapter 8

Output Parameters

The catalogs with output parameters is what the whole exercise is all about! You can finally start constructing your Hertzsprung-Russel diagrams or lensing shear fields or whatever. The parameters you want in your catalogs should be listed in the file you gave to PARAMETERS_NAME. Unless you keep using the same file for this, I really recommend using ASCII_HEAD type catalogs [1] The output catalogs will have a nice header with a list of all the parameters.

Parameters in the SE outputcatalog can be divided into geometric parameters ,photometric parameters, astrometric parameters and fitted parameters. Geometric parameters will tell you what shape the object is in (basically how the light of the object is distributed over the pixels of that object) and the photometric parameters tell you simply how much light there is. Astrometric parameters give the position of the object in the image, be it in pixels or other coordinates. Fitted parameters are calculated from fitting for instance a PSF to the data of the object. Most of these are still being developed.

There are a few that do not fall in any of these convenient categories; the catalog number, the flag parameters and the parameters associated with crosscorrellating catalogs.

---

[1] By setting the parameter CATALOG_TYPE to ASCII_HEAD in the configuration file.

Name	description	unit
NUMBER	Running object number	-
FLAGS	Extraction flags	-
IMAFLAGS_ISO	FLAG-image flags OR'ed over the iso. profile	-
NIMAFLAGS_ISO	# flagged pixels entering IMAFLAGS_ISO	-

8.1 Photometric Parameters

Photometric parameters are either flux or magnitude determined by SE. However SE has five different ways of determining these; isophotal, isophotal-corrected, automatic, best estimate and aperture.

These are discussed in the Photometry section in the input chapter. To recap:

ISO	Photometry derived from the counts above the threshold minus the background
ISOCOR	ISO photometry, corrected for loss as a Gaussian profile
AUTO	Photometry from flexible elliptical aperture. (?)
BEST	Choice between AUTO and ISOCOR. AUTO, except when influence from neighbors is more then 10%
APER	Photometry from circular, user specified (PHOT_APERTURES in the config file), apertures.

There are two other photometric parameters of interest: MU_MAX, the surface brightness of the brightest pixel and the MU_THRESHOLD, the the surface brightness corresponding to the threshold. This last parameter is good to inspect if the threshold is set with respect to the background RMS.
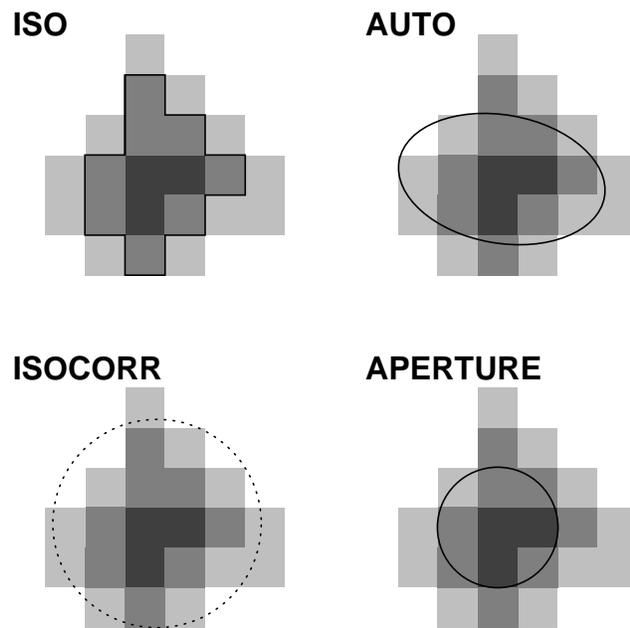
Figure 8.1: Illustration of the different apertures possible; ISO, ISOCOR, AUTO and APER (user specified in PHOT_APERTURES)

Name	description	unit
FLUX_ISO	Isophotal flux	count
FLUXERR_ISO	RMS error for isophotal flux	count
MAG_ISO	Isophotal magnitude	mag
MAGERR_ISO	RMS error for isophotal magnitude	mag
FLUX_ISOCOR	Corrected isophotal flux	count
FLUXERR_ISOCOR	RMS error for corrected isophotal flux	count
MAG_ISOCOR	Corrected isophotal magnitude	mag
MAGERR_ISOCOR	RMS error for corrected isophotal magnitude	mag
FLUX_AUTO	Flux within a Kron-like elliptical aperture	count
FLUXERR_AUTO	RMS error for AUTO flux	count
MAG_AUTO	Kron-like elliptical aperture magnitude	mag
MAGERR_AUTO	RMS error for AUTO magnitude	mag
FLUX_BEST	Best of FLUX_AUTO and FLUX_ISOCOR	count
FLUXERR_BEST	RMS error for BEST flux	count
MAG_BEST	Best of MAG_AUTO and MAG_ISOCOR	mag
MAGERR_BEST	RMS error for MAG_BEST	mag
FLUX_APER	Flux vector within fixed circular aperture(s)	count
FLUXERR_APER	RMS error vector for aperture flux(es)	count
MAG_APER	Fixed aperture magnitude vector	mag
MAGERR_APER	RMS error vector for fixed aperture mag.	mag
FLUX_PROFILE	Flux weighted by the FILTERed profile	count
FLUXERR_PROFILE	RMS error for PROFILE flux	count
MAG_PROFILE	Magnitude weighted by the FILTERed profile	mag
MAGERR_PROFILE	RMS error for MAG_PROFILE	mag

Name	description	unit
MU_THRESHOLD	Detection threshold above background	mag $\times$ arcsec-2
MU_MAX	Peak surface brightness above background	mag $\times$ arcsec^{-2}
BACKGROUND	Background at centroid position	counts
THRESHOLD	Detection threshold above background	counts

8.2 Astrometric Parameters

The astrometric parameters are simple enough; they tell you where the object is located. However which pixel do you take for the center of the objects? The maximum flux pixel? Or the barycenter? [2] Do you want it in RA and DEC or in prosaic pixels? All of these you should be able to specify from the list below.

NOTE: the x and y positions (with the exception of 'peak' values) are the barycenters of objects; the weighted mean position.

The definition for the barycenter of an object is:

$$X = \overline{x} = \frac{\Sigma I_i x_i}{\Sigma I_i}$$

$$Y = \overline{y} = \frac{\Sigma I_i y_i}{\Sigma I_i}$$

[3]

The minima and maximum x and y pixelvalues speak for themselves.

NOTE: all the x_i and I_i values in these formulae are the values from the pixels identified in the segmentation map as belonging to the object. It can be influenced by detection thresholds and segemntation settings

---

[2]The Barycenter is the flux-weighted average position. the first moment, SE computes all the second moments as well, see the geometric parameters section and chapter 9 in the manual v2.1.3.

[3]Yes this is straight from the sextrractor manual. All this stuff is quite complete in the manual v 2.1.3... Is there a problem?

Name	description	unit
XMIN_IMAGE	Minimum x-coordinate among detected pixels	pixel
YMIN_IMAGE	Minimum y-coordinate among detected pixels	pixel
XMAX_IMAGE	Maximum x-coordinate among detected pixels	pixel
YMAX_IMAGE	Maximum y-coordinate among detected pixels	pixel
YPEAK_IMAGE	y-coordinate of the brightest pixel	pixel
XPEAK_WORLD	World-x coordinate of the brightest pixel	deg
YPEAK_WORLD	World-y coordinate of the brightest pixel	deg
ALPHAPEAK_SKY	Right ascension of brightest pix (native)	deg
DELTAPEAK_SKY	Declination of brightest pix (native)	deg
ALPHAPEAK_J2000	Right ascension of brightest pix (J2000)	deg
DELTAPEAK_J2000	Declination of brightest pix (J2000)	deg
ALPHAPEAK_B1950	Right ascension of brightest pix (B1950)	deg
DELTAPEAK_B1950	Declination of brightest pix (B1950)	deg
X_IMAGE	Object position along x	pixel
Y_IMAGE	Object position along y	pixel
X_IMAGE_DBL	Object position along x (double precision)	pixel
Y_IMAGE_DBL	Object position along y (double precision)	pixel
X_WORLD	Barycenter position along world x axis	deg
Y_WORLD	Barycenter position along world y axis	deg
X_MAMA	Barycenter position along MAMA x axis	m^{-6}
Y_MAMA	Barycenter position along MAMA y axis	m^{-6}
ALPHA_SKY	Right ascension of barycenter (native)	deg
DELTA_SKY	Declination of barycenter (native)	deg
ALPHA_J2000	Right ascension of barycenter (J2000)	deg
DELTA_J2000	Declination of barycenter (J2000)	deg
ALPHA_B1950	Right ascension of barycenter (B1950)	deg
DELTA_B1950	Declination of barycenter (B1950)	deg

8.3 Geometric Parameters

Geometric parameters describe the shape and size of the object. SE computes the moments of an object and determines elliptical parameters from these. Both the moments and the elliptical parameters and derivatives from these can be included in the output. These parameters are treated much more extensive in chapter 9 of the manual (version 2.1.3).

8.3.1 Moments

The first order moments are the barycenters of course but the second order moments can also be given (see section 9.1.4 in the manual v2.1.3):

$$X2 = \overline{x^2} = \frac{\Sigma I_i x_i^2}{\Sigma I_i} - \overline{x}^2$$

$$XY = \overline{xy} = \frac{\Sigma I_i x_i y_i}{\Sigma I_i} - \overline{xy}$$

Name	description	unit
X2_IMAGE	Variance along x	pixel^2
Y2_IMAGE	Variance of position along y	pixel^2
XY_IMAGE	Covariance of position between x and y	pixel^2
X2_WORLD	Variance of position along X-WORLD (alpha)	deg^2
Y2_WORLD	Variance of position along Y-WORLD (delta)	deg^2
XY_WORLD	Covariance of position X-WORLD/Y-WORLD	deg^2

NOTE: By adding ERR+parameter, the error can be obtained as well.

8.3.2 Ellipse parameters

From these moments, the position angle, the minor and major axis of an ellipse can be derives (section 9.1.5 in the manual v2.1.3) and also a second way of describing this ellipse is given (sec 9.1.6):

The following definitions for these **computed** parameters are: (see for the derivation the v2.1.3 manual page 27 and 28)

$$A^2 = \frac{\overline{x^2} + \overline{x^2}}{2} + \sqrt{\left(\frac{\overline{x^2} - \overline{y^2}}{2}\right)^2 + \overline{xy}^2}$$

$$B^2 = \frac{\overline{x^2} + \overline{x^2}}{2} - \sqrt{\left(\frac{\overline{x^2} - \overline{y^2}}{2}\right)^2 + \overline{xy}^2}$$
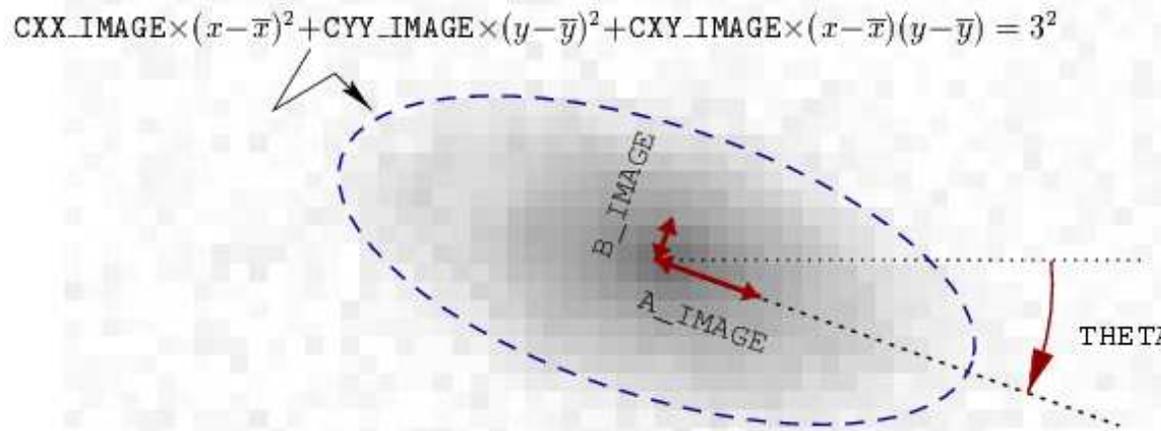
Figure   8.2:   Illustration   of   DEBLEND_NTHRESH   and DEBLEND_MINCONT

Name	description	unit
A_IMAGE	Profile RMS along major axis	pixel
B_IMAGE	Profile RMS along minor axis	pixel
THETA_IMAGE	Position angle (CCW/x) **counterclockwise!!!**	deg
A_WORLD	Profile RMS along major axis (world units)	deg
B_WORLD	Profile RMS along minor axis (world units)	deg
THETA_WORLD	Position angle (CCW/world-x)	deg
THETA_SKY	Position angle (east of north) (native)	deg
THETA_J2000	Position angle (east of north) (J2000)	deg
THETA_B1950	Position angle (east of north) (B1950)	deg
CXX_IMAGE	Cxx object ellipse parameter	pixel^{-2}
CYY_IMAGE	Cyy object ellipse parameter	pixel^{-2}
CXY_IMAGE	Cxy object ellipse parameter	pixel^{-2}
CXX_WORLD	Cxx object ellipse parameter (WORLD units)	deg^{-2}
CYY_WORLD	Cyy object ellipse parameter (WORLD units)	deg^{-2}
CXY_WORLD	Cxy object ellipse parameter (WORLD units)	deg^{-2}
ELONGATION	A_IMAGE/B_IMAGE	
ELLIPTICITY	1 - B_IMAGE/A_IMAGE	

Again the error can be put out as well by the ERR+parameter variable. Only the ELONGATION and the ELLIPTICITY parameters do not come with a error. Note that there is an extensive discussion of these in section 9.1.8 of the manual. [4]

8.3.3 Area Parameters

This can be done by the parameters describing the isophotes as fitted by SE or the conclusion of the neural network classification. To get an idea of the size of an object, reasonably independent of the brightness, the FWHM or the FLUX_RADIUS

SE divides an object up into 7 isophotes above the ANALYSIS_THRESH. The areas above the isophotes is fed to the neural network. These can be put into the catalog with the ISOn parameters.

---

[4]I'd be saying the exact same thing and typing in that many functions is a pain, even in LaTeX.

Name	description	unit
ISOAREA_WORLD	Isophotal area above Analysis threshold	deg^2
ISOAREAF_WORLD	Isophotal area (filtered) above Detection threshold	deg^2
ISO0	Isophotal area at level 0	$pixel^2$
ISO1	Isophotal area at level 1	$pixel^2$
ISO2	Isophotal area at level 2	$pixel^2$
ISO3	Isophotal area at level 3	$pixel^2$
ISO4	Isophotal area at level 4	$pixel^2$
ISO5	Isophotal area at level 5	$pixel^2$
ISO6	Isophotal area at level 6	$pixel^2$
ISO7	Isophotal area at level 7	$pixel^2$
FWHM_WORLD	FWHM assuming a Gaussian core	pixel
FWHM_WORLD	FWHM assuming a Gaussian core	deg
VIGNET	Pixel data around detection	count
VIGNET_SHIFT	Pixel data around detection corrected for shift	count
VECTOR_ASSOC	ASSOCiated parameter vector	
NUMBER_ASSOC	Number of ASSOCiated IDs	
THRESHOLDMAX	Maximum threshold possible for detection	count

8.4   Object classification

This is a section completely devoted to the CLASS_STAR parameter; SE's classification of the objects on the basis of a Neural Network Output. [5]

Name	description	unit
CLASS_STAR	S/G classifier output	none

It can have a value between 0 (galaxy, more to the point non-star) and 1 (star).

8.4.1   Input Dependency

Fortunately, you do not need to understand Neural Networks to use this but there are several input parameters which are directly linked to the

---

[5]If you DO want to know more on Neural Networks and how they are used for complex issues such as object classifiacation, I found the book 'An introduction to Neural Networks' by Kevin Gurney very useful.

CLASS_STAR parameter:

- PIXEL_SCALE: Pixel size in arcsec. (for surface brightness parameters, FWHM and star/ galaxy separation only).

- SEEING_FWHM: FWHM of stellar images in arcsec. This quantity is used only for the neural network star/galaxy separation as expressed in the CLASS_STAR output.

These are obvious. The NNW has to take in account the scale and blurring of the objects before judgment. These are inputparameters of the NNW. The ratio between the two is something you can play around with. (remember that SE does not use the pixelsscale for anything else...) But there are parameters where it depend more indirectly on:

- BACK_SIZE Size, or Width, Height (in pixels) of a background mesh.

- THRESH_TYPE Meaning of the DETECT THRESH and ANALYSIS_THRESH parameters :

    - RELATIVE scaling factor to the background RMS.
    - ABSOLUTE absolute level (in ADUs or in surface brightness).

- ANALYSIS_THRESH Threshold (in surface brightness) at which CLASS STAR and FWHM operate.

    - 1 argument: relative to Background RMS.
    - 2 arguments: mu ($mag/arcsec$"^{2}), Zero-point (mag).

Obviously the brightness level from which objects are considered influence heavily the classification. If only the top levels are considered, even clear galaxies might be classified as stars (the disk lies below the threshold) and if the threshold is too low, random noise can pass itself off as a faint galaxy.

Note: if the threshold is relative to the RMS, the BACK_SIZE is VERY important for the CLASS_STAR

- DEBLEND_MINCONT Minimum contrast parameter for de-blending.

- DEBLEND_NTHRESH Number of deblending sub-thresholds.

NOTE! if deblending is too course, a clump of stars can become a 'galaxy'.

NOTE! And if deblending is too picky, a single galaxy might be chopped up into several objects.

There is no single remedy but a DEBLEND_NTHRESH of 32 and a DEBLEND_MINCONT of order 0.01 are a good place to start.

HMMMM: If the ANALYSIS_THRESH is different from the DETECT_THRESH, the objects are NOT detected again. So carefull with making these different.

8.4.2   Reliability

This section is one big **Note:** of caution. The NNW classification by SE is NOT perfect and will break down at the lower magnitude end. Unfortunately it does not give another value for unreliable classification (42 for instance) but assigns a random value between 0 and 1. (see figure)
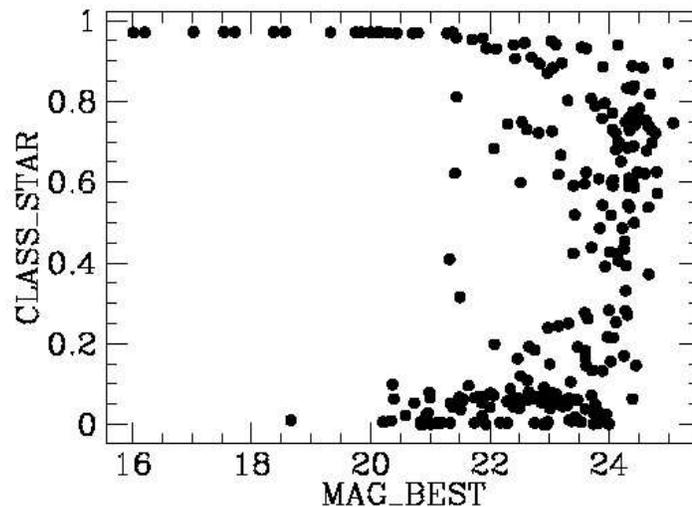
Figure 8.3: The dependence of CLASS_STAR on the luminosity of objects. The reliability clearly disappears at the lower end.

8.5 Fitted Parameters

There is a experimental section of SE which will hopefully become workable soon enough. This deals with the fitting of the Point Spread Function to stars and light profiles to extended objects.

8.5.1 PSF fitting

The point spread function of an instrument describes how the light from a point source is distributed over the detection element of a CCD. If you have a little fits file of a model of the PSF and you've figured out how to feed that to SE (if you do let me know...) then a part of SE known as PSFeX can fit this to every object. The following parameters should then available to you. As you can see some of these are either photometric, geometric or positional parameters. I've just list them here as this is still an experimental bit of SE but should be very useful later.

The PSF is also used for fitting the components of a galaxy in the next section.

Name	description	unit
XPSF_IMAGE	X coordinate from PSF-fitting	pixel
YPSF_IMAGE	Y coordinate from PSF-fitting	pixel
XPSF_WORLD	PSF position along world x axis	deg
YPSF_WORLD	PSF position along world y axis	deg
ALPHAPSF_SKY	Right ascension of the fitted PSF (native)	deg
DELTAPSF_SKY	Declination of the fitted PSF (native)	deg
ALPHAPSF_J2000	Right ascension of the fitted PSF (J2000)	deg
DELTAPSF_J2000	Declination of the fitted PSF (J2000)	deg
ALPHAPSF_B1950	Right ascension of the fitted PSF (B1950)	deg
DELTAPSF_B1950	Declination of the fitted PSF (B1950)	deg
FLUX_PSF	Flux from PSF-fitting	count
FLUXERR_PSF	RMS flux error for PSF-fitting	count
MAG_PSF	Magnitude from PSF-fitting	mag
MAGERR_PSF	RMS magnitude error from PSF-fitting	mag
NITER_PSF	Number of iterations for PSF-fitting	-
CHI2_PSF	Reduced chi2 from PSF-fitting	-

Name	description	unit
ERRX2PSF_IMAGE	Variance of PSF position along x	$pixel^2$
ERRY2PSF_IMAGE	Variance of PSF position along y	$pixel^2$
ERRXYPSF_IMAGE	Covariance of PSF position between x and y	$pixel^2$
ERRX2PSF_WORLD	Variance of PSF position along X-WORLD (alpha)	deg^2
ERRY2PSF_WORLD	Variance of PSF position along Y-WORLD (delta)	deg^2
ERRXYPSF_WORLD	Covariance of PSF position X-WORLD/Y-WORLD	deg^2
ERRCXXPSF_IMAGE	Cxx PSF error ellipse parameter	$pixel^{-2}$
ERRCYYPSF_IMAGE	Cyy PSF error ellipse parameter	$pixel^{-2}$
ERRCXYPSF_IMAGE	Cxy PSF error ellipse parameter	$pixel^{-2}$
ERRCXXPSF_WORLD	Cxx PSF error ellipse parameter (WORLD units)	deg^{-2}
ERRCYYPSF_WORLD	Cyy PSF error ellipse parameter (WORLD units)	deg^{-2}
ERRCXYPSF_WORLD	Cxy PSF error ellipse parameter (WORLD units)	deg^{-2}
ERRAPSF_IMAGE	PSF RMS position error along major axis	pixel
ERRBPSF_IMAGE	PSF RMS position error along minor axis	pixel
ERRTHTPSF_IMAGE	PSF error ellipse position angle (CCW/x)	deg
ERRAPSF_WORLD	World PSF RMS position error along major axis	pixel
ERRBPSF_WORLD	World PSF RMS position error along minor axis	pixel
ERRTHTPSF_WORLD	PSF error ellipse pos. angle (CCW/world-x)	deg
ERRTHTPSF_SKY	Native PSF error ellipse pos. angle (east of north)	deg
ERRTHTPSF_J2000	J2000 PSF error ellipse pos. angle (east of north)	deg
ERRTHTPSF_B1950	B1950 PSF error ellipse pos. angle (east of north)	deg

The reason you'd want a decent fit of the PSF to an object is to do really good photometry on it as you're now accounting for al that leaked light. Note that such a correction would only work if the object was originally a pointsource. This is fairly useless for extended objects but for that we have the galaxy fitting parameters.

8.5.2 Galaxy profile fitting

Another experimental bit of SE. As you might know, average radial light-profile of a galaxy can be described by fit. In the case of a spiral galaxy it is an exponential for the disk and a $(\frac{1}{r^4})$ profile for the bulge. The $(\frac{1}{r^4})$ profile fits an elliptical galaxy quite well. So if the galaxy is reasonably resolved, this new bit of SE can fit these profiles. It can give you the bulge disk ratio's and everything. provided

Name	description	unit
FLUX_GALFIT	Flux derived from the galaxy fit	count
FLUXERR GALFIT	RMS error for GALFIT flux	count
MAG_GALFIT	Magnitude derived from galaxy fit	mag
MAGERR_GALFIT	Magnitude error derived from galaxy fit	mag
ERROR_GALFIT	Reduced Chi-square error of the galaxy	fit
GALDANG_IMAGE	Galaxy disk position angle from the galaxy fit	deg
GALDSCALE_IMAGE	Galaxy disk-scale from the galaxy fit	pixel
GALDASPEC_IMAGE	Galaxy disk aspect ratio from the galaxy fit	
GALDE1_IMAGE	Galaxy disk ellipticity nr1 from the galaxy fit	
GALDE2_IMAGE	Galaxy disk ellipticity nr2 from the galaxy fit	
GALBRATIO_IMAGE	Galaxy bulge ratio from the galaxy fit	
GALBANG_IMAGE	Galaxy bulge position angle from the galaxy fit	deg
GALBSCALE_IMAGE	Galaxy bulge-scale from the galaxy fit	pixel
GALBASPEC_IMAGE	Galaxy bulge aspect ratio from the galaxy fit	

8.6   Principle Component

Again these are related to the psf fitting SE is -hopefully- capable of in the near [6] future.

Name	description	unit
X2PC_IMAGE	PC variance along x	pixel^2
Y2PC_IMAGE	PC variance along y	pixel^2
XYPC_IMAGE	PC covariance between x and y	pixel^2
APC_IMAGE	PC profile RMS along major axis	pixel
BPC_IMAGE	PC profile RMS along minor axis	pixel
THETAPC_IMAGE	PC position angle (CCW/x)	deg
PC	Principal components	

---

[6]Meaning: 'possible in your lifetime'.

Chapter 9

Strategies for SE use

"I have a cunning plan, sir" Pvt. Baldric

This section is reserved for some lecturing on what I found were good tricks to use with SE. There are some tricks you can do to extend the depth of your exposures, to make it easier to use SE on a batch of objects or tune your detections more to your needs.

9.1 Image types to use?

There are several options open to you for the image type. It is quite usual for astronomical images to be the result of stacked exposures. There are several ways to stack. The interaction between the type of addition and the photometric inputparameters is discussed here.

You may want to ask yourself whether you want to use the original integrated image with the total counts of every object or to use these divided by the exposure time, essentially counts-per-second images. Using counts-per-second images is not as dumb as it sounds. By using counts-per-second images, the dynamic range is the same for images with wildly varying exposure times. When examining them, it makes them much easier to compare. The signal-to-noise ratio (S/N) is of course still the same.

In the case of counts-per-second images, you can determine the MAG_ZEROPOINT for an 1 sec exposure and put this in the parameter file. Only the GAIN varies with every different exposure. The value for the GAIN in this case is

Effective Gain	Magnitude zeropoint	Type of image
gain $\times$ total exposure time	zeropoint(1 sec)	input image is c/s
gain	zeropoint(1 sec) + 2.5 $\log_{10}$(exp. time)	sum of N frames
N$\times$gain	zeropoint(1 sec) + 2.5 $\log_{10}$(av. exp. time)	average of N frames
2$\times$N$\times$gain/3	zeropoint(1 sec) + 2.5 $\log_{10}$(av. exp. time)	median of N frames

the ccd gain times the exposure time. [1]

9.1.1   Thresholds

This last sheme works best when your thresholds are RMS noise related. Otherwise you'd have to modify the thresholds anyway to account for the signal-to-noise. The zeropoints of the thresholds remain the same for counts-per-second images and integrated images. This can be *very* confusing, using the zeropoint-for-one-sec-exposure in the MAG_ZEROPOINT parameter and the zeropoint-for-one-sec-exposure + 2.5log(exptime) as the zeropoint for the detection/analyse threshold.

9.2   How to get faint objects?

Faint objects are the ahrdest and often the most interesting objects in you image. So how to get them? There are tricks for the detection images, the thresholds and the filters you use.

9.2.1   Different Detection Images

There are several strategies employed to detect and classify sources, using Source Extractor. Source Extractor has the in-built ability to detect sources in one image and subsequently do photometry on the found apertures in another image. This has the advantage that the photometry on an object

---

[1] As you've probably guessed I used counts-per-second images. It's personal preference but it's much easier to run SE in batches if the only thing you have to change is the GAIN. The zeropoint changes as soon you change filter of use a different ccd in an array like changing from WF2 to WF3 of the Wide Field camera on Hubble.

has the same apertures in all bands and the catalogs need not be matched (the numbering for all the catalogs is the same). Many schemes use this ability.

"Meta"-images

One idea is to add data out of two or several bands and do the detection there while doing the photometry in the actual data. The image where the detection is done is referred to as a "meta-image" (i.e. (Williams et al., 1996) used an I+V image to detect sources in the HDF). This is quite widespread use; you can use a median, mean or a stacked images. Just beware of differences in PSF when making these images. [2]

The advantages are that apertures can be determined more accurate for objects, espescially faint ones. However, strucural parameters (all those computed from the object's moments) are determined for the meta-image. These may not always be very representative of your objects.

Maximum likelyhood images

Another approach using multi-band images is to construct a χ^2 image and do the detection on these. (Szalay et al., 1999a) introduced this technique on the HDF. However this method appears to be most successful in the truly multiband (more then three at least) exposures.

Both of these are aimed at getting as faint as possible sources.

Optimal image

Most commonly, the band where your target objects are the brightest (i.e. the reddest band available for field galaxies) or the band with the longest exposure is used as the detection image and the apertures of that band are used for the photometry in all the other bands. The detection of background galaxies is done in the band with the longest wavelength available is used on the premises that background galaxies appear red.

Alternatively SE can be run on each band separately and the catalogs are subsequently matches by position. Fixed aperture photometry, corrected for the PSF can then be used to avoid differences due to different apertures. This had obvious drawbacks for extended or faint sources; you don't know how well the apertures match.

---

[2]The PSF of the resulting image should be about that of the worst PSF component. Some convolution may be needed.

9.2.2   Thresholds and Filters

Again the settings within sextractor can be optimized for faint objects. The threshold can be set as low as you want it but a DETECT_THRESH of 1 σ RMS above the background is probably the best you can do. However how that background is determined is very imposrtant. Depending on your images and objects, changing the BACK_SIZE background mesh size can improve your detection of faint stuff, especially near bright objects (v hard in any case). Which brings us to the CLEAN option you might want to turn off so you can better detect faint stuff near brighter things.

And some of the filters used to smoothe the image before detection are influential in the faint sources detection. Have look at the filtering section earlier. Usually a Gauss approximating the PSF in size is used.

NOTE!; whatever your settings, MAG_AUTO is the best estimate of the magnitude but it still underestimates by as much as several tens of magnitude for faint objects. [3]

Best way to figure this out is to inject you images with simulated objects of known magnitude and check their magnitude in the SE catalog.

9.3   How to get good colours of objects?

Colours of objects found in surveys are a very important tool for astronomers. So having a good strategy how to get the colours from SE catalogs is probably a good idea. There are several ways to determine these. Do you use a detection image and photometry image(s) or use values from several independent catalogs? Which magnitudes to use;ISO, ISOCOR, MAG_AUTO, MAG_APER or something else?

9.3.1   separate detection and photometry images?

SE has this ability to use the apertures from one detection image in the other, photometry image. This should give accurate colours as the apertures the same right? Some things to consider:

- The images better be aligned to the pixel. Small shifts can result in dramatically different colours.

---

[3]I have not experienced this myself but it was reported from several sides, espescially for faint galaxies. Some of the flux simply still is from outside the kron radius. The correction did not look very linear.

- The PSF's and seeing better be similar (i.e. the pixels in the detection image are indeed the same part of the object in the photometry image)

9.3.2   which output to use?

There is only two to consider MAG_ISO and MAG_APER, the others are too dependent on the SE settings in the detection.

MAG_ISO

The best aperture for colours is the MAG_ISO. This gives the flux in an as big an aperture that will fit in the object, will be roughly the same shape as the object and when using separate detection and photometry images, these will give you the most accurate colours. This of course does not hold of you crosscorellate catalogs.

MAG_APER

MAG_APER seems ideal for the colour of objects; either used in separate detection and photometry images or from crosscorrellated catalogs, the apertures are user specified and comparable over the bands used (you can correct the radii for instance with seeing...) BUT! there are a few things to consider with fixed apertures:

- crowding; does your aperture overlap with another object?

- objects smaller than the aperture will have less reliable colours

- Still, are they aligned? Is this the same part of the object?

- In the case of small apertures, is the colour of the center of the object the same as the for the whole?

- PSF aperture corrected?

But you can take these in account. The only (dis)advantage [4] MAG_APER has is that really the fact that it is fixed for every object. However it is not so dependent on the detection by sextractor.

---

[4]good bad? you call

In conclusion: with detection and photometry images, use MAG_ISO, with separate detections, use MAG_APER. Ponder which ones would be best for you type of objects.

9.4   Finding your objects of interest

This is a very broad query. Literate searches and some intense staring at typical objects should give you an idea what the characteristic values of SE output of your objects are. Sometimes a specialised parameter added to SE might be able to help or a ratio of SE parameters (aperture fluxes for instance). Best thing to do is get a trainingset and start plotting. FWHM, ELLIPTICITY, MAG_APER(S) FLUX_RADIUS and concentration (MU_MAX over MAG_AUTO is a reasonable indication but there are more) are good places to start.

And of course colours always help.

9.5   Strategies to find galaxies in crowded fields

To increase SE's ability to find field galaxies in crowded fields, several strategies may be considered.

(González et al., 1998) used B-I images of NGC 3664 to detect sources instead of the I images. This has the advantage that stars can be negated somewhat, leaving only objects of a severely differing color. The disadvantage is the increase in the noise, increasing the chances of spurious detections. And there actually has to be a B band exposure.

An alternative approach is to remove the stars, using DAOPHOT. It is honed to find and model stars in crowded fields and there is the option to produce an image with the model subtracted. Again, this could increase the number of spurious detections because of noise.

Alternatively SE can simply be run on the I images and photometry can be done on the V image. This additional information can be used to filter the catalogs for actual galaxies. As an additional filter, an visual inspection of the data can be employed.

A technique to enhance faint extended sources is called unsharp masking. Originally developed for photographic plate, ccd images would be smoothed and the smoothed version would be subtracted from the original. This technique works fine for filamentary structures like shells and tidal tails but is

unsuited for finding galaxies in crowded fields. (SE does however smooth the image slightly and then does the detection)

Metaimages, the addition of two band, suitably weighed, as detection images is another popular option. Again this opens the possibility of many spurious detections.

Chapter 10

Available Packages

Maybe it's not a good idea to advertise the *competing* software but SE might not exactly be what you need. Better if you figure this early. At the moment there are several programs and approaches in use that attempt to detect and classify objects; Sextractor (Bertin and Arnouts, 1996), FOCAS (Jarvis and Tyson, 1979; Jarvis and Tyson, 1981), DAOPHOT (Stetson, 1987) and self organising maps (SOM),

All of these have been tried and tested in relatively uncrowded fields and are expected to break down to some extent in crowded fields. The notable exception is DAOPHOT which is honed on crowded stellar fields. However it is focused on accurate stellar photometry, not on detecting extended objects.

FOCAS is widely rumored not to work, especially in crowded environments but still popular. I personally haven't used it but SE is somewhat more recent so I chose SE over FOCAS.

Self Organizing Maps (SOM) are still in an experimental phase and have proved themselves superior to SE in classifying galaxies by using an additional neural network. However they are not distributed in a package. If you *really need* detailed classification of objects (your thesis hangs on it being Sab galaxies) then you'd better contact the authors of papers on SOMs or write something yourself. But if you can think up a reasonable parameter that will help you classify, you can always define your own in SE (see

This leaves however SE as the current program of choice, almost by default.

Chapter 11

SE use in the Literature

In the literature SE is used for a myriad of problems. In this section the usage for detecting galaxies in the literature is discussed. It is not up to date completely, lots of people continue to use SE. Keep a look out in the literature.

(Casertano et al., 2000a) used WFPC2 (U B V I) and a V+I detection image of the HDF-S with the following settings: THRESH 0.65 σ,MIN_AREA 16 (0.05" pixelsize) and MIN_CONTAST 0.03 . They used the weight maps made by drizzle for RMS background estimate.

(Brown et al., 2000) used SE on photographic plates (UBRI) and made separate catalogs for each band. Classified as star if class > 0.7 in three bands or class > 0.75 in two bands or class > 0.85 in one band.

(Rodighiero et al., 2000) used SE on the HDF-N plus JHK band. detection was in the K-band excluding Vaucouleur-profiles. FLUX_RADIUS was used in photometry correction.

(Smail et al., 2001) used SE on WFPC2 BVI with THRESH 1.5 σ 21.2 mag arcsec^{-2} and MIN_AREA 10. Rejection criterion: k ¡ 19 within 1-2 half light radii.

(Kalirai et al., 2001) CFHT BVR data. Has a plot of magnitude - stellarity with a cut for the best star-galaxy separation. Mention of PSFex mentioned for correction of photometry.

(Williams, 2001) run SE on KeckII data (BVI) to V = 26.5. All objects with V¡24 and CLASS ¡ 0.2 is a galaxy.

(Rodighiero et al., 2001) run SE same as (Franceschini et al., 1998). SE is used to perform simultaneous slicing. From this, the PSF and the profiles

for all objects are constructed. Galaxies are then selected as those objects resembling the Veaucouleurs profile convolved with the PSF... This is done for morphological classification.

(Trujillo et al., 2001) used BVRI data from the NOT. I band detection with LOWTHRESHOLD 1.5σ MINAREA 4. 2.5 kron radius photometry.

(Väisänen et al., 2000) used SE on J,K data from the 1.2 m telescope. Used MAG_BEST and eyeball identification.

(Zabludoff and Mulchaey, 2000) used SE in a standard config on rosat's PSPC considering everything with CLASS > 0.5 as a galaxy. Checked with isophotal area. MAG_BEST photometry.

(Kambas et al., 2000) used ground based data with MINAREA 5 (2".3 pixelsize) and THRESH 1 σ. The SE catalog was then filtered for flagged objects. VLSB sample selection: $\mu_0 > 23$ R $\alpha > 3$" isophotal area > 148 arcsec^2.

(Hogg et al., 2000) used SE separately on UGR and K images. the FWHM of the smoothing Gauss was picked equal to the seeing. THRESH 1.2 σ and MIN_CONTAST 0.01 There was no Star/Galaxy separation.

(Volonteri et al., 2000a) used SE on the data of HDF (UBVI) south, in separate detections. The DEBLEND_MINCON 0.01 DEBLEND_NTHRESH 32 smoothing Gauss 0.16 arcsec and MINAREA is equal to the seeing disk (!?!) THRESH 1.34. Photometry was ISOPHOT_CORR or aperture, depending on diameter ISO (1.2 arcsec limit). Stars have class > 0.9 and I < 22

(Volonteri et al., 2000b) used SE separately on the UVBI data of the HDF-S and later on a 'meta-image' of all filters combined. with THRESH 1.34 σ and MIN_AREA 13 (seeing disk of $\approx$ arcsec^2. Photometry was either ISOPHOT or aperture depending on isophotal diameter. The limit was 1.2 arcsec. To estimate the *number* of spurious sources, the image was inverted and SE was run again on this image detecting only spurious sources.

(Sowards-Emmerd et al., 2000) give no details on their use of SE except that UBVRI detections were done separate and photometry was done on 20" apertures.

(Lubin et al., 2000) used a χ^2 image for the detection. This was made from three bands (BRi)using the Hale 5m telescope.

(Szalay et al., 1999b) proposed the χ^2-images as the detection image. Seems only applicable when there is true multicolor information.

(Broadhurst and Bouwens, 2000) give absolutely no details except that stars are badly fit by a redshifted galaxy spectra (Well DUH!)

(Castander and Lamb, 1999)used SE to find candidates for visual inspection.

(Gebhardt and Kissler-Patig, 1999) used SE to search for Globular clusters in HST (VI) data. Globular clusters are those objects with an mag_err < 0.1, V-I between 0 and 2, ellipticity < 0.5, FWHM between 1 and 4 pixels and 'nonstar' classification. They estimate a constant contamination from background galaxies.

(Poli et al., 1999) use a 'meta-image' from BVRI images. UBVRIJK photometry was determined from I=23.25 isophotes and 2".2 and 5" apertures. Half light radius, z and Magnitude relations. Not very useful for galaxy id however.

(Treu and Stiavelli, 1999) used SE on NICMOS, WFPC and ground data. They distinguished between resolved and unresolved object by comparing with stellar FWHM's. The limiting magnitude was the bin where all objects had a MAGERR_BEST < 0.15.

(Menanteau et al., 1999) used SE on HST archival data for spheroidal galaxies. They used both visual identification and the concentration and asymmetry parameter from (Abraham et al., ; ?). They plot how to separate spirals from ellipticals using these parameters.

(Hashimoto and Oemler, 1999; Hashimoto et al., 1998) used this concentration parameter to discern between early and late type galaxies.

(Simard et al., 1999) used Source Extractor on the Groth survey strip, THRESH 1.5 σ, MIN_AREA 10 and then fitting the light profiles of found galaxies.

(Nonino et al., 1999) describe the SE program in detail, especially the detection process. The star galaxy separation is less stringent and said to favor complete star catalogs at class > 0.5 and more complete galaxy catalogs at class < 0.75. They present a flag list of SE. Parameters used:BACK_SIZE 64 (17"), THRESH 0.6 σ. They use RMS images as weight maps (made by Weight watchers routine). MAG_AUTO is preferred and it is explained why. Good paper on the behavior of SE.

(van Dokkum et al., 2000) touches only lightly on the use of SE but describes a CLEANing method to improve the resolution of HST/WFPC data.

The study of high-z clusters by (Oke et al., 1998) used FOCAS, SE and MDS 'find' algorithm. SE is used on the K' band images from the IRIM camera at the 4m Kitt peak. THRESH 1.5 σ, ($\mu_{K'} = 22.2$ arcsec^2), 7x7 top-hat filter, MIN_AREA 35(0".15) The authors conclude that FOCAS and SE do not show significant differences in photometry, astrometry of

classification).

(Hilker et al., 1999) present ground-based data and use SE in the V-band.. The background mesh is either 56x56 or 128x128 FWHM of the convolving gauss is either 1".5 or 2" MIN_AREA 5. The CLASS identification is taken to be accurate up to V < 21. The limit was taken to be 0.35 up to V = 22. Color is used to identify cluster membership.

(Smail et al., 1998) naively used the MAG_BEST but used 1" diameter apertures on WFPC data and 3" on ground based data for galaxy colors. In the cases where Iband data was not available R band data was converted assuming R-I $\approx 0.5 \pm 0.2$

(Teplitz et al., 1998) used SE on NICMOS data with MIN_AREA 6 and used the FWHM of object, provided the were bright enough for a reliable estimate, to resolve stars from galaxies. All fainter objects were assumed to be galaxies.

(Marleau and Simard, 1998) use SE for the initial detection of galaxies in the HDF and then determine a string of parameters with GIM2D to quantify the morphology. THRESH 1.5σ, MIN_AREA 30, DEBLEND_MINCON 0.001.

(Yan et al., 1998) used NICMOS data FWHM 0.3" and a 2σ detection and a 1σ analysis threshold. Plot of half-light radius vs H magnitude with stars and galaxies.

(Brown et al., 2001) analyze UBRI data from photographic plates from the SGP and UK Schmidt field. No details.

(Fasano and Filippi, 1998) used SE on the HDF to find early type galaxies. THRESH 1.3 σ. The selectionlimits for a galaxy were: $V_{606}(STMAG) < 26.5 \quad N_{pix} \geq 200 \quad CLASS \leq 0.6$. (Fasano et al., 1998) used this data to establish a SB- effective radius relation for the early type galaxies. Note that they chose to use the V band for identification due to better S/N.

(?) used their own program to make a survey of southern hemisphere faint galaxies. More information then surmisable here.

(Hogg et al., 1997) use SE on U and R band images. No specifics are given except that the detection was done on images smoothed with the PSF.

(Bertin and Dennefeld, 1997) used SE on digitized Schmidt plates. It contains a good description of the usage of SE on photographic plates and the particular problems encountered.

(Gardner et al., 1996) use SE on BVIand K band ground based images. They use SE on all of them and note that I-K and B-I color are good separators for star galaxy for all except the bluest objects. All identified galaxies up to certain limits were confirmed by eye.

(Lanzetta et al., 1996) use SE on HST/HDF images. Detection is in the I band. The FWHM smoothing was 0.12 arcsec (approx width PSF) to aid detection of faint sources. Finally there is a reference as to *why* you should bother to smooth again with approximately the PSF: (Irwin, 1985).MIN_AREA 10 THRESH 1.4 σ and surprisingly a CLEAN_PARAM pf 2.0 to get rid of closely-packed objects (taken to be one object) BACK_SIZE was an unusual 41x41 pixels. As detection was done in the I band image, photometry was done in dual mode for the rest of the bands.

(Smail et al., 1997) produced catalogs on 10 clusters images by HST. Detections as in the reddest band available (either F702W or F814W). The modified SE code to produce the concentration index introduced by (Abraham et al., 1994; ?) and the contrast index which measures the fraction of light in the brightest 30% of its pixels. Object were however checked visually. THRESH 1.3 σ is the only parameter mentioned.

Chapter 12

SE parameter additions

It's also discussed in the manual but there is a possibility that the user can define his/her own output parameters in SE. In order to do that, you need to do the following things: First ask yourself if you can't do it with ANY of the parameters above or combinations thereof. After all there is a lot of them and it a lesser headache.

If not, then define what this parameter should be and then modify the analyze.c, types.h and param.h as follows: Define your parameter after the definition of the FWHM function in analyse.c (after line 371 in version 2.2.2). Define the parameter name in types.h and param.h like so:

In types.h

```
  float           conc;                                 /* IRS concentration index */
```

and in param.h:

```
  {"CONCENTRATION", "Abraham concentration parameter",
        &outobj.conc, H_FLOAT, T_FLOAT, "%8.3f",""},
```

This way, if you include CONCENTRATION in the file given to PARAMETERS_NAME, then the concentration will be calculated and put in the value 'conc'.

This parameter is the concentration parameter as defined in (Abraham et al., 1994; Abraham et al.,) and previously implemented by Ian Smail in (?; ?). I've modified it so that it works in version 2.2.2.

The contrast and asymmetry parameters are given as examples later on.

```
/* Abrahams concentration index calculated - IRS/BWH */
/* fraction of light in central 30% of the objects area, measured in an ellip
if (FLAG(obj.conc)) # check if concentration is in the PARAMETERS_NAME file
  {
  double xm,ym,dx,dy,rv,cv,amp; # local variables
  double AA,BB,CC,DD,EE,FF,sintheta,costheta ;

  xm = obj->mx; # get the median x position from the object struct
  ym = obj->my; # get the median y position from the object struct
  amp = tv/(2*PI*obj->a*obj->b*obj->abcor);
  rv = cv = 0.0;
  for (pixt=pixel+obj->firstpix;pixt>=pixel;pixt=pixel+PLIST(pixt,nextpix))
    # run through all the pixels in the object struct
    {
dx = PLIST(pixt,x) - xm; # calculate dx and dy
dy = PLIST(pixt,y) - ym;
pix = PLIST(pixt,value)<prefs.satur_level?
  PLIST(pixt,value)
  : amp * exp(-0.5*(obj->cxx*dx*dx + obj->cyy*dy*dy + obj->cxy*dx*dy)
    /obj->abcor);

costheta= cos(obj->theta*PI/180.) ;
sintheta= sin(obj->theta*PI/180.) ;
EE = (obj->b>0.0)?sqrt(obj->npix*obj->a/(PI*obj->b)):1.0;
FF = (obj->a>0.0)?sqrt(obj->npix*obj->b/(PI*obj->a)):1.0;

AA= costheta*costheta/EE + sintheta*sintheta/FF ;
BB= CC= sintheta*costheta*(1.0/EE - 1.0/FF) ;
DD= + sintheta*sintheta/EE + costheta*costheta/FF ;
if ((AA*dx+BB*dy)*(AA*dx+BB*dy)+(CC*dx+DD*dy)*(CC*dx+DD*dy) < 0.09)
  cv += pix;
if ((AA*dx+BB*dy)*(AA*dx+BB*dy)+(CC*dx+DD*dy)*(CC*dx+DD*dy) < 1.00)
  rv += pix;
    }
  obj->conc = (rv>0.0)? (cv/rv): 99.0;
  }
```

Chapter 13

Acknowledgments

I'd like to thank Ian Smail, Anton Koekemoer, Ed Smith, Emmanuel Bertin, Harry Ferguson, Roza Gonzalez and Ron Allen.

Appendix A

'Drizzle' and RMS weight images

Suppose you're a HST/WFPc user and you want to use SE on the data. The program 'drizzle' which combines a series of exposures produces also weight maps of the 'drizzling'. These weight maps can be used as MAP_RMS maps if processed as follows: $weightmap = \frac{F_A}{sqrt(drizzle_weight_map)}$ with F_A a certain correction parameter to get pure, uncorrelated RMS noise. This value depends on the area over which the noise is determined. (Casertano et al., 2000b) estimated this in their appendix and came to this: If the pixfrac of drizzle is p and the scale of the output pixels is s (0.5 half of the original pixels) then the ratio between uncorrelated noise and the drizzle map value is:

$$sqrt(F_A) = (\frac{s}{p}(1 - \frac{1}{3}\frac{s}{p}) \ , \ if \ s < p$$

$$sqrt(F_A) = 1 - \frac{1}{3}s/p) \ , \ if \ s > p$$

If you want to use this trick, please go over this bit in (Casertano et al., 2000b) to be sure.

The resulting map can be used with WEIGH_TYPE MAP_RMS and thus takes in account all the weird thing drizzle did to the data (per pixel different contributions from pixels with maybe different exposures).

Appendix B

SE parameters

Below the settings of SE are listed with their function and the best way to use them in a crowded field. These values were obtained after using SE on a series of simulations of Sextans A (north pointing wf2) and the HDF. The success rate was measured by the number of objects classified as galaxies in the simulations actually found in the HDF with the same settings.

```
CATALOG_TYPE ASCII # This means no header with the values
# in different colums is printed.
PARAMETERS_NAME /net/bartoli/bartoli/data2/Programs/default2.param
# file with the names of the parameters to be
# printed in the catalogs.
DETECT_TYPE CCD # type of image (alt PHOTO)

DETECT_MINAREA 16 # Minimum area of connected pixels in an object.
FILTER Y
FILTER_NAME /net/bartoli/bartoli/data2/Programs/sextractor2.1.6/config/gauss_4.0_7x7.conv

DEBLEND_NTHRESH 32 # the number of thresholds the intensity
# range is devided up in. 32 is the most
# common number.
DEBLEND_MINCONT 0.005 # percentage of flux a separate peak must
# have to be considered a separate object.
CLEAN Y # Should objects close to bright object be
# removed from the catalogs?
CLEAN_PARAM 1.0 # measure for 'clean'ing. wish I knew what it ment...
```

```
PHOT_APERTURES 5,10,20,30 # Fixed apertures in pixels
PHOT_AUTOPARAMS 2.5, 3.5
SATUR_LEVEL 200000. # saturation occurs at?
MAG_GAMMA 4.0 # Emulsion response slope
# SE doesn't work without this for some reason
GAIN 7.0 # number of photons / counts ratio
# if run over a counts per second image then the
# gain*exp time is used.
PIXEL_SCALE 0.05 # arcsec
SEEING_FWHM .17 #
STARNNW_NAME /net/bartoli/bartoli/data2/Programs/sextractor2.1.6/config/default
BACK_SIZE 32 # size in pixels of the area used to estimate the
# background
BACK_FILTERSIZE 1 # type of filter used in the background
BACKPHOTO_TYPE LOCAL # local estimate or an estimate for the entire
# image at once...
BACKPHOTO_THICK 32 #

#CHECKIMAGE_TYPE APERTURES # output image of SE.
#CHECKIMAGE_NAME Roza_F555W_wf3.fits # name of that image.
CHECKIMAGE_TYPE NONE
MEMORY_OBJSTACK 10000 # memory parameters
MEMORY_PIXSTACK 1500000 # defaults work best
MEMORY_BUFSIZE 1024
#SCAN_ISOAPRATIO 0.6
VERBOSE_TYPE NORMAL
```

Bibliography

Abraham, R. G., Valdes, F., Yee, H. K. C., and van den Bergh, S.: 1994, *Astrophys. J.* **432**, 75, Provided by the NASA Astrophysics Data System

Abraham, R. G., van den Bergh, S., Glazebrook, K., Ellis, R. S., Santiago, B. X., Surma, P., and Griffiths, R. E.

Bertin, E. and Arnouts, S.: 1996, **117**, 393, Provided by the NASA Astrophysics Data System

Bertin, E. and Dennefeld, M.: 1997, **317**, 43, Provided by the NASA Astrophysics Data System

Broadhurst, T. and Bouwens, R. J.: 2000, *Astrophys. J., Lett.* **530**, L53, Provided by the NASA Astrophysics Data System

Brown, M. J. I., Boyle, B. J., and Webster, R. L.: 2001, **122**, 26, Provided by the NASA Astrophysics Data System

Brown, M. J. I., Webster, R. L., and Boyle, B. J.: 2000, **317**, 782, Provided by the NASA Astrophysics Data System

Casertano, S., de Mello, D. ., Dickinson, M., Ferguson, H. C., Fruchter, A. S., Gonzalez-Lopezlira, R. A., Heyer, I., Hook, R. N., Levay, Z., Lucas, R. A., Mack, J., Makidon, R. B., Mutchler, M., Smith, T. E., Stiavelli, M., Wiggs, M. S., and Williams, R. E.: 2000a, **120**, 2747, Provided by the NASA Astrophysics Data System

Casertano, S., de Mello, D. ., Dickinson, M., Ferguson, H. C., Fruchter, A. S., Gonzalez-Lopezlira, R. A., Heyer, I., Hook, R. N., Levay, Z., Lucas, R. A., Mack, J., Makidon, R. B., Mutchler, M., Smith, T. E., Stiavelli, M., Wiggs, M. S., and Williams, R. E.: 2000b, **120**, 2747, Provided by the NASA Astrophysics Data System

Castander, F. J. and Lamb, D. Q.: 1999, *Astrophys. J.* **523**, 602, Provided by the NASA Astrophysics Data System

Fasano, G., Cristiani, S., Arnouts, S., and Filippi, M.: 1998, **115**, 1400, Provided by the NASA Astrophysics Data System

Fasano, G. and Filippi, M.: 1998, **129**, 583, Provided by the NASA Astro-

physics Data System

Franceschini, A., Silva, L., Fasano, G., Granato, L., Bressan, A., Arnouts, S., and Danese, L.: 1998, *Astrophys. J.* **506**, 600, Provided by the NASA Astrophysics Data System

Gardner, J. P., Sharples, R. M., Carrasco, B. E., and Frenk, C. S.: 1996, **282**, L1, Provided by the NASA Astrophysics Data System

Gebhardt, K. and Kissler-Patig, M.: 1999, **118**, 1526, Provided by the NASA Astrophysics Data System

González, R. A., Allen, R. J., Dirsch, B., Ferguson, H. C., Calzetti, D., and Panagia, N.: 1998, *Astrophys. J.* **506**, 152, Provided by the NASA Astrophysics Data System

Hashimoto, Y. and Oemler, A. J.: 1999, *Astrophys. J.* **510**, 609, Provided by the NASA Astrophysics Data System

Hashimoto, Y., Oemler, A. J., Lin, H., and Tucker, D. L.: 1998, *Astrophys. J.* **499**, 589+, Provided by the NASA Astrophysics Data System

Hilker, M., Kissler-Patig, M., Richtler, T., Infante, L., and Quintana, H.: 1999, **134**, 59, Provided by the NASA Astrophysics Data System

Hogg, D. W., Pahre, M. A., Adelberger, K. L., Blandford, R., Cohen, J. G., Gautier, T. N., Jarrett, T., Neugebauer, G., and Steidel, C. C.: 2000, **127**, 1, Provided by the NASA Astrophysics Data System

Hogg, D. W., Pahre, M. A., McCarthy, J. K., Cohen, J. G., Blandford, R., Smail, I., and Soifer, B. T.: 1997, **288**, 404, Provided by the NASA Astrophysics Data System

Irwin, M. J.: 1985, **214**, 575, Provided by the NASA Astrophysics Data System

Jarvis, J. F. and Tyson, J. A.: 1979, in *Instrumentation in Astronomy III*, Vol. 172, pp 422–428, Provided by the NASA Astrophysics Data System

Jarvis, J. F. and Tyson, J. A.: 1981, **86**, 476, Provided by the NASA Astrophysics Data System

Kalirai, J. S., Richer, H. B., Fahlman, G. G., Cuillandre, J., Ventura, P., D'Antona, F., Bertin, E., Marconi, G., and Durrell, P. R.: 2001, **122**, 266, Provided by the NASA Astrophysics Data System

Kambas, A., Davies, J. I., Smith, R. M., Bianchi, S., and Haynes, J. A.: 2000, **120**, 1316, Provided by the NASA Astrophysics Data System

Kron, R. G.: 1980, *Astrophys. J.* **43**, 305, Provided by the NASA Astrophysics Data System

Lanzetta, K. M., Yahil, A., and Fernandez-Soto, A.: 1996, *Nature* **381**, 759, Provided by the NASA Astrophysics Data System

Lubin, L. M., Brunner, R., Metzger, M. R., Postman, M., and Oke, J. B.:

2000, *Astrophys. J., Lett.* **531**, L5, Provided by the NASA Astrophysics Data System

Marleau, F. R. and Simard, L.: 1998, *Astrophys. J.* **507**, 585, Provided by the NASA Astrophysics Data System

Menanteau, F., Ellis, R. S., Abraham, R. G., Barger, A. J., and Cowie, L. L.: 1999, **309**, 208, Provided by the NASA Astrophysics Data System

Nonino, M., Bertin, E., da Costa, L., Deul, E., Erben, T., Olsen, L., Prandoni, I., Scodeggio, M., Wicenec, A., Wichmann, R., Benoist, C., Freudling, W., Guarnieri, M. D., Hook, I., Hook, R., Mendez, R., Savaglio, S., Silva, D., and Slijkhuis, R.: 1999, **137**, 51, Provided by the NASA Astrophysics Data System

Oke, J. B., Postman, M., and Lubin, L. M.: 1998, **116**, 549, Provided by the NASA Astrophysics Data System

Poli, F., Giallongo, E., Menci, N., D'Odorico, S., and Fontana, A.: 1999, *Astrophys. J.* **527**, 662, Provided by the NASA Astrophysics Data System

Rodighiero, G., Franceschini, A., and Fasano, G.: 2001, **324**, 491, Provided by the NASA Astrophysics Data System

Rodighiero, G., Granato, G. L., Franceschini, A., Fasano, G., and Silva, L.: 2000, **364**, 517, Provided by the NASA Astrophysics Data System

Simard, L., Koo, D. C., Faber, S. M., Sarajedini, V. L., Vogt, N. P., Phillips, A. C., Gebhardt, K., Illingworth, G. D., and Wu, K. L.: 1999, *Astrophys. J.* **519**, 563, Provided by the NASA Astrophysics Data System

Smail, I., Dressler, A., Couch, W. J., Ellis, R. S., Oemler, A. J., Butcher, H., and Sharples, R. M.: 1997, **110**, 213+, Provided by the NASA Astrophysics Data System

Smail, I., Ivison, R. J., Blain, A. W., and Kneib, J.-P.: 1998, *Astrophys. J., Lett.* **507**, L21, Provided by the NASA Astrophysics Data System

Smail, I., Kuntschner, H., Kodama, T., Smith, G. P., Packham, C., Fruchter, A. S., and Hook, R. N.: 2001, **323**, 839, Provided by the NASA Astrophysics Data System

Sowards-Emmerd, D., Smith, J. A., McKay, T. A., Sheldon, E., Tucker, D. L., and Castander, F. J.: 2000, **119**, 2598, Provided by the NASA Astrophysics Data System

Stetson, P. B.: 1987, *Publ. Astron. Soc. Pac.* **99**, 191, Provided by the NASA Astrophysics Data System

Szalay, A. S., Connolly, A. J., and Szokoly, G. P.: 1999a, **117**, 68, Provided by the NASA Astrophysics Data System

Szalay, A. S., Connolly, A. J., and Szokoly, G. P.: 1999b, **117**, 68, Provided by the NASA Astrophysics Data System

Teplitz, H. I., Gardner, J. P., Malumuth, E. M., and Heap, S. R.: 1998,
    Astrophys. J., Lett. **507**, L17, Provided by the NASA Astrophysics Data
    System

Treu, T. and Stiavelli, M.: 1999, *Astrophys. J., Lett.* **524**, L27, Provided by
    the NASA Astrophysics Data System

Trujillo, I., Aguerri, J. A. L., Gutiérrez, C. M., and Cepa, J.: 2001, **122**, 38,
    Provided by the NASA Astrophysics Data System

Väisänen, P., Tollestrup, E. V., Willner, S. P., and Cohen, M.: 2000, *Astro-
    phys. J.* **540**, 593, Provided by the NASA Astrophysics Data System

van Dokkum, P. G., Franx, M., Fabricant, D., Illingworth, G. D., and Kelson,
    D. D.: 2000, *Astrophys. J.* **541**, 95, Provided by the NASA Astrophysics
    Data System

Volonteri, M., Saracco, P., and Chincarini, G.: 2000a, **145**, 111, Provided
    by the NASA Astrophysics Data System

Volonteri, M., Saracco, P., Chincarini, G., and Bolzonella, M.: 2000b, **362**,
    487, Provided by the NASA Astrophysics Data System

Williams, K. A.: 2001, **122**, 55, Provided by the NASA Astrophysics Data
    System

Williams, R. E., Blacker, B., Dickinson, M., Dixon, W. V. D., Ferguson,
    H. C., Fruchter, A. S., Giavalisco, M., Gilliland, R. L., Heyer, I., Katsanis,
    R., Levay, Z., Lucas, R. A., McElroy, D. B., Petro, L., Postman, M.,
    Adorf, H., and Hook, R.: 1996, **112**, 1335+, Provided by the NASA
    Astrophysics Data System

Yan, L., McCarthy, P. J., Storrie-Lombardi, L. J., and Weymann, R. J.:
    1998, *Astrophys. J., Lett.* **503**, L19, Provided by the NASA Astrophysics
    Data System

Zabludoff, A. I. and Mulchaey, J. S.: 2000, *Astrophys. J.* **539**, 136, Provided
    by the NASA Astrophysics Data System

Index