Dissertation

submitted to the
Combined Faculties for the Natural Sciences and for Mathematics
of the Ruperto-Carola University of Heidelberg, Germany
for the degree of
Doctor of Natural Sciences

Put forward by

Diplom-Physicist: Gregor Seidel

Born in: Chemnitz, Germany

Oral examination: 17.11.2009

Automatic Detection of Gravitational Lenses in Astronomical Image Data

Referees:	Prof. Dr. Matthias Bartelmann
	Prof. Dr. Joachim Wambsganss

Zusammenfassung

Der starke Gravitationslinseneffekt kann sowohl entfernte Lichtquellen vergrößert abbilden also auch die Massenverteilung zwischen einer entfernten Quelle und dem Beobachter sondieren, und hat sich daher zu einem wichtigen astronomischen Hilfsmittel entwickelt, welches verwendet wird um kosmologische Parameter einzugrenzen, die projizierte Massenverteilung von Galaxienhaufen zu rekonstruieren und deren Dynamik zu analysieren. Die statistischen Anzahldichten von durch den starken Gravitationslinseneffekt an Galaxienhaufen in Bögen projizierten Hintergrundgalaxien zu bestimmen könnte zu unserem Verständniss der kosmologischen Strukturbildung beitragen; diese Bogenstatistik wird durch potentiell erhebliche systematische Effekte in der Auswahl von als Linsen wirkenden Galaxienhaufen aber wesentlich erschwert. Um eine größere Menge von Gravitationslinsen, die nicht durch undefinierte Systematiken belastet ist, bereitzustellen, wurde im Rahmen dieser Doktorarbeit ein effizientes, automatisiertes Bogenerkennungsprogramm und eine neue Methode zur Erkennung von länglichen Strukturen in Bildern entwickelt. Der Detektionsalgorithmus bedient sich dabei lokal kohärenter Strukturen und kann selbst lichtschwache Bögen effizient erkennen. Algorithmen zur weiteren Klassifikation von Detektionen und zur Entfernung von Fehldetektionen wurden ebenfalls entwickelt. Zur Kalibrierung und um die Empfindlichkeit des Programmes auf Bögen mit verschiedenen scheinbaren Helligkeiten zu bestimmen wurde das fertige Programm auf simulierte Bilder angewandt. Die Anwendung auf reale ACS Bilder resultierte in 24 neuen potentiellen galaktischen Gravitationslinsen.

Abstract

Strong gravitational lensing can magnify distant sources and also provides a direct probe of the mass density between source and observer. For these reasons, it has become an important tool in astronomy. Among other applications, it is used to constrain cosmological parameters, reconstruct the projected mass distribution of galaxy clusters and study their dynamics. Determining the statistical abundance of background galaxies projected into heavily distorted arcs by galaxy cluster lenses could improve our understanding of large scale structure formation, but arc statistics is particularly difficult due to possibly significant biases in the selection of galaxy cluster lenses. In order to provide a larger, unbiased sample of strong lenses, an efficient, automated arc detection software that uses a novel approach to detect elongated structures in images was developed for this PhD thesis. The new detection algorithm is based on locally coherent features and is capable of detecting even faint arcs with high computational efficiency. Postprocessing algorithms were developed to classify the detections and to remove false positives. For calibration and to determine its sensitivity to arcs of different magnitudes the completed software was applied to simulated images. The application to real ACS images resulted in 24 new galaxy-type lens candidates.

Contents

1 Introduction **1**
- 1.1 Scientific Relevance of Giant Arcs . . . 1
  - 1.1.1 General Classes of Gravitational Lensing . . . 1
  - 1.1.2 Cluster Reconstruction . . . 2
  - 1.1.3 Arc Statistics . . . 2
- 1.2 Recent Arc Detection Algorithms . . . 4
  - 1.2.1 Lens Based Algorithms . . . 5
  - 1.2.2 Source Based Algorithms . . . 6

2 Image Analysis **9**
- 2.1 The Data . . . 9
  - 2.1.1 Image Degrading Effects . . . 9
  - 2.1.2 Dithering / Drizzling . . . 10
- 2.2 Arcfinder Detection Algorithm . . . 12
  - 2.2.1 Cell Placement . . . 13
  - 2.2.2 Cell Transport . . . 13
  - 2.2.3 Cell Orientation . . . 18
  - 2.2.4 Finding Coherent Features . . . 19
  - 2.2.5 Object Generation . . . 24
- 2.3 Postprocessing . . . 25
  - 2.3.1 Background & Noise Estimation . . . 25
  - 2.3.2 Histogram Equalisation . . . 26
  - 2.3.3 Cell Reevaluation . . . 27
  - 2.3.4 Generation of Path Graphs . . . 29
  - 2.3.5 Active Contour Segmentation . . . 31
  - 2.3.6 Basic Contour Evolution . . . 32
  - 2.3.7 Arcfinder Image Segmentation . . . 34
  - 2.3.8 Classification of Candidates . . . 39
  - 2.3.9 Masking of Stellar Artefacts . . . 39
  - 2.3.10 Spiral Galaxy Detection . . . 41
- 2.4 Summary of Detection and Filter Parameters . . . 44

3 Application to Simulated Images **45**
- 3.1 Preparing the Data . . . 46
- 3.2 Regarding the Completeness and Spurious Detections . . . 46

4 Application on COSMOS ACS Images 51
4.1 Galaxy Lens Candidates 51
4.2 Spurious Detections 61

5 Summary, Conclusions and Outlook 63
5.1 Method and Results 63
5.2 Outlook 64

6 Appendix 67
6.1 Relation of the Ellipse Orientation and the Complex Ellipticity 67
6.2 Pixel Reweighting to Account for a Partial Disc Overlap 68

1 Introduction

Light from distant astronomical sources reacts to the mass distribution encountered on its way to the observer and is deflected towards massive objects acting as gravitational lenses. Next to the precession of the perihelion of Mercury, the shift in the apparent position of a star due to gravitational light deflection on the Sun, measured in 1919 during a solar eclipse, became one of the first observational tests for Einstein's general theory of relativity (Dyson et al. 1920).

1.1 Scientific Relevance of Giant Arcs

Today, we often distinguish between three classes of gravitational lensing, each associated with specific physical properties of the lensing system and scientific analysis methods.

1.1.1 General Classes of Gravitational Lensing

In the strong lensing regime, sources are projected into heavily distorted, magnified and even multiple images[1]. Background galaxies distorted into strongly elongated giant arcs by an interjacent galaxy cluster are a typical example, which is illustrated in Figure 1.1 with one of the first observations of gravitational lensing arcs (Narayan & Bartelmann 1997). This is the class of lensing this thesis is concerned with, and applications will be discussed at a later point.

Weak lensing in contrast cannot create multiple images of the same source and distortions are not significant for single sources. However, by averaging over the distortions of large numbers of weakly lensed galaxies, we obtain a powerful observational tool that can be applied to reconstruct projected cluster mass densities (Kaiser & Squires 1993; Bartelmann & Schneider 2001) and infer cosmological parameters from a statistical cosmic shear analysis (Heavens 2003; Castro et al. 2005; Heavens et al. 2006).

Microlensing events fall into the strong lensing regime, but the angular separations between lens and source images are too small to be resolved by current instruments. Systems with changing lensing configuration are still accessible to light curve measurements, however, which can be used to study binary systems or search for extrasolar planets (Wambsganss 2006).

---

[1] Although images can appear considerably magnified, the surface brightness is conserved by gravitational lensing.

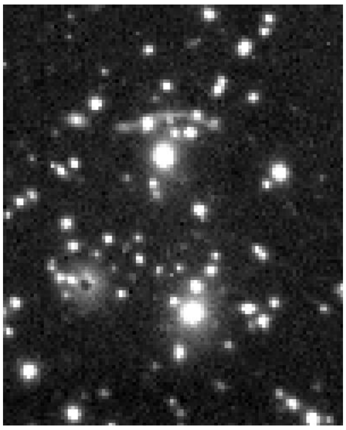

Figure 1.1: In 1986, the first arcs were independently discovered near the cores of the cluster lenses Abell 370 and CL 2244-02 (Soucail et al. 1987; Lynds & Petrosian 1986). This figure shows the prominent giant arc in A370 in a zoom into the original CFHT R-band observation.

1.1.2 Cluster Reconstruction

One possible application of strong lensing makes use of its magnification effect to study some of the earliest galaxies, by chance strongly lensed by a foreground galaxy or galaxy cluster (Allam et al. 2007). The precise layout of arc images, conversely, can be used to study the lens itself, where it turns out to be helpful to consider also data from weak lensing:

Reconstructing projected cluster mass densities by weak lensing alone is limited in two ways. The first is the so called mass sheet degeneracy: weak lensing reconstruction techniques considering only distortion in contrast to distortion and magnification are oblivious to a transformation $\Sigma \rightarrow \lambda\Sigma + (1 - \lambda)$ of the projected mass density for any scalar value λ. Second, cluster cores are inaccessible to such an analysis, since smaller scales need to be resolved, background galaxies blend with galaxies in the cluster core and the lensing properties are more complex and inhomogeneous than in the weak lensing regime.

Since the configurations of strongly lensed images are highly sensitive to the mass distributions in cluster cores in particular, their constraints are complementary to the ones from weak lensing. Several combined weak and strong reconstruction techniques are developed (Broadhurst et al. 2005; Leonard et al. 2007; Merten et al. 2009) that aim at a detailed study of clusters, their dynamics and evolution, and in extension a greater understanding of large scale structure formation. Using the redshift of lensed sources and the reconstructed mass densities, the cosmological mass density Ω_M can be inferred (Soucail et al. 2004; Broadhurst et al. 2005). However, a larger number of good lensing systems is desirable to improve the results' accuracy and generality.

1.1.3 Arc Statistics

The total number of giant arcs observable in the sky mainly depends on

1. the number density and redshift distribution of extended sources.

2. the abundance of sufficiently compact and massive galaxy clusters.
3. the strong lensing cross-sections of galaxy clusters.

Only the distribution of distant galaxies with redshift $z \gtrsim 1.0$ that act as sources is fairly well known through observations. The abundance of massive clusters surpassing the critical surface mass density for strong lensing probes the steep high-mass end of the galaxy cluster mass function, and is therefore sensitive to the normalisation of the power spectrum and non-linear structure formation. Strong lensing cross-sections are significantly enhanced by asymmetries and irregularities in the central density distributions of clusters (Grossman & Narayan 1988; Kovner 1989; Miralda-Escude 1993; Bartelmann et al. 1995).

During the last decade, there has been an ongoing debate over whether theoretical estimates can reproduce the number of observed arcs in the sky or not:

Using cosmological dark matter simulations and simulated sets of in total nine galaxy clusters in three projections each, Bartelmann et al. (1998) estimated a total number of approximately 280 giant arcs in a ΛCDM cosmology, and noted that this result is nearly an order of magnitude less than the observational value of $1500 - 2300$ extrapolated from observed arc counts in clusters taken from the Einstein Extended Medium Sensitivity Survey (EMSS).

While Bartelmann et al. (1998) found a strong dependence of the arc abundance on the cosmic dark energy density Ω_Λ, analytical calculations using singular isothermal spheres by Cooray (1999) and Kaufmann & Straumann (2000) find near degeneracy towards Ω_Λ and dispute the numerical results. However, Meneghetti et al. (2003) used improved analytical models and found considerable discrepancies between the cross-sections derived from analytical models and realistic numerical simulations.

The low theoretical estimate for the arc abundance was attributed to a steep dependence on source redshift by Wambsganss et al. (2004), yet Li et al. (2005) and Fedeli et al. (2006) noted that the light bundle magnification used by Wambsganss et al. (2004) to approximate the number of giant arcs is not a good estimator and the redshift dependence of the strong lensing cross-section is shallower than assumed.

By increasing the density of background galaxies and decreasing the extrapolated arc count, Dalal et al. (2004) reproduced the number of observed arcs using the lensing cross-sections determined in Bartelmann et al. (1998). This does not appear to be a valid solution, though, given a comparatively high number of arcs observed in distant clusters (Thompson et al. 2001; Gladders et al. 2003; Zaritsky & Gonzalez 2003).

Horesh et al. (2005) took five of the simulated clusters used by Bartelmann et al. (1998), again in three principle projections, and found that a new theoretical estimate that uses real Hubble Deep Field galaxies as background sources, realistic cluster foreground and an arc detection algorithm based on the SExtractor software is in agreement with arc abundances in a sample of 10 observed clusters.

Also, after applying a semi-analytic method to model the influence of triaxiality on lensing cross-sections, Oguri et al. (2003) claimed that the order of magnitude increase in the predicted number of arcs solves the arc statistics problem.

However, Li et al. (2006) and Fedeli et al. (2008) found that a steep dependence on the power spectrum normalisation σ_8 again causes almost a magnitude discrepancy of predicted and observed arc abundances when a low value of approximately 0.8 consistent with current 5-year WMAP results (Dunkley et al. 2009) is considered. In particular, the discrepancy does not appear to be resolved by the findings in the earlier studies noted above, as they unanimously assumed a high normalisation value of $\sigma_8 \geq 0.9$.

Several studies were untertaken on which cluster properties are most relevant to strong lensing. Flores et al. (2000) concluded that cluster galaxies have a negligible effect on the cross-section, although Meneghetti et al. (2000) noted that a central cluster cD galaxy can enhance lensing by about 50%. Puchwein et al. (2005) used several increasingly complex gas models to describe the effect of intracluster gas on lensing, and concluded that it can enhance cross-sections significantly in models that include cooling and star formation, although there was some overcooling that could account for part of the increase in central density and consequently also lensing. Torri et al. (2004) and Fedeli et al. (2006) determined that ongoing mergers can double the cross-section of a cluster for a time, after which it goes back to the pre-merger level.

Since cluster dynamics clearly have a large impact on the lensing cross-section, selecting a sufficiently random cluster sample for observations of arc abundances might be problematic. Bartelmann & Steinmetz (1996) and Fedeli & Bartelmann (2007) concluded that extrapolating the number of observed arcs from those found around clusters selected by their x-ray luminosity or optical richness is likely to produce a biased result. Also, an observer bias is unavoidable at the moment because all known giant arcs in galaxy clusters were detected by manual inspection of clusters. To avoid these biases, an automatic detection and classification algorithm that can be applied both to observations and simulated data is required.

1.2 Recent Arc Detection Algorithms

The amount of astronomical data suitable for arc detection is rapidly increasing, calling for efficient arc detection algorithms[2] even if the observer bias were to be discounted. Current arc detection algorithms either focus on the lens and process galaxy spectra and image data in order to detect possible strong lensing features in the immediate surrounding of a lens, or they focus on the sources and attempt to detect arc features independently, without processing data on the lens explicitly. Algorithms pursuing the first approach are typically concerned with galaxy-scale lenses and can be very sensitive for this important type of lensing, but their dependence on spectral or multi-band data limits the range of applications. The second approach is suitable to find galaxy cluster or galaxy group type lenses, can principally be used to detect dark clusters lenses – if they exist – and can operate blindly on available survey data without a priory information. However, since lens and arc features are not processed together, it is generally less sensitive to arcs in close proximity to a galaxy lens. Follow up observations are generally necessary to verify initial detections.

---

[2] or armies of fresh astronomers inspecting the data manually

Without aspiring to be complete, I will present some recent automatic arc detection algorithms in the following.

1.2.1 Lens Based Algorithms

Though not strictly an arc detection algorithm, the method proposed by Bolton et al. (2004) does detect strong gravitational lensing in galaxies. They used spectroscopic data from about 51000 luminous red galaxies (LRG) in the Sloan Digital Sky Survey (SDSS) to search for redshifted OII($3728\AA$) emission features in their residual spectra, determined by modelling the spectrum of each foreground LRG and subtracting the model from the original spectrum. If a significant and sufficiently redshifted OII line and at least two lines out of the corresponding Hβ($4863\AA$), OIII($4960\AA$) and OIII($5008\AA$) lines were found, the galaxy was counted as a lens candidate. In the SDSS sample, 163 such initial candidates were further analysed to remove spurious detections, after which 49 likely lensing systems remained.

The RING routine developed by Cabanac et al. (2007) was successfully applied to the Canadian France Hawaii Telescope (CFHT) Legacy Survey, where the algorithm found 12 previously unknown Einstein rings. The principle idea of this method is to produce residual images for catalogued E/S0 galaxies by subtracting α times their i' band image from the respective g' band observation, and select residual objects above the image's noise level in a 0.5" to 2.5" annulus around the galaxy as arc candidates. The algorithm was reported to be sensitive to most rings with radius larger than 0.8". Final candidates were checked bye eye in order to remove spurious detections.

The method proposed by Marshall et al. (2009) produces residual images by subtracting elliptically symmetric models from original galaxy images preselected by their colour and signal-to-noise ratio. The lensing system is then modelled by a singular isothermal sphere galaxy with external shear by projecting the residual pixels into the source plane and assessing the consistency of each source image given the possible lensing configurations. For the best fit lens model, (1) the deviation of the residual image from the back projection of the source plane into the image plane, (2) the magnitude of the reconstructed source and (3 and 4) two quantifiers for the uncertainty of the Einstein radius are combined into a data vector $\boldsymbol{d}$. To relate a scalar value to the four-dimensional $\boldsymbol{d}$, visual inspection was used to assign a scalar H ranging from 0: 'definitely not a lens' to 3: 'definitely a lens' to the galaxies in a training set of 97 bright red galaxies in the Extended Groth Strip (EGS) Hubble Space Telescope (HST) survey, and a posterior probability distribution

$$P(H|\boldsymbol{d}) = P(\boldsymbol{d}|H)P(H)$$

was constructed from the $\boldsymbol{d}$ and H values of each galaxy. Applied on new data, the classification 'robot' then determines average values

$$H_r = \frac{\sum_H HP(H|\boldsymbol{d})}{\sum_H P(H|\boldsymbol{d})}$$

for the calculated vectors $\boldsymbol{d}$ that are used to select lens candidates. In one of the presented search strategies, they expect to find approximately 672 lens candidates among 10000 LRGs, including 8.9 out of 10 galaxy-scale lenses, in a one deg^2 field observed with the HST, an estimate consistent with a blind test on EGS data they conducted.

1.2.2 Source Based Algorithms

Algorithms searching for arcs explicitly include the method developed by Lenzen et al. (2004), which filters an image I uses anisotropic diffusion $\partial_t I(t,\boldsymbol{x}) - \mathrm{div}(D(\boldsymbol{x}, I, \nabla I)\nabla I(t,\boldsymbol{x})) = 0$ to enhance edges in the image. The diffusion matrix D is calculated using a smoothed structure tensor

$$J_\mu(\boldsymbol{x}) = \left(K_\mu * \left[(\nabla(K_\sigma * I))(\nabla(K_\sigma * I))^T\right]\right)(\boldsymbol{x}),$$

where K_μ, K_σ are isotropic smoothing kernels and $*$ denotes the convolution operation: local averaging is applied not only on the image but also the edge orientation contained in the eigenvectors and eigenvalues of the structure tensor. Arc candidates are selected using a watershed method on the filtered image.

Horesh et al. (2005) developed an algorithm that uses repeated SExtractor segmentation on the same image for six different sets of object extraction parameters. Objects below a set length-to-width ratio are removed from the segmentation images in each step, and every second SExtractor operation is applied on an image where objects extracted in the previous segmentation and below the corresponding length-to-width threshold were masked. The final output is a combination of the three resultant segmentation masks. The algorithm was applied on simulated and real cluster lenses as discussed in 1.1.3 above.

The ARC detector presented by Alard (2006) uses second order moments in small areas of side length M, set to approximately three times the seeing, to determine a local orientation $\boldsymbol{e}$ at each image coordinate $\boldsymbol{x}$ in a Mexican-hat filtered image. This is used to compute two intensity profiles

$$\begin{aligned} I_{||}(\xi) &= \int_{-M}^{M} I(\boldsymbol{x} + \lambda\boldsymbol{e} + \xi\boldsymbol{e}^\perp)\mathrm{d}\lambda \text{ and} \\ I_\perp(\xi) &= \int_{-M}^{M} I(\boldsymbol{x} + \xi\boldsymbol{e} + \lambda\boldsymbol{e}^\perp)\mathrm{d}\lambda\,, \end{aligned}$$

where $||$ and $\perp$ denotes the direction of integration relative to the orientation $\boldsymbol{e}$. These are then used to estimate the local elongation as

$$Q(\boldsymbol{x}) = \frac{1}{2M}\frac{I_{||}(0)}{\sup_{\xi=-M}^{M}(I_\perp(\xi))}\,.$$

If $I_{||}(0)$ is integrated following the ridge line of an arclet at position $\boldsymbol{x}$, the corresponding $Q(\boldsymbol{x})$ is naturally expected to be large. Arc candidates are constructed as continuous areas with large values of Q. 24 good arc candidates were selected in the CFHT Legacy Survey

(CFHTLS) using this method (Cabanac et al. 2007).

Kubo & Dell'Antonio (2008) used SExtractor segmentation by pixel thresholding (Bertin & Arnouts 1996) to obtain a list of general objects in the four square degree F2 field of the Deep Lens Survey (DLS), and determined octopole moments and ellipticities (as in 2.2.3) to identify galaxy-scale lenses among them. Octopoles were measured in a coordinate system aligned with the major axis of each feature, and two octopole moments were used to define 'arcness' and 'anti-arcness'. An initial number of about 40000 spurious detections was subjected to further parameter cuts and visual inspection, after which 2 good candidates remained.

Moldovan (2009) applied the Gentle AdaBoost (Freund & Schapire 1996; Friedman et al. 1998) algorithm to construct a classifier from polar shapelet coefficients (Massey & Refregier 2005) of 100x100 pixel training images including 10000 simulated galaxy-scale lenses, 10000 simulated 'spurious' galaxy pairs and 100000 actual background patches from the Deep Lens Survey (DLS). Using a sliding 100x100 pixel window, the algorithm is then applied to the DLS and selects approximately 500 spurious detections and one good galaxy-type lens candidate.

2 Image Analysis

Initial data analysis is performed on astronomical images using the information contained in a single spectral band without a-priori information on likely arc positions. Postprocessing optionally includes data in the form of star and galaxy catalogues to remove detections caused by diffraction spikes (see 2.1.1 and 2.3.9) and spiral galaxies, as well as galaxy and cluster catalogues to obtain the vicinity and orientation of a detection relative to a potential lens. The following section remarks, on a basic level, on data properties that become relevant in data postprocessing; it can be safely skipped for the purpose of understanding the initial detection algorithm, which is detailed in 2.2. Section 2.3 deals with postprocessing steps that aim to significantly reduce the amount of spurious detections and to classify detections for the creation of an arc candidate catalogue.

2.1 The Data

For the science grade images used, each pixel is the end result of several contributory effects and processing steps. For the purpose of understanding the input data, I will summarise the most common sources of image degradation and noise before describing the arcfinder processing pipeline.

2.1.1 Image Degrading Effects

When working with astronomical observations, one has to content with photons other than those from the relevant sources and blended objects close to the line of sight: photons scattered by dust in our Solar System – the so called zodiacal light – the galactic foreground, cosmic rays, both thermal emissions and reflections in the telescope optics and stray light from the Moon are important light polluting effects. Ground based observations are affected by thermal emission and stray light in the atmosphere, originating for example in nearby cities. These effects produce an additive non-uniform signal, and while some can be corrected for by subtracting their mean contribution to the image, the Poisson distributed noise they introduce remains and reduces the signal-to-noise ratio of the observation.

In addition to the atmospheric seeing that affects ground based observations, all photons passing through the telescope are subject to diffraction and other optical aberration effects in the telescope which, depending on the photon energy, result in a shift of the final position on the detector and causes point sources to be projected into extended diffraction patterns. The cumulative effect is usually described by a point spread function (PSF) that is convolved with the image and results in a specific form of blurring that limits the maximum amount of information in an observation regardless of the pixel size. A common approximation for the PSF is the Airy disk (see 2.3.9), which is the exact solution for diffraction on a circular aperture, but the precise form of the PSF is dependent on the colour of the observed object and the observation time, due to thermal expansion in the telescope and variations in the detector position. Seeing and optical aberrations are problematic for arc detection because the smoothing effect blends nearby objects together and reduces the signal-to-noise ratio for barely resolved sources. Also, diffraction on the metal bars holding the secondary mirror results in extended optical artefacts appropriately termed diffraction spikes, which are clearly visible for bright stars. In particular if they extend far beyond the star, the faint end of these spikes also resembles straight arc segments (see 2.3.9).

Only a fraction of the incoming photons pass the telescope assembly and induce current in a photo diode on the charge-coupled device (CCD) detector, which increases the charge in one pixel element. This depends primarily on the colour filters used and the CCD quantum efficiency. Altogether, the peak efficiency of this process is commonly between 0.1 to 0.5. The quantum efficiency of the CCD is a function of the photon energy and typically assumes values between 0.2 and 0.8 for the relevant spectral range. Since each CCD element can only take a finite charge, termed saturation charge level, sensor response at some point becomes nonlinear when this charge is approached. If the saturation level is exceeded, charge can be transferred to adjacent pixels, preferentially in the same transfer row, creating stripes of oversaturated pixels. Similar to the diffraction spikes above, the blooming effect can cause spurious arc detections, but these can be trivially avoided by masking areas above a limiting intensity threshold. Oversaturation and blooming are mainly problematic when attempting to estimate the flux and position of bright stars. During read-out, each pixel charge is converted into a digitally stored value: a DN, for digital number, or ADU, for analogue digital unit. This value is the starting point for any digital image processing.

There are other CCD related effects aside from saturation that introduce noise or deteriorate the image quality: Dark current describes charge from thermally excited electrons accumulating on each pixel at a different rate and is subtracted using dark frames. The conversion of charge into a digital number involves the CCD transfer efficiency from one row to another, different gain ratios for the conversion of electron counts into digital values, amplifier bias levels and electronic noise during read-out. The transfer efficiencies, gain ratios and bias levels are usually well known, and again, everything but the noise can be removed.

2.1.2 Dithering / Drizzling

It is often advantageous to stack several astronomical images taken within a fraction of the total exposure time together instead of using a single observation, and to spatially offset the individual images by small shifts, a process known as dithering in astronomy. First, multiple smaller exposures might be necessary simply to avoid pixel oversaturation. Second, it becomes possible to identify and remove variable features like cosmic rays and satellite tracks. Third, by shifting the pointing by few pixels for each exposure, hot pixels and other single pixel defects on the CCD can be removed while retaining most of the information, and imperfections in the dark current and flat field subtraction are smoothed out. Finally, sub-pixel shifts can be used to sample the PSF of an instrument even if it is not sufficiently resolved by the CCD pixels (see Figure 2.1).

A popular algorithm to reconstruct observations from dithered images is drizzling[1], implemented in the MultiDrizzle software (Fruchter & Hook 1997; Fruchter et al. 2009).

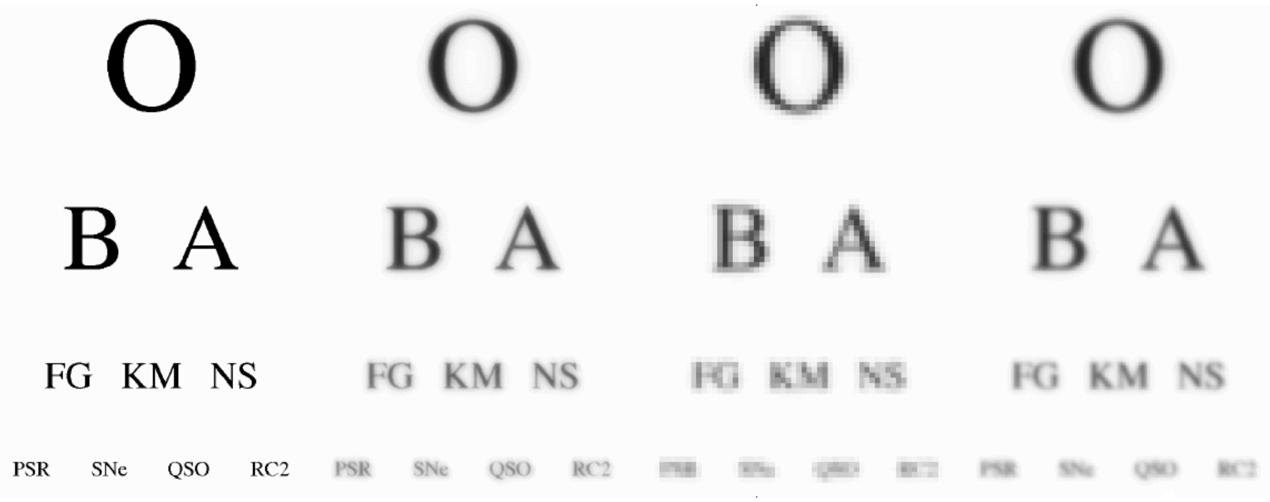

Figure 2.1: To illustrate the drizzling algorithm, an example image is shown in the upper left, the same image after PSF convolution in the upper right, the image sampled by CCD pixels in the lower left and the linear reconstruction of several dithered samples in the lower right segment (images were made by Fruchter & Hook (1997)).

By design, pixels in the reconstructed image are superpositions of original pixels normalised by the respective area overlap and a weighting factor, which is the inverse variance in the statistically optimal case. Unfortunately, this removes statistical independence of pixel intensities: the intensity variance on a flat background is reduced and does not increase linearly with the number of pixels anymore (Casertano et al. 2000). The resultant pixel correlations additionally generate connected regions of similar intensity on a very small scale that can generate spurious detections when searching for arclets at the limit of the image resolution and can connect point sources which then appear as extended features.

---

[1] or more informally: variable-pixel linear reconstruction

2.2 Arcfinder Detection Algorithm

Similar to other automated arc detection methods in the optical, the initial detection routine for the arcfinder algorithm attempts to locate all strongly elongated features in an image brighter than their local background. Colour information is not used in the current implementation. The immense amount of data in recent and upcoming optical surveys combined with the requirement of running ’blind’, that is without a priori data on lens positions, demands an efficient detection algorithm. In this section I will describe a solution to this image processing problem that is effective also for objects with low signal-to-noise ratios (see also Seidel & Bartelmann (2007)). An integral part of the presented method is to segment the image into cells instead of processing each pixel individually, which increases its computational efficiency significantly. The price is a loss of elegance in describing the detection characteristics: in order to account for all possible degradating effects, we have to calibrate the algorithm using realistic lensing simulations. Some basic statistical properties will be discussed in the following, however.

Intuitively, the underlying idea is to

1. uniformly throw ’needles’ on the image (see 2.2.1),
2. attract them to locally bright features (see 2.2.2),
3. align them parallel to these features using second moments (see 2.2.3) and then to
4. look for ’paths’ formed of similarly oriented needles, ignoring areas where they point into random directions (see 2.2.4 and 2.2.5).

The immediate consequences of such a scheme are that:

- significantly fewer ’needles’ than pixels are required, which reduces the computational footprint.
- it is independent of large scale variations in the background, since it relies only on local feature properties.
- the computation of second moments inside an area A enhances the signal-to-noise ratio by approximately $\sqrt{A}$ compared to the signal of a single pixel.

This method is fast and well suited to find faint and separate features, but there are noteworthy limitations: it is not well adapted to detect corners and disentangle closely packed features, a task for which other image processing methods may be preferable.

Needles will henceforth be called cells. A cell is defined as the set of all pixels A inside a minimal radius $d_0/2$ from its variable centre point $\boldsymbol{x}_c$. The scale size d_0 is a shared property for all cells and the main user defined parameter that depends on the scale of the features that are to be detected. Further cell properties like the centre of brightness $\bar{\boldsymbol{x}}$ and the second brightness moments Q_{ij} are computed at run time over the set of pixels A.

2.2.1 Cell Placement

In an initial step, cells are distributed uniformly on the image area such that the union of cell sets A covers the complete image. Naturally, both requirements might be loosened if a priori data on arc positions are available. For the sake of simplicity I use a square grid with characteristic grid distance $d_1 < d_0$ aligned with the image coordinate system, although other possibilities including a hexagonal grid could be considered to increase isotropy. We find that it is useful to specify d_1 not directly but through a proportionality factor d_1/d_0 in relation to the scale size, which ensures a constant pixel coverage for varying scale sizes. In most cases, it is not necessary to adapt the ratio d_1/d_0 to different feature scales.

In first approximation, the cell density and the number of pixels in each cell has a direct influence on the computational efficiency, and the execution time of the initial detection algorithm increases with $(d_0/d_1)^2$. The total complexity including postprocessing steps, on the other hand, will also depend on the number and size of the preliminary detections.

2.2.2 Cell Transport

The second step in the algorithm is to attract cells to locally bright features, where we assume that an elongated feature is for a significant part of its length brighter than its local background. It is achieved by iterative transport of each cell to its weighted first moments respectively centre of brightness

$$\bar{\boldsymbol{x}} = \frac{\int_A \boldsymbol{x} q(I(\boldsymbol{x}))\,\mathrm{d}^2x}{\int_A q(I(\boldsymbol{x}))\,\mathrm{d}^2x}\,. \tag{2.1}$$

Here, $I(\boldsymbol{x})$ is the intensity at position $\boldsymbol{x} = (x_1, x_2)$ and q a weight function, which is set to

$$q(I) = \begin{cases} I - \bar{I}, & \text{for } I > \bar{I} \\ 0, & \text{else} \end{cases} \tag{2.2}$$

where $\bar{I}$ is the average brightness $\int_A I(\boldsymbol{x})\,\mathrm{d}^2x/A$.

The main advantage in employing the centre of brightness in contrast to an image gradient method is numerical robustness. Under most circumstances, a cell's position $\boldsymbol{x}_c$ converges on the nearest local intensity maximum eventually if it is transported towards $\bar{\boldsymbol{x}}$ in each step of the iteration. The rate of convergence is invariant both under translation and scaling of I. In the following, I briefly mention a few considerations that are useful when varying the number of steps. Empirically, a fixed number of three iterations appears to be a good choice for arc detection in the optical.

If discretisation errors from the finite pixel size can be neglected for a moment, the distance traversed in one dimension in a single step can be evaluated analytically. Without loss of generality, I use a coordinate system centred on the previous cell position $x_{c,t}$ and a scale size of two. For the purpose of this example, let the intensity function be $I(x) = x$. Then

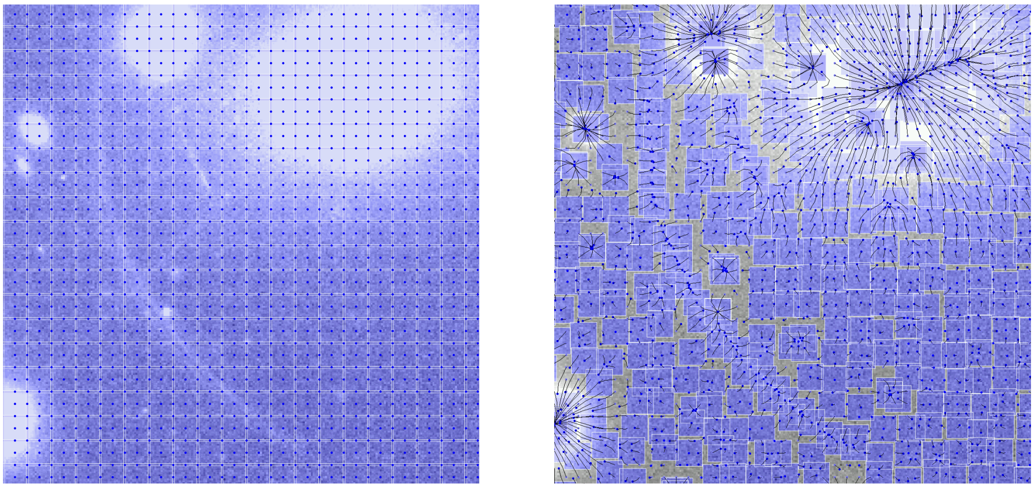

Figure 2.2: Zeroth and third cell transport iteration in a cutout of an HST WFPC2 observation of Abell 2390. Each cell is marked by a point, and every second row / column of cells by a square of side-length d_0.

$q(I(x)) = 0$ for $x \leq 0$, $q(I(x)) = x$ for $x \geq 0$ and without noise the cell shifts by exactly

$$x_{c,t+1} = \bar{x}_t = \frac{\int_{-1}^{1} xq(I(x))\,\mathrm{d}x}{\int_{-1}^{1} q(I(x))\,\mathrm{d}x} = \frac{\int_{0}^{1} x^2\,\mathrm{d}x}{\int_{0}^{1} x\,\mathrm{d}x} = \frac{2}{3}\,. \tag{2.3}$$

In order to include noise, we integrate over I in addition to x using a probability density function $f(I, I_s(x))$, where $I_s(x)$ is the unperturbed signal intensity. With the previous simplifications and Gaussian distributed noise with standard deviation σ, the probability density is proportional to $\exp(-(I - I_s(x))^2/2\sigma^2) = \exp(-(I - x)^2/2\sigma^2)$ for $I > 0$. Since $q(I) = 0$ for $I \leq 0$ it suffices to integrate only over positive intensities and the term for the cell shift reads

$$x_{c,t+1} = \bar{x}_t = \frac{\int_{-1}^{1}\int_{0}^{\infty} xI\mathrm{e}^{-(I-x)^2/2\sigma^2}\,\mathrm{d}x\mathrm{d}I}{\int_{-1}^{1}\int_{0}^{\infty} I\mathrm{e}^{-(I-x)^2/2\sigma^2}\,\mathrm{d}x\mathrm{d}I}\,. \tag{2.4}$$

This result is easily transferred to the more general case with arbitrary scale size d_0 and an intensity $I_s(x) = I_0 + xd_xI$ by inserting a σ value divided by d_xI and by multiplying the result in Eq. 2.4 with $\frac{1}{2}d_0$. Figure 2.3 contains the resulting one-step distances for varying noise levels. As expected, the distance traversed by a cell decreases with decreasing signal-to-noise values.

In order to ensure that neighbouring cells move sufficiently close to the ridge line of an arc, which shall be defined as the smooth curve running through its full length, the required number of steps to reach a set distance from a feature increases with decreasing signal-to-noise levels $\propto d_xI/\sigma$. A more realistic calculation would employ the same principles, but

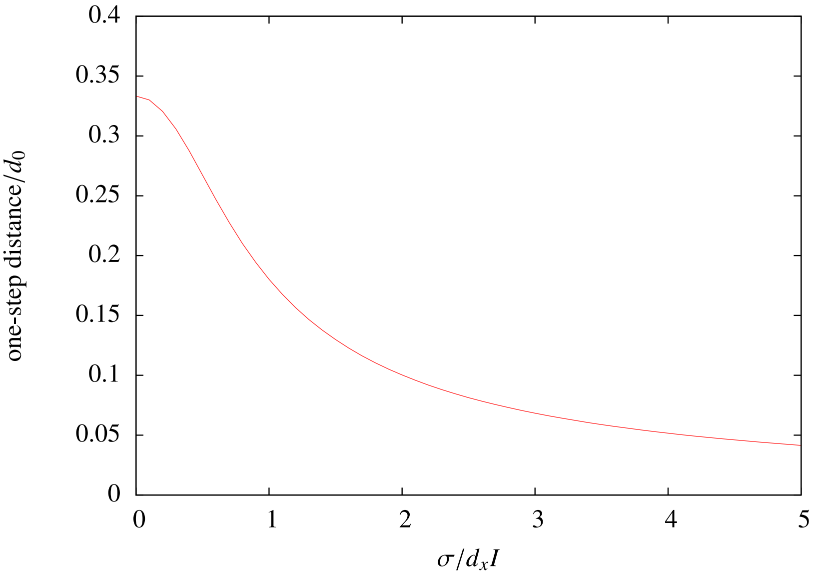

Figure 2.3: Cell transport distance for a signal with constant slope d_xI and Gaussian distributed noise for varying noise levels.

use feature models instead of a constant intensity slope for the intensity profile. Accounting for the steps described in 2.2.3, it would also take into account the transition point from perpendicular to parallel orientation relative to the ridge line, which happens earlier for wider brightness profiles, and the correlation measure in 2.2.4.

Considering finite pixel sizes, what is the mean absolute distance a cell travels in a purely noise dominated signal? Let us assume a Gaussian distributed signal $I(x)$ with mean I_0 and standard deviation σ from the noise, consider only one dimension $x \in -\frac{1}{2}(d_0 - 1), ..., \frac{1}{2}(d_0 - 1)$ and a number of n transport iterations. The mean random walk distance under these assumptions can be approximated by $\xi_n = \frac{1}{2}\sqrt{n(1 - 1/\pi)(d_0 - 1/d_0)/6} \propto \sqrt{nd_0}$ with an error of less than 5% compared to numerical results for scale sizes $d_0 \geq 3$. This estimate can be motivated with a short calculation and several rough approximations. Assuming the measured average $\bar{I} \approx I_0$, the weight

$$q(I) = \begin{cases} I - I_0, & \text{for } I > I_0 \\ 0, & \text{else} \end{cases} \tag{2.5}$$

shares – aside from a different mean value – a normal distribution with the signal I for positive values and is cut off below zero. Since the probability for one pixel of $I > 0$ is $\frac{1}{2}$and we sum over d_0 pixels, the expected value for the total weight is

$$\mathrm{E}\left(\sum_x q\right) = \frac{d_0}{2}\frac{1}{2}\int_0^\infty I\frac{1}{\sqrt{2\pi}\sigma}e^{-\frac{I^2}{2\sigma^2}}\mathrm{d}I = \frac{d_0}{2}\sqrt{\frac{2}{\pi}}\sigma\,. \tag{2.6}$$

The $\sqrt{2/\pi}$ factor is noteworthy because it converts from the standard deviation to the average absolute value in a normal distributed variable. The mean of the first moment term $\sum_{x=-h}^{h} xq(I(x))$ – where $h = \frac{1}{2}(d_0 - 1)$ – is zero, but the expected absolute distance traversed can be determined through the variance

$$\begin{aligned}\operatorname{Var}\left(\sum_x xq\right) &= \lim_{N\to\infty} \frac{1}{N} \sum_{i=1}^{N} \left(\sum_{x=-h}^{h} xq_i(x) \right)^2 \qquad (2.7)\\ &= \lim_{N\to\infty} \frac{1}{N} \sum_{i=1}^{N} \sum_{x=-h}^{h} \sum_{y=-h}^{h} xyq_i(x)q_i(y)\\ &= \lim_{N\to\infty} \frac{1}{N} \sum_{i=1}^{N} \left(\sum_{x=-h}^{h} x^2 q_i^2(x) + \sum_{x=-h}^{h} \sum_{y\neq x} xyq_i(x)q_i(y) \right) .\end{aligned}$$

The averaging can now be applied to the innermost terms, where products of dependent and independent random variables have been separated:

$$\begin{aligned}\ldots &= \lim_{N\to\infty} \left(\sum_{x=-h}^{h} x^2 \frac{1}{N} \sum_{i=1}^{N} q_i^2(x) + \sum_{x=-h}^{h} \sum_{y\neq x} xy \frac{1}{N} \sum_{i=1}^{N} q_i(x)q_i(y) \right)\\ &= \sum_{x=-h}^{h} x^2 \frac{1}{2}\sigma^2 + \sum_{x=-h}^{h} \sum_{y\neq x} xy \frac{1}{4} \left(\frac{2}{\pi}\sigma^2 \right)\\ &= \frac{\sigma^2}{6} h(h+1)(2h+1) \left(1 - \frac{1}{\pi}\right)\\ &= \frac{\sigma^2}{24} \left(1 - \frac{1}{\pi}\right) d_0(d_0^2 - 1) .\end{aligned}$$

In calculating the averages, I apply that the probability $P(q > 0) = \frac{1}{2}$, $\mathrm{E}(q^2) = \sigma^2$ and for $x \neq y$ the probability $P(q(x) > 0 \wedge q(y) > 0) = \frac{1}{4}$. The general rule that $\mathrm{E}(q(x)q(y)) = \mathrm{E}(q(x))\mathrm{E}(q(y))$ for independent variables is employed, and we note that $\mathrm{E}(q) = \sqrt{2/\pi}\sigma$ (see Eq. 2.6). For two normal distributed random variables X and Y with means $\mu_{x/y}$, standard deviations $\sigma_{x/y}$ and correlation ρ the variance of their ratio can be estimated as

$$\operatorname{Var}(Y/X) \approx \sigma_x^2 \mu_y^2/\mu_x^4 + \sigma_y^2/\mu_x^2 - 2\rho\sigma_x\sigma_y\mu_y/\mu_x^3 \qquad (2.8)$$

under certain conditions (Hayya et al. 1975). Let $Y = \sum_x xq$ and $X = \sum_x q$. The estimate applies because both random variables are roughly normal distributed for $d_0 \geq 3$, the coefficient of variation σ_y/μ_y is infinite, $\sigma_x/\mu_x = \sqrt{(\pi - 1)/d_0}$ is sufficiently small at least for $d_0 \gg 1$ and $X \geq 0$. $Y = \sum xq$ is sufficiently mixed to be independent from $X = \sum q$, hence $\rho \approx 0$, and the first term drops out with $\mu_y = 0$. It follows that the variance of the combined variable

$$\operatorname{Var}\left(\sum_x xq \Big/ \sum_x q\right) \approx \left(\operatorname{Var}\left(\sum_x xq\right) \Big/ \mathrm{E}^2\left(\sum_x q\right)\right) = \frac{\pi}{12}\left(1 - \frac{1}{\pi}\right)\left(d_0 - \frac{1}{d_0}\right) . \qquad (2.9)$$

Multiplying its square root with $\sqrt{\frac{2n}{\pi}}$ gives the mean distance estimate for a random walk with n steps as above:

$$\xi_n = \sqrt{\frac{n}{6}\left(1 - \frac{1}{\pi}\right)\left(d_0 - \frac{1}{d_0}\right)}. \tag{2.10}$$

The necessary transport distance depends on the initial cell placement, where we chose a square grid with grid distance d_1. Therefore, the maximal distance of an extended ridge line to a cell in the first iteration is $\frac{1}{2}d_1$, for a straight horizontal or vertical feature. The maximal distance on one particular side is d_1, however. If there is a chance that the brightness profile of a feature is significantly perturbed by a neighbouring object, for example for a galaxy-type lens where an arc is situated inside the lensing galaxy, it is sensible to consider the larger distance d_1 as the worst case and choose appropriately small values for d_1.

The number of steps is limited from above as well: since a large fraction of arcs do not display a uniform intensity distribution along their length, an arc's substructure can eventually pull all surrounding cells towards it given enough iterations. Yet the second moments in A used to determine the orientation of cells (see 2.2.3) do not vary strongly on scales significantly smaller than d_0. Therefore, a cluster of cells in close proximity to a single point contains little more information than one cell, and exactly as much if all cells accrete on a single point. In contrast, it is ideal for the evaluation of features in 2.2.4 if cells are equally distributed over the length of a feature, where the amount of information contained in the cells becomes maximal.

Optionally, this step also contains a filter: cells near to bright features such as stars and foreground galaxies are continuously transported into the direction of the feature's centre even if their initial distance is well above the scale size, and the resulting cell aggregates are likely to produce false positives. Hence an upper limit for the transport distance set by the user can be used to disregard these cells from the start. A typical threshold distance would be $d_{\max} \approx 0.7 d_0$. However, using this filter can remove unusually bright arcs from consideration as well.

Finally, I use the Chebyshev distance $\max_i(|x_i - x_{c,i}|) \leq \frac{1}{2}d_0$ to determine the pixel set A for the two-dimensional integrals, which is computationally efficient but enhances the average diagonal step-size by a factor of $3/2\sqrt{2} \approx 1.06$ compared to the horizontal or vertical step-size for the same absolute slope – and diagonal respectively horizonal and vertical intensity gradients. Empirically, this small bias compared to using an Euclidean metric can be neglected.

2.2.3 Cell Orientation

A straightforward way to determine the orientation of any image feature is to use the complex ellipticity constructed from the tensor of second brightness moments

$$Q_{ij} = \frac{\int_A (x_i - \bar{x}_i)(x_j - \bar{x}_j) q(I(\boldsymbol{x}))\, \mathrm{d}^2 x}{\int_A q(I(\boldsymbol{x}))\, \mathrm{d}^2 x} , \tag{2.11}$$

where $i, j \in \{1, 2\}$. Using Q_{ij}, the complex ellipticity is defined as

$$\chi = \frac{Q_{11} - Q_{22} + 2\mathrm{i}Q_{12}}{Q_{11} + Q_{22}} \tag{2.12}$$

and in the idealised case that A actually contains an ellipse with $q(I) > 0$ and background level $q(I) = 0$, its axis ratio $r \leq 1$ and orientation of the semi-major axis relative to the x_1 axis ϑ can be obtained through the relation:

$$\chi = \frac{1 - r^2}{1 + r^2} \exp(2\mathrm{i}\vartheta) . \tag{2.13}$$

Among other applications, these relations were used to measure the ellipticities of galaxies by Kaiser & Squires (1993) in their seminal work on weak lensing mass reconstruction. Instead of measuring the orientation of a complete image object, we use it to determine the local orientation inside each cell's area, which ideally coincides with the orientation of an arc's ridge line for the cells that were transported sufficiently close to it, and to align each cell's orientation. As ϑ is invariant under $q(I)$ translation, our implementation of the algorithm uses the minimal intensity I_{min} in A instead of $\bar{I}$ to constrain the influence of intensity outliers:

$$q(I) = \begin{cases} I - I_{\mathrm{min}}, & \text{for } I > I_{\mathrm{min}} \\ 0, & \text{else.} \end{cases} \tag{2.14}$$

There are perhaps two things worthy of note regarding this third step in the detection routine:

First, the phase of χ in $\mathbb{C}$ is twice the angle ϑ in image space (see 6.1 for an illustration), which is compatible with ellipses being invariant under 180° in contrast to 360° rotations. A numerically sound method to bisect the phase and obtain a vector $\boldsymbol{e}$ parallel to the semi-major axis is to compute

$$\boldsymbol{d} = \begin{cases} (\chi_1 + |\chi|, \chi_2) & \text{for} \quad \chi_1 \geq 0 \\ (\chi_2, |\chi| - \chi_1) & \text{for} \quad \chi_1 < 0 \end{cases} \quad \text{and} \quad \boldsymbol{e} = \frac{\boldsymbol{d}}{|\boldsymbol{d}|} , \tag{2.15}$$

where the orientation $\boldsymbol{e} = (e_1, e_2)$ is stored for each cell (see 6.1).

Second, a determination of the complex ellipticity in a noisy image inside a square aperture A introduces a significant bias in the phase of χ towards the diagonal direction in image space. Responsible is the noise in the four corners of the square outside the $\frac{1}{2}d_0$ radius disk around $\boldsymbol{x}_c$ that is intrinsically isotropic: the corners are dominated by points with low $|x_1^2 - x_2^2|$

and high $|x_1 x_2|$ contribution to the integral terms for $Q_{11} - Q_{22}$ and $2iQ_{12}$ respectively. To give an idealised example, I 'integrate' only over the four diagonal points $x_1 = \pm 1$ and $x_2 = \pm 1$ around $\bar{\boldsymbol{x}} = \boldsymbol{0}$:

$$\begin{aligned} Q_{11} &= \frac{1}{I_{\text{total}}} \sum_{x_1=-1}^{+1} \sum_{x_2=-1}^{+1} x_1^2 \ \ I(\boldsymbol{x}) = \frac{1}{I_{\text{total}}} (I_{--} + I_{-+} + I_{+-} + I_{++}) = 1 \\ Q_{22} &= \frac{1}{I_{\text{total}}} \sum_{x_1=-1}^{+1} \sum_{x_2=-1}^{+1} x_2^2 \ \ I(\boldsymbol{x}) = \frac{1}{I_{\text{total}}} (I_{--} + I_{-+} + I_{+-} + I_{++}) = 1 \qquad (2.16) \\ Q_{12} &= \frac{1}{I_{\text{total}}} \sum_{x_1=-1}^{+1} \sum_{x_2=-1}^{+1} x_1 x_2 I(\boldsymbol{x}) = \frac{1}{I_{\text{total}}} (I_{--} - I_{-+} - I_{+-} + I_{++}) = ? \end{aligned}$$

Since $Q_{11} = Q_{22}$ regardless of the individual intensities, only the imaginary part of $\chi = iQ_{12}$ survives with phase 90° or 270° respectively orientation parallel to 45° or 135° in the image. An increased level of noise necessarily results in greater intensity variations along the two diagonals and therefore an enhanced bias.

The apparent solution is to use the Euclidean instead of the Chebyshev distance metric. However, particularly for small radii of the integration disk A and correspondingly a large fraction of pixels on its border, the algorithm now has to account for pixels which are only partly covered by the disk. This is implemented in the form of an additional weighting factor $A_p(\boldsymbol{x})$ for the pixel coverage and appropriately modified centre coordinates $(x_1^p(\boldsymbol{x}), x_2^p(\boldsymbol{x}))$ for each pixel:

$$Q_{ij} = \frac{\sum_{\boldsymbol{x} \in A} (x_i^p - \bar{x}_i)(x_j^p - \bar{x}_j) q(I(\boldsymbol{x})) A_p(\boldsymbol{x})}{\sum_{\boldsymbol{x} \in A} q(I(\boldsymbol{x})) A_p(\boldsymbol{x})} . \qquad (2.17)$$

To save processing time, the above weighting factors are computed only once, at the affordable expense of having to use discrete values for the cells' centre coordinates. I give the integrals pertaining to the necessary weighting factors in 6.2.

2.2.4 Finding Coherent Features

The fourth step in the initial sketch of the algorithm is to locate paths of similarly oriented cells, where we utilise the orientation $\boldsymbol{e}$ of each cell and its spatial location $\boldsymbol{x}_c$. Since all cells are organised in an array with corresponding index running from left to right and top to bottom over the complete square grid during initial cell placement, and relative cell distances do not change significantly during cell transport, the efficient selection of cell neighbours is trivial. Henceforth, I enumerate cells with superscripts i and j and write $\mathcal{N}^i$ for the neighbourhood of cell i.

For any choice of d_0 and d_1 a square of cells encompassing $d_{\mathcal{N}} = 1 + 2\max(1, \lfloor d_0/d_1 \rfloor)$ rows and columns in the initial cell grid centred onto cell i is included into the neighbourhood $\mathcal{N}^i$, excluding cell i itself. Assuming that cells overlap, or $d_1 \leq d_0$, this effectively collects cells up to an initial distance of d_0.

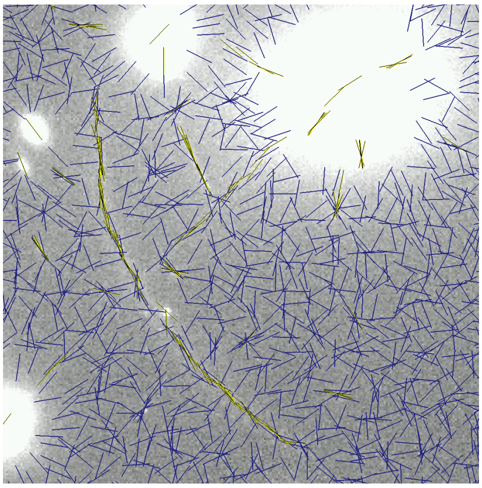

Figure 2.4: Cell orientations for a giant arc in Abell 2390. Cells with coherence values above $\frac{1}{2}$ are marked in yellow.

To write down a measure of spatial coherence, we introduce slashed coordinates in a Cartesian coordinate frame centred on $\boldsymbol{x}_c^i$ and with first and second axes parallel to $\boldsymbol{e}^i$ and $\boldsymbol{e}_\perp^i := (-e_2^i, e_1^i)$ respectively. The distance Δ of a cell j in the neighbourhood of cell i in this coordinate frame is:

$$\boldsymbol{\Delta} = \boldsymbol{x}_c^j - \boldsymbol{x}_c^i \quad \text{and} \quad \boldsymbol{\Delta}' = \begin{pmatrix} \Delta_1 e_1^i & + & \Delta_2 e_2^i \\ -\Delta_1 e_2^i & + & \Delta_2 e_1^i \end{pmatrix} . \tag{2.18}$$

If cells i and j are on a common path aligned with the orientation of cell i, we clearly expect a low perpendicular distance Δ_2'. We normalise this distance with the initial cell distance d_1 to get a measure of coherence

$$c_\Delta^{ij} = \begin{cases} 1 - \frac{|\Delta_2'|}{d_1} & \text{for} \quad |\Delta_2'| < d_1 \\ 0 & \text{else} \end{cases} . \tag{2.19}$$

To get a measure of cell alignment, we simply use the scalar product between orientation vectors:

$$c_e^{ij} = |\boldsymbol{e}^i \cdot \boldsymbol{e}^j| . \tag{2.20}$$

Since both coherence measures are constructed to live in the interval [0, 1] – where 0 denotes minimal coherence – we write the total coherence between cells i and j as the product of each,

$$c^{ij} = c_\Delta^{ij} c_e^{ij} , \tag{2.21}$$

whose arithmetic mean we use to determine for any particular cell i its coherence with its cell neighbourhood $\mathcal{N}^i$:

$$c^i = \frac{1}{|\mathcal{N}^i|} \sum_{j \in \mathcal{N}^i} c^{ij} . \tag{2.22}$$

At this point, different weighting factors can be considered, for example one could weight cells with low Δ_1' less to account for the lack in information such a cell can provide. Since this is a matter of some arbitrariness, it is more fruitful to look at the general behaviour of the coherence coefficients.

Using the pixel weighting discussed in 2.2.3 and on a flat background with uncorrelated pixel noise, the orientational coherence c_e^{ij} has a mean value of $2/\pi \approx 0.637$ and a standard deviation of $\sqrt{1/2 - 4/\pi^2} \approx 0.308$ for non-overlapping cells. If the cell distance falls below d_0, however, the second moments of each cell will no longer be independent; the expected value finally reaches unity if both cells occupy the same position. For partial overlap, I numerically estimated the expected value and the standard deviation of c_e^{ij} assuming uncorrelated and Gaussian pixel noise, and we see the results in Figure 2.5.

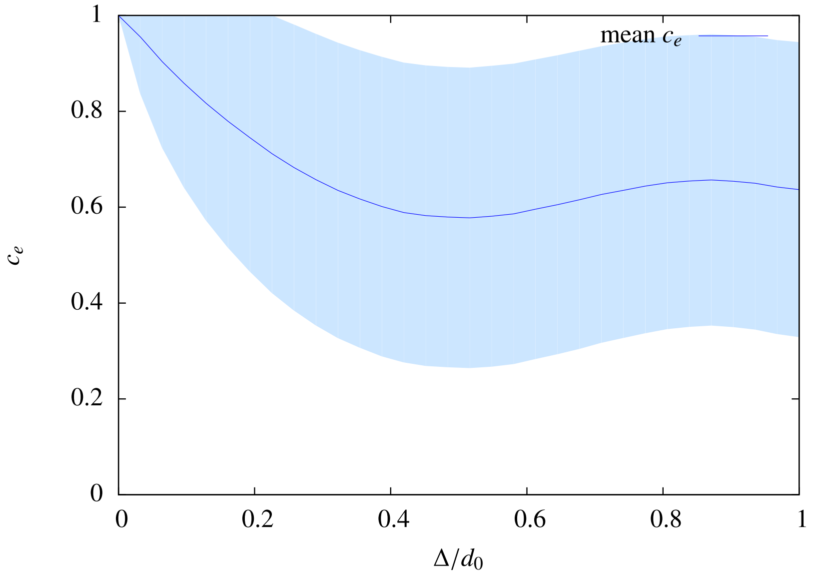

Figure 2.5: Mean orientational coherence c_e^{ij} versus relative cell distance Δ/d_0 on a flat but noisy background. The 1σ standard deviation is indicated by the shaded area.

Due to the anisotropy of the initial cell distribution, it is difficult not to suspect a bias in the spatial coherence. To determine this and the general statistics of the coherence on a flat, random background, let us first observe the mean spatial coherence c_Δ^{ij}.

Using the above estimate of the mean absolute displacement of a cell $\xi_n = \frac{1}{2}\sqrt{n(1-1/\pi)(d_0 - 1/d_0)/6}$ (see Eq. 2.10), the general ansatz from Eq. 2.4 can be applied again: the mean spatial coherence of each cell $j \in \mathcal{N}^i$ is the integral of the c_Δ^{ij} coherence value weighted with a probability density function $f^{ij}(\boldsymbol{x})$ in an area $\{\boldsymbol{x} \,\big|\, |x_2'| < 1\}$. Neglecting the approximately 6% bias towards diagonal cell transport and under the assumption of independent random walks, the probability density functions take the form of two-dimensional Gaussian functions centred on the initial cell positions $\boldsymbol{x}_c^j$:

$$f^{ij}(\boldsymbol{x}) = \frac{1}{2\pi^2 \xi_n^2} \mathrm{e}^{-\frac{(\boldsymbol{x}-\boldsymbol{x}_c^j)^2}{2\pi\xi_n^2}} , \tag{2.23}$$

where I applied that the standard deviation of f^{ij} equals ξ_n multiplied by $\sqrt{\pi/2}$ (see Eq. 2.6) and multiplied by $\sqrt{2}$ to account for the uncertainty in relative positions between cell i and its neighbours in contrast to the uncertainty in absolute positions. Figure 2.6 illustrates the necessary integration area for eight cells in $\mathcal{N}^i$. In order to solve the integrals it is convenient to go into the slashed coordinate system and employ the isotropy of f^{ij} and its product form to reduce the two-dimensional integral terms $c_\Delta^{ij}(\boldsymbol{x}) f^{ij}(\boldsymbol{x})$ to one-dimensional terms dependent only on x_2, which can be solved analytically.

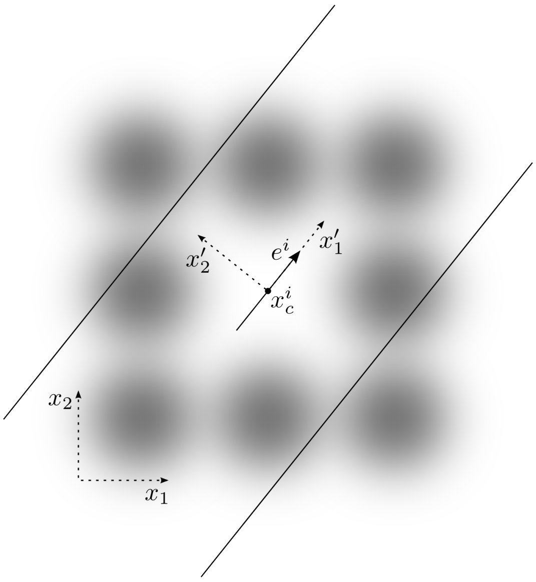

Figure 2.6: Cell i with 8 neighbours and associated probability density functions. To determine the bias in c_Δ, the integration is carried out in-between the two diagonal lines where the $c_\Delta^{ij} > 0$.

In approximation I assume that c_e and c_Δ are independent and carry out the integral over $f^{ij} c_e^{ij}$ separately. This assumption is violated when extended features are present or cell distances are small compared to d_0, but on a flat background the error is negligible compared to the noise contribution. The relevant results are shown in Figure 2.7 for a realistic scenario of an initial cell distance $d_1 = (d0 - 1)/2$ and correspondingly $|\mathcal{N}^i| = 24$ neighbours: on the left side, we see for two exemplary scale sizes and for $d_0 \to \infty$ how the mean coherence varies from a minimal value for horizontally and vertically aligned cells to a maximum value for diagonal orientations. It is evident from the plot on the right side that the angular average does not vary significantly and is approximately 0.125 for most scales. The standard deviation in c was calculated correspondingly and can be approximated roughly by $c/d_0^{1/3}$. As the upper limit for $c_e^{ij} \leq 1$ limits its maximal positive error, I also included the standard deviation of the spatial coherence alone in the blue shaded region. Also on the right side, the ratio of maximal to minimal mean coherence is displayed and increases with increasing scale size on account of the square root dependency of xi_n on d_0, up to a rather significant bias of approximately 1.4 for a – extremely large – scale size of 99.

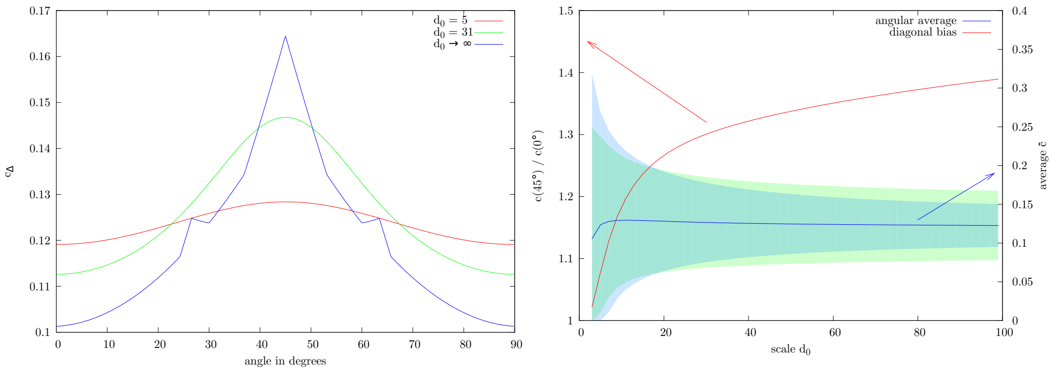

Figure 2.7: Mean coherence c on a flat, random background. Left: variation of c with the angle spanned by $\boldsymbol{e}^i$ and the x_1 axis for scale sizes 5 and 31 and in the limit of an infinite scale size where the mean displacement $\xi_n \rightarrow 0$. Right: bias $c(45°)/c(0°)$ (left ordinate) and angular average $\bar{c}$ (right ordinate) ± the angular average of the standard deviation (green shaded) for scale sizes 3 to 99. The blue shaded region accounts only for the standard deviation in the spatial coherence assuming $c_e^{ij} \equiv 1$.

The bias, unfortunately, cannot be completely accounted for by simply rescaling the spatial coherence for each orientation in $\boldsymbol{e}^i$, since the bias in the accretion velocity onto image features is comparatively low with approximately 6% (see 2.2.2) and the accretion distance is often greater than d_1, where the anisotropy in neighbouring cell counts diminishes. Increasing the normalisation distance in c_Δ to higher values than d_1 reduces the bias, but increases the noise contribution from cells not on a ridge line. However, modifying the initial cell distribution to increase isotropy, for example by using a hexagonal grid, will alleviate this potential problem and might allow for fainter detections under some conditions. That said, the background bias was not an apparent influence in the arcfinder applications carried out for this thesis.

A lower threshold c_{th} is set by the user to indicate what level of coherence c^i suggests an underlying feature and marks a cell as valid. All other cells are excluded from further processing. From the above analysis, it is immediately apparent that scale sizes $d_0 < 10$ – assuming $d_1/d_0 \approx \frac{1}{2}$ – require relatively high threshold settings of $c_{\text{th}} \approx 0.35$ to avoid a large number of spuriously valid cells even on a perfectly flat background with uncorrelated noise, while for example a low threshold of $c_{\text{th}} \approx 0.25$ will ensure that less than one in 80 cells is falsely counted as valid for a scale size of $d_0 = 31$ and above. A reasonable value for the threshold is $c_{\text{th}} \approx 0.5$.

2.2.5 Object Generation

The final act in the initial detection algorithm is the creation of preliminary detections by grouping cells into objects, using in principle a friends of friends algorithm.

As indicated in the first paragraph of 2.2.4, cells are organised in an array and the index $i = i_x + i_y n_x$ can be decomposed into i_x- and i_y- coordinates on the square grid, where n_x is the number of cells in one row. For each valid cell i with $c^i \geq c_{\mathrm{th}}$, we can therefore apply a modified Bresenham's line-drawing algorithm (Bresenham 1965) to create a new neighbourhood $\tilde{\mathcal{N}}^i$ of cells extended by $2 \cdot \max(1, \lfloor d_0/d_1 \rfloor)$ cells into the directions parallel to the cell's orientation $\boldsymbol{e}^i$ and by $1 + \frac{1}{2}\max(1, \lfloor d_0/d_1 \rfloor)$ cells perpendicular to it. As above, $i \notin \tilde{\mathcal{N}}^i$.

To facilitate the friends of friends algorithm, each cell i is assigned an object index k^i which is initially set to -1, and a new threshold $\tilde{c}_{\mathrm{th}}$ is set by the user equivalent to a linking radius. With these ingredients, the actual scheme of combining coherent cells into objects is straightforward, as is shown in Listing 2.1.

Listing 2.1: pseudo code for object generation

```
for each valid cell $i$
    for each valid cell $j$ in $\tilde{\mathcal{N}}^i$
        $c^{ij}$ = coherence($i$,$j$)
        if $c^{ij} < \tilde{c}_{\mathrm{th}}$
            continue // on to the next cell
        if $k^i < 0$ and $k^j < 0$
            create object O($k$)
            add $i$ to O($k$)
            add $j$ to O($k$)
            $k^i = k^j = k$
        else if $k^j < 0$
            add $j$ to O($k^i$)
            $k^j = k^i$
        else if $k^i < 0$
            add $i$ to O($k^j$)
            $k^i = k^j$
        else
            add all cells in O($k^j$) to O($k^i$)
            delete object O($k^j$)
```

The result of these steps is illustrated in Figure 2.8. Before the actual postprocessing starts, objects with few cells and low estimated length are deleted, where both thresholds N_{cells} and L_{min} can be set by the user to avoid obvious spurious detections. The length estimate follows the method detailed in (the appendix of) Seidel & Bartelmann (2007).

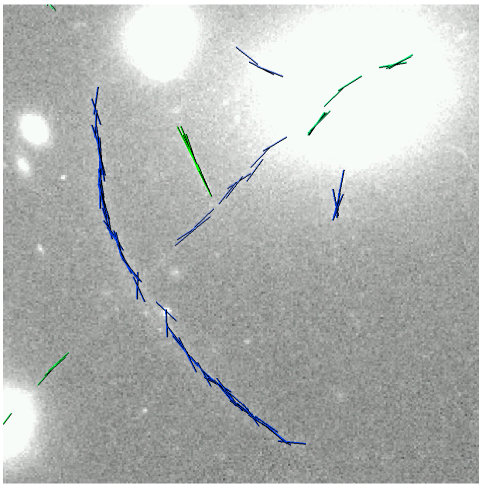

Figure 2.8: Seven initial detections in the vicinity of Abell 2390 in an HST WFPC2 observation. Two spurious detections are generated by an extended diffraction spike and three more by cluster foreground (postprocessing as described in 2.3 reliably removes these spurious detections).

2.3 Postprocessing

The aim of the subsequent postprocessing methods is to filter out false positives, and to characterise detections beyond the number, location and orientation of the cells they are composed of. Spurious detections are caused by a range of image degrading effects as well as astronomical objects other than strongly lensed galaxies. This is reflected in the number of different sub-algorithms used to detect them, which are discussed in 2.3.2, 2.3.3, 2.3.9 and 2.3.10. The most common arc descriptors require the shape of the area in the image that the lensed galaxy is projected into, but the cells in a valid detection are selected so that they are part only of its ridge line: by construction, they contains no information on its width. The graph generation and active contour segmentation methods used to obtain the projected area shapes are discussed in 2.3.5 and below.

2.3.1 Background & Noise Estimation

The previous image analysis required neither the background intensity $I_{\mathrm{bg}}(\boldsymbol{x})$ nor the noise $\sigma(\boldsymbol{x})$ in the image. Some of the postprocessing methods (see 2.3.5 in particular) however rely on their values. Since noise respectively χ^2 images from the data reduction pipeline are not always available, both values are computed as part of the arcfinder algorithm.

The algorithm typically does not need to know the background and noise for all image pixels and hence they are computed only when needed. For each coordinate $\boldsymbol{x}$, the image attributes are linearly interpolated between up to four – less if $\boldsymbol{x}$ is on the border of the image – closest square areas S with side-length $10 \cdot d_0$ whose the background and noise levels are estimated only once and stored for later recall.

The evaluation inside each square S again involves a segmentation into at most 10×10 squares S' of at least 10 pixels side-length each and average intensities I'_{bg}. The background intensity $I_{\mathrm{bg}}(S)$ of S is then set to the supremum of the faintest 20% intensities

$I'_{\mathrm{bg}} = F_I^{-1}(0.2)$ in order to avoid including bright features[2], where F_I^{-1} is the inverse of the cumulative distribution function of I in S'.

Each square S' is locally flat-fielded using two-dimensional linear regression, resulting in intensities I'_{ff} with zero mean and no apparent slope in S'. Using these intensities, noise levels are determined as $\sigma'^2(\boldsymbol{x}) = (1/|S'|)\sum_{\boldsymbol{x}} I'^2_{\mathrm{ff}}(\boldsymbol{x})$. If small scale pixel correlations with an approximate correlation length $d_{\mathrm{corr}} \in \mathbb{N}$ are present, they are trivially accounted for by including only pixels whose coordinates fulfil the conditions $x_1 \bmod d_{\mathrm{corr}} = 0$ and $x_2 \bmod d_{\mathrm{corr}} = 0$, at the expense of some certainty in σ' and by using larger sub-squares S' if necessary. The noise $\sigma(A)$ is set to the median over all σ' values.

2.3.2 Histogram Equalisation

We find that it is convenient to scale the intensity inside the disc shaped area A of a cell by histogram equalisation before any further analysis is applied (see 2.3.3 below). In general, histogram equalisation maps each intensity $I(\boldsymbol{x})$ to the normalised fraction $F(I)$ of pixels in A at or below that intensity, where F is the cumulative distribution function. Using the characteristic function

$$\chi_S(x) = \begin{cases} 1, & \text{for } x \in S \\ 0, & \text{for } x \notin S \end{cases} \tag{2.24}$$

where S is a subset of A and $x \in A$, this can also be written in the form:

$$I_{\mathrm{H}}(I_{\mathrm{ref}}) = \frac{1}{|A|} \sum_{\boldsymbol{x} \in A} \chi_{I(\boldsymbol{x}) \leq I_{\mathrm{ref}}} . \tag{2.25}$$

Unless some pixels have identical intensities, all intervals of length $1/|A|$ in I_{H} contain one pixel: the histogram of I_{H} is a constant function. More importantly, this scaling maps all kernels whose intensity drops isotropically – respectively symmetrically – with distance into the same function. This is illustrated in Figure 2.9: the first two of the four plots display histogram scaling on two symmetrically and with respect to $|x - 50|$ strictly monotonously falling functions. Despite their significant difference in width, the result is equal.

By definition, this is true for all intensity distributions where the fraction $F(I)$ of pixels below an intensity $I(\boldsymbol{x})$ share the same dependence on $\boldsymbol{x}$, or in this case the distance $|\boldsymbol{x} - \boldsymbol{x}_{\mathrm{centre}}|$ from the centre coordinate $\boldsymbol{x}_{\mathrm{centre}} = 50$. The third plot illustrates histogram scaling for a heavily skewed intensity distribution, where the flat interval on the right forces a steep increase on the left side of the scaled function. The effect of oscillations is shown in the final plot: they clearly disturb the intermediate intensity segment of the scaled function while they leave the lower and higher segments intact. It should be noted that the triangular shape of the scaled function is characteristic for intensity distributions varying in one dimension, while similar isotropic functions in two dimensions are scaled to one minus a parabola.

---

[2] If no bright features are present, using the supremum of the faintest 20% I'_{bg} values instead of the median can systematically underestimate $I_{\mathrm{bg}}(S)$, but not more than approximately 0.05σ.

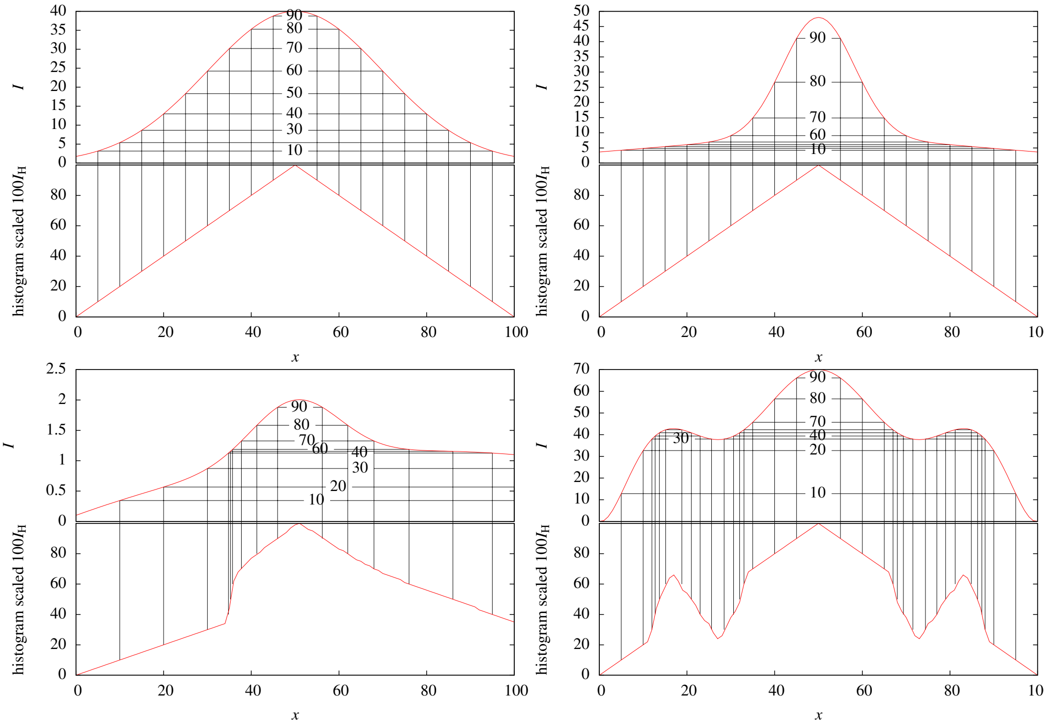

Figure 2.9: The top diagrams show four progressively irregular intensity functions $I(x)$, and the diagrams directly below the corresponding histogram equalised functions rescaled by a constant factor of 100. Each horizontal line in the top diagram marks an increase by 10 in the number of pixels below that intensity, which is denoted also by the labels in-between, and a corresponding increase by 10 of the histogram scaled function.

2.3.3 Cell Reevaluation

I mentioned in passing that groups of cells transported to the exact same location do not contain more information than a single cell in the penultimate paragraph of 2.2.2. Nevertheless, point sources such as distant and unresolved galaxies or faint stars pull together clusters of cells none or few pixels apart that generate high coherence values c. Single clusters are removed by the minimal length threshold applied at the end of the initial detection algorithm (see 2.2.5), but pairs or groups of point sources can still generate extended objects.

It is therefore helpful to distinguish cells on a point source from those on a ridge line. Our method is based on kernel fitting, where the advantage of mapping kernels of different width into one by histogram equalisation becomes apparent, since it removes the width from parameter space and thus gives a direct solution by linear regression for spatially fixed kernels. We fit a function composed of a constant offset and two distinct kernels: a smoothed line kernel to estimate how significant a ridge line feature is and an isotropic point kernel to fit the significance of a contained point-like source.

True to its name, the cell reevaluation method, including the initial histogram equalisation, uses only pixels inside a disc shaped cell area A with radius $r_A = \frac{1}{2}d_0$ around $\boldsymbol{x}_c$. To reduce the impact of noise, we subtract an offset of 0.6 from the histogram scaled intensities I_H and set all negative values to zero, resulting in intensities I_C.

To construct both kernels, the centre of brightness $\bar{\boldsymbol{x}}$ (see Eq. 2.1) and the orientation $\boldsymbol{e}$ (see Eqs. 2.11-2.15) are determined and define the centre of the point and the orientation of the line kernel. Consequently, the point kernel $K_\mathrm{point}(\boldsymbol{x})$ follows a function $I_0^\mathrm{point} - (\boldsymbol{x} - \bar{\boldsymbol{x}})^2$ and the line kernel $K_\mathrm{line}(\boldsymbol{x})$ a function $I_0^\mathrm{line} - \boldsymbol{e}^\perp \cdot (\boldsymbol{x} - \boldsymbol{x}_c)$, where the kernel offsets I_0^point and I_0^line are set such that positive values cover an area of about $0.4A$ and negative values are set to zero – corresponding to the rescaling of I_H to I_C that maps approximately $0.6|A|$ pixels to zero. Both kernels are then normalised to a total intensity of one.

Using linear regression we model I_C with $I_0 + c_\mathrm{point}K_\mathrm{point} + c_\mathrm{line}K_\mathrm{line}$, where c_point, c_line and I_0 are free parameters. A positive value of the line kernel coefficient c_line that is in the range of or even larger than the point kernel coefficient c_point generally calls for a strongly elongated feature, while we consider it a good indicator for a point source if c_line is small compared to c_point. However, if a negative c_point balances a positive c_line of similar absolute value, the centre part of the line kernel component is often nullified by the point kernel to simulate two local maxima: the negative point kernel component indicates a cell located between two point sources. A significant intensity slope in A is conveniently fitted by a point kernel located in the higher intensity side of A, which is used to invalidate cells that are pulled towards bright stars or galaxies from a greater distance.

Empirically, we say a cell is valid if and only if

$$c_\mathrm{line} > 0\,, \tag{2.26}$$

$$(c_\mathrm{line} + c_\mathrm{point})/c_\mathrm{line} \geq 0.6 \text{ and} \tag{2.27}$$

$$(c_\mathrm{line} - c_\mathrm{point})/c_\mathrm{line} \geq R_\mathrm{line}\,, \tag{2.28}$$

where a good value for the 'ridgeness' threshold is $R_\mathrm{line} \approx -0.6$.

To sort out cells generated by random background noise, the significance of a cell is estimated by transforming the maximum of the signal component $I_0 + c_\mathrm{line}K_\mathrm{line}$ back into an unscaled intensity I_line and computing the ratio

$$\frac{I_\mathrm{line} - I_\mathrm{bg}}{\sigma}\,, \tag{2.29}$$

where I_bg and σ are the background and pixel noise set in 2.3.1. Cells below a user set threshold of $S_\mathrm{th} \approx 2.0$ are invalidated.

After this step, the cell count and length filter mentioned at the very end of 2.2 are applied again on the remaining valid cells to remove further spurious detections.

2.3.4 Generation of Path Graphs

The focus of the detection up to this point was on the ridge lines of strongly elongated features, but they were defined only by a set of cells distributed in a relatively undefined fashion along its length during object generation (see 2.2.5). Certainly for the active contour segmentation method described in 2.3.5, but also for the spiral galaxy filter (see 2.3.10), an efficient method to generate path graphs from arbitrary sets of cells becomes necessary.

We use the Delaunay triangulation to construct a connected graph G including all valid cells in the initial detection, and a modified Dijkstra's algorithm to find the most probable ridge lines as path subgraphs of G.

Delaunay Triangulation

The Delaunay triangulation (Delaunay 1934) subdivides the convex hull of an arbitrary set of sites on a plane into triangles so that no site is inside the circumcircle of any triangle[3] (see Figure 2.10). It also maximises the minimal angle in all the created triangles compared to any other possible triangulation, hence avoiding extended, skinny triangles, and tends to connect points which are close together: in fact, if there is a circle passing through two points such that no further points are in its interior, those two points are connected by an edge in the Delaunay triangulation. The cells on a ridge line are typically close neighbours, therefore this property helps to create undisrupted paths through these cells and to conserve the geometry of the ridge line.

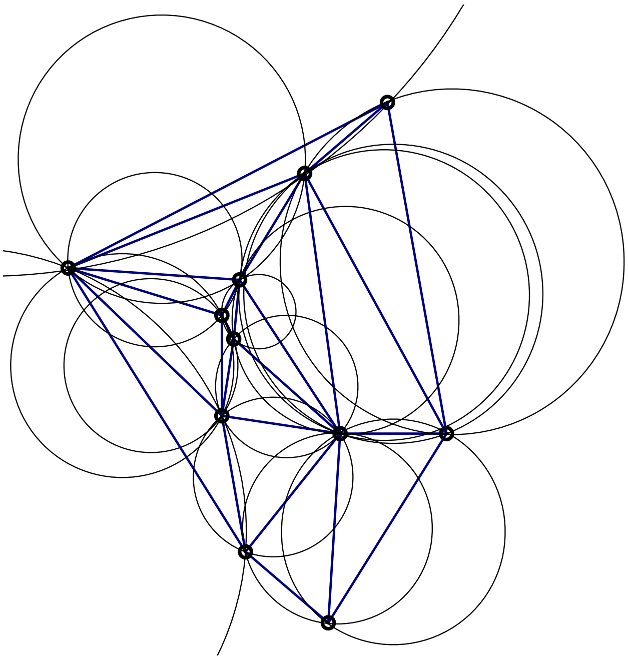

Figure 2.10: Delaunay triangulation of a small object with eleven cells, including three outliers, and corresponding circumcircles.

---

[3] It follows that the Delaunay triangulation is the dual graph of the two-dimensional Voronoi tessellation.

We construct Delaunay triangulations using Fortune's sweepline algorithm (Fortune 1986; Austin 2006), which has a low computational complexity of order $N \log N$ elemental operations.

Modified Dijkstra's Algorithm

Dijkstra's Algorithm (Dijkstra 1959) is a well known method for choosing the shortest connection between two nodes in a graph with non-negative edge lengths. One possible implementation, that we use, is given in Listing 2.2.

Listing 2.2: pseudo code for Dijkstra's algorithm

```
for each node $n$ in graph $G$ // initialisations
        distance[$n$] = $\infty$
        previous[$n$] = $-1$
        organised[$n$] = false
distance[$n_{\text{start}}$] = 0 // start at distance zero
add $n_{\text{start}}$ to border $B$ // all unorganised nodes with ...
while $B$ is not empty // ... finite distances are in $B$
        $n$ = node with minimal distance in $B$
        remove $n$ from $B$
        organised[$n$] = true
        for each node $n'$ connected with $n$
                if organised[$n'$] == true
                        continue
                newdist = distance[$n$] + length of edge between $n$ and $n'$
                if newdist < distance[$n'$]:
                        distance[$n'$] = newdist
                        if previous[$n'$] < 0 // $n'$ is not already in $B$
                                add $n'$ to $B$
                        previous[$n'$] = $n$
```

Dijkstra's algorithm can be easily understood by an illustration: we imagine all the nodes in the graph connected by edges instead as beads connected with strings whose lengths are equal to the edges' distances. If the graph is pulled up, hanging by its starting bead, the shortest connection to any other bead is marked by the tense strings keeping it aloft. The algorithm distincts three types of nodes: organised nodes with known shortest connection, border nodes connected to an organised node, and unconsidered nodes. Similar to slowly lifting up the graph by its starting bead, only the distances of nodes in the border B connected to at least one organised node – a bead hanging in the air whose shortest path is established – are considered in one iteration. At the start of each step, the node with minimal distance from the starting node in B becomes organised – the bead is lifted from the ground – and accordingly updates the distance of any node it is connected to with its own distance plus the connecting edge's length, unless the node was updated before with a shorter distance, and adds it to the border if it was previously unconsidered.

In Figure 2.10 the astute observer will immediately spot a path from the lowermost to the topmost cell, passing through five intermediate cells slightly to the left. This example is typical for the general case that the most probable path of a ridge line, which is smoothly curved and contains a relatively large number of cells, is rarely the shortest path in an Euclidean metric. We therefore choose a different distance measure, which incorporates cell orientations:

$$L_{\text{edge}} = \frac{\Delta_1'^2 + \Delta_2'^2}{|\Delta_1'|} \frac{\Delta_1''^2 + \Delta_2''^2}{|\Delta_1''|}\,, \tag{2.30}$$

where $\boldsymbol{\Delta}'$ is the distance of two cells i and j in the coordinate frame of cell i (see Eq. 2.18) and $\boldsymbol{\Delta}''$ is the same distance in the corresponding coordinate frame of cell j. If the cells are aligned, this length increases with distance squared.

It is not directly possible to include the angle between edges into the path length in the above implementation, since it depends on the positions of three consecutive nodes. We therefore modified the algorithm to store the `distance`, `previous` node and `organised` state of each node separately in a sub-node for each preceding organised node. When the distance between a sub-node and a node is determined, the length is penalised by a multiplier of $1 + (\boldsymbol{e}_{\text{edge 1}} - \boldsymbol{e}_{\text{edge 2}})^2$, where $\boldsymbol{e}_{\text{edge 1/2}}$ are normalised edge directions.

Path Finding

Using the above method, the shortest paths are determined between all starting cells in a detection with no further object cells in one of the directions $\pm\boldsymbol{e}^i$ nearer than an upper distance limit of $4d_0$. Among this set of paths P, the one with the maximum weight

$$w_{\text{path}} = \left(\sum_{(i,j)\in P} |\boldsymbol{x}^i - \boldsymbol{x}^j| \right)^3 \Bigg/ n \sum_{(i,j)\in P} (\boldsymbol{x}^i - \boldsymbol{x}^j)^2 \tag{2.31}$$

is chosen, where n is the number of valid cells in the path. This empirically determined weighting function is maximal and equal to the total Euclidean length L of a path for evenly spaced cells and equal to L/n in the worst case – if the goal is to find plausible ridge lines – of two clusters of cells connected by one edge of length L. In practise, we implemented a slightly more elaborate weighting function that also penalises changes in curvature in a path.

In the rare case that two or more parallel paths with comparable weight are found, the initial detection is separated into multiple objects.

2.3.5 Active Contour Segmentation

Initial detections trace the ridge line of an image object, yielding its position, orientation, length and curvature, but obtaining the arc width, integrated flux and signal-to-noise ratios requires further segmentation. Most lens reconstruction techniques also need the image area covered by the arc as input for the algorithm. To this end, the approximate shape of a detection is determined using an active contour segmentation algorithm (Kass et al. 1988). The mathematical principles of active contour segmentation are detailed in 2.3.6, and the application to arc segmentation, the method of initialising the contour and the adaptions used in the arcfinder algorithm in 2.3.7.

2.3.6 Basic Contour Evolution

For the purpose of this derivation, the contour C is defined as the closed curve $\boldsymbol{c}(s) \in \mathbb{R}^2$, with $s \in [0,1]$ and $\boldsymbol{c}(0) = \boldsymbol{c}(1)$. The idea behind active contour evolution is to minimise an energy functional E_C to arrive at a smooth segmentation curve nestling up against the object. Writing $\frac{d}{ds}\boldsymbol{c}(s) = \boldsymbol{c}'(s)$ and $\frac{d^2}{ds^2}\boldsymbol{c}(s) = \boldsymbol{c}''(s)$ for the first and second derivatives of the contour function, contour evolution is governed by an energy functional

$$E_C = \int_0^1 \frac{\alpha(s)}{2}\boldsymbol{c}'^2(s) + \frac{\beta(s)}{2}\boldsymbol{c}''^2(s) + E(\boldsymbol{c}(s))\,\mathrm{d}s. \tag{2.32}$$

The $\frac{\alpha}{2}\boldsymbol{c}'^2$ term minimises the length and insures that $\boldsymbol{c}(s)$ is traversed uniformly in s, while the $\frac{\beta}{2}\boldsymbol{c}''^2$ term minimises changes in curvature. The image energy $E(\boldsymbol{c})$ connects the contour with the image and is chosen such that it becomes small at the border of the detection: for example, it could be set to the absolute difference between a threshold intensity and the pixel value at position $\boldsymbol{c}$. Calculus of variations gives the equation for the minimum condition for each value of s. Omitting parameters s for brevity, we start out with

$$\boldsymbol{\partial E_C} = \int_0^1 \alpha\boldsymbol{c}'\partial\boldsymbol{c}' + \beta\boldsymbol{c}''\partial\boldsymbol{c}'' + \frac{\partial E(\boldsymbol{c})}{\partial\boldsymbol{c}}\partial\boldsymbol{c}\,\mathrm{d}s. \tag{2.33}$$

Partial integration and using $\frac{\partial E(\boldsymbol{c})}{\partial\boldsymbol{c}}\partial\boldsymbol{c} = \nabla E\partial\boldsymbol{c}$ gives

$$\boldsymbol{\partial E_C} = \alpha\boldsymbol{c}'\partial\boldsymbol{c} + \beta\boldsymbol{c}''\partial\boldsymbol{c}' + \nabla E\partial\boldsymbol{c}\Big|_0^1 - \int_0^1 \frac{\partial(\alpha\boldsymbol{c}')}{\partial s}\partial\boldsymbol{c} + \frac{\partial(\beta\boldsymbol{c}'')}{\partial s}\partial\boldsymbol{c}' + \nabla E\partial\boldsymbol{c}\,\mathrm{d}s. \tag{2.34}$$

Since the curve is closed, the leading integration term equals zero. Taking this into account again and applying partial integration on the second term results in

$$\boldsymbol{\partial E_C} = \int_0^1 -\frac{\partial(\alpha\boldsymbol{c}')}{\partial s}\partial\boldsymbol{c} + \frac{\partial^2(\beta\boldsymbol{c}'')}{\partial s^2}\partial\boldsymbol{c} + \nabla E\partial\boldsymbol{c}\,\mathrm{d}s. \tag{2.35}$$

We can therefore write the Euler equations for E_C as:

$$-\frac{\partial(\alpha\boldsymbol{c}'(s))}{\partial s} + \frac{\partial^2(\beta\boldsymbol{c}''(s))}{\partial s^2} + \nabla E(\boldsymbol{c}(s)) = \boldsymbol{0}. \tag{2.36}$$

It is convenient to discretise the equations at this point in order to solve them numerically. I choose the following discretisation scheme:

$$\begin{aligned}
\frac{\partial(\alpha \boldsymbol{c}'(s))}{\partial s} &\rightarrow \quad \alpha_{i+\frac{1}{2}}(\boldsymbol{c}_{i+1} - \boldsymbol{c}_i) - \alpha_{i-\frac{1}{2}}(\boldsymbol{c}_i - \boldsymbol{c}_{i-1}), \qquad (2.37)\\
\frac{\partial^2(\beta \boldsymbol{c}''(s))}{\partial s^2} &\rightarrow \quad \beta_{i+1}\,(\boldsymbol{c}_{i+2} - 2\boldsymbol{c}_{i+1} + \boldsymbol{c}_i)\\
&\quad -2\beta_i \quad (\boldsymbol{c}_{i+1} - 2\boldsymbol{c}_i + \boldsymbol{c}_{i-1})\\
&\quad +\ \beta_{i-1}\,(\boldsymbol{c}_i - 2\boldsymbol{c}_{i-1} + \boldsymbol{c}_{i-2}),
\end{aligned}$$

where I used integer plus $\frac{1}{2}$ indices to illustrate that in this discretisation, the distance stress controlled by e.g. $\alpha_{i+\frac{1}{2}}$ is located between successive nodes i and $i+1$. Integer indices i run from 0 to $N-1$ and index terms j outside this range are mapped to $j \bmod N$ to arrive at a closed curve. Applying this discretisation to the Euler equations 2.36 and sorting by $\boldsymbol{c}$ coefficients we arrive at N independent equations:

$$\begin{aligned}
&\beta_{i-1}\boldsymbol{c}_{i-2} + (-\alpha_{i-\frac{1}{2}} - 2\beta_i - 2\beta_{i-1})\boldsymbol{c}_{i-1}\\
&\qquad + (\alpha_{i-\frac{1}{2}} + \alpha_{i+\frac{1}{2}} + \beta_{i+1} + 4\beta_i + \beta_{i-1})\boldsymbol{c}_i\\
&\qquad\quad + (-\alpha_{i+\frac{1}{2}} - 2\beta_{i+1} - 2\beta_i)\boldsymbol{c}_{i+1} + \beta_{i+1}\boldsymbol{c}_{i+2} + (\nabla E)_i = 0. \qquad (2.38)
\end{aligned}$$

In order to get matrix equations the coefficients in 2.38 are collected in a matrix A with entries

$$\begin{aligned}
A_{i,i-2} &= \beta_{i-1}, &(2.39)\\
A_{i,i-1} &= -\alpha_{i-\frac{1}{2}} - 2\beta_i - 2\beta_{i-1}, &(2.40)\\
A_{i,i} &= \alpha_{i-\frac{1}{2}} + \alpha_{i+\frac{1}{2}} + \beta_{i+1} + 4\beta_i + \beta_{i-1}, &(2.41)\\
A_{i,i+1} &= -\alpha_{i+\frac{1}{2}} - 2\beta_{i+1} - 2\beta_i, &(2.42)\\
A_{i,i+2} &= \beta_{i+1}, &(2.43)
\end{aligned}$$

that has a pentadiagonal form plus additional non-zero entries in the upper right and lower left corners – remembering the $j \bmod N$ index mapping due to the closed curve – and vectors

$$\boldsymbol{x} = (c_{1i})_{i=0-(N-1)}, \qquad \boldsymbol{f}_x = (\frac{\partial}{\partial x_1}E_i)_{i=0-(N-1)}, \qquad (2.44)$$

$$\boldsymbol{y} = (c_{2i})_{i=0-(N-1)}, \qquad \boldsymbol{f}_y = (\frac{\partial}{\partial x_2}E_i)_{i=0-(N-1)} \qquad (2.45)$$

are constructed for the x_1- and x_2-components of the contour and the derivative of the external energy respectively. This gives the conveniently short notation:

$$A\boldsymbol{x} + \boldsymbol{f}_x = 0 \qquad A\boldsymbol{y} + \boldsymbol{f}_y = 0. \qquad (2.46)$$

The external energy will depend on the position of the contour nodes on the underlying image and its intensity distribution, but it is reasonable to assume that A, $\boldsymbol{f_x}$ and $\boldsymbol{f_y}$ will be approximately constant for small changes in the contour. Hence one can use a mixed explicit / implicit Euler method to solve the equations iteratively. The evolution equations take the following form:

$$\gamma(\boldsymbol{x}_{t+1} - \boldsymbol{x}_t) = -A\boldsymbol{x}_{t+1} - \boldsymbol{f}_{x,t} \qquad \gamma(\boldsymbol{y}_{t+1} - \boldsymbol{y}_t) = -A\boldsymbol{y}_{t+1} - \boldsymbol{f}_{y,t} \tag{2.47}$$

$$\Downarrow \qquad\qquad\qquad\qquad \Downarrow$$

$$\boldsymbol{x}_{t+1} = (A + \gamma I)^{-1}(\gamma\boldsymbol{x}_t + \boldsymbol{f}_{x,t}) \qquad \boldsymbol{y}_{t+1} = (A + \gamma I)^{-1}(\gamma\boldsymbol{y}_t + \boldsymbol{f}_{y,t}). \tag{2.48}$$

These equations can be solved in order N complexity due to the particular form of matrix $A + \gamma I$. A modified Gauss algorithm that considers only elements A_{ij} with either $|j - i| \leq 2$, $i \geq N - 3$ or $j \geq N - 3$ was implemented for this.

2.3.7 Arcfinder Image Segmentation

Three caveats quickly became apparent in our efforts to automatically distinguishing arcs from their surrounding by image segmentation:

1. the signal-to-noise ratio of arcs is low on average. This is particularly felt by first and second intensity derivatives. Additionally, the high noise contribution lets a large fraction of pixels inside the area of a faint arc fall below the background intensity. This precludes the application of straightforward intensity thresholding and the default solution, isotropic smoothing, has other disadvantages: it enhances blending with neighbouring objects, reduces length-to-width ratios and smooths away faint and narrow features, artificially separating objects with bright substructure.
2. the generally smooth intensity distributions of arcs do not lend themselves to segmentation because they lack clearly defined edges and dominant arc substructure can hide faint features belonging to an arc by raising a segmentation threshold defined by average or median object intensity to their level.
3. the background underlying an arc is typically varying both along its length and perpendicular to it, partly due to the close proximity of the lens and blending with separate astronomical objects. In particular the total variation along the length of giant arcs is often larger than the average arc intensity.

For true arc detections, their smooth features devoid of small curvature radii are in our favor. Nevertheless, a robust segmentation algorithm is necessary in order to filter out spurious detections with more irregular geometries, and implemented in the arcfinder algorithm using active contour segmentation.

Adressing the first caveat, the curvature stress of the active contour method indirectly combines the signal over an extended contour length and partly compensates for a low signal-to-noise ratio locally. We combine a maximum gradient approach with intensity thresholding

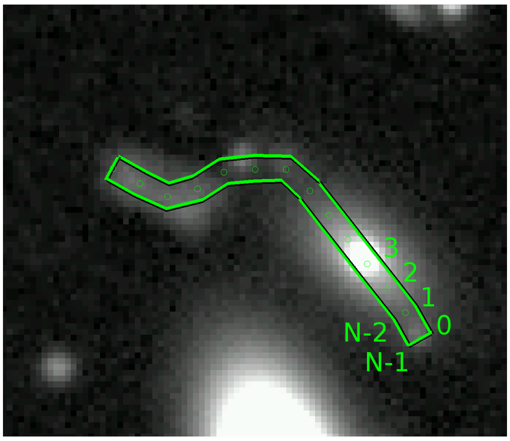

Figure 2.11: Initial contour around a (in this instance deliberate) spurious detection in a simulated CFHT image, with node indices for the first and final contour nodes. Circles mark points on the ridge line which are initially in the centre of a node pair.

by an adaptive isophote level to account for the second point. Regarding the third caveat, we update an estimate for the arc background in each iteration of the contour evolution, using averaged intensities and Catmull Rom splines (Catmull & Rom 1974) to avoid discontinuities.

Contour Initialisation

As above, a contour C is defined by its contour nodes $\boldsymbol{c}_i$, $i \in 0, \ldots, N-1$, tracing its perimeter counterclockwise. To create an initial contour, we place opposite nodes $\boldsymbol{c}_i$ and $\boldsymbol{c}_{N-1-i}$ on each side of a ridge line path (see 2.3.4) so that their centres $(\boldsymbol{c}_i + \boldsymbol{c}_{N-1-i})/2$ are equidistantly distributed on the estimated ridge line with a pairwise distance of approximately $\frac{1}{2}d_0$ – adjusted for the total length – between consecutive centre points. The initial distance of individual nodes $\boldsymbol{c}$ from the path is set to $r_i = \frac{1}{2}d_0$, resulting in a contour width of d_0 (see Figure 2.11).

Since the curvature for the natural border of an arc will be large at both ends of its ridge, which the active contour must emulate, we set the coefficients regulating the curvature at these points, that is $\beta_0, \beta_{N/2-1}, \beta_{N/2}$ and β_{N-1}, to zero. During contour evolution, this results in two smooth edges at the sides of a feature meeting at an arbitrarily sharp angle.

Image Energy Term

The image energy $E(\boldsymbol{c})$ (see Eq. 2.32) of the contour relies on intensities $I_C(i, r)$ averaged parallel to the path over one contour segment i for perpendicular distances r from the ridge line, in effect introducing anisotropic smoothing, but more importantly noise fluctuations in the node positions perpendicular to the path cause a counterforce from the curvature stress $\frac{\beta}{2}\boldsymbol{c}''^2$ and therefore tend to cancel each other out in each iteration of the contour evolution (see Eq. 2.48). The averaging in I_C has the drawback of constraining node movement in parallel to the path. However, as the intrinsic intensity variation along an arc is often low compared to the intensity variation induced by close or blended objects, it is sufficient – even helpful – to consider only the force component F_i perpendicular to the path direction in the image energy, even if this fixes the object length during contour evolution. All distance stress coefficients α are set to zero: along both sides to compensate for the absence of a parallel force extending the contour line and at the ends connecting the sides to avoid artificially reducing the arc width.

We treat the image energy force as a composite of two distinct properties: the second derivative of the intensity perpendicular to the contour and its distance from an isophote. To reduce the impact of substructure and noise on the second derivative of I_C, a simple adaptive smoothing scheme is used where the convolution kernel takes the form

$$K_{2\text{nd}} = -\frac{(r-r_i)^2 - w^2}{\sigma^4} \mathrm{e}^{-\frac{(r-r_i)^2}{2w^2}} \tag{2.49}$$

with $w = \max(1, \frac{1}{3}r_i)$. Assuming a feature has a constant Gaussian function intensity profile $I_C \propto \exp(-r^2/(2\sigma^2))$ of width σ along its length, the final distance $r_{\frac{1}{2}}$ from the ridge line where each nodes second derivative force becomes zero can be determined analytically:

$$r_{\frac{1}{2}} = \begin{cases} 3\sigma/\sqrt{3^2-1} \approx 1.06\sigma, & \text{for } r_i > 3 \\ \sqrt{\sigma^2+1}, & \text{else.} \end{cases} \tag{2.50}$$

Although this appears to be the straightforward solution, the isophote force is not determined by the gradient alone in order to avoid the influence of varying intensity gradients from substructure and noise. Instead, the closest distance $r_{\text{iso},i}$ to r_i where both $I_C(i, r)$ crosses the isophote level $I_{\text{iso}}(i)$ and the gradient $\partial/\partial_r I_C(i, r) < 0$ is determined and the force set to $r_{\text{iso},i} - r_i$. Both perpendicular force components $F_{\text{grad}}(i)$ and $F_{\text{iso}}(i)$ are normalised with the respective standard deviations $\sigma^2_{F\text{grad/iso}} = \sum_i F^2_{\text{grad/iso}}(i)$ over all nodes such that they share approximately equal influence on node shifts in each iteration.

Background Estimation and Termination

It is crucial to assign different isophote levels $I_{\text{iso}}(i)$ to each node if the background intensity is varying strongly along the length of an arc (see Figure 2.12). This background estimate is obtained at a narrow range of pixels at least $\frac{1}{2}d_0 + 1$ pixels outside the contour as supremum $F_I^{-1}(0.25)$ of the 25% minimal intensities parallel to the ridge line for each pair of nodes[4]. These values are then distributed on a set of fiducial points spaced approximately $10d_0$ pixels apart that provide the basis for a Catmull-Rom spline interpolation resulting in smooth background intensities $I_{C\text{bg}}(i)$. The isophote levels for each node are computed as:

$$I_{\text{iso}}(i) = I_{C\text{bg}}(i) + \mu_{\frac{1}{2}}(I_C(j, r_j) - I_{C\text{bg}}(j)),\ j \in \{0, \ldots, N-1\}, \tag{2.51}$$

---

[4] We consciously avoid modelling background variations perpendicular to the detection as the variation scale perpendicular to arcs blended with their lens is often too short for reliable estimates. As before, F^{-1} is the inverse of the cumulative distribution function.

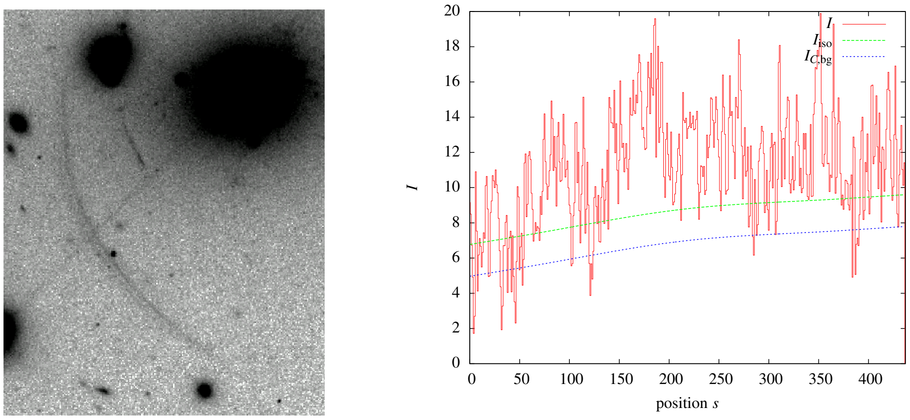

Figure 2.12: The red line in the diagram on the right shows averaged intensities perpendicular to the ridge line along the length (from bottom-right to top-left) of the giant arc in Abell 2390 displayed on the left. As expected, it is brightest in the centre, but it is also significantly brighter on its upper-left end. The green and blue lines show the automatically determined isophote and background levels respectively.

where $\mu_{\frac{1}{2}}(X) \equiv \tilde{X} \equiv F_X^{-1}(\frac{1}{2})$ denotes the median of a random variable X here and in the following.

The contour evolution terminates when the total deviation in node shifts goes below 0.1 pixels (see Figure 2.13). Since both force components are normalised by their standard deviation, this would not happen under normal circumstances, therefore two additional multipliers m_{L} and m_{R} are used that reduce the step size for each side L and R separately if their average positions start to oscillate around one value. More precisely,

$$\text{if} \sum_{i=0}^{N/2-1} |r_i(n) - r_i(n-2)| - |r_i(n) - r_i(n-1)| \leq 0 \text{ then } m_{\mathrm{L}} \rightarrow \frac{1}{2} m_{\mathrm{L}} \tag{2.52}$$

$$\text{if} \sum_{i=N/2}^{N-1} |r_i(n) - r_i(n-2)| - |r_i(n) - r_i(n-1)| \leq 0 \text{ then } m_{\mathrm{R}} \rightarrow \frac{1}{2} m_{\mathrm{R}}\,, \tag{2.53}$$

where values $n, n-1$ and $n-2$ denote the iterative step.

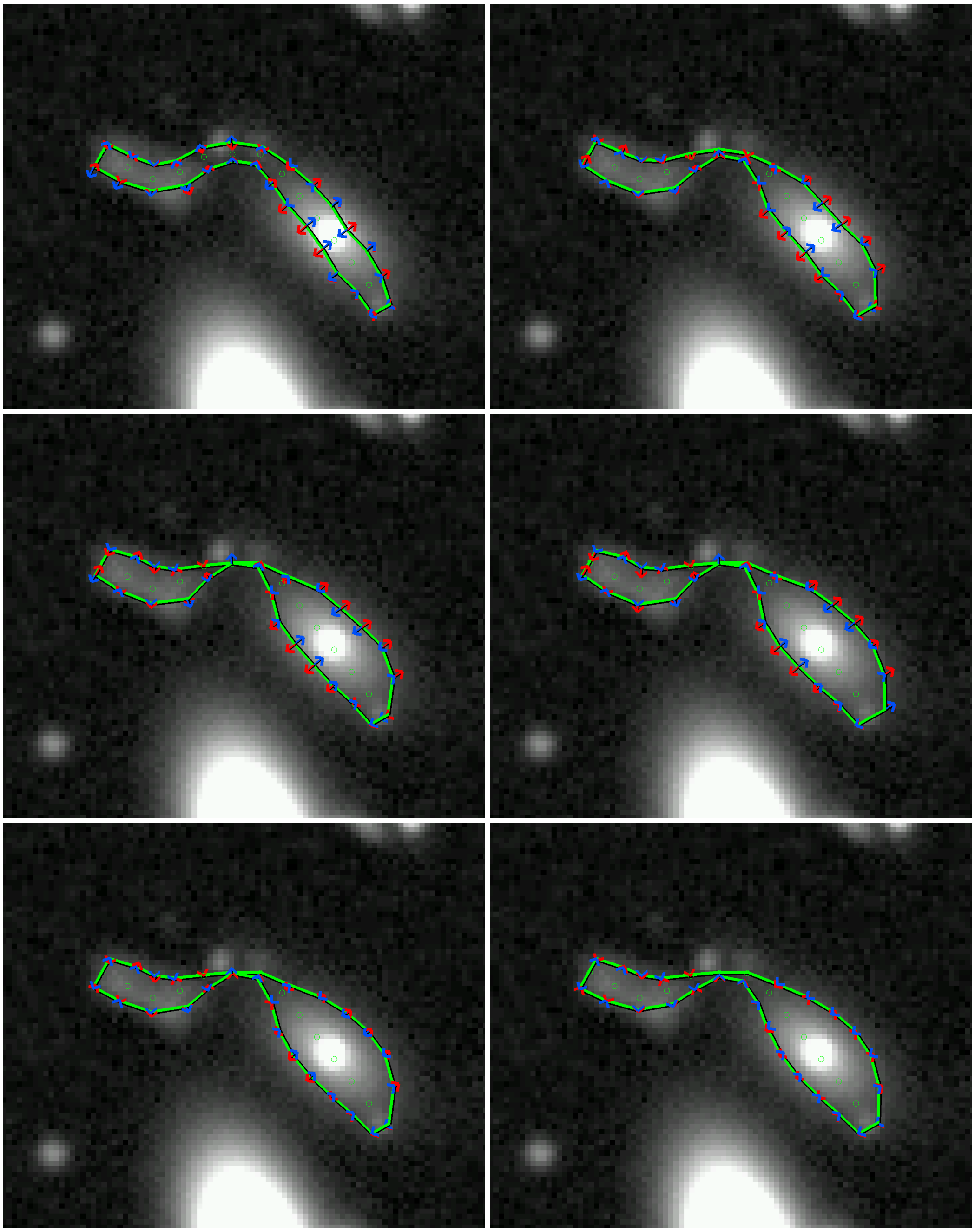

Figure 2.13: The initial contour in Figure 2.11 after 1, 4, 8, 12, 16 and 20 more iterations, where the final contour is accepted in this case. Red arrows indicate the isophote and blue arrows the second derivative force that maximises the intensity gradient. The reduction in arrow lengths in the penultimate iteration corresponds to a diminished m_L and m_R due to oscillations in the contour. The curvature stress prevents the contour from following the isophote completely.

2.3.8 Classification of Candidates

The final contours combined with the image background and noise determined earlier (see 2.3.1) provide the basis for the feature characteristics compiled into the catalogue of possible arc candidates. The (mean contour) position, length, total flux and signal-to-noise are computed straightforwardly.

Following the approach from Meneghetti et al. (2008) we determine the width w as the median width along the contour length and correct for the widening effect of the PSF by setting

$$w_{\mathrm{corr}} = f_1 \sqrt{\frac{w^2}{f_1^2} + \frac{\mathrm{FWHM}^2}{f_2^2}}\,, \tag{2.54}$$

where $f_1 = \frac{3}{\sqrt{3^2-1}}$ (see Eq. 2.50), FWHM is the full width at half maximum of the PSF in pixels and $f_2 = 2\sqrt{2\ln 2}$ is a correction factor to determine the standard deviation of a Gaussian function from the FWHM. While this procedure is the ideal analytical result only for Gaussian intensity profiles, it appears to provide realistic estimates also for the profiles observed (although further tests with simulated images are necessary, see also chapter 5).

The $1/r$ curvature of an arc is determined by fitting a circle through the complete ridge line path.

2.3.9 Masking of Stellar Artefacts

Since stars other than the sun appear as point sources, their images correspond to the instrument's point spread function (PSF). Bright stars can additionally cause oversaturated pixels and blooming artefacts (see Figure 2.3.9).

The central part of the PSF can usually be approximated by an Airy disc, the ideal diffraction pattern of a circular telescope aperture:

$$I_{\mathrm{A}}(r) \propto \left(\frac{2J_1(r)}{r}\right)^2\,, \tag{2.55}$$

where J_1 is the Bessel function of the first kind of order one and $r = \frac{1}{2}Dk\sin\theta$ with wavenumber $k = 2\pi/\lambda$, aperture diameter D and observation angle θ. For most relevant astronomical instruments, this is superimposed by diffraction spikes, extended features caused by diffraction on the telescope's secondary mirror spiders. Given spider widths w and lengths l, the analytical solution for this diffraction pattern is (Harvey & Ftaclas 1995):

$$I_{\mathrm{S}}(x_1, x_2) \propto \mathrm{sinc}^2\left(\frac{wx_1}{\lambda f}\right)\mathrm{sinc}^2\left(\frac{lx_2}{\lambda f}\right)\,, \tag{2.56}$$

where $\mathrm{sinc}(x) = \sin(x)/x$, λ is a wavelength, f the telescope's focal length and x_1, x_2 are orthogonal coordinates in the image plane such that the spike extends in the x_1 direction.

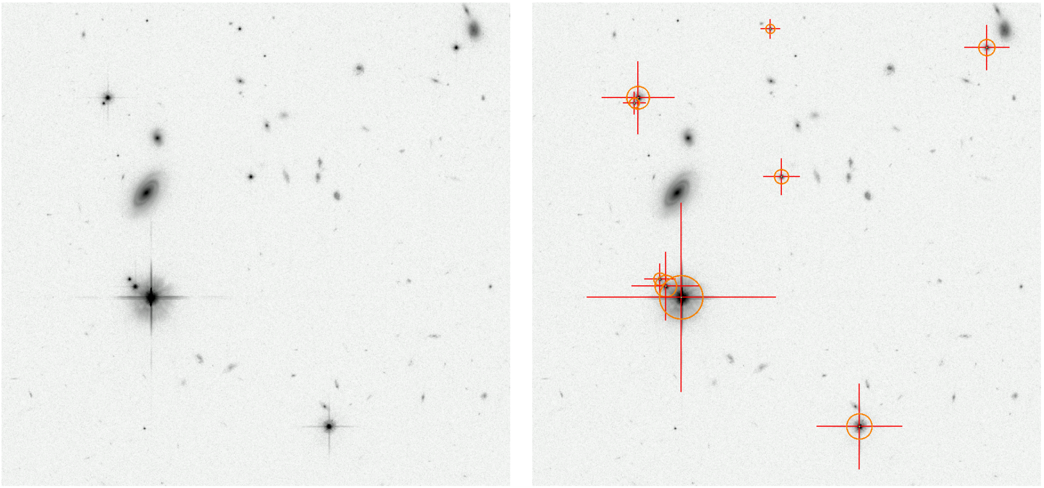

Figure 2.14: Small patch from the COSMOS survey taken with the HST ACS containing several bright stars, the brightest also causing a small amount of pixel blooming. On the right, the same stars are shown with their respective disk and spike masks.

Since both variations in the Airy disc as well as extended diffraction spikes can give rise to spurious detections, a simple masking routine invalidates detections overlapping either one of them with more than one half of their total area. To determine the respective masking areas for a star, its position and total flux must be known as well as the relative angles of the diffraction spikes and the telescope characteristics. The program uses catalogue data to obtain star positions and fluxes, but also includes basic star extraction and photometry algorithms, where the user has to provide an intensity threshold I_{star} and a minimal saturation intensity I_{sat}. Photometry for bright stars can be challenging given the high probability of oversaturated pixels hiding the bulk of the flux. I will therefore give a quick overview of the procedure as applied to oversaturated stars.

To extract stars, all connected regions with intensities greater than I_{star} are found using intensity thresholding. Region centres are determined inside iteratively shrinking apertures as weighted centres of brightness $\boldsymbol{c}_{\text{star}}$. For each region, a radial intensity profile is then constructed from the median intensities in sets $R(r)$ that contain all pixels at a distance of approximately r from the centre with intensity below I_{sat}:

$$I(r) = \mu_{\frac{1}{2}}(I(\boldsymbol{x})) - I_{\text{bg}},\ \boldsymbol{x} \in R(r) = \left\{ \boldsymbol{x} \middle| \, I(\boldsymbol{x}) < I_{\text{sat}} \wedge r - \frac{1}{2} \leq |\boldsymbol{x} - \boldsymbol{c}_{\text{star}}| < r + \frac{1}{2} \right\}, \tag{2.57}$$

where I_{bg} is the image background and $I(r)$ is not defined – and not taken into account in the algorithm – if $R(r)$ is empty.

To obtain a realistic flux estimate even if $I(r) > I_{\text{sat}}$ for low r, we use that the central lobe of the Airy disc containing most of the flux can be well approximated by a Gaussian function

$I(r) \approx I_0 \exp(-r^2/2\sigma^2)$. Its normalisation and width is determined by linear regression: fitting $C_0+r^2C_1$ to $\ln(I(r^2)) \approx \ln(I_0)-r^2/2\sigma^2$ inside an r-interval chosen so that $\frac{1}{5}(I_{\rm sat}-I_{\rm bg}) \leq I(r) \leq \frac{4}{5}(I_{\rm sat} - I_{\rm bg})$ we obtain the flux estimate $F_{\rm star} \approx 2\pi I_0\sigma^2$ with $I_0 = \exp(C_0)$ and $\sigma^2 = -1/2C_1$.

To find the extent of the Airy disc, we note the asymptotic behaviour of the Bessel function of the first kind J_1

$$J_1(r) \approx \sqrt{\frac{2}{\pi r}} \sin\left(r - \frac{\pi}{4}\right) \text{ for } r \gg \frac{3}{4} \tag{2.58}$$

and insert the non-oscillating term into Eq. 2.55 to find an upper value for the Airy disc

$$I_{\geq \rm A}(r) \approx \text{const.} \frac{F_{\rm star}}{r^3}\,, \tag{2.59}$$

where we used that the intensity is proportional to the flux. If the highest acceptable intensity that does not generate spurious detections is $I_{\rm max}$,

$$I_{\geq \rm A}(r_{\rm A}) \leq I_{\rm max} \Rightarrow r_{\rm A} \gtrsim C'_{\rm A} \left(\frac{F_{\rm star}}{I_{\rm max}}\right)^{1/3} \tag{2.60}$$

and similarly for diffraction spikes, setting $x_2 = 0$, replacing x_1 with $r_{\rm S}$ and observing only the maxima of the first sine in Eq. 2.56,

$$I_{\leq \rm S}(r_{\rm S}) \approx \text{const.} \frac{F_{\rm star}}{r_{\rm S}^2} \leq I_{\rm max} \Rightarrow r_{\rm S} \gtrsim C'_{\rm S} \sqrt{\frac{F_{\rm star}}{I_{\rm max}}}\,. \tag{2.61}$$

Two constants $C_{\rm A}$ and $C_{\rm S}$ corresponding to $C'_{\rm A}$ and $C'_{\rm S}$ scaled to pixel coordinates must be set by the user, as well as the angular orientation ϕ_i of diffraction spikes. Within the user interface of the arcfinder software, this can be done interactively by visual cross-checks of the resultant masks (Figure 2.3.9 is a screen capture of the main window, as are all astronomical images illustrated in this thesis except when noted otherwise).

2.3.10 Spiral Galaxy Detection

To detect galaxy-type lenses, an efficient method to filter spurious detections is to invalidate arc candidates by their distance and orientation to the closest galaxy. However, spurious detections generated by the arms of spiral galaxies are highly resistant to such a filter: in one spectral band, they can often be distinguished from arcs only by a faint connection to the galactic centre.

For Cosmological Evolution Survey (COSMOS) Advanced Camera for Surveys (ACS) images (see 4), a filter was implemented that further classifies galaxies read from a catalogue of COSMOS ACS observations and removes arc candidates it sees as spiral arms. The catalogue was compiled by (Leauthaud et al. 2007) using the SExtractor software and provides, among other data, angular coordinates and a measure of the area inside the segmented area

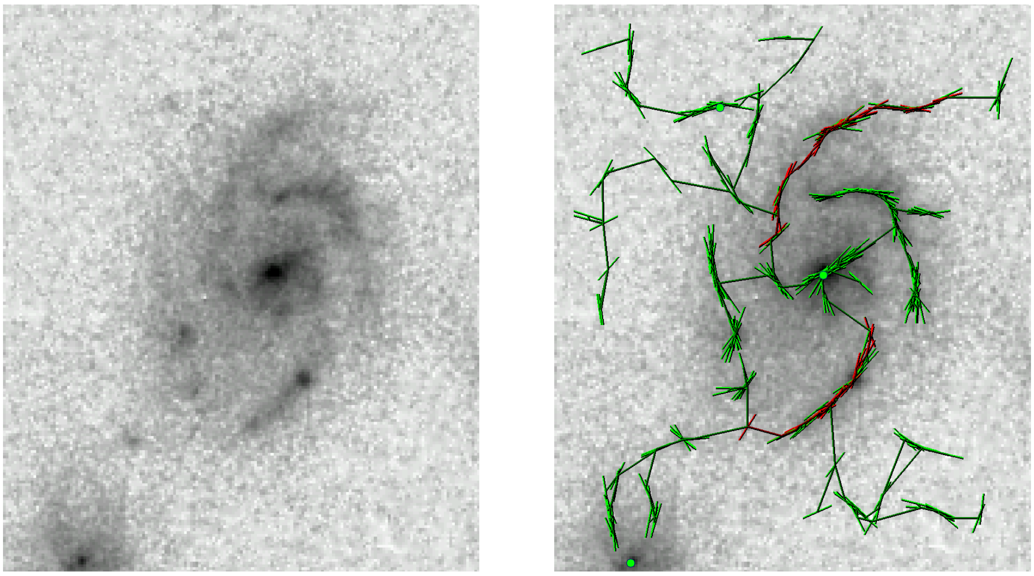

Figure 2.15: Another COSMOS ACS patch showing a foreground spiral galaxy on the left, and the same galaxy where detected structures are marked by coherent cells and routes through them starting at the galactic centre. Red cells mark paths that the algorithm consideres as spiral galaxy arms. The algorithm does not detect the tangential segment of the upper right arm as such and might consider it an arc if the other two arms were not present.

of each object in the form of ISOAREA and ELONGATION entries. Inside this designated isophotal area, the classification procedure proceeds to distribute cells of side length d_{glx} on a square grid as described in 2.2.1, with the galaxy in the centre of the grid. Mimicking the initial arc detection algorithm described above (2.2.2 to 2.2.4), coherent cells are determined and all cells below a threshold c_{glx} removed from further consideration. The remaining cells are treated as nodes in a graph, and shortest routes to all other nodes are determined from a node set to the galaxy's centre, using the spatial (Euclidean) cell distances for edge lengths (see Figure 2.15).

A breadth-first search then goes through the possible routes, stopping when a route qualifies as a spiral arm or is rejected, for example due to large gaps in the route. As the acceptance mechanism is complicated but purely heuristic, I will not go into great detail and instead present a short overview of the relevant route parameters and the acceptance condition. The parameters are

- the route length L
- initial and final radii r_1 and r_n from the galaxy's centre
- angles α_i edges in the route span with a radius through them to determine if the route changes the direction of its rotation

- angles ϕ_1 and ϕ_2 covered by the route segments below and above the intermediate radius $(r_1 + r_n)/2$
- the area A between the route and a mean radius through the route, integrated over absolute arc lengths

A route is considered a spiral arm if

- its radius has increased from the starting radius by $r_n/r_1 > 1\frac{2}{3}$ and
- the radial distance to length ratio $(r_n - r_1)/L > \frac{1}{2}$,
- the minimum of both covered angles $\min(\phi_1, \phi_2) > 5°$, where using the minimum of two separate covered angles was necessary in order to avoid filtering arcs with a spurious radial connection to the lenses centre,
- its length is greater than the minimum of the initial radius and the scale size, $L > \min(r_0, d_{\text{glx}})$, and
- the ratio of the area between route and mean radius and the length $A/L > 2.5$.

Applied to COSMOS ACS images, this scheme eliminates over 90% of the spurious detections generated by apparent spirals.

2.4 Summary of Detection and Filter Parameters

To conclude the description of the primary methods in the arcfinder algorithm, Table 2.1 contains the parameters set by the user that are relevant for the initial detection of elongated features and postprocessing of arc candidates. The most important parameter determining the sensitivity and the susceptibility to false positives in the initial detection algorithm is the scale size d_0, while many of the additional parameters generally do not require modification. The parameters controlling stellar masking, I_{star} and I_{sat} must be modified depending on the instrument characteristics and normalisation. Excluded from the list are obvious filter thresholds, for example lower limits for the arc candidate signal-to-noise and length-to-width ratios. An optional deblending method that was implemented recently and is not fully tested yet also requires an additional parameter.

parameter	possible value	short description
d_0	11	scale size (2.2 & 2.2.1)
d_1/d_0	0.32	characteristic grid distance over scale size (2.2 & 2.2.1)
d_{max}	0.73	maximal cell transport distance in units of d_0 (2.2.2)
c_{th}	0.54	cell coherence threshold to determine cell validity (2.2.4)
$\tilde{c}_{\text{th}}$	0.54	cell coherence threshold for object generation (2.2.5)
N_{cells}	5	minimal number of valid cells in an object (2.2.5)
L_{min}	20	minimal length of arc candidates in pixels (2.2.5)
R_{line}	-0.6	minimal cell 'ridgeness' (2.3.3)
S_{th}	5.0	minimal cell significance (2.3.3)
FWHM	0.0	fwhm of the point spread function in pixels (2.3.8)
I_{star}	10	star detection intensity (2.3.9)
I_{sat}	80	lower bound of the pixel saturation intensity (2.3.9)
ϕ_i	0,90,87.7	angles of diffraction spikes in degrees (2.3.9)
C_{A}	4.8	normalisation factor for Airy disc masking (2.3.9)
C_{S}	4.3	normalisation factor for Spike masking (2.3.9)
d_{glx}	11	scale size used in the spiral galaxy detection algorithm (2.3.10)
c_{glx}	0.5	coherence threshold used for spiral galaxy detection (2.3.10)

Table 2.1: List of user set parameters controlling the detection and filter algorithms.

3 Application to Simulated Images

To determine the completeness of the algorithm independent of a potentially biased arc classification by human observers, we applied the arcfinder software on simulated images with known arc positions. These were created for the CFHT, the projected Dark Universe Explorer (DUNE), HST ACS and Subaru instruments with raytracing software developed by Meneghetti et al. (2008). For all images, one numerically simulated cluster, one set of foreground galaxies taken from the GOODS-ACS archive and three different source distributions were assumed, and the images calculated in a 100x100 square arcsecond window. The seeing for the four instruments was assumed to be 0.6” for the CFHT and Subaru, 0.23” for the DUNE and 0.12” for the HST. Band filter characteristic are in the infrared for the most part: the simulated CFHT and Subaru operates with an i-band filter, the HST the F814W filter and only DUNE a broad riz-band filter characteristic.

For each instrument and set of source distributions, a foreground image and an image of the lensed sources was computed separately, taking into account the instrument and CCD characteristics, the filter and the seeing (see Figure 3.1). To get a sample of arcs at various magnitudes these images were combined, applying Poisson distributed photon noise consistent with exposure times of 1000s, 1500s and 3000s and rescaling the arc image with 28 different weight factors W_{source} ranging from 0.1 to 800 while the foreground image remained unscaled.

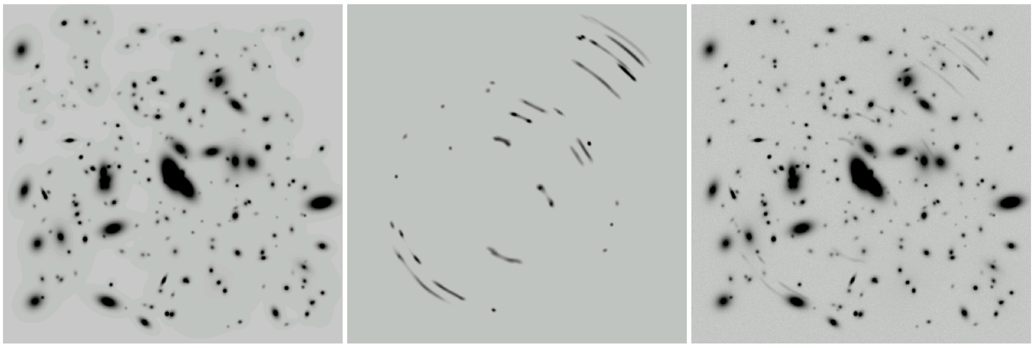

Figure 3.1: The basis for the simulated images provide foreground and arc images for the different instruments, here shown for Subaru with an I-band filter and seeing of 0.6”. The third image is the combined result with a scaling factor of 20 applied to the second image and noise added.

3.1 Preparing the Data

To automatise the completeness measurement, intensity thresholding was applied on the original arc images to create arc masks for the four separate instruments and the three sets of source distributions. Unfortunately, due to substructure and varying surface brightness for the individual arcs, a global threshold resulted in merged objects for some higher intensity arcs and separated substructures for fainter ones. Although a completely automated procedure would have been prefered, small manual adjustments to the masks were therefore necessary, but to remain objective, these were applied before applying any calibration steps (see Figure 3.2).

The total flux of each arc is then determined by integrating over the area belonging to it according to the mask in the original arc image. To obtain the flux in the combined image, this initial flux is multiplied by the respective weight values W_{source}. To convert into AB magnitudes, a magnitude zero point is used that is assigned to each of the reconstructed images and takes the combined instrument and detector efficiency for the respective filter into account. The geometry of arcs is determined from the mask as well, applying the principal methods in Meneghetti et al. (2008).

Note that using this method, the number of arcs for each instrument might be different, and turns out to be lower for the ground based CFHT and Subaru instruments, where arcs that can still be seen as separate entities with the HST and the DUNE are blended together. When comparing the results, it is helpful to note that using the integrated flux to determine AB magnitudes leads to lower surface brightnesses for extended sources compared to compact sources of the same magnitude. Large giant arcs can therefore easily vanish in the image noise even if their magnitude is in the range of a clearly visible star.

3.2 Regarding the Completeness and Spurious Detections

The statistical analysis of the data becomes straightforward after these preparatory steps. I adopt the definition of completeness used in Marshall et al. (2009), that is the ratio of detected over existing features in a set. The purity, here the ratio of true positives over the total number of detections, unfortunately cannot be determined in a meaningful way inside a small patch of 100x100 square arc seconds, but since spurious detections occur with some frequency for one of the more peculiar scale sizes used with ACS images, the number of spurious detections is counted for comparison with the other results. Since dithering of multiple exposures and drizzling were not simulated, a systematic bias towards fewer spurious detections for small scale sizes is expected as a result of missing pixel correlations.

For the test, the algorithm runs over the complete image sample, and records for each image the list of included arcs, storing with each arc its current magnitude, the observing instrument, the exposure time, the scale size d_0 and a flag indicating whether it was detected, which is set if at least one of the final arc candidates' contours overlaps with its mask. Logically, the completeness is determined later by counting the number of detected versus the

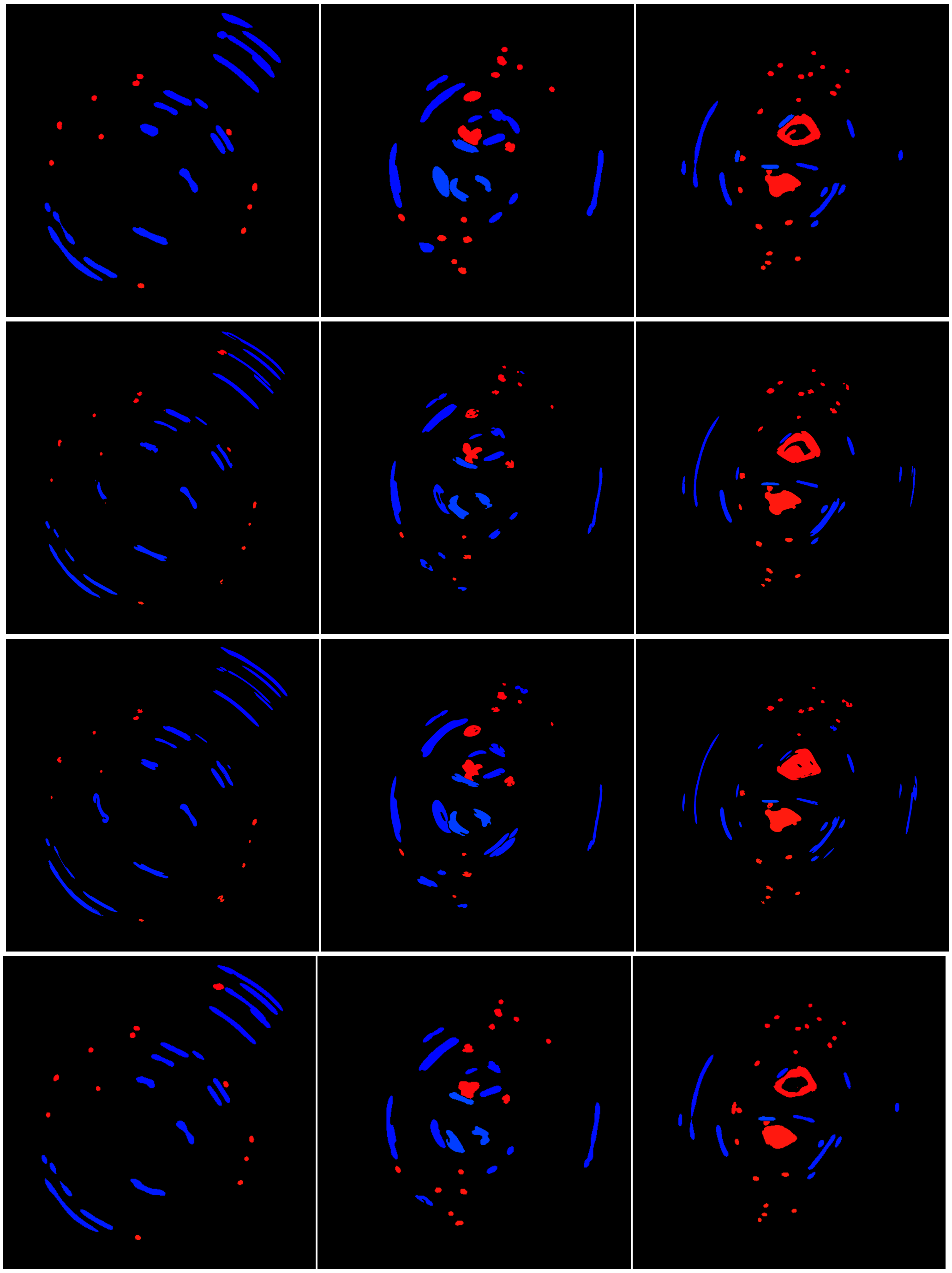

Figure 3.2: Mask images for (from first to last row) CFHT, DUNE, HST and Subaru data sets. Blue areas mark valid arcs, while red areas mark images not counted towards the total arc number.

total number of arcs for any particular instrument, exposure time, magnitude interval and scale size d_0. To obtain the number of spurious detections, the list of detections each with instrument, exposure time, scale size d_0 and a flag indicating whether it can be identified with an arc by contour overlap is similarly recorded. As spurious detections are generally caused by foreground objects, although perhaps only in some manifestations of the pixel noise, their magnitude is not considered.

We plot the completeness in one-magnitude steps for i-band CFHT, riz-band DUNE, I-band Subaru and F814W-band HST ACS images for exposure times of 1000s, 1500s and 3000s (see Figures 3.3 – 3.8). In each plot, the shaded areas correspond to the Poisson error $\sqrt{N_{\rm det}}/N_{\rm arc}$ in each sample. While blue tinted lines and areas illustrate the overall completeness, green tinted lines and areas consider only arcs longer than 12" and with length-to-width ratio greater than 7. We note that extended, thin arcs are more reliably detected, as they are less likely to be obscured by foreground objects. This balances their on average lower surface brightness compared to smaller arcs in the same magnitude range. As expected, an increase in exposure time from 1000 to 3000 seconds has a marginal influence in the higher magnitude range, but increases the number of detections at the magnitude limit for all instruments.

The ratio of spurious detections over the total number of detections for one instrument and one exposure time are indicated by a red arrow pointing to the left ordinate. For a scale size of 9 pixels, neither CFHT nor Subaru images generated spurious detections in this small test frame, but they appear prominently for the better resolved DUNE and HST images. These spurious detections are caused by edge on galaxies in the dense cluster environment, which appear to be more elongated in the higher resolution images. In particular for the large scale size $d_0 = 29$ in Figure 3.8, the cell reevaluation method tends to 'see' ridge line structures in foreground edge on galaxies, while smaller cells reject the comparatively large intensity slope up to the centre of a foreground galaxy (see 2.3.3). On the other hand, small scale sizes result in a lower signal to noise ratio in each cell's area A and might therefore not notice an underlying structure, resulting in a higher minimal flux for detection (see Figure 3.6).

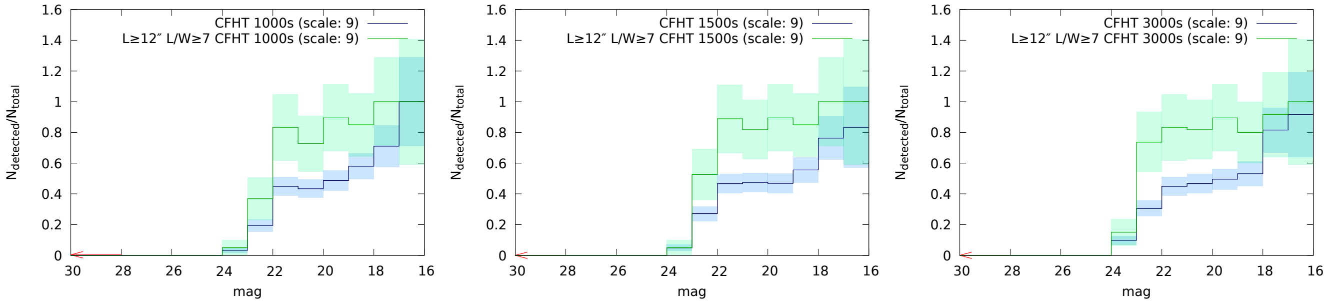

Figure 3.3: Detection ratio for i-band CFHT images, $d_0 = 9$ pixels.

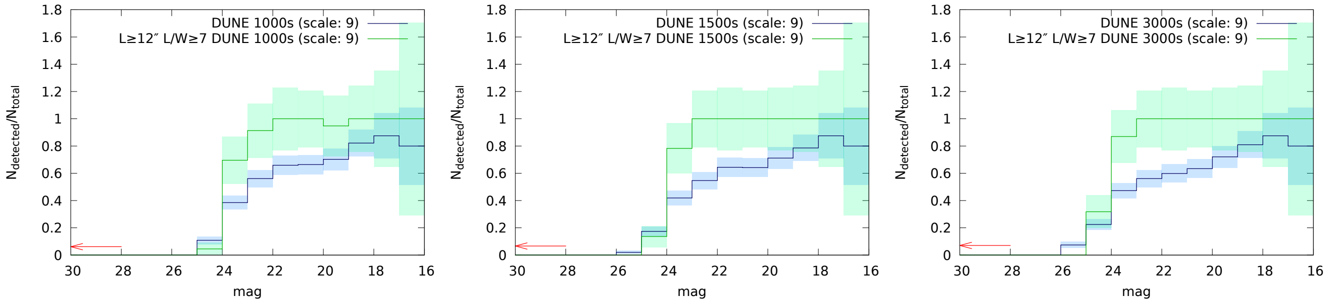

Figure 3.4: Detection ratio for simulated broad riz-band DUNE images, $d_0 = 9$ pixels.

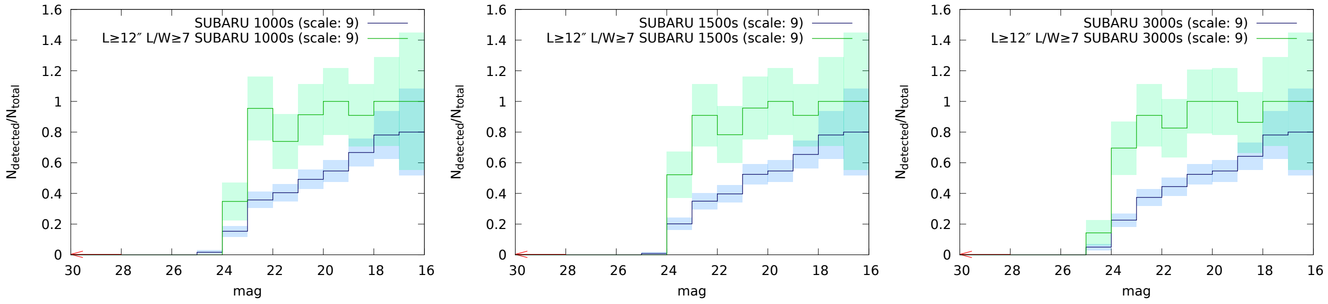

Figure 3.5: Detection ratio for simulated I-band SUBARU images, $d_0 = 9$ pixels.

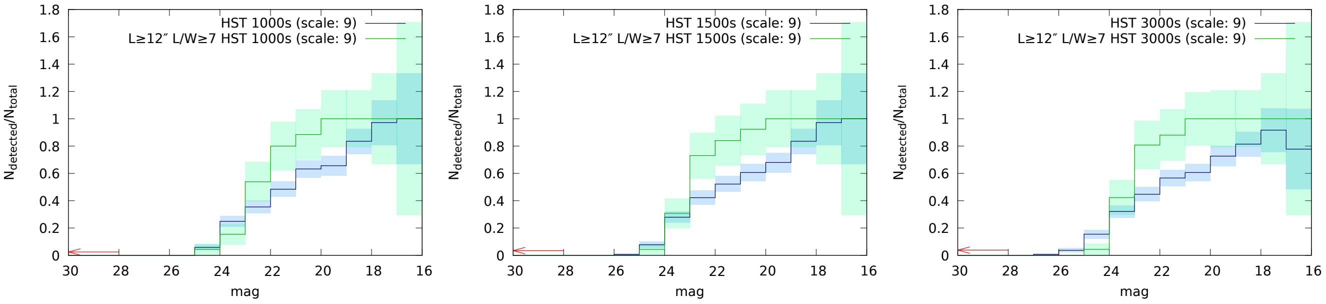

Figure 3.6: Detection ratio for simulated HST ACS images in the F814W-band, applying a scale size of $d_0 = 9$ pixels.

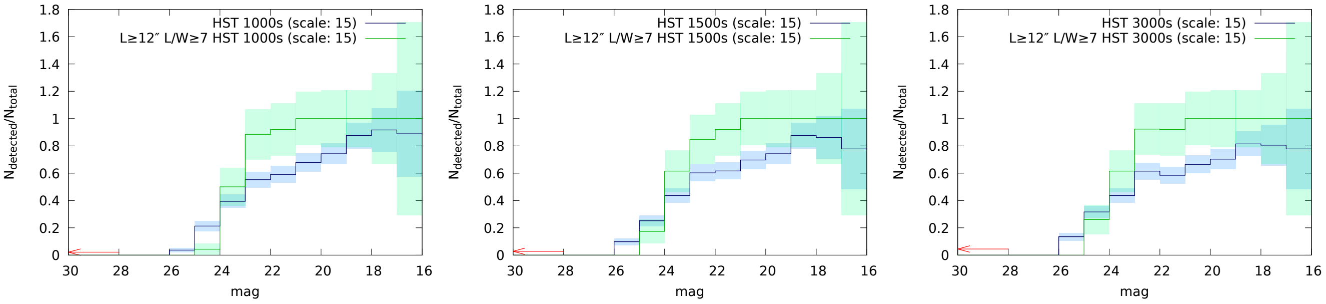

Figure 3.7: Detection ratio for simulated HST ACS images in the F814W-band, applying a scale size of $d_0 = 15$ pixels.

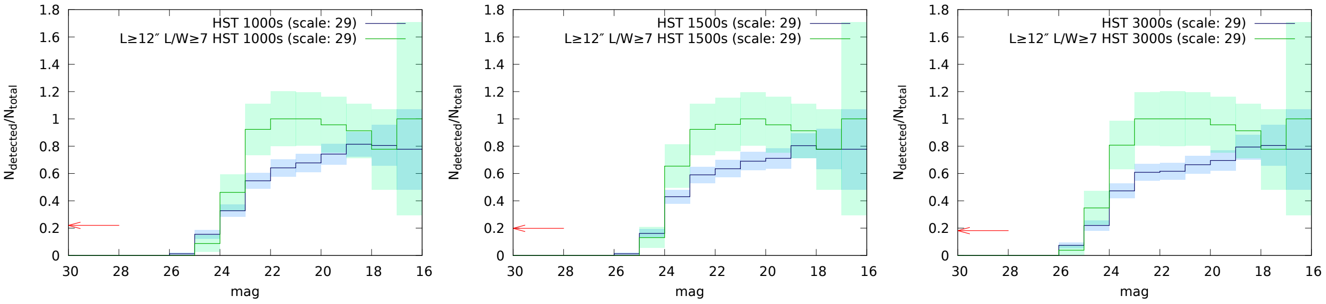

Figure 3.8: Detection ratio for simulated HST ACS images in the F814W-band, applying a scale size of $d_0 = 29$ pixels.

4 Application on COSMOS ACS Images

The algorithm was applied to images observed with the Hubble Space Telescope (HST) Advanced Camera for Surveys (ACS) for the Cosmological Evolution Survey (COSMOS). COSMOS aims to probe the formation and evolution of galaxies in relation to their large scale structure environment. It is an HST Treasury project that surveys a field of approximately two square degrees in the constellation of the Sextants up to IAB magnitudes of 27.2 in the F814W band. A dithering strategy with four exposures was used to filter cosmic rays and bad pixels and to bridge the gap between ACS CCD arrays, and the MultiDrizzle software was used for image reconstruction (see 2.1.2).

To adapt the algorithm to ACS images, its parameters were calibrated using galaxy-type lenses found earlier by Faure et al. (2008). We therefore expected to find some galaxy-type lenses similar to the ones found before, and possibly an arc from cluster lensing. Unfortunately, we did not detect the latter, which is completely consistent with the size of the field.

The final set of parameters determined during calibration can be found in Table 2.1; the FWHM value is set to zero there only because the width correction was not yet implemented. While the code ran in parallel, the estimated execution time on one processor is 48.5 hours, corresponding to a processing speed of approximately $1.85 \cdot 10^5$ pixels per second.

4.1 Galaxy Lens Candidates

Table 4.1 lists possible new arc candidates found among a large number of spurious detections (see 4.2), where detections coinciding with the arc candidates in Faure et al. (2008) – having been used for calibration – were naturally removed from the detection catalogue. Figures 4.1 – 4.24 show all candidates and the contours assigned to them during postprocessing.

For candidates 14, 16, 17, 20, 22 and 24 we note that an observer might find longer contours than the algorithm, which is primarily caused by cell invalidation controlled by the minimal 'ridgeness' and significance thresholds R_{line} and S_{th} (see 2.3.3). In the case of candidate 17, a shorter segment below the displayed contour was initially detected, but rejected during postprocessing due to its low signal-to-noise ratio in the relatively high intensity environment of the foreground galaxy. Multiple images might exist for candidates 1, 2, 7, 11, 20 and 24, but the algorithm is not designed to look for them specifically and in all of these cases they fall short of the set length and length-to-width thresholds. In Figure 4.12 a radial spurious detection is visible next to the arc candidate.

candidate index	RA (J2000)	DEC (J2000)	angular separation
01	09 58 02.46	02 24 25.90	1.44”
02	09 58 09.97	02 29 42.85	2.20”
03	09 58 29.89	01 37 37.42	2.90”
04	09 58 50.83	01 44 55.01	1.67”
05	09 58 55.33	02 08 50.79	0.89”
06	09 58 55.83	01 38 46.20	5.17”
07	09 59 30.96	02 34 28.39	0.83”
08	09 59 40.48	02 32 52.23	1.59”
09	09 59 54.13	02 30 38.72	1.30”
10	09 59 57.54	02 09 22.55	1.26”
11	10 00 00.79	02 43 07.70	1.60”
12	10 00 10.43	02 28 55.84	1.75”
13	10 00 11.61	02 35 29.82	2.68”
14	10 00 16.02	02 37 09.22	0.89”
15	10 00 16.59	02 25 05.22	2.70”
16	10 00 20.57	02 21 52.64	1.68”
17	10 00 23.83	02 52 23.42	0.59”
18	10 00 27.21	02 03 35.37	2.68”
19	10 01 03.62	02 16 08.98	2.07”
20	10 01 16.40	02 02 45.69	2.22”
21	10 01 32.42	02 48 13.49	0.90”
22	10 02 05.79	02 02 46.51	2.51”
23	10 02 35.20	01 52 42.46	1.24”
24	10 02 40.01	01 48 04.83	2.40”

Table 4.1: Possible new arc candidates in the COSMOS ACS survey.

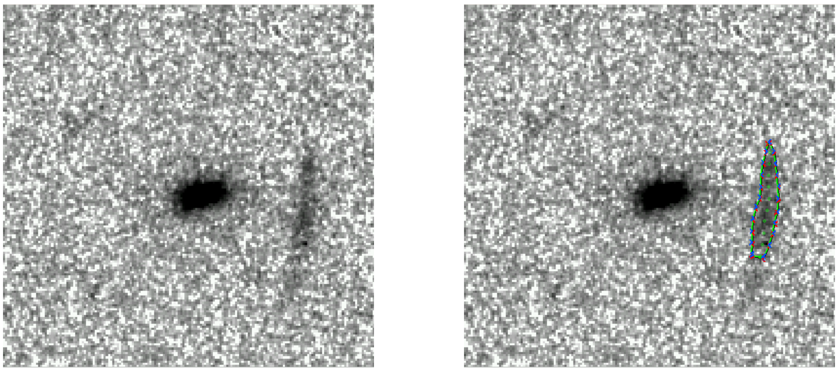

Figure 4.1: Candidate 1

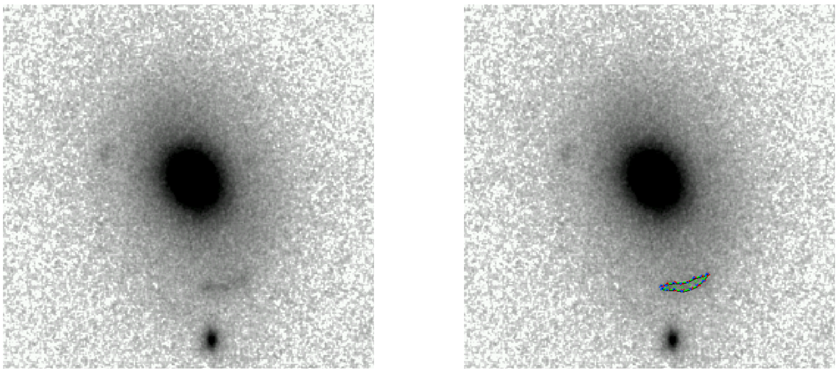

Figure 4.2: Candidate 2

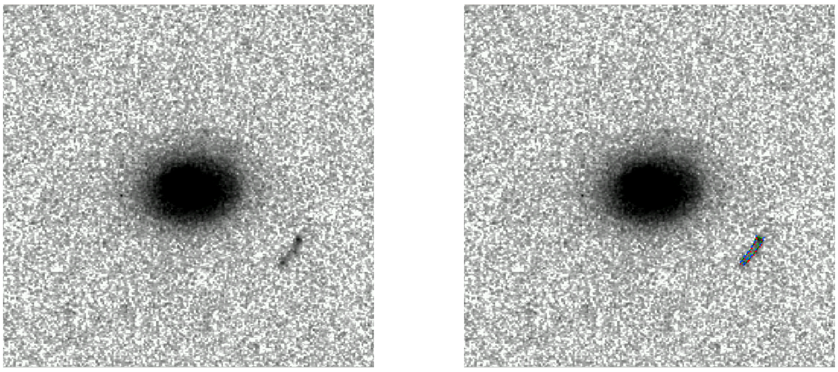

Figure 4.3: Candidate 3

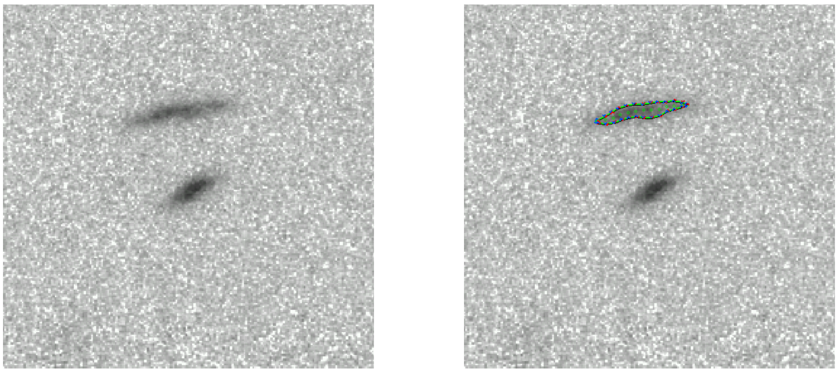

Figure 4.4: Candidate 4

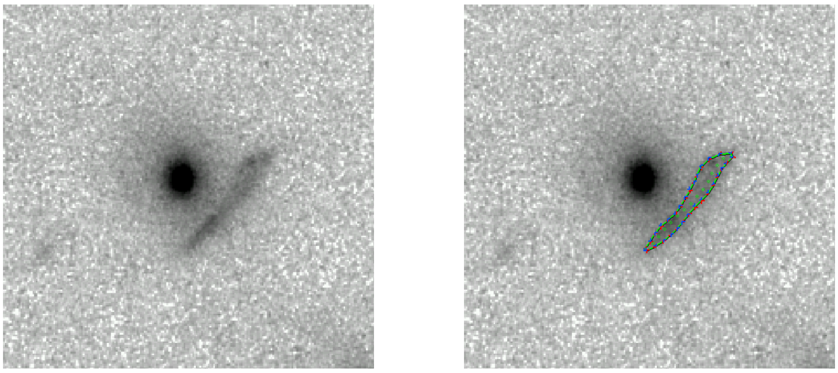

Figure 4.5: Candidate 5

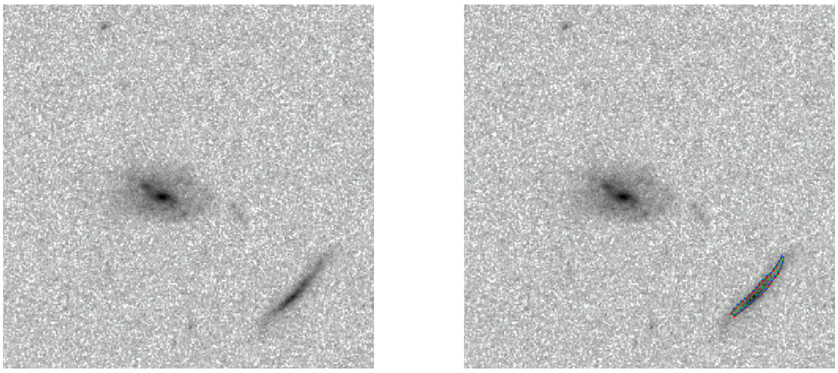

Figure 4.6: Candidate 6

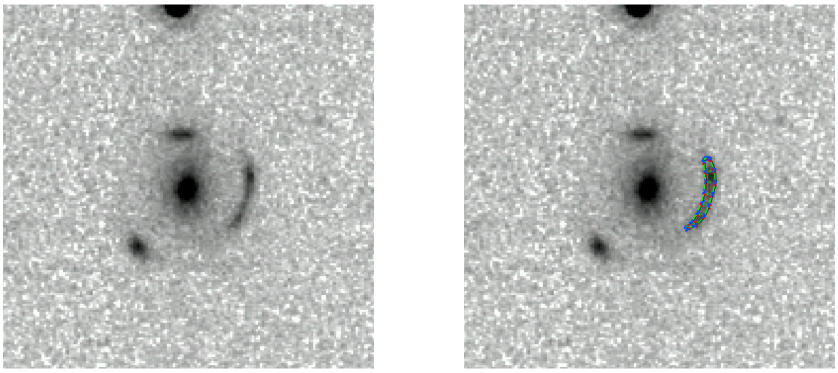

Figure 4.7: Candidate 7

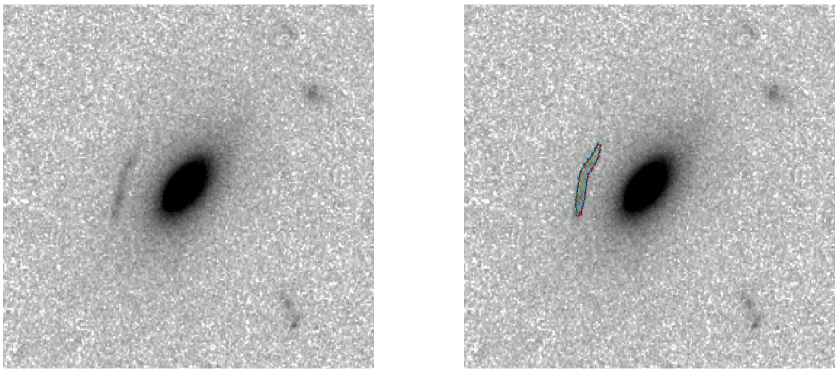

Figure 4.8: Candidate 8

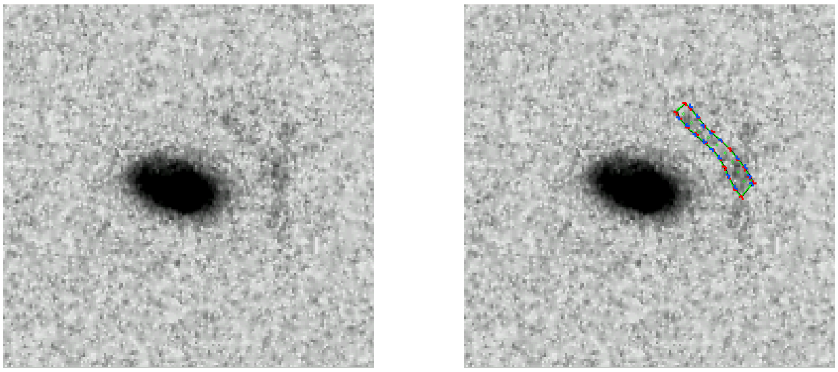

Figure 4.9: Candidate 9

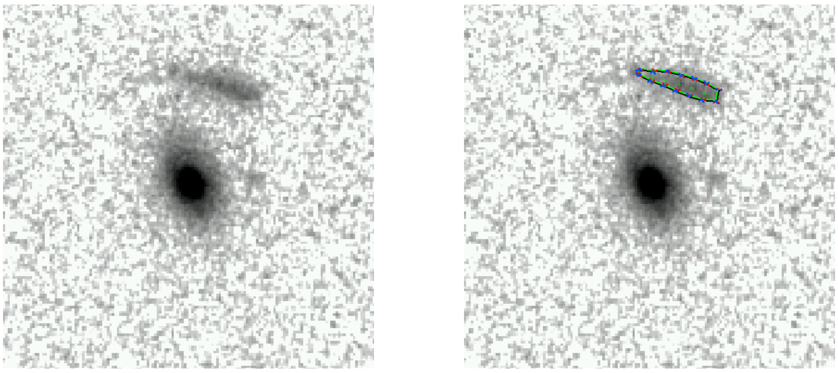

Figure 4.10: Candidate 10

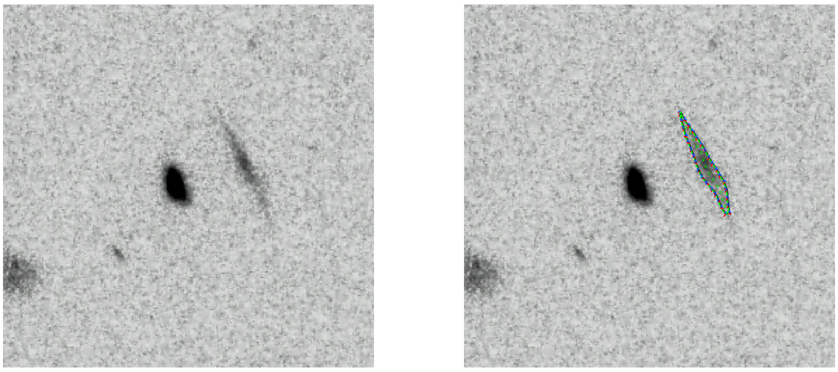

Figure 4.11: Candidate 11

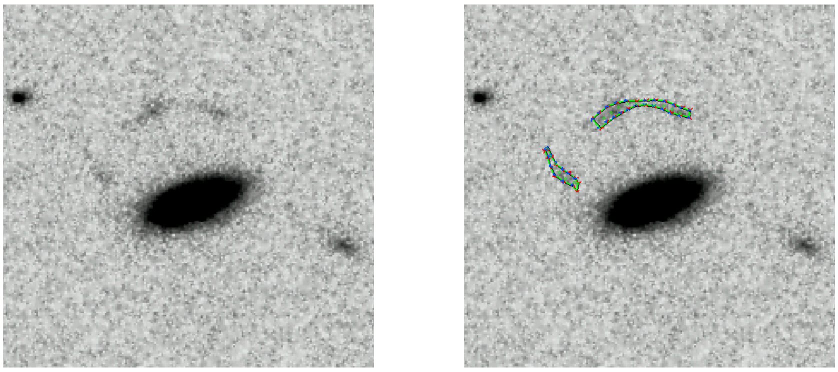

Figure 4.12: Candidate 12

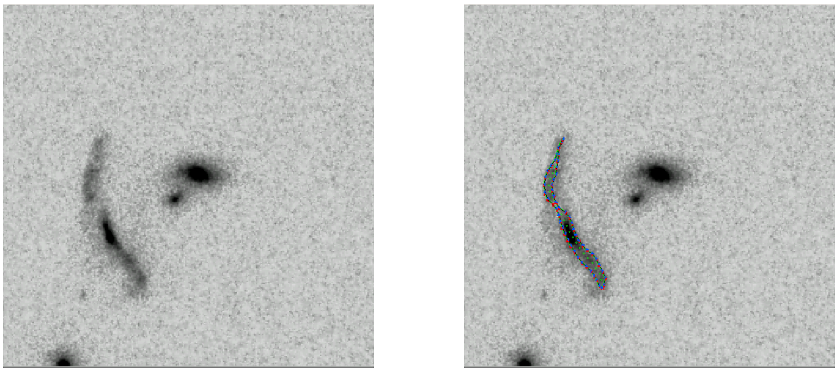

Figure 4.13: Candidate 13

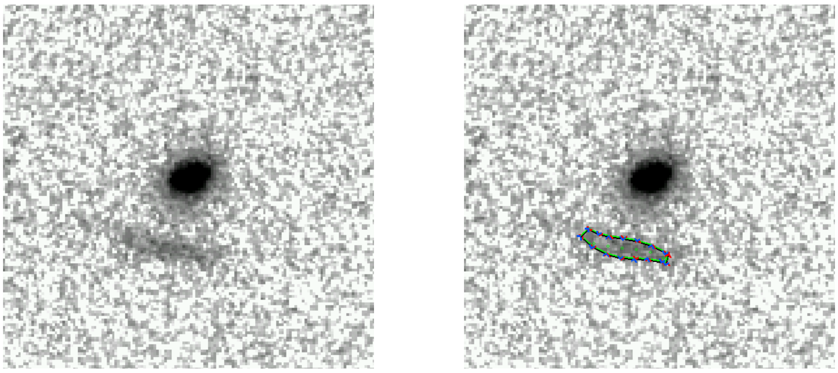

Figure 4.14: Candidate 14

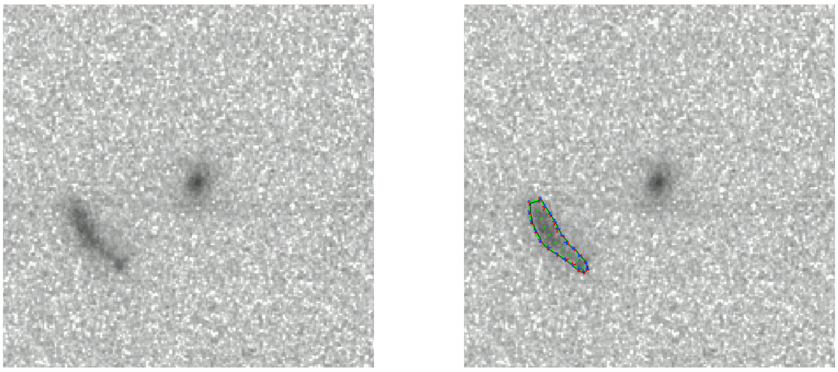

Figure 4.15: Candidate 15

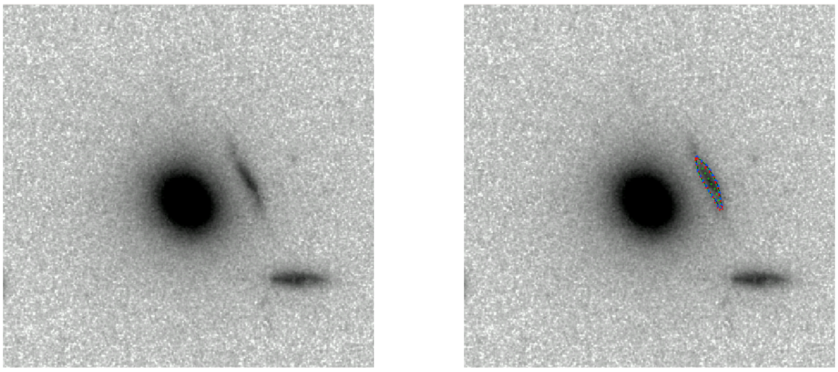

Figure 4.16: Candidate 16

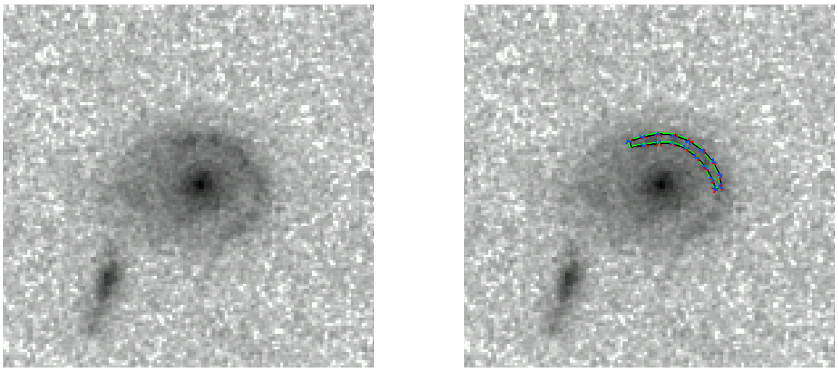

Figure 4.17: Candidate 17

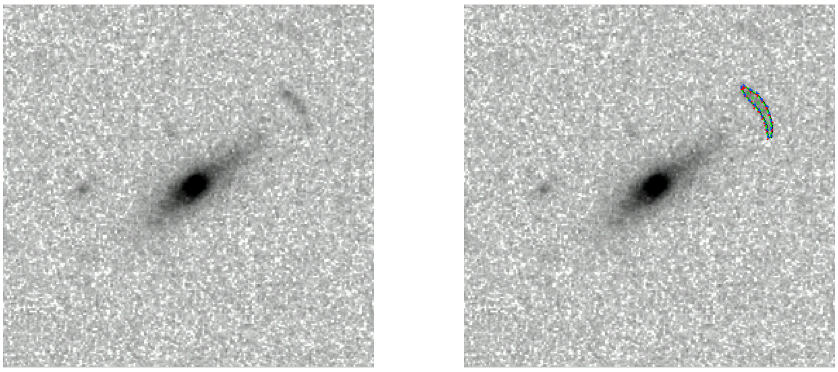

Figure 4.18: Candidate 18

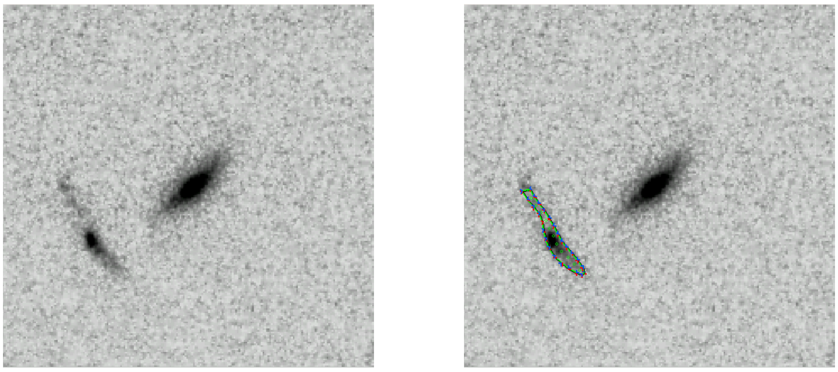

Figure 4.19: Candidate 19

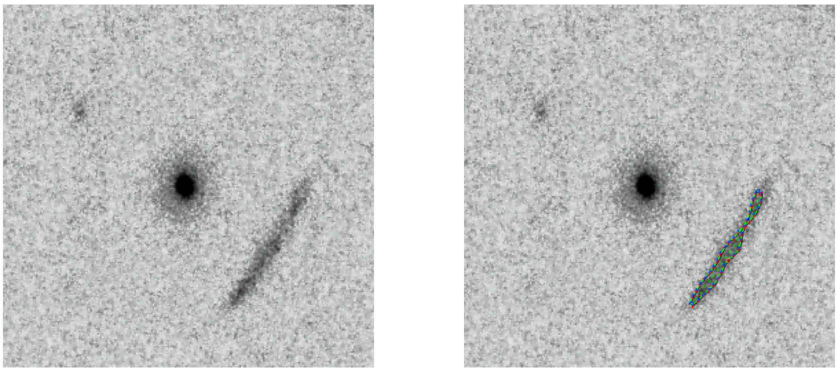

Figure 4.20: Candidate 20

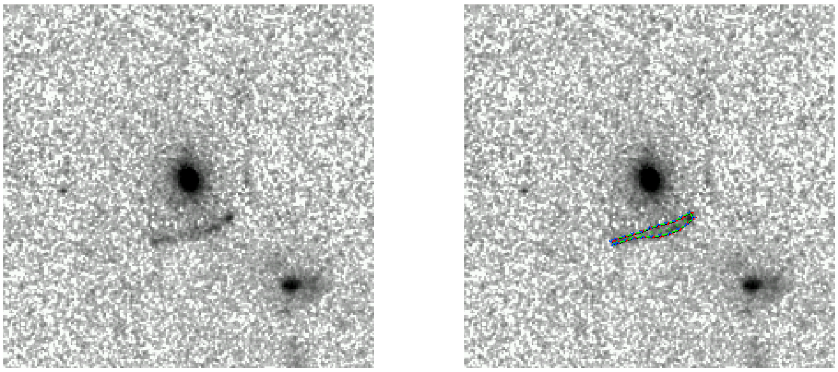

Figure 4.21: Candidate 21

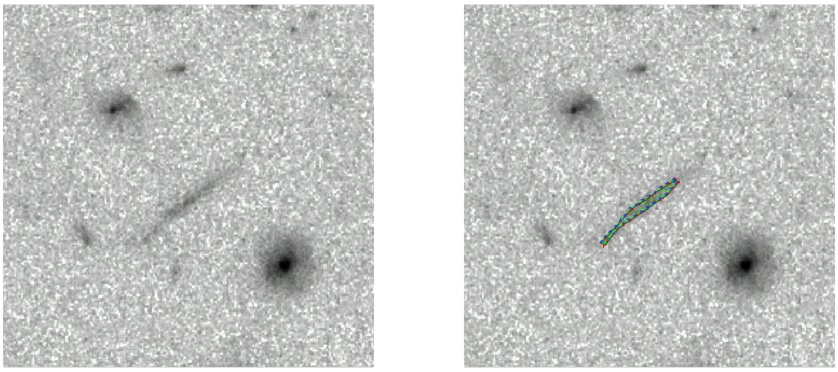

Figure 4.22: Candidate 22

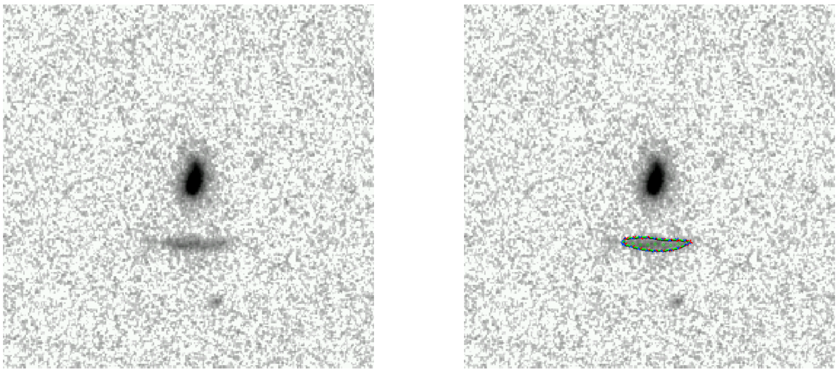

Figure 4.23: Candidate 23

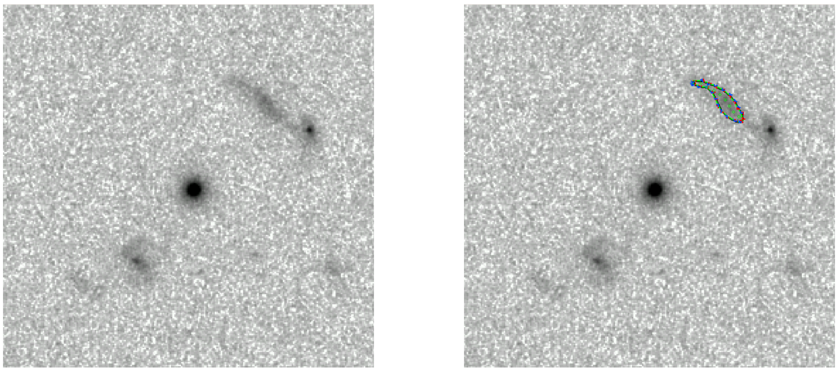

Figure 4.24: Candidate 24

4.2 Spurious Detections

Processing the 575 COSMOS ACS images containing approximately $3.2 \cdot 10^{10}$ pixels in a 1.64 square degree field produced a total number of 10409 arc candidates with the settings detailed in Table 2.1, an uncorrected length-to-width threshold of 3.5 and a minimal integrated signal-to-noise ratio of 30. This number stands in relation to approximately $3.9 \cdot 10^5$ galaxies in the COSMOS weak lensing catalogue and a total of about $1.2 \cdot 10^6$ objects in the complete COSMOS catalogue below a limiting F814W magnitude of about 26.5, but is still about 400 times larger than the number of new arc candidates shown above (Figure 4.25 shows the distribution of all detections in the COSMOS ACS data).

Figure 4.25: All arc candidates detected in the COSMOS ACS survey. Some exceptionally long diffraction spikes are noticeable.

The majority of spurious detections, and possibly some of the new arc candidates presented above, are edge on galaxies, which is consistent with the total number of galaxies in the field. Further false positives are generated by blended galaxy images, barred galaxies, and spiral galaxies that were not successfully masked (see 2.3.10). Another contribution comes from exceptionally bright stars: successfully masking them is challenging both because their diffraction spikes extend over long distances and because their total flux irrespective of CCD oversaturation is difficult to estimate (see 2.3.9). Finally, groups of point sources induce spurious detections in particular if they are connected by filaments of higher intensity pixels manifested in the noise. The likelyhood of such features is increased due to pixel correlations introduced by drizzling (see 2.1.2).

If the primary goal were to search for giant arcs, an upper length limit could drastically reduce the amount of spurious detections, as is illustrated by Figure 4.26, plotting the number of spurious detection per length interval over the length in pixels. Only 327 detections are above a length of 100 pixels, and 40 remain that are at least 200 pixels long.

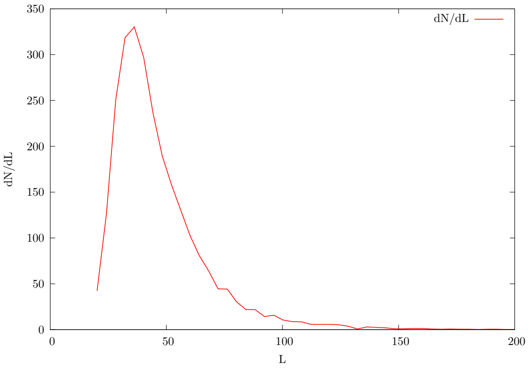

Figure 4.26: Distribution of arc candidate lengths in the COSMOS survey for scale size 11.

To find galaxy lenses, detections oriented tangentially or radially towards a catalogue item that is closer than four times the detection's length are flagged in the output catalogue. The arc candidates above are the result of visually inspecting the 2044 detections flagged as tangentially oriented.

Finally, an experimental routine was developed that finds clusters of arcs around a similar centre, but while this method successfully identified the Abell 2218 and Abell 2390 cluster lenses imaged by the HST WFPC2 instrument, it did not detect a lens in the COSMOS ACS images.

5 Summary, Conclusions and Outlook

Strong lensing magnifies background sources and directly probes the intermittent mass distribution between a source galaxy and the observer. Both effects can be used for a variety of astrophysical and cosmological applications. These include exploiting the magnifying effect of strong lensing to study the most distant observable galaxies and reconstructing the projected mass density of lensing galaxy clusters from the spatial distribution of source images in combination with weak lensing data, which provides insights into the cluster dynamics and evolution. Arc statistics has the potential to provide insights into non-linear large scale structure formation, but the predicted and the observed number of giant arcs from cluster lensing is currently inconsistent, which warrants further investigation. For any of these applications, it is both essential to increase the number of observed arcs without bias towards a certain cluster population and to provide an objective arc classification method (see 1.1). The goal of this thesis was therefore to develop an automated arc detection and classification algorithm that can be applied to large scale surveys without a priori information on possible lens positions.

5.1 Method and Results

We developed a novel image processing algorithm for detecting elongated features based on an adaptive sampling method of second moments in image 'cells' and an evaluation of local coherence in the image orientations derived from the second moments.

This arcfinder algorithm first distributes cells of several pixels side length, defined by a scale size parameter, uniformly over the complete image such that their respective areas overlap. It then attracts these cells to locally bright features using the local centre of brightness in each cell, and proceeds to determine the orientation of any underlying elongated structures using second moments and the complex ellipticity derived from them. This aligns cells with elongated features while cells on a noisy background receive random orientations. This is used to mark cells which are spatially and orientationally aligned with the cells in their neighbourhood. A friends of friends type algorithm is then used to combine coherent cells into preliminary detections. This method is computationally efficient and extremely sensitive even to faint features (see 2.2).

In order to remove false positive and to classify the initial detections, a number of postprocessing algorithms were developed, including filters that further analyse the data contained in cells, an algorithm that generates a graph marking the ridge line of each detection, an active contour segmentation method that measures the shape of preliminary detections and algorithms that mask stellar diffraction spikes and other point spread function artefacts as well as the arms of spiral galaxies (see 2.3).

We applied our software to simulated CFHT, DUNE, HST ACS and Subaru images and obtained the completeness as a function of the arc magnitude for arcs hidden inside a cluster foreground (see 3.2). It is apparent that in the same magnitude range, thin and extended arcs are detected preferentially, which is largely due to them being less affected by blending with galaxies in the cluster foreground.

As a proof of concept, the software was also applied to HST ACS images observed as part of the COSMOS survey (see 4). As simulated images were not available at this point, it was calibrated using galaxy-type lenses detected earlier by Faure et al. (2008). Among a large number of spurious detections primarily generated by edge on galaxies, we find 24 new arc candidates in the observed 1.64 square degree field.

The arcfinder software detects and further processes elongated features with approximately $3.8 \cdot 10^5$ pixels per second for a scale size of nine pixels up to $1.3 \cdot 10^6$ pixels per second for a scale size of 29 pixels in simulated images (see 3). Applied to ACS images, the software processes approximately $1.85 \cdot 10^5$ pixels per second for a scale size of 11 pixels. The software's high computational efficiency combined with the possibility to process several sets of images in parallel makes it well suited for application to current and upcoming surveys.

5.2 Outlook

With the arc detection software completed, the next logical step will be the application to observational data. Immediately available is the CFHTLS-Wide survey (Tereno et al. 2005) imaging a 170 square degree field in five bands down to i' magnitudes of 24.5. Data from the Pan-STARRS PS 1 instrument (Kaiser et al. 2002) will probably become available within several months. While the Subaru telescope's Hyper-Suprime Camera will not see first light until late in 2011, calibrating the algorithm on simulated data for this instrument will allow swift application in the future.

Where images in multiple bands are available, we expect that extending the algorithm to include colour information can further reduce the amount of spurious detections primarily by separating blended sources and by using photometric redshift information to discern foreground from background objects.

Further tests on simulated images are necessary to determine the consistency of flux and shape measurements, in particular the magnitude and the length-to-width ratio of arcs, which are important measures for comparison with theoretical estimates from arc statistics. To more accurately determine the selection function for varying arc characteristics, the algorithm has to be applied to a larger data sample, including multiple foreground clusters and projections. Large foreground images can be used to quantify the purity of the detected arc candidates.

Most importantly, detected cluster lenses can be used in two ways: first, systems with multiple images are promising candidates for cluster reconstruction in combination with weak lensing data and second, a sufficient number of detected cluster lenses can be used to improve the observational constraints for arc statistics.

6 Appendix

6.1 Relation of the Ellipse Orientation and the Complex Ellipticity

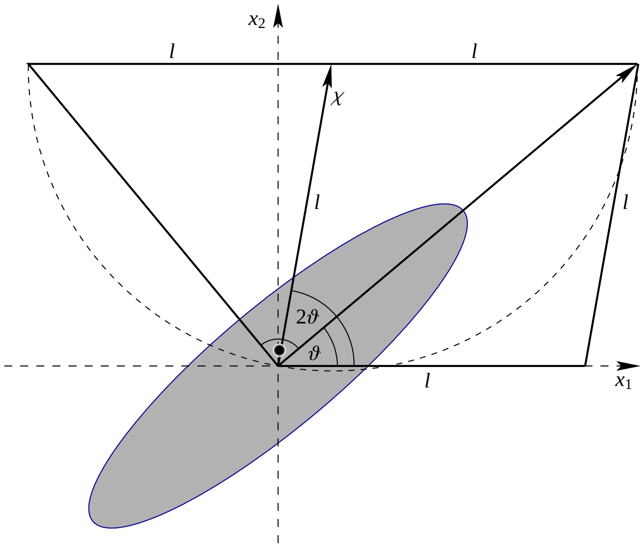

Figure 6.1: Geometric construction of the orientation ϑ.

The complex ellipticity $\chi = \frac{1-r^2}{1+r^2}\exp(2\mathrm{i}\vartheta)$ and the real axis span the angle 2ϑ in $\mathbb{C}$, twice the angle ϑ between the x_1-axis and the semi major axis of an underlying ellipse in image space. Let $(\chi_1, \chi_2) = (\Re(\chi), \Im(\chi))$. We can obtain a vector parallel to the ellipse's semi major axis by computing the diagonal $(\chi_1 + |\chi|, \chi_2)$ of the rhombus with side length $l = |\chi|$ seen on the right side of Figure 6.1.

Since the length of this diagonal approaches zero for $2\vartheta \to \pi$, this method is unsuitable for bisecting angles 2ϑ close to π. Using Thales' theorem we notice that the vector $(\chi_1 - |\chi|, \chi_2)$ and the semi major axis of the ellipse span an angle of $\frac{\pi}{2}$ (see slashed semicircle in Figure 6.1). Therefore, the perpendicular vector $(\chi_2, |\chi| - \chi_1)$ also bisects 2ϑ, although it approaches zero for $2\vartheta \to 0$. We can compute the orientation reliably using both com-

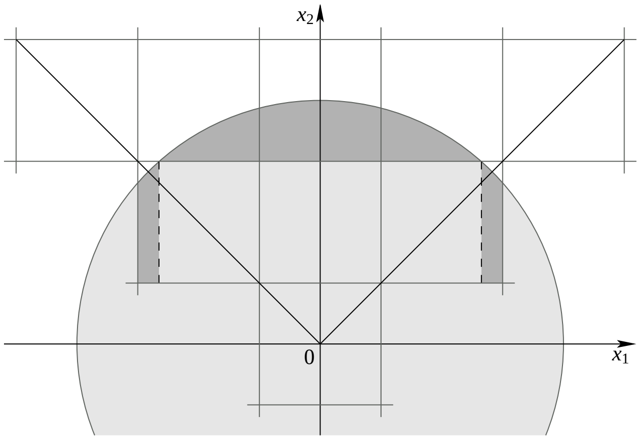

Figure 6.2: Nine pixels and their overlap with a circular disk of radius two at the coordinate origin. Dark shaded areas correspond to the integral Eqs. 6.1 – 6.3.

plementary methods where they are numerically stable:

$$\boldsymbol{d} = \begin{cases} (\chi_1 + |\chi|, \chi_2) & \text{for} \quad \chi_1 \geq 0 \\ (\chi_2, |\chi| - \chi_1) & \text{for} \quad \chi_1 < 0 \end{cases} .$$

6.2 Pixel Reweighting to Account for a Partial Disc Overlap

In the final paragraph of 2.2.3 I remarked on the necessity to weight pixels and use modified pixel centres $\boldsymbol{x^p}(\boldsymbol{x})$ according to their overlap $A_p(\boldsymbol{x})$ with a circular disc of radius $R = \frac{1}{2}d_0$ in order to avoid a bias in the cell orientation $\boldsymbol{e}$ in the diagonal directions. Without loss of generality, let the circle centre be at the coordinate origin as in Figure 6.2 and the vertical pixel coordinate $x_2 \geq 0$. I denote the integration limits with $a_1 \leq x_1 \leq b_1$ and $a_2 \leq x_2 \leq b_2$. Then the integral terms separate into a trivial term for the rectangular segment and

$$\begin{aligned} \iint_A \mathrm{d}^2 x &= \int_{a_1}^{b_1} \sqrt{R^2 - x_1^2} - a_2 \, \mathrm{d}x_1 \qquad (6.1) \\ &= \frac{1}{2} \left(x_1 \sqrt{R^2 - x_1^2} + R^2 \arctan\left(\frac{x_1}{\sqrt{R^2 - x_1^2}} \right) \right) - x_1 a_2 \Bigg|_{a_1}^{b_1} \end{aligned}$$

for computing the non-rectangular segments of the overlap A_p, as well as

$$\begin{aligned}
\iint_A x_1 \mathrm{d}^2 x &= \int_{a_1}^{b_1} x_1 \left(\sqrt{R^2 - x_1^2} - a_2 \right) \mathrm{d}x_1 \\
&= -\frac{1}{3}(R^2 - x_1^2)^{3/2} - \frac{x_1^2 a_2}{2} \bigg|_{a_1}^{b_1}
\end{aligned} \tag{6.2}$$

and

$$\begin{aligned}
\iint_A x_2 \mathrm{d}^2 x &= \int_{a_1}^{b_1} \int_{a_2}^{\sqrt{R^2 - x_1^2}} x_2 \, \mathrm{d}x_1 \mathrm{d}x_2 \\
&= \int_{a_1}^{b_1} \frac{1}{2}(R^2 - x_1^2 - a_2^2) \mathrm{d}x_1 = \frac{1}{2} x_1 (R^2 - a_2^2) - \frac{1}{6} x_1^3 \bigg|_{a_1}^{b_1}
\end{aligned} \tag{6.3}$$

for finding the modified pixel centre $\boldsymbol{x^p}$.

Bibliography

Alard, C. 2006, ArXiv Astrophysics e-prints astro-ph/0606757 1.2.2

Allam, S. S., Lin, H., Tucker, D., et al. 2007, in Bulletin of the American Astronomical Society, Vol. 38, 910–+ 1.1.2

Austin, D. 2006, `http://www.ams.org/featurecolumn/archive/voronoi.html` 2.3.4

Bartelmann, M., Huss, A., Colberg, J. M., Jenkins, A., & Pearce, F. R. 1998, A&A, 330, 1 1.1.3

Bartelmann, M. & Schneider, P. 2001, Phys. Rep., 340, 291 1.1.1

Bartelmann, M. & Steinmetz, M. 1996, MNRAS, 283, 431 1.1.3

Bartelmann, M., Steinmetz, M., & Weiss, A. 1995, A&A, 297, 1 1.1.3

Bertin, E. & Arnouts, S. 1996, A&A, 117, 393 1.2.2

Bolton, A. S., Burles, S., Schlegel, D. J., Eisenstein, D. J., & Brinkmann, J. 2004, AJ, 127, 1860 1.2.1

Bresenham, J. E. 1965, IBM Systems J., 4, 25 2.2.5

Broadhurst, T., Benítez, N., Coe, D., et al. 2005, ApJ, 621, 53 1.1.2

Cabanac, R. A., Alard, C., Dantel-Fort, M., et al. 2007, A&A, 461, 813 1.2.1, 1.2.2

Casertano, S., de Mello, D., Dickinson, M., et al. 2000, AJ, 120, 2747 2.1.2

Castro, P. G., Heavens, A. F., & Kitching, T. D. 2005, PhRvD, 72, 023516 1.1.1

Catmull, E. & Rom, R. 1974, in Computer Aided Geometric Design: Proceedings of a Conference Held at the University of Utah, Salt Lake City, Utah, March 18-21, 1974 (New York: Academic Press), 317–326 2.3.7

Cooray, A. R. 1999, ApJ, 524, 504 1.1.3

Dalal, N., Holder, G., & Hennawi, J. F. 2004, ApJ, 609, 50 1.1.3

Delaunay, B. 1934, Bulletin of the Academy of Sciences of the USSR, 793 2.3.4

Dijkstra, E. W. 1959, Numerische Mathematik, 1, 269 2.3.4

Dunkley, J., Komatsu, E., Nolta, M. R., et al. 2009, ApJS, 180, 306 1.1.3

Dyson, F. W., Eddington, A. S., & Davidson, C. 1920, Royal Society of London Philosophical Transactions Series A, 220, 291 1

Faure, C., Kneib, J.-P., Covone, G., et al. 2008, ApJS, 176, 19 4, 4.1, 5.1

Fedeli, C. & Bartelmann, M. 2007, A&A, 461, 49 1.1.3

Fedeli, C., Bartelmann, M., Meneghetti, M., & Moscardini, L. 2008, A&A, 486, 35 1.1.3

Fedeli, C., Meneghetti, M., Bartelmann, M., Dolag, K., & Moscardini, L. 2006, A&A, 447, 419 1.1.3

Flores, R. A., Maller, A. H., & Primack, J. R. 2000, ApJ, 535, 555 1.1.3

Fortune, S. 1986, in SCG '86: Proceedings of the second annual symposium on Computational geometry (New York: ACM), 313–322 2.3.4

Freund, Y. & Schapire, R. E. 1996, in Thirteenth International Conference on Machine Learning (San Francisco: Morgan Kaufmann), 148–156 1.2.2

Friedman, J., Hastie, T., & Tibshirani, R. 1998, Annals of Statistics, 28, 2000 1.2.2

Fruchter, A. & Hook, R. N. 1997, in Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, Vol. 3164, 120–125 2.1.2, 2.1

Fruchter, A., Sosey, M., Hack, W., et al. 2009, The MultiDrizzle Handbook, version 3.0 (Baltimore: STScI) 2.1.2

Gladders, M. D., Hoekstra, H., Yee, H. K. C., Hall, P. B., & Barrientos, L. F. 2003, ApJ, 593, 48 1.1.3

Grossman, S. A. & Narayan, R. 1988, ApJL, 324, L37 1.1.3

Harvey, J. E. & Ftaclas, C. 1995, Applied Optics, 34 2.3.9

Hayya, J., Armstrong, D., & Gressis, N. 1975, Management Science, 21, 1338 2.2.2

Heavens, A. 2003, MNRAS, 343, 1327 1.1.1

Heavens, A. F., Kitching, T. D., & Taylor, A. N. 2006, MNRAS, 373, 105 1.1.1

Horesh, A., Ofek, E. O., Maoz, D., et al. 2005, ApJ, 633, 768 1.1.3, 1.2.2

Kaiser, N., Aussel, H., Burke, B. E., et al. 2002, in Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, ed. J. A. Tyson & S. Wolff, Vol. 4836, 154–164 5.2

Kaiser, N. & Squires, G. 1993, ApJ, 404, 441 1.1.1, 2.2.3

Kass, M., Witkin, A., & Terzopoulos, D. 1988, International Journal of Computer Vision, 1, 321 2.3.5

Kaufmann, R. & Straumann, N. 2000, Annalen der Physik, 9, 384 1.1.3

Kovner, I. 1989, ApJ, 337, 621 1.1.3

Kubo, J. M. & Dell'Antonio, I. P. 2008, MNRAS, 385, 918 1.2.2

Leauthaud, A., Massey, R., Kneib, J.-P., et al. 2007, ApJS, 172, 219 2.3.10

Lenzen, F., Schindler, S., & Scherzer, O. 2004, A&A, 416, 391 1.2.2

Leonard, A., Goldberg, D. M., Haaga, J. L., & Massey, R. 2007, ApJ, 666, 51 1.1.2

Li, G.-L., Mao, S., Jing, Y. P., et al. 2005, ApJ, 635, 795 1.1.3

Li, G. L., Mao, S., Jing, Y. P., et al. 2006, MNRAS, 372, L73 1.1.3

Lynds, R. & Petrosian, V. 1986, in Bulletin of the American Astronomical Society, Vol. 18, 1014–+ 1.1

Marshall, P. J., Hogg, D. W., Moustakas, L. A., et al. 2009, APJ, 694, 924 1.2.1, 3.2

Massey, R. & Refregier, A. 2005, MNRAS, 363, 197 1.2.2

Meneghetti, M., Bartelmann, M., & Moscardini, L. 2003, MNRAS, 340, 105 1.1.3

Meneghetti, M., Bolzonella, M., Bartelmann, M., Moscardini, L., & Tormen, G. 2000, MNRAS, 314, 338 1.1.3

Meneghetti, M., Melchior, P., Grazian, A., et al. 2008, A&A, 482, 403 2.3.8, 3, 3.1

Merten, J., Cacciato, M., Meneghetti, M., Mignone, C., & Bartelmann, M. 2009, A&A, 500, 681 1.1.2

Miralda-Escude, J. 1993, ApJ, 403, 497 1.1.3

Moldovan, T. M. 2009, Automatic detection of strong gravitational lensing using AdaBoost in shapelet space, Bachelor's thesis, Brown University 1.2.2

Narayan, R. & Bartelmann, M. 1997, Proceedings of the 1995 Jerusalem Winter School, eds. A. Dekel and J.P. Ostriker 1.1.1

Oguri, M., Lee, J., & Suto, Y. 2003, ApJ, 599, 7 1.1.3

Puchwein, E., Bartelmann, M., Dolag, K., & Meneghetti, M. 2005, A&A, 442, 405 1.1.3

Seidel, G. & Bartelmann, M. 2007, A&A, 472, 341 2.2, 2.2.5

Soucail, G., Fort, B., Mellier, Y., & Picat, J. P. 1987, A&A, 172, L14 1.1

Soucail, G., Kneib, J.-P., & Golse, G. 2004, A&A, 417, L33 1.1.2

Tereno, I., Doré, O., van Waerbeke, L., & Mellier, Y. 2005, A&A, 429, 383 5.2

Thompson, D., Pozzetti, L., Hasinger, G., et al. 2001, A&A, 377, 778 1.1.3

Torri, E., Meneghetti, M., Bartelmann, M., et al. 2004, MNRAS, 349, 476 1.1.3

Wambsganss, J. 2006, Gravitational Microlensing, 453–+ 1.1.1

Wambsganss, J., Bode, P., & Ostriker, J. P. 2004, ApJL, 606, L93 1.1.3

Zaritsky, D. & Gonzalez, A. H. 2003, ApJ, 584, 691 1.1.3

Acknowledgements

For his dedicated organisational support, scientific advice and for introducing me to fellow scientists in the strong lensing community I thank my doctoral adviser Prof. Matthias Bartelmann, who made this thesis possible. My heartfelt thanks go to Emanuel, for keeping the IT infrastructure running in addition to his work as a PhD student, even if he had to stay till late in the evening. I thank the past and present ITA cosmology group for being good colleagues that I enjoyed working with and for making the coffee breaks much more interesting.

I thank Johannes and Ekaterina for becoming good friends to me and for being patient with me during the final phase of this thesis. Johannes and Isabel I also thank for new insights into music, advanced botany, politics and on simulating stellar formation, all of which I greatly enjoyed discussing.

I thank Albert both for his demanding Karate training that provided some balance to my work in the office and for his rare talent for humour. Svetlana I thank for diverting my attention and cheering me up when I needed it.

I thank my parents for supporting me even when I rarely found the time to go home during the final phase of the PhD.

Finally, I also want to express my appreciation of the WikipediA community, whose efforts were immensely helpful to me and I believe many other scientists today in looking up basic and sometimes even detailed scientific information, and thank the Sonderforschungsbereich 439, Galaxies in the young Universe, of the Deutsche Forschungsgemeinschaft, for supporting my research financially.

Erklärung

Ich versichere, dass ich diese Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

Declaration

I confirm that I produced this work by my own hand, and that I did not use any other resources and help except for those mentioned.

Heidelberg, 29. September 2009:

....
Unterschrift / Signature