

Yet another object detection application (YODA)

Object detection and photometry for multi-band imaging data

N. Drory*

Universitäts-Sternwarte München, Scheinerstr. 1, 81679 München, Germany

Received 18 August 2002 / Accepted 8 October 2002

Abstract. We present a software package performing object detection, photometry and star-galaxy classification specially developed to cope with *multi-band* imaging data as are common in modern extragalactic imaging surveys, and to be modular and therefore easily adaptable to specific needs. Specifically, the package is designed to work under conditions of inhomogeneous background noise across the detection frame, and, additionally, perform accurate aperture photometry in image sets not sharing a common coordinate system or pixel scale as is often the case in present-day extragalactic survey work. The package has been used for the MUNICS survey. The package is described in detail and its performance is compared to existing software using real imaging data.

Key words. methods: data analysis – techniques: image processing – galaxies: photometry

1. Introduction

Large two-dimensional CCD detectors in the optical regime and also large array detectors in the near-IR have fostered the realisation of deep *wide-area* imaging surveys also covering *multiple bands* in wavelength. Such surveys have become a major tool especially in extragalactic research. Automated object extraction is essential for such projects, and considerable effort has been put into the development of such systems by many authors. Some earlier systems were developed specifically for photographic plate surveys, e.g. Kron 1980, FOCAS (Jarvis & Tyson 1981), COSMOS (Beard et al. 1990), the APM package (Maddox et al. 1990), while more recently developed packages were aimed at CCD based surveys, e.g. PPP (Yee 1991), and SExtractor (Bertin & Arnouts 1996). Common to all these systems is their focus on single-band datasets, allowing for processing of multi-band datasets only under tough prerequisites, namely that the images share a common coordinate system and plate scale.

Today's large-area multi-band imaging surveys commonly suffer from various inhomogeneities in the dataset. Firstly, the use of different imaging instruments in different passbands results in variations in the pixel scale and in the field distortion from one passband to another. Secondly, variability in transparency, sky brightness, and seeing conditions will cause the signal-to-noise ratio to vary across the survey area. Thirdly, constructing mosaics out of smaller images will cause overlapping regions of the resulting frame to be deeper. And finally, illumination gradients will cause the background to vary even

across single frames. For most scientific applications of such surveys it is nevertheless necessary to measure colours accurately, e.g. for studies of the sources' stellar populations or the determination of photometric redshifts. This means photometry in equal (physical) size apertures in all wavebands has to be performed for each source in such an inhomogeneous dataset.

The software package presented in this paper, YODA, is designed to handle inhomogeneous multi-band imaging data which do not share a common pixel coordinate system and have position dependent background noise in an automated way. Also, modularity in the sense of allowing for additional processing to be added easily at each step of the pipeline, most importantly during evaluation of the objects' properties, was considered to be desirable. The software has been developed for, and used in the MUNICS extragalactic multi band imaging survey (Drory et al. 2001) for object extraction, photometry, and source classification, while keeping general purpose use in mind. The present work is meant both as a documentation for these data processing steps in MUNICS as well as a reference for users of the software.

This paper is organised as follows: Sect. 2 gives a brief overview of the YODA package. The following three sections are devoted to in-detail discussion of the main building blocks of the package, namely object detection (Sect. 3), photometry (Sect. 4), and image classification (Sect. 5). Section 6, finally, summarises this work.

2. Overview of YODA

Object detection is usually performed using a single image which can be either a frame in one of the available passbands

* e-mail: drory@usm.uni-muenchen.de

(for selection reasons) or a weighted sum of frames in several of the available passbands (for gaining signal to reach fainter detection limits). Subsequent photometry has then to be performed in the *same apertures* applied in all other images.

YODA input, therefore, consists of a “master frame” which is used for object detection as a first step. During this stage, the background value and the rms of the noise in the background are computed as a function of position on the master frame. Pixels lying above a certain threshold relative to the *local* background and noise are then assembled to form objects. The basic properties of these objects, like first and second moments and isophotal fluxes, are written to a catalogue file, the “master catalogue”.

The master catalogue can then be processed further through the photometry and classification steps if desired, for the purpose of pre-selecting objects to enter the final catalogues, or it can be used directly for processing the multi-band data. The next step is to transform the positions of the objects in the master catalogue to all frames to be measured and to choose suitable apertures. The subsequent photometry and classification phases use the information in the master catalogue (describing the objects’ shapes) and the transformed coordinates and apertures to analyse the objects in all images, irrespective of their actual detection in images other than the master frame.

To allow the user to easily extend the package and adapt it to specific needs, for example by adding custom processing steps, all input and output of the various processing stages are handled via simple columnar-format text files. Furthermore, inclusion of user-supplied functions operating on the current object is possible at each step.

YODA’s processing steps can be executed consecutively on a single frame using a single YODA invocation, or individually, using distinct invocations of the software. This gives the user the possibility to process the output of any YODA phase before invoking the next phase. It also allows each of YODA’s processing steps to be fed catalogues generated otherwise, and using only specific capabilities of YODA where needed, e.g. to use only the classifier on catalogs generated by another source.

YODA is implemented in C++, and is developed under UNIX. It should be portable to essentially any hardware and software architecture where a reasonably ANSI-compliant C++ compiler is available, ideally GCC. It has been so far verified to run under Linux/x86, Linux/PowerPC, Solaris/SPARC, Digital-UNIX/Alpha, and MacOS X/PowerPC using GCC versions 2.95.2, 3.0 and 3.1.

The performance of the package is roughly 1 Mpixel/s for background estimation and detection, and 250 objects/s for aperture photometry, and finally 100 objects/s for star-galaxy classification, measured on a 1 GHz class present day PC or workstation hardware.

The software is publicly available, either by request from the author or on the web at <http://www.usm.uni-muenchen.de/people/drory/yoda/index.html>.

3. Object detection

Two principle approaches to object detection have been proposed in the literature, searching for local maxima

Kron (1980); Yee (1991) and collecting consecutive pixels lying above a given threshold relative to the background noise, often called “thresholding”, Jarvis & Tyson (1981), Bertin & Arnouts (1996). While searching for local maxima is better in the presence of crowding since close-by objects are naturally detected as distinct objects, it is less robust at low signal-to-noise and with low surface brightness objects.

We have therefore implemented thresholding, after convolution of the image with a Gaussian of *FWHM* equal to the seeing in the frame. The optimum correlation kernel for the detection of faint sources would be the image’s PSF Irwin (1985), but for all practical purposes a Gaussian approximation is good enough, as comparisons have shown.

3.1. Background determination, thresholding, and object assembly

Thresholding effectively means collecting pixels above a certain surface brightness and signal-to-noise ratio limit. Usually this limit is expressed in terms of the variance of the background noise present in the image. Of course one would like the statistical significance of such a detection to be independent of position in the image. Therefore an accurate *local* estimate of the background value and noise is needed.

The background and the rms noise in the background are estimated as a function of position in the convolved image by inspecting the histogram of pixel values in rectangular regions of adjustable size in the image. The pixels within each region are $\kappa - \sigma$ filtered to minimise impact of bright objects and outliers, and the histogram of their values is computed. The background and the rms value in each grid cell is estimated to be the mode and half of the width of the distribution at $1/e$ of the mode, respectively. The final background and rms values for each pixel are produced by bilinear interpolation between the grid cells. If necessary, background estimation can be improved by an iterative process of masking out pixels assigned to objects after thresholding and redetermining the background. This can improve background estimation significantly in the presence of bright stars, large galaxies, or moderate crowding, making object detection more robust and uniform in the vicinity of such objects.

Objects are detected by requiring that a minimum number N of consecutive pixels lie at least a certain thresholding factor t of the local rms above the local background. Consecutive is defined here as that at least one of the eight closest neighbours be above the threshold.

The values for N and t reflect a compromise between completeness at a given signal-to-noise – the magnitude limit – and the number of tolerable spurious detections per unit image area (Saha 1995; Neuschaefer et al. 1995; Harris 1990), depending on the form and size of the PSF and the pixel scale in addition to the characteristics of the noise present in the image. They have to be chosen carefully for any individual application of the data.

3.2. Splitting of multiple components

Close objects can overlap at the detection isophote, in which case they will be wrongly assembled into a single object by the thresholding phase. Therefore each object is re-examined by thresholding it at a number of linearly spaced, increasingly higher isophotes up to a fixed fraction of its maximum flux.

If an object decomposes into several components at some isophote, the component containing the pixel of maximum flux of the original object retains this identity, the other components being considered as new objects. The new objects are added to the end of the catalogue (their “detection isophote” set to the current splitting isophote) and the original object is continued to be examined. To avoid splitting noise peaks in the wings of objects, the subcomponents are required to consist of an adjustable minimum number of consecutive pixels to be regarded as real.

3.3. Evaluation of shape parameters

After de-blending, structural parameters within the detection isophote for each object are computed, namely the intensity-weighted radius,

$$R_e = \frac{\sum rI(r)}{\sum I(r)},$$

the intensity-weighted first and second moments,

$$C_x = \frac{\sum xI(x,y)}{\sum I(x,y)}, \quad C_y = \frac{\sum yI(x,y)}{\sum I(x,y)},$$

$$I_{xx} = \frac{\sum (x-C_x)^2 I(x,y)}{\sum I(x,y)}, \quad I_{yy} = \frac{\sum (y-C_y)^2 I(x,y)}{\sum I(x,y)},$$

$$I_{xy} = \frac{\sum (x-C_x)(y-C_y)I(x,y)}{\sum I(x,y)},$$

the elongation,

$$E = \frac{\sqrt{(I_{yy} - I_{xx})^2 + (2I_{xy})^2}}{I_{xx} + I_{yy}},$$

and the “full width at half maximum”,

$$FWHM = 2 \sqrt{\ln 2 (I_{xx} + I_{yy})},$$

where $I(x, y)$ denotes the background subtracted intensity at pixel (x, y) , and $r^2 = (x - C_x)^2 + (y - C_y)^2$, the Euclidean distance from the object’s centre. The sums are to be taken over all pixels within the detection isophote.

For computing elliptical aperture shapes in the case where adaptive-size (“Kron-like”) apertures are to be used for photometry, the major and minor axes A and B , as well as the position angle θ , are computed based on the second moments:

$$A^2 = \frac{I_{xx} + I_{yy}}{2} + \sqrt{\left(\frac{I_{xx} - I_{yy}}{2}\right)^2 + I_{xy}^2},$$

$$B^2 = \frac{I_{xx} + I_{yy}}{2} - \sqrt{\left(\frac{I_{xx} - I_{yy}}{2}\right)^2 + I_{xy}^2}, \quad \text{and,}$$

$$\tan 2\theta = 2 \frac{I_{xy}}{I_{xx} - I_{yy}}.$$

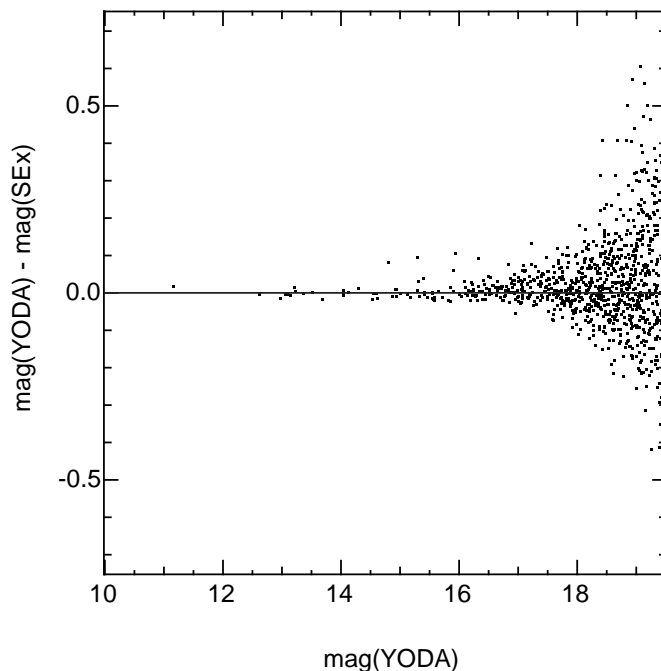


Fig. 1. Comparison of photometry in 5'' circular apertures between SExtractor and YODA. Outliers are due to differences in the treatment of blended objects. The data are taken from K -band images of high galactic latitude fields from the MUNICS survey (Drory et al. 2001), obtained at the 3.5-m telescope at Calar Alto Observatory. Tests using simulated data yield similar results.

The background flux, the flux at the detection isophote and the total flux within the detection isophote as well as the central and mean surface brightness are also computed. These quantities comprise the final output of the object detection and assembly phase.

4. Photometry

The photometry stage within the YODA package is capable of performing photometry in fixed size circular apertures and adaptive size elliptical apertures. Curves of growth in a series of concentric apertures can be computed for each object on demand, too.

The input for the photometry stage consists of pixel coordinate lists and aperture descriptions consisting of major axis, minor axis, and position angle. These can be taken from the output of the detection stage, with any geometric transformation applied to the coordinates (and apertures, if needed) to map between the detection frame and the frames to be measured.

The photometry algorithm sums up the sky subtracted flux in each aperture. The sky is determined locally for each object from pixels in an annulus of given distance from the object’s aperture and width, having the same form as the aperture. Pixels belonging to neighbouring objects that intersect with the sky annulus are masked out during this process. The sky is measured by examining the histogram of sky pixel values, taking the mode as the final sky value and the width of the histogram as the error per sky pixel.

The error of the resulting object flux is estimated as

$$\Delta f = \sqrt{\frac{f}{G} + N_{\text{pix}} \sigma_{\text{sky}}^2 + \frac{N_{\text{pix}}^2 \sigma_{\text{sky}}^2}{N_{\text{sky}}}}, \quad (1)$$

where f denotes the sky subtracted total flux within the aperture, G is the gain value in e^- per ADU, N_{pix} is the number of pixels in the aperture, N_{sky} is the number of pixels in the sky annulus, and σ_{sky} the error per sky pixel as derived from the histogram of sky values.

For the case where aperture photometry is applied to images with large pixel scale relative to the aperture size, the software offers the possibility to subsample the pixels within the aperture by an adjustable factor to account more accurately for signal in pixels touching the rim of the aperture.

Finally, Fig. 1 shows a comparison of magnitudes and errors measured in $5''$ circular apertures between YODA and SExtractor. The magnitudes agree well, with outliers being due to the somewhat different treatment of blended objects.

5. Star-galaxy separation

Three different principal approaches have been taken so far on the problem of classifying images of astronomical objects into distinct categories.

The first approach, a Bayesian one, uses the two-dimensional intensity distribution (including its noise characteristics) directly (Sebok 1979; Valdes 1982) to calculate the likelihood of a certain model given the observed counts. Here, the models can either be based on light distributions of observed galaxies (as in Sebok's version), or simply on the observed PSF as a template for an unresolved source (as in Valdes' implementation). This is particularly interesting if the discrimination between unresolved and resolved sources is a sufficient classification, as is generally the case for star-galaxy separation.

The second approach, often called the parametric approach, analyses functions of the two-dimensional intensity distribution of the objects' images. Classification is then performed by defining hyper-surfaces in this parameter space bounding the regions occupied by distinct classes of objects, e.g. magnitude vs. peak intensity, the r_{-2} moment of Kron (1980), and the ψ parameter used in the APM survey (Maddox et al. 1990).

The third approach, also the most recently developed one, uses artificial neural networks to classify objects (Odewahn et al. 1992; Bertin 1994; Bertin & Arnouts 1996; Bazell & Peng 1998).

While all methods are successful and consistent at high signal-to-noise, classification gets very difficult as one gets closer to the limiting brightness of one's data. This is not only because of the decrease in signal-to-noise, but also because images of faint objects look more and more similar to images of unresolved sources as their detectable area above the surface brightness limit decreases.

We have chosen to use the Bayesian approach for two reasons: firstly, with the likelihood analysis it is possible to derive a meaningful *probability* that a given object's image is produced by a certain class, say unresolved point-like sources.

Secondly, by calculating likelihoods for different model templates, one still gains information about the object's appearance, for example in terms of "compact source", "fuzzy star", or "stellar core" etc., even if a strict classification in the sense of star or galaxy fails. Another way of stating this is saying that the likelihood reflects a measurement of a physical parameter of the model used, in our case the width relative to the PSF, since we will use the PSF in the image as a model.

Note that neural-networks as those used for object detection behave like a Bayesian classifier with the subtlety that they have a built-in prior corresponding to the statistical distribution of object classes in the training set. So at decreasing signal-to-noise, the neural network output converges towards the distribution of classes in the training set, or to the ratio of faint stars to galaxies in the training set when only those two classes are of interest.

YODA's classification stage closely follows the *resolution classifier* presented in Valdes (1982), which is also used in the FOCAS package. The main difference lies in that Valdes uses a set of template images discretely sampling the parameter space of his models to find the template of maximum likelihood (for computing time reasons) while we use simplex maximisation to find the parameters of the most likely model, thus scanning parameter space continuously.

We briefly outline the algorithm and present some tests we have performed. For a detailed description of the algorithm's motivation and background we refer the reader to Valdes' thorough treatment in his original paper.

Mathematically stated, a Bayesian classifier picks a class C such that an object of the chosen class would be most likely to produce the observed image counts $I(x, y)$, thus maximising $P(C|I)$. Applying Bayes' theorem, we can state $P(C|I)$ in terms of $P(I|C)$, the probability of observing the counts $I(x, y)$ given an object of class C , which we can compute using the noise characteristics of the measurement and the expected intensity distribution of an object of class C .

The resolution classifier is based on the observation that all resolved sources will give better fits to a resolved template than to a totally unresolved template (the PSF) and therefore the PSF and broadened versions of the PSF can be used as model classes C independent of any assumptions on the intrinsic profiles of galaxies present in the data. Models are generated solely from the measured PSF $\Phi(\mathbf{r})$ in the image, where \mathbf{r} denotes the position relative to the source's centre.

Two sets of models are used in our case, one consisting of narrowed and broadened versions of the pure PSF

$$T(\mathbf{r}) = \Phi\left(\frac{\mathbf{r}}{\alpha}\right), \quad (2)$$

with the scale parameter α ranging from 0.7 to 10. The second set consists of objects with a stellar core but some additional flux from a slightly broadened component

$$T(\mathbf{r}) = \beta\Phi\left(\frac{\mathbf{r}}{2}\right) + (1 - \beta)\Phi(\mathbf{r}) \quad (3)$$

with $0 < \beta < 1$. Thus $C = C(\alpha, \beta)$ for our classifier and we can write $P(I|C)$ as $P(I|\alpha, \beta)$ which is the quantity we have to maximise with respect to α and β . For unresolved, stellar images we expect the maximum to occur at $(\alpha = 1, \beta = 0)$.

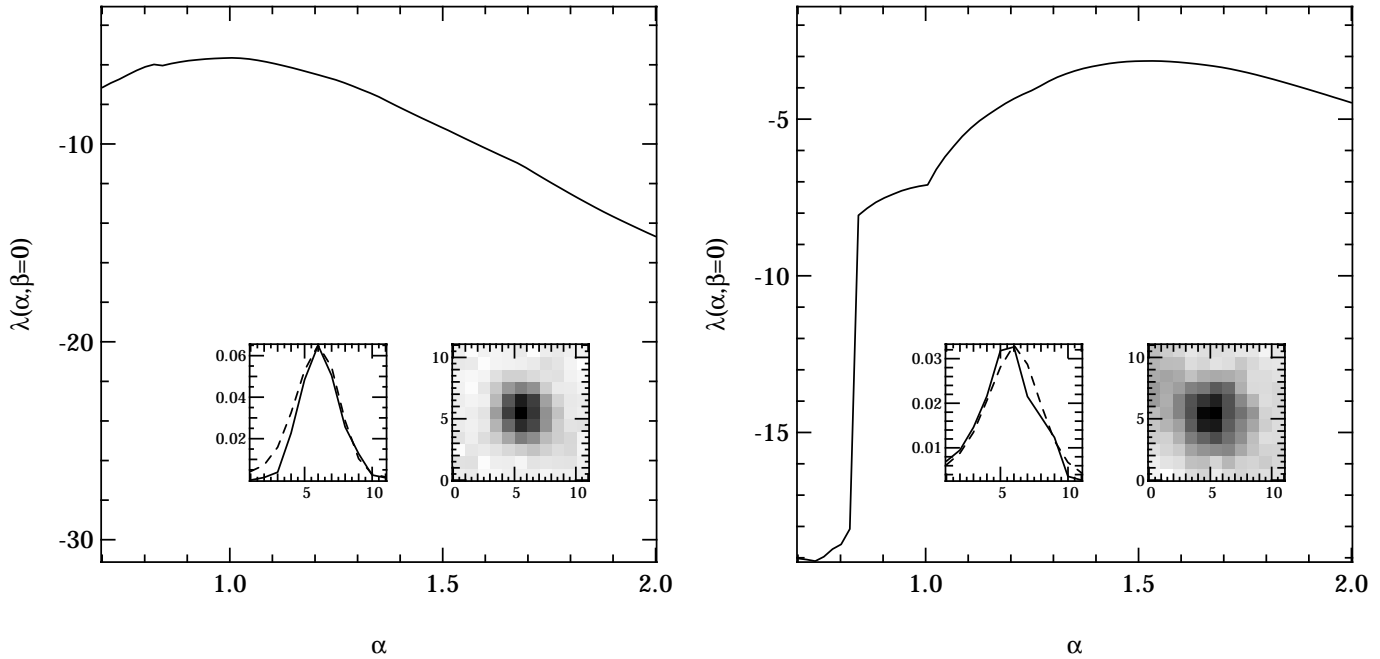


Fig. 2. Cuts through the likelihood surface at $\beta = 0$ for a faint stellar source (**left panel**) and a faint clearly resolved source (**right panel**). The insets show the profile of the object (solid line) and the profile of the model of maximum likelihood (dashed line) in units of ADUs as a function of pixel position, as well as the image of the source. The data are taken from *R*-band images of high galactic latitude fields from the MUNICS survey (Drory et al. 2001).

Using the Poisson nature of the photon count noise in the image, the likelihood function $\lambda = \ln P(I|\alpha, \beta)$ becomes

$$\lambda = \sum_k \left[\frac{I_k}{LT_k + I_{\text{sky}}} \left(1 - \ln \frac{I_k}{LT_k + I_{\text{sky}}} \right) - 1 \right], \quad (4)$$

where the sum is to be taken over all image pixels, L is the models' luminosity, I_k and $T_k = T_k(\alpha, \beta)$ are the object's and the models' counts in the pixel k , respectively, and I_{sky} is the background intensity at the object's position. This function is maximised using a simplex algorithm with respect to the luminosity L and (α, β) .

Figure 2 shows a cut through the likelihood surface at $\beta = 0$ for two faint sources, an unresolved stellar source and a clearly resolved source, illustrating the functioning of the method.

The crucial point is clearly the determination of the PSF in the image to be processed. For this a list of high signal-to-noise stellar sources has to be made known to YODA. This can be done semi-automatically, by letting YODA try to find bright stellar sources by their unique location in the *FWHM* vs. magnitude diagramme, and subsequent inspection of these sources, or by supplying a list of manually selected sources in the first place.

If a classification into distinct classes, e.g. stars, fuzzy stars, compact galaxies, galaxies, etc. is desirable, this is possible by defining a region in (α, β) -space for each class C_i . Then the probability that a source belongs to a specific class can be directly given as

$$P(I|C_i) = \int_{\alpha, \beta \in C_i} P(I|\alpha, \beta) d\alpha d\beta. \quad (5)$$

Figures 3 and 4 show the application of this classifier to artificial data as well as real images taken at various ground based telescopes. Also shown are results of applying SExtractor's neural network based classifier to the same datasets.

To assess the performance of the classifier, we need datasets with a known morphology for all sources. We have chosen to use data containing practically only stars, therefore we use images of the outskirts of the globular clusters M 92 and NGC 288, as well as an image of the surroundings of the Local Group dwarf galaxy IC10. These images have been taken at different telescopes – the Wendelstein 0.8-m, the Calar Alto 2.2-m, and the ESO VLT – to cover a realistic range in imaging characteristics of current instruments. The artificial data were created using the IRAF package ARTDATA with Moffat-type profiles.

The artificial data can be used to define the region in (α, β) -space occupied by stars, and the real data to check the robustness of the classification under different imaging conditions. The region in parameter space occupied by unresolved sources can be restricted to $0.8 \leq \alpha \leq 1.2, \beta = 0$, in accordance to Valdes' finding, over all the range of conditions covered by these datasets. The objects that scatter away from the stellar sequence are due to crowding, especially those objects which have a high value of β . This effect can be seen in the image of the dwarf galaxy IC10 even for very bright sources, as may be expected because of the presence of background flux from the galaxy causing the objects to have extra flux in the wings. Bright objects that scatter away in α usually are blended with a close-by object.

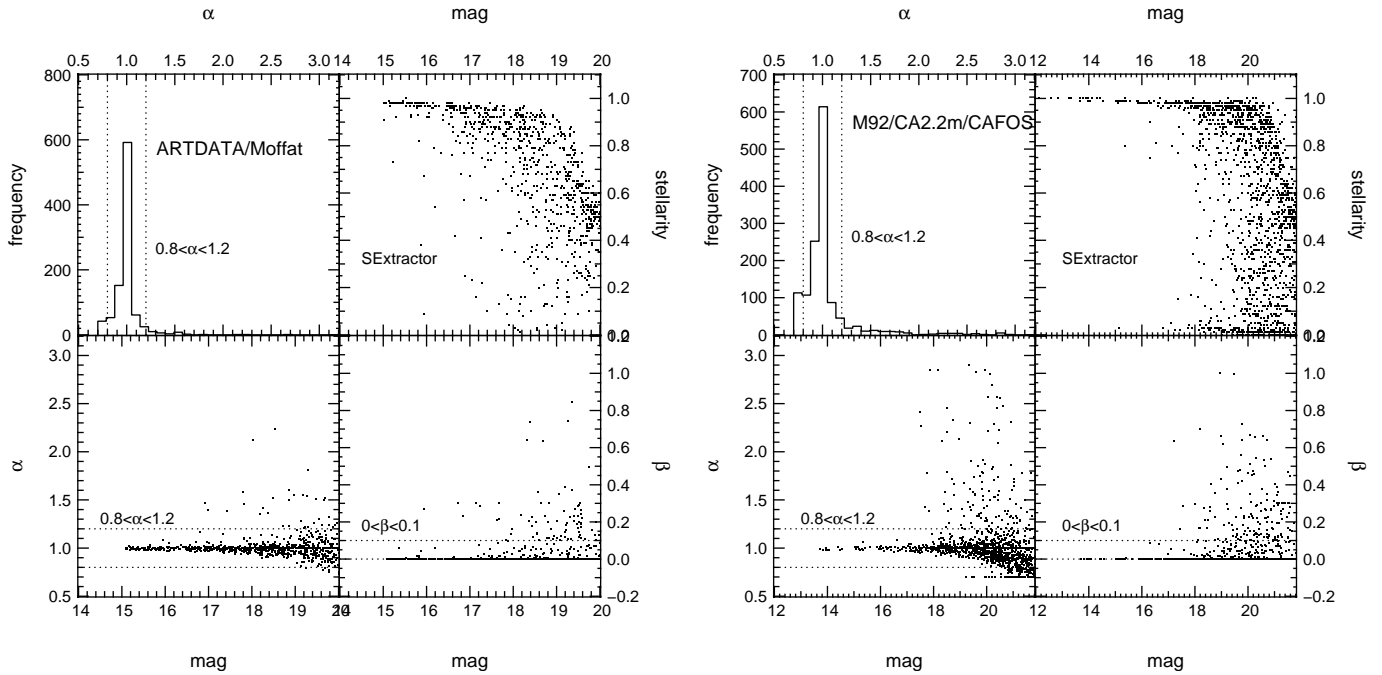


Fig. 3. Results of applying the resolution classifier to artificial stellar images having Moffat-type profiles created with IRAF/ARTDATA (**left**), and to an R -band image of the globular cluster M 92 (**right**) in (α, β) -parameter space in comparison with the *stellarity* parameter of SExtractor. The image of M 92 was obtained at the Calar Alto 2.2-m telescope using the focal reducer CAFOS in imaging mode. The dotted lines mark the regions in (α, β) -space populated by unresolved (stellar) images. The **top-left panel** shows a histogram of the distribution of the scale parameter α , the **bottom-left panel** shows α vs. magnitude and the **bottom-right panel** shows β vs. magnitude. The **top-right panel** shows the output of SExtractor's neural net based classifier.

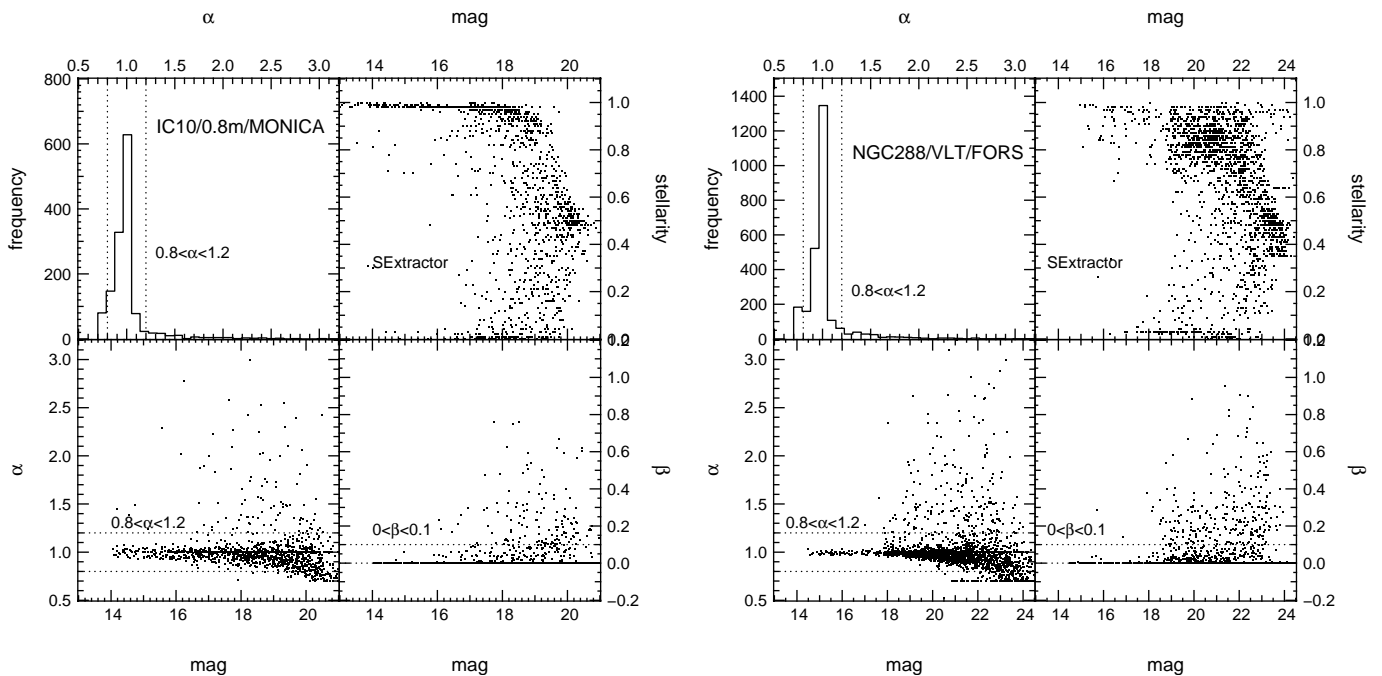


Fig. 4. Results of applying the resolution classifier to an R -band image of the (resolved) dwarf galaxy IC10 (**left**) obtained at the 0.8-m telescope at Wendelstein Observatory and a V -band image of the globular cluster NGC 288 (**right**) obtained with FORS at VLT UT1. Subpanels as in Fig. 3.

It is worth noting that the resolution classifier does not scatter images of point sources systematically away from the stellar

locus as they get fainter. Clearly, the signal-to-noise becomes too low to say anything meaningful about the objects' light

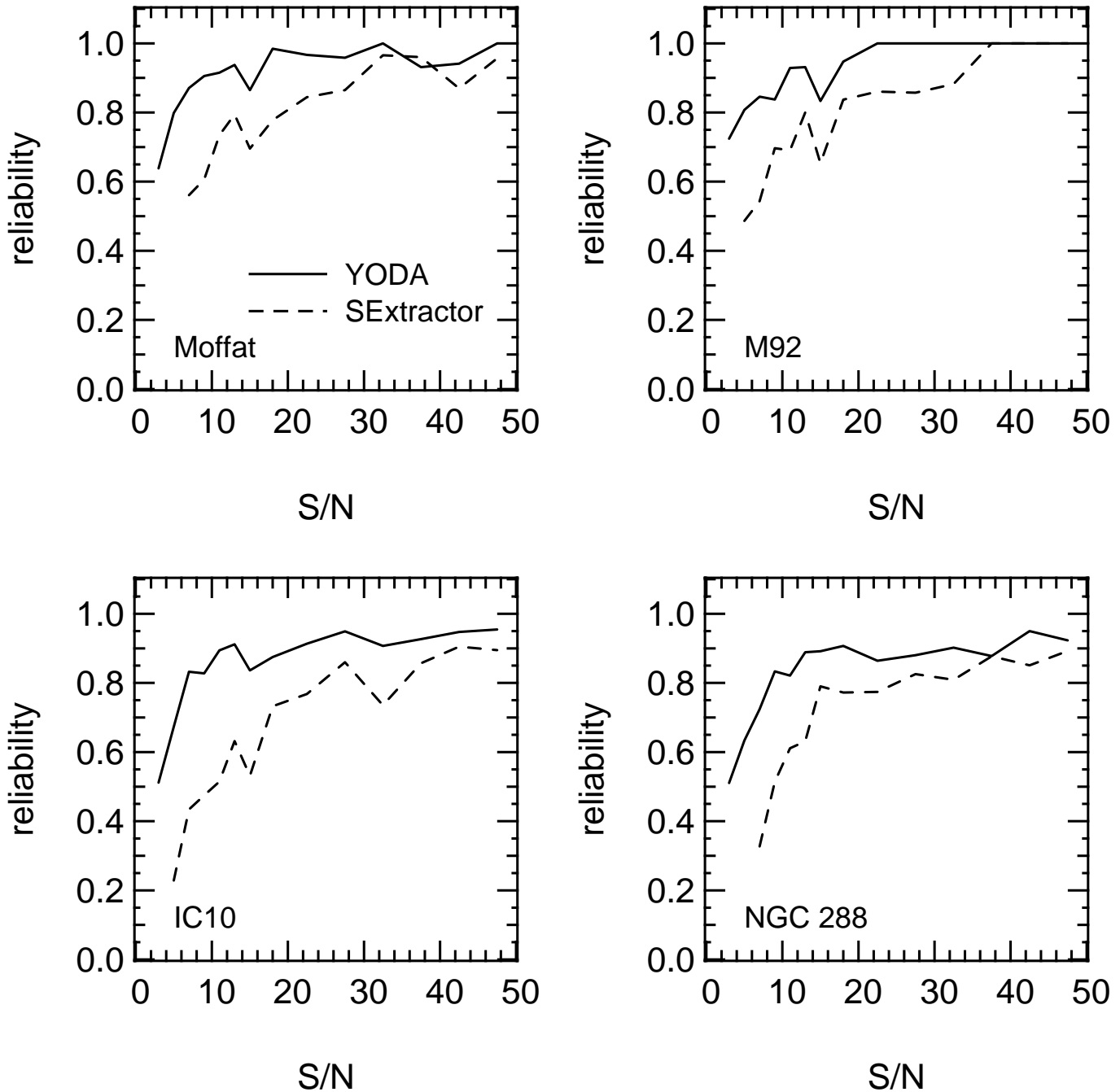


Fig. 5. Comparison of classification reliability of point-sources as a function of the source's total signal-to-noise. The data are the same as in Figs. 3 and 4.

profile as we approach the detection limit. But at least the information on the compactness of the source is recovered, expressing itself in small values of α . If resolved sources were present in the image, those would overlap with the stellar locus at faint magnitudes when their images become more and more compact.

In the case of SExtractor's artificial neural network based classifier, faint objects tend to converge towards a value of the stellarity parameter of 0.5, presumably the statistical ratio of stars to galaxies in the training sample. Since information on the profile shape of the source cannot be recovered anymore

at low signal-to-noise, the neural net correctly says it cannot classify the objects and converges towards a statistical answer.

Figure 5 shows the success rate of the classification for point sources as a function of the source's total signal-to-noise ratio for the data presented above, both for the resolution classifier and SExtractor. A successful classification is defined as stellarity parameter greater than 0.5 in the case of SExtractor, and as $0.8 \leq \alpha \leq 1.2, \beta = 0$ for the resolution classifier.

The comparison with SExtractor's classification shows that the resolution classifier is somewhat more robust at lower

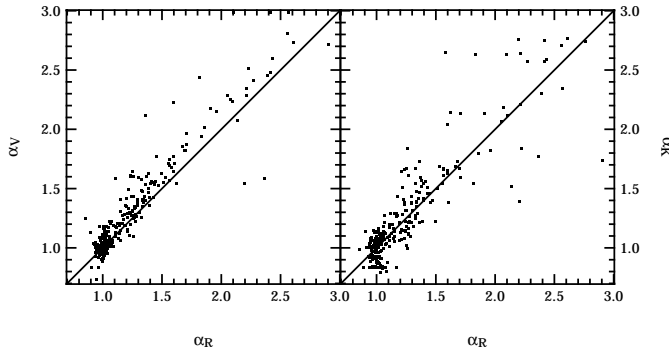


Fig. 6. Comparison of the scale parameter α derived by the resolution classifier between measurements in different wavebands. The **left panel** shows V vs. R , the **right panel** K' vs. R . Plotted are objects with photometric errors smaller than 0.1 mag taken from the MUNICS survey (Drory et al. 2001). The V and R -band data were obtained at the 2.2-m telescope, the K' -band data were obtained at the 3.5-m telescope at Calar Alto Observatory.

signal-to-noise ratio than the application of SExtractor’s neural network *without special training*.

Figure 6 compares the scale parameter α obtained for objects in deep images from the MUNICS survey in the V , R , and K bands, showing that the classifier yields consistent results for the same objects imaged in different wavebands with different instruments. This can be taken advantage of by using the classification in the band where the source has the highest signal-to-noise ratio.

Finally, Fig. 7 shows the Distribution of α vs. magnitude for typical extragalactic imaging data taken from the MUNICS survey, illustrating many of the afore mentioned effects. The sequence at $\alpha = 1$ are stars, the sources at larger values of α are galaxies. A displacement of the stellar sequence with respect to $\alpha = 1$ indicates an overestimated or underestimated PSF width and thus provides a useful test for the quality of the PSF template used.

At fainter magnitudes sources become more and more compact and thus clump at values closer to that for a point source, $\alpha = 1$. At magnitudes fainter than roughly 2 mag above the threshold, where the stellar sequence and the distribution of resolved sources merge, an object in the area defined to be the stellar locus in Figs. 3 and 4, can be either a star or a galaxy. But we have seen that stars do not scatter out of that area, and so all other sources are likely to be resolved.

6. Summary

We have presented YODA, a software package for source detection, photometry and star-galaxy separation developed for use with inhomogeneous multi-band datasets as are common in present extragalactic research. Unlike similar packages introduced in the literature so far, YODA is not restricted to frames sharing a common coordinate system and pixel scale when photometry and classification is performed on sources detected on a separate frame. Additionally, YODA does not assume the background or the noise to be uniform across the frame used

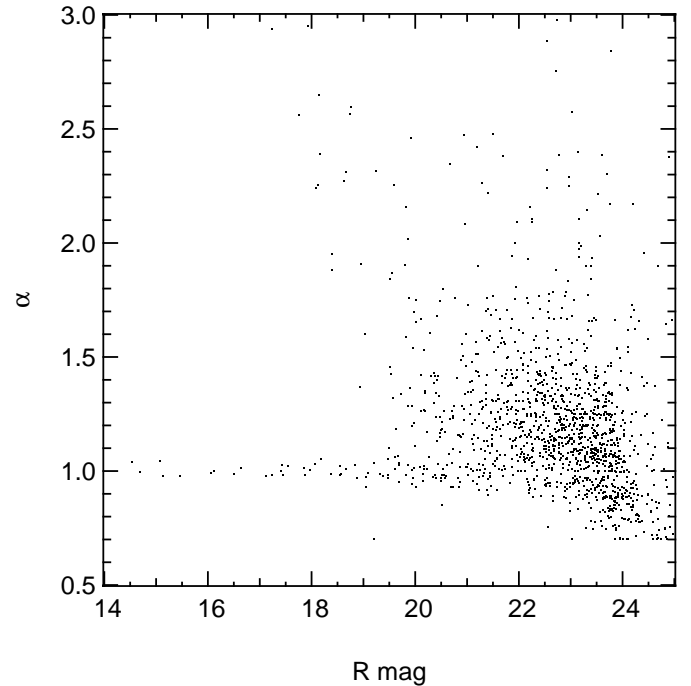


Fig. 7. Distribution of α vs. magnitude for typical extragalactic imaging data, again taken from the MUNICS survey. The sequence at $\alpha = 1$ are stars, the sources at larger values of α are galaxies. At fainter magnitudes sources become more and more compact and thus clump at values closer to that for a point source, $\alpha = 1$.

for detection: the background and its noise are determined locally. The algorithms used in this package were described in detail and the reliability of the photometry and star-galaxy separation was discussed and compared with SExtractor, a package commonly in use.

The main features of YODA can be summarised as follows:

- Thresholding is used for object detection, the background and the variance of the background noise are determined locally.
- Photometry is performed in elliptical apertures. Aperture sizes can be adaptive (Kron-like) or fixed. Curves of growth can be computed, too.
- Apertures derived for sources in an image can be applied to any number of other images after application of a coordinate transform, so that accurate colours can be measured without having to resample the images to a common coordinate system.
- Adding additional parameters to be evaluated for each object can be done easily.
- Star-galaxy separation is done using a resolution classifier similar to the one described in Valdes (1982). This classifier gives very robust results under a wide range of imaging conditions and PSF shapes.

Acknowledgements. My thanks are due to U. Hopp, R. P. Saglia, and R. Bender for many enlightening discussions, as well as to G. Feulner for his valuable input on various aspects of this work, and, together with A. Riffeser, for notoriously confronting me with all the bugs

they could find during testing. C. Gössl is kindly acknowledged for making the Wendelstein data available, U. Hopp, again, for providing the FORS data and his review of the manuscript. This work was supported by the German *Deutsche Forschungsgemeinschaft*, *DFG*, SFB 375 Astroteilchenphysik.

References

- Bazell, D., & Peng, Y. 1998, *ApJS*, 116, 47
Beard, S. M., MacGillivray, H. T., & Thanisch, P. F. 1990, *MNRAS*, 247, 311
Bertin, E. 1994, *Ap&SS*, 217, 49
Bertin, E., & Arnouts, S. 1996, *A&AS*, 117, 393
Drory, N., Feulner, G., Bender, R., et al. 2001, *MNRAS*, 325, 550
Harris, W. E. 1990, *PASP*, 102, 949
Irwin, M. J. 1985, *MNRAS*, 214, 575
Jarvis, J. F., & Tyson, J. A. 1981, *AJ*, 86, 476
Kron, R. G. 1980, *ApJS*, 43, 305
Maddox, S. J., Efstathiou, G., Sutherland, W. J., & Loveday, J. 1990, *MNRAS*, 243, 692
Neuschaefer, L., Ratnatunga, K. U., Griffiths, R. E., & Valdes, F. 1995, *PASP*, 107, 590
Odewahn, S. C., Stockwell, E. B., Pennington, R. L., Humphreys, R. M., & Zumach, W. A. 1992, *AJ*, 103, 318
Saha, P. 1995, *AJ*, 110, 916
Sebok, W. L. 1979, *AJ*, 84, 1526
Valdes, F. 1982, 331, 465
Yee, H. K. C. 1991, *PASP*, 103, 396