Deep VIsible Multi-Object Spectrograph U-band Imaging of the GOODS-South Field

by

Lillian Otteson

A Dissertation Presented in Partial Fulfillment
of the Requirements for the Degree
Bachelor of Science

Approved April 2021 by the
Undergraduate Supervisory Committee:

Rogier A. Windhorst
Rolf A. Jansen
Seth H. Cohen

ARIZONA STATE UNIVERSITY

May 2021

ABSTRACT

We present the optimal resolution and depth mosaics as described in Ashcraft et al. (2018) using deep, ground-based U-band imaging of the Great Observatories Origins Deep Survey (GOODS) South field as part of the near-UV imaging program, UVCANDELS. We utilize the capabilities of the VIsible Multi-Object Spectrograph on the European Southern Observatory (ESO) Very Large Telescope (Nonino et al. 2009). The data consists of 552 single chip images covering an area of 630 square arcminutes. The best resolution mosaic corresponds to a full-width half maximum (FWHM) ≤ 0.8" and utilizes roughly 50% of the data. The best depth mosaic includes images with FWHM ≤ 1.5", which corresponds to 100% of the data. Prior to being combined into the mosaics, the fluxes from the images are corrected to match a 3D-HST photometric catalog of the same data (Skelton et al. 2014). These corrections are made to improve the accuracy and uncertainty in the zero point. These mosaics provide deep U-band data complementary to HST WFC3 F275W and ACS F435W images, and will eventually complement JWST 1-5 μm images.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

INTRODUCTION

U-band imaging is critical to our understanding of galactic evolution. Without it, the bulk of cosmic time beginning after the Universe's peak star formation rates cannot be fully understood. Through the Cosmic Assembly Near-infrared Deep Extragalactic Legacy Survey (CANDELS; Grogin et al. 2011; Koekemoer et al. 2011) program, the *Hubble Space Telescope* (HST) has extensively imaged five fields using multiwavelength observations in order to study this evolution. The UVCANDELS program is a continuation of Hubble's work for CANDELS but it focuses specifically on ultraviolet observations. The UVCANDELS program incorporates four of the CANDELS fields: Great Observatories Origins Deep Survey (GOODS; Giavalisco et al. 2004)-North and GOODS-South fields, the Extended Groth Strip (EGS; Davis et al. 2007), and the Cosmic Evolution Survey (COSMOS; Scoville et al. 2007; Koekemoer et al. 2007) field. This program uses HST's WFC3/F275W filters. The WFC3 F336W filter is not included, however, because it can be successfully substituted by ground-based *U*-band observations.

Ground-based imaging has the considerable advantage of being less expensive than using space-based telescopes, like HST. However, it requires significant image processing in order to create optimized images that mitigate atmospheric affects. In 2009 Mario Nonino et al. presented the results of deep U-band and R-band imaging of the GOODS-South field using the VIsible Multi-Object Spectrograph (VIMOS) instrument on the Very Large Telescope (VLT) in the Atacama Desert in Chile. The images from this work are deeper and better match sensitivity, or signal-to-noise

ratio, of other GOODS multi wavelength photometry, such as HST data from the CANDELS programs. By utilizing a number of image processing techniques, Nonino et al. (2009) were able to create an image mosaic out of the 552 single chip images. The final mosaic from this work had a full width at half maximum (FWHM) of 0.8" and reached a magnitude depth of 29.8 AB (Nonino et al. 2009).

In 2018, Teresa Ashcraft used observations from the Large Binocular Telescope (LBT) of the GOODS-North field and used a sub-stacking method to create two image mosaics: one being the best resolution mosaic, cut off at 0.8" seeing, and the second being the best depth mosaic, cut off at 1.8" seeing (Ashcraft et al. 2018). This stacking method is useful in reducing atmospheric effects associated with ground-based imaging. Ashcraft et al. (2018) found that the optimal resolution mosaic is best for studying bright galaxies, and shows structure in galaxies much more clearly than the best depth mosaic. The optimal depth stack, however, is more sensitive to low surface brightness and is better for studying the faintest objects. Ashcraft et al. (2018) showed that galaxy number counts fall off much more slowly for the best depth mosaic, meaning it detects more of the faintest galaxies. McCabe et al. (2021) presents a sequel to Ashcraft et al. (2018)'s work using LBT data of the COSMOS field.

The work presented here focuses on the GOODS-South field and builds upon on the work of Nonino et al. (2009) and uses the seeing sorted stacking method utilized by Ashcraft et al. (2018) to create an optimal resolution mosaic. We also recreate the mosaic in Nonino et al. (2009), which is referred to as the optimal depth mosaic in this work. We follow a similar process detailed in McCabe et al. (2021) to correct zero-point discrepancies between the two mosaics. In the summary, we describe work that is currently being done to complete the analysis of the two mosaics.

Chapter 2

OBSERVATIONS

The 552 observations used in this work were taken by the VIsible Multi-Object Spectrograph (VIMOS) on ESO's VLT (Nonino et al. 2009). VIMOS is a four-chip charge-coupled device (CCD) camera with a pixel scale of 0.205"/pixel. VIMOS operates within wavelengths of 360 to 1000 nm, with the U-band at $\lambda_0 \simeq 370$ nm. Each chip utilizes an array of 2048 $\times$ 4096 pixels of an EEV 44-82 backside illuminated CCD and has a field of view (FOV) of 4 $\times$ 7' $\times$ 8', adding up to a total FOV of 224 arcmin^2. The chips are separated by 2' gaps, so special care was taken when dithering so as to get uniform coverage of the field (Nonino et al. 2009). The observations were taken between August 2004 and October 2006. A summary of the observations from Nonino et al. (2009) can be found in the appendix.

Chapter 3

ANALYSIS

3.1 Software

In order to create these image mosaics, several specialized software packages were utilized. The data is stored in Flexible Image Transport System, or FITS, files. These files are structured for ease of use. Each FITS image has a primary header data unit (HDU) that includes a header and data array. The header contains keywords and values that are relevant to the image. Keywords define the number of pixels in each axis, the RA and DEC of the pointing, the world coordinate system (WCS) used, the date of the observation, etc. The data array contains the image data, with each element of the array corresponding to a pixel.

Several software packages can be used to analyze FITS files. SCAMP, SWARP, and SExtractor (Source-Extractor) were all used to create these image mosaics (Bertin et al. 2006, Bertin et al. 2002; Bertin 2010, Bertin & Arnouts 1996). SCAMP works by comparing input images with a reference catalog and making header files with astrometric corrections. Adverse effects such as poor atmospheric conditions or instrument defects can create errors in the astrometry. Astrometric corrections are necessary so that the positions of the stars and galaxies in the images are as accurate as possible. The SCAMP files used in this work were created by our INAF collaborators, who used data from OmegaCAM on the VLT as the reference catalog. SWARP is then used to re-sample and co-add the FITS images into an image mosaic. SWARP has the option to use SCAMP file parameters to override the internal FITS parameters.

SExtractor creates a catalog of objects from FITS images. Details about specific swarp configuration parameters used for these mosaics can be found in Section 3.3.

3.2 Flux Ratio Corrections

Ashcraft et al. (2018) found a ∼0.2 magnitude difference in the photometric zero point between the optimal resolution and optimal depth mosaics. This difference was attributed to variations in transparencies amongst different exposures and over different nights. Transparency refers to how clear the sky is and is affected by phenomena such as humidity, clouds, smoke, and dust. That is why making observations from high altitudes is ideal; it allows for better data by having less atmosphere to interfere with observations.

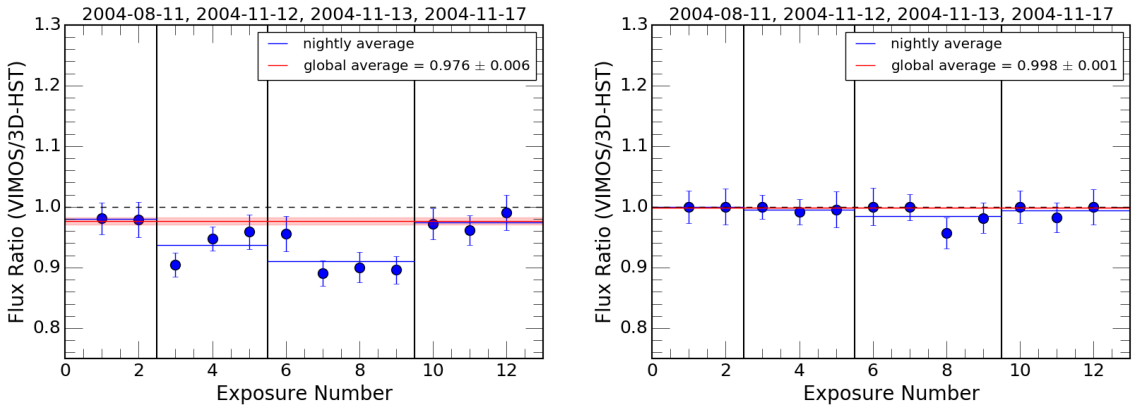
For this work, the flux in each image was corrected to a 3D-HST photometric catalog from Skelton et al. (2014) of the same data. These corrections were made by first matching objects between the 3D-HST catalog and VIMOS data. The VIMOS images were combined into their full, four-chip exposures in order to do the matching. This was necessary because significant astrometric differences were found between the single chip images and the reference catalog. Therefore, swarp was used to combine the images to incorporate the scamp corrections. An average of ∼50 objects were matched to the catalog for each exposure. An object was determined to be matched if its coordinates matched within 2/3 of a pixel between the image and reference catalog. A magnitude selection of $18 \leq u \leq 24.5$ AB was used as well. Once these objects were found, their magnitudes were converted to fluxes using the equation:

$$F = 10^{(M+48.6)/-2.5} \tag{3.1}$$

where -48.6 is a zero point used to convert AB magnitudes to real cgs units of $erg/s/cm^2/Hz$. The ratio of image flux to the catalog flux was then calculated, and the reciprocal of this value was used as the correction so that:

$$F_{Image} \times Correction = F_{Image} \times \frac{1}{F_{ratio}} = F_{Image} \times \frac{F_{Catalog}}{F_{Image}} = F_{Catalog} \tag{3.2}$$

The above scheme was applied to all 139 combined exposures. An example python code of the calculations can be found in the appendix. Figure 1 shows an example of the uncorrected and corrected flux ratios for the first four nights of observations. Prior to corrections, the average ratio across all nights was 97.6% ± 0.6%. After corrections, this was increased to 99.8% ± 0.1%, better matching the 3D-HST catalog. Figures 9 and 10 in the Appendix show flux ratios for all nights.



(a) Uncorrected (b) Corrected

Figure 1: Uncorrected and corrected flux ratio distributions for the first four nights of observations. Nights are separated by solid vertical lines. The horizontal, dashed black line represents flux equal to 3D-HST. The solid red line with confidence bands represents the global average for all 139 exposures.

In order to gauge the effect of the flux ratio corrections on the data, the signal-to-

Object	RA (J2000)	DEC (J2000)	SNR (1.5" Mosaic)	SNR (Nonino et al. 2009)
1	53.2747026	-27.9417335	73.84248191	71.23122587
2	53.1577418	-27.9935246	85.72581355	86.99744168
3	52.9929541	-27.9874944	42.5706638	37.68623351
4	53.1531596	-27.8121795	49.93633531	48.13396386
5	53.0476709	-27.8650923	73.83037777	88.549837
6	53.2180312	-27.761717	53.73164744	65.61006273
7	53.0569765	-27.7401067	25.55869916	31.87997239
8	53.1375462	-27.67143	45.99333434	35.88705155
9	52.9799182	-27.5952787	118.8996673	111.2428935
10	53.1599427	-27.599355	46.35789573	56.61610978
11	53.1021818	-27.8189918	28.45946775	29.57397623
12	53.2911014	-27.7965987	48.18649636	55.4778845

Table 1: This table compares the signal-to-noise ratio (SNR) values for twelve objects between the optimal depth (1.5") mosaic with the flux ratio corrections and the original mosaic from Nonino et al. (2009).

noise ratio, or SNR, was calculated for a handful of objects. The SNR is expressed by the following equation:

$$\frac{S}{N} = \frac{S_*}{\sqrt{S_* + n_{pix} \cdot (1 + \frac{n_{pix}}{n_{sky}}) \cdot (S_s + t \cdot d + R^2)}} \tag{3.3}$$

where S_* is the signal from the object, S_s is the signal from the sky, t is the exposure time, d is the dark current expressed in $e^-/pix/hour$, and R is the read-noise (in e^-). The signal from the sky S_s was calculated by taking the mean over n_{sky} background pixels. This explicitly includes the uncertainty in determining the sky background level. An example python code for these calculations can be found in the appendix. Table 1 shows the SNR values for twelve objects in both the 1.5" mosaic with the flux ratio corrections and the original mosaic from Nonino et al. (2009). The values favor neither mosaic. It is possible that the sample size is too small to determine whether or not the flux ratio corrections improved the SNR of the data.

3.3 Seeing Sorted Stacks

All 552 single chip exposures were sorted based on their seeing, or FWHM. Seeing is a measure of atmospheric turbulence. Turbulent air caused by wind, convection, or clouds can make stars appear to twinkle and blur, reducing the quality of data. FWHM values were calculated by the Italian LBT collaborators. A few images did not have FWHM measurements. Therefore, SExtractor was used to measure the FWHM of ∼25 unsaturated stars in each of those images.

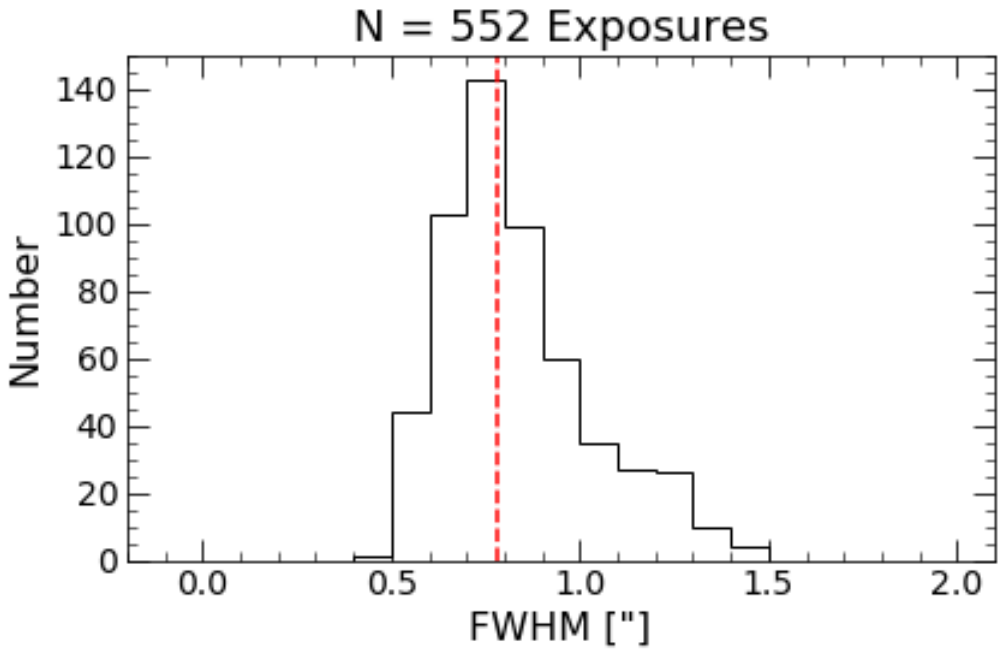


Figure 2: Histogram of the seeing for all 552 single chip images. The median seeing is ∼0.78". Exposures with FWHM $\leq$ 0.8" are included in the best resolution mosaic. All exposures are used in the best depth mosaic.

The FWHM of the image was then taken to be the median value of the sample of stars. The seeing values calculated this way were in agreement with those provided. Figure 2 shows a histogram of the seeing of the VIMOS data. Compared to the LBT

data used in Ashcraft et al. (2018) and McCabe et al. (2021), the VLT images have better seeing. The median seeing is found to be ∼0.78", which is lower than Ashcraft et al. (2018) and McCabe et al. (2021).

SWARP was then used to create two image mosaics. A 5σ clipping was used for outliers. For resampling, the LANCZOS3 function was used. The best resolution mosaic, cut off at 0.8", used 288 images, just over 50% of the data. The best depth mosaic utilized images with seeing under 1.5", corresponding to 100% of the data. Nonino et al. (2009) had already created a mosaic corresponding to the optimal depth image. In this work, this mosaic was recreated with the addition of making the zero point corrections as described in Section 3.2. Figure 3 shows exposure maps corresponding to the best resolution and best depth mosaic.

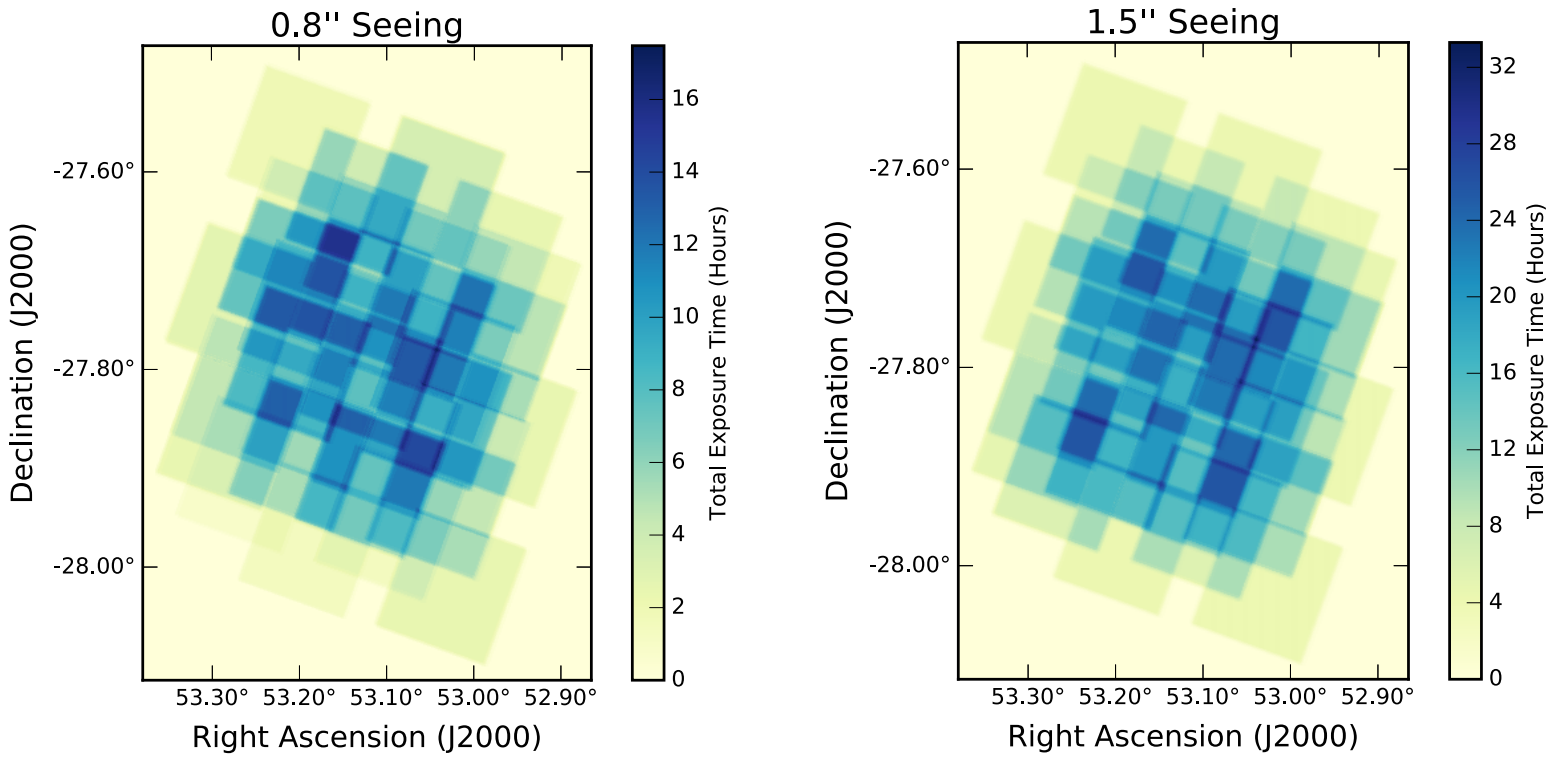


Figure 3: Exposure maps for the optimal resolution mosaic (left) and optimal depth mosaic (right). The color bar is a function of exposure time. Areas that are darker represent regions with more exposure time. The best resolution mosaic contains just over 50% of the images, while the best depth mosaic is made from all 552 single chip images.

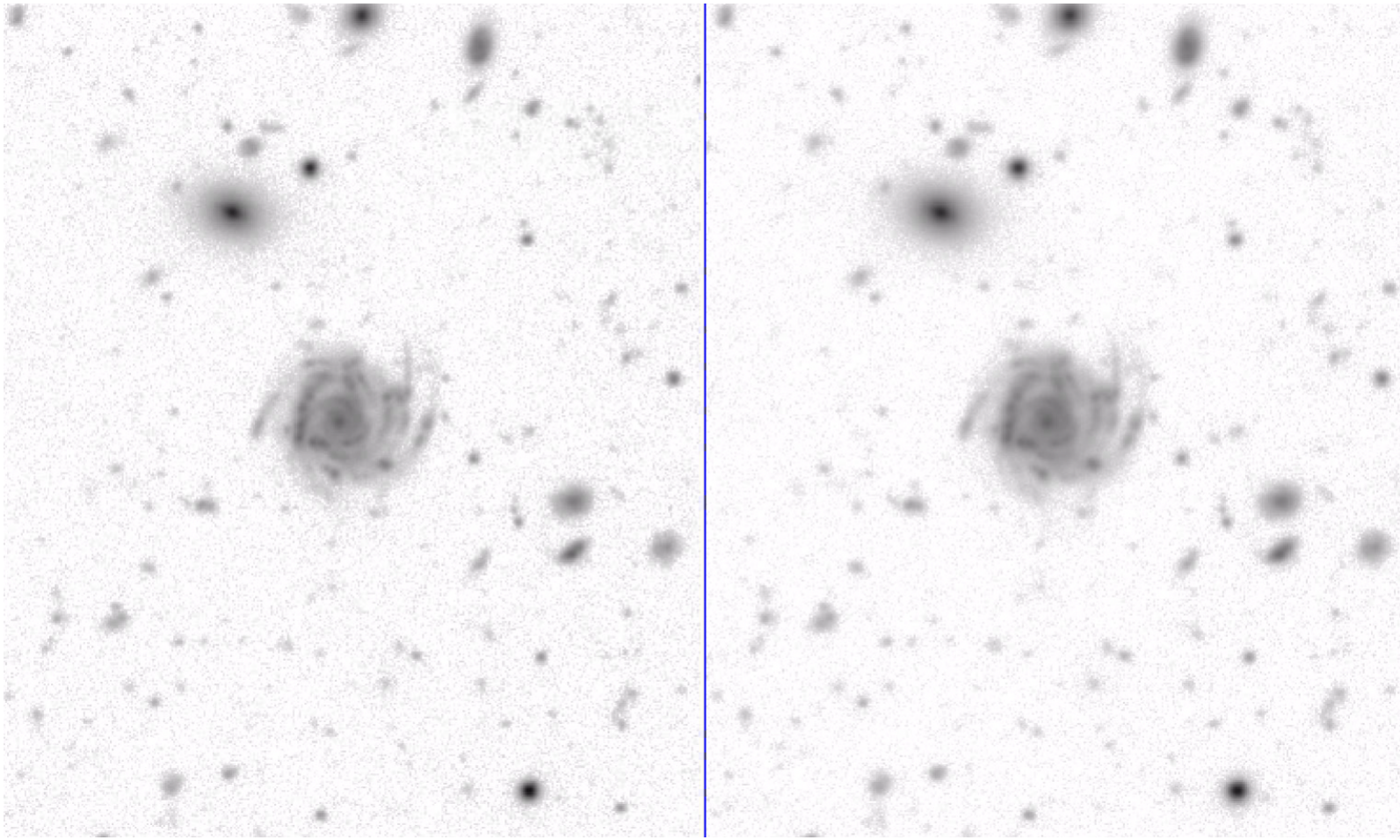


Figure 4: This image compares the optimal resolution mosaic (left) and optimal depth mosaic (right). While the difference is not as obvious as other works (due to the VIMOS data having good seeing to begin with), the right image shows more of the low surface brightness features of the galaxy.

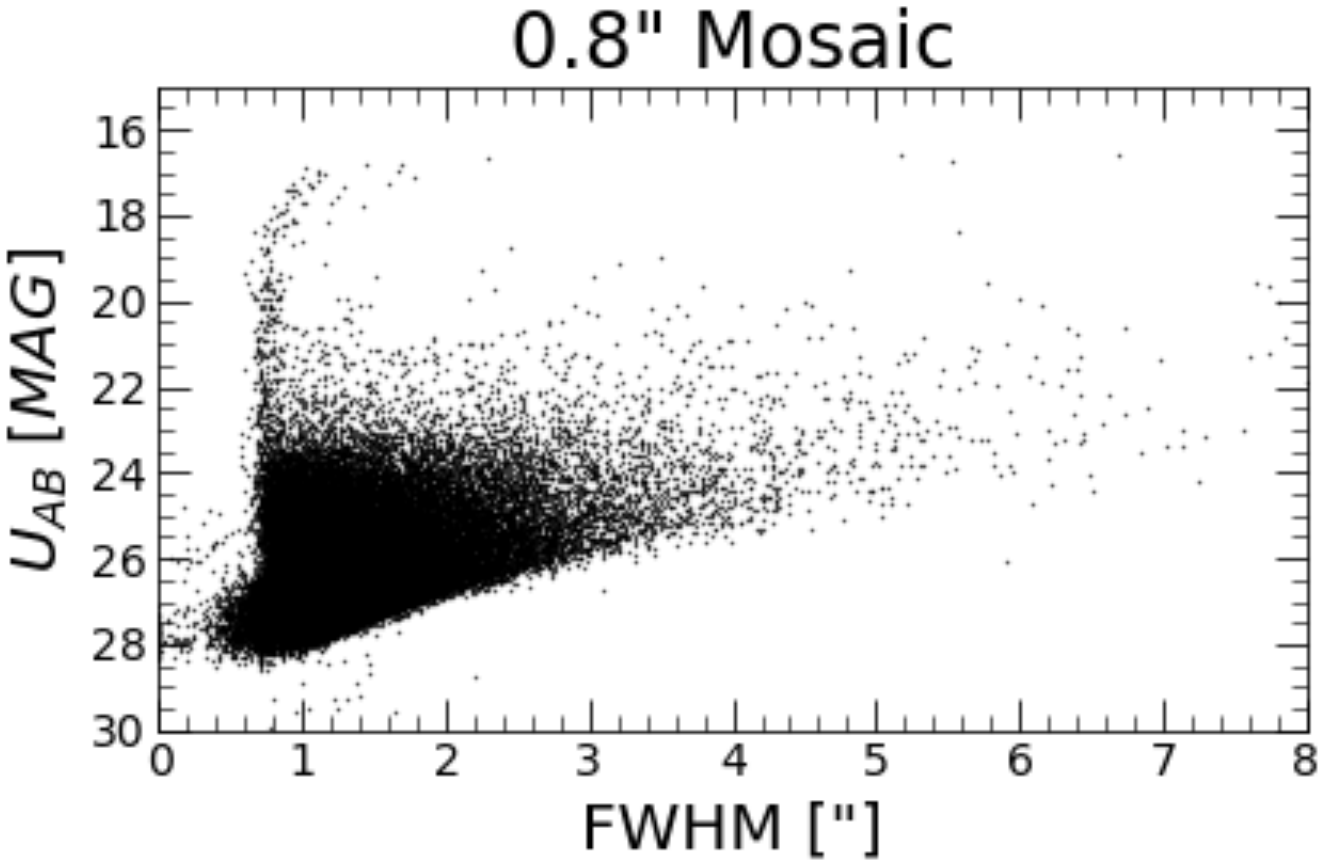


Figure 5: AB magnitude as a function of FWHM for the current best catalog from SExtractor for the best resolution mosaic. The unsaturated stars reside in the "elephant's trunk," or the strip centered around ∼0.8". The curl of that strip represents saturated stars. Everything to the right of ∼0.9" are galaxies.

Chapter 4

SUMMARY

We present the results of deep U-band imaging of the GOODS-South field from VIMOS on the VLT using data from Nonino et al. (2009). Two image mosaics were made using SWARP and by following the procedure detailed in Ashcraft et al. (2018). All 552 single chip images were sorted based on FWHM. The optimal resolution mosaic was then made from images with FWHM $\leq$ 0.8".The optimal depth mosaic was made from images with FWHM $\leq$ 1.5". Before creating the mosaics, however, 139 four-chip exposures were corrected to better match the flux from a 3D-HST catalog of the same data (Skelton et al. 2014).

Currently, work is being done in several areas to complete the analysis of the mosaics. The first is to quantify the change in the zero-point from the flux corrections.This is done by comparing the magnitude difference between the both the corrected and uncorrected mosaic and the 3D-HST catalog and identifying changes between the two. McCabe et al. (2021), who did a similar process to correct the zero-point for COSMOS data, was able to reduce the $\sim$0.2 magnitude discrepancy to 0.05 magnitude.

Work is also being done to create accurate SExtractor catalogs of the two mosaics. It is important to find the correct parameters in order to avoid phenomena such as improper deblending, where two objects might be considered one object or one object is inadvertently split into two. This can lead to inaccurate object counts. Once the catalogs have been optimized, the galaxies can be separated from stars by creating a diagram of magnitude vs. FWHM, as described in Windhorst et al. (2011). Figure 5 shows an example of this diagram using the most current SExtractor

catalog for the best resolution mosaic. Then, once the galaxies have been isolated, a histogram of the galaxy counts per square degree as a function of magnitude can be created. The depth of the mosaic can be determined from the turnover of the galaxy counts. An example of this plot can be seen in Figure 7, which shows the galaxy counts for the COSMOS field found by McCabe et al. (2021). From this plot, it can be determined that the best resolution mosaic reaches a depth of $\sim$26 m_{AB} and the best depth mosaic reaches $\sim$26.5 m_{AB} . Both Ashcraft et al. (2018) and McCabe et al. (2021) found that galaxy counts fall off more slowly for the best depth mosaic, which means it contains more faint objects than the best resolution mosaic.

Once these mosaics for GOODS-South are complete, three of the four UVCANDELS fields will have ground-based images complementary to HST data. LBT data for the EGS is currently being observed, making for the completion of this seeing sorted stacking analysis for the UVCANDELS program. In the coming years, these mosaics will complement JWST $1 - 5$ μm observations and help us better understand galactic evolution over the past 9-10 billion years.

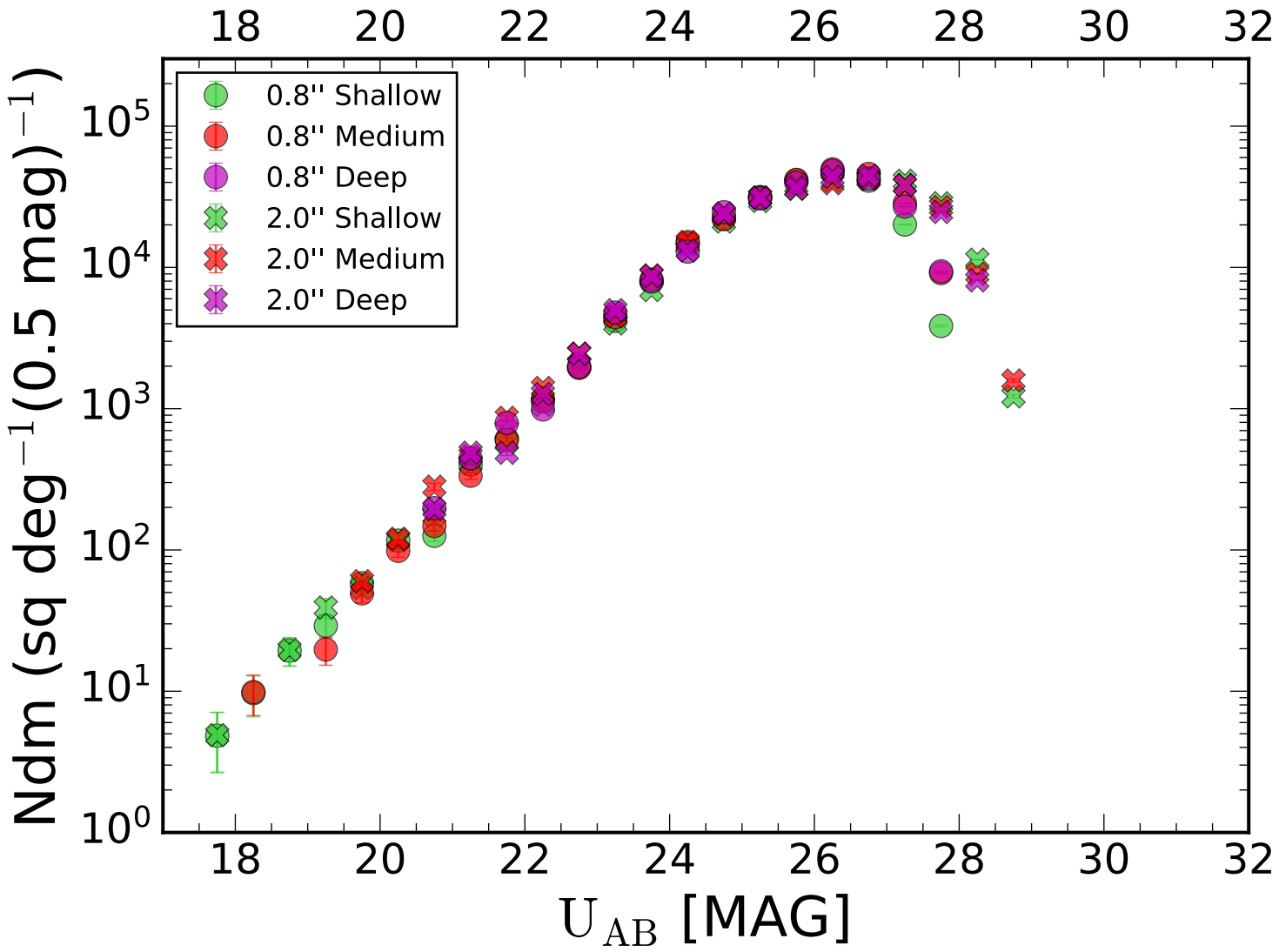


Figure 6: Galaxy count histogram of COSMOS data for the best resolution (0.8") mosaic and best depth (2.0") mosaic (McCabe et al. 2021). The number of galaxies per square degree as a function of magnitude follow a power law as magnitude increases. The point at which the counts turn over determine the depth of the image. Here, the best resolution mosaic reaches a depth of ∼26 m_{AB} and the best depth mosaic reaches ∼26.5 m_{AB} .

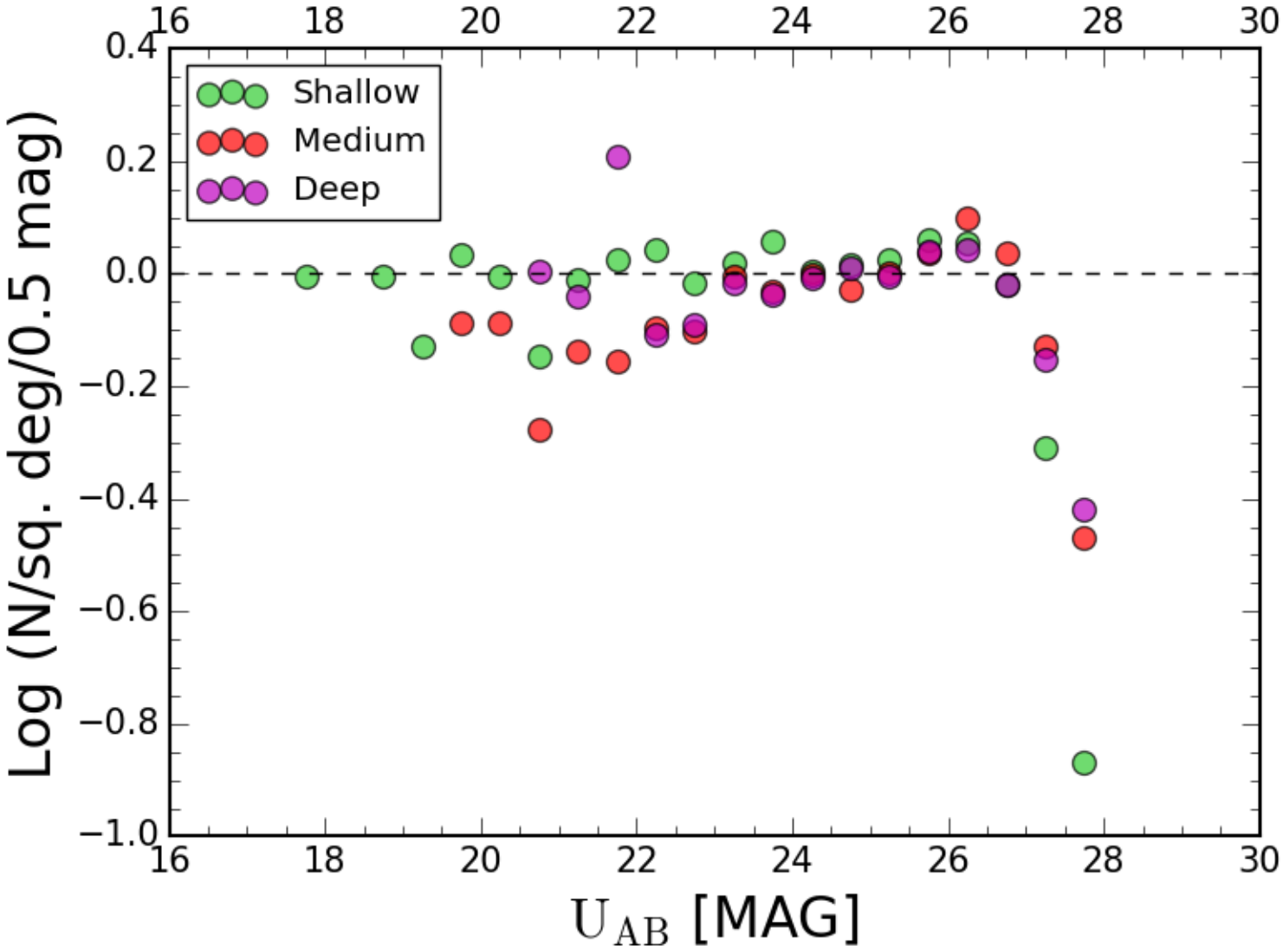


Figure 7: Logarithmic difference between the best resolution (0.8") galaxy counts and best depth (2.0") galaxy counts for COSMOS data (McCabe et al. 2021). This diagram highlights the discrepancies between the number counts more clearly than the previous plot.

Chapter 5

REFERENCES

Ashcraft, T. A., Windhorst, R. A., Jansen, R. A., Cohen, S. H., Grazian. A., Paris, D., Fontana, A., Giallongo E., Speziali, R., Testa, V., Boutsia, K., O'Connell, R. W., Rutkowski, M. J., Ryan, R. E., Scarlata, C., and Weiner, B. 2018, Publications of the Astronomical Society of the Pacific, 130, 64102

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

Bertin, E., Mellier, Y., Radovich, M., et al. 2002, Astronomical Data Analysis Software and Systems XI, 281, 228

Bertin, E. 2006, Astronomical Data Analysis Software and Systems XV, 351, 112

Bertin, E. 2010, Astrophysics Source Code Library, 1010.068

Davis, M., Guhathakurta, P., Konidaris, N. P., et al. 2007, ApJL, 660, L1

Giavalisco, M., Ferguson, H. C., Koekemoer, A. M., et al. 2004, ApJ, 600, L93

Grogin, N. A., Kocevski, D. D., Faber, S. M., et al. 2011, ApJS, 197, 35

Koekemoer, A. M., Aussel, H., Calzetti, D., et al. 2007, ApJS, 172, 196

Koekemoer, A. M., Faber, S. M., Ferguson, H. C., et al. 2011, ApJS, 197, 36

McCabe, T., Otteson, L. H., Ashcraft, T. A., Windhorst, R. A., Jansen, R. A., Cohen, S. H., et al. 2021. In progress.

Nonino, M., Dickinson, M., Rosati, P., et al. 2009, ApJS, 183, 244

Scoville, N., Aussel, H. et al. 2007, ApJS, 172, 1

Skelton, R. E., Whitaker, K. E., Momcheva, I. G., et al. 2014, ApJS, 214, 24

Windhorst, R. A., Cohen, S. H., Hathi, N. P., et al. 2011, ApJS, 193, 27

Appendix

Table 1
Log of Observations

Date	Field ID	Total Exposure Time (s)	DIMM Seeing Range (arcsec)	Air-Mass Range
2004 Aug 11	01	3000	0.81–0.87	1.17–1.29
2004 Nov 12	01	3000	0.60–0.70	1.06–1.12
2004 Nov 13	01	4000	0.59–0.74	1.08–1.25
2004 Nov 17	01	3000	0.58–0.91	1.12–1.21
2005 Sep 6	02	6000	0.98–1.44	1.00–1.06
2005 Sep 7	02	4000	0.68–0.84	1.07–1.21
2005 Sep 8 (*)	02	3000	0.99–1.03	1.01–1.03
2005 Oct 8	02	2000	0.78	1.07–1.11
2005 Oct 10	06	2000	0.58–0.74	1.07–1.10
2005 Oct 11	06	7000	0.37–0.82	1.00–1.08
2005 Oct 12 (*)	03	3000	0.69–0.80	1.01–1.03
2005 Oct 12 (*)	05	3000	0.64–0.74	1.05–1.11
2005 Oct 12 (*)	06	3000	0.80–1.10	1.00–1.01
2005 Oct 27	02	3000	1.45–1.50	1.00–1.01
2005 Oct 27	03	6000	1.17–1.42	1.02–1.18
2005 Oct 29 (*)	06	3000	0.90–1.21	1.02–1.06
2005 Oct 29 (*)	07	4000	1.21–1.32	1.09–1.24
2005 Oct 30	03	3000	0.90–1.19	1.12–1.21
2005 Oct 31	03	5000	0.61–0.90	1.07–1.15
2005 Nov 1	03	1000	0.72	1.20–1.26
2005 Dec 2	03	3000	0.50–0.61	1.00–1.01
2005 Dec 2	05	6000	0.46–0.78	1.02–1.16
2005 Dec 4 (*)	05	6000	0.64–1.04	1.00–1.02
2005 Dec 4 (*)	07	6000	0.78–0.99	1.04–1.23
2006 Jan 26	07	3000	0.55–0.64	1.25–1.41
2006 Aug 19 (*)	01	3000	1.22–1.47	1.03–1.06
2006 Sep 22	07	6000	0.79–1.30	1.00–1.04
2006 Sep 24	07	3000	0.95–1.12	1.09–1.16
2006 Sep 24	04	2000	0.72–0.78	1.02–1.04
2006 Sep 25	04	2000	0.39–0.48	1.02–1.04
2006 Oct 13	08	3000	0.85–1.14	1.05–1.12
2006 Oct 16	04	6000	0.51–0.72	1.01–1.13
2006 Oct 16	08	9000	0.54–0.75	1.00–1.22
2006 Oct 17	04	3000	0.79–0.83	1.07–1.13
2006 Oct 18	08	3000	0.88	1.09–1.17
2006 Oct 21	04	3000	0.45–0.52	1.02–1.06
2006 Oct 27	04	3000	0.50–0.66	1.14–1.24

Figure 8: This table shows a log of the observing conditions of the GOODS-South field from Nonino et al. (2009). The nights with an asterisk were judged to be photometric.

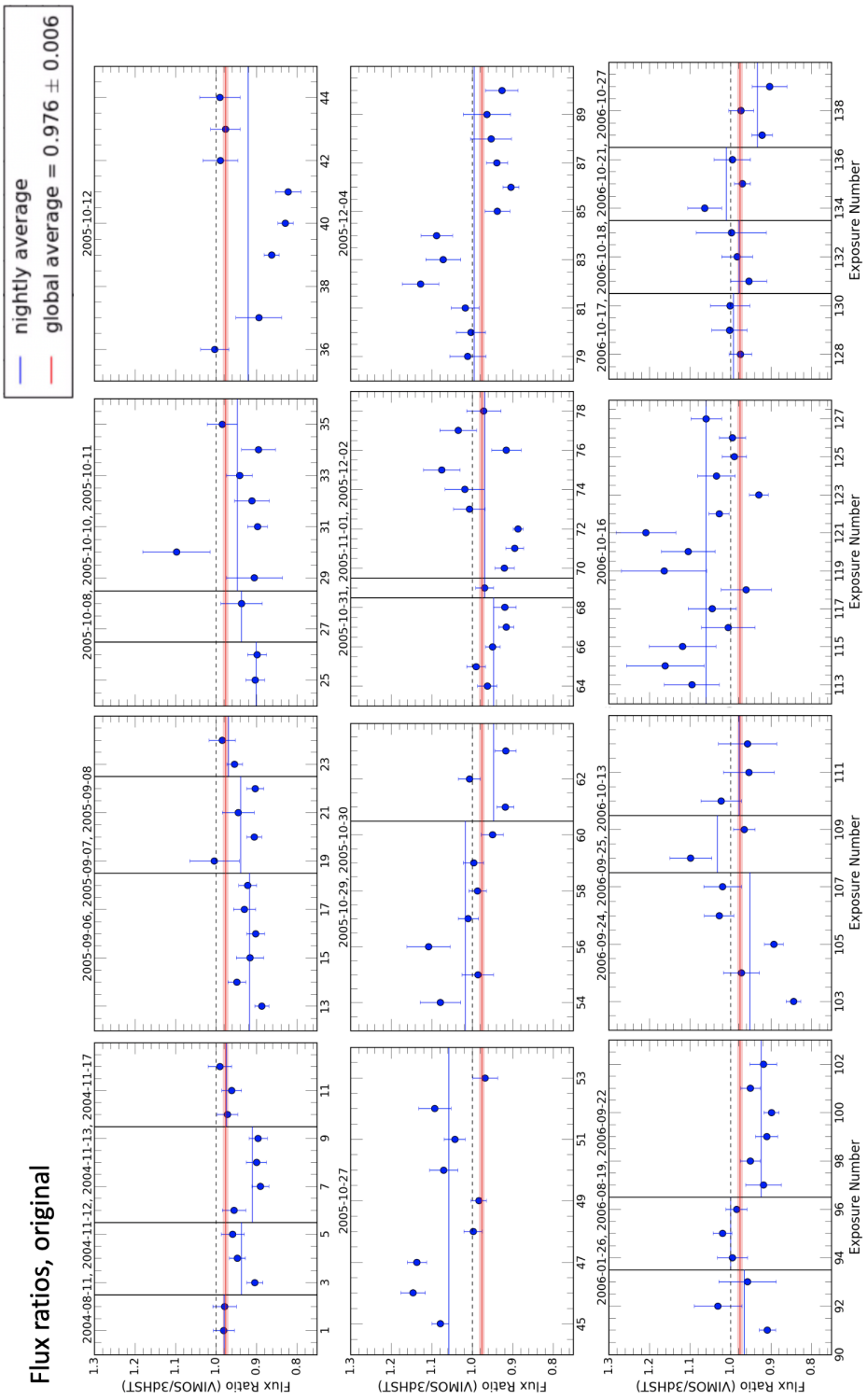


Figure 9: Uncorrected flux ratios for all nights. Each point represents a full, four-chip combined exposure. Nights are separated by black vertical lines. The global average (red) is 0.976 ± 0.006. Nightly averages are represented by the blue horizontal lines.

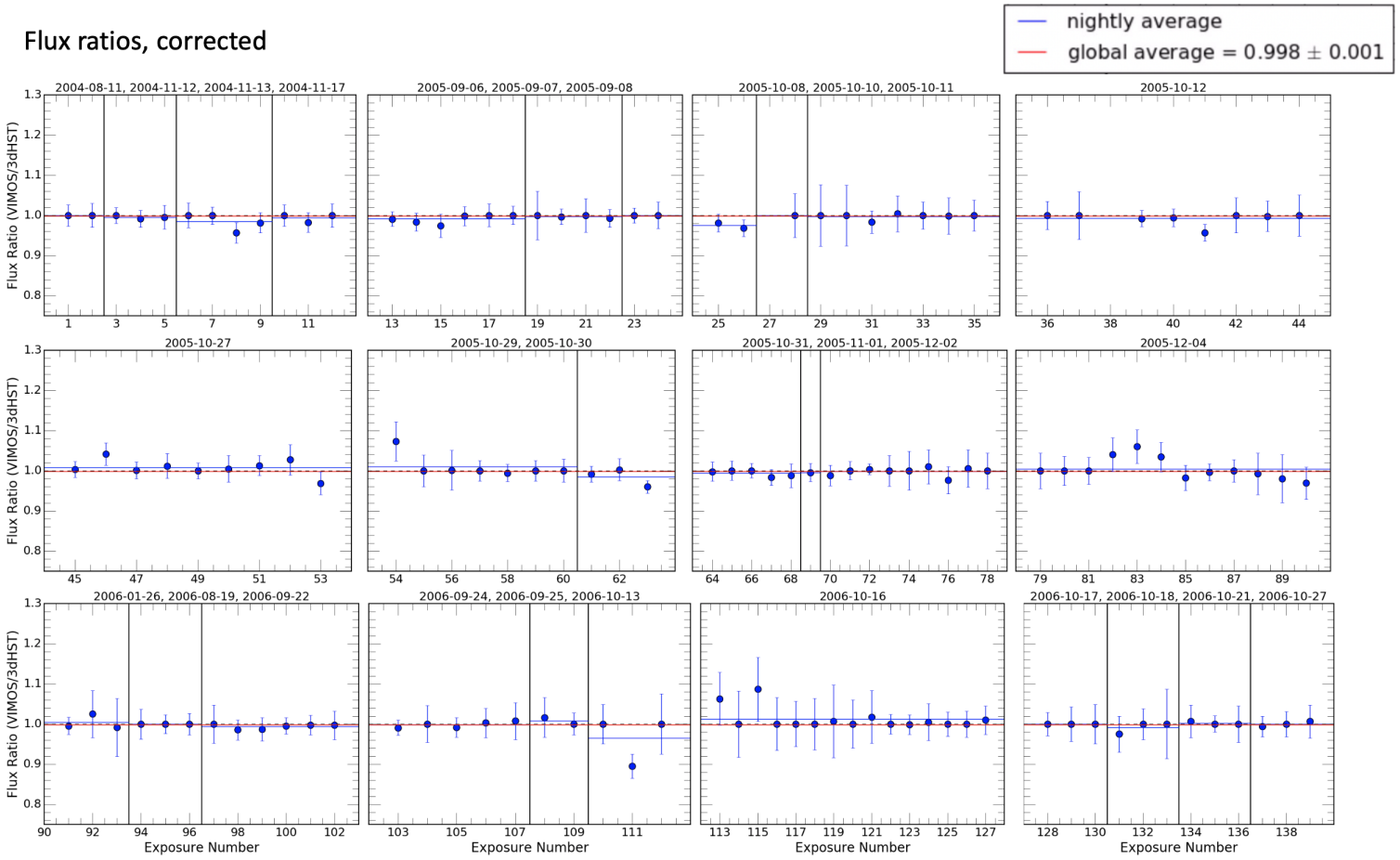


Figure 10: Corrected flux ratios for all nights. Each point represents a full, four-chip combined exposure. Nights are separated by black vertical lines. The global average (red) is 0.998 ± 0.001. Nightly averages are represented by the blue horizontal lines.

Flux Ratio Calculations

[1]:
```
import numpy as np
import pandas as pd
import sys
import math
import matplotlib
import matplotlib.pyplot as plt
from matplotlib.ticker import MultipleLocator, FormatStrFormatter,␣
↪FuncFormatter, MaxNLocator
from matplotlib.ticker import MultipleLocator, FormatStrFormatter, MaxNLocator,␣
↪FixedLocator
import astropy
from astropy.stats import sigma_clipped_stats
from astropy.io import fits
from astropy.table import Table
```

[2]:
```
hst_cat = './Catalogs/3DHST/Catalog/goodss_3dhst.v4.1.cat.FITS'

with fits.open(hst_cat) as hdu:
        head_3d = hdu[1].header
        data_3d = hdu[1].data
```

[3]:
```
def get_overlap(ra_1,dec_1,mag_1,ra_2,dec_2,mag_2):
    """This function returns the overlap of the ra, dec, and magnitude of␣
↪objects in two catalogs. """
    ra_overlap_1 = np.asarray([])
    dec_overlap_1 = np.asarray([])
    mag_overlap_1 = np.asarray([])
    ra_overlap_2 = np.asarray([])
    dec_overlap_2 = np.asarray([])
    mag_overlap_2 = np.asarray([])  #25.5
    for i in range(len(ra_1)):
        for j in range(len(ra_2)):
            if np.abs(ra_1[i] - ra_2[j]) <= 0.00011388888/3 and np.abs(dec_1[i]␣
↪- dec_2[j]) <= 0.00011388888/3: ## 2 pixels  (2*0.205 / 3600)
                if mag_1[i] > 24.5 or mag_2[j] > 24.5 or mag_1[i] < 18 or␣
↪mag_2[j] < 18:
                    pass
```

```
                else:
                    ra_overlap_1 = np.append(ra_overlap_1,ra_1[i])
                    dec_overlap_1 = np.append(dec_overlap_1,dec_1[i])
                    mag_overlap_1 = np.append(mag_overlap_1,mag_1[i])
                    ra_overlap_2 = np.append(ra_overlap_2,ra_2[j])
                    dec_overlap_2 = np.append(dec_overlap_2,dec_2[j])
                    mag_overlap_2 = np.append(mag_overlap_2,mag_2[j])

    return ra_overlap_1, dec_overlap_1, mag_overlap_1, ra_overlap_2,␣
↪dec_overlap_2, mag_overlap_2
```

[4]:
```
def get_flux_ratio(catalog, start, end):
    """
        This function takes a reference catalog, the index of desired images,␣
↪and returns the coordinates

    and magnitudes of objects in both the reference catalog and the image␣
↪catalog, the sample size of matched

    objects, and the sigma-clipped flux ratio of matched objects and their␣
↪standard deviation. """


    filt_seeing = []
    airmass = np.asarray([])
    flux_ratio = np.asarray([])
    flux_ratio_c = np.asarray([])
    stdev = np.asarray([])
    fwhm = np.loadtxt('./vimos_fwhm.txt')

    cat_mag = np.asarray([])
    image_mag = np.asarray([])
    image_ra = np.asarray([])
    image_dec = np.asarray([])
    cat_ra = np.asarray([])
    cat_dec = np.asarray([])
    ss = np.asarray([])

    #Getting info from catalogs
    with fits.open(catalog) as hdu:
        ra_3d = hdu[1].data['ra']
        dec_3d = hdu[1].data['dec']
        mag_3d = 25 - 2.5*np.log10(hdu[1].data['f_u']) #conversion to AB

    #Getting rid of zero values in catalog magnitude array
    ra_3d = ra_3d[mag_3d > 0]
    dec_3d = dec_3d[mag_3d >0]
```

```
    mag_3d = mag_3d[mag_3d > 0]

    for i in range(start, end): #140
            print(i)
            filt_seeing.append(fwhm[i])

            #Getting info from fits catalogs
            image = fits.open('./Mosaics/Corrected_mosaic/0.8_corrected_mosaic.
↪cat')
            #image = fits.open('./test/vimos_' + str(i) +'_combined_test.cat')
            mag_file = image[1].data['MAG_AUTO']
            ra_file = (image[1].data['ALPHA_J2000'])# + 8/3600
            dec_file = (image[1].data['DELTA_J2000']) #+ 4/3600
            image.close()

            #Getting matched objects
            ra_im, dec_im, mag_im, ra_cat, dec_cat, mag_cat =␣
↪get_overlap(ra_file, dec_file, mag_file, ra_3d, dec_3d, mag_3d)
            print("Sample size: "+str(len(ra_im)))
            ss = np.append(ss, len(ra_im))

            #Appending pertinent info to parent arrays
            image_ra = np.append(image_ra, ra_im)
            image_dec = np.append(image_dec, dec_im)
            cat_ra = np.append(cat_ra, ra_cat)
            cat_dec = np.append(cat_dec, dec_cat)
            cat_mag = np.append(cat_mag, mag_cat)
            image_mag = np.append(image_mag, mag_im)

            #Getting fluxes, flux_ratio (flux ratios), stats, and appending to␣
↪parent arrays
            flux_im = np.array([(10**(((j) + 48.6)/ -2.5)) for j in mag_im])
            flux_cat = np.array([(10**(((j) + 48.6)/ -2.5)) for j in mag_cat])
            flux_rat = flux_im/flux_cat
            clipped_mean, clipped_median, clipped_stdev =␣
↪sigma_clipped_stats(flux_rat)
            flux_ratio = np.append(flux_ratio, clipped_median)
            stdev = np.append(stdev, clipped_stdev)


    return image_mag, cat_mag, image_ra, image_dec, cat_ra, cat_dec,␣
↪flux_ratio, ss, stdev
```

[ ]:
```
start = 1
end = 140
```

```
a, b, c, d, e, f, g, h, i = get_transparency(hst_cat, start, end)
```

[ ]:
```
flux_ratio = g

correction_values = [1/i for i in flux_ratio]
for i in range(len(correction_values)):
    if np.isinf(correction_values[i]) == True:
        correction_values[i] = 0


plt.savetxt('./flux_ratios.txt', g)
plt.savetxt('./correction_values.txt', correction_values)
```

Signal-to-noise Ratio Calculations

[ ]:
```
from astropy.io import fits
import numpy as np
import imexam
import os
import astropy
from astropy.io import fits
import astropy.wcs as wcs
from astropy.coordinates import Angle, SkyCoord
import regions
from astropy.stats import sigma_clipped_stats
import matplotlib.pyplot as plt
```

[ ]:
```
#Array containing exposure time info

time = np.loadtxt("./Mosaics/Corrected_mosaic/seeing_1.5_data_time.txt")
```

[ ]:
```
#Getting arrays of coordinates of all pixels
def ra_dec_array(image):
    hdu = fits.open(image)
    print('NAXIS1: '+str(hdu[0].header['NAXIS1']))
    print('NAXIS2: '+str(hdu[0].header['NAXIS2']))
    NAXIS1 = hdu[0].header['NAXIS1']
    NAXIS2 = hdu[0].header['NAXIS2']
    x = np.arange(NAXIS1)
    y = np.arange(NAXIS2)
    X, Y = np.meshgrid(x, y)
    w = wcs.WCS(hdu[0].header)
    ra, dec = w.all_pix2world(X, Y, 0)
    return ra, dec, w

ra_mosaic, dec_mosaic, wcs = ra_dec_array('./Mosaics/Corrected_mosaic/1.
↪5_corrected_lanczos3.fits')
```

[1]:
```
def SNR(ra_mosaic, dec_mosaic, wcs, mosaic, region, time_data):
    """
    This function takes the pixel coordinates, WCS, mosaic file name, region␣
↪file name, and time_data array
```

```
    and returns the SNR, coordinates of the source, and the corresponding␣
↪exposure time.

    ARGUMENTS:
    ----------
    ra_mosaic: array
               right ascension coordinates of pixels
    dec_mosaic: array
                declination coordinates of pixels

    wcs: WCS of mosaic

    mosaic: string
            file name of mosaic

    region: string
            file name of region file

    time_data: array
               array of exposure time for each pixel


    OUTPUTS
    -------
    snr: float
         signal-to-noise ratio of the source

    ra_star: float
        right ascencsion of source (from region file)

    dec_star: float
        declination of source (from region file)

    exp_time: float
        exposure time of source in seconds


    """

    #FITS info
    hdulist = fits.open(mosaic)
    data = hdulist[0].data
    header = hdulist[0].header

    skycoord = SkyCoord(ra_mosaic, dec_mosaic, unit='deg', frame='fk5')
    #skycoord = skycoord.transform_to('fk5')
    sky_region = regions.read_ds9(region)
```

```
    #Getting pixel values within the regions defining the source and annulus
    ring_small =  np.where(sky_region[0].contains(skycoord, wcs) == True)
    ring_mid =  np.where(sky_region[1].contains(skycoord, wcs) == True)
    ring_big = np.where(sky_region[2].contains(skycoord, wcs) == True)
    diff = np.array([np.array(np.setdiff1d(ring_big[i], ring_mid[i])) for i in␣
↪range(len(ring_mid))])
    diff = np.concatenate((diff[0], diff[1]))

    source = np.nansum(data[ring_small])
    sky = np.nanmean(data[diff])
    exp_time = np.mean(time[ring_small])


    def region_info(reg):
        #Function that pulls the relevant info from the region files
        file = np.loadtxt(reg, delimiter='\n', dtype = str)
        one = file[2].split(",")
        two = file[3].split(",")
        three = file[4].split(",")
        r_1 = float(one[2][:-3]) #small radius
        r_2 = float(two[2][:-3]) #medium radius
        r_3 = float(three[2][:-3]) #large radius

        ra = float(one[0][7:])
        dec = float(one[1])

        return r_1, r_2, r_3, ra, dec

    rad1, rad2, rad3, ra_star, dec_star = region_info(region)

    def area_term(r1, r2, r3, ps = 0.205):
        npix = (np.pi*(r1/ps)**2)
        nsky = (np.pi*(r3/ps)**2 - np.pi*(r2/ps)**2)
        return npix*(1+(npix/nsky))

    def SNR_calc(star, sky, G, n, t, d, a):
        """
        star: float
              signal from source

        sky: float
              signal from sky

        G: float
           instrument gain in e/adu
```

```
        n: float
            read noise of instrument in e

        t: float
            exposure time of source

        d: integer
            dark current in e/hour

        a: float
            areal term in SNR equation


        """
        return ((star-sky)*G)/np.sqrt(((star-sky)*G)+a*((sky*G)+t*(d/
↪3600)+n**2))

    snr = SNR_calc(source, sky, 1.82, 4.45, exp_time, 6, area_term(rad1, rad2,␣
↪rad3))

    return snr, ra_star, dec_star, exp_time
```

[ ]: