

# Proyecto Instancia Delfín

Renato Sánchez Loeza

8 de julio de 2024

## INTRODUCCIÓN

Desde los inicios de la astronomía, nuestra visión del firmamento ha estado limitada por la capacidad tecnológica para observarlo. Pasando por etapas tempranas donde el ojo humano se apoyaba con los primeros telescopios ópticos para la definición de objetos celestes, tal como el holandés Hans Lippershey, quien construyó uno de los primeros telescopios en 1608, y Galileo Galilei, quien mejoró este modelo en 1610 [Álv15].

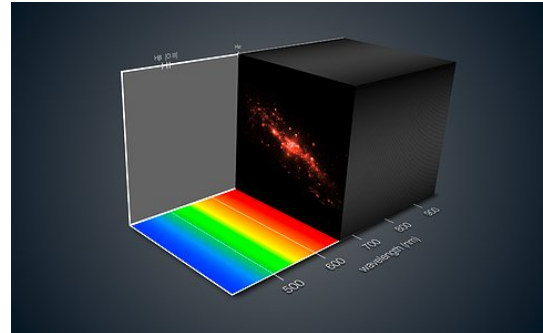
Posteriormente, la fotografía mejoró la precisión de las mediciones. El primero en hacerlo fue John William Draper, quien en 1840 utilizó un proceso de daguerrotipo<sup>1</sup> para capturar una imagen de la Luna. Cinco años más tarde, Foucault y Fizeau fotografiaron el Sol desde París [Ols21]. Pero los obstáculos se encuentran en la dificultad de compartir la información obtenida con base en este método.

Más recientemente, los CCD revolucionaron las observaciones astronómicas con una captura de entre el 90 % y el 95 % de la luz que llega a ellos, además de permitir la distribución de las fotografías obtenidas [Obs]. A estos avances se incorpora el análisis del espectro electromagnético (ondas de radio, infrarrojo, ultravioleta, rayos X y Gamma) y la espectroscopia, siendo la segunda la encargada de extraer datos sobre la distribución de la energía en una región del plano con el método de la rendija, donde por esta se permite entrar la luz del segmento a estudiar para posteriormente estudiar su espectrograma.

En la búsqueda de métodos que combinen lo mejor de la fotometría y la espectroscopia, se han propuesto diversas técnicas. Entre ellas, destacan el barrido con rendijas, el interferómetro de Fabry-Pérot (Etalon) y la espectroscopia multiobjetos (MOS). El barrido con rendijas implica observar la luz de una fuente a través de una rendija estrecha. Esto permite analizar su espectro y obtener información sobre composición química, velocidad y temperatura, mientras que el Fabry-Pérot es una técnica que utiliza múltiples reflexiones entre dos superficies altamente reflectantes y cercanas entre sí [Hyp]. Por otro lado, la MOS emplea máscaras con múltiples aperturas para capturar espectros de varios objetos simultáneamente [OCo], a menudo utilizando una matriz de fibras para capturar parte del campo de visión. Estas fibras se encuentran en el plano focal del instrumento de imagen. Luego, las fibras se redirigen y se alinean en la ranura de entrada del espectrógrafo, dispersando la luz en un detector. Se emplea en estudios de objetos débiles, como cúmulos de galaxias. Ejemplo de estos sistemas son el *Low Resolution Imaging Spectrometer* (LRIS) en el Keck I y el *Visible MultiObject Spectrograph* (VIMOS)<sup>2</sup> en el VLT [Oke+95].

<sup>1</sup>Técnica fotográfica primitiva mediante la cual las imágenes captadas con la cámara oscura se fijan sobre una chapa metálica convenientemente preparada.

<sup>2</sup>VIMOS operates in three different modes: Imaging (IMG), Multi-Object Spectroscopy (MOS), and with Integral Field Unit (IFU). [web08]



**Fig. 1.** Visualización de un cubo de datos resultante de una IFS. Crédito: ESO/MUSE consortium/R. Bacon/L. Calçada

## ESPECTROSCOPIA INTEGRAL DE CAMPO

Sin embargo, el sistema que hoy en día se usa con más frecuencia es la espectroscopia integral de campo o IFS (Integral Field Spectrograph), la cual nos permite obtener un espectrograma de una región de 2 dimensiones del espacio. Un IFS genera un "cubo de datos" (Fig. 1), que contiene el campo de visión en 2 dimensiones y una dimensión espectral, formado por un bloque de imágenes de la misma región, cada una en una longitud de onda diferente [Dep]. Los astrónomos pueden medir, por ejemplo, el movimiento del gas en una galaxia lejana o las distancias a las diferentes galaxias detectadas en un campo de visión, utilizando la riqueza de la información de los espectrógrafos de campo integral.

En este proceso se encuentran las Unidades Integrales de Campo o IFU (Integral Field Units), los cuales son ampliamente utilizados para estudiar objetos de gran extensión, como cúmulos, galaxias o nebulosas, en una sola toma. Durante este proceso, la señal de cada una de las celdas se procesa en un espectrógrafo, el cual genera un espectro para cada uno de los píxeles, generando así un *spaxel*, y al final del proceso obtenemos un cubo de datos o *datacube*.

### Métodos de IFS

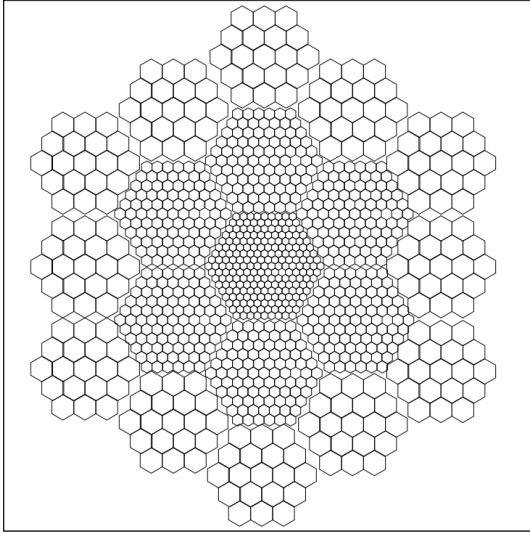
#### Arreglo de lentes o Lenslet arrays

Utiliza imágenes de pupila para distribuir espectros cortos. La magnificación y espacio entre espectros es importante para evitar mezcla de luz de diferentes longitudes de onda (Fig. 2) [All06].

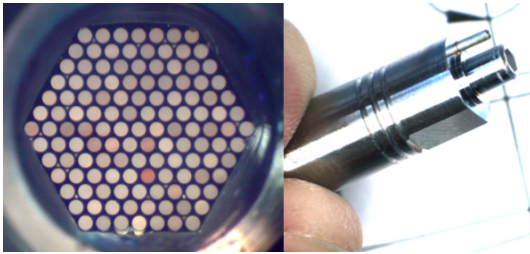
#### Arreglo de fibras o Fibre arrays

Esta técnica consiste en acoplar una serie de fibras y un arreglo de microlentes<sup>3</sup> que redirigen la luz a las rendijas para la generación de espectrogramas. Cada fibra recoge la luz de una ubicación específica en el campo de visión (Fig. 3). Además, requiere menos espacio muerto y, por lo tanto,

<sup>3</sup>Los arreglos de microlentes son necesarios para las IFU acopladas a fibras. Adaptan el cono de luz entrante a la fibra, al mismo tiempo que optimizan el factor de llenado del conjunto de entrada. [SRR06]



**Fig. 2.** Ejemplo de arreglo de lentes hexagonales con 3 tamaños de spaxels. Crédito: [SRR06]



**Fig. 3.** A la izquierda se muestra la cara del primer prototipo de IFU de 127 fibras. A la derecha se muestra la carcasa de la férula que sostiene la IFU y permite enchufarla a la placa SDSS. [14]

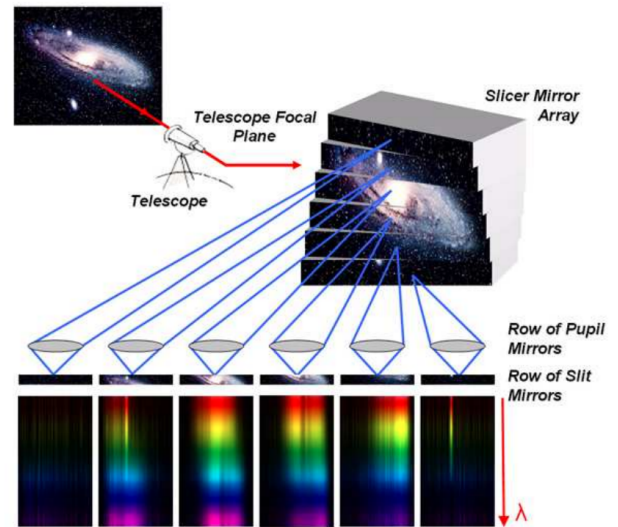
tiene el potencial de aumentar la densidad de información [All06].

#### Cortador o Slicer

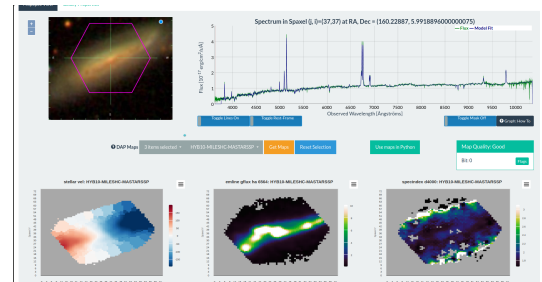
Un cortador de imágenes se compone generalmente de un conjunto de espejos cortadores ubicados en el plano de la imagen del telescopio y asociados a una fila de espejos pupilares y una fila de espejos de rendija [VP05]. Esta técnica busca dividir la imagen en tiras para después dispersarlas y formar un espectro (Fig. 4).

## VISUALIZACIÓN Y ANÁLISIS

Si bien se ha avanzado de forma significativa para mejorar nuestros métodos para observar las regiones espaciales, ahora es el momento de analizar dichos datos para obtener un aporte al conocimiento mucho más sofisticado. Hoy en día podremos encontrar distintos software de procesamiento de datos de este tipo para modelos y usos muy específicos. Sin embargo, no podemos decir que existe una herramienta universal que nos dé las herramientas suficientes para manipular datos de cualquier procedencia, con una interfaz intuitiva y gran variedad de opciones para procesar los datos.



**Fig. 4.** Visualización de un IFU Slicer. Crédito: [VP05]



**Fig. 5.** Interfaz web de Marvin.

## Soluciones existentes

### Marvin

Actualmente se encuentra Marvin [Che+19], un conjunto de herramientas que incluye un paquete de Python, una API y una aplicación web, diseñado para trabajar con los datos del proyecto MaNGA de SDSS-IV1. Este software está equipado con datos generales de las galaxias como las coordenadas celestes, la fecha de observación y una imagen de la galaxia observada, así como la visualización espectral en un spaxel específico (espectro observado y modelo de ajuste) con la opción de marcar los componentes químicos que refleja la espectroscopia. Incluye mapas de propiedades de la galaxia en cuestión, con datos como la velocidad estelar y el flujo de líneas de emisión (Fig. 5).

### Starlink

También contamos con Starlink [Ber+22], un proyecto de código abierto para el procesamiento de datos astronómicos. Este proyecto terminó en el 2005, el software sigue en desarrollo por la *East Asian Observatory*. Una desventaja con este software es el hecho de que las tecnologías que ocupa son muy desactualizadas, dentro de estas tecnologías se encuentran Fortran/C. Toda la referencia e instrucciones para la compilación del software se encuentran en el documento Starlink SSN/78 y su repositorio de GitHub. Su interfaz es igual que todo el sistema está implementada en tecnologías obsoletas y difíciles de mantener hoy en día (Fig. 6).

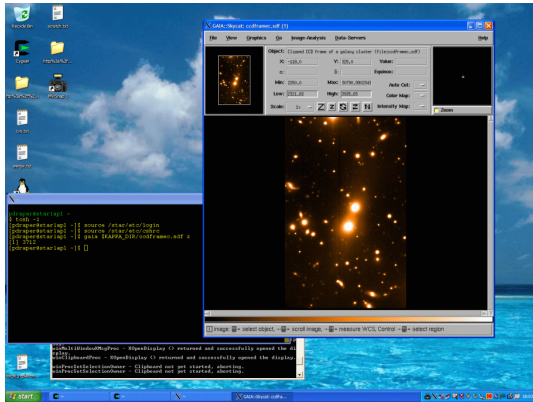


Fig. 6. Interfaz del software Starlink

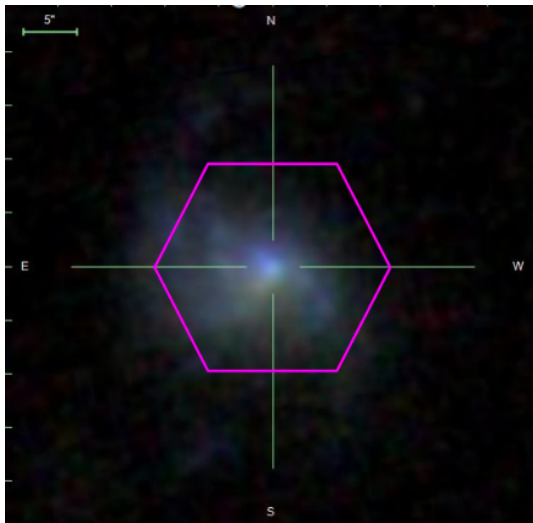


Fig. 7. Galaxia 7495-6102 - MaNGA - <https://ma-grathea.sdss.org/marvin/galaxy/7495-6102/>

## METODOLOGIA

Una vez claro el marco de referencia en el que nos encontramos se procede a iniciar el proceso de análisis de datos y exploración de tecnologías a utilizar. Para ello nos dirigiremos en un principio a la página web de **Marvin** con el fin de hallar una galaxia que nos resulte interesante para comenzar a analizarla. Por ahora tomaremos la galaxia **manga 7495-6102** y descargaremos su cubo de datos **lineal**<sup>4</sup> de la siguiente manera:

1. En el inicio de la página click en el botón **Go to SkyServer**
2. En la barra lateral derecha click al enlace **Explore**
3. Al final de la página click en **MaNGA observation(s)**
4. Click en **LIN Data Cube**

A continuación empezará la descarga de nuestro cubo de datos. Este archivo que se descargará será del tipo **.fits.gz** lo que indica que nuestro archivo está comprimido pero no será necesario descomprimirlo para el procesamiento.

<sup>4</sup>Optamos por el cubo de datos **lineal** ya que el modelo logarítmico requiere de más técnica para poder manipularse

## Procesamiento con Python

Una vez descargado el cubo de datos podremos analizarlo con la ayuda de **Python**, un lenguaje de programación interpretado multiplataforma además de una librería llamada **Astropy**, una librería de Python enfocada al análisis y procesamiento astronómico. Además de **numpy** y **matplotlib** para la graficación de nuestros datos.

### Implementación en código

Primero verificaremos cuáles son las dimensiones de nuestro cubo de datos.

```
from astropy.io import fits
import numpy as np
import matplotlib.pyplot as plt

fits_file = './manga-7495-6102-LINCUBE.fits.gz'
#ruta del archivo .fits
hdul = fits.open(fits_file) #abre el archivo
.flits
flux = hdul["FLUX"].data #lee los datos de flujo
print(flux.shape) #dimensiones del cubo de datos
(espectral x espacial x espacial)
#resultado: (6732, 54, 54)
```

El resultado del código anterior indica que tenemos un cubo de datos de 54(espacial) x 54(espacial) = 2916 spaxels. Con una profundidad de 6732 espectros. Dando como total 19,630,512 voxels.

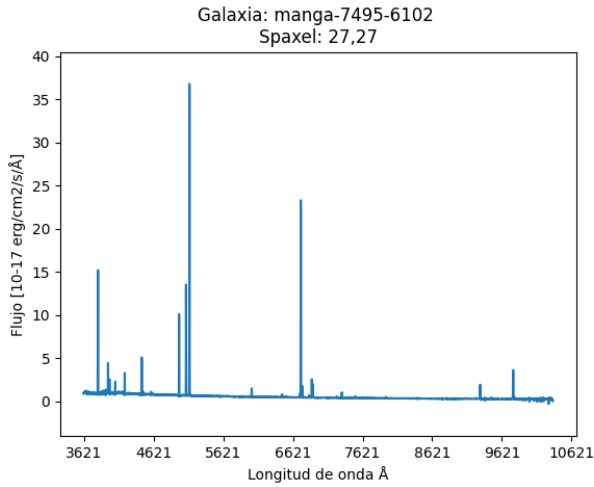
```
from astropy.io import fits
import numpy as np
import matplotlib.pyplot as plt
import matplotlib.ticker as ticker
fits_file = './manga-7495-6102-LINCUBE.fits.gz'
#ruta del archivo .fits
hdul = fits.open(fits_file) #abre el archivo
.flits
flux = hdul["FLUX"].data #lee los datos de flujo
print(flux.shape) #dimensiones del cubo de datos
#(espectral x espacial x espacial)
#el número de espectros es (6732, 54, 54)

fig=plt.figure()
ax = fig.add_subplot(111)
ax.plot(flux[:,27,27])
ax.set_ymargin(0.1)
#ax.set_xlim(self.datacube.getData("WAVE")[0])
ax.xaxis.set_major_formatter(
    ticker.FuncFormatter( lambda x, pos:
        '{}{}'.format('',
            str(int(x)+self.datacube.getData("WAVE")[0]))))
ax.set_xlabel("Longitud de onda Å")
ax.set_ylabel("Flujo [10-17 erg/cm2/s/Å]")
ax.set_title("Galaxia: manga-7495-6102 \nSpaxel:
27,27")

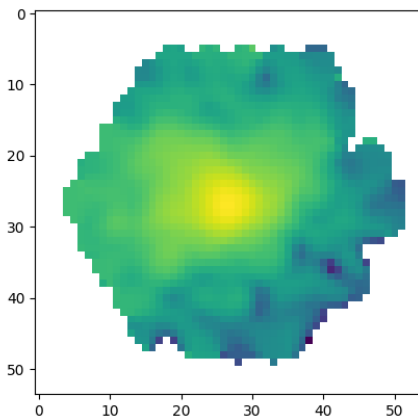
#extrayendo un corte del cubo en la longitud de
onda 900 a través de toda la imagen y
aplicando log10 - devuelve una imagen
fig=plt.figure()
bx=fig.add_subplot(111)
cx = bx.imshow(np.log10(flux[900,:,:]))

plt.show() #muestra los gráficos
```

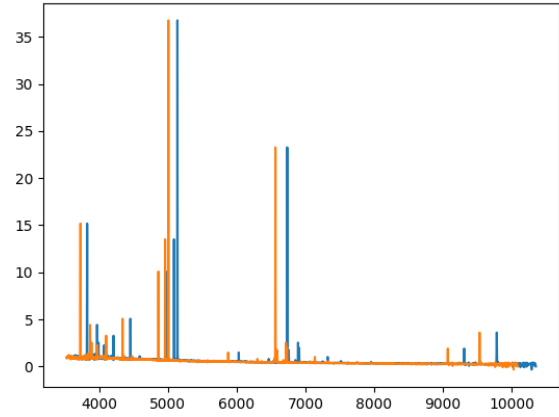
Al ejecutar este código se desplegarán 2 ventanas: una gráfica de con el espectro del spaxel (espectro: todos, espacial:27px, espacial:27px) **Fig. 8**, una gráfica con la imagen de la galaxia en (espectro: 900, espacial: todos, espacial: todos) **Fig. 9**.



**Fig. 8.** Espectro en (:, 27, 27)



**Fig. 9.** Imagen en (900, :, :) con logaritmo base 10 aplicado



**Fig. 10.** Ajuste de longitud de onda por redshift

#### **Ajuste de longitud de onda en los datos**

Dado que los datos de los espectros están situados en una longitud de onda específica, no empezando en 0, debemos adjuntar como información del eje x los datos que conforman las longitudes de onda específicas para cada espectro. Estos datos se hallan dentro del mismo archivo .fits en el apartado "WAVE".

#### **Ajuste de redshift o corrimiento al rojo**

Gracias al efecto Doppler, nuestros datos presentan un desfase en las longitudes que nos son presentadas. Por lo tanto, debemos realizar otro ajuste a las longitudes de onda basándonos en la cantidad de corrimiento al rojo. Para ello, recurriremos al DAPall del SDSS, un resumen de tablas finales de los resultados de MaNGA Data Analysis Pipeline (DAP), en el cual se encuentra la cantidad de redshift de nuestra galaxia en base al PLATEIFU.

Para ello se ocupara el siguiente código:

```
from astropy.io import fits
import numpy as np
import matplotlib.pyplot as plt
nombre_arch="./manga-7495-6102-LINCUBE.fits.gz"
hdu=fits.open(nombre_arch)
flujo = hdu["FLUX"].data
print("Dimensiones: ",flujo.shape)
hdu_catalog=fits.open(
    "./redshift_catalog/dapall-v3_1-3.1.0.fits")
data_catalog=hdu_catalog[1].data
ix=np.where(data_catalog["plateifu"] ==
    "7495-6102")
redshift=data_catalog[ "nsa_z"][ix][0]
longitud=hdu[ "WAVE"].data
wave=longitud/(1.0 + redshift)
fig=plt.figure()
ax=fig.add_subplot(111)
ax.plot(longitud, flujo[:,27,27])
ax.plot(wave,flujo[:,27,27])
plt.show()
```

Lo que nos da como resultado una gráfica con los dos espectros: espectro observado (azul) y el estado en reposo o rest frame (naranja), ambos ya calibrados en la longitud de onda respectivamente **Fig. 10**.

Para obtener una manipulación completa de los datos, me he basado en la información de la página del **Modelo de Datos**



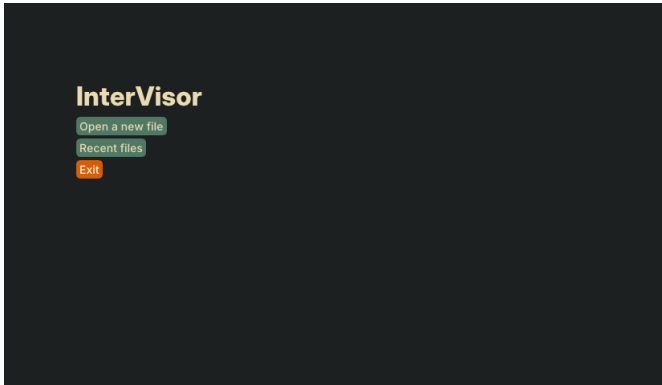


Fig. 11. Menu principal

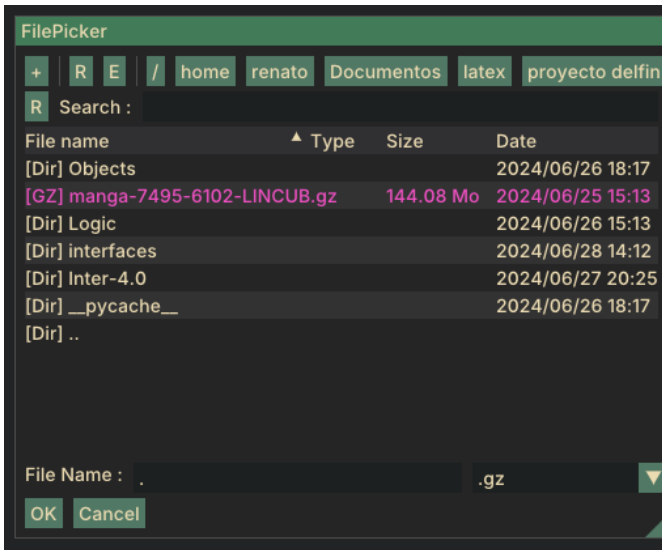


Fig. 12. Explorador de archivos

del SDSS, en la que se especifican los apartados de un archivo .fits.

### Creación de interfaz

Con el fin de facilitar el uso de la anterior implementación, me propuse crear un entorno gráfico para esta. Entre varias alternativas que hay en Python, he escogido Dearpygui, ya que es una librería gráfica multiplataforma, dinámica, acelerada por GPU y fácil de usar, un kit de herramientas de interfaz de usuario (GUI) para Python. Ya que en esta ocasión se manejarán grandes volúmenes de datos y requerimos una interfaz de alto rendimiento, cabe mencionar que esta librería también permite hacer gráficas o "plots", así que para el apartado del espectro me tomé la libertad de usar la alternativa de Dearpygui.

InterVisor permite abrir un nuevo archivo .fits, .gz o .fitz.gz desde el explorador de archivos con el botón **Open a new file**. Además de manejar un registro de los archivos abiertos con anterioridad, para ello está el botón **Recent files**. Para finalizar el programa, está la opción **Exit** (Fig. 11-13).

Al abrir algún archivo<sup>5</sup> se procede a la siguiente pantalla en la cual se presentan las diferentes opciones para procesar

<sup>5</sup>Si se trata de abrir 2 veces el mismo archivo, el programa no realizará esa nueva apertura.

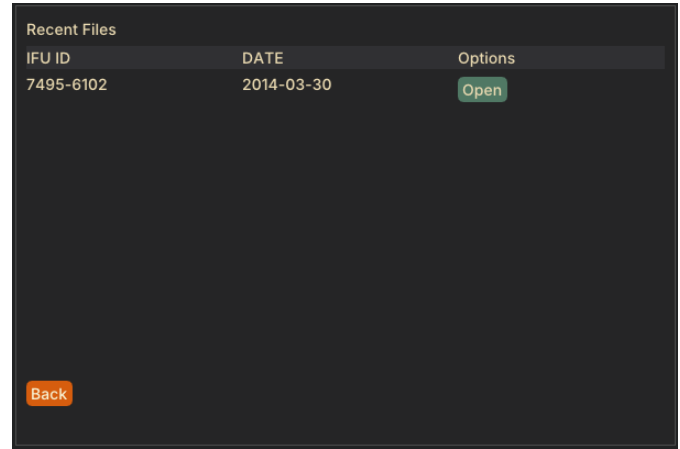


Fig. 13. Archivos Recientes

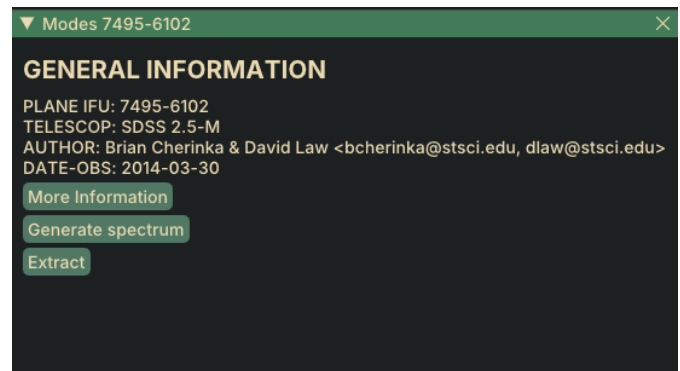


Fig. 14. Ventana de modos de procesamiento.

el archivo Fig. 14.

Actualmente, se cuenta con los siguientes modos de procesamiento:

1. **Visor de espectro:** En este apartado se permite escoger el spaxel (i,j) en el rango de longitud de onda en el que se quiere ver el espectro (Fig. 15).
2. **Extractor de imagen:** Después de ingresar una longitud de onda específica, se muestra la imagen del mapa. También está la opción de aplicar  $\log_{10}$  y  $\log_2$  (Fig. 16).
3. **Más información:** Un apartado para observar más datos sobre el archivo .fits (Fig. 17).
4. **Visor de líneas de emisión:** Se calcula el valor total las líneas mas importantes del espectro de cada uno de los pixeles en el cubo de datos. Para esta caracteriztica se calcula la integral de una gaussiana ajustada al segmento de datos que representa la línea de emision, todo esto de forma iterativa.

### OBSERVACIÓN DE LÍNEAS DE EMISIÓN

Cuando analizamos una línea de emisión debemos de examinar toda la curva de emisión ya que hasta ahora hemos visto y analizado solo un corte en el cubo de datos a lo largo del espectro. Por ejemplo tenemos la línea de emisión de la (Fig. 18) donde podemos apreciar que la formación de la

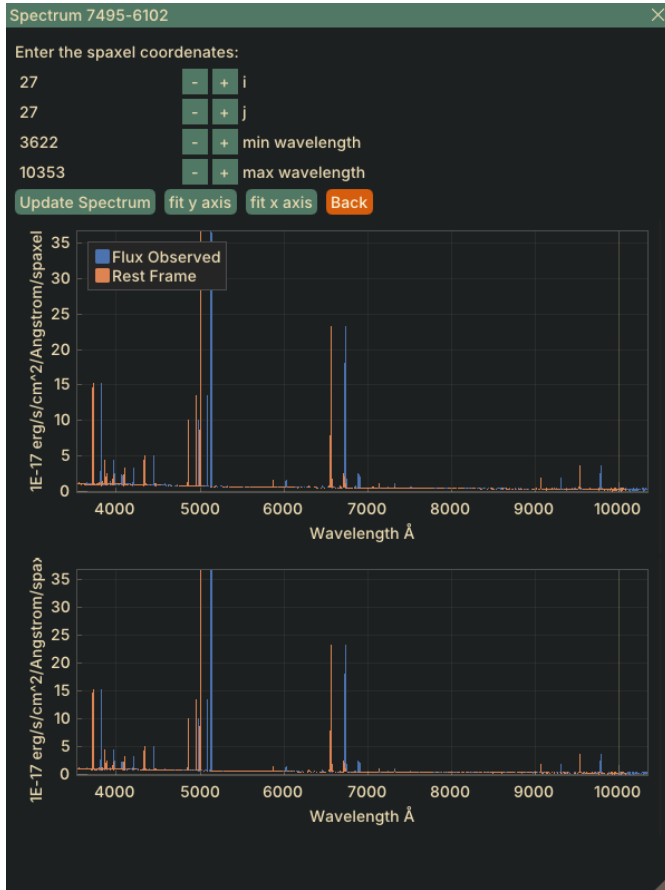


Fig. 15. Visor de espectro.

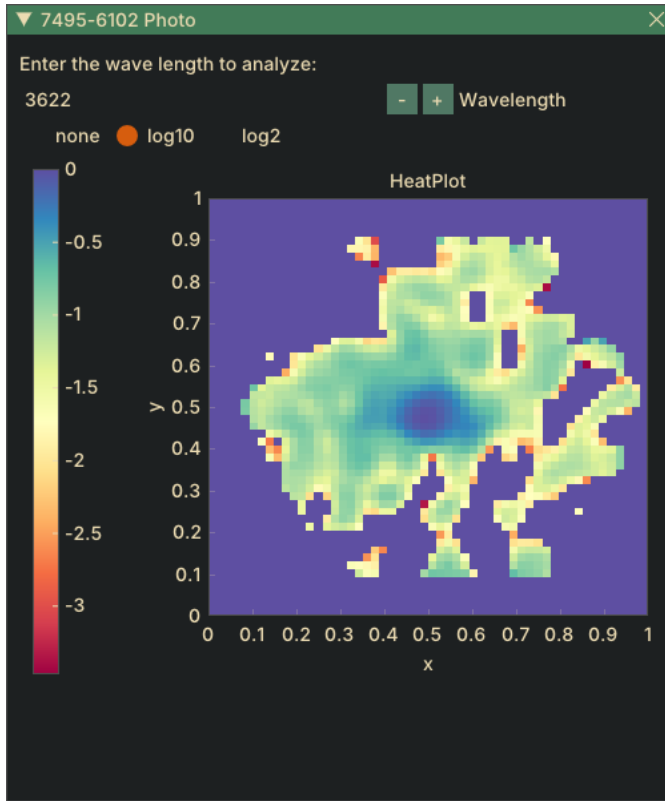


Fig. 16. Extractor de imagen.



Fig. 17. Mas informacion.

linea de emisión en el fondo esta formada por un segmento del cubo de datos y no solo por una longitud de onda en específico.

Para iniciar hay que marcar el metodo que ocuparemos. Primero hay que encontrar el valor mas alto en el espectro actual ( $m$ ) al igual que la longitud onda en la que se encuentra este valor ( $l$ ), a lo que procederemos a obtener el segmento de datos que conforman la linea de emision tanto en el espectro ( $flujo[]$ ) como en las longitudes de onda ( $long[]$ ), para esto definimos un margen  $p$  alrededor de  $l$ , de forma que obtenemos los intervalos cerrados de ( $flujo[l - p : l + p]$ ) y ( $long[l - p : l + p]$ ).

En seguida dividiremos ambos intervalos en en 3 partes, 2 laterales y una central, esto con el fin de obtener un promedio del continuo del espectro y obtener un mejor calculo de las lineas de emisión. Y la parte central sera utilizada para la generacion de una campana gaussiana la cual representa la linea de emisión completa. Para finalizar, calculamos la integral de la anterior gaussiana y la almacenamos en una matriz de  $n * m$  de acuerdo al tamaño de la imagen en el cubo de datos.

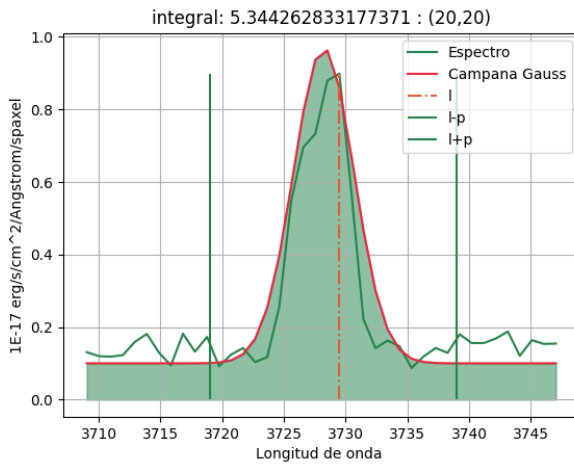
Este proceso tiene que realizarse para cada pixel en la imagen del cubo de datos. De esta manera obtendremos un mapa de lineas de emisión. Para generar la gaussiana utilizamos la siguiente formula:

$$f(x) = a \cdot \exp\left(-\frac{(x - x_0)^2}{2\sigma^2}\right)$$

Ademas usamos la funcion **curve fit** de la libreria **scipy.integrate** para hacer la gaussiana con un ajuste a los datos. El codigo utilizado se encuentra en mi repositorio de [GitHub](#).

De manera que se puede observa en la (Fig. 18) se encuentra el espectro observado, la campana de Gauss, los margenes que se emplearon para calcular la media del continuo alrededor de la linea de emision y el valor de la integral en la gaussiana generada.

Una vez con el proceso claro para un pixel en especifico en el cubo de datos podemos proceder a automatizar este proceso para cada uno de los pixeles en el cubo. Al igual que almacenar los datos en un arreglo de 2 dimensiones para al final generar una imagen (mapa) de lineas de emisión.



**Fig. 18.** Ejemplo de línea de emisión.

## DEFINICIONES

### pixel

Un píxel se refiere a un píxel físico del detector que es leído por una serie de dispositivos electrónicos.

### spaxel

Un spaxel (píxel espacial) se refiere a un elemento espacial de un cubo de datos reconstruido.

### voxel

Un voxel (píxel volumétrico) se refiere a un elemento de volumen (espacial  $\times$  espacial  $\times$  espectral) de un cubo de datos.

### redshift

El desplazamiento al rojo, como lo dice su nombre, sucede cuando las ondas de luz se estiran por el efecto Doppler y tienden a la región roja del espectro.

### pársec

Unidad de medida de distancia utilizada en astronomía, equivalente a aproximadamente 3.26 años luz.

### SDSS

El *Sloan Digital Sky Survey* es uno de los proyectos de investigación astronómica más grandes jamás emprendidos. Este tiene como objetivo la expansión del conocimiento humano sobre la evolución y estructura del espacio, formación de estrellas y galaxias.

### MaNGA

*Mapping Nearby Galaxies at APO* es un estudio espectroscópico de campo integral de galaxias dentro del Sloan Digital Sky Survey de cuarta generación (SDSS-IV) [Wei15].

### Unidad Astronómica / au

La unidad astronómica de distancia es una unidad de longitud que se utiliza en astronomía para las dimensiones típicas del Sistema Solar y se define 1 au como una longitud igual a exactamente 149 597 870 700 metros.

### S/N

La relación señal-ruido, SNR (Signal/Noise Ratio), S/N, mide qué tan bien se mide un objeto. Valores típicos:

S/N = 2-3: objeto apenas detectado

S/N = 10: podemos comenzar a hacer mediciones

S/N = 100: medición excelente.

## Archivos .fits

FITS (Flexible Image Transport System) es el formato de datos más utilizado dentro de la astronomía para transportar, analizar y archivar archivos de datos. FITS es mucho más que otro formato de imagen (como como JPG o GIF) y está diseñado principalmente para almacenar conjuntos de datos científicos que consta de matrices multidimensionales (imágenes) y tablas bidimensionales organizados en filas y columnas de información.

## REFERENCIAS

- [Oke+95] J. B. Oke et al. "The Keck Low-Resolution Imaging Spectrometer". En: *Publications Astronomical Society Pacific* 107 (1995), pág. 375. DOI: [10.1086/133562](https://doi.org/10.1086/133562). URL: <https://doi.org/10.1086/133562>.
- [VP05] Sébastien Vivès y Eric Prieto. "An original image slicer designed for Integral Field Spectroscopy with NIRSpec/JSWT". En: *Physics > Optics* (2005). URL: <https://arxiv.org/abs/physics/0512002>.
- [All06] Jeremy Allington-Smith. "Basic principles of integral field spectroscopy". En: *New Astronomy Reviews* 50.4-5 (2006), págs. 244-251. DOI: [10.1016/j.newar.2006.02.024](https://doi.org/10.1016/j.newar.2006.02.024). URL: [https://fcaglp.fcaglp.unlp.edu.ar/~observacional/papers/PDFs/tech-spectroscopy/spec\\_allington2006newar50.pdf](https://fcaglp.fcaglp.unlp.edu.ar/~observacional/papers/PDFs/tech-spectroscopy/spec_allington2006newar50.pdf).
- [SRR06] J. Schmoll, D. Robertson y D. Ryder. "Design and manufacture of micro-optical arrays using 3D diamond machining techniques". En: (2006). URL: <https://arxiv.org/pdf/astro-ph/0606145>.
- [14] *Bundle Ferrule Image*. Image from the Sloan Digital Sky Survey (SDSS). 2014. URL: [https://www.sdss4.org/wp-content/uploads/2014/11/Bundle\\_ferrule\\_new.png](https://www.sdss4.org/wp-content/uploads/2014/11/Bundle_ferrule_new.png).
- [Álv15] Enrique Joven Álvarez. *¿Quién inventó el telescopio?* Oct. de 2015. URL: <https://www.iac.es/es/blog/vialactea/2015/10/quien-invento-el-telescopio>.
- [Wei15] Anne-Marie Weijmans. "MaNGA: Mapping Nearby Galaxies at Apache Point Observatory". En: *arXiv preprint arXiv:1508.04314* (2015). URL: <https://arxiv.org/abs/1508.04314>.
- [Che+19] Brian Cherinka et al. "Marvin: A Tool Kit for Streamlined Access and Visualization of the SDSS-IV MaNGA Data Set". En: 158.2, 74 (ago. de 2019), pág. 74. DOI: [10.3847/1538-3881/ab2634](https://doi.org/10.3847/1538-3881/ab2634). arXiv: [1812.03833](https://arxiv.org/abs/1812.03833) [astro-ph.IM].
- [Ols21] Christopher Olsen. *The Birth of Astrophotography - Vatican Observatory*. Abr. de 2021. URL: <https://www.vaticanobservatory.org/sacred-space-astronomy/the-birth-of-astrophotography/>.
- [Ber+22] D. Berry et al. "Starlink - the original and best". En: *NA-SA/ADS* (2022). n.d. URL: <https://ui.adsabs.harvard.edu/abs/2022ASPC..532..559B/abstract>.
- [Dep] Department of Astronomy, UW-Madison. *Integral Field Spectroscopy*. <https://www.astro.wisc.edu/research/integral-field-spectroscopy/>. Consultado en junio de 2024.
- [Hyp] R. Nave HyperPhysics M. Olmo. *Fabry-Perot Interferometer*. n.d. URL: <http://hyperphysics.phy-astr.gsu.edu/hbasees/phyopt/fabry.html>.
- [OC0] R. O'Connell. *ASTR 511, Lecture notes*. n.d. URL: <https://rwoconne.github.io/rwoclass/ast511/lec10-f03.html>.

- [Obs] Las Cumbres Observatory. *Astronomical cameras*. n.d. URL: <https://lco.global/spacebook/telescopes/cameras/>.
- [web08] webteam@eso.org. *ESO - VIMOS*. n.d. ESO, 1999-2008. URL: <https://www.eso.org/sci/facilities/paranal/decommissioned/vimos.html>.