

**UNIVERSIDAD DE SANTIAGO DE CHILE**  
**FACULTAD DE INGENIERÍA INFORMÁTICA**  
**DEPARTAMENTO DE INGENIERÍA INFORMÁTICA**



## **REGISTRO DE IMÁGENES ASTRONÓMICAS**

**RODRIGO ANÍBAL JARA CARTAGENA**

Profesor Guía: Dr. Víctor Parada Daza  
Profesor Co-guía: Dr. Diego Mardones

Trabajo de titulación presentado  
en conformidad a los requisitos  
para obtener el Grado de Ingeniero  
de Ejecución en Computación e  
Informática

Santiago - Chile

2014

© Rodrigo Aníbal Jara Cartagena. Queda prohibida la reproducción parcial o total de esta obra en cualquier forma, medio o procedimiento sin permiso por escrito del autor.

SANTIAGO, MAYO 26 2014

REF.: Aprobación Trabajo de Titulación

En vista de los informes y calificaciones emitidas por los profesores designados por el Departamento corresponde y tomando en cuenta, además las disposiciones establecidas en las normas de Trabajo de Titulación y circular N° 140 del 20 de Abril de 1973, se procede a APROBAR el Trabajo de Titulación “**REGISTRO DE IMÁGENES ASTRONÓMICAS**” presentada por:

**JARA CARTAGENA RODRIGO ANÍBAL**

**Para optar al Título de Ingeniero en Ejecución en Computación e Informática**

Las calificaciones parciales obtenidas son las siguientes:

Sr. Superman                NOTA: \_\_\_\_\_

Sr. Hi-Man                NOTA: \_\_\_\_\_

Sra. Mujer Maravilla        NOTA: \_\_\_\_\_

Promedio de notas: \_\_\_\_\_

Escala de Notas: 1 a 7

Este trabajo de Titulación fue aprobado con esta fecha: MES DIA AÑO

---

Director

UNIVERSIDAD DE SANTIAGO DE CHILE

DEPARTAMENTO DE INGENIERÍA INFORMÁTICA

INGENIERÍA EN EJECUCIÓN EN COMPUTACIÓN E INFORMÁTICA

Av. Ecuador N° 3659

Estación Central - Santiago - Chile - Teléfono (56 -2 ) 27777777

[www.informatica.usach.cl](http://www.informatica.usach.cl)

## **DEDICATORIA**

Ahora tus mayores, quieren verte dedicado al estudio y a la formación de tu carácter y personalidad, porque estás en un periodo de formación. Tras los libros y de este humilde cuaderno, te espera un Chile grande y hermoso..., te esperan sus valles para que los cultives y cubras de bosques; te esperan sus cerros que esconden infinidad de minerales..., te espera el mar, te esperan los ríos para que conviertas sus fuerzas en electricidad. En fin..., tus Padres y tu Patria esperan verte convertido en un hombre de valer, en un ser útil y puedas así incorporarte al trabajo creador y progresista de esta querida tierra chilena.

Te desea muchos éxitos, tu amigo:



**ENRIQUE JARA URBINA  
COLECCIÓN “MI MEJOR AMIGO”  
PINTEMOS CHILE FÍSICO  
Fondo Ed. Educ. Moderna, 1966**

## **AGRADECIMIENTOS**

## TABLA DE CONTENIDOS

<b>CAPÍTULO 1. INTRODUCCIÓN.....</b>	<b>12</b>
1.1 ANTECEDENTES Y MOTIVACIÓN.....	12
1.2 DESCRIPCIÓN DEL PROBLEMA.....	14
1.3 SOLUCIÓN PROPUESTA .....	16
1.4 OBJETIVOS Y ALCANCES DEL PROYECTO .....	16
<i>1.4.1 Objetivo general .....</i>	<i>16</i>
<i>1.4.2 Objetivos específicos.....</i>	<i>16</i>
<i>1.4.3 Alcances .....</i>	<i>17</i>
1.5 METODOLOGÍA Y HERRAMIENTAS UTILIZADAS .....	17
1.6 ORGANIZACIÓN DEL DOCUMENTO .....	17
<b>CAPÍTULO 2. ASPECTOS GENERALES .....</b>	<b>19</b>
2.1 ¿QUÉ ES ALMA? .....	19
2.2 ¿QUÉ SE OBSERVA? .....	20
2.3 ¿CÓMO FUNCIONA ALMA? .....	21
2.4 ¿CÓMO SE FORMA UNA IMAGEN?.....	23
2.5 FORMATO DE UNA IMAGEN .....	23
2.6 NACE UN NUEVO PROYECTO .....	25
<b>CAPÍTULO 3. ALGORITMO DEL REGÍSTRO DE IMÁGENES.....</b>	<b>26</b>
3.1 DEFINICIÓN DEL ALGORITMO.....	26
3.2 PREPARACIÓN DE LA INFORMACIÓN .....	27
3.3 PROCESO DE NORMALIZACIÓN .....	31
<i>3.3.1 Recortar .....</i>	<i>31</i>
<i>3.3.2 Rotación .....</i>	<i>33</i>
<i>3.3.3 Escalamiento.....</i>	<i>36</i>
<i>3.3.4 Deproyectar .....</i>	<i>39</i>
3.4 TRAS BAMBALINAS.....	49
<i>3.4.1 Distancia entre puntos .....</i>	<i>49</i>
<i>3.4.2 Distancia eje mayor .....</i>	<i>50</i>
<i>3.4.3 Ángulo a rotar.....</i>	<i>51</i>
<i>3.4.4 Centro geométrico .....</i>	<i>52</i>

<b>CAPÍTULO 4. PRUEBAS .....</b>	<b>54</b>
4.1 MÓDULOS SUPERIORES .....	54
4.1.1 <i>Recortar</i> .....	55
4.1.2 <i>Rotación</i> .....	59
4.1.3 <i>Escalar</i> .....	63
4.1.4 <i>Alinear</i> .....	67
4.2 MÓDULOS INFERIORES .....	69
4.2.1 <i>Distancia entre puntos</i> .....	70
4.2.2 <i>Medida del eje mayor</i> .....	70
4.2.3 <i>Ángulo a rotar</i> .....	72
4.2.4 <i>Centro geométrico</i> .....	73
<b>CAPITULO 5. CONCLUSIONES .....</b>	<b>75</b>
<b>6. BIBLIOGRAFÍA .....</b>	<b>79</b>
<b>7. ANEXOS.....</b>	<b>81</b>
APILADO DE IMÁGENES. APLICACIÓN POST REGISTRO.....	82
INICIOS DEL APILADO DE IMÁGENES.....	84

## ÍNDICE DE TABLAS

Tabla 4.1: Recorte de una imagen .....	57
Tabla 4.2: Pruebas de rotación para una imagen .....	60
Tabla 4.3: Pruebas de escalamiento para una imagen .....	63
Tabla 4.4: Comparativa en tiempo de algoritmo con y sin interpolación.....	66
Tabla 4.5: Alineación de un punto en torno a coordenada (100, 100).....	67
Tabla 4.6: Tiempo para la deproyección de un objeto.....	68
Tabla 4.7: Medida de la distancia entre dos puntos .....	70
Tabla 4.8: Ejemplo de vector de puntos .....	71
Tabla 4.9: Búsqueda del eje mayor.....	72
Tabla 4.10: Determinación del ángulo complementario.....	73

## ÍNDICE DE ILUSTRACIONES

Figura 1-1: Entrada y salida del proceso de registro .....	15
Figura 1-1: Distribución de antenas en el Llano Chajnantor, Chile .....	19
Figura 2-2: Representación espectro electromagnético .....	20
Figura 2-3: Proceso de captura, digitalización y almacenamiento de datos .....	22
Figura 2-4: Formato de archivo fits para la galaxia NGC 5128 .....	24
Figura 3-1: Representación del proceso para registrar una imagen.....	26
Figura 3-2: Calculo del eje mayor de la elipse .....	28
Figura 3-3: Determinación del ángulo de proyección del eje mayor de la elipse con respecto al eje horizontal del plano.....	29
Figura 3-4: Cálculo del punto central de la elipse .....	30
Figura 3-5: Traslación de un objeto en el plano (x,y).....	38
Figura 3-6: Galaxia espiral Andrómeda.....	40
Figura 3-7: Galaxia espiral “sombrero” .....	40
Figura 3-8: Proyección de un punto elipse a la circunferencia.....	41
Figura 3-9: Catetos e hipotenusa de un triangulo rectángulo inscrito en una elipse .....	42
Figura 3-10: Elipse posicionada en el primer cuadrante de un plano cartesiano.....	43
Figura 3-11: Representación matricial de una elipse en un arreglo bidimensional .....	44
Figura 3-12: Medición de grados en un plano cartesiano con centro en (0,0).....	45
Figura 3-13: Medición de grados en un arreglo bidimensional con centro en ( $n/2, m/2$ )	45
Figura 3-14: Proyección de un punto elíptico a un punto del círculo.....	48
Figura 4-1: Imagen de prueba de la galaxia NGC 5128 .....	55
Figura 4-2: Imagen de prueba con objeto elíptico .....	56
Figura 4-3: Imagen de prueba con objeto elíptico .....	56
Figura 4-4: Imagen de prueba con objeto elíptico .....	57
Figura 4-5: Extracción de objeto .....	58
Figura 4-6: Extracción de objeto .....	58
Figura 4-7: Extracción de objeto .....	59

Figura 4-8: Rotación de una imagen en 5°.....	61
Figura 4-9: Rotación de una imagen en 45°.....	61
Figura 4-10: Rotación de una imagen en 73°.....	62
Figura 4-11: Rotación de una imagen en 128°.....	62
Figura 4-12: Escalamiento de imagen sin interpolación.....	64
Figura 4-13: Escalamiento a razón de dos .....	65
Figura 4-14: Escalamiento a razón de cinco .....	65
Figura 4-15: Imagen resultante al agrandar una imagen.....	67
Figura 4-16: Alineación de un objeto en torno al punto (100,100) .....	68
Figura 4-17: Deproyección de un objeto .....	69
Figura 4-18: Imagen de prueba con objeto elíptico .....	71
Figura 4-19: Imagen de prueba con objeto elíptico .....	71
Figura 4-20: Imagen de prueba con objeto elíptico .....	72
Figura 7-1: Proceso de apilado de imágenes a partir de un conjunto de imágenes normalizadas .....	83

## **RESUMEN**

El registro corresponde al proceso de transformación de diferentes imágenes a un sistema de coordenadas común. La ventaja de esta herramienta es que permite trabajar con datos de diferentes sensores, diferentes épocas o diferentes puntos de vista.

El espacio de aplicaciones es bastante amplio, siendo un área bastante atractiva para diversos autores. En la bibliografía existente, se pueden encontrar los usos de esta herramienta en el área de la medicina nuclear, donde se utiliza para testear la efectividad de un tratamiento en diferentes pacientes.

Atacama *Large Milimetric/Submilimetric Array*, esta generando actualmente un gran volumen de datos, los cuales necesitan ser analizados mediante algoritmos eficientes.

El presente documento aborda el registro de imágenes aplicándolo a las ciencias astronómicas, donde se normalizarán galaxias que posean puntualmente forma elíptica. Cada etapa de ajuste involucra un fundamento matemático, el cual será explicado mediante gráficas y posteriormente se indicarán los algoritmos involucrados que realizan el registro. Para finalizar se realizan una serie de pruebas que medirán la eficiencia, consistencia y el rendimiento del sistema generado.

Palabras claves: Registro de imágenes, normalización.

# CAPÍTULO 1. INTRODUCCIÓN

## 1.1 ANTECEDENTES Y MOTIVACIÓN

A partir del interés presentado por cinco universidades del país en el ámbito de la astronomía, nace la iniciativa de crear un observatorio virtual (CHIVO; *Chilean virtual obsevatory*) contando con el apoyo del observatorio ALMA, REUNA e instituciones privadas.

ALMA (Heisig, 2007), constantemente está generando un gran volumen de datos, surge la necesidad entonces de desarrollar nuevas herramientas para analizar esta información y algoritmos que sean capaces de procesarlos.

Una de estas herramientas, que posee gran utilidad debido a sus diversos campos de aplicación, corresponde al registro de imágenes; técnica que se adapta dependiendo del área de aplicación y el objetivo de estudio, aunque dicha herramienta posee gran complejidad, ha presentado mucho interés pues existe una amplia bibliografía de diversos autores que la avalan.

El registro de imágenes consiste en ajustar geométricamente un conjunto de ellas, a partir de una imagen tomada como referencia, de modo que al procesarlas coincidan espacialmente en un plano.

En el caso de imágenes astronómicas, es muy limitada su obtención. Por un lado siempre los instrumentos de medición tendrán un margen de error que se tiene que considerar, como también, no siempre se contará con las mejores condiciones atmosféricas. Adicionalmente la cantidad de factores extraterrestres que pueden influir, como por ejemplo, una nebulosa obstaculizando el objeto de estudio, conlleva a mejorar las técnicas de observación, y el tratamiento de la información generada. El registro de imágenes, permite complementar la información existente para posteriormente analizarla.

Una aplicación importante dentro del registro de imágenes, es en el área de la medicina nuclear. Los avances tecnológicos que involucran las imágenes médicas, han permitido pasar de una radiografía convencional a una tomografía axial computarizada (CT), resonancia magnética nuclear (MRI), resonancia magnética funcional (fMRI), tomografía computarizada por emisión de fotón único (SPECT), o la tomografía por emisión de positrones (PET). El médico actual se da cuenta de las posibilidades que ofrece la combinación de múltiples imágenes, pero para llevarlo a cabo y poder compararlas directamente, es necesario eliminar las diferencias de tamaño, posicionamiento, orientación o incluso la distorsión espacial. Todo este proceso, es el que conoceremos como registro de imágenes. A grandes rasgos, el registro de imágenes se puede dividir en dos etapas: primero una transformación geométrica, con la intención de lograr la mayor concordancia entre ellas y segundo un proceso de fusión, que es la visualización conjunta de ellas.

Una aplicación del registro de imágenes en medicina, es el seguimiento de la evolución de una enfermedad en pacientes epilépticos. Gracias a esta técnica se pueden comparar estudios de un mismo paciente en diferentes momentos, pudiendo así cuantificar diferencias entre estudios ictales e inter-ictales con SPECT, y así localizar mejor el foco epiléptico (Lewis, et al., 2000). Este ajuste de imágenes, también conocido como proceso de normalización, permite comparar resultados entre pacientes, ajustando cada paciente a una plantilla común, de esta manera se puede realizar por ejemplo, un análisis estadístico sobre como afecta la enfermedad, como evoluciona en diferentes pacientes en el tiempo, o la efectividad del tratamiento.

Existen aplicaciones dentro del campo de la cartografía. Al sur de Orinoco, en Venezuela, existe un proyecto conocido como CARTOSUR (Miguel, 2003), el cual buscó determinar si la fusión de ortoimágenes del radar SAR del proyecto CARTOSUR e imágenes *Landsat Tematic Mapper*, permiten lograr un mejor análisis visual de los elementos del territorio. La nubosidad propia de la región, impedía en muchos casos hacer vuelos fotogramétricos, o que estos lograran un área de cobertura libre de nubes. Ambos tipos de imágenes tuvieron que pasar por filtros para poder combinarlas, por

ejemplo, las imágenes de radar presentan un cierto tipo de ‘ruido’, conocido como moteado, el cual produce un efecto en la imagen de granularidad. En el caso de las imágenes de Landsat, presentan un desplazamiento en comparación a las orto-imágenes de radar (J., 1995), las cuales posteriormente tuvieron que ser corregidas. Finalmente se pudo concluir después de varias pruebas, que la fusión de estos dos tipos de imágenes ayudó definitivamente a identificar elementos de carácter antrópico y por lo tanto, apoyar las labores de captura de datos a partir de la interpretación de orto-imágenes de radar.

En este sentido el registro de imágenes cuenta con un conjunto de información que permite también el estudio a nivel astronómico, pudiendo abordar las características y variaciones de los elementos con los que se generarán estadísticas y estimaciones en el futuro.

## **1.2 DESCRIPCIÓN DEL PROBLEMA**

Actualmente los astrónomos para abordar el registro de imágenes deben realizar una serie de procedimientos. Se recopilan las imágenes del objeto a estudiar, se seleccionan aquellas que no perjudican la muestra, por ejemplo, si se desea estudiar la estructura de las galaxias elípticas y dentro de mi portafolio de imágenes existen algunas con ruido térmico intratable, es decir, con fuentes luminosas potentes que contaminan a sus vecinas, se deben eliminar para que se mantenga una muestra homogénea. Una vez que ya se tienen las imágenes pasan a una segunda etapa que consiste en extraer de la imagen general el objeto de interés. Ya teniendo los objetos detectados, se ‘normalizan’ ajustando su posición, escala, rotación y deproyección, bajo un valor común. Con esto, el astrónomo obtiene un conjunto de imágenes con las cuales puede trabajar y realizar estadísticas. Por ejemplo, si se realiza una suma matricial de ellas, calculando el promedio, se obtiene una imagen que destaca las características medias de las anteriores.

En la figura 1-1, se grafica de manera general la forma mediante la cual se extrae el objeto de interés de cada una de las imágenes pertenecientes a una muestra, con el fin de obtener un conjunto de imágenes ajustadas bajo los mismos parámetros.



Figura 1-1: Entrada y salida del proceso de registro

El problema existente en la serie de pasos realizados por el astrónomo está en la manualidad de las operaciones y saber en qué medida ajustar las imágenes. No solo cada astrónomo lo lleva a cabo con su propio estilo de trabajo, sino que además, el diseño de sus programas no obedece al diseño de los algoritmos que típicamente se sigue en el ámbito de la Ingeniería Informática. Un algoritmo como el que se propone desarrollar en éste trabajo de título, permite a los astrónomos combinar métodos de detección de objetos con el registro de imágenes.

En el registro de imágenes los parámetros de ajustes son determinados de forma instantánea por el sistema, sin necesidad de que la persona deba estimar o calcular la medida de dicho ajuste.

## **1.3 SOLUCIÓN PROPUESTA**

Se desarrollará un módulo en Python (Rogan & Muñoz, 2012), utilizando matrices y transformaciones geométricas en el plano que normalicen imágenes en dos dimensiones. Para poder abrir, operar y escribir sobre cada archivo, se utilizará el módulo Astropy (Astropy) y para operaciones matemáticas generales el módulo math (Python). De esta manera realizar un ajuste de ellas en relación a su posición, tamaño, rotación y depyección en un plano de dos dimensiones, generando una colección de imágenes a analizar.

## **1.4 OBJETIVOS Y ALCANCES DEL PROYECTO**

### *1.4.1 Objetivo general*

Crear una aplicación de escritorio para el registro de imágenes astronómicas mediante matrices y transformaciones geométricas en el plano.

### *1.4.2 Objetivos específicos*

1. Realizar un estado del arte sobre el problema de registro de imágenes.
2. Diseñar un algoritmo para las diferentes etapas del registro de imágenes.
3. Generar un conjunto matricial de imágenes para experimentación.
4. Indicar el objeto de interés para focalizar el registro en esa área.
5. Identificar e implementar parámetros de ajustes como: extracción, alineación, rotación, escala y depyección de un objeto.
6. Diseñar un experimento computacional para evaluar desempeño.

#### *1.4.3 Alcances*

En cuanto a los alcances y limitaciones, se debe tener presente que el software: trabaja únicamente con imágenes cuya extensión de archivos sea FITS (Greisen, 2012); corresponde a una herramienta de ajuste, por lo que solo modifica la información ya existente; trabaja con imágenes astronómicas; la información del archivo debe ser en dos dimensiones; no genera la información respecto a la posición del objeto, ésta debe ser entregada por el operador con el objeto de interés ya detectado; por último, el resultado obtenido dependerá netamente de la calidad de los datos de entrada.

### **1.5 METODOLOGÍA Y HERRAMIENTAS UTILIZADAS**

En este punto se describe la metodología empleada para lograr los objetivos de la investigación. Dada la naturaleza del objetivo de desarrollo, el método más adecuado es el ‘Desarrollo Incremental’. El proyecto se dividirá en etapas, en donde cada una de ellas entregará una nueva funcionalidad, es decir, cada nivel consiste en requerimientos del astrónomo, diseño del algoritmo con su codificación y pruebas. Esto permitirá atender de forma organizada, oportuna y eficiente los requerimientos, asegurar la calidad del producto resultante, proveer visibilidad del progreso, asegurar los compromisos con el cliente para lograr, en última instancia, la satisfacción de ellos.

### **1.6 ORGANIZACIÓN DEL DOCUMENTO**

El capítulo dos contextualiza el problema abordado, informa el estado actual de esta técnica y sus avances, da las definiciones y aspectos generales del proyecto.

El capítulo tres explica y presenta la manera en que se diseñaron los algoritmos de normalización y la forma en que se prepararon los datos.

El capítulo cuatro define las pruebas realizadas a las distintas funciones, presentando tablas comparativas de resultados, como también las imágenes obtenidas.

El capítulo cinco analiza los resultados obtenidos y da las conclusiones finales.

## CAPÍTULO 2. ASPECTOS GENERALES

### 2.1 ¿QUÉ ES ALMA?

A 5200 metros de altura, a casi 300 kilómetros de Antofagasta, crece día a día el observatorio terrestre más poderoso de toda la historia de la humanidad. Se trata de ‘Atacama Large Millimetric/Submillimetric Array’, un radio telescopio formado por 66 antenas de 12 y 7 metros de diámetro que trabajan como un solo cuerpo.

Al estar ubicado a gran altura, su atmósfera estable y baja humedad se transforma en una zona privilegiada para la radioastronomía, además su gran extensión permite distribuir las antenas sobre una superficie de 16 kilómetros de diámetro. En la figura 2-1 se muestra una vista aerea de la distribución de antenas en el llano de Chajnantor.



Figura 1-1: Distribución de antenas en el Llano Chajnantor, Chile

## 2.2 ¿QUÉ SE OBSERVA?

Astrónomos estudian diferentes tipos de objetos, como galaxias y sistemas planetarios, los que emiten variados tipos de onda, de las cuales el ser humano puede percibir una mínima parte (EXPLORA CONICYT, 2009).

Los lugares más fríos del universo, como las nubes de gas y polvo donde se forman las estrellas, emiten un tipo de onda llamadas milimétricas y submilimétricas, que pueden medir incluso menos del grosor de un cabello humano, como estos objetos emiten luz invisible, se requiere de instrumentos de un impresionante poder recolector de luz. Dado que ALMA posee una gran cantidad de antenas, cubriendo alrededor de 6.500 metros cuadrados, algo equivalente a una cancha de fútbol, esto permite detectar radiaciones muy débiles. Pero combinar las señales provenientes de las antenas es mucho mas complejo. La imagen de a continuación, figura 2-2, expone el espectro electromagnético y la similitud en tamaño que existe entre la longitud de onda y algún objeto.

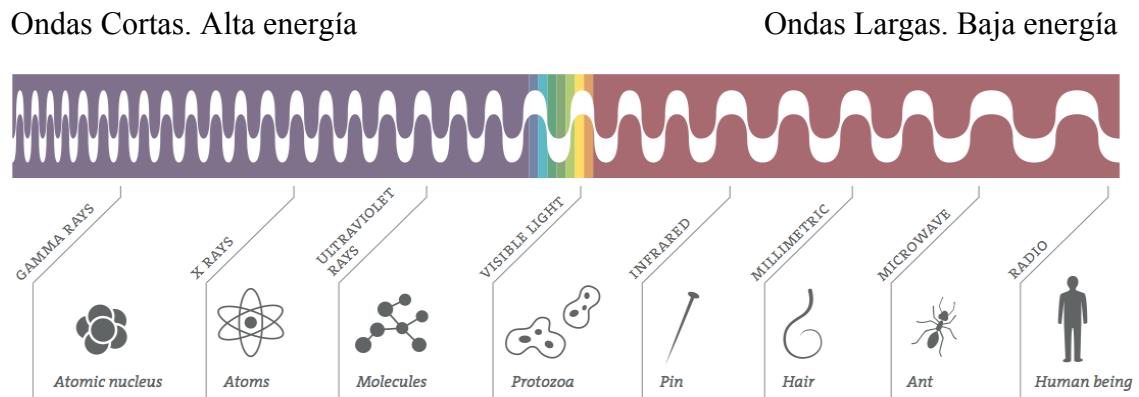


Figura 2-2: Representación espectro electromagnético

## 2.3 ¿CÓMO FUNCIONA ALMA?

La interferometría consiste en recolectar señales provenientes del espacio usando como mínimo dos antenas, al combinarlas se obtiene información de la fuente de emisión (Garay).

Esta técnica permite obtener imágenes de alta precisión, resultado que se obtendría con una antena o telescopio de 16 kilómetros de diámetro, pero que actualmente es técnicamente imposible construir, por lo cual es mas factible construir un conjunto de antenas y combinarlas, pero en la práctica no es sencillo.

Las señales desde que son captadas por las 66 antenas, hasta que son digitalizadas, deben superar perturbaciones y la posible atenuación al viajar varios kilómetros por la fibra óptica hacia el computador central. Incluso la señal cuando apenas penetra la atmósfera terrestre, es absorbida, desviada y retardada por moléculas de CO<sub>2</sub>, Oxígeno y Agua, efectos que son medidos y corregidos posteriormente.

ALMA primero debe tratar de apuntar todas las antenas a la misma región del cielo y captar con cada antena la señal astronómica, la que luego se convierte a formato digital. Posteriormente son transmitidas a un edificio central, donde un supercomputador combinará las señales transformándolas en datos que serán analizados. Esta gran computadora es llamada correlacionador y el número de operaciones ejecutadas es equivalente a lo que tres millones de computadoras portátiles realizan por segundo (ALMA). A continuación se detalla en cuatro pasos, figura 2-3, como la señal capturada es procesada antes de llegar al centro de operaciones.

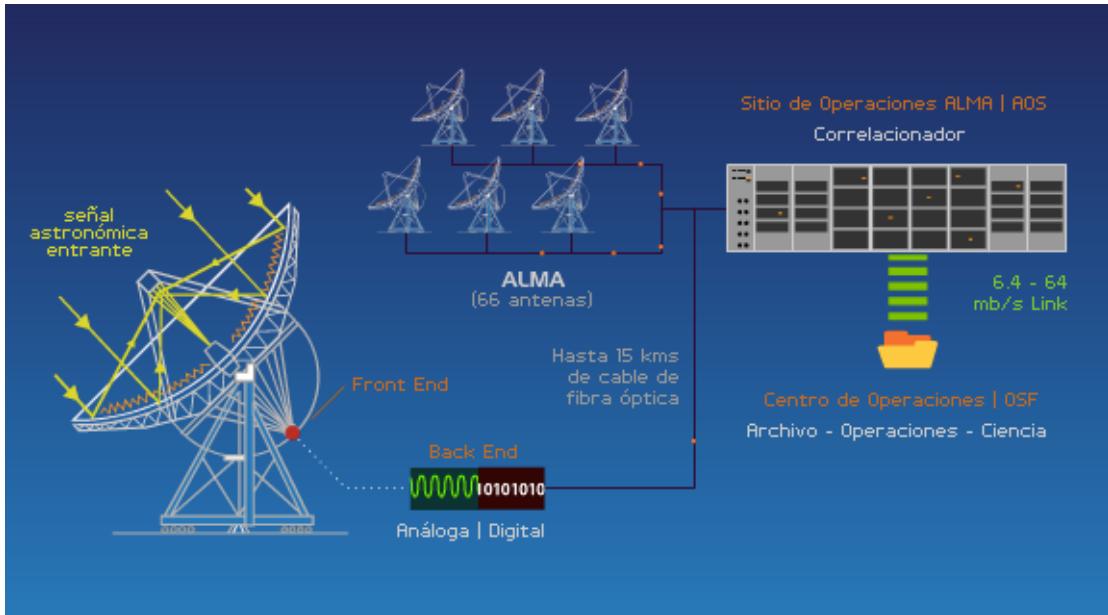


Figura 2-3: Proceso de captura, digitalización y almacenamiento de datos

El trayecto parte por la captura de una señal astronómica desde una antena, su función es recolectar y concentrar todas las señales a un punto llamado foco, el que después lo refleja a una segunda superficie llamada reflector, el que envía hacia atrás de la parábola las señales concentradamente. Luego pasa por el *Front End*, un proceso complejo de recepción, amplificación, conversión y digitalización, es el primer instrumento eléctrico por donde pasa la señal del cielo. Para disminuir el ruido es enfriado por criogenización a 4 Kelvin, donde se reduce su frecuencia. Luego la señal es digitalizada por el *Back End*, donde la señal recibida se convierte a una frecuencia mas baja, de entre 2 y 4 Ghz, luego un sistema de digitalización procesa la señal para que posteriormente sea transmitida por fibra óptica hasta el edificio central, donde el correlacionador combina la señal de todas las antenas. ALMA es controlado desde el Centro de Operaciones, donde se recibe, procesa y almacena la información.

## **2.4 ¿CÓMO SE FORMA UNA IMAGEN?**

El ojo humano puede extraer una cantidad impresionante de información de la luz que los atraviesa y eso se logra mapeando la distribución de la luz sobre el campo de visión. A ese ‘mapa’ se le llama imagen.

ALMA para conformar una imagen correlaciona las señales provenientes de cada par de antenas, las que son matemáticamente comparadas miles de millones de veces por segundo, obteniendo una precisión de una millonésima de millonésima de segundo, logrando una resolución diez veces mayor que el Telescopio Espacial Hubble (Instituto de Astrofísica de Andalucía, 2000).

## **2.5 FORMATO DE UNA IMAGEN**

Un formato de archivo comúnmente utilizado en astronomía es FITS o *Flexible Image Transport System*. Es utilizado para almacenar espectro electromagnéticos, lista de fotones, cubos de datos, etcétera. Cada archivo FITS consiste en una o más cabeceras que contienen secuencias de 80 cadenas de caracteres fijos que llevan pares de valores, los cuales detallan todo lo respectivo a la observación, observatorio, tiempo de observación, fecha, objeto de estudio, coordenadas del cielo, número de dimensiones, etcétera.

El recuadro de a continuación, figura 2-5, muestra un ejemplo de cabecera para la galaxia NGC 5128, conocida también como Centauro A.

```

BITPIX = -64 / No.Bits per pixel
NAXIS = 2 /No.dimensions
NAXIS1 = 1095 / Length X axis
NAXIS2 = 1095 / Length Y axis
DATE = '14/11/96' /Date of FITS file creation
ORIGIN = 'CASB -- STScI' /Origin of FITS image
PLTLABEL= 'J 1313' /Observatory plate label
PLATEID = '00AK' /GSSS Plate ID
REGION = 'S270' /GSSS Region Name
DATE-OBS= '11/03/75' /UT date of Observation
UT = '15:40:00.00' /UT time of observation
EPOCH = 1.9751907958984E+03 /Epoch of plate
PLTRAH = 13 /Plate center RA
PLTRAM = 35 /
PLTRAS = 4.0093000000000E-01 /
PLTDECSN= '-' /Plate center Dec
PLTDEC0 = 45 /
PLTDECM = 15 /
PLTDECS = 2.0015190000000E+01 /
EQUINOX = 2.0000000000000E+03 /Julian Reference frame equinox
EXPOSURE= 9.0000000000000E+01 /Exposure time minutes
BANDPASS= 0 /GSSS Bandpass code
PLTGRADE= 1 /Plate grade
PLTSCALE= 6.7200000000000E+01 /Plate Scale arcsec per mm
SITELAT = '-31:16:24.00' /Latitude of Observatory
SITELONG= '+149:03:42.00' /Longitude of Observatory
TELESCOP= 'UK 48-inch Schmidt' /Telescope where plate taken
CNPIX1 = 10178 /X corner (pixels)
CNPIX2 = 11256 /Y corner
DATATYPE= 'INTEGER*2' /Type of Data
SCANIMG = 'S270_00AK_00_00.PIM' /Name of original scan
SCANNUM = 0 /Identifies scan of the plate
DCHOPPED= F /Image repaired for chopping effects
DSHEARED= F /Image repaired for shearing effects
DSCNDNUM= 0 /Identifies descendant of plate scan image
XPIXELSZ= 2.5284450000000E+01 /X pixel size microns
YPIXELSZ= 2.5284450000000E+01 /Y pixel size microns
PP01 = 0.0000000000000E+00 /Orientation Coefficients

AMDY10 = 2.0957116003475E-06 /
AMDY11 = -7.7230568130285E-08 /
AMDY12 = 0.0000000000000E+00 /
AMDY13 = 0.0000000000000E+00 /
AMDY14 = 0.0000000000000E+00 /
AMDY15 = 0.0000000000000E+00 /
AMDY16 = 0.0000000000000E+00 /
AMDY17 = 0.0000000000000E+00 /
AMDY18 = 0.0000000000000E+00 /
AMDY19 = 0.0000000000000E+00 /
AMDY20 = 0.0000000000000E+00 /

Based on photographic data obtained using The UK Schmidt Telescope.
The UK Schmidt Telescope was operated by the Royal Observatory
Edinburgh, with funding from the UK Science and Engineering Research
Council, until 1988 June, and thereafter by the Anglo-Australian
Observatory. Original plate material is copyright (c) the Royal
Observatory Edinburgh and the Anglo-Australian Observatory. The
plates were processed into the present compressed digital form with
their permission. The Digitized Sky Survey was produced at the Space
Telescope Science Institute under US Government grant NAG W-2166.

Investigators using these scans are requested to include the above
acknowledgements in any publications.

Copyright (c) 1993, 1994, Association of Universities for Research in
Astronomy, Inc. All rights reserved.
DATAMAX = 19680 /Maximum data value
DATAMIN = 1579 /Minimum data value
OBJECT = 'NGC 5128'
OBJCTRA = '13 25 27.609' /Object Right Ascension (J2000)
OBJCTDEC= '-43 01 08.69' /Object Declination (J2000)
OBJCTX = 10725.02 /Object X on plate (pixels)
OBJCTY = 11803.70 /Object Y on plate (pixels)

```

*Figura 2-4: Formato de archivo FITS para la galaxia NGC 5128*

Cuando inicialmente fue publicado el formato FITS, un número de palabras claves fueron entregadas pero desafortunadamente no estaban las definiciones explícitas. El recuadro anterior muestra como es la cabecera generalmente, pudiendo presentar ciertas diferencias entre archivos, como por ejemplo en el nombre de la etiqueta, o la cantidad de información que la persona deseé poner.

## 2.6 NACE UN NUEVO PROYECTO

Hay una creciente necesidad de almacenar, distribuir y procesar grandes volúmenes de datos generados por observatorios en Chile, los cuales día a día transmiten hacia los centros de almacenamiento distribuidos principalmente en Europa, Estados Unidos y Japón.

Para mejorar el desarrollo científico y permitir un mayor crecimiento en investigaciones, se está desarrollando una plataforma virtual distribuida, que apoyaría a los astrónomos chilenos en la recopilación y acumulación de datos, la cual incluye el desarrollo de algoritmos eficientes en búsqueda semántica, procesamiento de datos a gran escala, minería de datos, entre otros.

Se trata de una innovación a nivel país que tiene como objetivo la creación e implementación de una plataforma astro-informática. El proyecto es gestionado por alumnos y profesores de la UTFSM como parte de un proyecto FONDEF, junto a ALMA y REUNA, y que beneficiará a las Universidades de Chile, Católica, de Concepción y de Santiago.

Cada universidad cuenta con un equipo de trabajo desarrollando algoritmos para esta plataforma virtual, en el caso de la Universidad de Santiago, es una herramienta avanzada de alto interés astronómico, la cual se explicará detalladamente en el capítulo siguiente.

## CAPÍTULO 3. ALGORITMO DEL REGÍSTRO DE IMÁGENES

### 3.1 DEFINICIÓN DEL ALGORITMO

Para llevar a cabo la normalización se han definido cuatro etapas, figura 3-1, en las que son ajustadas un conjunto de imágenes a un formato general antes de ser analizadas. Se entiende que en cada imagen existe el mismo objeto de estudio, pero puede variar su posición, tamaño y forma. Al ir procesando cada imagen, se van aplicando los filtros respectivos, los cuales permitirán que las imágenes queden con un patrón común que facilite su estudio. El siguiente esquema visualiza como se desarrolla el proceso de normalización en un conjunto de imágenes.

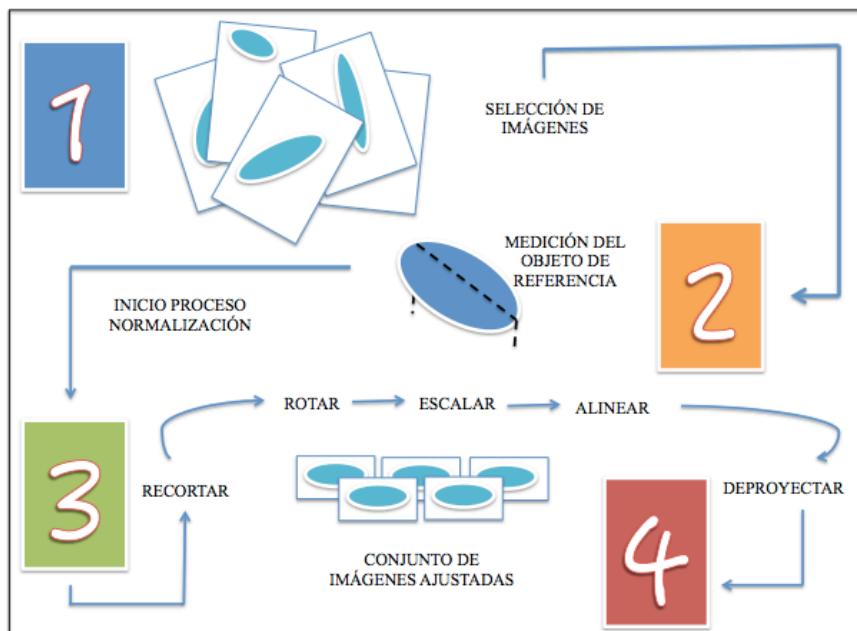


Figura 3-1: Representación del proceso para registrar una imagen

El diagrama parte con la selección de imágenes que poseen objetos elípticos que al usuario le interesa estudiar. Una de estas imágenes será tomada como referencia

para realizar el ajuste de las restantes. Cada vez que una imagen es leída por el programa, este realiza la toma de medidas, es decir, la longitud del eje mayor del objeto y su ángulo de proyección respecto a un eje horizontal. Luego pasa por cinco etapas de ajuste, extracción del objeto de interés, rotación del objeto con respecto al eje mayor, el cual se deja sin pendiente, luego se escala y se alinea en relación a la imagen de referencia, para que finalmente se pueda deproyectar, transformando la elipse en un círculo. Cada nivel de ajuste contempla el ingreso de información externa respecto a la posición actualizada del objeto, es decir, se necesita saber antes y después de realizar el ajuste en qué coordenadas va quedando el objeto.

### **3.2 PREPARACIÓN DE LA INFORMACIÓN**

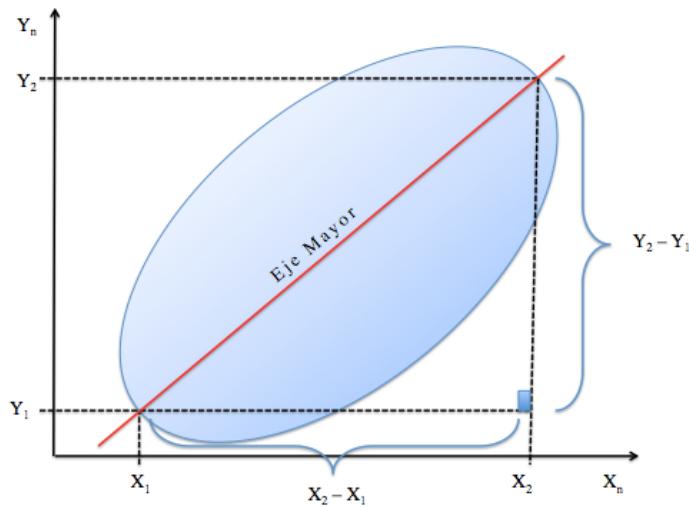
Corresponde en esta instancia seleccionar diversos objetos con forma similar a una elipse, a los cuales se les realizarán transformaciones geométricas en el plano (Fundación Polar). En este proceso se toma una imagen de referencia con la cual se trabajará para llevar a cabo la normalización. Los objetos elípticos a manipular corresponden a galaxias que presentan una forma elíptica.

Es esencial conocer en cada etapa las coordenadas que componen el borde de la elipse, lo que ayuda a procesar la información útil dentro de la imagen, determinar la medida del eje mayor e inclinación de esta pseudo elipse, es aquí donde se puede determinar que tan grande o que tan pequeña es del resto, su medida de deproyección y a su vez saber su ángulo de rotación respecto al eje horizontal.

En una elipse el eje mayor se define como los puntos que están a una mayor distancia, los cuales se determinan calculando las diferencias que existen entre los puntos correspondientes al borde de la elipse. Para saber la distancia entre dos puntos en un plano, es utilizado el “Teorema de Pitágoras”, fórmula (3.1), el cual dice, que la suma de los catetos al cuadrado es igual a la hipotenusa al cuadrado, que en este caso es la distancia entre dos puntos de la elipse, como lo muestra la figura 3-2.

*Fórmula (3.1): Expresión algebraica para el Teorema de Pitágoras.*

$$a^2 = b^2 + c^2$$



*Figura 3-2: Calculo del eje mayor de la ellipse*

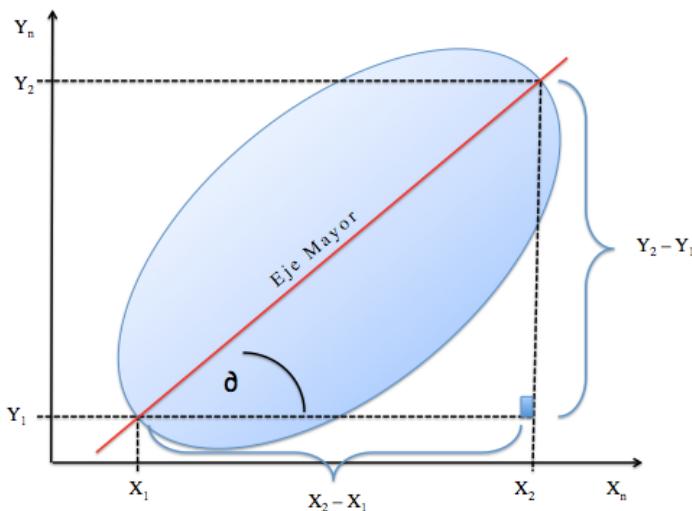
Matemáticamente la expresión que calcula la distancia entre dos puntos se escribe:

$$\text{eje mayor} = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

Conociendo la distancia del eje mayor, se procede a determinar su ángulo en relación al eje horizontal del plano. Esto permitirá reconocer cuánto se necesita rotar el objeto antes de ser comparado. Para encontrar el ángulo de rotación, basta calcular la inversa de la función tangente, fórmula (3.2), en la que aplicando el arco tangente a la razón existente entre la diferencia de los puntos en el plano vertical, dividida la diferencia de los puntos en el plano horizontal, dará como resultado el ángulo de la recta. Gráficamente se puede apreciar en la figura 3-3.

*Fórmula (3.2): Cálculo del ángulo formado entre el eje mayor de la elipse y el eje horizontal del plano.*

$$\theta = \tan^{-1} \left( \frac{Y_2 - Y_1}{X_2 - X_1} \right)$$



*Figura 3-3: Determinación del ángulo de proyección del eje mayor de la elipse con respecto al eje horizontal del plano*

Por último, se calcula el punto centro del objeto, lo que permite ajustar el resto de los objetos a ese punto de referencia. Para definir el punto, basta con dividir el eje mayor en dos y estimar a qué coordenada corresponde. Para identificar la posición que ocupa en el plano el centro del objeto, se le suma a una de las coordenadas que pertenece a la esquina del eje mayor la proyección de la mitad de este eje.

Para saber la posición en el plano del centro del objeto se toma una esquina del eje mayor como referencia y se le suma tanto en el eje x (eje horizontal) como en el eje 'y' (eje vertical) la proyección de la mitad del eje mayor.

La expresión matemática de a continuación corresponde al cálculo de la distancia en los dos ejes del plano.

$$\text{Distancia en eje } x = \cos(\theta) * \text{distancia eje mayor} * 0.5$$

$$\text{Distancia en eje } y = \sin(\theta) * \text{distancia eje mayor} * 0.5$$

Una vez reconocida la distancia del eje respecto al punto menor ya es posible determinar la coordenada del punto central. La expresión que se muestra a continuación, suma estas distancias a la coordenada menor del eje obteniendo así el punto central del objeto.

$$\text{Punto central eje} = (X_1 + \text{distancia en eje } x, Y_1 + \text{distancia en eje } y)$$

Gráficamente, lo podemos apreciar en la siguiente imagen.

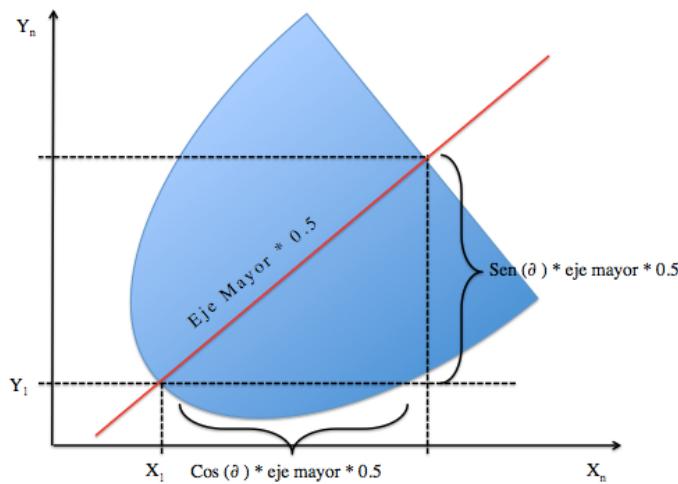


Figura 3-4: Cálculo del punto central de la elipse

Los mecanismos antes expuestos desde el punto de vista algorítmico se detallan en el punto siguiente.

### 3.3 PROCESO DE NORMALIZACIÓN

La normalización define cuatro etapas:

- Extraer de la imagen el objeto de interés.
- Rotar el objeto dejándolo paralelo al eje horizontal.
- Escalar el objeto dejándolo del mismo tamaño que el objeto de referencia.
- Alinear el objeto a un punto de referencia común.
- Deproyectar el objeto hasta que quede como un círculo.

#### 3.3.1 Recortar

Para extraer de la imagen el objeto de interés, descartando todo el resto de la información que no es necesaria, se debe recortar la imagen y así trabajar solo con la información útil, lo que permite realizar sólo los cálculos necesarios, entregando la información más precisa. Al realizar la observación de una determinada ubicación en el espacio, puede ocurrir que la imagen posea más de un objeto, pero el foco de interés radica en uno solo de ellos, en este caso es sumamente útil individualizar y concentrar el estudio en la información que se necesita descartando el resto. Para recortar la imagen se debe determinar los puntos extremos del cuerpo elíptico, es decir, mirándolo desde el punto matricial, encontrar el mayor y el menor valor de la fila y columna. Para esto se recorre el borde del objeto hasta encontrar los cuatro puntos. El algoritmo que presento a continuación realiza la extracción.

```
1 Función recortar(matriz, borde):  
2     fila_mayor = 0  
3     fila_menor = borde[0]  
4     columna_mayor = 0
```

```

5      columna_menor = borde[1]
6      i = 0
7      mientras i != largo(borde)-1:
8          Si i es impar:
9              continúe
10         Si fila_mayor < borde[i]:
11             fila_mayor = borde[i]
12         Si fila_menor > borde[i]:
13             fila_menor = borde[i]
14         Si columna_mayor < borde[i+1]:
15             columna_mayor = borde[i+1]
16         Si columna_menor > borde[i+1]:
17             columna_menor = borde[i+1]
18     i = 0
19     mientras i != largo(borde)-1:
20         Si i es impar:
21             continúe
22             bord[i] = borde[i] - fila_menor
23             borde[i+1] = borde[i+1] - columna_menor
24             i = fila_menor
25     mientras i != fila_mayor+1:
26         j = columna_menor
27         mientras j != columna_mayor+1:
28             matriz_final[i - fila_menor][j -
29                 columna_menor] = matriz[i][j]
30     retorno matriz_final

```

### 3.3.2 Rotación

La rotación consiste en girar un objeto en torno a punto. Por cada imagen que ingresa se calcula su ángulo de proyección respecto al eje horizontal para que posteriormente se rote en esta medida hasta que el ángulo sea cero, dado que es mas sencillo dejar bajo la misma pendiente que tratar de hacer coincidir a una de referencia. Todo esto es bajo la preferencia, de que para que un conjunto de imágenes pueda ser estudiada, toda la muestra debe poseer los mismos parámetros.

Como en la toma de medidas, el ángulo esta determinado, en esta etapa se da la orden de rotar en sentido antihorario el complemento del ángulo. La imagen es básicamente una matriz de ‘n’ filas y ‘m’ columnas, con posiciones del (0,0) al (n-1,m-1). Para poder rotarla se multiplica una matriz de rotación definida con seno y coseno por cada coordenada. La rotación se realiza en torno al punto (0, 0), pero para poder rotarla en torno a su centro debe ser trasladada antes y después de girarla.

```
1 Función rotar(matriz,NAXIS1,NAXIS2,angulo):
2     i = j = 0
3     punto_central_x = NAXIS1*0.5
4     punto_central_y = NAXIS2*0.5
5     Mientras i distinto NAXIS1:
6         Mientras j distinto NAXIS2:
7             a = (i-punto_central_x)*coseno + (j-punto_central_y)*seno)-
8                 traslacion_vuelta_x
9             b=-(i-punto_central_x)*seno + (j-punto_central_y)*coseno)-
10                traslacion_vuelta_y
11                matriz_final[a][b] = matriz[i][j]
```

La función en el punto 7 y el punto 9 realiza la multiplicación de la matriz de rotación por la posición (i,j), para eso cada punto es trasladado hasta que el centro de la

imagen quede en el punto (0,0), luego se multiplica para finalmente trasladarla nuevamente a su origen. Esto se realiza de esta manera, ya que la matriz de rotación definida, fórmula (3.3), rota una matriz en torno a su punto de origen.

*Fórmula (3.3): Matriz de rotación.*

$$\begin{matrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{matrix}$$

Cuando una imagen es rotada, existen puntos que quedan situados en pixeles intermedios. Para no dejar puntos vacíos, cuando un punto queda en una posición decimal, se aproxima hacia arriba y hacia abajo, dejando en ambas posiciones el valor deseado, por ejemplo si hay que dejar en la posición (2.4 , 3.7) el valor 1032 se guarda en la posición (2.0 , 3.0), (2.0 , 4.0), (3.0 , 3.0) y (3.0 , 4.0). El pseudo código de a continuación realiza esta tarea.

```

1   mientras x distinto NAXIS1:
2   mientras y distinto NAXIS2:
3       bandera_decimal_a = 100
4       bandera_decimal_b = 100
5       a = ((x-punto_central_x)*coseno + (y-punto_central_y)*seno)+(-
6           traslación_vuelta_x)
7       b = (-(x-punto_central_x)*seno + (y-punto_central_y)*coseno)+(-
8           traslación_vuelta_y)
9       Si a es decimal:
10          bandera_decimal_a = 101
11       Si b es decimal:
12          bandera_decimal_b = 110
13       suma_banderas = bandera_decimal_a + bandera_decimal_b
14       while(VERDADERO):

```

```

15      Si suma_banderas == 201:
16          matriz_final[int(a)][b] = matriz[x][y]
17          matriz_final[ceil(a)][b] = matriz[x][y]
18          break
19      Si suma_banderas == 210:
20          matriz_final[a][int(b)] = matriz[x][y]
21          matriz_final[a][math.ceil(b)] = matriz[x][y]
22          break
23      Si suma_banderas == 211:
24          matriz_final[int(a)][int(b)] = matriz[x][y]
25          matriz_final[math.ceil(a)][math.ceil(b)] =
26              matriz[x][y]
27          matriz_final[int(a)][math.ceil(b)] = matriz[x][y]
28          matriz_final[math.ceil(a)][int(b)] = matriz[x][y]
29          break
30      Si suma_banderas == 200:
31          matriz_final[a][b] = matriz[x][y]
32          break

```

La base de este algoritmo es identificar que posiciones, al rotar la imagen, quedan en un valor decimal, de esta manera se ajusta hacia arriba y hacia abajo. Para esto se definen dos banderas, cada una con un valor 100, entonces cuando el valor en ‘x’ es decimal se le asigna el valor 101 a la ‘bandera\_x’, si el valor de ‘y’ es decimal se le asigna a la ‘bandera\_y’ el valor 110, de esta manera si la suma de las bandera da 201, significa que tan solo ‘x’ es decimal, si suman 210 significa que tan solo ‘y’ es decimal, si suman 211 significa ambos son decimales y si suman 200 significa que ambos son números enteros. De esta manera se sabe cuantos puntos al contorno en esa posición se debe llenar, por ejemplo, si al rotar la coordenada queda en la posición (2.3 , 4), quiere decir que la suma de las bandera da 201, en otros términos el valor de x = 2.3 quedará

como 2 y como 3, entonces en la posición (2 , 4) y en la posición (3 , 4) se guardará el valor correspondiente.

### 3.3.3 Escalamiento

Escalar es agrandar o achicar una imagen en un factor determinado. Para escalar una imagen se necesita saber la razón de escala, es decir, cuanto mas grande o mas pequeña es el eje mayor de la imagen actual con respecto a la imagen de referencia. Una vez identificado el valor, se procede a multiplicar cada coordenada de la matriz por este factor, fórmula (3.4), permitiendo distribuir los puntos uniformemente dentro del plano. Si el factor de escala es un número entero, quedarán puntos intermedios sin información, por lo que deberán ser formados a partir del promedio de los cuatro puntos mas cercanos. En caso de que el propósito sea achicar la imagen, no será necesario aplicar lo antes expuesto.

*Fórmula (3.4): Multiplicación de una coordenada por un factor de escala*

$$(X_2, Y_2) = (X_1 * \text{factor}, Y_1 * \text{factor})$$

El algoritmo recorre cada posición de la matriz y en la posición resultante, que es la posición original por el factor de escala, guarda el valor de la posición original. Esto se realiza siempre y cuando se necesite escalar la imagen, si el tamaño actual es igual al original no es necesario llevar a cabo esta tarea.

```
1 Función escalar(matriz, NAXIS1, NAXIS2, razón):
2     Si razón == 1:
3         retornar matriz
4     mientras x distinto NAXIS1:
5         mientras y distinto NAXIS2:
```

6

$$\text{matriz\_final}[x*\text{razón}][y*\text{razón}] = \text{matriz}[x][y]$$

Luego de escalar la imagen y solo en caso de que se haya agrandado, se deberá interpolar, es decir, completar la información faltante a partir de los puntos cercanos. Los puntos a considerar no se ven respecto al punto vacío, sino que a partir de los cuatro puntos con información se rellenan todos los puntos intermedios, fórmula (3.5), esto es mas sencillo de diseñar, ya que desde un punto con información a otro, la distancia es igual al valor de la razón de escala. El algoritmo 2.4 muestra la manera en que se realiza la interpolación.

*Fórmula (3.5): Aproximación de un punto a partir de los cuatro vecinos.*

$$\text{media} = \frac{(x, y) + (x + \text{razón}, y) + (x, y + \text{razón}) + (x + \text{razón}, y + \text{razón})}{4}$$

```

1   x = y = 0
2   mientras x < NAXIS1 * razón
3       mientras y < NAXIS2 * razón
4           media = matriz_final[x][y] + matriz_final[x+razón][y] +
5           matriz_final[x][y+razón] + matriz_final[x+razón][y+razón]
6           media = media / 4.0
7       y = y + razón
8   y = 0
9   x = x + razón

```

### 3.3.4 Alineación

Alinear geométricamente representa el desplazamiento de un punto o un conjunto de puntos según un vector fijo no nulo. El proceso de alinear consiste en llevar el centro del objeto de estudio a punto de referencia común, con el fin de que todos los

objetos en cada imagen tenga la misma posición antes de ser analizados. Como la imagen que está siendo procesada ya posee el mismo tamaño y rotación que la de referencia, se puede trasladar tranquilamente el objeto a ese punto, sin preocuparse de que la información quede fuera de los márgenes.

Si la imagen de referencia es más pequeña que la actual, al desplazar los puntos, solo aquellos que estén dentro de la dimensión de la imagen de referencia quedarán guardados. El resto, al no poseer información, serán descartados automáticamente. Cuando la imagen de referencia es más grande, se traspasará la información completamente sin descartar puntos.

Para determinar el vector de translación, se calcula la distancia en ‘x’ y en ‘y’, entre el punto de referencia y el punto centro actual, de esta manera los valores son desplazado en esa distancia. La figura 3.4 grafica un ejemplo de un punto que debe ser desplazado 1 unidades en ‘x’, y 2 unidades en ‘y’.

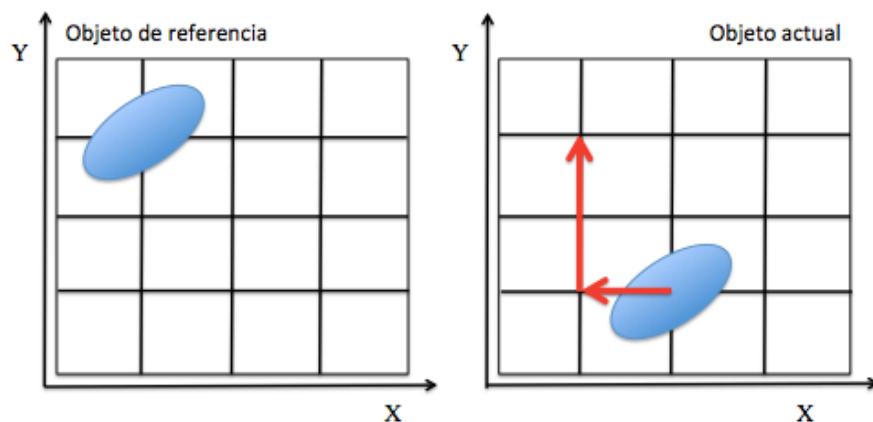


Figura 3-5: Traslación de un objeto en el plano ( $x,y$ )

La función alinear no necesita interpolar la información dado que el vector de translación, tiene valores enteros, entonces la posición resultante siempre caerá en una posición entera.

```

1 Función alinear(x1,y1,x2,y2,dim_1_x,dim_1_y,dim_2_x,dim_2_y, actual):
2     diferencia_x = x1 - x2
3     diferencia_y = y1 - y2
4     mientras i distinto dim_2_x:
5         mientras j distinto dim_2_y:
6             x = i + diferencia_x
7             y = j + diferencia_y
8             if ( x > 0 and x < dim_1_x and y > 0 and y < dim_1_y):
9                 matriz_final[x][y] = actual[i][j]
10    retornar matriz_final

```

### *3.3.4 Deproyectar*

Es el proceso por el cual se cambia la perspectiva en que se ve un objeto. Si pusiéramos las imágenes en un plano tridimensional (x, y, z,), siendo z la coordenada que da la profundidad, se apreciaría que el ángulo de inclinación respecto al eje varía en cada imagen, lo que también se podría llamar perspectiva. Para poder comparar correctamente las imágenes es necesario dejar todos los objetos paralelos al plano (x , y).

La imagen 3-6 y 3-7 muestra como dos galaxias espirales recrean esta misma situación, por un lado una esta mas recostada, destacando más su canto, en cambio en la otra, es mas visible su centro y su extensión.

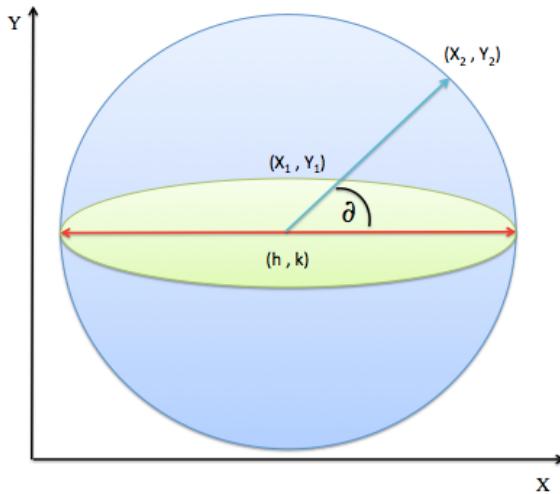


*Figura 3-6: Galaxia espiral Andrómeda*



*Figura 3-7: Galaxia espiral “sombrero”*

Para corregir esta situación, es necesario dejar ambos objetos bajo el mismo enfoque, es decir, cambiar el angulo del objeto en relación a un tercer eje. Lamentablemente esto no es factible, dado que no existe una tercera coordenada. Para lograr un efecto similar, la elipse se puede ‘estirar’ de forma tal, que manteniendo el mismo eje mayor de la elipse, se trasforme en un círculo, es decir, encontrar una relación entre un punto de la elipse y un punto del círculo. En la figura 3-8 tenemos un círculo con una elipse inscrita, compartiendo el mismo diámetro.



*Figura 3-8: Proyección de un punto elipse a la circunferencia*

Podemos notar que tanto  $(X_1, Y_1)$  e  $(X_2, Y_2)$  comparten el mismo ángulo, es decir sabiendo el ángulo compartido y la distancia de cada punto al centro geométrico del cuerpo, se podría determinar la coordenada equivalente en el círculo. Como el punto  $(X_1, Y_1)$  y el centro  $(h, k)$  es conocido, aplicando propiedades trigonométricas sencillas se puede obtener el ángulo.

Supongamos que la distancia del punto  $(h, k)$  a un punto  $(X, Y)$  de la elipse mide ‘ $a$ ’, y trazando una perpendicular desde ese punto al eje horizontal de largo  $Y - k$ , se estaría formando un triángulo rectángulo donde  $Y - k$ , es el cateto opuesto y ‘ $a$ ’, es la hipotenusa, situación graficada en la imagen 3-9.

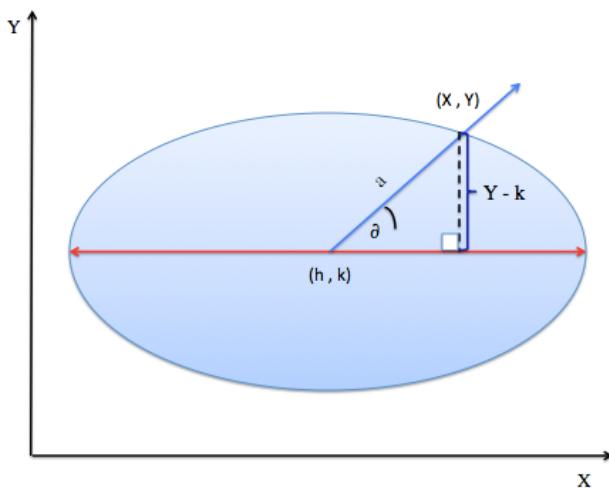


Figura 3-9: Catetos e hipotenusa de un triangulo rectángulo inscrito en una elipse

Para determinar el ángulo  $\theta$ , se calcula el arcoseno de la razón entre el cateto opuesto al ángulo con la hipotenusa. Fórmula 3.6.

Fórmula (3.6): Estimación de un ángulo a partir de la inversa de la función seno.

$$\sin^{-1} \left( \frac{Y - k}{a} \right) = \theta$$

Este ángulo es vital para encontrar el punto equivalente en el círculo, ya que teniendo la distancia, que en el caso de un punto borde es igual al radio, y escribiendo el punto del círculo en forma paramétrica se puede obtener la coordenada. La expresión de a continuación, fórmula (3.7), representa la forma paramétrica de un punto (x,y).

Fórmula (3.7): Coordenada perteneciente al círculo escrita de forma paramétrica.

$$(x, y) = (d\cos(\theta), d\sin(\theta))$$

Remplazando el ángulo encontrado a partir del punto en la elipse y la distancia del centro del cuerpo al punto proyectado, se obtiene la coordenada dentro del círculo, esta distancia varía según el punto, pero se determina en relación al punto mayor que está a distancia de un radio. Lamentablemente este escenario cambia un poco en el ámbito computacional, ya que el primer cuadrante del plano cartesiano no es equivalente al plano matricial de una imagen, sino más bien es similar. Primero el plano cartesiano tiene el punto  $(0,0)$  en el origen, en cambio en una imagen el  $(0,0)$  se encuentra en la primera posición de la esquina superior de la matriz, como lo vemos en la imagen 3-10 y 3-11.

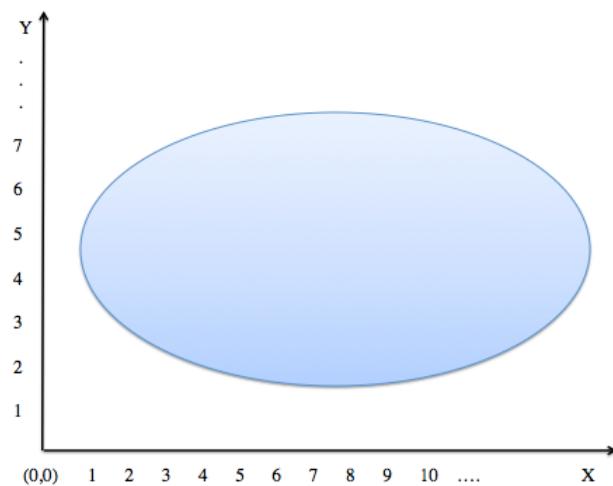
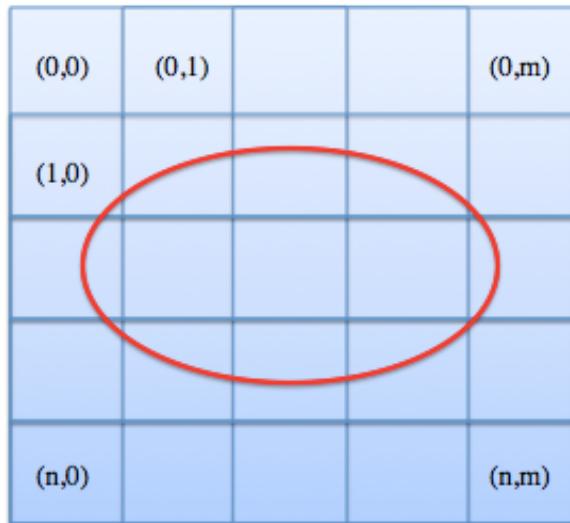


Figura 3-10: Elipse posesionada en el primer cuadrante de un plano cartesiano



*Figura 3-11: Representación matricial de una elipse en un arreglo bidimensional*

Esto cambia completamente la forma de aplicar las funciones trigonométricas al plano normal dado que la coordenada  $(x,y)$  representada en una “matriz computacional” se lee  $(y,x)$ , entonces la expresión paramétrica del círculo también se escribe al revés, como lo muestra la fórmula (3.11). Otro aspecto a considerar, es la forma de leer el ángulo, en el plano cartesiano se hace en sentido antihorario, figura 3-12, pero en este caso se hace en sentido horario, figura 3-13, también por el mismo motivo.

*Fórmula (3.11): Expresión parámetrica de una coordenada del círculo en un plano computacional.*

$$(x, y) = (d\sin(x), d\cos(x))$$

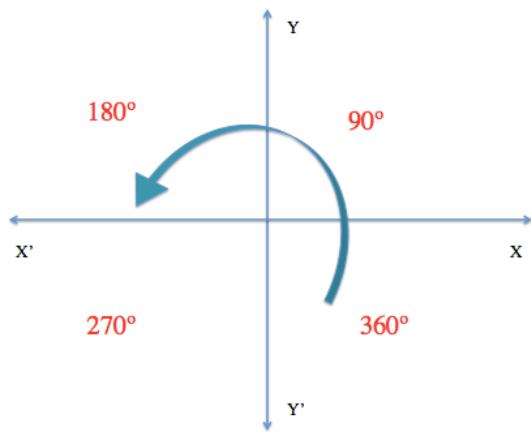


Figura 3-12: Medición de grados en un plano cartesiano con centro en (0,0)

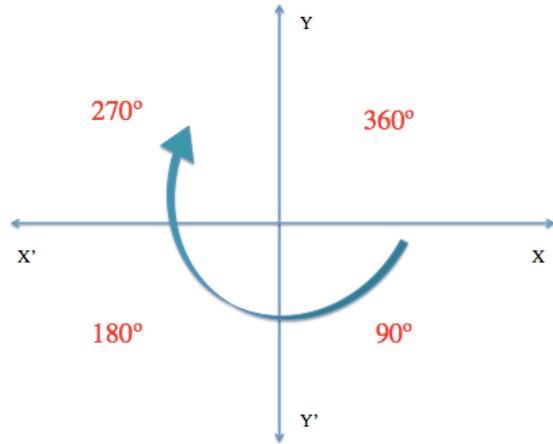


Figura 3-13: Medición de grados en un arreglo bidimensional con centro en ( $n/2, m/2$ )

Teniendo claro esta relación ya es posible proyectar los puntos correctamente. El algoritmo de a continuación refleja la manera en que se elige un punto de la elipse y en base al ángulo que posee se proyecta hacia el círculo. Este recibe como parámetro de entrada el centro geométrico del objeto, la dimensión de la imagen NAXIS1, NAXIS2, un vector con información de las coordenadas del borde del objeto, la matriz de información del objeto y el radio de la circunferencia, que sería la mitad del eje mayor de la elipse.

```

1 Función deprojectar(x,y,NAXIS1,NAXIS2,borde,matriz,radio):
2   mientras i != NAXIS1:
3     mientras j != NAXIS2:
4       a = b = c = d = 0
5       mientras r != largo(borde-1):
6         Si ( r es impar):
7           continúe
8         Si i < borde[r]:
9           a = 1
10        Si i > borde[r]:
11          b = 10
12        Si j > borde[r+1]:
13          c = 100
14        Si j < borde[r+1]:
15          d = 1000
16        Si i == borde[r] and j == borde[r+1]:
17          a,b,c,d = 1,10,100,1000
18        break
19        Si (a + b + c + d != 1111 or x == i and y == j):
20          continúe
21          ángulo = arcoseno((j - y)/distancia(x,y,i,j))
22        mientras(TRUE):
23          Si i > x and j > y:
24            ángulo = ángulo
25            break
26          Si i < x and j > y:
27            ángulo = 360 - ángulo
28            break

```

```

29           Si i < x and j < y:
30               ángulo = 180 + ángulo
31               break
32           Si i > x and j < y:
33               ángulo = 180 – ángulo
34               break
35           Si i == x and j > y:
36               ángulo = 0
37               break
38           Si i == x and j < y:
39               ángulo = 180
40               break
41           Si i > x and j == y:
42               ángulo = 90
43               break
44           Si i < x and j == y:
45               ángulo = 270
46               break
47       fil = x + (seno(ángulo*3.14/180))*distancia
48       col = y + (coseno(ángulo*3.14/180))*distancia
49   matriz_final[fil,col] = matriz[i][j]
50   retornar matriz_final

```

La clave de este algoritmo es determinar que puntos pertenecen al objeto antes de desplazar, para esto podemos establecer lo siguiente: todo punto perteneciente al objeto va a estar entre cuatro puntos a menos que sea el borde de este. Por esta razón el algoritmo parte recorriendo toda la matriz imagen hasta detectar un punto objeto. Una vez localizado se calcula el ángulo relacionado, el cual siempre va a estar entre 0 y 90 grados, ya que se calcula en base a la distancia entre los puntos y el cateto opuesto, por

lo que determinando en que parte del plano esta en relación al punto  $(h,k)$  se le suma lo faltante. La imagen 3-14 recrea esta situación. Como el punto  $(x,y)$  a proyectar se encuentra en la esquina superior derecha del centro y dado que los ángulos se cuentan en sentido antihorario, el ángulo no vale  $\partial^\circ$ , sino mas bien  $360 - \partial$ , si hubiésemos tomado un punto de la esquina superior izquierda, su ángulo sería  $180 + \partial$ , si el punto está en la parte inferior izquierda  $180 - \partial$ , y finalmente si ubiese estado en el lado inferior derecho, el ángulo se mantiene. Ya determinado el ángulo, el algoritmo ocupando la ecuación paramétrica de un punto en el circulo, y calculando brevemente la distancia que debería estar el punto en relación al punto mas distante, se obtiene la coordenada del punto proyectado. Este proceso se repite tantas veces como puntos pertenezcan al objeto de estudio. Una vez finalizado todo el proceso, se retorna una matriz que posee la proyección de la elipse a un circulo.

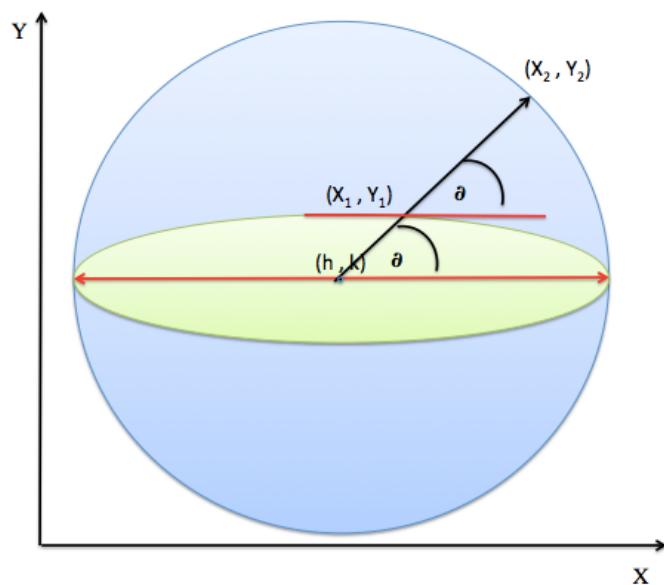


Figura 3-14: Proyección de un punto elíptico a un punto del círculo

## 3.4 TRAS BAMBALINAS

Si bien el el registro de imágenes contempla cuatro niveles de ajustes, detrás de estos módulos actúan constantemente funciones que entregan información valiosa antes de realizar cualquier ajuste a un objeto en el plano. Estas funciones son ‘invisibles’ para el usuario, es decir, no entregan resultados tangible u observables, sino mas bien es el sistema de cómputo que aporta información sobre los cuerpos geométricos, como tamaño, grado de inclinación y posición, para que los módulos que realizan el registro sepan a priori en que medida ajustar el objeto.

### 3.4.1 Distancia entre puntos

Para un objeto elíptico posicionado dentro de una imagen es necesario medirlo antes de realizar antes de poder escalarlo, de esta manera se puede determinar que tan grande o mas pequeño es el objeto de la imagen actual con la imagen de referencia. La medición de un objeto se basa en determinar la distancia entre los puntos mas extremos de la elipse en el plano, conocida también como distancia euclíadiana, la cual se calcula mediante la fórmula (3.12).

*Fórmula (3.12): Distancia euclíadiana*

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

Esta fórmula indica que la raíz cuadrada de la suma de la diferencia de las coordenadas al cuadrado es igual a la distancia ‘d’. Algorítmicamente se ve de la siguiente manera:

- 1 Función distancia (x1, y1, x2, y2):
- 2        retornar raiz\_cuadrada((x2-x1)^2 + (y2-y1)^2))

La función recibe como parámetros de entrada el par de coordenadas de las cuales se desea saber su distancia, para luego al igual que la fórmula anterior retornar la distancia.

### *3.4.2 Distancia eje mayor*

Esta función no es más que una aplicación de la función distancia, lo distinto es que busca la mayor distancia entre todos los puntos que componen el borde del objeto elíptico. De esta manera cuando encuentra los puntos mas distantes, sabe de inmediato que corresponde al eje mayor de la elipse. El algoritmo siguiente realiza la búsqueda del eje mayor:

```
1 Función eje_mayor(borde):
2     info_elipse = [0,0,0,0,0]
3     mientras i distinto largo(borde)-2:
4         Si ( i es impar):
5             continúe
6             i = i + 2
7             mientras j distinto largo(borde)-1:
8                 Si ( j es impar):
9                     continúe
10                dis = distancia(borde[i],borde[i+1],borde[j],borde[j+1])
11                Si dis > info_elipse[0]:
12                    info_elipse[0] = dis
13                    info_elipse[1] = borde[i]
14                    info_elipse[2] = borde[i+1]
15                    info_elipse[3] = borde[j]
16                    info_elipse[4] = borde[j+1]
17    retornar info_elipse
```

La función recibe como parámetro de entrada un arreglo que contiene todas las posiciones del borde de la elipse, de las cuales se localizará las más distantes.

En la línea número dos, se crea un vector que guardará en su posición 0 la distancia del eje, y en las posiciones restantes sus coordenadas, por ejemplo, si el eje mayor mide 30 unidades, y sus coordenadas son ( $X_1$  ,  $Y_1$ ) y ( $X_2$  ,  $Y_2$ ), el vector se escribirá de la siguiente manera:

$$\text{info\_vector} = [30, X_1, Y_1, X_2, Y_2]$$

Este vector es iniciado con ceros, ya que cada posición actuará como una variable auxiliar mientras se busca el eje mayor.

Los ciclos ‘mientras’ de la línea 3 y 7 permitirán encontrar la mayor distancia, para esto el ciclo de la línea 3 va seleccionando una coordenada del vector y el ciclo de la línea 7 compara la coordenada elegida en el ciclo 3 con todas las coordenadas restantes. Si esta resulta ser mayor que la encontrada anteriormente, se reemplaza en la posición cero del vector ‘info\_vector’ la distancia encontrada y se actualizan las coordenadas de las cuales se extrajo la distancia.

### 3.4.3 Ángulo a rotar

Para rotar un objeto en el plano es necesario saber cuánto es lo que se necesita rotar. La siguiente función determina a partir de la información generada anteriormente cuánto es lo que necesita rotarse el objeto para que su eje mayor quede paralelo al eje horizontal en un plano.

- 1 Función angulo\_a\_rotar ( $x_1, y_1, x_2, y_2$ ):
- 2 Si ( $y_2 - y_1$ ) == 0
- 3       retorne 90
- 4       Sino:
- 5           retorne  $180 - \text{arcotangente}((y_2 - y_1)/(x_2 - x_1))$

La función recibe como entrada las coordenadas extremas del eje mayor del objeto. La linea dos pregunta si el eje mayor es paralelo al eje ‘y’, de esta manera se debe rotar, de lo contrario calcula el arcotangente del ángulo. En la linea cinco se le resta esta cantidad a 180, esto es debido a que la función que realiza la rotación gira en sentido antihorario el cuerpo, por tal motivo cuando se determina el ángulo de la recta se calcula su complemento.

Matemáticamente consiste en determinar la tangente del ángulo ‘ $\theta$ ’ que forma la recta con la dirección positiva del eje de las abscisas, por lo tanto, para determinar el ángulo se realiza el proceso inverso, fórmula (3.13).

*Fórmula (3.13): Cálculo de la pendiente y el ángulo de una recta en el plano.*

$$m = \frac{(y_2 - y_1)}{(x_2 - x_1)}$$

$$\theta = \arctan(m)$$

#### 3.4.4 Centro geométrico

Al momento de alinear un objeto es necesario determinar el punto central de este para poder trasladar esa coordenada al punto que se tiene como referencia. Para poder determinar la coordenada del centro geométrico del objeto, se calcula a partir del tamaño de la mitad del eje mayor de la elipse, la proyección del ángulo en el eje horizontal y vertical.

- 1 Función centro\_geométrico(dis,x1,y1,x2,y2,angulo):
- 2 Si ( $\text{ángulo} < 90$ ):
- 3     punto\_central = [x1 + 0.5\*coseno(ángulo)\*dis, y1 + 0.5\*seno(ángulo)\*dis]
- 4 Si ( $\text{ángulo} > 90$ ):

```

6     punto_central = [x2 - 0.5*cos(angulo)*dis, y2 -
7         0.5*seno(ángulo)*dis]
8     Si (ángulo == 90):
9         punto_central = [(x1-x2)/2 , y1]
10    retornar punto_central

```

La función recibe como parámetro de entrada la medida del eje ‘dis’, las coordenadas extremas y el ángulo que corresponde a la pendiente.

La clave de este algoritmo es saber si la pendiente del eje es positiva o negativa, ya que la proyección de la recta en los ejes, se hace a partir de la coordenada mas baja del objeto al punto central, por este motivo se pregunta si la recta es mayor o menor que  $90^\circ$ . Al obtener el punto central del eje, debe sumarse la coordenada menor del eje del objeto, esto se hace de esta manera ya que el punto centro encontrado es el centro del objeto y no el centro del objeto en relación al plano. Por tal motivo en la linea tres y cinco del plano se suma  $(x_1,y_1)$  y  $(x_2,y_2)$ . Finalmente se retorna la coordenada.

Estos cuatro últimos algoritmos entregan toda la información necesaria para que cada etapa del registro realice correctamente un ajuste de la información. Partiendo desde la toma de medidas del objeto de referencia, hasta la actualización del mismo en cada etapa.

## CAPÍTULO 4. PRUEBAS

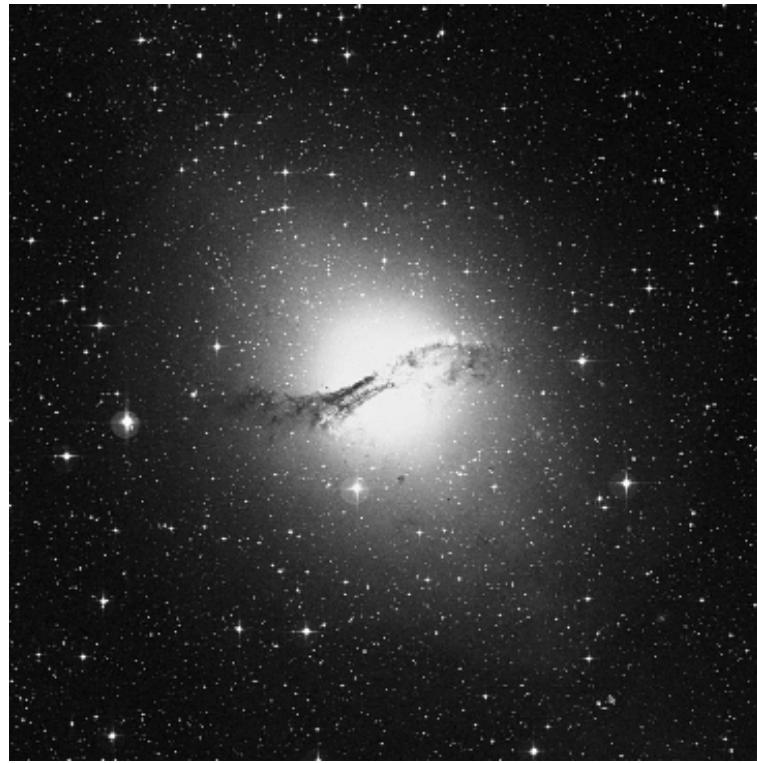
En este punto la teoría y la implementación de los diferentes algoritmos se pasan por diversas pruebas, que medirán su fidelidad, consistencia y efectividad, para que así astrónomos o investigadores, con toda confianza puedan explotar la información que tienen, automatizando las acciones y optimizando el tiempo dedicado al análisis.

Las pruebas son llevadas a cabo en una computadora portátil marca Apple, modelo Macbook Air, la cual cuenta con un procesador de 1.6 Ghz Intel Core i5, Memoria DDR3 de 2 GB y 1333 MHz, Gráficos Intel HD Gráficos 3000 288 MB y sistema operativo Maverick versión 10.9.3.

En primera instancia se prueban uno a uno los módulos testeando si realizan correctamente lo esperado y a su vez los tiempos de respuesta. En cada etapa del registro se ingresan diferentes parámetros de entrada y su consistencia en ellos. Los módulos están separados en dos tipos: Módulos superiores, que corresponden a todos aquellos que son protagonistas en las etapas del registro y módulos inferiores que realizan tareas de cálculo que colaboran con los módulos superiores.

### 4.1 MÓDULOS SUPERIORES

En esta etapa se probarán los módulos por separado, midiendo su efectividad, consistencia de la información y el tiempo de procesamiento para diferentes cantidades de imágenes. Se irán mostrando tablas comparativas en cada tipo de algoritmo y los resultados visuales obtenidos. Para poder visualizar los resultados de cada imagen, se ha utilizado DS9, software que permite entre muchas cosas ver una imagen. La imagen de a continuación corresponde a la Galaxia Lenticular NGC 5128, la cual fue capturada por el Telescopio Schmidt del Reino Unido, tiene dos dimensiones las cuales son de 1095 x 1095, esta será utilizada para las diferentes pruebas llevadas a cabo.



*Figura 4-1: Imagen de prueba de la galaxia NGC 5128*

#### *4.1.1 Recortar*

Módulo que tiene como entrada un vector de coordenadas (x,y), las cuales representan el borde del objeto elíptico y la matriz de información de la imagen que contiene el objeto. El sistema recorre punto a punto determinando el mayor y menor valor en ‘x’ y en ‘y’, los cuales definen el límite de la zona en donde se encuentra el objeto. Las imágenes de a continuación, figura 4-2, 4-3 y 4-4, representan una imagen de forma matricial la cual posee un objeto elíptico.

0,0	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9	0,10
1,0	1,1	1,2	1,3	1,4	1,5	1,6	1,7	1,8	1,9	1,10
2,0	2,1	2,2	2,3	2,4	2,5	2,6	2,7	2,8	2,9	2,10
3,0	3,1	3,2	3,3	3,4	3,5	3,6	3,7	3,8	3,9	3,10
4,0	4,1	4,2	4,3	4,4	4,5	4,6	4,7	4,8	4,9	4,10
5,0	5,1	5,2	5,3	5,4	5,5	5,6	5,7	5,8	5,9	5,10
6,0	6,1	6,2	6,3	6,4	6,5	6,6	6,7	6,8	6,9	6,10
7,0	7,1	7,2	7,3	7,4	7,5	7,6	7,7	7,8	7,9	7,10
8,0	8,1	8,2	8,3	8,4	8,5	8,6	8,7	8,8	8,9	8,10
9,0	9,1	9,2	9,3	9,4	9,5	9,6	9,7	9,8	9,9	9,10
10,0	10,1	10,2	10,3	10,4	105	10,6	10,7	10,8	10,9	10,10

Figura 4-2: Imagen de prueba con objeto elíptico

0,0	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9	0,10
1,0	1,1	1,2	1,3	1,4	1,5	1,6	1,7	1,8	1,9	1,10
2,0	2,1	2,2	2,3	2,4	2,5	2,6	2,7	2,8	2,9	2,10
3,0	3,1	3,2	3,3	3,4	3,5	3,6	3,7	3,8	3,9	3,10
4,0	4,1	4,2	4,3	4,4	4,5	4,6	4,7	4,8	4,9	4,10
5,0	5,1	5,2	5,3	5,4	5,5	5,6	5,7	5,8	5,9	5,10
6,0	6,1	6,2	6,3	6,4	6,5	6,6	6,7	6,8	6,9	6,10
7,0	7,1	7,2	7,3	7,4	7,5	7,6	7,7	7,8	7,9	7,10
8,0	8,1	8,2	8,3	8,4	8,5	8,6	8,7	8,8	8,9	8,10
9,0	9,1	9,2	9,3	9,4	9,5	9,6	9,7	9,8	9,9	9,10
10,0	10,1	10,2	10,3	10,4	105	10,6	10,7	10,8	10,9	10,10

Figura 4-3: Imagen de prueba con objeto elíptico

0,0	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9	0,10
1,0	1,1	1,2	1,3	1,4	1,5	1,6	1,7	1,8	1,9	1,10
2,0	2,1	2,2	2,3	2,4	2,5	2,6	2,7	2,8	2,9	2,10
3,0	3,1	3,2	3,3	3,4	3,5	3,6	3,7	3,8	3,9	3,10
4,0	4,1	4,2	4,3	4,4	4,5	4,6	4,7	4,8	4,9	4,10
5,0	5,1	5,2	5,3	5,4	5,5	5,6	5,7	5,8	5,9	5,10
6,0	6,1	6,2	6,3	6,4	6,5	6,6	6,7	6,8	6,9	6,10
7,0	7,1	7,2	7,3	7,4	7,5	7,6	7,7	7,8	7,9	7,10
8,0	8,1	8,2	8,3	8,4	8,5	8,6	8,7	8,8	8,9	8,10
9,0	9,1	9,2	9,3	9,4	9,5	9,6	9,7	9,8	9,9	9,10
10,0	10,1	10,2	10,3	10,4	10,5	10,6	10,7	10,8	10,9	10,10

Figura 4-4: Imagen de prueba con objeto elíptico

La tabla de a continuación detalla la salida obtenida al ingresar las matrices anteriormente mas el vector de coordenadas.

Tabla 4.1: Recorte de una imagen

Matriz de entrada	x mayor	x menor	y mayor	y menor	Tiempo (seg)
Figura 4-2	8	1	7	1	0.0001530
Figura 4-3	9	5	10	4	0.0001630
Figura 4-4	10	0	6	4	0.0001177

El tiempo de procesamiento es bastante rápido en general, si en una matriz de 10x10 tarda en promedio 0.00014456 segundos, comportamiento casi constante, en una matriz de 2000 x 2000, siguiendo el mismo comportamiento lineal se puede estimar que no tardaría mas de 1 segundo en hacer una extracción.

Las imágenes de a continuación, figura 4-4, 4-5 y 4-6 representan la matriz de entrada ingresada al sistema y su respectiva salida.

```
Matriz de entrada:  
-----  
[[ 0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0 ],  
 [ 0,  0,  0,  0,  0,  7,  7,  7,  0,  0,  0,  0,  0,  0,  0 ],  
 [ 0,  0,  0,  7,  7,  0,  0,  0,  7,  0,  0,  0,  0,  0,  0 ],  
 [ 0,  0,  7,  0,  0,  0,  0,  0,  7,  0,  0,  0,  0,  0,  0 ],  
 [ 0,  7,  0,  0,  0,  0,  0,  7,  0,  0,  0,  0,  0,  0,  0 ],  
 [ 0,  7,  0,  0,  0,  0,  7,  0,  0,  0,  0,  0,  0,  0,  0 ],  
 [ 0,  7,  0,  0,  0,  0,  0,  7,  0,  0,  0,  0,  0,  0,  0 ],  
 [ 0,  7,  0,  7,  7,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0 ],  
 [ 0,  7,  7,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0 ],  
 [ 0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0 ],  
 [ 0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0 ]]  
  
Matriz resultante:  
-----  
[[ 0,  0,  0,  0,  7,  7,  7,  7,  7,  7,  7,  7,  7,  7,  7,  7 ],  
 [ 0,  0,  7,  7,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0 ],  
 [ 0,  7,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0 ],  
 [ 7,  7,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0 ],  
 [ 7,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0 ],  
 [ 7,  0,  0,  0,  0,  7,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0 ],  
 [ 7,  0,  7,  7,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0 ],  
 [ 7,  7,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0 ]]
```

*Figura 4-5: Extracción de objeto*

```
Rodrigos-MacBook-Air:site-packages rodrigo$ ./prueba.py

Matriz de entrada:
-----
[[ 0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.],
 [ 0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.],
 [ 0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.],
 [ 0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.],
 [ 0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.],
 [ 0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.],
 [ 0.,  0.,  0.,  0.,  0.,  0.,  7.,  7.,  7.,  7.,  7.,  0.],
 [ 0.,  0.,  0.,  0.,  0.,  7.,  0.,  0.,  0.,  0.,  0.,  7.],
 [ 0.,  0.,  0.,  0.,  7.,  0.,  0.,  0.,  0.,  0.,  0.,  7.],
 [ 0.,  0.,  0.,  0.,  7.,  0.,  0.,  0.,  0.,  0.,  7.,  0.],
 [ 0.,  0.,  0.,  0.,  7.,  7.,  7.,  7.,  7.,  0.,  0.],
 [ 0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.]]]

Matriz resultante:
-----
[[ 0.,  0.,  7.,  7.,  7.,  7.,  0.],
 [ 0.,  7.,  0.,  0.,  0.,  0.,  7.],
 [ 7.,  0.,  0.,  0.,  0.,  0.,  7.],
 [ 7.,  0.,  0.,  0.,  0.,  7.,  0.],
 [ 0.,  7.,  7.,  7.,  7.,  0.,  0.]]]
```

*Figura 4-6: Extracción de objeto*

*Figura 4-7: Extracción de objeto*

### 4.1.2 Rotación

Modulo que tiene como entrada la matriz que se desea rotar, su dimensión, y el ángulo en el cual será rotada. El módulo retorna la matriz rotada en el ángulo respectivo en sentido antihorario.

*Tabla 4.2: Pruebas de rotación para una imagen*

Número de Iteraciones	Ángulo a rotar (grados)	Rota en el ángulo que se pide	Existe consistencia en el resultado	Tiempo de ejecución (seg)
1	5	Si	Si	15.4021849632
1	45	Si	Si	15.2303009033
1	73	Si	Si	15.9277729988
1	128	Si	Si	16.5094139576
5	90	Si	Si	61.0421690941
10	90	Si	Si	120.126850128
50	90	Si	Si	605.092447042
100	90	Si	Si	1225.95393801

Mirando la table se puede apreciar, que el tiempo de computo en procesar una rotación de 90° es mas rápido que los otros ángulo, un 12 segundos contra 15 que tarda una rotación en 5°, esto se puede entender dado que cuando la matriz es rotada en 90°\*n, con n pertenecientes a los cardinales, las coordenadas rotadas siempre caerán en valores enteros, por lo que no será necesario interpolar la información. Independiente de eso, en interpolar solo se invierte 3 segundos. No siempre será necesario rotar la imagen, por lo que se puede decir, que rotando una muestra de 100, el astrónomo invertirá aproximadamente 1225 segundos si el programa no necesita interpolar.

A continuación se muestran los resultados visuales para una rotación de 5, 45, 73 y 128 grados. Ver figura 4-8, 4-9, 4-10 y 4-11.

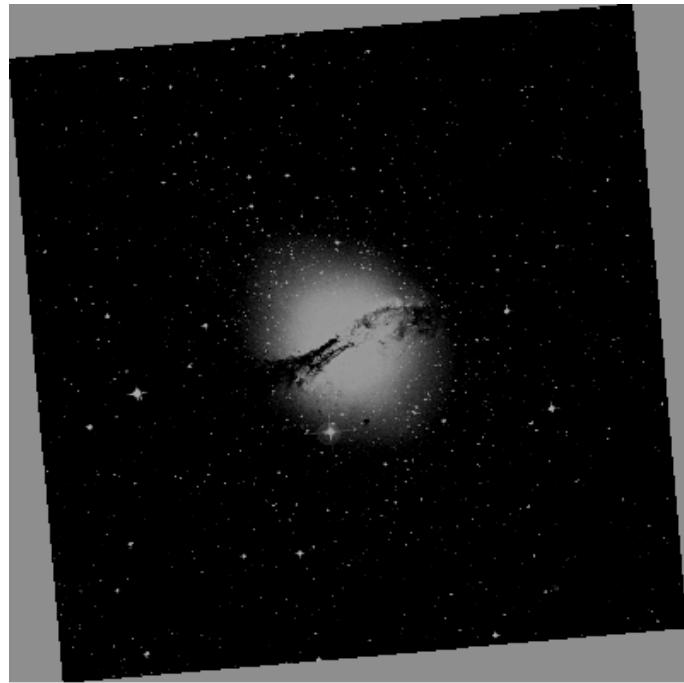


Figura 4-8: Rotación de una imagen en  $5^\circ$

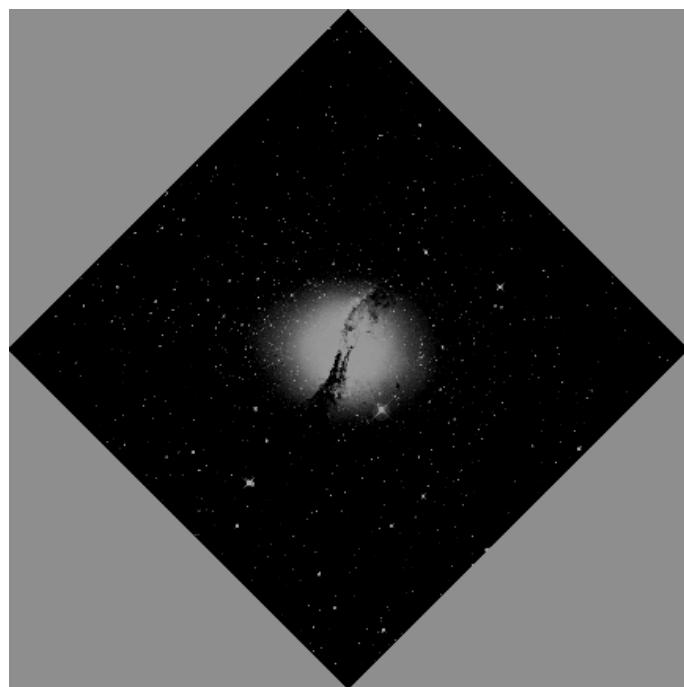


Figura 4-9: Rotación de una imagen en  $45^\circ$

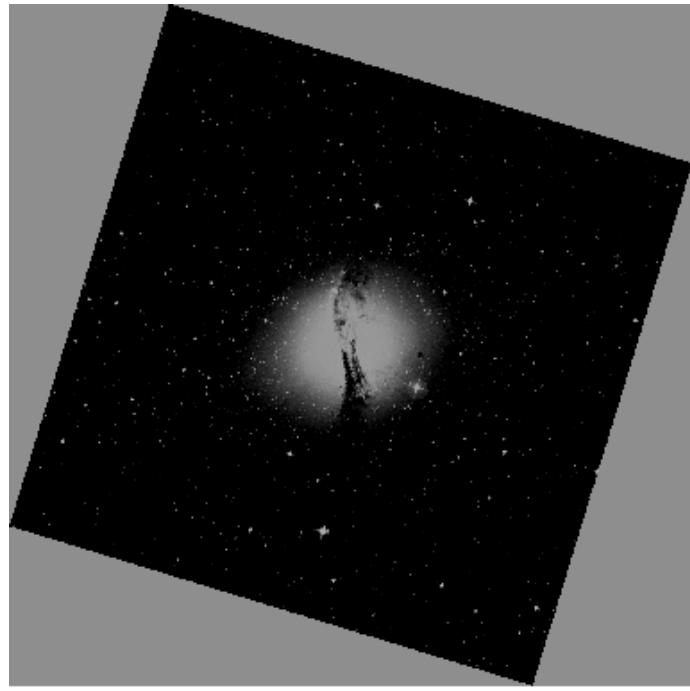


Figura 4-10: Rotación de una imagen en  $73^\circ$

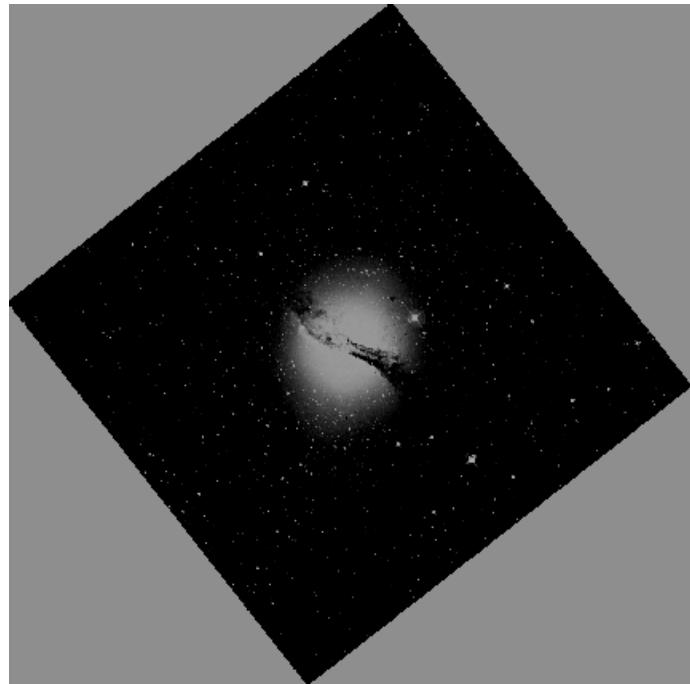


Figura 4-11: Rotación de una imagen en  $128^\circ$

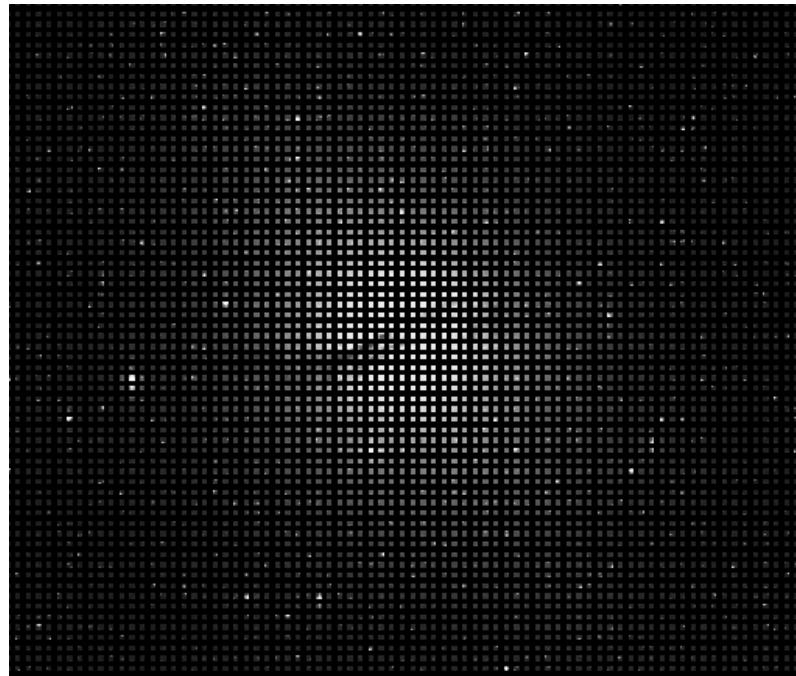
#### 4.1.3 Escalar

Modulo que tiene como entrada la matriz que se desea rotar, su dimensión, y la razón en que será escalada. Para notar la diferencia se mostrará la imagen escalada con y sin interpolación. Cuando termina la ejecución, el módulo retorna la matriz escalada.

*Tabla 4.3: Pruebas de escalamiento para una imagen*

Razón	Escalamiento + / -	Hay consistencia	Interpolación	Tiempo de ejecución
2	+	Si	Si	24.782819032
5	+	Si	Si	62.765192031
1/4	-	Si	No	12.748407125
1/5	-	Si	No	12.996289014

Observando la tabla se puede apreciar la diferencia tajante de cuando una imagen es ampliada o no, por ejemplo, cuando se amplía en una razón de dos se requieren casi 25 segundo para que se lleve a cabo, en cambio cuando la imagen se achica en una razón de  $\frac{1}{4}$ , tan solo se necesitan 12.75 segundos, esta diferencia surge dado que no es necesario crear información intermedia entre puntos. A medida que crece la taza de ampliación, es mas costoso el tiempo de procesamiento. La imagen de a continuación representa una ampliación para una razón de 2 sin interpolar, en ella se puede apreciar como los puntos al distribuirlos quedan sin información entre ellos.



*Figura 4-12: Escalamiento de imagen sin interpolación*

Si viéramos esta misma imagen mas de cerca, apreciaríamos que los puntos están uniformemente distribuidos, y a dos puntos de distancia, figura 4-13, hay un nuevo punto con información. Si la razón fuera 5, figura 4-14, los puntos estarían aún mas disperso.

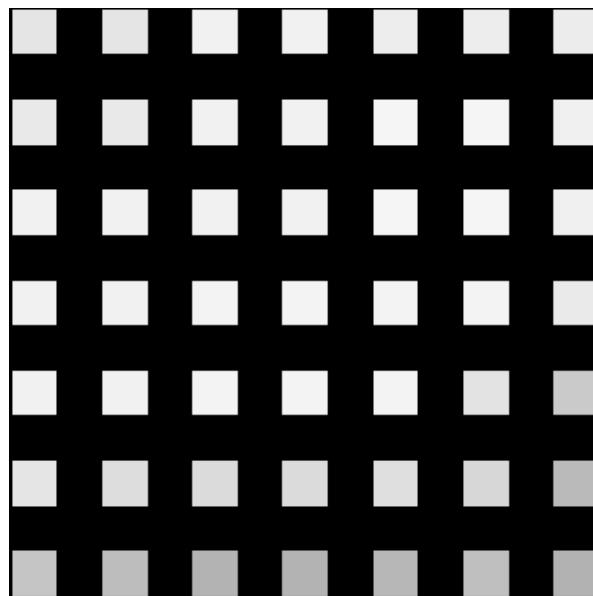


Figura 4-13: Escalamiento a razón de dos

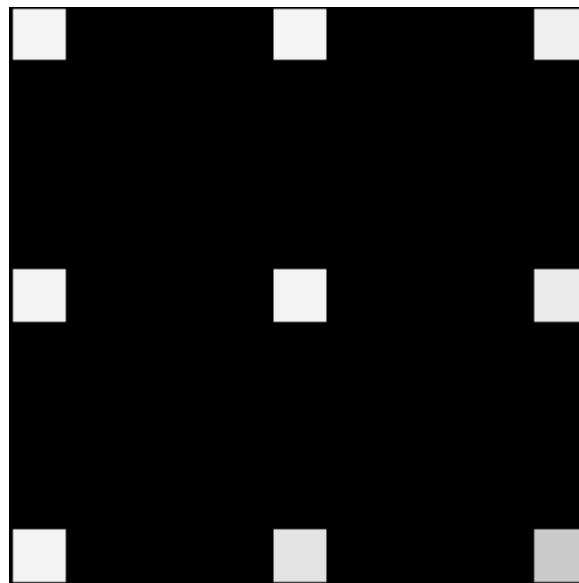


Figura 4-14: Escalamiento a razón de cinco

Cuando la imagen se agranda el algoritmo entra en una segunda fase, que consiste en completar la información vacía, la cual es la parte más costosa del algoritmo, ya que se invierte mínimo el 92% del tiempo, con una tendencia creciente muy marcada.

La tabla 4.3 muestra como influye en el tiempo de respuesta la interpolación. El tiempo sin interpolar es insignificante y casi invariante el tiempo de escalar.

*Tabla 4.0.4: Comparativa en tiempo de algoritmo con y sin interpolación*

Razón	Interpolación	Tiempo de ejecución
2	Si	24.782819032
2	No	2.1595578193
3	Si	35.017352819
3	No	2.1890161037
5	Si	62.765192031
5	No	2.3549020290

A continuación, figura 4-15, se puede apreciar el resultado de la imagen escalada e interpolada.

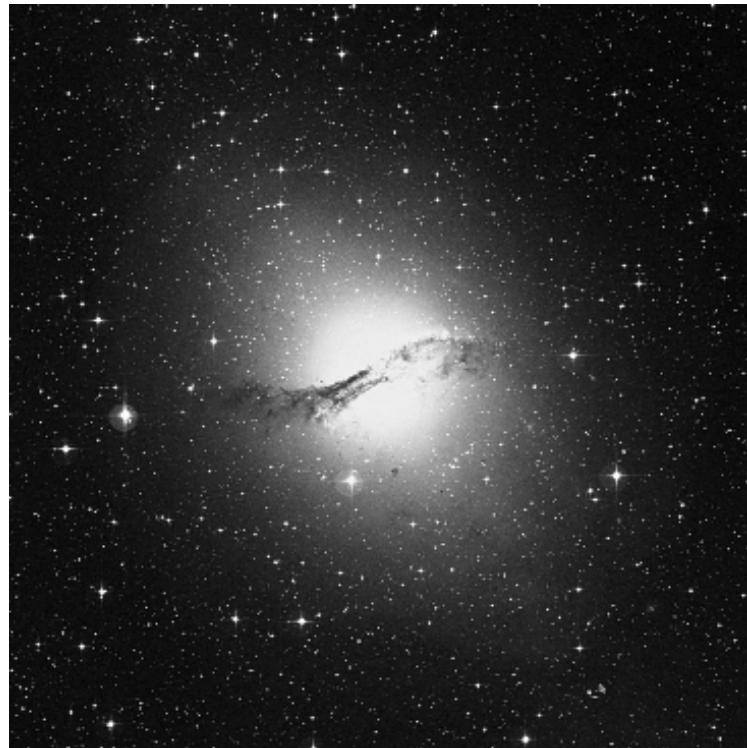


Figura 4-15: Imagen resultante al agrandar una imagen

#### 4.1.4 Alinear

El módulo alinear tiene como entrada la coordenada del punto central del objeto en la imagen de referencia, del objeto de la imagen actual, la dimensión de la imagen de referencia y de la imagen actual, y la matriz de información de la imagen actual. Este módulo retorna una matriz de tamaño igual a la de referencia, con el objeto de la imagen actual centrado en el mismo punto que el objeto de referencia. Este proceso no requiere de mucho tiempo de ejecución ya que básicamente es trasladar puntos.

Tabla 4.0.5: Alineación de un punto en torno a coordenada (100, 100)

Referencia	Actual	Referencia	Actual	Tiempo de	Hay
------------	--------	------------	--------	-----------	-----

$(x_i, y_j)$	$(x_i, y_j)$	$(x, y)$	$(x, y)$	Ejecución	Consistencia
(100,100)	(499,499)	(500,500)	(1095,1095)	3.99397	Si



Figura 4-16: Alineación de un objeto en torno al punto (100,100)

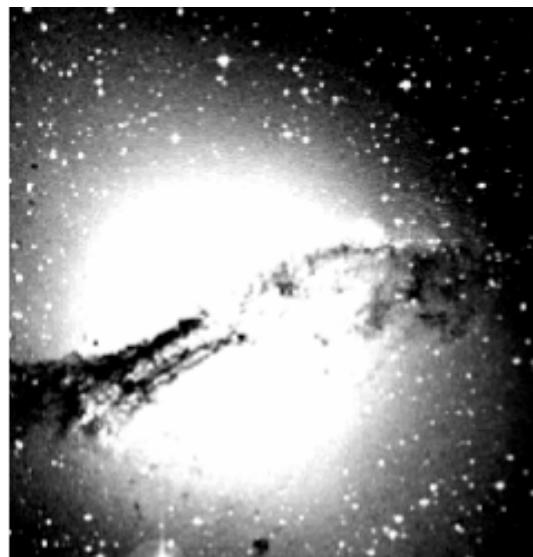
#### 4.1.5 Deproyectar

El siguiente módulo tiene como entrada el punto  $(x, y)$  donde se llevará el centro del objeto actual, la dimensión de la imagen de referencia, las coordenadas del borde del objeto que se esta analizando, su matriz de información y el radio.

Tabla 4.0.6: Tiempo para la deproyección de un objeto

Radio	Coordenada Elipse	Coordenada Circunferencia	Tiempo de ejecución
250	(7,4)	(43,246)	27.782819032

El efecto logrado en la imagen no ha sido muy notorio, dado que el objeto ya poseía una forma circular. El tiempo de ejecución no es muy alto a pesar de la cantidad de puntos que se deben interpolar. La figura 4-10 muestra como una serie de puntos fueron trasladados una distancia igual al radio del objeto, en sentido parametrico de la circunferencia.



*Figura 4-17: Deproyección de un objeto*

## 4.2 MÓDULOS INFERIORES

Esta etapa de pruebas consiste en validar el sistema de cómputo oculto dentro del sistema, que colabora con los módulos superiores entregando información vital de los objetos en las variadas imágenes antes de realizar cualquier transformación geométrica en el plano.

#### *4.2.1 Distancia entre puntos*

Función que tiene como entrada dos coordenadas ( $X_1, Y_1$ ) y ( $X_2, Y_2$ ), de las cuales calcula su distancia lineal. Se prueban diferentes valores para ‘X’ e ‘Y’. Por cada resultado se tomará el tiempo de cálculo.

La tabla 4.7, para las diferentes entradas de (X,Y) se muestra su distancia, y el tiempo en segundo que tarda en calcularla.

*Tabla 4.0.7: Medida de la distancia entre dos puntos*

$X_1$	$Y_1$	$X_2$	$Y_2$	Distancia	Tiempo (seg)
2	1	3	2	1.41421356237	2.78949737549
3	4	5	4.3	2.02237484162	5.00679016113
6.7	8	7	6	2.02237484162	4.05311584473
21.9	22	9	8	19.0370691021	3.81469726562
23.23	1	0	7	23.9923508644	3.09944152832
4.4	4.4	20.20	20.20	22.3445742855	3.09944152832
12	3	1	5	11.1803398875	3.09944152832
6	9	1	3	7.81024967591	2.86102294922
23.12	23.12	23.50	23.50	0.537401153702	3.09944152832
10	30	10	60	30.0	1.90734863281

#### *4.2.2 Medida del eje mayor*

Algoritmo que a partir de un vector que se compone de todas las coordenadas que son parte del borde del objeto elíptico, determina cual es la mayor distancia entre ellas. El vector se recibe de la siguiente manera, tabla 4.8:

Tabla 4.0.8: Ejemplo de vector de puntos

X <sub>1</sub>	Y <sub>1</sub>	X <sub>2</sub>	Y <sub>2</sub>	X <sub>3</sub>	Y <sub>3</sub>	...	...	X <sub>n</sub>	Y <sub>n</sub>
----------------	----------------	----------------	----------------	----------------	----------------	-----	-----	----------------	----------------

A continuación probaremos tres objetos elípticos, figura 4-18, figura 4-19 y figura 4-20. La tabla

0,0	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9	0,10
1,0	1,1	1,2	1,3	1,4	1,5	1,6	1,7	1,8	1,9	1,10
2,0	2,1	2,2	2,3	2,4	2,5	2,6	2,7	2,8	2,9	2,10
3,0	3,1	3,2	3,3	3,4	3,5	3,6	3,7	3,8	3,9	3,10
4,0	4,1	4,2	4,3	4,4	4,5	4,6	4,7	4,8	4,9	4,10
5,0	5,1	5,2	5,3	5,4	5,5	5,6	5,7	5,8	5,9	5,10
6,0	6,1	6,2	6,3	6,4	6,5	6,6	6,7	6,8	6,9	6,10
7,0	7,1	7,2	7,3	7,4	7,5	7,6	7,7	7,8	7,9	7,10
8,0	8,1	8,2	8,3	8,4	8,5	8,6	8,7	8,8	8,9	8,10
9,0	9,1	9,2	9,3	9,4	9,5	9,6	9,7	9,8	9,9	9,10
10,0	10,1	10,2	10,3	10,4	105	10,6	10,7	10,8	10,9	10,10

Figura 4-18: Imagen de prueba con objeto elíptico

0,0	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9	0,10
1,0	1,1	1,2	1,3	1,4	1,5	1,6	1,7	1,8	1,9	1,10
2,0	2,1	2,2	2,3	2,4	2,5	2,6	2,7	2,8	2,9	2,10
3,0	3,1	3,2	3,3	3,4	3,5	3,6	3,7	3,8	3,9	3,10
4,0	4,1	4,2	4,3	4,4	4,5	4,6	4,7	4,8	4,9	4,10
5,0	5,1	5,2	5,3	5,4	5,5	5,6	5,7	5,8	5,9	5,10
6,0	6,1	6,2	6,3	6,4	6,5	6,6	6,7	6,8	6,9	6,10
7,0	7,1	7,2	7,3	7,4	7,5	7,6	7,7	7,8	7,9	7,10
8,0	8,1	8,2	8,3	8,4	8,5	8,6	8,7	8,8	8,9	8,10
9,0	9,1	9,2	9,3	9,4	9,5	9,6	9,7	9,8	9,9	9,10
10,0	10,1	10,2	10,3	10,4	105	10,6	10,7	10,8	10,9	10,10

Figura 4-19: Imagen de prueba con objeto elíptico

0,0	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9	0,10
1,0	1,1	1,2	1,3	1,4	1,5	1,6	1,7	1,8	1,9	1,10
2,0	2,1	2,2	2,3	2,4	2,5	2,6	2,7	2,8	2,9	2,10
3,0	3,1	3,2	3,3	3,4	3,5	3,6	3,7	3,8	3,9	3,10
4,0	4,1	4,2	4,3	4,4	4,5	4,6	4,7	4,8	4,9	4,10
5,0	5,1	5,2	5,3	5,4	5,5	5,6	5,7	5,8	5,9	5,10
6,0	6,1	6,2	6,3	6,4	6,5	6,6	6,7	6,8	6,9	6,10
7,0	7,1	7,2	7,3	7,4	7,5	7,6	7,7	7,8	7,9	7,10
8,0	8,1	8,2	8,3	8,4	8,5	8,6	8,7	8,8	8,9	8,10
9,0	9,1	9,2	9,3	9,4	9,5	9,6	9,7	9,8	9,9	9,10
10,0	10,1	10,2	10,3	10,4	10,5	10,6	10,7	10,8	10,9	10,10

Figura 4-20: Imagen de prueba con objeto elíptico

Tabla 4.0.9: Búsqueda del eje mayor

Vector de entrada	Coordenada menor	Coordenada mayor	Distancia	Tiempo (seg)
Figura 4-11	(1 , 7)	(8 , 1)	9.21954445729	0.00046181678772
Figura 4-12	(6 , 1)	(8 , 4)	6.32455532034	0.000265121459961
Figura 4-13	(0 , 5)	(10 , 5)	10.0	0.000530958175659

#### 4.2.3 Ángulo a rotar

Función que recibe como parámetros de entrada las coordenadas del borde de la ellipse, retornando el complemento del ángulo de la pendiente.

*Tabla 4.0.10: Determinación del ángulo complementario*

Figura de referencia	Coordenada menor	Coordenada mayor	Ángulo complementario	Tiempo (seg)
Figura 4-11	(1 , 7)	(8 , 1)	139.398705355	0.00004100799
Figura 4-12	(6 , 1)	(8 , 4)	123.690067526	0,00003.790851
Figura 4-13	(0 , 5)	(10 , 5)	90	0,00001.096725

#### *4.2.4 Centro geométrico*

Función que recibe como parámetros de entrada el tamaño del eje mayor del objeto, sus coordenadas y la pendiente de la recta, entregando como salida, las coordenadas del centro del objeto.

*Tabla 0.11: Obtención del centro geométrico del objeto elíptico*

Figura de referencia	Coordenada menor	Coordenada mayor	Tamaño del eje	Pendiente	Centro geométrico	Tiempo
Figura 4-11	(1 , 7)	(8 , 1)	9.21954445729	40.601294645	(4 , 10)	0.000431537628
Figura 4-12	(6 , 1)	(8 , 4)	6.32455532034	56.309932474	(8 , 4)	7.009506225593
Figura 4-13	(0 , 5)	(10 , 5)	10.0	90	(5 , 5)	0.000005.00679

## CAPITULO 5. CONCLUSIONES

A lo largo de este documento se ha visto como se aplica el registro de imágenes a objetos astronómicos y la utilidad junto con el valor que genera su uso en investigaciones en diferentes áreas mas allá de las ciencias del universo.

En el capítulo número uno se aprecia la importancia que ha tenido el registro en el ámbito de la medicina nuclear, ayudando a observar, caracterizar y definir alteraciones físicas a nivel cerebral que presentan pacientes, permitiendo comparar resultados en el tiempo y así definir que tratamiento es el mas adecuado.

También se habla de un proyecto llamado CARTOSUR donde se mezclaron imágenes en diferentes longitudes de onda. Aquí se aprecia que si bien el registro de imágenes es una técnica que remota desde los inicios de la imagen digital, el incursionar en nuevas técnicas, como mezclar imágenes de radio con imágenes ópticas, es la prueba de lo amplio y extenso que puede ser el uso del registro.

La necesidad existente en poseer mejores herramientas de análisis que apoyen la toma de decisiones es lo que motiva la exploración, experimentación y desarrollo de técnicas que procesen la información.

Atacama *Large Milimetric/Submilimetric Array*, actualmente el radio telescopio más grande del mundo, genera tanta información, alrededor de 750Gb de información por día, que los astrónomos requieren contar con una plataforma abierta para acceder virtualmente a todas las observaciones realizadas (La Tercera, 2013). Esta plataforma virtual nombrada ‘ChiVO’, necesita contar con herramientas que procesen de manera inteligente los datos capturados, generando información de calidad, útil para los astrónomos y comunidad en general. En el caso del registro de imágenes, es una herramienta avanzada de mucha utilidad tanto para ALMA como otros observatorios, dando un valor agregado a la información obtenida. Esto es de vital importancia, ya que el registro no es tan solo disminuir el ruido o aumentar la calidad de una imagen, sino las variadas posibilidades de análisis que permite realizar, por ejemplo: estudiar la

composición u estructura de una galaxia en particular a partir de la información obtenida en una observación; determinar la edad que posee un cúmulo globular; analizar la evolución o cambios que ha tenido un cuerpo en el tiempo.

En general, los algoritmos implementados a lo largo de las etapas del registro de imágenes son operaciones sencillas sobre matrices y transformaciones geométricas en el plano, por lo que en la bibliografía existente no hay mucha variación. El registro básicamente es una aplicación que hace corresponder a cada punto  $P$  de coordenadas  $(x,y)$  del plano, otro punto  $P'$  de coordenadas  $(x',y')$  del mismo plano. Si bien la distribución de coordenadas en el plano cartesiano no poseen exactamente la misma estructura en que se lee una matriz computacionalmente, la geometría aplicada varía levemente, solo hay que tener presente que lo que habitualmente leemos en un plano cartesiano como abscisa y luego ordenada, aquí es al revés, se lee como fila columna, y los ángulos formados en los ejes van en sentido horario. Asumiendo lo anterior, ya es más sencillo trabajar con la información.

Otro aspecto fundamental en el registro de imágenes, es la forma en que completamos la información cuando se realiza una modificación, una matriz computacional no posee puntos densos, es decir, al subdividir el espacio entre dos coordenadas, no necesariamente existirá otra coordenada entera y por otro lado al trasladar la información existente, puede que queden puntos sin información. En estos casos la interpolación es imprescindible para completar la información faltante. Si bien cada autor tiene su forma, la idea base es completar a partir de los puntos vecinos el punto que está ausente, algunos ocupan los 4 vecinos más cercanos, otros 16, pero depende del fin que se quiere conseguir. En el caso de los algoritmos implementados en este trabajo tienen una mirada distinta. Por un lado, cuando ocurren situaciones en que la información cae sobre un punto ‘decimal’, por ejemplo  $(2.4, 3)$ , el sistema decide dejar la información repartida entre el punto  $(2.0, 3)$  y  $(3.0, 3)$ . Este esquema es mucho más sencillo, por un lado asegura que todos los puntos quedarán uniformemente distribuidos y por otro lado no es invasivo, es un trabajo más sutil. El segundo caso ocurre cuando una imagen es ampliada, la cantidad de información vacía intermedia que queda, es

proporcional a la razón de escala utilizada. Mirar los objetos mas cercanos para completar el punto faltante es algo tedioso, ya que desde un punto vacío los vecinos no están equidistantes, dada esta situación los vecinos colaboran para completar lo que entre ellos falta, esto es mucho mas sencillo, ya que desde un punto con información a otro punto con información la distancia es la razón. La cantidad de vecinos a comunicar son en grupos de cuatro, de los cuales, dos de ellos se ocuparán para el conformar el grupo siguiente.

Un aspecto importante dentro del proceso de ajuste, es contar con el conjunto extra de funciones que entregan la información necesaria para que el ajuste tome valor, es decir, ya no tan solo son herramientas que extraen, rotan, escalan, alinean o deproyectan un objeto, sino que aportan la información necesaria para determinar en que grado conviene rotar, en que punto alinear, que tanto agrandar o achicar, o donde desplazar los puntos para que queden deproyectado. Con esto ya se marca una diferencia más a una herramienta de registro convencional.

A lo largo de las pruebas realizadas para medir la eficacia de los algoritmos, se puede decir, que fueron satisfactoriamente logradas, pero no son lo mas óptimo al momento de operar en un computador de escritorio. El tiempo requerido para realizar algunas operaciones como la rotación de un objeto, registro una duración de alrededor de 20 minutos sobre un conjunto de 100 imágenes de 1095 x 1095 pixeles, sin contar el tiempo que requeriría para terminar con el resto.

Si bien el objetivo de la tesis de realizar un módulo en Python que realice el registro de imágenes astronómicas se cumple, es el puntapié inicial para mejorar, optimizar y desarrollar todo lo realizado en una plataforma virtual como es el Observatorio Virtual Chileno. Adicionalmente se puede expandir las funcionalidades, y realizar aplicaciones en otros tipos de objetos extrasolares, como por ejemplo, nebulosas, discos de acreción, exoplanetas, etcétera, como también trabajar con imágenes en otras dimensiones, por ejemplo, líneas espectrales o cubos de información. La idea es apuntar a crear o mejorar herramientas que potencien aún mas la información,

colaborando en el desarrollo científico y así permitir a los investigadores invertir su tiempo en analizar o estudiar la información, mas que en como procesarla.

Lo mas hermoso de todo lo realizado, es ver como un código toma vida y se plasma como un sentimiento en la tarea de un astrónomo que busca cada día descubrir y conocer lo que hay mas allá de sus ojos.

## 6. Bibliografía

- [1] ALMA. (n.d.). El Universo de ALMA. Santiago, Chile.
- [2] Astropy. (n.d.). Edit a FITS header. Retrieved 12 de 07 de 2013 from Astropy Tutorials: <http://www.astropy.org/astropy-tutorials/FITS-header.html>
- [3] EXPLORA CONICYT. (2009). Noticias del Universo. Expoastronomía , 21.
- [4] Fundación Polar. Matrices y Transformaciones.
- [5] Garay, G. (n.d.). Introduction to Radio-Interferometry. Departamento de Astronomía, Universidad de Chile . Santiago, Chile.
- [6] Greisen, E. W. (2012). AIPS FITS File Format .
- [7] Instituto de Astrofísica de Andalucía. (2000). Diez años de investigación con el telescopio espacial HUBBLE. Información y Actualidad Astronómica , 20.
- [8] Heisig, M. (2007). Cerca del Cielo. Santiago: ESO.
- [9] J., L. A. (1995). Fusión de Imágenes del Satélite Landsat Tematic Mapper y de Radar de Apertura Sintética Aerotransportado para Facilitar la Delineación del Litoral. USA: Louisiana State University.
- [10] La Tercera. (12 de 05 de 2013). Datos de Alma serán parte de la Alianza Internacional de Observatorios Virtuales .
- [11] Lewis, P. J., Siegel, A., Siegel, A. M., Studholme, C., Sojko- va, J., Roberts, D. W., et al. (2000). Does performing image registration and subtraction in ictal brain SPECT help localize neocortical seizures? . USA: Department of Radiology.
- [12] Miguel, J. (2003). INTEGRACIÓN DE ORTOIMÁGENES DE RADAR DEL PROYECTO CARTOSUR CON DATOS ÓPTICOS PROVENIENTES DE LAS IMÁGENES LANDSAT CON FINES DE INTERPRETACIÓN VISUAL . Venezuela: Instituto Geográfico de Venezuela Simón Bolívar.
- [13] Python. (n.d.). Mathematical functions. Retrieved 12 de 05 de 2013 from Numeric and Mathematical Modules : <https://docs.python.org/2/library/math.html>

- [14] Rogan, J., & Muñoz, V. (2012). Programación y Métodos numéricos. Chile: Universidad de Chile .

## **7. Anexos**

### **GALAXIAS**

Todo sistema independiente de estrellas situado fuero de la Vía Láctea, se le atribuye el nombre de galaxia. Las galaxias que poseen un diámetro de entre 6.000 a 170.000 años luz, contienen entre 3.000 millones a 1 billón de estrellas, así como una cantidad de gas y polvo que representa el 10% de su masa total.

Las galaxias se clasifican, atendiendo a su morfología, en tres grandes grupos:

- a. Galaxias elípticas: Este conjunto de galaxias representan el 20% de las galaxias conocidas. Poseen forma esferoidal y carecen por completo de estructura interna definida. Además, no contienen apenas materia interestelar y las estrellas que la componen son viejas y se encuentran en estados muy avanzados de evolución.
- b. Galaxias espirales: Estas están divididas en dos grupos, el de las espirales normales y el de las espirales barradas. Las primeras, que representan el 75% de todas las galaxias, tienen forma de lente aplanada y están dotadas de 2 a 3 brazos que parten de su centro, así como de gran cantidad de materia interestelar y muchas estrellas brillantes y jóvenes. La Vía Láctea pertenece este tipo. Por otro lado, las espirales barradas, disponen de dos brazos rectos opuestos que parten de su centro y de cuyos extremos surgen casi perpendicularmente sus brazos espirales, que en algunos casos se desarrollan hasta formar un círculo alrededor del núcleo.
- c. Galaxias irregulares: Representando el 5% del total, su forma no presenta simetría alguna, siendo su aspecto y forma muy variables. Por lo general son pequeñas y contienen gran cantidad de materia interestelar. Se subdividen en las de tipo I, que pueden resolverse en estrellas, nebulosas o cúmulos, y las

de tipo II, que no admiten este tipo de resolución y parecen ser completamente amorfas.

## APILADO DE IMÁGENES. APLICACIÓN POST REGISTRO

Las imágenes astronómicas obtenidas por cualquier telescopio se ven limitadas por varios factores. Entre ellos, las condiciones atmosféricas, la resolución del telescopio o la razón señal/ruido en cada pixel de la imagen. La señal captada en cada pixel, depende del flujo de la fuente astronómica o el tiempo de integración. El ruido existente en cada pixel depende netamente del instrumento de observación. Por ejemplo, en un circuito de carga acoplada (CCD), el ruido se genera por la agitación térmica de los portadores de carga (conductores), por la lectura del CCD y el ‘fondo’ de emisión del cielo y/o telescopio. En general, un mayor tiempo de integración en una fuente óptica proporciona imágenes con una señal más expuesta al ruido.

En la obtención de imágenes de radio el ruido se genera principalmente por la agitación térmica de los receptores ubicados en cada telescopio. Aquí, interacciones más largas dan lugar a imágenes con mejores niveles de señal respecto al ruido, debido a que la integración ya produce una disminución.

Para un telescopio como ALMA, la relación señal/ruido es inversamente proporcional a la raíz cuadrada del producto del ancho de banda y el tiempo de integración total. Es decir, para un ancho de banda fijo, la duplicación de la señal al ruido requiere cuatro veces el tiempo de integración.

En el caso del observatorio ALMA, en muchos casos, no se detectan objetos astronómicos individuales en una imagen, es decir, la señal de éste objetivo es más pequeño que el ruido (o tres veces el ruido) en la imagen. Sin embargo, si uno tiene una muestra de los objetivos (por ejemplo, 1.000 fuentes) con posiciones conocidas con precisión, los cuales están en la imagen, se podrían extraer sub-imágenes centradas en cada posición y la media de todas las 1000 sub-imágenes. La imagen promedio es equivalente a una imagen con 1000 veces el tiempo de integración, es decir, el ruido en la imagen se reduce por un factor de la raíz cuadrada de 1000 o aproximadamente 30.

Por tanto, es más probable que la imagen promediada muestre una detección de señales muy débiles que requerirán de varios días o meses de observación (la señal en la imagen promediada es la señal media de las 1000 fuentes, mientras que el ruido es 30 veces menor que el ruido en cualquiera de las imágenes en las 1000 fuentes). Ésta técnica, conocida como apilamiento, es una herramienta muy potente y se utiliza por ejemplo, en imágenes de rayos X, formación de imágenes ópticas e imágenes de radio. El apilamiento da una idea de las características medias de una muestra, algo muy útil para el análisis.

Como el archivo de ALMA sigue creciendo, una fracción cada vez mayor del cielo está siendo capturada. Con una muestra suficientemente amplia de objetos astronómicos, por ejemplo, un millón de cuásares con posiciones precisas del Sloan Digital Sky Survey, SDSS, se encuentra que muchos objetos de la muestra estarán en áreas que ya han sido observadas por ALMA. Incluso, si no se detectan de forma individual, se puede apilar todos los objetos que han sido vistos por ALMA y obtener un flujo promedio para ésta muestra. Si la muestra también tiene desplazamientos al rojo, el apilamiento se puede realizar en el espacio de velocidades con el fin de obtener las propiedades medias de cualquier línea de emisión dada.

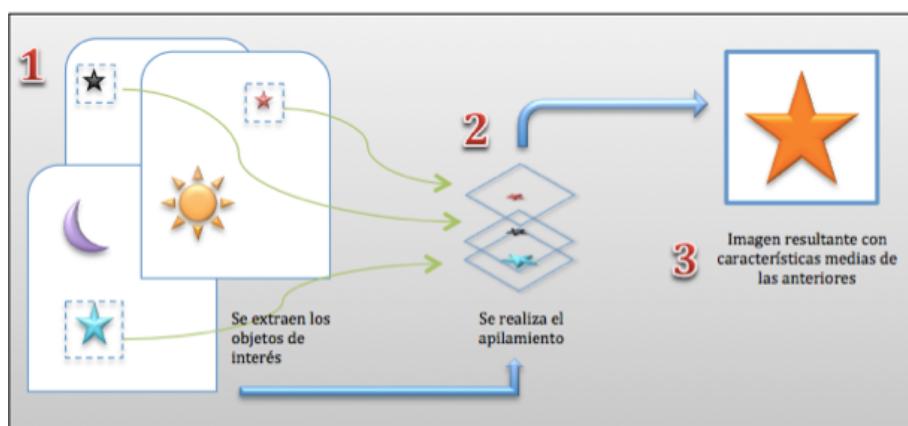


Figura 7-1: Proceso de apilado de imágenes a partir de un conjunto de imágenes normalizadas

## INICIOS DEL APILADO DE IMÁGENES

Todas las observaciones de imágenes astronómicas poseen un umbral de sensibilidad por debajo del cual los objetos no son detectables. Si uno tiene razones para creer que en esa posición del cielo puede existir algo, basándose en otras observaciones, se puede aplicar apilamiento a una zona delimitada a un cierto rango de longitudes de onda, pudiendo variar dicha longitud, y así conseguir el flujo medio de emisiones.

En una aplicación inicial de esta técnica, Caillault y Helfand, detectaron el flujo medio de rayos X de estrellas G en Pleiades, utilizándola para determinar la edad respecto al deterioro de la emisión. Los detectores de emisiones de rayos X han estado disponible durante más de dos décadas, por lo que el apilamiento se ha convertido en una técnica de análisis estándar. Las aplicaciones comenzaron desde la determinación de la luminosidad media de rayos X, por ejemplo las galaxias normales, galaxias Lyman y las fuentes de radio, para determinar la emisión de rayos X de grupos distantes en el Röntgensatellit (ROSAT) All-Sky Survey.

Los detectores digitales lineales han llegado a dominar los estudios del cielo óptico e infrarrojo. La técnica de apilamiento ha sido ampliamente adoptada: por ejemplo, Zibetti et al. detectaron la luz intracluster apilando Sloan Digital Sky Estudio 683 (SDSS6); Lin et al., apiló datos sobre las galaxias del cúmulo Two Micron All Sky Survey (2MASS); Hogg et al. apiló datos del Keck IR para conseguir galaxias de colores tenues; Minchin et al. fue tan lejos como para apilar películas digitalizadas desde el telescopio Schmidt del Reino Unido.