UNIVERSIDAD DE CHILE
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

VALIDACIÓN DE UN ALGORITMO DE DETECCIÓN DE CÚMULOS DE GALAXIAS (VOCLUDET) Y VISUALIZACIÓN SOBRE UN WALL DISPLAY

MEMORIA PARA OPTAR AL TÍTULO DE INGENIERO CIVIL EN COMPUTACIÓN

SEBASTIÁN ALFREDO PEREIRA GALLARDO

PROFESOR GUÍA:
NANCY HITSCHFELD KAHLER

MIEMBROS DE LA COMISIÓN:
LUIS CAMPUSANO BROWN
JUAN MARÍN CAIHUAN

SANTIAGO DE CHILE
JUNIO 2014

RESUMEN DE LA MEMORIA PARA OPTAR
AL TÍTULO DE: Ingeniero Civil en Computación
POR: Sebastián Alfredo Pereira Gallardo
FECHA: JUNIO 2014
PROFESOR GUÍA: Nancy Hitschfeld Kahler

VALIDACIÓN DE UN ALGORITMO DE DETECCIÓN DE CÚMULOS DE GALAXIAS (VOCLUDET) Y VISUALIZACIÓN SOBRE UN WALL DISPLAY

Vocludet es un algoritmo de detección de cúmulos de galaxias, el cual está diseñado para detectar múltiples cúmulos en un espacio tridimensional. El algoritmo se basa en la Teselación de Voronoi para detectar regiones de alta densidad en el espacio, además de utilizar propiedades astrofísicas para determinar los componentes de cada cúmulo detectado.

El objetivo de esta memoria es validar el algoritmo utilizando datos ficticios que simulen un survey de galaxias y la creación de un software de visualización que permita realizar el estudio de los resultados obtenidos. El motivo por el cual se utilizan datos ficticios es el conocer los datos a utilizar de forma absoluta y sin incertidumbre, de tal modo que la validación pueda ser realizada apropiadamente. Debido al tamaño y complejidad del conjunto de datos a visualizar, se plantea la utilización de un wall display de alta resolución. Esto permite, además del incremento en la cantidad de información desplegada, una forma física de interacción con los datos: El usuario puede observar un panorama general de la información desde una distancia, o bien ver una mayor cantidad de detalle al acercarse a la pantalla.

En esta memoria se describe la extracción y procesamiento de los datos obtenidos a partir de una simulación astronómica, la elaboración del software de visualización de datos y la validación del algoritmo Vocludet bajo diversos criterios.

Agradecimientos

En primer lugar quiero agradecer a mis padres, quienes me han apoyado de forma incondicional a lo largo de toda mi vida.

También quisiera agradecer a mis profesores Nancy y Luis, quienes me han apoyado y guíado de manera excepcional. Sin su ayuda este trabajo no hubiera sido posible.

A Inria Chile, por permitirme el uso de su wall display, en particular Emmanuel Pietriga quien además me guió en el desarrollo de la visualización y mostró gran interés durante todo el proyecto.

A quienes dedicaron parte de su tiempo para ayudarme en diversos ámbitos, Chris Haines y Fernando del Campo.

Finalmente agradecer a mis amigos. Tanto a aquellos que están en Iquique como a aquellos que me han acompañado durante todo mi paso por esta facultad. Además, no puedo dejar de mencionar al gran grupo de personas que conocí este semestre a través del curso IN5502 y con los que compartí grandes experiencias.

Tabla de contenido

Índice de tablas

Índice de figuras

Índice de códigos

Capítulo 1

Introducción

Actualmente la astronomía se ve enfrentada al desafío de manejar y analizar una enorme cantidad de información que está siendo generada día a día. Este desafío se vuelve más relevante aún con el paso del tiempo, pues existen proyectos a futuro en los que la cantidad de datos que requieren ser procesados es cada vez mayor. Un ejemplo de esto es el proyecto Large Synoptic SurveyTelescope (LSST) [7], el cual podrá realizar un mapeo detallado y rápido de todo el cielo visible en un par de noches, generando datos del orden de Terabytes de información por día.

Muchas de las interrogantes que existen actualmente en astronomía necesitan mediciones de alta precisión de diversos componentes del universo. Es por esto que se requiere de algoritmos de procesamiento que puedan tomar la gran cantidad de datos generados y clasificarlos para obtener catálogos de información ricos y que permitan ser accedidos de forma coherente y simple. En particular, el acceso a estos catálogos se vuelve esencial para el estudio de la cosmología.

Los cúmulos de galaxias son sistemas que contienen desde cientos hasta miles de galaxias, dominados por galaxias elípticas, y cuya extensión es del orden de unos pocos millones de años luz. Cada galaxia, a su vez, está compuesta típicamente de cientos de miles de millones de estrellas. Estos objetos, al ser altamente masivos permiten trazar las regiones de alta densidad en la distribución de materia, por lo que una muestra completa provee, entre otros resultados, una descripción de la formación de estructuras en el universo temprano. Es por esta razón que el estudio de la distribución de cúmulos en el universo es fundamental para responder a muchas de las interrogantes de la cosmología, además de testear teorías ya existentes o plantear nuevas en base a la información disponible.

En este primer capítulo se detalla la motivación tras el problema a resolver, el objetivo general y los objetivos específicos. Además se menciona la metodología adoptada para el desarrollo del presente trabajo.

1.1. Motivación

Durante el año 2006 se desarrolla la tesis de magíster titulada "Galaxy Cluster Detection using Nonparametric Maximum Likelihood Estimation of Features in Voronoi Tessellations" [11]. En ella se describe el desarrollo de un algoritmo (Vocludet) de detección de cúmulos de galaxias, el cual está diseñado para detectar múltiples cúmulos en un espacio tridimensional. El algoritmo se basa en la Teselación de Voronoi para detectar regiones de alta densidad en el espacio, además de utilizar propiedades astrofísicas para determinar los componentes de cada cúmulo detectado.

Un aspecto que queda pendiente del trabajo anterior es la validación del algoritmo utilizando datos ficticios que simulen un survey de galaxias. El objetivo de usar datos ficticios es el conocer los datos a utilizar de forma absoluta y sin incertidumbre, de tal modo que la validación pueda ser realizada apropiadamente.

Además de esto, se hace necesaria la creación de un software de visualización de los datos y el resultado del algoritmo. Debido al tamaño y complejidad del conjunto de datos a visualizar, se plantea la utilización de un wall display de alta resolución. Esto permite, además del incremento en la cantidad de información desplegada, una forma física de interacción con los datos: El usuario puede observar un panorama general de la información desde una distancia, o bien ver una mayor cantidad de detalle al acercarse a la pantalla [1].

1.2. Objetivos

Los objetivos planteados para el desarrollo de este trabajo se presentan a continuación. Se incluye tanto el objetivo general como los objetivos específicos.

1.2.1. Objetivo general

El objetivo general consiste en la validación del algoritmo de detección de cúmulos de galaxias Vocludet y la creación de una herramienta de visualización interactiva sobre un wall display.

1.2.2. Objetivos específicos

- Obtener y procesar datos ficticios que simulan una distribución realista de una gran cantidad de galaxias.

- Validar el algoritmo Vocludet de detección de cúmulos de galaxias, usando los datos del punto anterior.

- Diseñar e implementar la visualización de galaxias y cúmulos en software que permita

el despliegue en un wall display.

1.3. Metodología

La metodología a seguir para cumplir los objetivos es la siguiente.

Para la validación del algoritmo:

- Estudiar y analizar material existente: tesis de software Vocludet, documentación y bibliografía relacionada. Esto incluye el estudio de la forma y características de los cúmulos de galaxias, además de la física utilizada en campos relacionados como la cosmología, astrofísica u otras aéreas relevantes de la astronomía.

- Adquirir y estudiar los datos a utilizar para la validación, seleccionar un subconjunto de datos apropiado y realizar el procesamiento necesario para poder ejecutar el algoritmo sobre ellos.

- Evaluar los resultados de la ejecución del algoritmo. Generar estadísticas que permitan comparar y evaluar la calidad los resultados obtenidos.

- Mejorar la calidad de la detección de cúmulos en caso que los resultados no sean los esperados.

Para la visualización del algoritmo:

- Determinar la mejor forma de representar los cúmulos de galaxias detectados. Esto incluye distintas alternativas, como: conjunto de puntos, polígonos convexos, segmentos, etc.

- Diseñar e implementar la visualización 3D tanto de todas las galaxias como de los cúmulos detectados por distintos algoritmos, de tal manera de poder observar diferencias y similitudes entre los cúmulos detectados. Debe ser posible navegar por el espacio y seleccionar algunos cúmulos que parezcan interesantes de analizar.

- Estudiar, determinar e implementar una forma de interacción con el wall display en la visualización de cúmulos de galaxias. Existen diversas técninas de interacción que ofrecen la posibilidad de integración con dispositivos móviles como celulares o tablets.

Capítulo 2

Investigación preliminar

Durante la investigación preliminar se estudia y analiza material existente. Esto incluye la tesis de Vocludet, documentación del software y bibliografía relacionada

2.1. Algoritmo Vocludet

En esta sección se describe en detalle el algoritmo Vocludet y el software correspondiente que lo implementa.

2.1.1. Descripción del algoritmo

El algoritmo es una extensión a tres dimensiones del algoritmo 2D desarrollado por Söchting, Clowes y Campusano [5]. Consiste de 2 etapas claramente definidas y desacopladas. La primera etapa se enfoca en la búsqueda de regiones en el espacio con una alta densidad local, en un proceso que depende exclusivamente de la densidad y que no se ve sesgado por suposiciones astrofísicas o de morfología. El siguiente paso utiliza los datos generados por el paso anterior para crear una vecindad en torno al punto de alta densidad, la cual incluye los miembros reales del cúmulo correspondiente. En esta segunda etapa sí se utiliza información astrofísica y de morfología para obtener los resultados.

Primera etapa

Esta etapa es llamada VT-MLE (Voronoi Tesellation - Maximum Likelihood Estimator) debido al uso de la teselación de Voronoi y al método estadístico MLE. El algoritmo recibe como input un conjunto P de n puntos en un espacio 3D, donde cada punto representa una galaxia. Previo a la ejecución de esta etapa se calcula la teselación de Voronoi T y la triangulación de Delaunay D para el conjunto P. La teselación de Voronoi entrega una estimación de la densidad local asociada a cada punto $p_{\mathrm{i}} \in P$ como el recíproco del volumen

de la celda de Voronoi $t_\mathrm{i} \in T$. Cada punto asociado a una celda es llamado 'semilla', ya que a partir de cada uno de estos puntos se hace crecer un cúmulo. Por otro lado, los vértices y nodos de la teselación de Delaunay forman un grafo de adyacencia para las celdas de T, de tal forma que para cada celda $t_\mathrm{i} \in T$ asociada a un punto p_i, se puede obtener su conjunto de celdas vecinas.

Una vez que se tiene el volumen asociado a cada semilla, estas son ordenadas en orden creciente. A la posición determinada por su tamaño se le llama ranking de la semilla. Posteriormente, para cada una de ellas y siguiendo el orden recién definido, se realiza lo siguiente:

- Se realiza la unión de la semilla con cada una de sus vecinas y se calcula el valor del MLE.

- Si es posible aumentar el MLE agregando nuevas semillas vecinas a la unión, estas se agregan.

El proceso continúa hasta que no sea posible aumentar el valor del MLE, en cuyo caso el grupo se registra como cúmulo. Si en algún punto anterior existe una semilla agregada al cúmulo que ya pertenecía a otro definido previamente, el grupo es descartado.

Finalmente, se calcula el centroide para cada cúmulo detectado y su resultado se entrega a la siguiente etapa.

Segunda etapa

La segunda etapa (llamada ZHG) consiste en la aplicación de una versión simplificada del algoritmo desarrollado por Zabludoff, Huchra y Geller [12], el cual se describe a continuación.

El algoritmo recibe como input una posición en el cielo $\vec{\mathrm{d}}$ junto con un Redshift z correspondiente. En primer lugar se extrae un dominio de búsqueda S en forma de cono de radio $r = 1{,}72/z$ (llamado radio de Abell) en la dirección $\vec{\mathrm{d}}$. El algoritmo analiza las separaciones en los valores de z de las galaxias contenidas en S en 2 pasos. En el primer paso, se busca en el dominio un subconjunto S_1 delimitado por separaciones de $z_{gap} = 0{,}0033$ en redshift por cada lado. En el segundo paso, se extrae un subconjunto S_2 de S_1, el cual está delimitado esta vez por la desviación estándar σ_z de valores de z en S_1. Debido a que existe una relación entre z y v_r, se puede hablar de σ_v (dispersión de velocidad) y v_{gap} en vez de σ_z y z_{gap}. El valor de v_{gap} equivalente a $z_{gap} = 0{,}0033$ es $v_{gap} = 1000 kms^{-1}$.

Es importante notar que existe 2 parámetros específicos para esta segunda etapa del algoritmo: $z_{gap} = 0{,}0033$ y el radio de Abell $R_{Abell}(\vec{\mathrm{d}}) = 1{,}72/z(\vec{\mathrm{d}})$.

2.1.2. Implementación

El algoritmo Vocludet está implementado en Java y su diseño se basa en una arquitectura de 3 capas. En la figura 2.1 se muestra el diagrama de la arquitectura general de Vocludet.

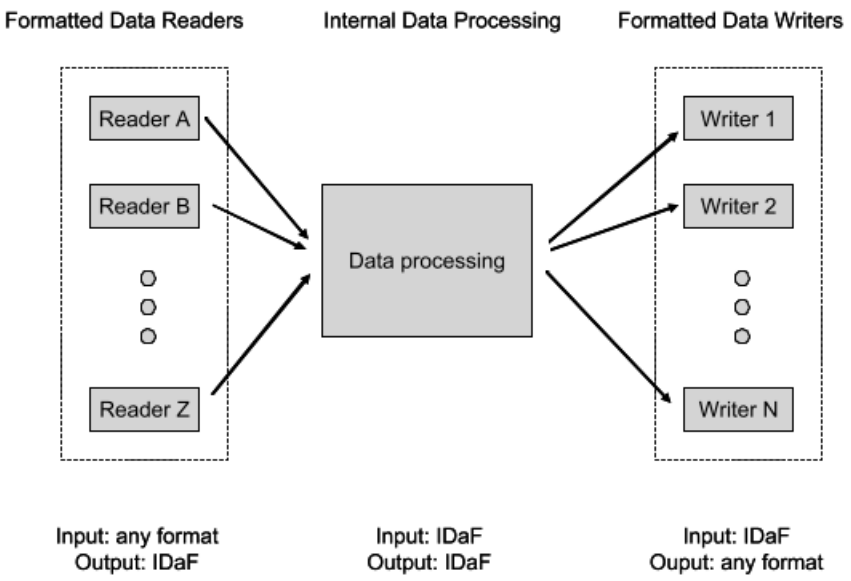

Figura 2.1: Arquitectura general de Vocludet

Debido a la gran diversidad de los potenciales datos que recibe la aplicación, la primera capa (FDR) se encarga de la lectura de datos y posterior traducción a un formato interno que es estándar. La segunda capa (IDP) recibe estos datos en formato estándar y se encarga del procesamiento y la ejecución del algoritmo. Finalmente la tercera capa (FDW) despliega los resultados generados en algún formato específico según el uso que se le requiera dar, e.g., estadísticas, visualizaciones, etc.

2.2. Fuente de los datos

Para la validación del algoritmo se requiere una cantidad significativa de datos que permitan realizar pruebas y obtener distintas mediciones de su calidad. Estos datos deben ser generados artificialmente, para poder así conocer de forma exacta sus propiedades. Esto permite analizar el desempeño del algoritmo de forma objetiva, sin contar con fuentes externas de distorsión en la calidad de la información producto de mediciones incorrectas, limitaciones instrumentales o errores sistemáticos, entre otros. Adicionalmente, en pruebas posteriores se puede introducir, de forma controlada, ruido o variaciones a la muestra utilizada para testear el algoritmo bajo condiciones más realistas.

Al momento de seleccionar la fuente de los datos para la validación del algoritmo se analizan 2 posibilidades. La primera considera la generación propia de los datos. Esto implica un gran desafío técnico y teórico, ya que la naturaleza de los datos es bastante compleja. Otra desventaja se presenta en que también se hace necesaria la validación de los mismos datos, antes de realizar la validación del algoritmo.

La segunda alternativa es el uso de datos simulados que ya se encuentren validados por la comunidad científica y sean de fácil acceso (datos públicos). Este es el caso de la simulación

Millennium [8], la cual ha sido utilizada en diversas publicaciones con motivos similares al de este trabajo. Esta simulación permite observar la formación de estructuras en el universo bajo la cosmología **ΛCDM**, utiliza 10^{10} partículas para seguir la distribución de Materia oscura en una región cúbica de 500 $h^{-1}Mpc$ de lado. Los datos generados a partir de este experimento se encuentran parcialmente disponibles para acceso público y en su totalidad para investigadores. Estos pueden ser accedidos mediante consultas SQL sobre sus diversas tablas.

Un ejemplo de la utilización de los datos de la simulación Millennium en investigaciones del área se encuentra en el trabajo realizado por Milkeraitis et al. [9]. En él se describe un algoritmo de detección de cúmulos de galaxias, junto con su validación, descripción de la extracción de los datos utilizados y sus resultados.

Finalmente, y luego de obtener acceso a las bases de datos completas, se opta por la segunda opción debido a los argumentos mencionados anteriormente que favorecen en gran medida esta alternativa.

2.3. Visualización en wall display

Debido al tamaño y complejidad del conjunto de datos a visualizar, se plantea la utilización de un wall display de alta resolución. Esto permite, además del incremento en la cantidad de información desplegada, una forma más física de interacción con los datos: El usuario puede observar un panorama general de la información desde una distancia, o bien ver una mayor cantidad de detalle al acercarse a la pantalla [1].

Gracias al apoyo de Inria Chile [1] se cuenta con el acceso al wall display "Andes" ubicado en sus oficinas. Andes es una matriz de pantallas LED de alta resolución, como se puede ver en la figura 2.2. Está compuesto de 24 paneles de marco fino posicionados en una matriz de 6x4, controlado por un cluster de 13 computadores. Actualmente se utiliza para conducir investigación y actividades de desarrollo relacionadas con la visualización interactiva de conjuntos de datos masivos. Sus áreas de aplicación incluyen: astronomía, centros de gestión de crisis, salas de control de grandes infraestructuras, etc.

Las especificaciones técnicas del display son las siguientes:

- 24 paneles LED FullHD para una resolución total de 11520 x 4320 pixeles
- 24 nVidia Quadro 2000 en 12 servidores 64-bit Dell Precision R5500, corriendo Linux Fedora 19 y cada uno equipado con:
  - CPU: 2x Intel Xeon E5606@2.13GHz
  - Memoria: 12GB DDR3 RAM
  - Almacenamiento: 256GB SSD + 500GB HDD@7200RPM

[1] `http://www.inria.cl`

Figura 2.2: Wall display en oficinas de Inria Chile

Capítulo 3

Extracción y procesamiento de datos

En este capítulo se describe el proceso de extracción de los datos y su posterior procesamiento para poder ser utilizados por el algoritmo Vocludet.

3.1. Selección de catálogo y subconjunto de datos

Los datos generados por la simulación Millennium se encuentran almacenados en múltiples tablas de bases de datos [8], las cuales contienen información particular de un cierto elemento como galaxias o halos de materia oscura, o catálogos construidos a partir de un post-procesamiento de los datos originales. Esto último es el caso del conjunto de datos Blaizot2006_Allsky [2], el cual es contiene 6 catálogos que simula la posición de galaxias a través de todo el cielo, desde el punto de vista de un observador arbitrario. De esta forma, se tiene para cada galaxia, su posición relativa en el cielo (declinación y ascención recta), su distancia (redshift) y otros parámetros astrofísicos.

Nombre	**Tipo de dato**	**Descripción**
ObjID	long	ID único de la galaxia en el catálogo
fofID	long	ID del halo central del grupo friends-of-friends en el cual reside la galaxia
type	short	Indicador del tipo de galaxia (central o satélite)
ra	float	Ascención recta
dec	float	Declinación
app_redshift	float	Redshift aparente

Tabla 3.1: Estructura de tabla de galaxias

3.1.1. Catálogo de galaxias

Debido a que la ejecución del algoritmo Vocludet requiere de un uso intensivo de recursos computacionales, se extrae una región pequeña del cielo, en lugar de usar el conjunto de datos completo. Además se consideran restricciones adicionales al subconjunto de datos a extraer,

debido a las características particulares del algoritmo. Un ejemplo de esto es el hecho que Vocludet está diseñado para funcionar en forma óptima a partir de un cierto valor de redshift, que corresponde a 0,009. Otro filtro utilizado busca eliminar las galaxias que puedan ser muy lejanas mediante la fijación de un brillo aparente mínimo. Además se obtienen datos que relacionan cada galaxia a un cúmulo. Lo anterior queda resumido en la siguiente consulta SQL, con la cual se adquiere el subconjunto de datos de galaxias utilizado.

```
SELECT ObjID, ra, dec, app_redshift, fofID, type
FROM mpamocks..blaizot2006_allsky_rt_1
WHERE app_redshift > 0.009
AND dec < -20
AND dec > -35
AND ra < 60
AND ra > -30
AND (SDSS_g + 0.15 + 0.13*(SDSS_g-SDSS_r)) < 19.45
```

Código 3.1: Consulta SQL catálogo de galaxias

Cuyo resultado se muestra a continuación:

ObjID	**ra**	**dec**	**app_redshift**	**fofID**	**type**
887359	57.204338	-31.17752	0.27106178	53000000000000	0
5184596	59.50467	-31.80862	0.25429428	54000100000000	0
2414740	54.82982	-31.780613	0.26495385	53000000000574	0
1973385	58.46274	-30.691324	0.24905908	54000000000942	0
1944023	58.993526	-30.626064	0.24768531	54000000000757	0
5342768	-29.979748	-20.283619	0.22658455	54000000000757	0
1246567	39.722115	-30.523909	0.13383389	57000000000826	2
5556860	39.7021	-30.542847	0.13422072	57000000000826	2
575552	39.664795	-30.582113	0.1344341	57000000000826	2
3566578	39.603336	-30.554523	0.1345272	57000000000826	1
...	...	...	...	...	...

Tabla 3.2: Ejemplo output consulta SQL catálogo de galaxias

3.1.2. Catálogo de cúmulos

Para poder validar el resultado de la ejecución de Vocludet se debe contar no sólo con un catálogo de galaxias, sino que también con un catálogo de cúmulos de referencia contra el cual comparar los resultados obtenidos. Debido a que la simulación no cuenta con un catálogo de cúmulos explícitamente definido, este debe ser generado a partir la información disponible y tomando criterios definidos de forma específica para la aplicación. Los criterios utilizados provienen de la publicación de Milkeraitis et al. [9] la cual también utiliza datos de la simulación Millennium para validar un algoritmo de detección de cúmulos y define como cúmulo toda aquella agrupación de galaxias que:

- Posea a lo menos 5 miembros
- Cada miembro posea el mismo identificador de Friend-of-friends

En donde el identificador Friend-of-friends está relacionado con agrupaciones que están ligadas gravitacionalmente. De esta forma es posible identificar a cada cúmulo del catálogo y sus respectivos miembros. Considerando estos criterios, se genera la siguiente consulta SQL:

```
SELECT distinct CandidateClusters.fofid, CandidateClusters.ObjID
FROM (
    SELECT *
    FROM mpamocks..blaizot2006_allsky_rt_1
    WHERE mpamocks..blaizot2006_allsky_rt_1.fofid in
        (
            SELECT fofid
            from mpamocks..blaizot2006_allsky_rt_1
            WHERE dec < -20 AND dec > -35 AND ra < 60 AND ra > -30
            AND (SDSS_g + 0.15 + 0.13*(SDSS_g-SDSS_r)) < 19.45
            AND app_redshift > 0.009
            GROUP BY fofid
            having count(fofid) > 4
        )
    AND dec < -20 AND dec > -35 AND ra < 60 AND ra > -30
    AND (SDSS_g + 0.15 + 0.13*(SDSS_g-SDSS_r)) < 19.45
    AND app_redshift > 0.009
    ) as CandidateClusters
```

Código 3.2: Consulta SQL catálogo de cúmulos

Con esto se obtiene una lista de identificadores de cúmulos asociados a cada galaxia. Se elige este formato para facilitar el parsing que es descrito en la siguiente sección.

fofID	**ObjID**
54000400000000	392849
54000400000000	675327
54000400000000	4109479
54000400000000	4650519
54000400000000	4902504
55000900000523	692717
55000900000523	1448845
55000900000523	2238044
55000900000523	2322996
55000900000523	2661518
...	...

Tabla 3.3: Ejemplo output consulta SQL catálogo de cúmulos

3.2. Análisis del conjunto de datos

Es necesario conocer en profundidad el catálogo de referencia que va a ser utilizado. Con este objetivo se realiza un análisis preliminar, lo que implica el cálculo de diversas estadísticas.

3.2.1. Distribución de redshift

Una propiedad de interés es la distribución de redshift, ya que como se menciona en la sección 2.1.1, uno de los parámetros del algoritmo depende del redshift z (el radio de Abell). En la figura **??** se muestra la distribución de redshift de los cúmulos pertenecientes al catálogo de la simulación Millennium.

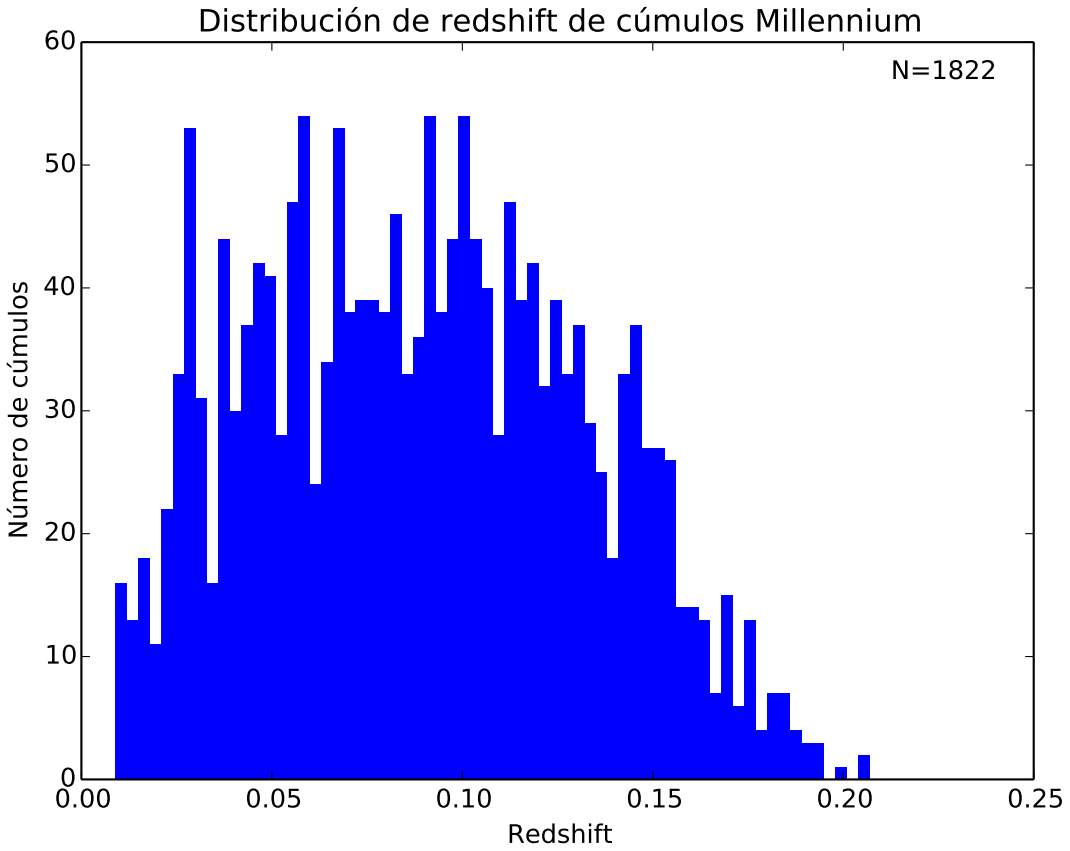

Figura 3.1: Distribución de redshift de cúmulos Millennium

El redshift promedio de la muestra es de 0,092 y su rango varía entre 0,009 y 0,21.

3.2.2. Distribución de dispersión de velocidad

Otra propiedad de interés es la distribución de dispersión de velocidad σ_v, ya que como también se menciona en la sección 2.1.1, el algoritmo utiliza este valor al momento de seleccionar qué galaxias pertenecen a un determinado cúmulo. En la figura 3.2 se muestra la distribución de dispersión de velocidad de los cúmulos pertenecientes al catálogo de la simulación Millennium.

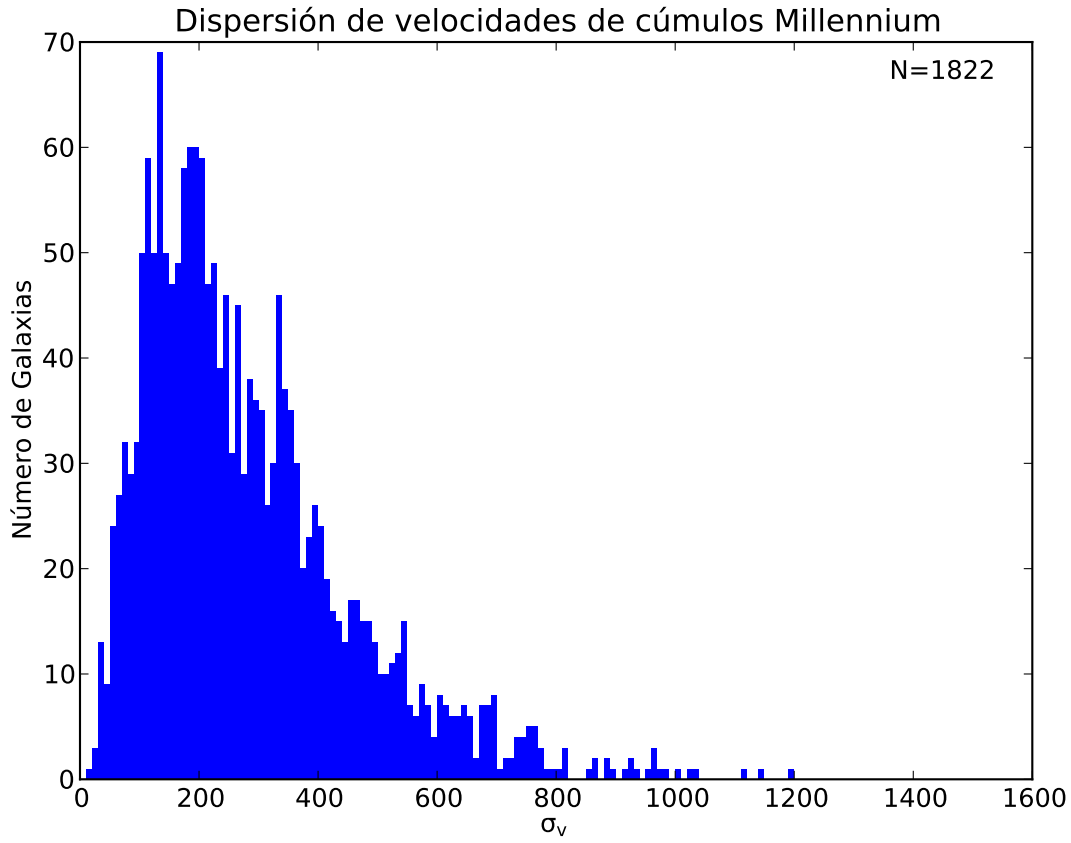

Figura 3.2: Distribución de dispersión de velocidad de cúmulos Millennium

3.3. Lectura de datos

Como se menciona en el capítulo anterior, la implementación del algoritmo Vocludet sigue una arquitectura de 3 capas, de las cuales la primera está encargada de la lectura de datos. Se requiere, por lo tanto, crear un módulo de lectura específico para los datos de la simulación Millennium. El módulo consiste de una clase principal (parser) y 2 clases auxiliares (tipo de datos). En la figura 3.3 se muestra el diagrama simplificado de clases del Módulo de lectura.

Al ejecutarse, el módulo lee los archivos que contienen los datos en bruto y los transforma al formato interno (IDaF) definido por la implementación de Vocludet, el cual incluye

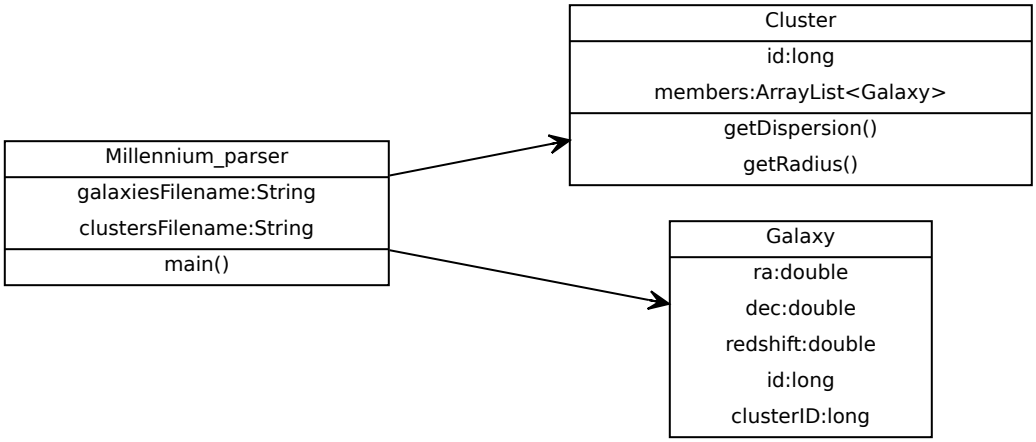

Figura 3.3: Diagrama simplificado de clases de Módulo de lectura

índices, posiciones de las galaxias en distintas coordenadas y archivos de máscara (para delimitar regiones), entre otros [11]. Además se generan datos adicionales que facilitan el análisis posterior del catálogo que está siendo utilizado.

En esta etapa también se aborda un problema encontrado en etapas posteriores del trabajo, el cual consiste en la existencia de cúmulos con galaxias en posiciones inconsistentes en la simulación. Este problema se debe a la existencia de galaxias duplicadas, las cuales son generadas durante el paso de coordenadas absolutas a coordenadas relativas a un observador, como es descrito en la publicación de Blaizot et al. [2]. Debido a que el volumen de la simulación es finito, para crear un cono de observación que se extienda más allá de los límites predefinidos, se deben replicar todas las galaxias para crear una extensión virtual. Esta extensión virtual es equivalente a utilizar la operación módulo en cada dimensión, tomando como base el largo del volumen de la simulación. Por ejemplo (tomando el caso de 1 dimensión), si el largo de la simulación es L y se requiere realizar una observación a una distancia $L+1$, esto equivale a observar la posición $1 \equiv L + 1 \mod L$ Esto produce que algunos cúmulos posean 1 o 2 galaxias a distancias excesivamente lejanas, ya que estos pueden contener miembros que fueron duplicados. El problema se soluciona eliminando estas galaxias inconsistentes al definir una distancia máxima a la deben estar para ser consideradas como miembros válidos de los cúmulos.

3.4. Procesamiento de datos

Teniendo el módulo de lectura, el algoritmo ya puede ser ejecutado para obtener resultados. Estos resultados se encuentran en el formato interno, por lo que se hace necesaria la creación de un módulo de estadísticas que permita comparar los resultados del algoritmo con los del catálogo de referencia.

3.4.1. Módulo de estadísticas

El módulo de estadísticas consiste de 4 clases que permiten tomar los datos generados por el algoritmo y compararlos con un segundo catálogo. Este segundo catálogo puede ser uno de referencia (como es el caso del catálogo Millennium) o bien con el resultado de una ejecución previa del algoritmo (posiblemente con parámetros distintos). En la figura 3.4 se puede ver el diagrama de clases simplificado de este módulo.

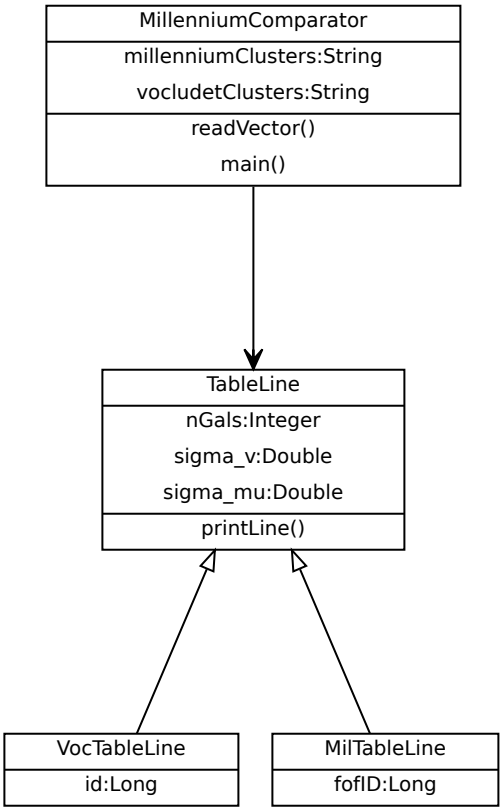

Figura 3.4: Diagrama simplificado de clases de Módulo de estadísticas

El módulo genera dos tablas de datos, una por cada catálogo de cúmulos. En estas tablas, para cada cúmulo se indican los siguientes valores:

- ID del cúmulo
- Número de galaxias en el cúmulo
- ID del cúmulo del otro catálogo con el que se posea una mayor intersección (en adelante, cúmulo complementario).
- Número de galaxias en el cúmulo complementario.
- Porcentaje de sus galaxias que pertenecen al cúmulo complementario (Completitud).
- Porcentaje de galaxias del cúmulo complementario que no pertenecen a este cúmulo

(Falsos positivos).

- Otros parámetros astrofísicos como dispersión de velocidad, redshift promedio y tamaño angular.

Finalmente, al tener el módulo de lectura y estadísticas implementados, se crea un script encargado de llamar a los distintos módulos secuencialmente.

Capítulo 4

Visualización

El creación del software de visualización tiene como objetivo el facilitar el análisis de los datos y resultados obtenidos. A continuación se detallan los distintos aspectos que se consideran durante su desarrollo.

4.1. Elección de framework

Para la implementación se analizan dos alternativas de framework que permiten el despliegue en wall displays: Equalizer [1] y ZVTM [2].

4.1.1. Equalizer

Equalizer es un framework estándar que permite crear y desplegar aplicaciones paralelas basadas en OpenGL. Entrega la posibilidad de usar múltiples tarjetas gráficas, procesadores y computadores para escalar la eficiencia del rendering, calidad visual y tamaño de la visualización [6].

Prototipo

Con motivo de prueba se desarrolla un primer prototipo basado en Equalizer, con funcionalidades básicas. En este prototipo se pueden visualizar los datos de los archivos que contienen la información de galaxias y cúmulos con sus respectivas coordenadas. Para cada galaxia se realiza la conversión de coordenadas celestes a coordenadas cartesianas. Luego cada galaxia es representada por un cubo de tamaño y color fijo. Además se representa el centro geométrico de cada cúmulo como un cubo de tamaño superior al de las galaxias y

[1] `http://www.equalizergraphics.com`
[2] `http://zvtm.sourceforge.net`

un color distinto que permita observar un contraste. En las figuras 4.1 y 4.2 se pueden ver capturas de pantalla que muestran el prototipo en funcionamiento.

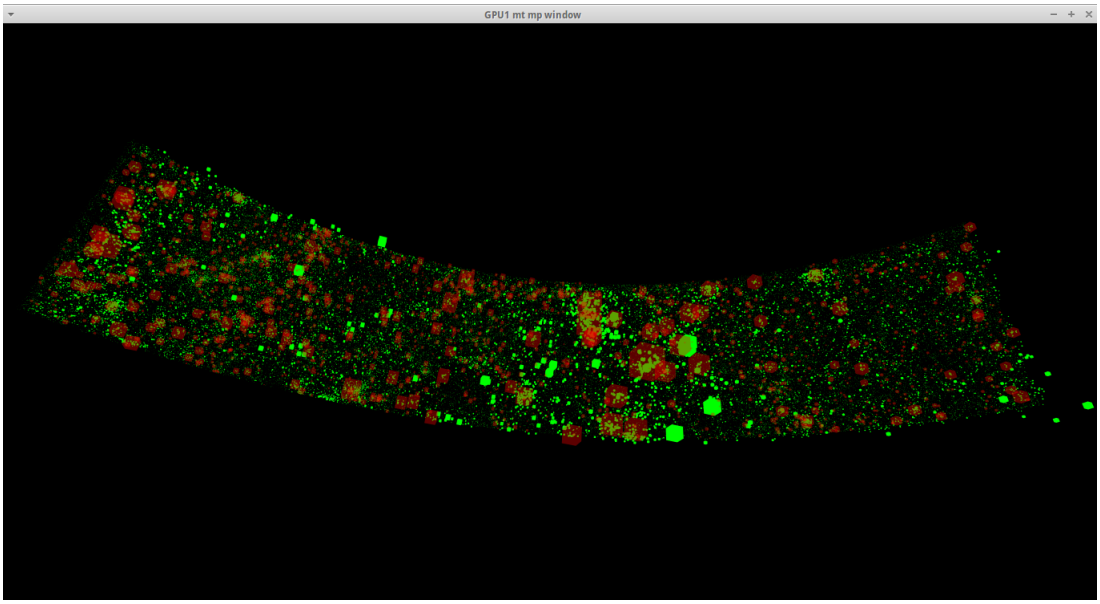

Figura 4.1: Imagen prototipo Equalizer. Galaxias en verde y centros de cúmulos en rojo.

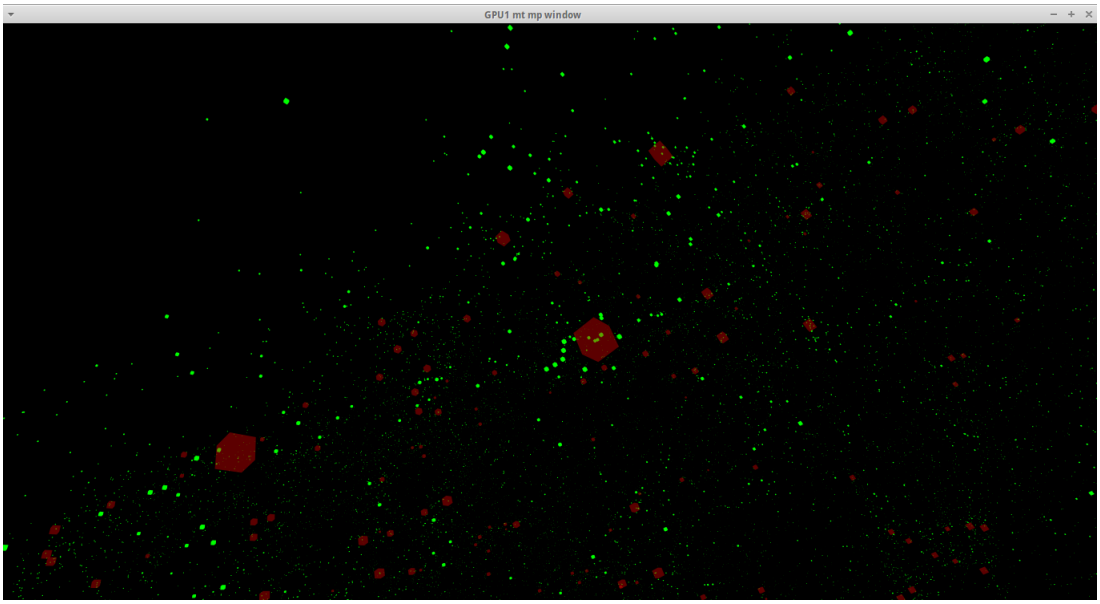

Figura 4.2: Imagen prototipo Equalizer, vista alternativa. Galaxias en verde y centros de cúmulos en rojo.

Además se implementa la interacción basica con el usuario, la cual permite navegar a través del espacio utilizando el teclado.

4.1.2. ZVTM

ZVTM (Zoomable Visual Transformation Machine) es una herramienta diseñada para facilitar la tarea de crear componentes de interfaz compleja, requeridas por ambientes de lenguaje visual, meta-herramientas de lenguaje visual o editores de grafos, a la vez que favorece la rápida integración de técnicas de interacción novedosas a través de las cuales estas aplicaciones sepueden ver ampliamente beneficiadas [10]. ZVTM además permite el desarrollo de aplicaciones destinadas a ser ejecutadas en ambientes altamente paralelizados, como es el caso del ambiente requerido por wall displays. Para lograr esto, se provee una API que proporciona un conjunto de características que permite reutilizar el mismo código inicialmente desarrollado para un ambiente de escritorio, en la aplicación que se ejecuta en wall displays.

ZVTM está basado en la metáfora de universos infinitos llamados virtual spaces que pueden ser observados a través de cámaras móviles y con capacidad de hacer zoom, y que pueden contener grandes cantidades de objetos gráficos llamados glyphs: formas geométricas, imágenes bitmap o texto. Típicamente se utilizan círculos, polígonos y líneas rectas o curvas.

Todos los glyphs dependen del mismo modelo de objeto polimórfico. Un glyph pertenece a un virtual space específico, pero puede ser observado por múltiples cámaras simultáneamente. Las cámaras están asociadas a ventanas gráficas llamadas views que corresponden a ventanas en la interfaz del usuario. El modelo gráfico de los glyphs soporta el canal alpha, y por lo tanto, pueden ser opacos, traslúcidos o transparentes. La transparencia es una de las múltiples variables visuales que definen a un glyph. Modificaciones a estas variables pueden ser animadas utilizando diversos esquemas temporales. Traslaciones de la posición de las cámaras pueden ser modificadas.

Los eventos de input del usuario son manejados a través de callbacks de alto nivel asociados a cada view. Cada método de callback provee un contexto sobre el evento, como por ejemplo, una lista de los objetos que intersectan con el cursor.

Prototipo

Al igual que con el framework anterior se implementa un prototipo que entrega la posibilidad de visualizar las galaxias. Cada galaxia se representa como un círculo de tamaño y color que dependen de su posición. Debido a que ZVTM no soporta nativamente coordenadas 3D, se utilizan distintas proyecciones 2D para entregar la información espacial que es necesaria. La información que se pierde al disminuir una dimensión se recupera al representar profundidad a través del color y tamaño. El tamaño varía dependiendo de la distancia a la que se encuentre una galaxia, siendo esta más pequeña mientras más lejana esté. El color también varía dependiendo de la distancia a cada galaxia. Se implementan dos esquemas de colores:

1. Variación lineal de vector de color en espacio RGB: $\vec{C} = (1, \mathrm{d}, \mathrm{d})$, con d representando la distancia (redshift), normalizado para variar entre $[0, 1]$.

2. Variación lineal de vector de color en espacio HSV: $\vec{C_2} = (\mathrm{d}, c_1, c_2)$, con d representando

la distancia (redshift), normalizado para variar entre $[0, \pi]$ y c_1, c_2 constantes.

En las figuras 4.3 y 4.4 se pueden ver los distintos tipos de esquemas de color implementados en el prototipo.

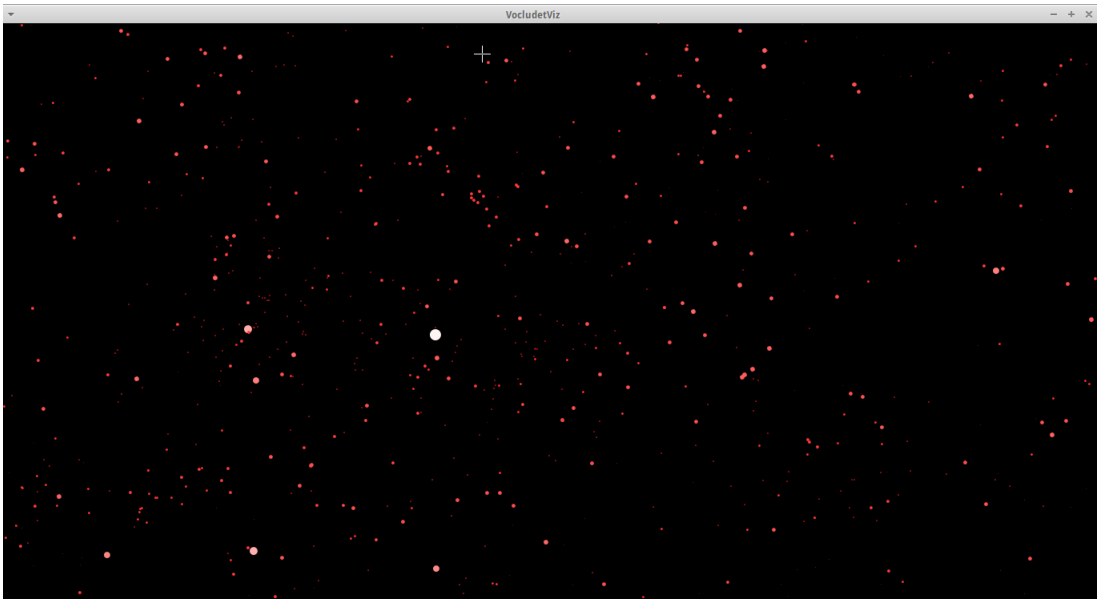

Figura 4.3: Imagen prototipo ZVTM, esquema RGB. Cada punto representa una galaxia y su color y tamaño depende de su distancia (redshift).

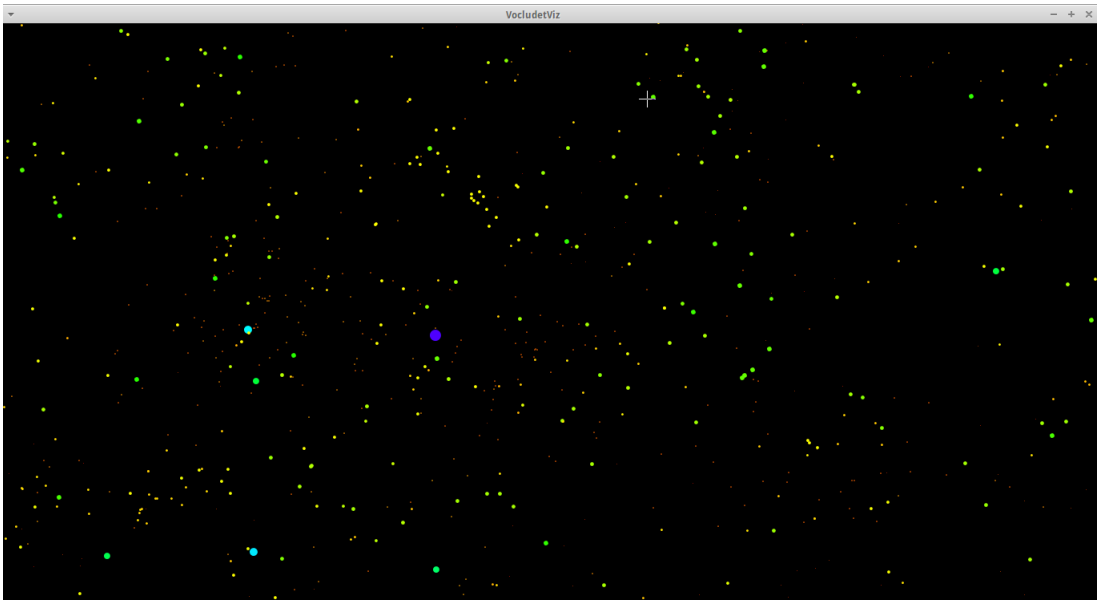

Figura 4.4: Imagen prototipo ZVTM, esquema HSV. Cada punto representa una galaxia y su color y tamaño depende de su distancia (redshift).

Además se implementan dos tipos de proyecciones que permiten ver el conjunto de datos desde diferentes perspectivas, lo cual facilita el análisis de los datos:

1. Transformación directa de Coordenadas ecuatoriales (RA, DEC) a cartesianas (x, y)

mediante escala de tamaño fijo:

$$(x, y) = (RA \cdot K, DEC \cdot K)$$

donde K es una constante de escala, RA es la ascención recta y DEC es la declinación.

2. Transformación utilizando información de redshift:

$$(x, y) = K_2 \cdot z \cdot cos(RA) \cdot sin(DEC), K_2 \cdot z \cdot sin(RA) \cdot sin(DEC)$$

donde K_2 es una constante de escala y z es el redshift.

En las figuras 4.5 y 4.6 se pueden ver los dos tipos de proyecciones implementadas en el prototipo.

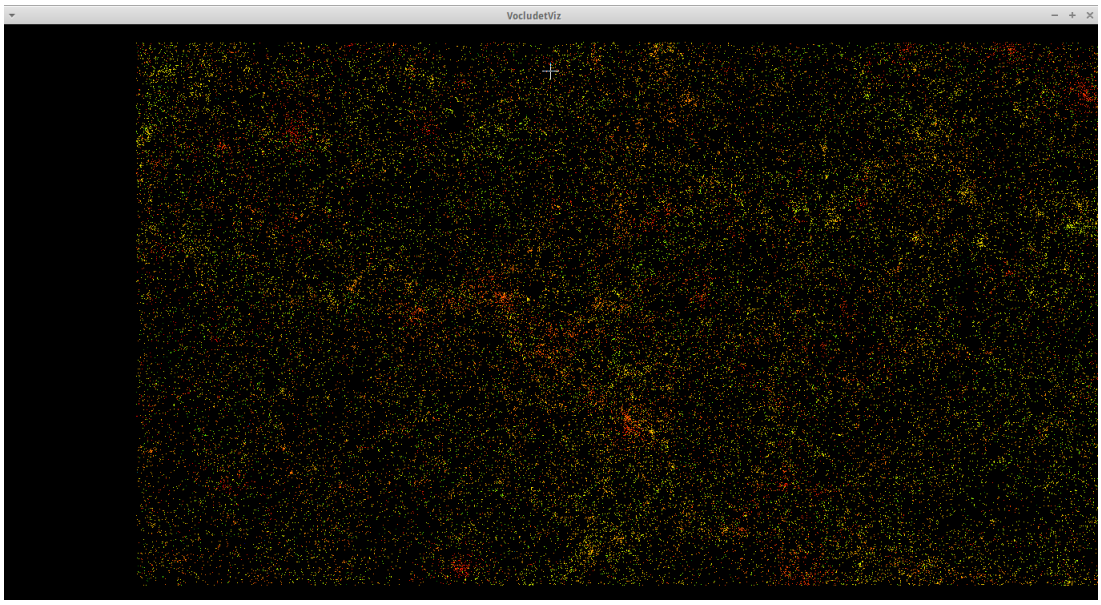

Figura 4.5: Imagen prototipo ZVTM. El eje vertical representa declinación (DEC) y el eje horizontal ascensión recta (RA).

4.1.3. Framework seleccionado

Finalmente se selecciona el framework ZVTM para continuar el desarrollo. Sus principales ventajas por sobre Equalizer son su gran cantidad de herramientas provistas para la interacción dinámica con los datos, tales como la existencia de listeners de eventos de teclado, mouse y táctiles, a través de dispositivos móviles. Además de lo anterior, ZVTM permite un desarrollo más ágil, al no requerir implementar aspectos de más bajo nivel, como shaders en OpenGL.

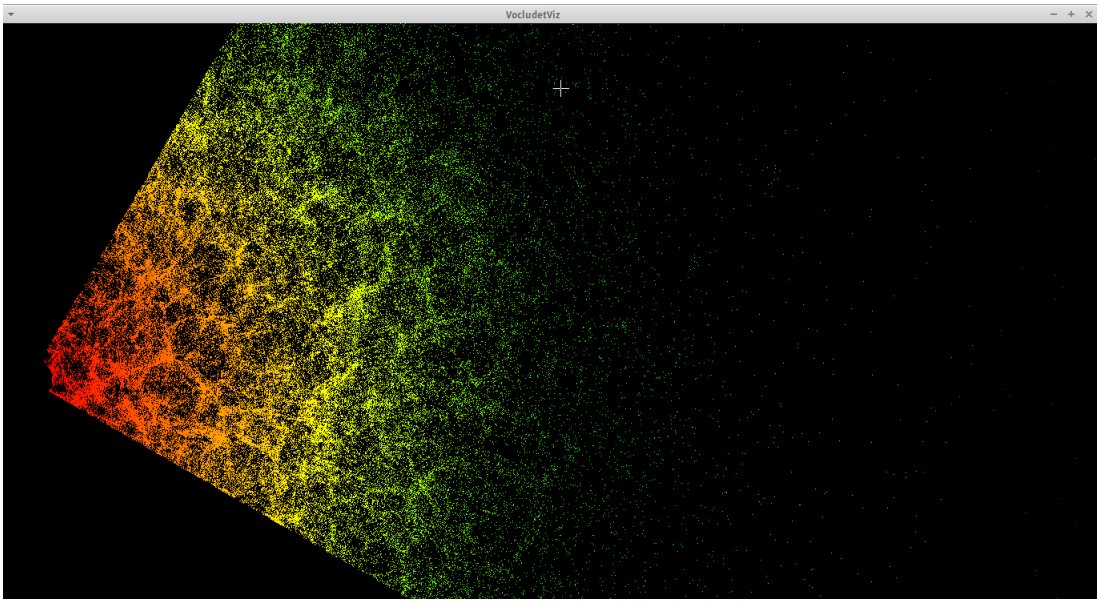

Figura 4.6: Imagen prototipo ZVTM. El punto de convergencia de las galaxias representa la ubicación del observador. El ángulo en el plano de la visualización representa ascensión recta (RA), y la distancia al observador representa redshift.

4.2. Diseño

Para el diseño de la aplicación se propone la arquitectura MVN, para lograr mantener la funcionalidad de los distintos componentes de forma modularizada e interconectada. La capa del modelo debe realizar toda la manipulación de los datos, incluyendo la lectura y caracterización de los mismos. La capa de controlador interactúa con los datos y el usuario, encargándose de recibir su input y reflejar los cambios en los datos que se manejan. Finalmente, la capa vista realiza los cambios en la forma de desplegar los datos, ya sea el color, la forma o la cantidad de información.

Como se ve en el diagrama de la figura 4.7, la capa de vista se encarga del despliegue local o en el wall display. Además el controlador soporta distintos tipos de comunicación, como mouse, teclado y dispositivos móviles.

4.3. Implementación

Tomando en cuenta el diseño basado en la arquitectura MVC, se implementan las distintas clases que corresponden a las 3 capas. A continuación se explica en detalle la implementación de cada una de las capas.

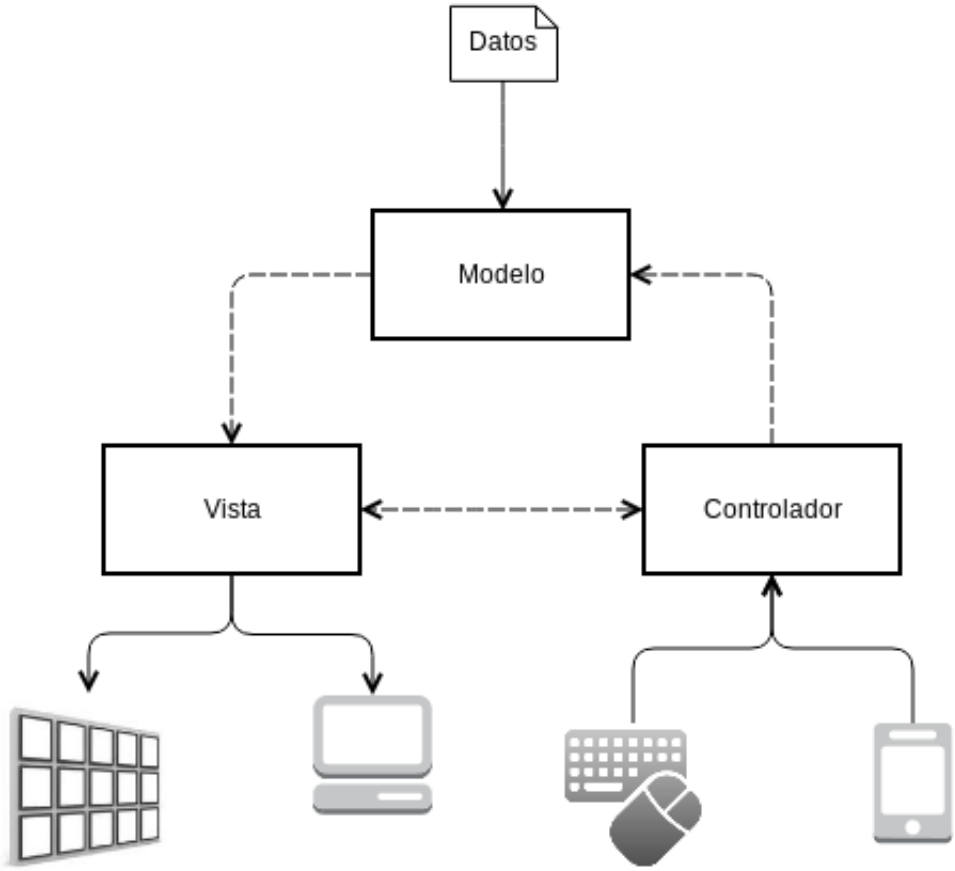

Figura 4.7: Arquitectura de software de visualización.

4.3.1. Modelo

El modelo se compone de 3 clases, las cuales se muestran en la figura 4.8:

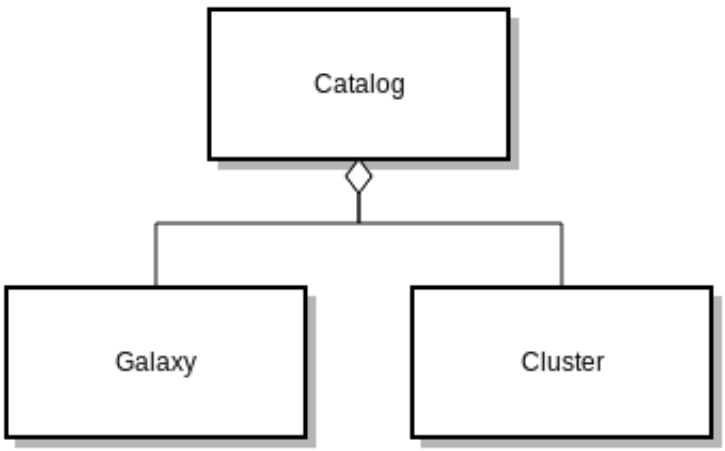

Figura 4.8: Diagrama de clases de la capa de modelo

La clase `Catalog` representa un catálogo astronómico. Contiene una colección de galaxias y cúmulos, la cual es generada al momento de instanciarlo. Recibe como parámetros la ruta de los archivos que contienen la información de sus miembros.

Por otro lado, la clase `Galaxy`, como su nombre lo indica, representa una galaxia. Contiene información acerca de su posición, redshift y pertenencia a cúmulos. Esto permite que cada galaxia pueda estar relacionada con más de un cúmulo, lo cual es útil al momento de comparar distintos catálogos, en los cuales una galaxia puede pertenecer a un cúmulo de un catálogo pero no a uno del otro.

Finalmente la clase `Cluster` representa un cúmulo de galaxias. Posee información acerca del conjunto de galaxias que pertenecen a cada instancia en particular. Además contiene métodos que permiten obtener información como el centro geométrico del cúmulo, su tamaño, redshift promedio, dispersión de velocidades, etc.

4.3.2. Vista

La capa de vista del software se compone de 5 clases, que se pueden ver en la figura 4.9:

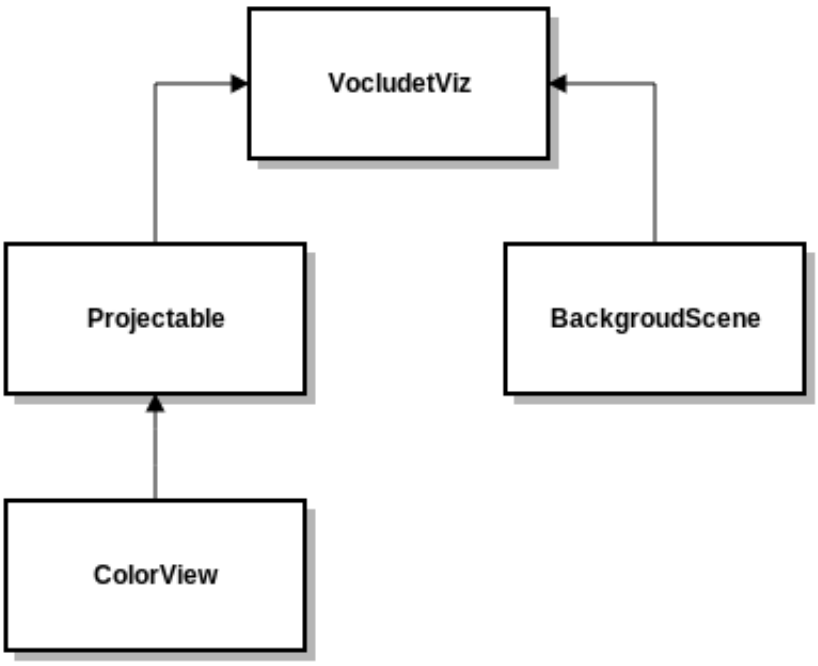

Figura 4.9: Diagrama de clases de la capa de vista

La clase principal corresponde a `VocludetViz`, que entre otras funciones, se encarga de desplegar los datos en la pantalla del computador que ejecuta la aplicación, o en el wall display, en caso de ser posible.

`ColorView` permite modificar y elegir entre los distintos esquemas de colores (mencionados en la sección 4.1.2) para representar las galaxias. El tener esta funcionalidad aislada en una

clase propia, permite extender fácilmente el posible uso de otros tipos de esquemas de colores que requieran representar otro tipo de variable en lugar de distancia.

Por otra parte, la clase `BackgroundScene` despliega información acerca de objetos individuales (galaxias o cúmulos) en forma de paneles de información. Por ejemplo, puede mostrar información sobre la posición de un cúmulo en particular o su dispersión de velocidades.

Para concluir esta capa, se tiene la clase `Projectable`, que representa cada uno de los objetos que pueden ser observados desde distintas proyecciones, es decir, galaxias y cúmulos. Esta clase no implementa las distintas proyecciones, ya que de eso se encarga la clase `ProjectionController`, mencionada en la siguiente capa.

4.3.3. Controlador

La tercera capa de la aplicación, la capa controlador, posee 4 clases, como se muestra en la figura 4.10:

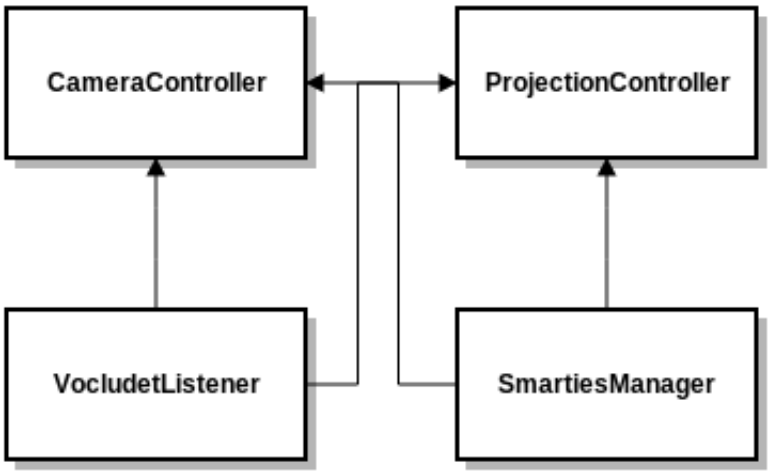

Figura 4.10: Diagrama de clases de la capa de controlador

Las clases `ProjectionController` y `CameraController` se comunican directamente con la clase `VocludetViz` para representar las diversas proyecciones y entregar la posibilidad de observar los datos desde diversas cámaras (Ver sección 4.1.2. `ProjectionController` implementa las dos proyecciones mencionadas en la sección 4.1.2, y además entrega la funcionalidad de agregar animaciones que permiten analizar datos en específico en más detalle.

Por otro lado, la clase `VocludetListener` maneja todo el input proveniente del teclado y mouse y lo refleja como cambios en la vista, a través de los otros controladores. Por ejemplo, esta clase contiene los listeners de eventos del mouse que realizan el zoom, el movimiento o la selección de algún objeto en particular.

Finalmente, la clase `SmartiesManager` implementa la interacción con la aplicación a través

de dispositivos móviles, como tablets o celulares con sistema operativo Android. Para esto se utiliza una biblioteca desarrollada en Inria, llamada Smarties [3]. Esta biblioteca permite asignar funcionalidad a una serie de input producidos a través de gestos en pantallas táctiles, como "pinch.º arrastre de múltiples dedos sobre la pantalla.

4.4. Características de la aplicación

La aplicación posee una serie de características que facilitan el estudio de los resultados del algoritmo.

4.4.1. Visualización de cúmulos

La aplicación cuenta con dos representaciones para los cúmulos. En primer lugar está la opción de representar los cúmulos como segmentos que unen distintas galaxias con su centro geométrico, como se muestra en la figura 4.11. Esta forma es adecuada para cúmulos separados, ya que facilita la visualización de cada galaxia perteneciente al mismo, pero se vuelve confusa al haber superposición entre distintos cúmulos.

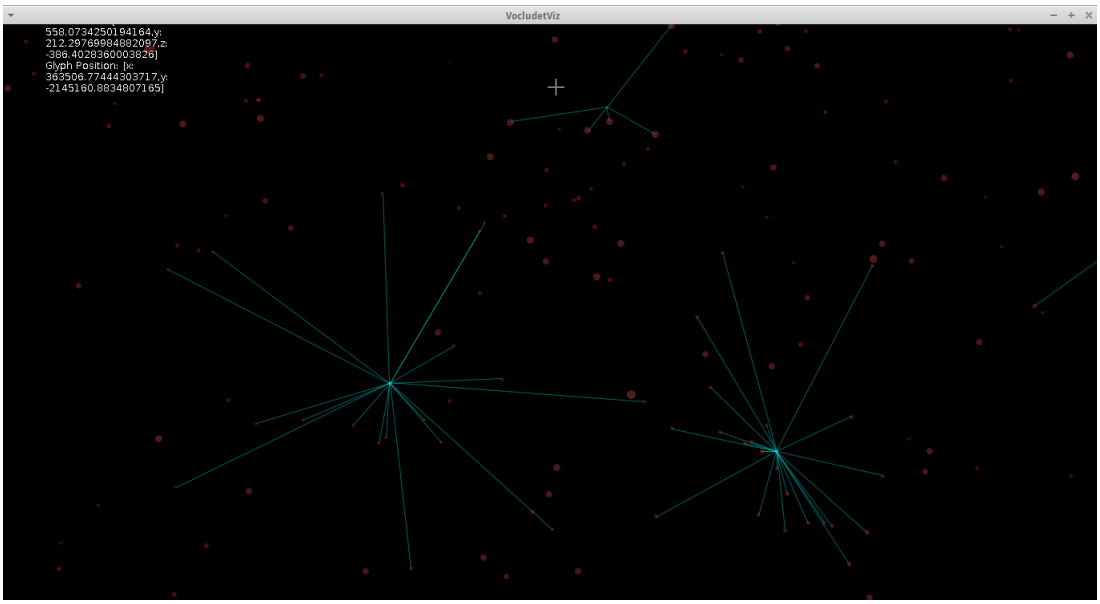

Figura 4.11: Visualización de cúmulos tipo segmentos

En segundo lugar se tiene la representación de polígonos, en la que para cada cúmulo se calcula la cerradura convexa y luego se muestra en forma de polígono semi-transparente, como se muestra en la figura 4.12. En este caso, se aprecia muy bien el tamaño de los cúmulos y se mantiene fácil de interpretar aún cuando hay intersecciones.

Además, la aplicación puede desplegar dos catálogos de cúmulos a la vez. Esto tiene como objetivo el poder comparar el resultado del algoritmo con el catálogo de referencia.

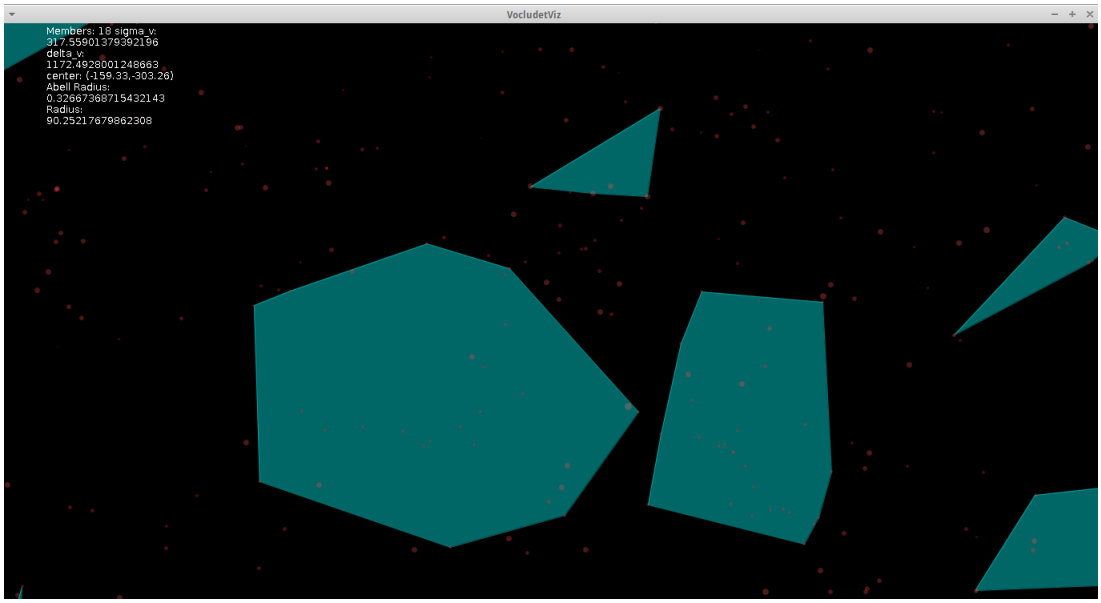

Figura 4.12: Visualización de cúmulos tipo polígonos

Los cúmulos perteneciente a cada catálogo son desplegados con colores distintos para poder diferenciarlos fácilmente. Cada uno de los catálogos puede activarse o desactivarse de forma dinámica. Esta característica puede verse en la figura 4.13.

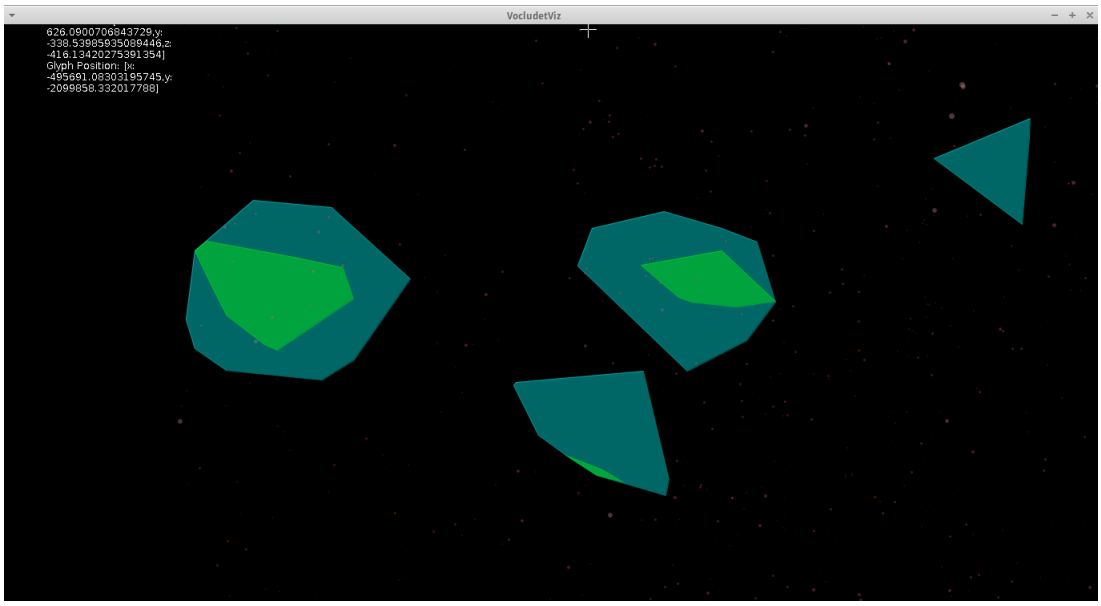

Figura 4.13: Visualización de cúmulos pertenecientes a dos catálogos

4.4.2. Vista individual

Además de la vista general en la que se pueden ver todos los cúmulos de los catálogos a la vez, se provee la funcionalidad de analizar cúmulos de forma individual. Para esto, al

momento de seleccionar un cúmulo, hay un cambio de escena en la que se puede ver: el cúmulo recién seleccionado, los cúmulos que intersecten con él, las galaxias pertenecientes a los cúmulos recién mencionados y un conjunto de galaxias que se encuentren a una cierta distancia, como se puede ver en la figura 4.14. Además se muestra un círculo que representa un círculo de radio R_{Abell}, el radio de Abell mencionado en la sección 2.1.1. Esto permite estudiar el desempeño del algoritmo en casos en que la detección no es adecuada, por ejemplo, cuando sólo se detecta una parte de un cúmulo de referencia.

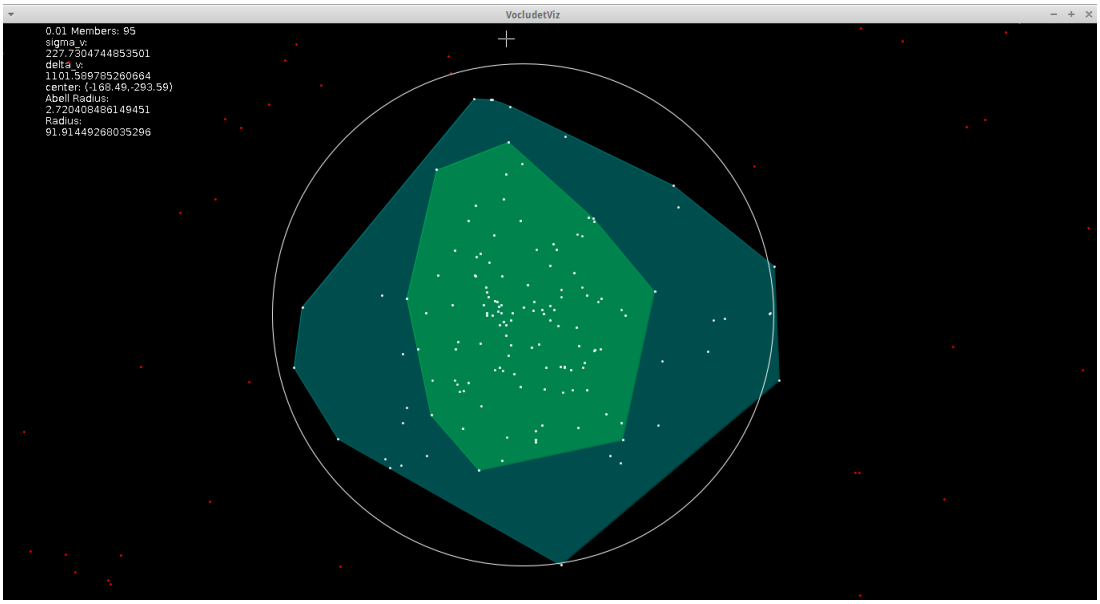

Figura 4.14: Vista de análisis individual de cúmulos

En esta escena también se implementa una animación, que permite observar al pequeño subconjunto de cúmulos y galaxias desde distintos ángulos, lo cual genera un aspecto tridimensional.

4.4.3. Filtro por intervalo de redshift

Para facilitar el análisis del desempeño del algoritmo al variar la distancia de los cúmulos, se implementa un filtro por intervalo de redshift. Esto consiste en que se puede ir variando los límites inferior y superior de redshift que los cúmulos deben tener para ser desplegados. Además, con esto se logra disminuir la cantidad de cúmulos en la vista general. En la figura 4.15 se puede ver el catalogo filtrado por el intervalo de redshift $[0{,}02, 0{,}03]$.

4.4.4. Interacción con la aplicación

La aplicación puede ser controlada a través del teclado para funcionalidades como desplegar los cúmulos, cambiar de proyección, cambiar esquema de color, entre otros. El mouse implementa la funcionalidad de la cámara, como el zoom, movimiento y selección de cúmulos.

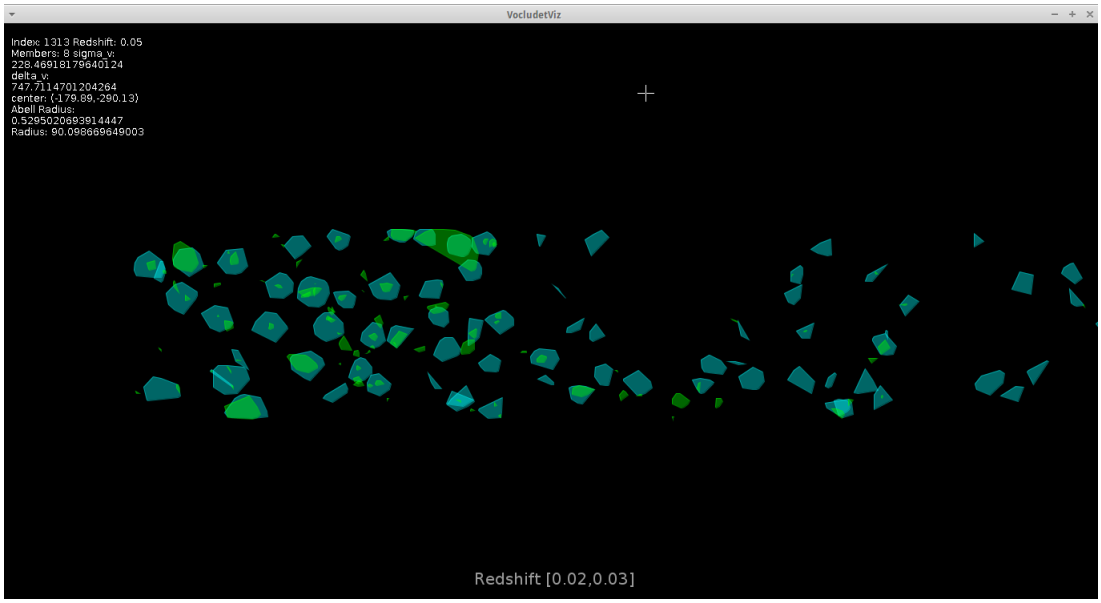

Figura 4.15: Vista con filtro por redshift activo

Adicionalmente se tiene la interacción a través de dispositivos móviles, la cual se enfoca en el análisis de los datos, por lo que implementa sólo un subconjunto de las funcionalidades provistas por mouse y teclado. Este es el caso de la selección de cúmulos a través de su identificador (provisto por el algoritmo Vocludet o la simulación Millennium) o el movimiento general de la cámara.

La implementación del control a través de tablets o celulares tiene como objetivo facilitar la interacción con el wall display, ya que resulta incómodo trabajar con computadores de escritorio o portátiles por las limitaciones de movimiento que esto significa.

4.4.5. Despliegue en wall display

Una vez implementada la aplicación, esta es desplegada en el wall display ubicado en las instalaciones de Inria Chile. La visualización en el wall display permite un mejor análisis al poder observar una mayor cantidad de información, además de proveer la posibilidad de observar un panorama general de la información desde una distancia, o bien ver una mayor cantidad de detalle al acercarse a la pantalla. En las figuras 4.16 y 4.17 se pueden ver fotografías del despliegue de la aplicación en el wall display.

Figura 4.16: Vista general en wall display

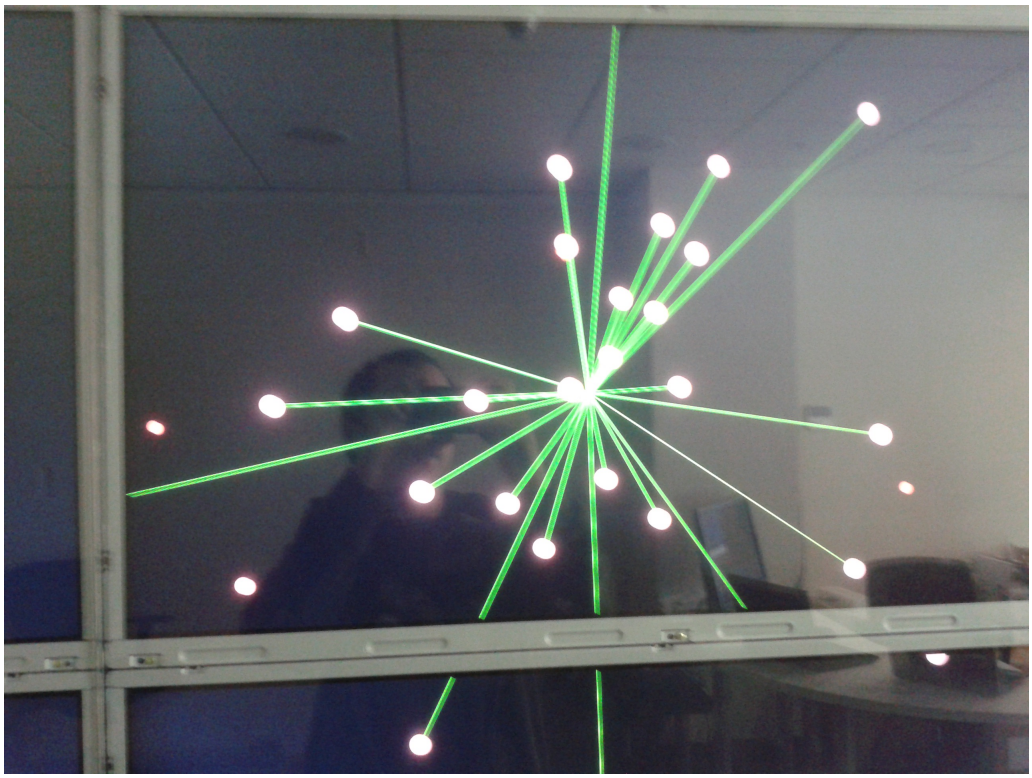

Figura 4.17: Vista de una pantalla individual en wall display

Capítulo 5

Análisis de resultados y validación

En este capítulo se describen los resultados que produce el algoritmo Vocludet al ejecutarlo con los datos de la simulación Millennium, junto con la posterior validación del mismo.

5.1. Análisis de resultados

La ejecución de Vocludet produce un catálogo con un número total de 4546 cúmulos. Este conjunto de datos contiene una gran cantidad de información que debe ser analizada en detalle. En esta sección se describen las características del catálogo generado bajo distintos puntos de vista.

5.1.1. Distribución de tamaños

Es interesante estudiar cómo se relaciona la cantidad de galaxias en los cúmulos detectados con el ranking de su semilla correspondiente (ver sección 2.1.1). En la figura 5.1 se ve como existe una relación entre el ranking del cúmulo y la cantidad de galaxias que contiene. Esto es consistente con el hecho que las semillas de más bajo ranking están ubicadas en regiones del espacio más densamente pobladas.

5.1.2. Distribución de dispersiones de velocidad

Saber cómo se comporta la distribución de las dispersiones de velocidades es importante debido a que uno de los parámetros específicos del algoritmo es el corte de redshift z_{gap} a utilizar (como se explica en la sección 2.1.1, el cual afecta directamente a la dispersión de velocidades. A medida que se disminuye z_{gap}, la dispersión de velocidades también debe disminuir, ya que esto produce cúmulos más concentrados en redshift (no en tamaño angular).

Luego de correr pruebas con distintos valores de z_gap, se observa que los resultados no

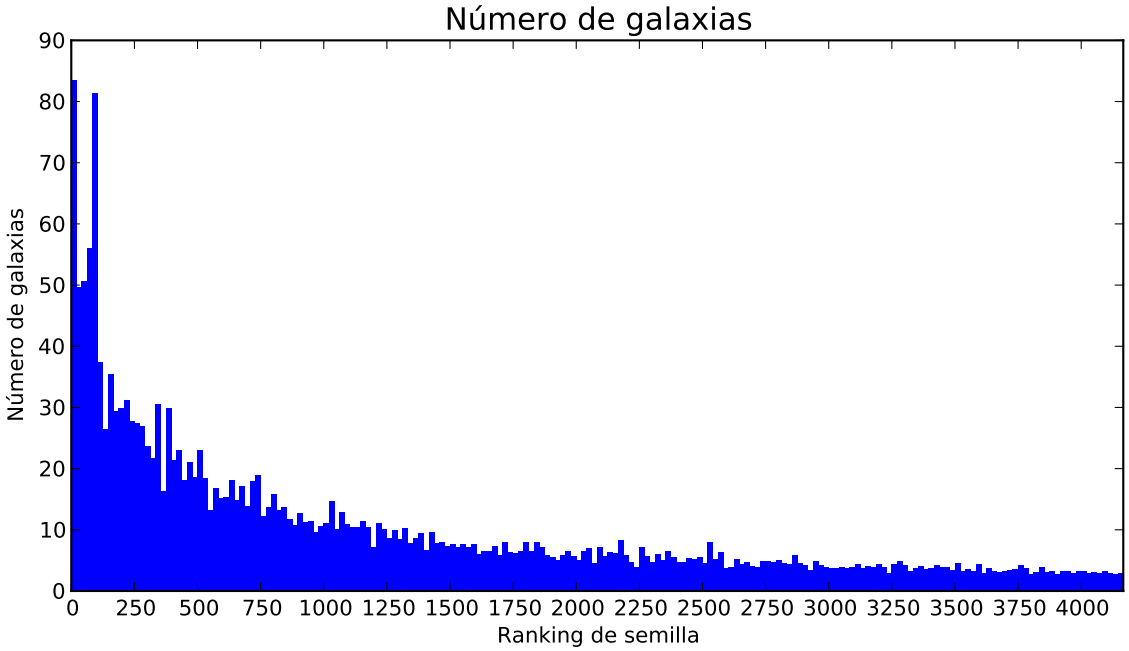

Figura 5.1: Cantidad de galaxias en cúmulos detectados

presentan una sensibilidad importante (detalles en el siguiente capítulo). Debido a esto, se escoge un valor de $z_{gap} = 0{,}0016$, lo que equivale a $v_{gap} = 500 kms^{-1}$.

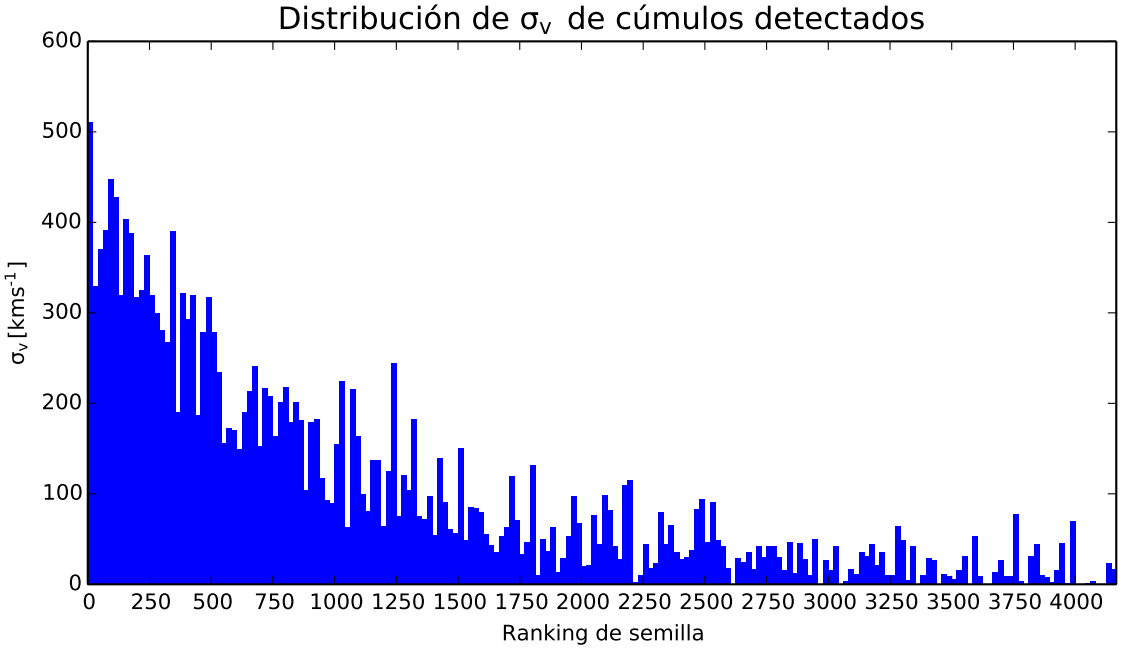

Figura 5.2: Distribución de dispersiones de velocidad de los cúmulos detectados

En la figura 5.2 se puede ver el resultado de la distribución de la dispersión de velocidad para los cúmulos detectados.

5.2. Validación

En este capítulo se detalla el proceso de validación del algoritmo: determinación de desempeño bajo diversas parámetros, condiciones favorables y desfavorables, y análisis visual. Al momento de evaluar la calidad del algoritmo, existe una gran cantidad de parámetros a considerar, cada uno enfocándose a la medición de cierto aspecto en específico de su desempeño. A continuación se describen los considerados en este trabajo.

5.2.1. Tasa de detección

La tasa de detección indica qué porcentaje de los cúmulos detectados tiene correspondencia con algún cúmulo de referencia, es decir, la tasa detecciones consideradas correctas (también llamada sensibilidad o tasa de verdaderos positivos). Debido a que los cúmulos no son entidades singulares (contienen múltiples galaxias) se debe definir un criterio al momento de decidir qué cúmulo se considera como detectado o no. Por ejemplo, se puede utilizar el criterio de la distancia entre centroides de cúmulos: Si se encuentra un cúmulo a una cierta distancia de un cúmulo de referencia y esta distancia es menor a un límite fijado previamente, se considera una detección positiva. Otro posible criterio es el del tamaño de intersección entre cúmulos. En este caso, si el cúmulo detectado contiene más que un cierto porcentaje de galaxias de un cúmulo de refencia, la detección se considera positiva.

Finalmente se elige el segundo criterio por sobre el primero, ya que este último es más débil, puesto que pueden existir 2 cúmulos cuyos centroides estén muy cerca, pero que compartan una cantidad mínima de galaxias. En contraste, utilizando el criterio de intersección, se tiene que con un buen porcentaje de coincidencia, los centroides necesariamente van a estar cerca, debido a los tamaños limitados de los cúmulos.

Para la fijación del criterio de intersección mínima p_m, se analiza como se comporta la muestra ante su variación.

En la figura 5.3 se puede ver cómo varía el porcentaje de detección para distintos valores de p_m. Se decide adoptar el valor $p_m = 20\,\%$ como criterio de detección, ya que representa una fracción considerable del cúmulo de referencia y permite maximizar el número de detecciones. Además, debido a que el algoritmo entrega sus resultados según el orden de las semillas generadas en la primera etapa del algoritmo, se decide restringir análisis posteriores a las primeras 500 semillas, las cuales representan las zonas de más alta densidad. Bajo estas consideraciones, se obtiene un porcentaje de detección de 84,16 %.

Otra medición importante que se desprende directamente de la tasa de detección es la de los cúmulos falsos positivos. Estos corresponden a aquellas agrupaciones de galaxias que fueron detectadas como cúmulos, pero que no se encuentran en el catálogo de referencia. Esta tasa corresponde al recíproco de la tasa de detección, es decir, un 15,84 %

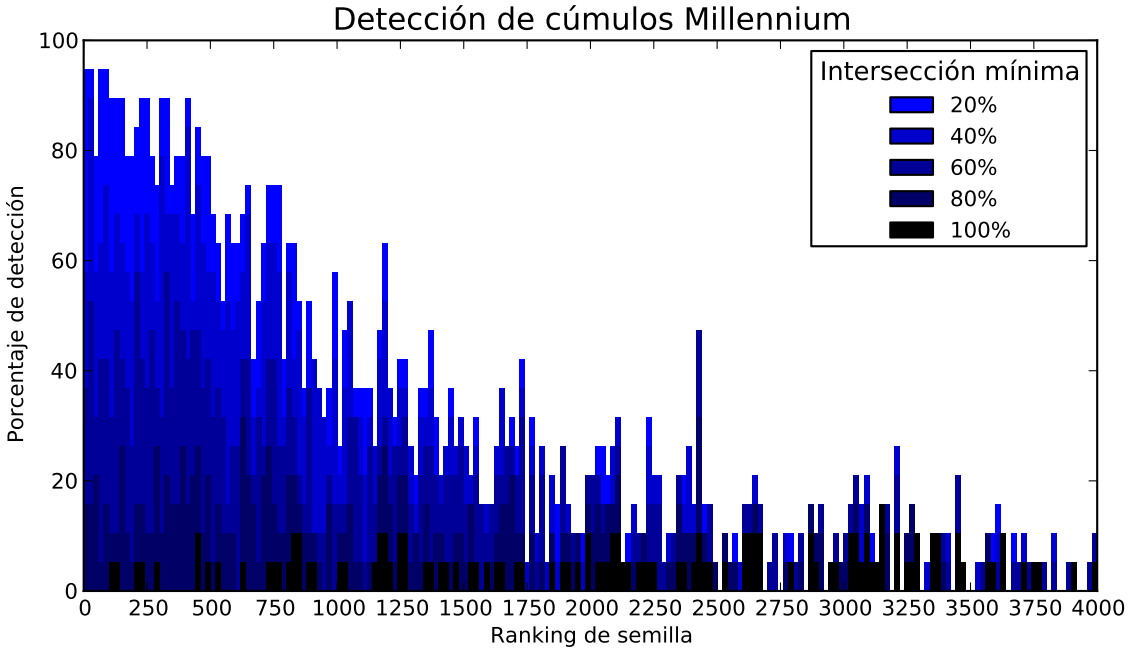

Figura 5.3: Tasa de detección según criterio de intersección mínima

5.2.2. Completitud

La completitud de la detección representa el porcentaje de galaxias de los cúmulos de referencia que se recupera a través del algoritmo. Por ejemplo, para un cúmulo de referencia en específico, una tasa de completitud de un 50 % indica que el cúmulo detectado correspondiente recupera la mitad de sus galaxias.

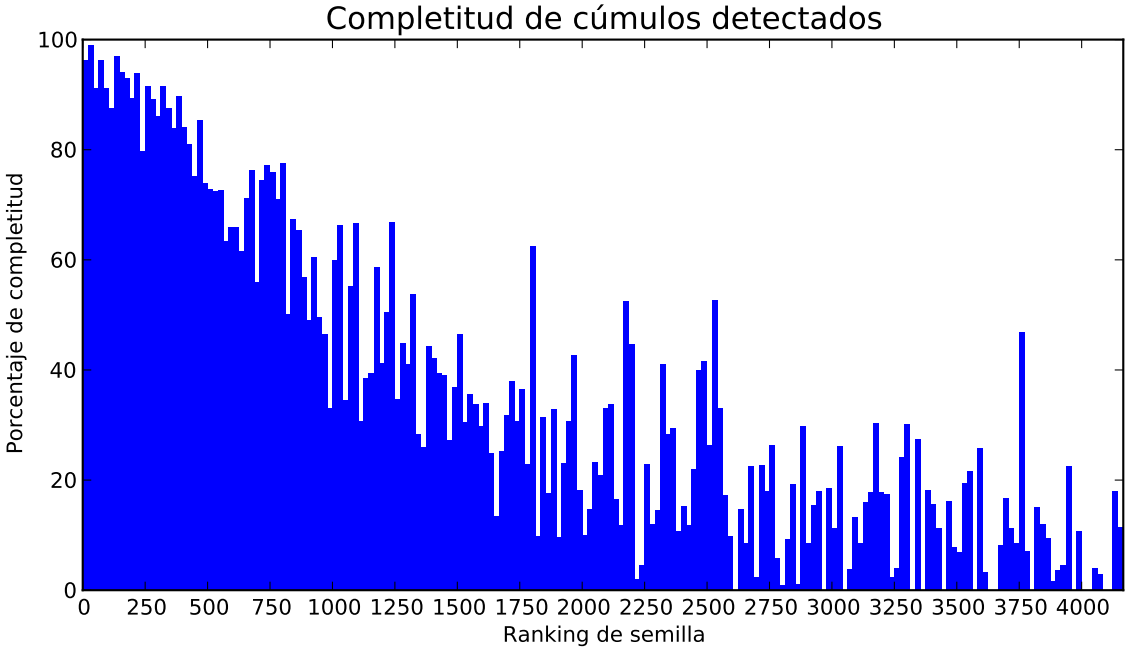

Figura 5.4: Tasa de completitud de cúmulos detectados

La figura 5.4 indica el comportamiento de la completitud según el ranking de semillas. Se puede ver que, al igual que con la tasa de detección, el porcentaje de completitud muestra una alta dependencia con el ranking de semillas. La tasa de completitud del catálogo se calcula como un promedio ponderado (por número de galaxias) de los cúmulos correspondientes a las primeras 500 semillas, y se obtiene como resultado un 90,08 %.

5.2.3. Galaxias Falsos positivos

Un tercer criterio de evaluación corresponde a la tasa de falsos positivos. Una galaxia es considerada como falso positivo si es que el algoritmo de detección indica que es parte de un cúmulo cuando en realidad esta no pertenece a él. La tasa corresponde al promedio ponderado (por número de galaxias) de los falsos positivos del total de cúmulos a estudiar. En la figura 5.2.3 se puede ver la distribución de la tasa de falsos positivos según el ranking de las semillas entregadas por el algoritmo.

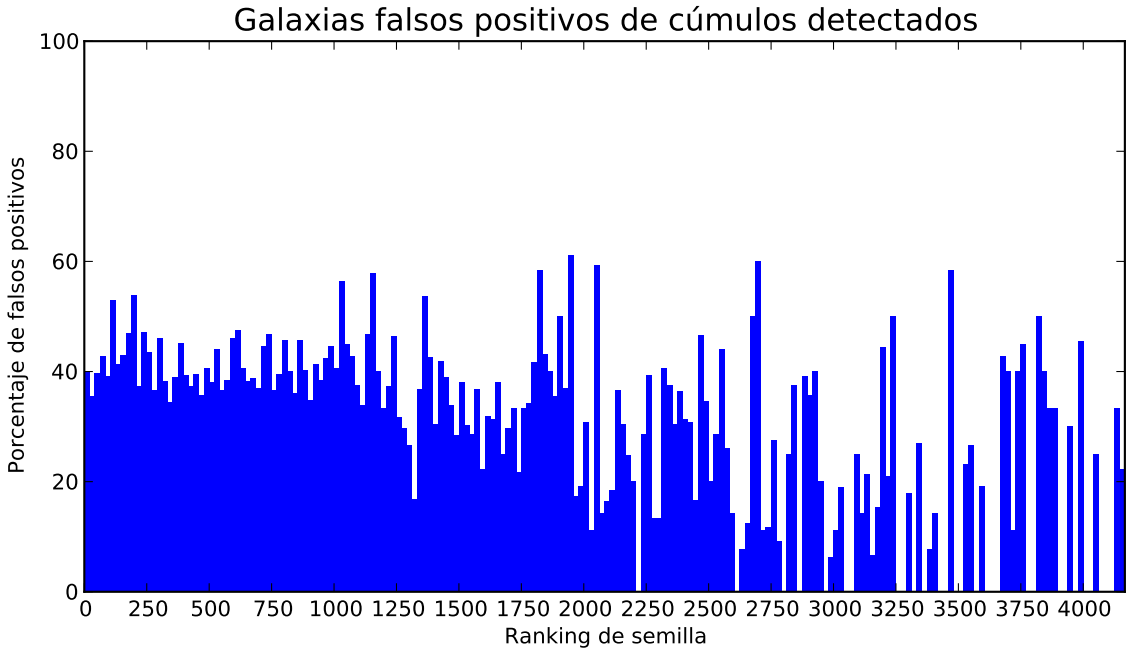

Figura 5.5: Tasa de galaxias falsos positivos de cúmulos detectados

La tasa de galaxias falsos positivos de los primeros 500 cúmulos corresponde a un 43,22 %. Este valor confirma que Vocludet está detectando cúmulos más grandes que los de referencia, como se puede apreciar en la visualización, en la figura 4.13.

5.2.4. Número de galaxias

Otro aspecto estudiado es el de la cantidad de galaxias de cada cúmulo. Para esto se compara el número de galaxias de cada cúmulo Millennium detectado con el número de galaxias del respectivo cúmulo Vocludet.

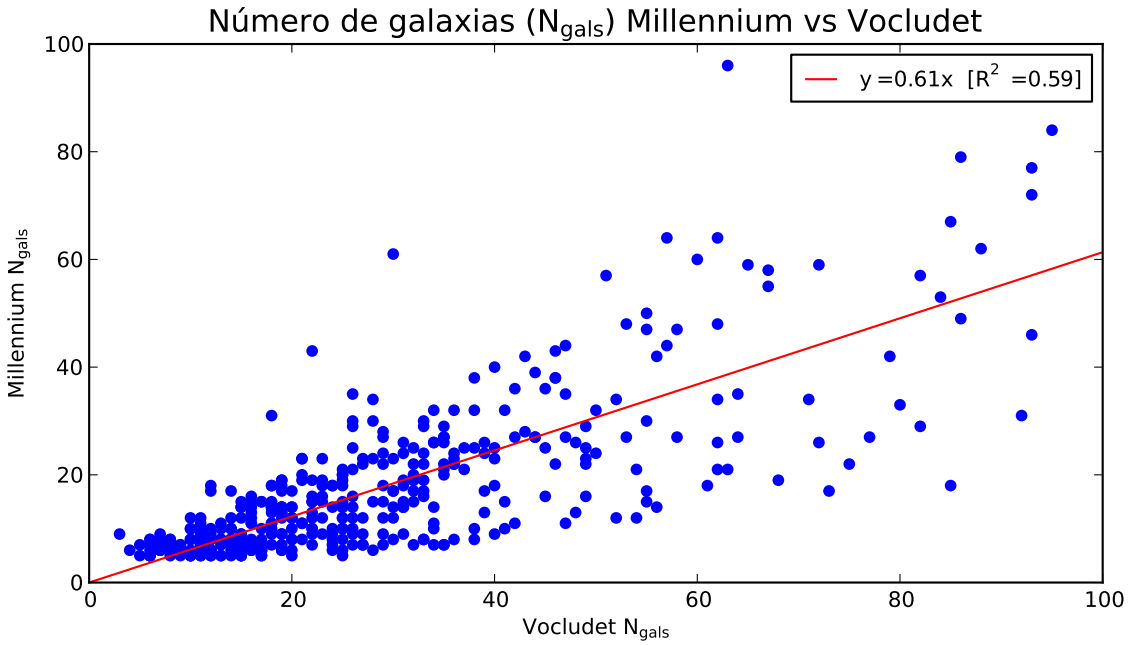

Figura 5.6: Numero de galaxias cúmulos Millennium vs Vocludet

Además se realiza un ajuste lineal (forzando el paso por el origen), que entrega una pendiente de 0,61, como se aprecia en la figura 5.6. Esta pendiente es concordante con el valor mencionado en la sección 5.2.3, que indica que los cúmulos Vocludet poseen 40 % de falsos positivos.

5.2.5. Análisis de masa y redshift

Debido a que la masa de los cúmulos está relacionada con la cantidad de galaxias y su densidad, se espera que Vocludet produzca mejores tasas de detección para cúmulos de alta masa. Además se requiere estudiar el comportamiento del algoritmo según la distancia a la que se encuentran los cúmulos (redshift). En la figura 5.7 se grafica la fracción de cúmulos detectados por intervalos de masa y redshift. Los intervalos de masa son seleccionados por cuartiles, es decir, cada intervalo posee un 25 % del total de cúmulos. Se puede observar una tendencia a obtener una mayor tasa de detección a medida que aumenta la masa de los cúmulos. Esta tendencia es más acentuada para el intervalo más masivo, en donde se alcanza el 100 % de recuperación de cúmulos más cercanos ($redshift < 0{,}04$).

En relación al comportamiento con respecto al redshift, en el caso de los cúmulos más masivos la tasa de detección disminuye a medida que aumenta el redshift. Para los cúmulos menos masivos no existe una tendencia clara, pero en general la tasa de detección oscila en torno al 75 %.

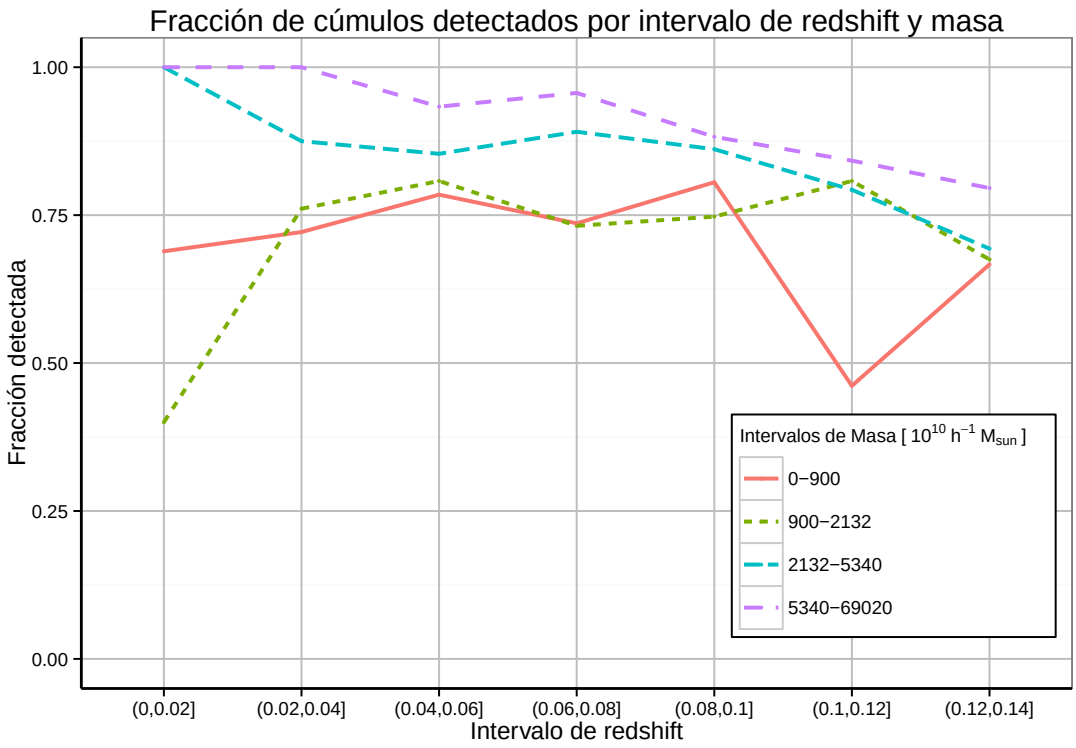

Figura 5.7: Tasa de detección por intervalos de masa

Capítulo 6

Conclusiones

En primer lugar se realiza un trabajo de recopilación y análisis de información con respecto al estado actual de la astronomía y su necesidad de contar con herramientas de procesamiento de datos que faciliten el estudio de las grandes cantidades de información producida día a día. Además se lleva a cabo un estudio sobre el algoritmo Vocludet de detección de cúmulos de galaxias: su funcionalidad, requerimientos y resultados.

Por otra parte, se logra obtener y procesar datos simulados de galaxias y cúmulos. Esto provee un catálogo de referencia que permite posteriormente visualizarlos y ejecutar el algoritmo Vocludet sobre ellos, para luego pasar a una etapa de validación.

Con el trabajo realizado en esta memoria se tiene la posibilidad de visualizar datos astronómicos como galaxias y cúmulos galaxias. Si bien la visualización está diseñada para una aplicación bastante específica, ésta puede ser extendida o modificada de forma mínima para aceptar otros tipos de datos que posean información de posición espacial 2 o 3 dimensional y agrupaciones en las que se tenga la información de pertenencia. Un ejemplo de esto podría ser el visualizar el resultado de algoritmos aplicados a la minería de datos como k-means.

Con respecto al algoritmo Vocludet, se realiza su validación utilizando datos simulados, obteniendo resultados positivos en todos los aspectos, entre los que destacan la tasa de detección de cúmulos de un 84,16 % y una tasa de completitud de los cúmulos detectados de 90,08.

6.1. Trabajo futuro

Si bien la visualización fue esencial durante el proceso de estudio del conjunto de datos de la simulación y bastante importante en el análisis de los resultados, hace falta el desarrollo de una forma de unir lo anterior con el proceso de validación estadístico final del algoritmo.

Además se propone como trabajo futuro el seguir con la optimización y refinamiento del algoritmo Vocludet, junto con su ejecución sobre otros conjuntos de datos.

Glosario

ΛCDM En cosmología, el modelo Lambda-CDM o ΛCDM (en inglés: Lambda-Cold Dark Matter) representa el modelo de concordancia del Big Bang que explica las observaciones cósmicas de la radiación de fondo de microondas, así como la estructura a gran escala del universo y las observaciones realizadas de supernovas, arrojando luz sobre la explicación de la aceleración de la expansión del Universo. Es el modelo conocido más simple que está de acuerdo con todas las observaciones.. 7

Coordenadas ecuatoriales Las coordenadas ecuatoriales (absolutas) son un tipo de coordenadas celestes que determinan la posición de un objeto en la esfera celeste respecto al ecuador celeste. Se denominan declinación (DE) y ascensión recta (RA) y son equivalentes a la latitud y longitud geográficas.. 20

Materia oscura Se denomina materia oscura a la hipotética materia que no emite suficiente radiación electromagnética para ser detectada con los medios técnicos actuales, pero cuya existencia se puede deducir a partir de los efectos gravitacionales que causa en la materia visible, tales como las estrellas o las galaxia. La materia oscura desempeña un papel central en la formación de estructuras y la evolución de galaxias y cúmulos.. 7

Redshift En física y astronomía, el redshift es un fenómeno en el cual la luz de un objeto es desplazada hacia el rojo en el espectro electromagnético. Se observa como un aumento en la longitud de onda o, equivalentemente, una disminución en la frecuencia de la radiación electromagnética. Comúnmente se le denota por una cantidad adimensional llamada z.

Una propiedad útil se ve reflejada en el hecho que el redshift de las fuentes de luz lejanas como galaxias, quasares o gas intergaláctico, aumenta proporcionalmente con la distancia hacia el objeto. Esto permite usarlo como una medida de distancia, por lo que dos coordenadas angulares más el redshift posicionan un objeto en el espacio tridimensional.. 5

Teselación de Voronoi En matemáticas, una teselación de Voronoi (también llamada diagrama de Voronoi o descomposición de Voronoi) es un tipo de descomposición de un espacio métrico, determinada por las distancias a un conjunto discreto de objetos en el espacio, e.g., un conjunto de puntos.

Definición: Para cualquier conjunto discreto S de puntos en un espacio euclidiano, y para cada punto x, existe un punto de S más cercano a x. Si S contiene sólo 2 puntos, a y b, entonces el conjunto de todos los puntos equidistantes a a y b es un hiperplano. Este hiperplano constituye el límite entre el conjunto de todos los puntos más cercanos a a que a b. En general, el conjunto de todos los puntos más cercanos a un punto c de S que a cualquier otro punto de S es el interior de un politopo llamado celda de Voronoi de c. El conjunto de todos estos politopos tesela todo el espacio, y corresponde a la teselación de Voronoi del conjunto S. . iii, 2

Bibliografía

[1] Julien Altieri. From a laptop to a wall-sized display. master, Üniversité Paris-Sud, 2011. 22 pages.

[2] J. Blaizot, Y. Wadadekar, B. Guiderdoni, S. T. Colombi, E. Bertin, F. R. Bouchet, J. E. G. Devriendt, and S. Hatton. MoMaF: the Mock Map Facility. , 360:159–175, June 2005.

[3] Olivier Chapuis, Anastasia Bezerianos, and Stelios Frantzeskakis. Smarties: An Input System for Wall Display Development. In ACM, editor, *Proceedings of the 32nd international conference on Human factors in computing systems*, CHI '14, pages 2763–2772, Toronto, Canada, April 2014.

[4] Olivier Chapuis, Anastasia Bezerianos, and Stelios Frantzeskakis. Smarties: An input system for wall display development. In *CHI '14: Proceedings of the 32nd international conference on Human factors in computing systems*, CHI '14, pages 2763–2772. ACM, 2014.

[5] R. G. Clowes, K. A. Harris, S. Raghunathan, L. E. Campusano, I. K. Söchting, and M. J. Graham. A structure in the early Universe at z

1.3 that exceeds the homogeneity scale of the R-W concordance cosmology. , 429:2910–2916, March 2013.

[6] S. Eilemann, M. Makhinya, and Renato Pajarola. Equalizer: A scalable parallel rendering framework. *Visualization and Computer Graphics, IEEE Transactions on*, 15(3):436–452, May 2009.

[7] Z. Ivezic, J. A. Tyson, T. Axelrod, D. Burke, C. F. Claver, K. H. Cook, S. M. Kahn, R. H. Lupton, D. G. Monet, P. A. Pinto, M. A. Strauss, C. W. Stubbs, L. Jones, A. Saha, R. Scranton, C. Smith, and LSST Collaboration. LSST: From Science Drivers To Reference Design And Anticipated Data Products. In *American Astronomical Society Meeting Abstracts #213*, volume 41 of *Bulletin of the American Astronomical Society*, January 2009.

[8] G. Lemson and t. Virgo Consortium. Halo and Galaxy Formation Histories from the Millennium Simulation: Public release of a VO-oriented and SQL-queryable database for studying the evolution of galaxies in the LambdaCDM cosmogony. *ArXiv Astrophysics*

e-prints, August 2006.

[9] M. Milkeraitis, L. van Waerbeke, C. Heymans, H. Hildebrandt, J. P. Dietrich, and T. Erben. 3D-Matched-Filter galaxy cluster finder - I. Selection functions and CFHTLS Deep clusters. , 406:673–688, July 2010.

[10] Emmanuel Pietriga. A toolkit for addressing hci issues in visual language environments. *IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)*, 00:145–152, 2005.

[11] D. Pizarro Pizarro. Galaxy cluster detection using nonparametric maximum likelihood estimation of features in voronoi tessellations. Master's thesis, Universidad de Chile, 2007.

[12] A. I. Zabludoff, J. P. Huchra, and M. J. Geller. The kinematics of Abell clusters. , 74:1–36, September 1990.