

Universidad Central «Marta Abreu» de las Villas
Facultad Matemática Física y Computación
Licenciatura en Ciencia de la Computación



TRABAJO DE DIPLOMA

Análisis visual de datos climáticos en Sistemas de Información Geográfica.

Autor: Rainer Martínez Fraga

Tutor: MSc. Romel Vázquez Rodríguez

Santa Clara

2014



Hago constar que el presente trabajo fue realizado en la Universidad Central Marta Abreu de Las Villas como parte de la culminación de los estudios de la especialidad de Ciencia de la Computación, autorizando a que el mismo sea utilizado por la institución, para los fines que estime conveniente, tanto de forma parcial como total y que además no podrá ser presentado en eventos ni publicado sin la autorización de la Universidad.

Firma del autor

Los abajo firmantes, certificamos que el presente trabajo ha sido realizado según acuerdos de la dirección de nuestro centro y el mismo cumple con los requisitos que debe tener un trabajo de esta envergadura referido a la temática señalada.

Firma del tutor

Firma del jefe de
laboratorio

Resumen

Las secuencias temporales de datos en forma de mallas regulares constituyen un reto en cuanto a su procesamiento y análisis de manera consistente y efectiva. La mayoría de los métodos existentes son incapaces de realizar un análisis abarcador de información espacio-temporal de múltiples variables a la vez. En este trabajo se aborda la problemática del análisis visual de secuencias temporales de variables climáticas en forma de mallas regulares. Mediante la utilización de técnicas de visualización científica se logra analizar eficazmente toda la información almacenada en las secuencias de datos. Los conceptos y métodos elaborados en este trabajo fueron implementados como una extensión al Sistema de Información Geográfica gvSIG. Adicionalmente se muestra un caso de estudio con datos climáticos mundiales de un período de 112 años. El uso de las herramientas desarrolladas hizo posible la extracción de importantes tendencias, correlaciones, y patrones relacionados con fenómenos climáticos.

Abstract

Temporal data sequences in form of regular grids make a challenge for effective and consistent processing and analysis. Most of the existent methods are unable to make a thorough analysis of spatio-temporal information with multiple variable at a time. In this work we present the problem of visual analysis on temporal sequences of climate variable in the form of regular grids. By means of Scientific Visualization techniques is possible to achieve an effective analysis of the whole information stored in the data sequences. The resulting concepts and methods were implemented as an extension of the Geographical Information System gvSIG. Additionally, a case study on world climate data over a 112 year period is shown. The use of the developed tools made possible the extraction of important tendencies, correlations and patterns related with climate phenomena.

TABLA DE CONTENIDOS

Introducción	1
1. Visualización científica y sistemas de información geográfica	6
1.1. Visualización Científica	6
1.2. Visualización para datos multiparamétricos	8
1.2.1. Técnicas geométricas	8
1.2.2. Técnicas basadas en íconos	12
1.2.3. Técnicas orientadas a píxel	15
1.3. Representación del tiempo	19
1.4. Visualización de grandes volúmenes de datos	20
1.5. Sistemas de Información Geográfica	22
1.5.1. GRASS GIS	25
1.5.2. QGIS	25
1.5.3. gvSIG	26
1.6. Geovisualización	26
1.6.1. Snap-Together Visualization	27
1.6.2. GeoVISTA <i>Studio</i>	28
1.6.3. VIS-STAMP	28
1.6.4. ArcView - xGobi	29
1.7. Formatos de datos científicos utilizados en la Visualización Científica	29
1.7.1. <i>Hierarchical Data Format</i> (HDF)	30
1.7.2. <i>Common Data Format</i> (CDF)	31
1.7.3. <i>Network Common Data Form</i> (netCDF)	31

1.7.4. <i>Flexible Image Transport System (FITS)</i>	32
1.8. Conclusiones	33
2. Diseño e implementación del módulo de geovisualización	34
2.1. Diseño de la extensión	34
2.1.1. Análisis de actores y casos de uso	35
2.1.2. Diagramas de transición de estados	37
2.1.3. Acceso a datos	39
2.1.4. Visualización coordinada	41
2.2. Implementación de la extensión	42
2.2.1. Elección de las tecnologías necesarias para la implementación	42
2.2.2. Metodología para la adición de nuevas técnicas de visualización	44
2.2.3. Consideraciones sobre las técnicas implementadas	46
2.3. Conclusiones	48
3. Uso y validación del sistema	49
3.1. Uso del sistema	49
3.2. Características fundamentales	50
3.2.1. Panel de configuración general de las técnicas	51
3.2.2. Espiral de tiempo	52
3.2.3. Coordenadas de estrella	54
3.2.4. Vista circular	55
3.2.5. Figuras con palillos	56
3.3. Caso de estudio: Datos climáticos mundiales	58
3.4. Conclusiones	62
Conclusiones	63
Recomendaciones	65

LISTA DE FIGURAS

1.1. Coordenadas paralelas (izquierda) y coordenadas paralelas en 3D (derecha).	10
1.2. Vista circular de varios atributos a través del tiempo.	11
1.3. Gráfico en coordenadas de estrella	13
1.4. Algunos parámetros configurables en una cara de Chernoff	14
1.5. Figura con palillos	15
1.6. Técnicas de visualización orientadas a píxel mostrando un conjunto de datos de 10 dimensiones.	17
1.7. Dos representaciones de una misma serie de tiempo. El gráfico en espiral de la derecha permite detectar un patrón cíclico aparente. .	20
1.8. Visualización de un conjunto de datos multivariados sobre países mediante <i>Table Lens</i> con seudocoloración en dos tonos (<i>Two-Tone Pseudo Coloring</i> , TTPC) y agrupamiento híbrido. Fuente: (John, Tominski y Schumann 2008).	21
1.9. Diferentes representaciones de los resultados obtenidos mediante el agrupamiento con mapa autoorganizado (<i>Self-Organizing Map</i> , SOM). .	23
2.1. Diagrama de casos de uso del módulo.	36
2.2. Diagrama de transición de estados correspondiente al caso de uso «Visualizar de forma coordinada».	38
2.3. Diagrama de clases del paquete <code>uclv.gvsig.sv.datasource</code>	40
2.4. Diagrama de clases del paquete <code>uclv.gvsig.sv.view</code>	42
3.1. Creación de una vista de visualización coordinada.	51

3.2. Opciones para la adición de un proyecto de visualización coordinada a una vista de visualización.	52
3.3. Vista de visualización.	53
3.4. Panel de configuración general.	54
3.5. Diálogo de filtrado de datos.	55
3.6. Visualización del gráfico en espiral.	56
3.7. Visualización del gráfico en coordenadas de estrella.	57
3.8. Visualización en vista circular.	58
3.9. Visualización de figuras con palillos.	59
3.10. Visualización de registros climáticos de la región oriental de Cuba utilizando patrones recursivos y espiral de tiempo.	60
3.11. Visualización de registros climáticos de la región oriental de Cuba utilizando <i>Table Lens</i> . El foco se encuentra aplicado al registro 753 (octubre de 1963).	61

LISTA DE TABLAS

2.1. Nuevas técnicas implementadas para el módulo de visualización científica	35
3.1. Variables climáticas incluidas en el conjunto de datos analizado. . .	60

INTRODUCCIÓN

En la actualidad los procesos de análisis de datos poseen un grado de complejidad en constante crecimiento. Los grandes volúmenes de datos que se manejan motivan el desarrollo de nuevas técnicas que permitan una mejor comprensión de los mismos. En particular un Sistema de Información Geográfica (SIG) se enfoca en la búsqueda de patrones geográficos en los datos y en las relaciones entre las distintas características de estos.

La diferencia entre el análisis estadístico y el análisis de los SIG recae principalmente en los tipos de datos con los que tratan y en los métodos usados. En ambos casos, el análisis de datos parece estar conducido por preguntas: las preguntas motivan el análisis a realizar, determinan la elección de los datos y de los métodos, y afectan la interpretación de los resultados.

Según el Centro Nacional de Información Geográfica y Análisis (*National Center of Geographic Information and Analysis*, NCGIA), un SIG es un sistema de hardware, software y procedimientos elaborados para facilitar la obtención, gestión, manipulación, análisis, modelado, representación y salida de datos espacialmente referenciados, para resolver problemas complejos de planificación y gestión. Sin duda, se trata de una definición funcionalista en la que se reflejan, aparte de la tecnología, todos aquellos aspectos y partes que conforman un SIG.

La visualización científica ha sido un área de interés creciente en los últimos años. El incremento constante de los volúmenes de datos generados en muchos campos de aplicación crea la necesidad de elaborar herramientas que permitan extraer de ellos información de manera eficiente, así como generar imágenes cada vez más

sofisticadas debido al aumento sostenido de la potencia de las interfaces gráficas modernas. Unido al desarrollo de nuevas técnicas de visualización se han creado numerosas utilidades que hacen uso de ellas, tanto en forma de bibliotecas como de programas.

En los últimos años, los avances en la tecnología han facilitado la obtención de grandes cantidades de información. Mediante imágenes de satélite, estaciones de medición de alta precisión, supercomputadoras o cualquier otra fuente de este tipo, se generan a diario volúmenes de datos muy grandes y complejos. Sin embargo solo una pequeña fracción de estos datos llega a ser analizada.

La visualización científica busca encontrar una representación visual apropiada para un conjunto de datos que permita mayor efectividad en el análisis y evaluación de los mismos. Permite a su vez la transformación de los datos numéricos o simbólicos y la información en imágenes geométricas generadas por computadora. Es una metodología para interpretar, a través de una imagen en la computadora, tanto datos de mediciones como los generados por modelos computacionales. La investigación y el desarrollo de la visualización científica se han centrado en cuestiones relacionadas con el renderizado de gráficos en tres dimensiones, animaciones de series temporales y visualización interactiva en tiempo real.

Una clase especial de datos son los datos multiparamétricos (datos multidimensionales o datos multivariados). Los datos multiparamétricos son los que poseen m variables o dimensiones de datos escalares distribuidos sobre puntos en el espacio de observación. Estas variables pueden ser cuantitativas o cualitativas y a su vez ordinales o nominales. La visualización científica ofrece grandes ventajas sobre otros métodos de análisis de datos cuando los mismos poseen más de dos dimensiones o variables, puesto que permite la visualización de todas las variables que se deseen al mismo tiempo. A este tipo de visualización se le conoce como visualización de datos multiparamétricos.

Por las ventajas señaladas anteriormente de los SIG y del visualización científica en la visualización y análisis de grandes volúmenes de datos, una integración de

estas dos tecnologías sería muy provechosa para el trabajo con datos multiparamétricos.

Planteamiento del problema

En el laboratorio de Computación Gráfica del Centro de Estudios de Informática (CEI) de la Universidad Central «Marta Abreu» de las Villas (UCLV) se cuenta con datos climáticos asociados a los cinco continentes del planeta tomados durante cien años almacenados en el formato de datos científico netCDF. Se precisa realizar la visualización de estos datos con el objetivo de realizar comparaciones, identificar patrones, encontrar correlaciones, detectar anomalías, variabilidad y las diferentes tendencias que se pueden presentar en las variables climáticas a lo largo del tiempo, teniendo en cuenta su ubicación espacial.

Dado que existen múltiples formatos para el almacenamiento de grandes volúmenes de datos, la cantidad de técnicas de visualización científica existentes y las particularidades de los SIG, se formularon varias preguntas de investigación.

Preguntas de investigación

- ¿Cuáles son las técnicas de visualización científica para datos multiparamétricos que brindan un mejor análisis de datos climáticos?
- ¿De qué manera pueden implementarse nuevas técnicas de visualización científica en SIG?
- ¿De qué manera puede lograrse un soporte uniforme para la formatos de datos científicos utilizados como fuentes de datos?
- ¿Qué modificaciones deben realizarse a las aplicaciones realizadas con anterioridad para la solución de esta problemática?

- ¿Qué hipótesis o correlaciones se pueden extraer de los datos a través de las técnicas de visualización seleccionadas?

Teniendo en cuenta la problemática planteada y las preguntas de investigación se realizó el planteamiento de los objetivos.

Objetivo general

Implementar nuevas técnicas de visualización científica en un Sistema de Información Geográfica para el análisis multivariado de grandes volúmenes de datos climáticos.

Objetivos específicos

- Seleccionar las técnicas de visualización científica más adecuadas para ser añadidas a un SIG.
- Implementar e integrar las técnicas de visualización seleccionadas como parte de un módulo de visualización científica en un Sistema de Información Geográfica.
- Extender el soporte del módulo para formatos de datos científicos.
- Extraer información de datos climáticos mundiales mediante técnicas de visualización científica.

Justificación de la investigación

Actualmente existen diversos formatos de datos científicos para almacenar grandes volúmenes de información, lo cual nos obliga a realizar una investigación para

ver sus ventajas y desventajas y seleccionar entre todos el que más se adapte a nuestro propósito. Además es necesario elegir las técnicas de visualización que nos faciliten el análisis espacio-temporal de los datos existentes.

Estas técnicas seleccionadas se visualizarán sobre el lugar que se quiere analizar y en caso de que la técnica no pueda ser visualizada sobre el mapa, podrán ser mostradas de manera independiente, siempre y cuando se sepa a qué localización corresponden. Estas ideas podrán ser aplicadas en cualquier tipo de problemas que contengan información multivariada de datos puntuales en el espacio.

Viabilidad de la investigación

El estado actual de las técnicas de visualización científica y el desarrollo de los SIG ofrece una amplia gama de ideas a desarrollar en este trabajo. Para el desarrollo de esta investigación se cuenta con los recursos necesarios para acometer las tareas propuestas, que incluyen los datos climáticos, los códigos fuentes de los SIG, la información cartográfica y la capacidad de procesamiento de datos y de gráficos. Por otro lado se cuenta con el apoyo del grupo de investigación Computación Gráfica del Centro de Estudios de Informática, que tiene suficiente experiencia en el área.

Hipótesis general

La utilización de un formato de datos científicos para almacenar grandes volúmenes de datos multiparamétricos permite reducir su tamaño en disco y mejorar el trabajo con estos. La integración de técnicas de visualización científica con SIG en una misma aplicación permite visualizar una gran cantidad de información multivariada y amplia en el tiempo e interactuar con la misma a través de las diferentes técnicas, asociándolas a lugares puntuales en el espacio.

1 VISUALIZACIÓN CIENTÍFICA Y SISTEMAS DE INFORMACIÓN GEOGRÁFICA

En el presente capítulo se tratan las generalidades de la visualización científica y específicamente la visualización de datos multiparamétricos y las series temporales. Se abordan las principales características de los Sistemas de Información Geográfica así como una descripción de sus exponentes libres más utilizados. Además se brindan elementos que evidencian la integración de la visualización científica con los Sistemas de Información Geográfica en el emergente campo de la geovisualización, incluyendo el soporte para datos científicos.

1.1. Visualización Científica

La ciencia ha desarrollado diversos métodos para la obtención de información, y uno de ellos se basa en la creación de imágenes a partir de los datos. Este método, conocido como visualización, ha sido utilizado como vía natural para mostrar información (Hansen y Johnson 2005). La visualización por computadora es un proceso de distribución de las representaciones hechas por la computadora a representaciones preceptuales, eligiendo técnicas de codificación para maximizar el entendimiento y comunicación con los seres humanos.

La visualización científica es la transformación de datos científicos y abstractos en imágenes. Es una forma especial de la visualización que procura encontrar una representación visual apropiada para un conjunto de datos que permita mayor efecti-

vidad en el análisis y evaluación de los mismos. Simplifica el análisis, comprensión y la comunicación de modelos, conceptos y datos en la ciencia y la ingeniería.

La visualización de datos permite alcanzar diferentes metas. La naturaleza del objetivo que se desee está en relación directa al conocimiento que se tenga sobre los datos iniciales. Los objetivos pueden ser los siguientes (Theisel 2000; Abello y Korn 2002)

- Análisis exploratorio.
- Análisis confirmativo.
- Presentación de información.

En el análisis exploratorio se tiene un conjunto de datos sin una hipótesis específica. Estos datos se someten a un proceso de búsqueda interactiva de información que va a arrojar como resultado una visualización que soporte una hipótesis sobre el conjunto de datos. En el análisis confirmativo se tiene un conjunto de datos sobre los que se plantea una hipótesis. Se realiza un procesamiento de dichos datos que genera una visualización mediante la cual se pueda validar o refutar la hipótesis que se tenía de ellos. La presentación de información parte de hechos que son fijos *a priori* y que se desean enfatizar y mostrar con extrema calidad.

El análisis exploratorio y el confirmatorio incluyen el análisis estadístico de los datos, la simulación y la educación. Dentro del área del análisis existen aplicaciones para el control de calidad, análisis y proyecciones financieras, análisis de esfuerzos, etc. Esta última podría considerarse dentro del área de simulaciones, junto con la de modelos atmosféricos. En cuanto a la educación, se tienen desde demostraciones matemáticas, hasta modelos de física cuántica y planetarios.

El análisis visual de datos es un nuevo enfoque que se puede utilizar con cualquiera de los objetivos anteriores. Como se ha mencionado anteriormente, integra tanto la percepción humana como los métodos computacionales automáticos, lo que permite una mejor comprensión y análisis de grandes y complejos conjuntos de datos.

1.2. Visualización para datos multiparamétricos

Existen una serie de problemas en que cada punto de dato contiene más de un atributo, estos atributos pueden ser fechas, precios o valores descriptivos. A este tipo de datos se les llama multiparamétricos y se encuentran generalmente en aplicaciones de minería de datos, estadísticas e inteligencia artificial (DA Keim 2002b). Los datos multiparamétricos, también llamados multidimensionales o datos n -dimensionales, consisten en un número de n registros donde cada uno está definido por un vector de d valores. Estos datos pueden ser vistos como una matriz de $n \times d$, donde cada fila representa un registro y cada columna representa una observación, variable o dimensión (M. O. Ward 2002).

El objetivo fundamental de los métodos de visualización para datos multiparamétricos es lograr que las representaciones revelen correlaciones o patrones entre los atributos (Theisel 2000; Eick 2000; DA Keim 2002b). Con este fin existe actualmente una amplia gama de técnicas de visualización, para las cuales se han creado además diversas mejoras. Las técnicas pueden ser clasificadas en geométricas, basadas en íconos, basadas en píxel y proyecciones (Theisel 2000; DA Keim 2002b) entre otras.

1.2.1. Técnicas geométricas

Las técnicas geométricas se basan en el establecimiento de una relación entre los datos correspondientes a los atributos y un espacio geométrico (Mazza 2009). Una de las formas más sencillas de este tipo de técnica es el diagrama de dispersión, en el cual se deben relacionar dos variables con cada uno de los ejes coordenados. No obstante la mayoría de las técnicas desarrolladas bajo esta categoría logran poder representar una cantidad arbitraria de atributos.

Matrices de diagramas de dispersión

Para datos multiparamétricos es muy frecuente utilizar matrices de diagramas de dispersión (Cleveland 1993). Las matrices resultantes son cuadradas y el elemento (i, j) de la matriz es un diagrama de dispersión de la dimensiones i y j . El diseño evita la pérdida de información pero en cambio los análisis complejos son engorrosos. Una deficiencia adicional es que la diagonal principal de la matriz es subutilizada. Algunos trabajos actuales están encaminados a aprovechar mejor esta región de la representación (Cui, MO Ward y Rundensteiner 2006).

Coordenadas paralelas

Las coordenadas paralelas son un método de visualización diseñado para crear una representación en 2D de datos multidimensionales sin pérdida de información. La técnica fue introducida por (Inselberg y Dimsdale 1990). En ella se visualiza una tupla de datos $\{x_1, x_2, \dots, x_n\}$ como una línea poligonal, conectando los puntos x_1, x_2, \dots, x_n en n ejes y paralelos (Figura 1.1).

Para volúmenes de datos suficientemente grandes la visualización de la técnica puede llegar a ser confusa. Una posible solución consiste en una extensión en 3D (Figura 1.1), donde el plano x/y representa la versión en 2D de las coordenadas paralelas, mientras la dimensión z representa la densidad de los eventos (Streit y col. 2006).

Gráfico de Andrews

En un Gráfico de Andrews (D. F. Andrews 1972) cada punto n -dimensional de datos $\mathbf{x} = \{x_1, x_2, \dots, x_n\}$ define una serie de Fourier finita:

$$f_{\mathbf{x}}(t) = \frac{x_1}{\sqrt{2}} + x_2 \sin(t) + x_3 \cos(t) + x_4 \sin(2t) + x_5 \cos(2t) + \dots$$

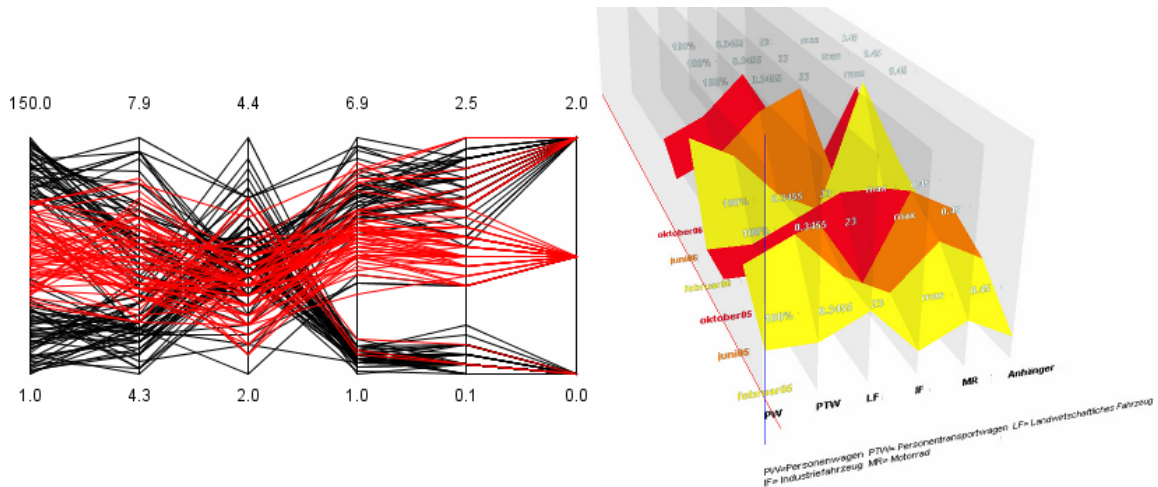


Figura 1.1: Coordenadas paralelas (izquierda) y coordenadas paralelas en 3D (derecha).

Esta función se grafica para el intervalo $-\pi < t < \pi$. De esta manera cada punto de datos puede verse como una línea entre $-\pi$ y π . Esta fórmula puede verse como una proyección del punto de datos en el vector:

$$\left(\frac{1}{\sqrt{2}}, \sin(t), \cos(t), \sin(2t), \cos(2t), \dots \right)$$

Si existiese determinada estructura en los datos, podría ser visible en las curvas de Andrews formadas con estos datos.

Table Lens

Table Lens es una técnica de visualización inspirada en las aplicaciones de hojas de cálculo, pero a diferencia de estas los datos se representan usando barras horizontales en lugar de valores numéricos (ver Figura 1.8a). Los datos se representan en una matriz donde los atributos se representan en columnas y cada instancia de los datos se reporta en una fila (Rao y Card 1994; Mazza 2009). Ha sido caracterizada

como una herramienta efectiva para comprender las características de datos numéricos y categóricos en conjuntos de datos multivariados (Pirolly y Rao 1996).

Vista circular

El proceso de visualización en una vista circular (Daniel A Keim, Schneidewind y Sips 2004) consiste en la división de un círculo en un número de segmentos dependiendo del número de variables del conjunto de datos a mostrar. Cada segmento a su vez es dividido en subsegmentos, a menudo indicando momentos de tiempo (Figura 1.2).

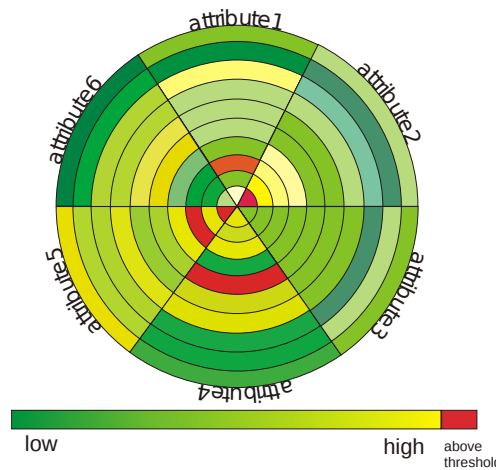


Figura 1.2: Vista circular de varios atributos a través del tiempo.

Coordenadas de estrella

Las coordenadas de estrella consisten en la disposición de los ejes coordenados de forma circular en un plano bidimensional, con ángulos iguales entre los ejes y origen en el centro de la circunferencia (Kandogan 2000). El sistema de coordenadas de estrella es un sistema coordenado curvilíneo, que puede ser convertido a coordenadas cartesianas mediante la definición de un punto bidimensional

$O_n(x, y) = (o_x, o_y)$ representando el origen y una secuencia de n vectores bidimensionales representando los ejes:

$$A_n = \langle \mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_i, \dots, \mathbf{a}_n \rangle$$

La conversión de un elemento D_j de un conjunto de datos D hacia un punto bidimensional en las coordenadas cartesianas está determinado por la suma de todos los vectores unitarios $\mathbf{u}_i = (u_{xi}, u_{yi})$ en cada coordenada multiplicado por el valor del elemento de datos de esa coordenada:

$$P_j(x, y) = \left(ox + \sum_{i=1}^n u_{xi} \times (d_{ji} - \min_i), oy + \sum_{i=1}^n u_{yi} \times (d_{ji} - \min_i) \right)$$

donde:

$$D_j = (d_{j0}, d_{j1}, \dots, d_{ji}, \dots, d_{jn}),$$

$$\left| \frac{\mathbf{a}_i}{\max_i - \min_i} \right|,$$

$$\min_i = \min \{d_{ji}, 0 \leq j < |D|\}, \max_i = \max \{d_{ji}, 0 \leq j < |D|\}.$$

En la Figura 1.3 se muestra un gráfico basado en coordenadas de estrella con la muestra de 9 variables. En este caso particular se ha definido un gradiente de color de acuerdo a los valores de una de las variables representadas.

1.2.2. Técnicas basadas en íconos

Las técnicas basadas en íconos visualizan datos multidimensionales mediante la asignación de cada objeto de datos sobre valores de los parámetros en pequeñas gráficas primitivas. Normalmente, los valores de los atributos están representados por la posición (x, y) del ícono así como la longitud, el ángulo o forma de algún componente cónico. Para lograr un buen resultado, los componentes dentro de un ícono deben ser distinguibles, los íconos separados deben ser claramente identifi-

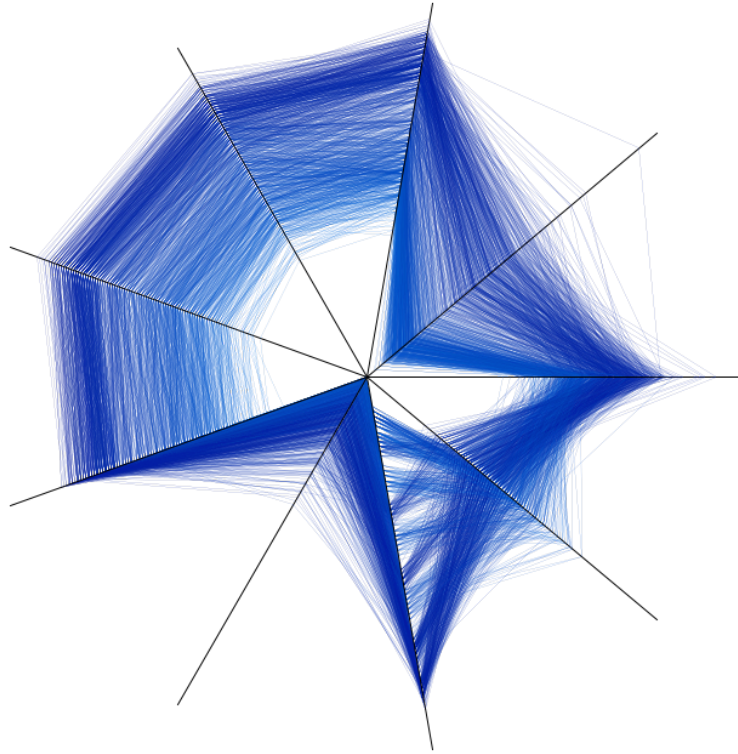


Figura 1.3: Gráfico en coordenadas de estrella

cables y los íconos deben ser percibidos como distintos si difieren en algunos de los componentes.

Las técnicas basadas en íconos tienen dos parámetros que la caracterizan. El primero es el tipo de figura que representará cada observación, o sea, la forma del ícono; el segundo parámetro es la forma en que se definirá la posición de cada ícono en la imagen (Theisel 2000; M. O. Ward 2002).

Estas técnicas no sufren de pérdida de información. Se logra evitar la pérdida de información al realizar una proyección de las dimensiones a los diferentes rasgos del ícono (Theisel 2000).

Las técnicas basadas en íconos son recomendadas cuando el número de dimensiones oscila entre diez y quince y el número de mediciones de las mismas es alto. Estas técnicas se pueden utilizar con una referencia espacial.

Ejemplos de técnicas basadas en íconos son: Caras de Chernoff, Íconos de flechas (cf. Daniel A. Keim 2000; Abello y Korn 2002), Íconos de estrellas (cf. M.O. Ward 1994), Figuras con palillos, Íconos de colores (cf. Levkowitz 1991; D.A. Keim y H. Kriegel 1994), Íconos de barras (cf. Hearst 1995), *Shape Coding* (cf. Beddow 1990), *Profile Glyphs* (cf. Chen, Härdle y A. Unwin 2008) y *Starfield* (cf. Eick 2000; DA Keim 2002a; M. O. Ward 2002; Xie y col. 2006).

Caras de Chernoff

Este tipo de diagrama surge como respuesta a la facilidad que tenemos las personas para reconocer y clasificar a otras personas por sus rostros, gracias a la capacidad de percepción que tenemos (Chernoff 1973). Esta capacidad de percepción es la que se desea explotar con los diagramas de Chernoff, los cuales usan representaciones de rostros de tipo trazos, los cuales a través del tamaño de los ojos, nariz, orejas y boca, agregándosele su forma o curvatura, incluso de la misma cabeza, permiten combinar los diferentes atributos de los datos multivariados en un único símbolo (Figura 1.4).

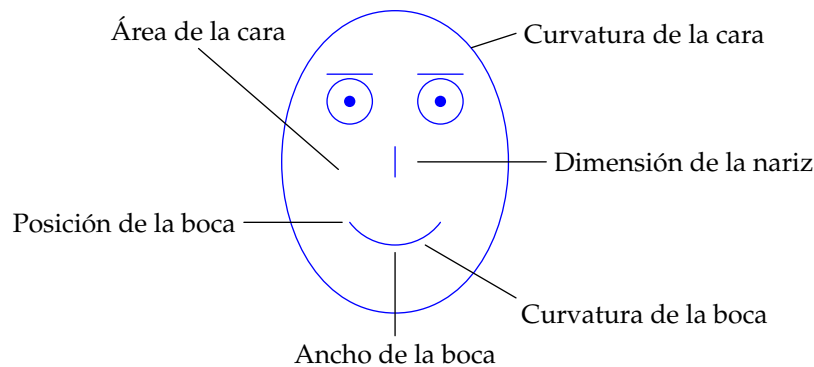


Figura 1.4: Algunos parámetros configurables en una cara de Chernoff

Figuras con palillos

La visualización de figura con palillos o *Stick Figure Icon* fue descrita por primera vez por Pickett y Grinstein (1988). Es rica en conceptos y aplicaciones prácticas (Wong y Bergeron 1994). En su forma más básica se representa cada dato multivariado del conjunto por un ícono gráfico, cuyas características visibles son controladas por los valores de cada una de las variables.

El ícono original es una figura formada por cinco palillos (cuatro miembros y un cuerpo) con ángulos controlables en las extremidades (Figura 1.5). De esta forma podrían representarse hasta 5 variables, no obstante existen variantes que permiten modificar además el tamaño, grosor o el color de los palillos.

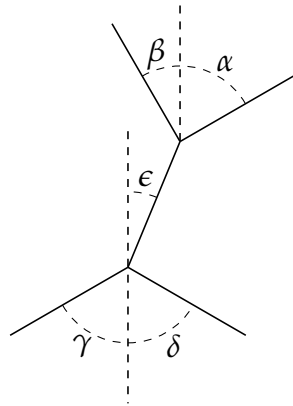


Figura 1.5: Figura con palillos

1.2.3. Técnicas orientadas a píxel

La visualización de un conjunto de datos de gran tamaño resulta un reto para técnicas geométricas y basadas en íconos. Al graficarlos suele surgir desorden en la imagen, que está originado por el tamaño de la figura que representa una observación simple. Partiendo de esta idea resulta lógico concluir que minimizando el espacio que ocupa un solo punto de dato en la imagen se mejoraría la percepción visual (Hansen y Johnson 2005; K. Andrews 2005).

Para lograr la maximización del número de elementos a representar, algunas técnicas utilizan los píxeles de la pantalla como unidades básicas de representación (Mazza 2009). El procedimiento consiste en relacionar cada valor de una dimensión a un color y agrupar los píxeles de cada dimensión en áreas adyacentes (DA Keim 2002a). De esta manera una computadora con una resolución de pantalla de 1024×800 píxeles podría potencialmente representar 786 432 elementos de un conjunto de datos univariado.

Este tipo de técnicas utiliza diferentes modos de posicionamiento de los píxeles para lograr diferentes objetivos. Colocar los píxeles en la forma apropiada ofrece la posibilidad de observar información sobre correlaciones, dependencias y regiones trascendentales. Dos de los modos de posicionamiento de los píxeles son los Patrones Recursivos y los Segmentos de Círculo. Otros ejemplos de técnicas orientadas a píxel son *Spiral Technique* (D.A. Keim y H. Kriegel 1994), y *Axes Technique* (D.A. Keim y H. Kriegel 1994).

Patrones recursivos

La técnica de «patrones recursivos» esta basada en un simple arreglo consecutivo de un lado hacia el otro. La idea general consisten en ubicar un número determinado de elementos de izquierda a derecha, luego una fila hacia abajo de derecha a izquierda, luego debajo de izquierda a derecha de manera sucesiva (ver Figura 1.6 (a)). El mismo arreglo básico se hace en todos los niveles de recursión con la única diferencia que los elementos básicos situados en el nivel i son los patrones resultantes de las ubicaciones del nivel $i - 1$. Si w_i es el número de elementos ubicados de izquierda a derecha en el nivel i y h_i es el número de filas en el nivel i entonces el patrón en el nivel i consiste de $w_i \times h_i$ nivel($i - 1$)-patrones, y el máximo número de píxeles que pueden ser representados en el nivel k está dado por $\prod_{i=1}^k w_i \times h_i$ (Daniel A. Keim, H.-P. Kriegel y Michael Ankerst 1995).

El algoritmo Draw permite realizar la visualización de «patrones recursivos». Inicialmente el algoritmo se llama con $\text{DRAW}(0, 0, \text{MAX-LEVEL})$ con el ancho y alto de

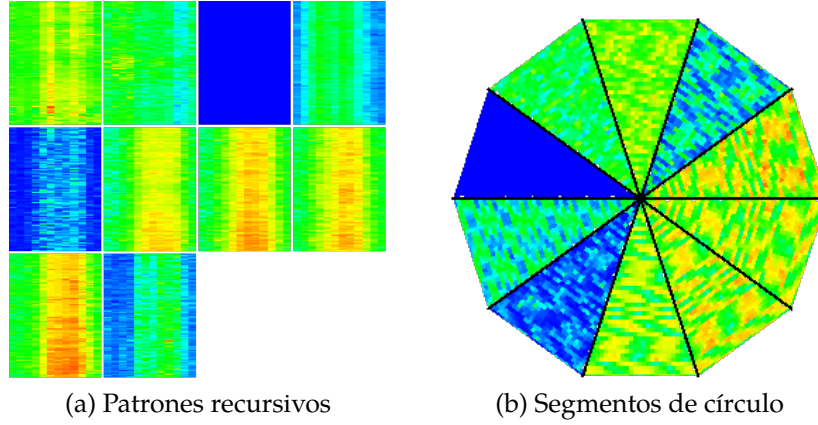


Figura 1.6: Técnicas de visualización orientadas a píxel mostrando un conjunto de datos de 10 dimensiones.

todos los niveles de recursividad almacenados en un arreglo previamente definido. La condición de parada es la llegada al nivel de recursión 0. Para los niveles $i (i \geq 1)$, el algoritmo dibuja w_i nivel($i - 1$)-patrones h_i veces alternadamente hacia la derecha y hacia la izquierda.

```

DRAW( $x, y, level$ )
1  if  $x = 0$ 
2    then SET-PIXEL( $x, y, color$ )
3  else for  $h \leftarrow 1$  to  $height[level]$ 
4    do if  $h \bmod 2 = 0$ 
5      then for  $w \leftarrow 1$  to  $width[level]$ 
6        do DRAW( $x, y, level - 1$ )
7         $x \leftarrow x + \prod_{i=1}^{level-1} w_i$ 
8      else for  $w \leftarrow 1$  to  $width[level]$ 
9        do  $x \leftarrow x - \prod_{i=1}^{level-1} w_i$ 
10       DRAW( $x, y, level - 1$ )
11   $y \leftarrow \prod_{i=1}^{level-1} h_i$ 

```

Segmentos de círculo

Como su nombre lo indica la idea fundamental de esta técnica es mostrar las dimensiones de los datos como segmentos de un círculo. Si el conjunto de datos consiste en n variables, el círculo es consecuentemente particionado en n segmentos. Los elementos de los datos dentro de cada segmento son organizados de un lado hacia el otro a través de la llamada «línea de dibujo» o *draw-line* ortogonal a la línea que divide los dos segmentos del borde (ver Figura 1.6 (b)). Cada vez que la «línea de dibujo» toca uno de las líneas del borde, la primera se mueve en paralelo junto a la línea divisoria del segmento hacia el exterior del círculo y la dirección de la «línea de dibujo» cambia. Este proceso se repite luego para cada una de las variables restantes (M. Ankerst y col. 1996; Daniel A. Keim 2000; Hansen y Johnson 2005).

El algoritmo `FILL-SEGMENT` se llama con las dos líneas del borde del segmento y se introduce en la subrutina `INITIAL-PIXELS`. La función `INITIAL-PIXELS` dibuja los primeros píxeles de un segmento hasta que las siguientes «líneas de dibujo» tengan al menos un píxel entre las dos líneas del borde. El valor de retorno de `INITIAL-PIXELS` es el número de píxeles dibujados hasta ahora. La función `COMPUTE-NEXT-POINT` se mueve hacia delante por la «línea de dibujo». La función `POINT-BETW-LINES` comprueba si un punto está aún en el segmento. Si el punto no está en el segmento, la «línea de dibujo» se mueve un píxel hacia el exterior en paralelo con la línea divisoria del segmento. La nueva «línea de dibujo» dibuja el píxel en la dirección opuesta a la anterior.

FILL-SEGMENT($l-1, l-2$)

```
1   $x, y, direction \leftarrow 1$ 
2   $record-count \leftarrow \text{INITIAL-PIXELS}(l-1, l-2, x, y)$ 
3  while  $record-count < \text{RECORD-ALL}$ 
4      do while  $\text{POINT-BETW-LINES}(l-1, l-2, x, y) = \text{TRUE} \wedge record-count < \text{RECORD-ALL}$ 
5          do  $record-count \leftarrow record-count + 1$ 
6               $\text{SET-PIXEL}(x, y, color)$ 
7               $\text{COMPUTE-NEXT-POINT}(draw-line, x, y, direction)$ 
8           $\text{MOVE}(draw-line)$ 
9           $\text{COMPUTE-NEXT-POINT}(draw-line, x, y, direction)$ 
10          $direction \leftarrow -direction$ 
11         while  $\text{POINT-BETW-LINES}(l-1, l-2, x, y) = \text{FALSE}$ 
12             do  $\text{COMPUTE-NEXT-POINT}(draw-line, x, y, direction)$ 
```

1.3. Representación del tiempo

Los datos que implican cambios a través del tiempo constituyen un reto para la visualización y el análisis de datos. Existen varias técnicas para la visualización de conjuntos de datos con componentes temporales. Sin embargo, dado que es difícil considerar todos los aspectos concernientes a la dimensión tiempo en una sola visualización, la mayoría de los métodos disponibles tratan únicamente casos específicos — mayormente la visualización de datos con un eje de tiempo lineal (Daniel A Keim, Kohlhammer y col. 2010).

No obstante las series temporales a menudo muestran estructuras periódicas que no pueden ser apreciadas a plenitud con este tipo de visualizaciones. La Figura 1.7 provee un ejemplo en que este tipo de patrón solo se hace evidente utilizando una visualización basada en una espiral de Arquímedes (Weber, Alexa y Müller 2001).

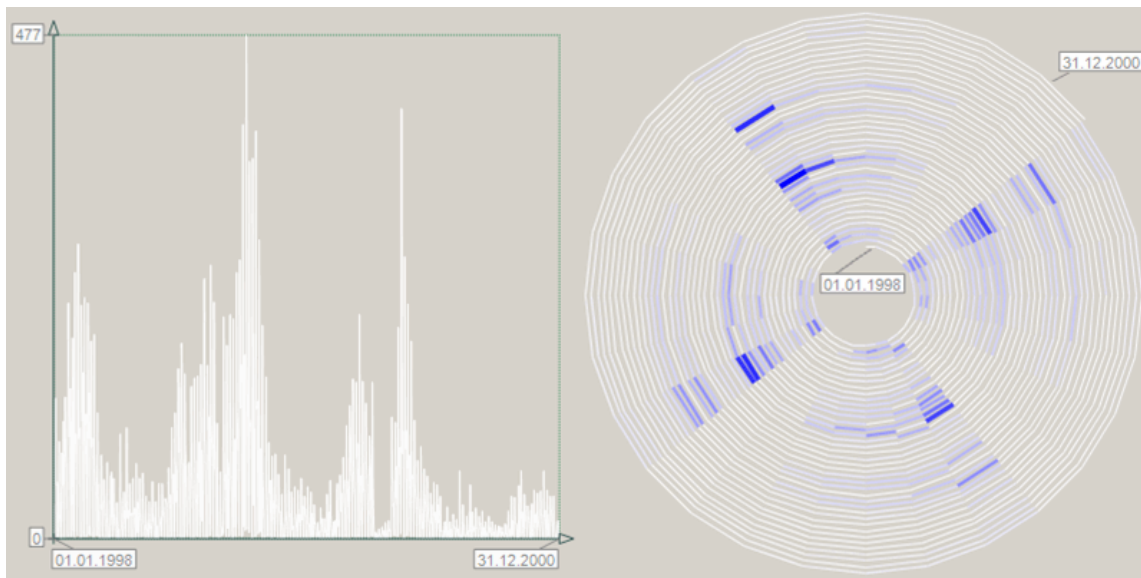


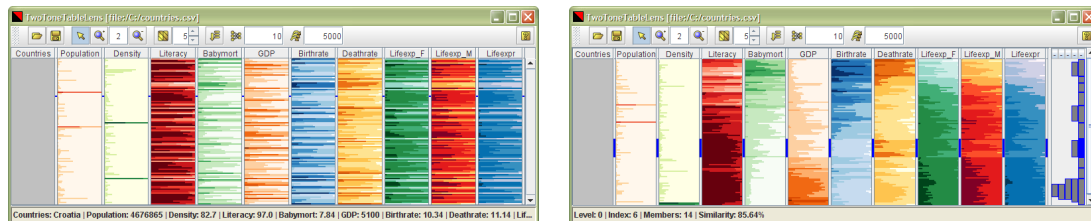
Figura 1.7: Dos representaciones de una misma serie de tiempo. El gráfico en espiral de la derecha permite detectar un patrón cíclico aparente.

1.4. Visualización de grandes volúmenes de datos

La forma tradicional de visualización está centrada en mostrar una representación directa de todo el conjunto de datos. Aunque este enfoque es válido y útil en varios casos, la complejidad y tamaño de muchos de los conjuntos de datos con los que se trata actualmente suelen disminuir la efectividad de tales visualizaciones. Una de las soluciones propuestas a esta problemática consiste en agregar una fase previa a la visualización en la que se utilicen métodos de agrupamiento de datos. Una segunda propuesta incluye la utilización de técnicas computacionales más avanzadas de minería de datos para extraer patrones o características *interesantes* de manera automática o semiautomática (Daniel A Keim, Kohlhammer y col. 2010).

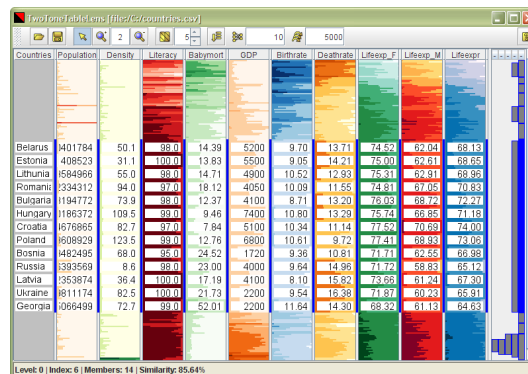
En el caso de *Table Lens* (cf. Sección 1.2.1) una variante es la aplicación de pseudocoloración en dos tonos (*Two-Tone Pseudo Coloring*, TTPC) con el objetivo de obtener representaciones visuales que puedan comunicar mayores volúmenes de datos sin

costo en precisión. Adicionalmente se integra un paso de análisis de datos que implementa un agrupamiento híbrido basado en SOM (cf. Sección 1.4) y agrupamiento jerárquico (John, Tominski y Schumann 2008). En la Figura 1.8 se ejemplifica este concepto sobre un conjunto de datos multivariados.



(a) Vista no ordenada.

(b) Posterior a la aplicación del agrupamiento, se puede seleccionar un clúster de interés (denotado por una barra azul).



(c) La expansión del clúster revela países del antiguo campo socialista.

Figura 1.8: Visualización de un conjunto de datos multivariados sobre países mediante *Table Lens* con TTPC y agrupamiento híbrido. Fuente: (John, Tominski y Schumann 2008).

Mapas autoorganizados

Los mapas autoorganizados (*Self-Organizing Map*, SOMs), también conocidos como redes de Kohonen fueron introducidos por el profesor finlandés Teuvo Kohonen

en los artículos (Kohonen 1982; Kohonen 1990; Kaski, Nikkilä y Kohonen 1998). Un mapa autoorganizado es una herramienta que analiza datos en muchas dimensiones con relaciones complejas entre ellos y los presenta en una visualización sencilla en sólo dos dimensiones.

La propiedad más importante de SOM es que preserva las propiedades topológicas de los datos, es decir, que datos próximos aparecen próximos en la visualización. Este tipo de red posee un aprendizaje no supervisado competitivo. La red auto-organizada debe descubrir rasgos comunes, regularidades, correlaciones o categorías en los datos de entrada, e incorporarlos a su estructura interna de conexiones. Se dice, por tanto, que las neuronas deben auto-organizarse en función de los estímulos (datos) procedentes del exterior. En el aprendizaje competitivo las neuronas compiten unas con otras con el fin de llevar a cabo una tarea dada. Se pretende que cuando se presente a la red un patrón de entrada, sólo una de las neuronas de salida (o un grupo de vecinas) se active. Por tanto, las neuronas compiten por activarse, quedando finalmente una como neurona vencedora y anuladas el resto, que son forzadas a sus valores de respuesta mínimos.

Un modelo SOM está compuesto por dos capas de neuronas. La capa de entrada (formada por n neuronas, una por cada variable de entrada) se encarga de recibir y transmitir a la capa de salida la información procedente del exterior. La capa de salida (formada por m neuronas) es la encargada de procesar la información y formar el mapa de rasgos. Normalmente, las neuronas de la capa de salida se organizan en forma de mapa bidimensional. En la Figura 1.9 se presentan diferentes representaciones visuales de los resultados de un agrupamiento por SOM.

1.5. Sistemas de Información Geográfica

Un Sistema de Información Geográfica (SIG) es una integración organizada de hardware, software y datos geográficos diseñada para capturar, almacenar, manipular, analizar y desplegar en todas sus formas la información geográficamente

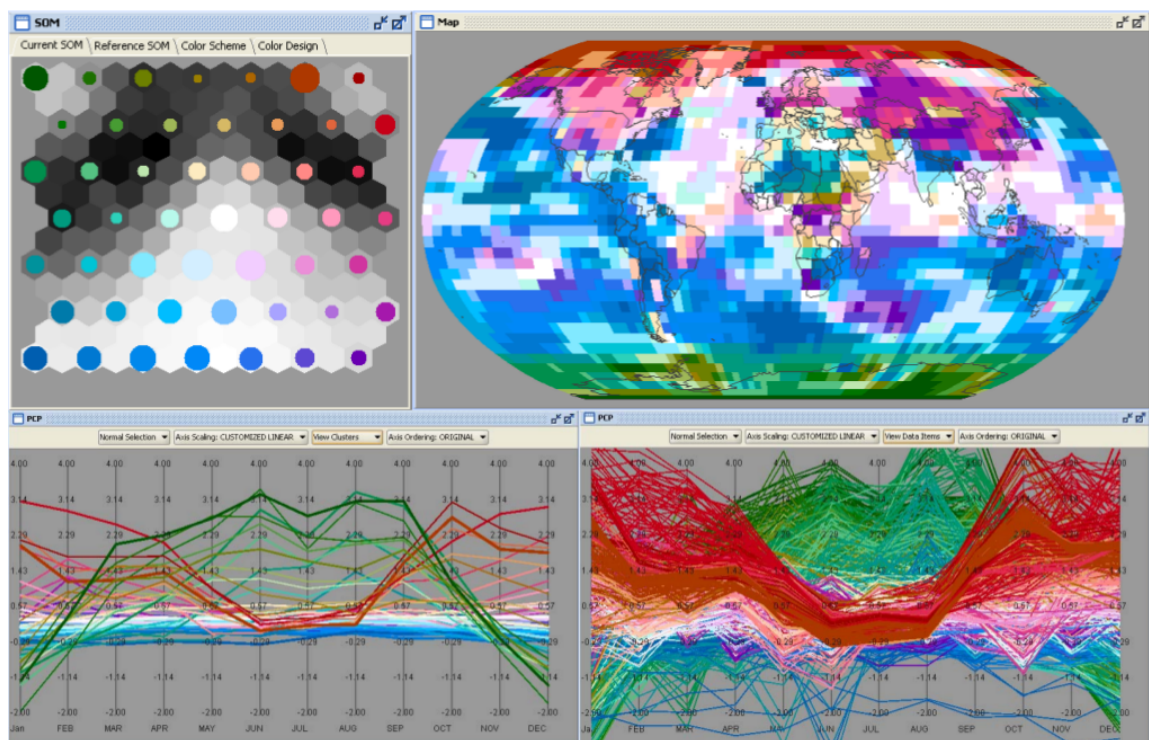


Figura 1.9: Diferentes representaciones de los resultados obtenidos mediante el agrupamiento con SOM.

referenciada con el fin de resolver problemas complejos de planificación y gestión geográfica (Chrisman y col. 1989). También puede definirse como un modelo de una parte de la realidad referido a un sistema de coordenadas terrestre y construido para satisfacer unas necesidades concretas de información.

En el sentido más estricto, es cualquier sistema de información capaz de integrar, almacenar, editar, analizar, compartir y mostrar la información geográficamente referenciada. En un sentido más genérico, los SIG son herramientas que permiten a los usuarios crear consultas interactivas, analizar la información espacial, editar datos, mapas y presentar los resultados de todas estas operaciones. Los SIG son herramientas cartográficas que permiten la creación de mapas en tiempo real y el análisis de los impactos de los cambios en el mapa de forma interactiva (TM Rhyne 1997).

Las principales cuestiones que puede resolver un SIG, ordenadas de manera descendente por su complejidad, son:

Localización preguntar por las características de un lugar concreto.

Condición el cumplimiento o no de las condiciones impuestas al sistema.

Tendencia comparación entre situaciones temporales o espaciales distintas de alguna característica.

Rutas cálculo de rutas óptimas entre dos o más puntos.

Pautas detección de pautas espaciales.

Modelos generación de modelos a partir de fenómenos o actuaciones simuladas.

La razón fundamental para utilizar un SIG es la gestión de información espacial. El sistema permite separar la información en diferentes capas temáticas y las almacena independientemente, permitiendo trabajar con ellas de manera rápida y sencilla, facilitando al profesional la posibilidad de relacionar la información existente a través de la topología de los objetos, con el fin de generar otra nueva que no podríamos obtener de otra forma.

1.5.1. GRASS GIS

GRASS es uno de los componentes principales del conjunto de herramientas provista por la *Open Source Geospatial Foundation* (OSGeo). Es un SIG multipropósito con amplio uso en la producción, análisis y representación de datos geospaciales (Neteler y col. 2012). Es software libre bajo la Licencia Pública General de GNU (*GNU General Public License*, GPL). Está soportado tanto en plataformas tipo UNIX® (GNU/Linux, MacOS X, BSD, ...) como en MS Windows, gracias a su código altamente portable en ANSI C con algunas funciones en C++ y Python. GRASS está escrito de una forma completamente modular, de manera tal que reduce el gasto en recursos. Ello permite correr el sistema, o partes de este aún en pequeños dispositivos móviles con limitada RAM (Stankovic, Neteler y Flor 2002).

GRASS es uno de las pocos SIG con una completa solución para la visualización en 3D, con perspectiva 2D simple y multimapa. Puede ser enlazado directamente a varias aplicaciones, incluyendo Quantum GIS, Sextante, herramientas estadísticas y geoestadísticas (R, gstat, MATLAB) y aplicaciones de visualización multidimensional (Neteler y col. 2012; Furlanello y col. 2003).

1.5.2. QGIS

QGIS es un Sistema de Información Geográfica libre distribuido bajo GPL. Se trata de una aplicación multiplataforma disponible para GNU/Linux, Mac OS X y MS Windows. Permite manejar formatos ráster y vectoriales a través de la Biblioteca de Abstracción de Datos Geoespaciales (*Geospatial Data Abstraction Library*, GDAL/OGR), así como bases de datos. QGIS está desarrollado en C++, usando la biblioteca Qt para su interfaz gráfica de usuario. Una de sus mayores ventajas es la posibilidad de usarlo como interfaz gráfica de usuario de GRASS, utilizando toda la potencia de análisis de este último en un entorno de trabajo más amigable.

1.5.3. gvSIG

gvSIG surgió como un proyecto amparado por la *Generalitat* de la Comunidad Valenciana (España) a finales de 2003. Es un SIG multiplataforma desarrollado en el lenguaje de programación Java™. Es software libre, distribuido bajo la versión 2 de la GPL.

Cuenta con una amplia gama de herramientas para trabajar con información geográfica entre las que se incluyen herramientas de consulta, la creación de diseño, geoprocetamiento, redes. Su diseño favorece la extensibilidad, lo que permite la mejora continua de las aplicaciones, así como permitir el desarrollo de soluciones a medida.

Es un programa informático para el manejo de información geográfica con precisión cartográfica. Permite acceder a información vectorial y rasterizada así como a servidores de mapas que cumplan las especificaciones del *Open Geospatial Consortium* (OGC). Esta es una de las principales ventajas de gvSIG con respecto a otros sistemas, la implementación de servicios OGC: *Web Map Service* (WMS), *Web Feature Service* (WFS), *Web Coverage Service* (WCS), Servicio de Catálogo y Servicio de Nomenclátor.

1.6. Geovisualización

Por más de dos décadas la visualización científica y los SIG se desarrollaron en paralelo y de forma independiente (T.M. Rhyne 1997). Los esfuerzos para desarrollar estándares de datos espaciales rara vez consideraron la forma en que estos se visualizaban. Las bibliotecas gráficas y los estándares evolucionaron independientemente de los modelos de datos. Como resultado de esto se evidenciaron muchas ineficiencias asociadas con la visualización de datos geográficos. Entre ellas se incluyen dificultades con el registro de los datos espaciales dentro de sistemas de visualización científica, engorrosas producciones de secuencias de animaciones

en SIG y quizás la más importante, la falta de conexión entre bases de datos y los ambientes de visualización que soportaban la visualización de datos espaciales (Hearnshaw y D. J. Unwin 1994).

La visualización geográfica o geovisualización es un campo emergente que se destaca por la incorporación de técnicas y herramientas para el análisis visual interactivo de datos espaciales y espaciotemporales. Una característica distintiva es la integración de los enfoques de múltiples disciplinas como la geografía, la ciencia de información geográfica, cartografía, visualización de información, minería de datos y otras disciplinas afines (Daniel A Keim, Kohlhammer y col. 2010).

1.6.1. Snap-Together Visualization

Snap-Together Visualization es una herramienta Web que permite que los datos de los usuarios sean mezclados y correlacionados de forma dinámica en visualizaciones coordinadas para la construcción personalizada de interfaces de exploración sin necesidad de programación. El modelo conceptual de Snap-Together se basa en un modelo de base de datos relacional. Este modelo permite que las relaciones sean cargadas durante la visualización y se coordinen basándose en las características que las unen. Los usuarios pueden crear diferentes tipos de coordinaciones tales como: barridos, vistas de detalles, vistas globales y desplazamientos sincronizados.

Los desarrolladores de visualización pueden integrar al sistema sus visualizaciones independientes con una Interfaz de Programación de Aplicaciones (*Application Programming Interface*, API) simple. La evaluación de esta herramienta reveló beneficios en cuanto a aspectos cognitivos y su usabilidad, mejorando el rendimiento de los usuarios entre un 30 % y un 80 %, dependiendo de la tarea realizada (North y Shneiderman 2000).

Algunas de las técnicas de visualización que están incluidas en esa herramienta son Diagramas de Dispersión y *TreeMap* (North y Shneiderman 2000).

1.6.2. GeoVISTA *Studio*

GeoVISTA *Studio* (Gahegan y col. 2002; Takatsuka y Gahegan 2002; MacEachren y col. 2003) es una herramienta de código abierto que implementa un ambiente de desarrollo basado en componentes. Suministra una interfaz de programación visual como muchos sistemas de visualización científica, a través de la cual los usuarios pueden construir aplicaciones de forma rápida utilizando JavaBeans™. El ambiente de programación visual permite a los analistas empaquetar funcionalidades dentro de un programa de trabajo. *Studio* soporta el desarrollo de aplicaciones geográficas y no geográficas.

Para soportar la interoperabilidad de datos OpenGIS, sus desarrolladores han comenzado a adaptar y extender la biblioteca GeoTools en cuanto al acceso a datos y los métodos de visualización. GeoTools es una biblioteca complementaria de software libre desarrollada en Java™ para el desarrollo de soluciones OpenGIS que permite el acceso a datos geoespaciales, el análisis y la representación de tareas.

El principal objetivo de GeoVISTA *Studio* es soportar la fusión de diversas capacidades visuales y analíticas en una herramienta de análisis que posibilite la multiperspectiva. Incluye además un conjunto de técnicas de visualización clásicas como diagramas de dispersión, coordenadas paralelas y mapas autoorganizados (Kohonen 1990; Gahegan y col. 2002; Takatsuka y Gahegan 2002). Varias herramientas han sido desarrolladas tomando como base el *Studio*, entre ellas se encuentran el Conjunto de Herramientas para el Análisis Espacio-Temporal Exploratorio (*Exploratory Spatio-Temporal Analysis Toolkit*, ESTAT) y el Sistema de Visualización para Patrones Espacio-Temporales y Multiparamétricos (*Visualization System for Space-Time and Multivariate Patterns*, VIS-STAMP).

1.6.3. VIS-STAMP

VIS-STAMP es un paquete de software que integra métodos computacionales, visuales y cartográficos para la exploración y visualización de datos espacio-temporales

multivariados. Uno de sus principales objetivos es proveer los medios para el descubrimiento de patrones desconocidos (Guo y col. 2006). En particular ha sido usado exitosamente en el estudio de patrones relacionados con el cambio climático (Jin y Guo 2009). La base del sistema es un mapa autoorganizado usado para el agrupamiento, ordenamiento y coloración (Aigner 2011). La presentación visual de los datos está compuesta por varias vistas integradas (interrelacionadas entre sí). En particular incluye mapas geospaciales, mapas SOM, gráficos de malla y coordenadas paralelas (Andrienko y col. 2010).

1.6.4. ArcView - xGobi

En la geovisualización, un buen ejemplo de integración es el enlace bidireccional entre ArcView y XGobi (Symanzik y col. 2000), trabajo donde se integran los gráficos interactivos de XGobi para manipular datos con muchas dimensiones con las herramientas de manipulación de datos espaciales de ArcView. El ArcView - XGobi - Xplore permite que los datos recogidos en lugares espaciales que se almacenan en ArcView pasen dinámicamente a XGobi y Xplore y ser explorados y analizados. El vínculo entre los datos de XGobi y Xplore y los lugares de los que fueron recogidos se mantienen a través de marcado y enlace (*linking and brushing*). El marcado y enlazado, tal como se utiliza en este contexto, es la capacidad de cambiar el tamaño / color de los puntos, ya sea en ArcView, XGobi o Xplore y para ver que los puntos correspondientes de las otras aplicaciones cambian simultáneamente.

1.7. Formatos de datos científicos utilizados en la Visualización Científica

Otra forma de integración de visualización científica-SIG es mediante la incorporación a los SIG de formatos de datos científicos. Se han desarrollado intentos de integrar algunos de los formatos de datos científicos más comúnmente utilizados en

la visualización científica, —como *Common Data Format* (CDF) (Atkinson y Hankin 1995), *Network Common Data Form* (netCDF) (Rew y Davis 1990), *Hierarchical Data Format* (HDF) (Zhao y col. 2010), y *Flexible Image Transport System* (FITS) (Hanisch y col. 2001)—, con sistemas de información geográfica, por ejemplo actualmente el SIG propietario ArcGIS permite la manipulación de algunos de estos formatos, como netCDF y HDF.

1.7.1. *Hierarchical Data Format* (HDF)

Hierarchical Data Format (HDF) es desarrollado por el Centro Nacional de Aplicaciones de Supercómputo (*National Center for Supercomputing Applications*, NCSA) de la Universidad de Illinois, su versión actual es HDF5. HDF está compuesto por un Modelo Abstracto de Datos, su representación estándar, el Modelo de Almacenamiento (definido por el formato de archivo HDF5) y el Modelo de Programación, cuya implementación concreta es la biblioteca HDF5 (Group y col. 2011). La biblioteca se encuentra disponible para su utilización en aplicaciones escritas en C, C++, Fortran o Java™. Entre las principales características de HDF se pueden destacar:

Versatilidad Posee soporte para diferentes modelos de datos. Cada modelo de dato define un conjunto específico de tipos de datos y provee una API para la lectura, escritura y organización de datos y metadatos del tipo correspondiente. Los modelos de datos soportados incluyen arreglos multidimensionales, imágenes de puntos y tablas.

Capacidad de autodescripción Permite interpretar la estructura y contenidos de un fichero sin ninguna información proveniente del exterior.

Flexibilidad Se pueden mezclar y asociar objetos relacionados agrupados en un fichero y acceder a ellos como un grupo o como objetos individuales. Los usuarios pueden además crear sus propias estructuras.

Extensibilidad Puede acomodar fácilmente nuevos modelos de datos, sin tener en cuenta si fueron adicionados por el equipo de desarrolladores de HDF o por los usuarios.

Portabilidad Los archivos HDF pueden ser compartidos a través de la mayoría de las plataformas comunes.

Compactabilidad Los datos pueden estar compactados permitiendo un ahorro en la capacidad de almacenamiento, sobre todo cuando no se necesita tener cargada toda la información en memoria.

1.7.2. *Common Data Format (CDF)*

Common Data Format (CDF) es una abstracción conceptual de datos para el almacenamiento, manipulación y acceso a conjuntos de datos multidimensionales (Space Physics Data Facility 2013). Es desarrollado por el Centro de Vuelos Espaciales Goddard de la Administración Nacional de la Aeronáutica y del Espacio (*National Aeronautics and Space Administration, NASA*) de los Estados Unidos. En CDF se describe la forma de los conjuntos de datos y los medios en que estos pueden ser manipulados. La utilización de este tipo de especificación en lugar de describir el formato interno (físico) a utilizar garantiza una mayor portabilidad. Sus creadores distribuyen un conjunto de herramientas para realizar operaciones usuales sobre los archivos CDF. Incluye además una biblioteca para la manipulación de los archivos CDF con APIs para los lenguajes de programación C, Java™, Fortran, C# y Perl.

1.7.3. *Network Common Data Form (netCDF)*

Network Common Data Form (netCDF), es una interfaz a una biblioteca de funciones de acceso a datos diseñada para el almacenamiento y recuperación de datos en forma de arreglos (Rew, Davis y col. 2011). Cada arreglo es una estructura rec-

tangular n -dimensional en la que todos sus elementos son del mismo tipo de dato (verbigracia carácter de 8-bits, entero de 32-bits). Dentro de este modelo un escalar equivale a un arreglo 0-dimensional. En la versión 4.0 se introdujo el formato netCDF4/HDF5 que constituye un subconjunto del formato HDF5.

Todos los formatos netCDF son autodescriptibles y son accesibles a través de una biblioteca escrita en C que provee APIs para C, C++ y Fortran. Adicionalmente existe una implementación escrita completamente en Java™. Ambas implementaciones son distribuidas como software libre bajo una licencia estilo BSD compatible con la GPL.

1.7.4. *Flexible Image Transport System (FITS)*

Flexible Image Transport System (FITS) es el formato de archivo más utilizado comúnmente en el mundo de la astronomía. Es a menudo utilizado para almacenar también datos que no son imágenes, como espectros electromagnéticos, listas de fotones, cubos de datos y muchos más. Un fichero FITS podría contener varias extensiones, y cada una de ellas podría contener datos de un objeto. Por ejemplo, es posible almacenar imágenes de rayos X y también imágenes pertenecientes al infrarrojo en el mismo archivo FITS.

La mayor ventaja de FITS para datos científicos es que la información de las cabeceras es legible en ASCII, de modo que un usuario puede examinar las cabeceras para investigar un archivo de procedencia desconocida. Cada archivo FITS consiste en una o más cabeceras que contienen secuencias de 80 cadenas de caracteres fijos que llevan pares de valores, interpolados entre los bloques de datos. Los pares de valores proveen información (metadatos) como son el tamaño, origen, formato binario de los datos, comentarios, historia de los datos y cualquier otra información que el creador desee: mientras varias palabras están restringidas para FITS, el estándar permite el uso arbitrario de todas las palabras. Está soportado mediante bibliotecas disponibles en los lenguajes más utilizados en el ámbito científico, incluyendo C, C++, C#, Fortran, Perl, Java™, PDL, Python e IDL.

1.8. Conclusiones

En este capítulo se presentaron los principales elementos teóricos sobre la visualización científica, profundizando en las técnicas de visualización de datos multiparamétricos. Además se analizaron características generales de los SIG más utilizados. Se analizaron los esfuerzos de integración entre estas dos disciplinas y se mostraron ejemplos de integración mediante el enfoque mezclado.

Las técnicas de visualización de datos multiparamétricos constituyen valiosas herramientas para el análisis visual de datos. Por ello se precisa el enriquecimiento del módulo de visualización científica del gvSIG con la adición de nuevas técnicas. La integración de este tipo de técnicas en gvSIG constituye un esfuerzo más en el acercamiento entre estas dos disciplinas.

2 DISEÑO E IMPLEMENTACIÓN DEL MÓDULO DE GEOVISUALIZACIÓN

En el presente capítulo se definen los principales aspectos de la extensión desarrollada para gvSIG y las técnicas que serán implementadas en el mismo. Además se exponen aspectos fundamentales en cuanto al diseño e implementación del módulo.

2.1. Diseño de la extensión

El proceso de desarrollo de técnicas de visualización para algún sistema tiene como punto de partida natural la estructura y tipos de datos que serán utilizados. Con el objetivo de aumentar las posibilidades de análisis del módulo de geovisualización de gvSIG se seleccionó un conjunto de técnicas para ser adicionadas. En la Tabla 2.1 se muestran las técnicas escogidas y la forma en que fueron implementadas —sobre el mapa o en paneles independientes.

El diseño de la extensión fue realizado utilizando el lenguaje Lenguaje Unificado de Modelado (*Unified Modeling Language*, UML). Este lenguaje tiene como objetivos principales la especificación, visualización, construcción y documentación de los productos de un sistema de software. Es utilizado por el Proceso Unificado de Rational (*Rational Unified Process*, RUP) como lenguaje de modelado, para lo cual se basa en todos sus tipos de diagramas, que constituyen diferentes vistas del modelo

Técnica	Sobre el mapa	Vista independiente
Figuras con palillos		★
Espiral de tiempo		★
Vista circular	★	★
Coordenadas de estrella	★	★

Tabla 2.1: Nuevas técnicas implementadas para el módulo de visualización científica

del producto. A lo largo de este capítulo se insertarán diagramas UML que faciliten la comprensión del diseño de la extensión realizada.

2.1.1. Análisis de actores y casos de uso

Los modelos de casos de uso proporcionan un medio sistemático e intuitivo de capturar requisitos funcionales del sistema basándose en los requerimientos de los usuarios. Este modelo se realiza identificando cada actor del sistema como los posibles usuarios para los cuales está realizado el mismo.

El diagrama de casos de uso de la extensión implementada está destinada a un solo tipo de actor (Figura 2.1). En este caso el actor es el especialista o investigador que busca explorar sus datos de manera visual. El especialista debe contar con los conocimientos suficientes para interpretar de forma correcta —a partir de la descripción de una visualización— los resultados que brinda la herramienta.

El caso de uso principal lo constituye la visualización coordinada de datos multiparamétricos. A partir de este caso de uso se derivan los más especializados «Visualización sobre el mapa» y «Visualización en vistas independientes». La visualización coordinada significa visualizar, mediante técnicas de visualización científica, múltiples conjuntos de datos correspondientes a varios puntos en el espacio. Es coordinada porque todas las visualizaciones se realizan normalizando los valores de las variables teniendo en cuenta los mínimos y máximos en todos los conjuntos de datos para cada variable. La visualización coordinada permite obtener informa-

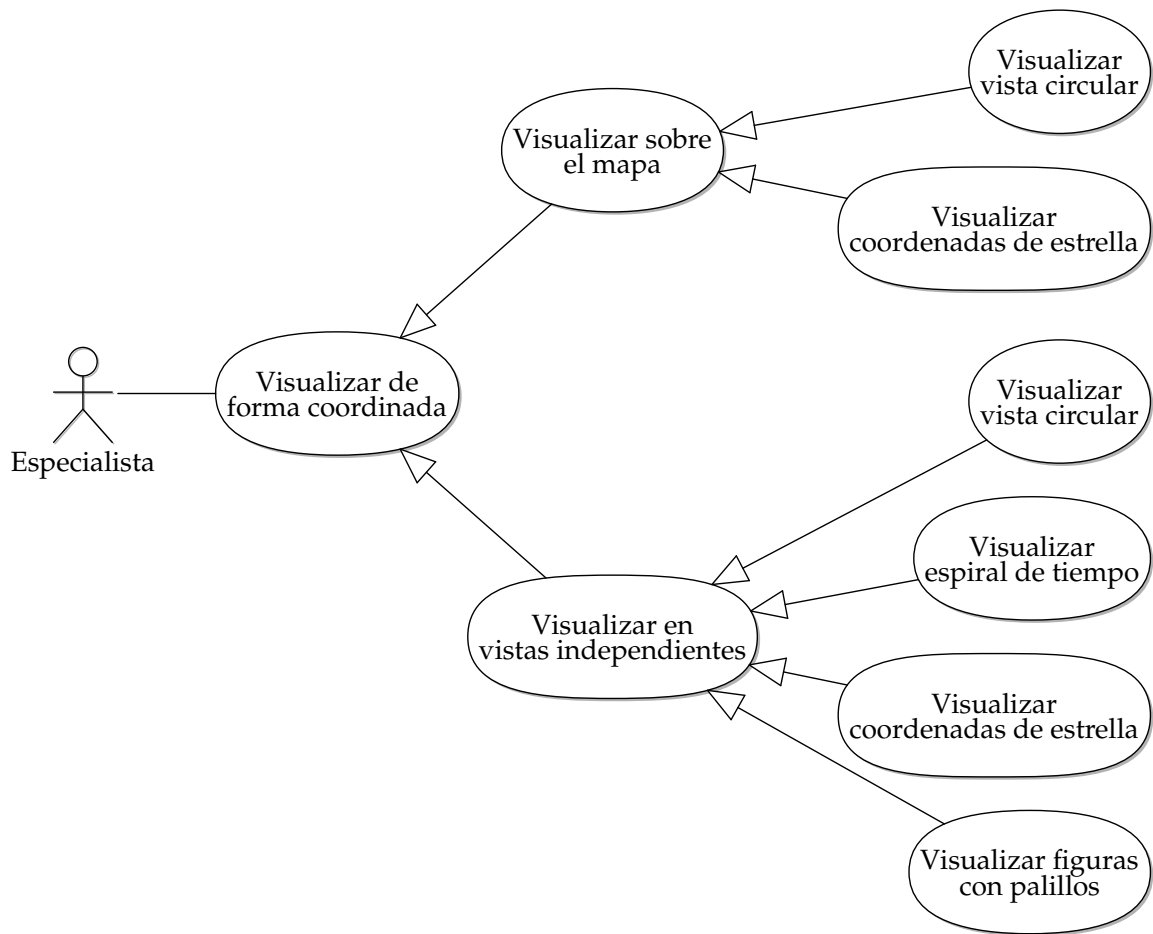


Figura 2.1: Diagrama de casos de uso del módulo.

ción sobre el origen de los datos visualizados, referenciando los gráficos generados por las técnicas a la localización de los datos en un mapa; esta referencia puede realizarse emplazando los gráficos sobre el punto de origen de los datos o visualizando los gráficos en ventanas independientes y relacionarlas con sus respectivas localizaciones en el mapa. Como parte final de la jerarquía se tienen los casos de uso de visualización de vista circular, coordenadas de estrella, figuras con palillos y espiral de tiempo —correspondientes a las nuevas técnicas implementadas para el módulo.

2.1.2. Diagramas de transición de estados

En su concepción el sistema pasa por una serie de estados que permiten completar el proceso de visualización coordinada. Este proceso se muestra en la Figura 2.2 como un diagrama de transición de estados.

Inicialmente el sistema puede pasar por dos estados: *Crear vista*, estado en que el usuario crea y configura una vista en gvSIG, y *Abrir vista*, estado en que el usuario abre una vista existente. De este último estado el sistema pasa a *Cargar capa de visualización* en el cual se carga el mapa de fondo como una capa de gvSIG que contiene además la información necesaria para acceder a la fuente de datos. De este estado pasa a *Seleccionar técnicas de visualización*, momento en el cual el usuario es capaz de seleccionar las técnicas que desea visualizar, especificando cuál de ellas se mostrará sobre el mapa y cuales en vistas independientes.

Luego se pasa al estado *Seleccionar localizaciones y mostrar representación visual de datos*. Durante este estado el usuario puede seleccionar los lugares del mapa de fondo de donde desea cargar los datos. Cuando el usuario presiona clic sobre un lugar del mapa se pasa al estado *Cargar Datos*. En el caso de la lectura del archivo HDF el proceso es el siguiente: Abrir el conjunto de datos que almacena los índices (Matriz de M por N) y cargarlo en memoria. Al acceder a esta matriz en (fila, columna), se devuelve el número de la celda si contiene información, si no contiene información el valor en el escaque dado es 0. Luego se abre cada uno de los conjuntos de datos que almacenan las variables climáticas y se lee solo los Q valores de cada conjunto de datos, accediendo por las filas al número de celda devuelto y leyendo en cada conjunto de datos los Q valores de la fila correspondiente, de esta forma se seleccionan $V \times Q$ valores. Además se lee los mínimos y máximos globales de cada variable.

Luego se pasa nuevamente al estado *Seleccionar localizaciones y mostrar representación visual de datos* y la información obtenida en el estado anterior se le pasa a cada técnica para que se visualice sobre la celda correspondiente en el mapa de fondo o en paneles independientes dependiendo de la configuración seleccionada.

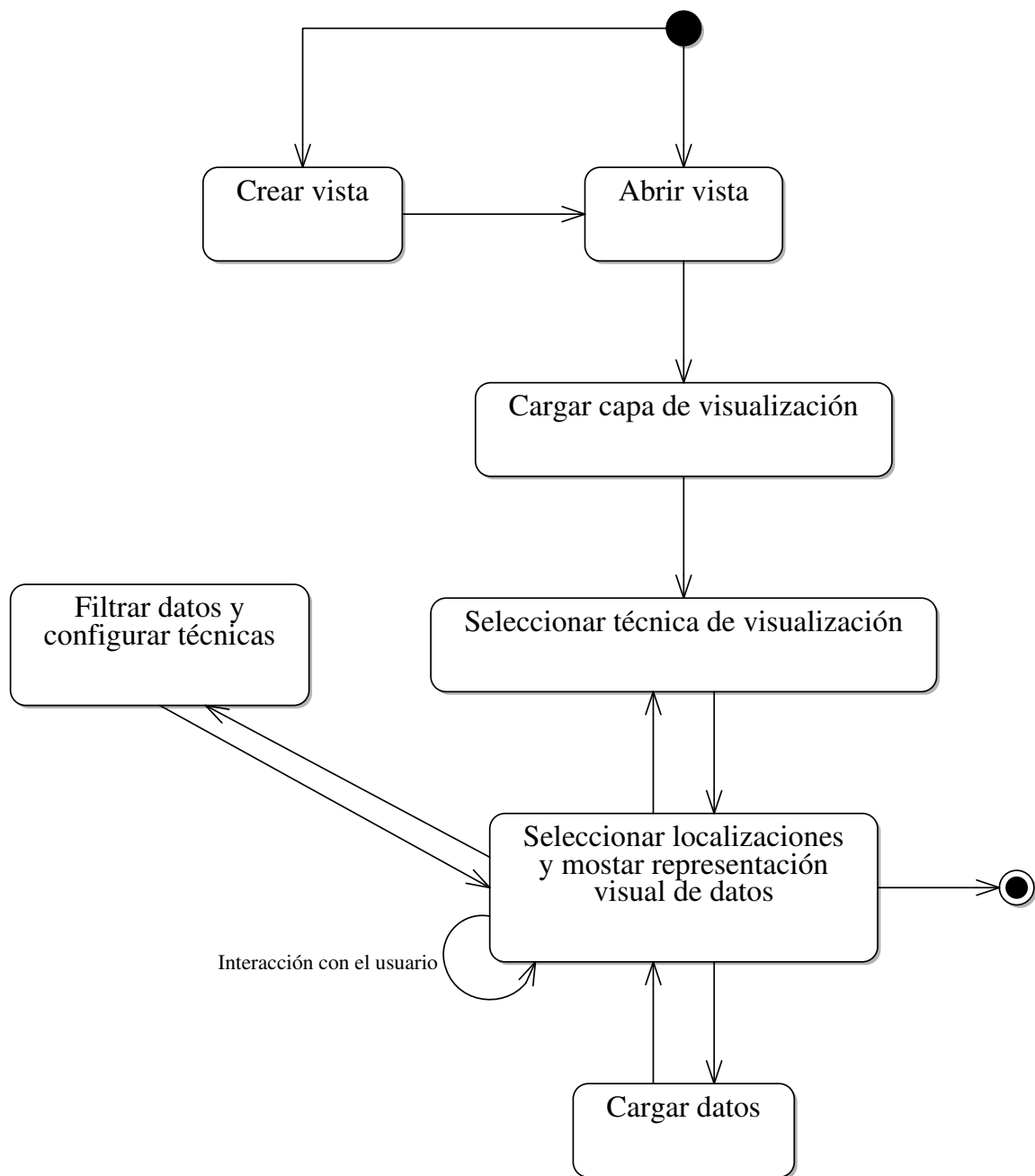


Figura 2.2: Diagrama de transición de estados correspondiente al caso de uso «Visualizar de forma coordinada».

Si el usuario hace clic en otra localización se repite el proceso para esa nueva celda. Si el usuario desea modificar los parámetros de visualización pasa al estado *Filtrar datos y configurar técnicas*. Durante este estado se puede seleccionar las variables que se quieren visualizar, el porcentaje de los datos que se quiere mostrar, seleccionar de acuerdo a un rango de valores de un atributo, entre otras opciones.

Además cada tipo de técnica tiene un panel de configuración que permite personalizar las características específicas de cada una de ellas. Luego de realizar los cambios deseados se retorna al estado *Seleccionar localizaciones y mostrar representación visual de datos* para mostrar los cambios realizados. En este estado el usuario puede interactuar con las técnicas.

2.1.3. Acceso a datos

Para el acceso a los datos se realizó el diseño de un paquete que permitiera proveer una interfaz uniforme para diversas fuentes de datos. De esta manera se decidió utilizar esta interfaz para proveer el acceso a los datos en el formato HDF. En la Figura 2.3 se muestra el diagrama de clases del paquete resultante de este diseño.

En la parte superior de la jerarquía de clases se encuentra la interfaz `DataSource`. Esta interfaz define el comportamiento que debe tener una fuente de datos multiparamétrica para ser visualizada. Adicionalmente fueron definidas las interfaces `TimeDataSource` y `SpaceDataSource` como extensiones que permiten además contener varios conjuntos multivariados para distintos espacios de tiempo o para distintas localizaciones en un mapa respectivamente. Para la primera de estas se decidió no proveer una implementación, aunque podría ser de utilidad para futuras extensiones al módulo. Se decidió realizar una implementación base de un `DataSource` que facilitaría la inclusión de soporte para otros tipos de fuentes de datos. La clase `DefaultDataSource` realiza la implementación de todos los métodos definidos en la interfaz. Para ello debe construirse utilizando un arreglo que represente un conjunto de datos.

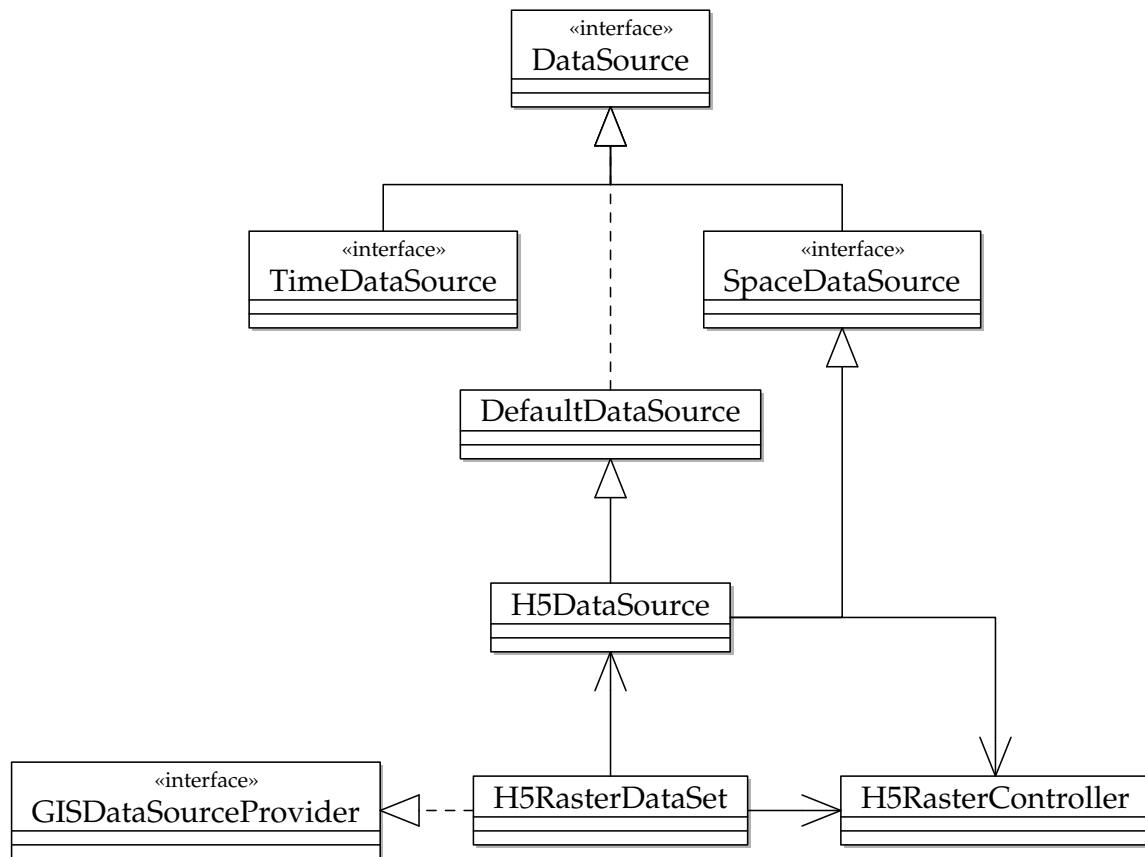


Figura 2.3: Diagrama de clases del paquete `uclv.gvsig.sv.datasource`.

La clase `H5DataSource` provee los medios para acceder al formato HDF como una fuente de datos para visualización coordinada. Por ello además de heredarse de la clase `DefaultDataSource`, se implementa la interfaz `SpaceDataSource`. De esta manera se permitirá acceder a un conjunto de datos dado un punto en el mapa. Para poder crear una capa ráster en gvSIG se creó una subclase de `RasterDataset` del paquete `org.gvsig.raster.dataset` denominada `H5RasterDataset`. Esta clase además implementa la interfaz `GISDataSourceProvider` lo que permitirá acceder a la fuente de datos desde una capa en gvSIG. Por último, el acceso real a los datos en el formato HDF se realiza por `H5Controller` utilizando la biblioteca `HDF-Java`.

2.1.4. Visualización coordinada

Uno de los objetivos del paquete para la visualización coordinada de varios conjuntos de datos es poder obtener una percepción sobre la localización de los datos visualizados. Para ello se hace necesario contar con una ventana o vista donde se puedan cargar los mapas para la localización de los gráficos y permitir la visualización de las técnicas. La elección de un Sistema de Información Geográfica permite la utilización de sus capacidades para visualizar mapas como base de la estructura que permitirá incorporar técnicas de visualización científica. En el caso específico de gvSIG se toma como base los proyectos para visualizar vistas, que permite la carga de diferentes formatos de capas ráster y vectoriales. Como resultado el paquete de visualización (cf. Figura 2.4) se encarga mayormente de los aspectos relacionados con la visualización científica, pues los distintos aspectos relacionados con la visualización del mapa son delegados a gvSIG.

Como parte superior dentro de la jerarquía de clases se encuentra la interfaz `VisualizationView`. Esta interfaz define el comportamiento que debe asumir una vista destinada a la visualización científica (ya sea sobre el mapa o de manera independiente). Adicionalmente se proveyó dos implementaciones por defecto. En el caso de `IndependentView` se utilizó la API Swing para implementar una vista destinada a la visualización independiente. Específicamente se realizó una subclase de `JPanel` y se implementó la interfaz `IWindow` del paquete `com.iver.andami.ui.mdiManager` provista por gvSIG para lograr una interacción adecuada con el sistema de ventanas de la aplicación. Por otra parte `MapView` es una implementación de `FGraphic` del paquete `com.iver.cit.gvsig.fmap.rendering`. Esta clase está destinada a la visualización sobre el mapa —específicamente una capa de gvSIG— delegando las acciones necesarias para el dibujo sobre el mapa a la API provista por el sistema. De manera adicional se ofrecen subclases más elaboradas que permiten delegar las acciones de dibujo a clases especializadas, mientras se encargan exclusivamente del manejo de la vista.

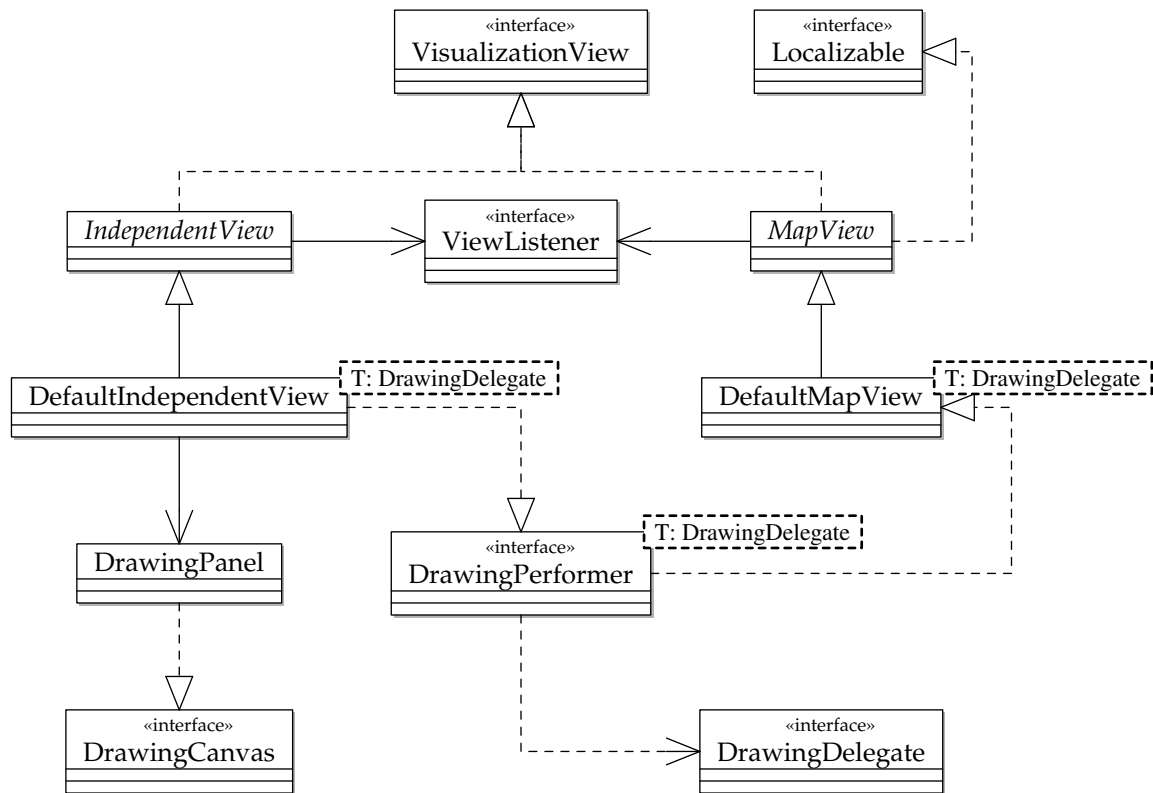


Figura 2.4: Diagrama de clases del paquete `uclv.gvsig.sv.view`.

2.2. Implementación de la extensión

En este epígrafe se exponen los temas referentes a la implementación de la extensión teniendo en cuenta las tecnologías utilizadas para la implementación y algunas consideraciones sobre las técnicas implementadas.

2.2.1. Elección de las tecnologías necesarias para la implementación

Durante el desarrollo o extensión de un sistema deben tomarse decisiones de implementación como la plataforma objetivo y el lenguaje de desarrollo. En cada pro-

yecto hay que puntualizar en determinadas áreas específicas de la aplicación. En el caso particular de este proyecto debe fijarse la API gráfica a utilizar.

Lenguaje de programación

El Sistema de Información Geográfica gvSIG es un software desarrollado completamente en el lenguaje de programación Java™. La forma más común y sencilla para desarrollar extensiones orientadas a esta herramienta están basadas precisamente en este lenguaje de programación. Ello se debe al alto nivel de integración que se logra en el proceso de interacción con gvSIG, difícilmente alcanzable con otros lenguajes de programación dado el reto que significa la interacción con la plataforma Java™. Aún más, Java™ es un lenguaje potente, popular y multiplataforma con una amplia comunidad de usuarios y un amplio soporte para el desarrollo de interfaces gráficas de usuario. Debido a ello este lenguaje se convierte en la elección obvia para el desarrollo de la extensión propuesta.

API para gráficos

Las técnicas que se decidieron implementar no requieren de gran complejidad en la API gráfica. En particular no necesitan de una API 3D. Por esto se decidió utilizar la tecnología Java 2D™ que brinda un excelente desempeño y potencialidad para la programación gráfica en dos dimensiones. Esta decisión facilita a los usuarios la utilización del sistema ya que no necesitan paquetes adicionales a los de la plataforma Java™.

La API Java 2D™ está constituida por un conjunto de clases que pueden ser usadas para crear gráficos de elevada calidad. Esto incluye técnicas como: transformaciones geométricas, disminución de curvas o líneas causadas por una baja resolución del gráfico, dibujo de mapas de píxeles, procesamiento de imágenes, diseño bidireccional de texto, además de excelentes vías para el uso del color, solo por men-

cionar algunos. Se utiliza esta API para crear las representaciones visuales de las técnicas de visualización.

Entorno de desarrollo

Fue escogido Eclipse Kepler (Eclipse RCP 4.3) como Entorno de Desarrollo Integrado (*Integrated Development Environment*, IDE). Eclipse es uno de los entornos más populares para la programación en Java™ con una amplia comunidad de usuarios. Es software libre bajo la Licencia Pública de Eclipse (*Eclipse Public License*, EPL). Provee técnicas avanzadas de refactorización, análisis de código y un sofisticado ambiente para la depuración de programas. Su arquitectura completamente modular ha condicionado la elaboración de multitud de extensiones que permiten incrementar las funcionalidades provistas por defecto.

2.2.2. Metodología para la adición de nuevas técnicas de visualización

Existen varias formas de implementar una nueva técnica de visualización científica al módulo de geovisualización. La elección de la más apropiada está dada principalmente en el grado de flexibilidad del que se pretende dotar a la nueva visualización. Como puede preverse el aumento de la flexibilidad obtenida conlleva asimismo una complejidad creciente a la hora de implementar la nueva técnica y viceversa.

La forma más sencilla de incorporar una nueva técnica es mediante la implementación de la interfaz `DrawingDelegate` o mediante la creación de una subclase de `BaseDrawingDelegate`. `BaseDrawingDelegate` se encarga de implementar las rutinas necesarias para manejar la fuente de datos de acuerdo a la definición de la interfaz y configura los aspectos más generales del dibujo. De esta manera solo debe implementarse el método `draw` que permite dibujar sobre un objeto `Graphics2D`. Es en este método donde debe incluirse la rutina de visualización.

En el caso de las vistas independientes es común proveer diversos niveles de interacción con el usuario utilizando las capacidades de AWT o Swing. Por ello se añadió soporte para realizar esto sin tener que implementar completamente una nueva vista. Simplemente se debe crear una subclase de `DefaultIndependentView` y proveer un objeto `DrawingCanvas` apropiado. Para simplificar aún más esta operación se provee una implementación de esta interfaz en una subclase de `JPanel` llamada `DrawingPanel`. En ella solo deben definirse las rutinas de visualización correspondientes a la técnica y las interacciones pertinentes usando la API provista por Swing.

Las capacidades provistas por defecto por las subclases de `VisualizationView` incluidas en el módulo podrían ser, no obstante, insuficientes o no completamente satisfactorias. En este caso podría ser conveniente crear subclases de una de ellas o directamente implementar la interfaz `VisualizationView`.

En todos los casos es necesario indicar mediante metadatos (annotations en Java) la información necesaria para registrar la técnica de visualización como parte del módulo. En cada caso las anotaciones deben realizarse sobre el tipo implementado, ya sea `VisualizationView` o `DrawingDelegate`. A continuación se realiza una explicación de las anotaciones necesarias.

@Info Indica la información a mostrar (nombre y descripción) sobre esta técnica de visualización en el menú de selección de técnicas seleccionado. En caso de no establecerse esta anotación la técnica utilizará el nombre de la clase en la que fue implementada y no poseerá descripción.

@VisualizationType Establece el tipo de vista (sobre el mapa o independiente) que mostrará la técnica implementada. En la práctica esto indica que se registrará la visualización en el menú de selección de técnicas apropiado. No obstante la elección no es exclusiva, puede escogerse mostrar la técnica en ambas vistas. El comportamiento por defecto es que esta anotación solo tiene efecto en caso de que modifique a una clase de tipo `VisualizationView` o `DrawingDelegate`, sin embargo ello no está requerido y las extensiones

al módulo podrían aumentar el número de clases afectadas. `MapView` e `IndependentView` están anotadas inicialmente con `OVER_MAP` e `INDEPENDENT` respectivamente. Este comportamiento afecta a todas las subclases a menos que la anotación sea sobrescrita. Si se modifica, en cambio, a una clase de tipo `DrawingDelegate` provoca adicionalmente que se elija entre `DefaultMapView` y `DefaultIndependentView` como la vista que servirá de envoltorio para la técnica implementada. Esta anotación es obligatoria en tanto de no establecerse queda a cuenta del programador proveer los medios para que el usuario pueda acceder a la visualización.

@ConfigurationView En esta anotación se especifica la clase del panel de configuración para la técnica seleccionada. En la práctica el panel será mostrado en la vista de configuración apropiada de acuerdo a como haya sido clasificada la técnica por `@VisualizationType`. Esta anotación es opcional, en caso de no habilitarse se mostrará un panel especificando que no existe configuración para la técnica.

@Factory Establece de ser necesario la clase encargada para construir un nuevo objeto de la visualización correspondiente de acuerdo al patrón de diseño *Factory*. Adicionalmente en la clase definida para este propósito el método estático encargado de la operación deberá ser marcado con la anotación `@FactoryMethod`.

2.2.3. Consideraciones sobre las técnicas implementadas

Figuras con palillos

La técnica de visualización «Figuras con palillos» fue implementada para su visualización en paneles independientes. La técnica y su configuración están implementados en el paquete `...techniques.stickfigure`. La visualización muestra como resultado un conjunto de íconos (uno por cada registro de datos). Los íconos poseen 12 características configurables: ángulos de inclinación de cada una de las

4 extremidades y del cuerpo, tamaño de las extremidades y del cuerpo, grosor de los palillos y proporción del tamaño del ícono. El usuario puede asignar valores directamente a los dos últimos parámetros. Cada uno de los restantes 10 parámetros deben ser asignados a una variable del conjunto de datos. De esta manera sus valores dependerán de la asignación realizada tomando en cuenta el valor del registro asignado al ícono.

Coordenadas de estrella

La técnica de visualización «Coordenadas de estrella» fue implementada para su visualización en paneles independientes. La técnica y su configuración están implementadas en el paquete `...techniques.starplot`. La visualización muestra una disposición de los ejes de manera circular, asociados a cada una de las variables del conjunto de datos. El gráfico resultante puede ser configurado para establecer el tamaño y color a conveniencia del usuario.

Espiral de tiempo

La técnica de visualización «espiral de tiempo» fue implementada para su visualización en paneles independientes. La técnica y su configuración están implementadas en el paquete `...techniques.spiral`. La visualización muestra como resultado una espiral de Arquímedes por cada variable en el conjunto de datos. Cada una de las espirales está dividida en un número igual al de registros de datos. La espiral puede ser configurada para establecer un período de tiempo a conveniencia del usuario. Adicionalmente se puede configurar el tamaño, grosor y color de la espiral.

Vista circular

La técnica de visualización «vista circular» fue implementada para su visualización en paneles independientes y sobre el mapa. La técnica y su configuración están implementadas en el paquete `...techniques.circlevew`. La visualización muestra un círculo dividido en un número de segmentos igual al número de variables en el conjunto de datos. A su vez estos se dividen en un número igual al número de registros en la muestra analizada. La técnica permite configurar el tamaño de la circunferencia y los colores del gradiente.

2.3. Conclusiones

La integración de técnicas de visualización científica para datos multiparamétricos en el SIG gvSIG permite al usuario visualizar uno o varios conjuntos de datos, de forma coordinada, obteniendo una percepción espacial del origen de los datos visualizados.

Se seleccionaron un conjunto de técnicas de visualización de datos multiparamétricos que fueron incorporadas a gvSIG. Las técnicas incluidas en el módulo de visualización ofrecen un amplio número de beneficios. Se incluyeron algunas de las técnicas tradicionales, con ciertas mejoras, lo que permite mostrar conjuntos de datos con disímiles características.

Se presentó el análisis de los actores y casos de uso del sistema. Se mostró la arquitectura general del sistema y el formato de dato que se utilizará. Se diseñaron e implementaron las nuevas funcionalidades para el módulo de visualización y se mostraron los cambios que fueron necesarios realizar en la estructura de clases para poder brindar las nuevas funcionalidades.

Además el proceso de diseño del nuevo módulo admite la realización de futuras extensiones y modificaciones, quedando propuesta una guía para la adición de nuevas técnicas.

3 USO Y VALIDACIÓN DEL SISTEMA

En el presente capítulo se realiza una presentación de las principales funcionalidades del módulo de visualización. Se efectúa un análisis detallado de las opciones y modos de uso de cada una de las formas de visualización implementadas, que constituye una guía para explotar los beneficios de la extensión realizada. También se presenta como caso de estudio el análisis visual de los datos climáticos mundiales recolectados por la unidad de investigación climática de la Universidad del Este de Anglia, en el Reino Unido con la herramienta desarrollada.

3.1. Uso del sistema

La extensión realizada a gvSIG no adiciona nuevos requerimientos del sistema para su ejecución. Para utilizar la extensión basta copiar el módulo compilado en el directorio de extensiones de gvSIG. Se recomienda utilizar la versión 1.12 de gvSIG, utilizada en las pruebas al módulo. El análisis visual de datos utilizando las técnicas de visualización con que cuenta la extensión implementada, requiere un conocimiento previo de cómo aplicarlas de forma adecuada para lograr una correcta comprensión de los resultados que se obtengan.

La extensión realizada a gvSIG permite la exploración y análisis visual de datos mediante las técnicas: Coordenadas paralelas, Gráfico de Andrews, Segmentos de círculo, Patrones recursivos, Caras de Chernoff, Figuras con palillos, Matriz de dispersión, *Table Lens*, Rueda de tiempo, *Multicomb*, Coordenadas de estrella, Espiral

de tiempo, Vista circular y Mapas autoorganizados. Permite visualizar coordinadamente varios conjuntos de datos donde los atributos de la visualización representan el comportamiento global de las variables para la totalidad de los datos.

3.2. Características fundamentales

El usuario del sistema tiene la posibilidad de realizar el análisis visual de varios conjuntos de datos de forma coordinada, teniendo una percepción del origen de los datos que están siendo visualizados referenciando las técnicas a un mapa. Para realizar esta operación es necesario tener un proyecto de vista en gvSIG en el que se pueda cargar una capa de visualización que contiene el archivo de datos y el mapa sobre el que se realizarán las visualizaciones. En el caso de la implementación provista por defecto, la capa es almacenada en un archivo HDF que contiene tanto los datos como el Modelo Digital de Elevaciones (MDE) utilizado como mapa ráster. Cuando se dispone de un proyecto previamente creado el primer paso para realizar la visualización lo constituye crear una vista de visualización.

La Figura 3.1 muestra los pasos para la creación de una vista de visualización coordinada. Se debe ① seleccionar el tipo de documento de visualización, ② crear un nuevo documento de visualización oprimiendo el botón Nuevo, ③ seleccionar el documento creado y ④ abrirlo oprimiendo el botón Abrir. Realizados estos pasos ⑤ se muestra una vista dispuesta para la visualización.

Al estar activa una vista se habilita la opción de carga de capa. Al seleccionar esta opción se muestra un diálogo para la selección de la capa (en este caso de visualización), que constituye el proyecto de visualización coordinada con el que se desea trabajar. Una vez seleccionado el archivo, se carga el mapa definido y los datos asociados (ver Figura 3.2).

La Figura 3.3 muestra los diferentes componentes de una vista luego de cargar una capa de visualización. A la izquierda se activa la vista de configuración de las visualizaciones. Este panel está separado en tres pestañas correspondientes a ① las

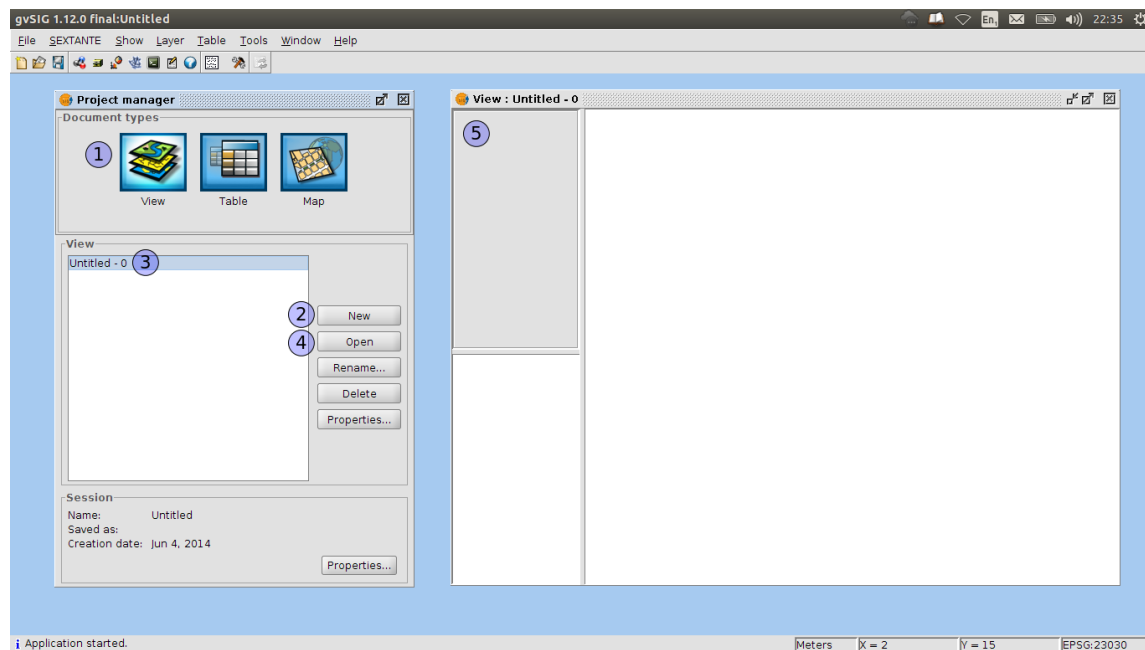


Figura 3.1: Creación de una vista de visualización coordinada.

configuraciones generales, ② las configuraciones de las técnicas mostradas sobre el mapa y ③ las configuraciones de las técnicas mostradas como vistas independientes. La región principal ④ constituye el área de visualización del mapa y las técnicas proyectadas sobre este.

3.2.1. Panel de configuración general de las técnicas

El panel de configuración general (Figura 3.4) está dividido en áreas dedicadas a ① la configuración de la fuente de datos y ② la configuración de los parámetros de dibujo, específicamente el nivel de calidad de los gráficos generados durante el proceso de visualización. Las opciones de configuración de los datos permiten la selección de los atributos a analizar ③, así como establecer el orden entre estos. Adicionalmente puede obtenerse un subconjunto de los datos ④ al establecer el por ciento de los registros que se quieren visualizar. Otra opción disponible es la

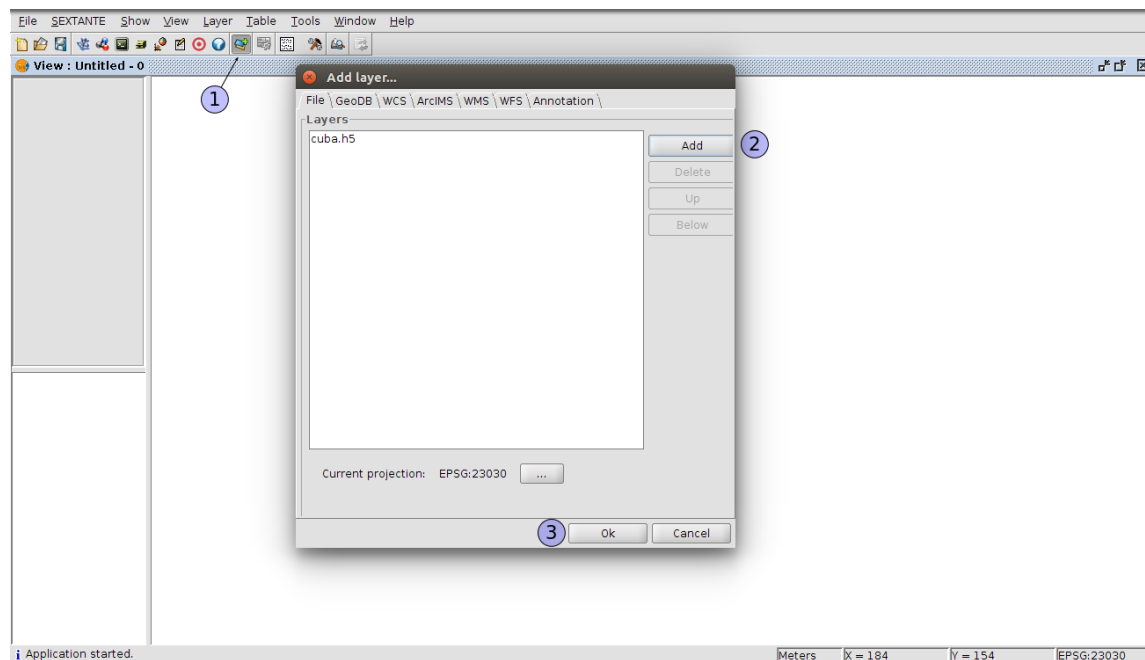


Figura 3.2: Opciones para la adición de un proyecto de visualización coordinada a una vista de visualización.

aplicación de un filtro ③ sobre los datos a partir de los valores de los distintos atributos.

La aplicación del filtro se realiza a través del cuadro de diálogo de selección de filtros (Figura 3.5). En este se permite establecer una expresión booleana a partir de comparaciones entre los atributos y valores suministrados por el usuario. Estas expresiones son a su vez enlazadas por operaciones lógicas de conjunción (\wedge) o disyunción (\vee).

3.2.2. Espiral de tiempo

La técnica de Espiral de tiempo, accesible desde la opción de selección de técnicas no es visualizada sobre el mapa. La visualización de cada conjunto de datos se realiza en paneles independientes y se agrega el nombre de la técnica y el número

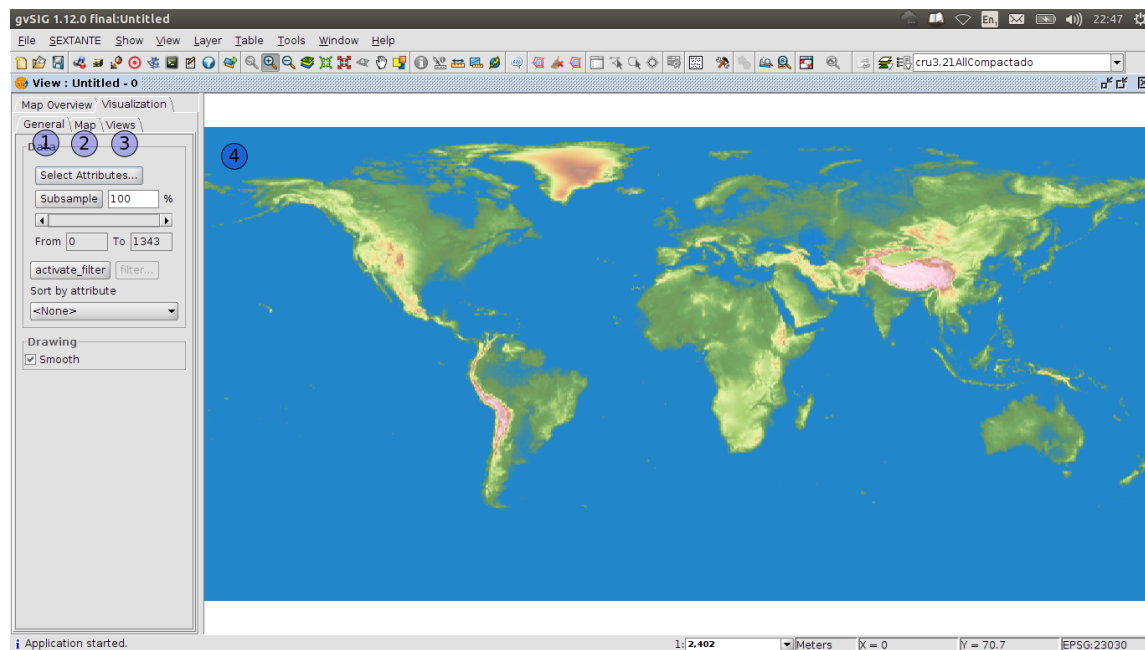


Figura 3.3: Vista de visualización.

de la localización que le corresponde en el mapa. Su implementación permite la configuración de diferentes parámetros asociados a la visualización de la espiral de Arquímedes que funciona como base de la técnica. La Figura 3.6 muestra un ejemplo de lo anteriormente expuesto.

A continuación se enumeran las opciones disponibles en el panel de configuración de la técnica:

Atributo Establece el atributo utilizado para realizar la visualización.

Periodo Establece el periodo base p de la visualización. De esta manera se mostrarán p registros en cada vuelta de la espiral.

Grosor Permite establecer el grosor de de la espiral de Arquímedes utilizada como base de la visualización.

Gradiente Seleccionar los colores base del gradiente utilizado para establecer la coloración de cada uno de los segmentos representativos de los registros.

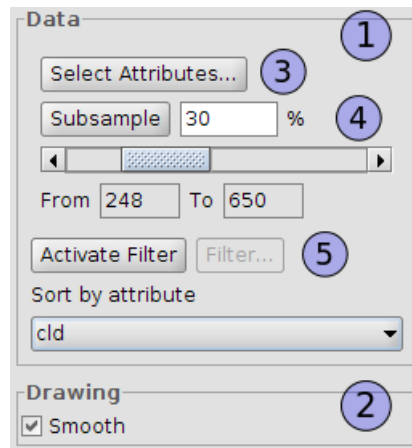


Figura 3.4: Panel de configuración general.

Separación Establece la separación existente entre las vueltas sucesivas de la espiral.

3.2.3. Coordenadas de estrella

La técnica de Coordenadas de estrella, accesible desde la opción de selección de técnicas permite ser visualizada tanto sobre el mapa como en paneles independientes. En el caso de la visualización en paneles independientes se agrega el nombre de la técnica y el número de la localización que le corresponde en el mapa. Esta técnica permite la configuración de diferentes aspectos del dibujo como el tamaño del gráfico, así como el color y grosor de las líneas. La Figura 3.7 permite ilustrar lo antes descrito.

A continuación se enumeran las opciones disponibles en el panel de configuración de la técnica:

Etiquetas Cuadro de selección que indica si se deben mostrar u ocultar las etiquetas correspondientes a los nombres de cada uno de los atributos. En el caso de la visualización independiente si está desactivada esta opción se oculta-

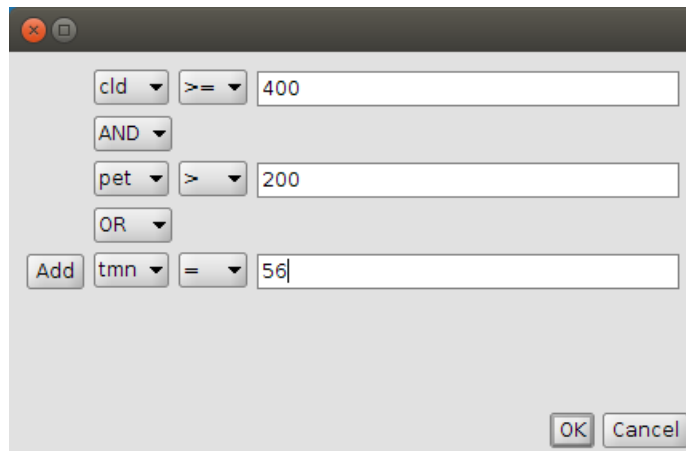


Figura 3.5: Diálogo de filtrado de datos.

rán las etiquetas a simple vista y solo se mostrarán al recorrer los ejes con el uso del ratón.

Grosor Indica el grosor de las líneas del gráfico de estrella.

Tamaño Permite seleccionar el tamaño del gráfico visualizado.

Color por atributo Permite seleccionar un atributo para ser usado como base para la coloración del gráfico. De esta manera el color de las líneas será seleccionado de acuerdo a un gradiente definido usando los valores de dicho atributo.

Color Selecciona el color o el gradiente —en caso de que se elija un atributo como base de la coloración— para ser utilizado en la coloración de las líneas del gráfico.

3.2.4. Vista circular

La técnica de Vista circular, accesible desde la opción de selección de técnicas permite ser visualizada tanto sobre el mapa como en paneles independientes. En el caso de la visualización en paneles independientes se agrega el nombre de la técnica y el número de la localización que le corresponde en el mapa. Esta técnica

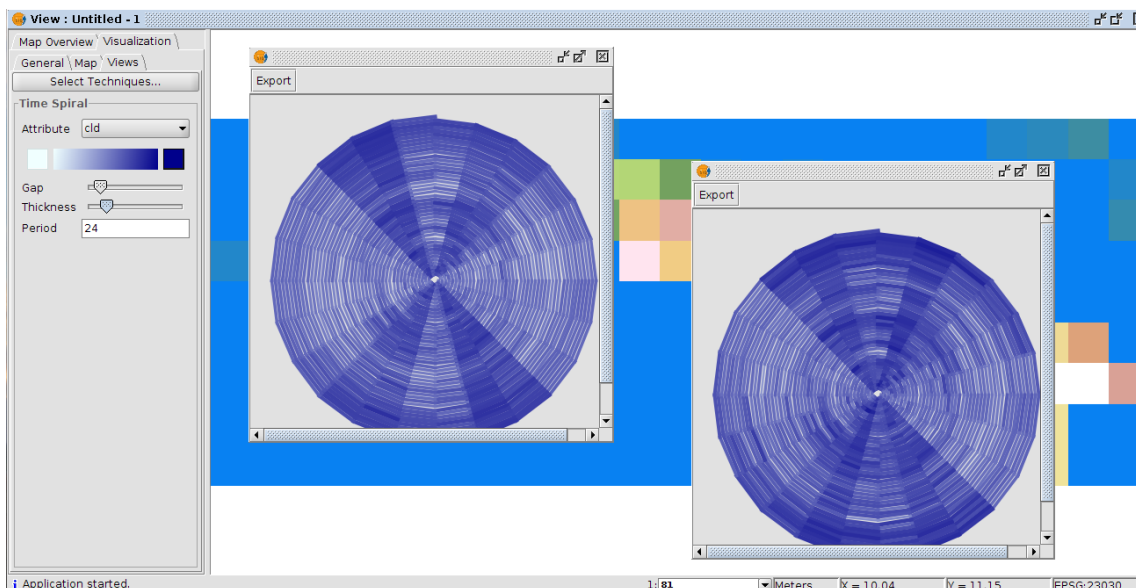


Figura 3.6: Visualización del gráfico en espiral.

permite la configuración de diferentes aspectos del dibujo como el tamaño del gráfico, así como los colores utilizados en el gradiente. La Figura 3.8 permite ilustrar lo antes descrito.

A continuación se enumeran las opciones disponibles en el panel de configuración de la técnica:

Gradiente Permite seleccionar los colores base del gradiente a utilizar para la coloración del gráfico.

3.2.5. Figuras con palillos

La técnica de Figuras con palillos, accesible desde la opción de selección de técnicas no es visualizada sobre el mapa. La visualización de cada conjunto de datos se realiza únicamente sobre paneles independientes y se agrega el nombre de la técnica y número de la localización que le corresponde en el mapa. Esta técnica muestra la asignación de las variables a cada rasgo configurable del ícono, permiti-

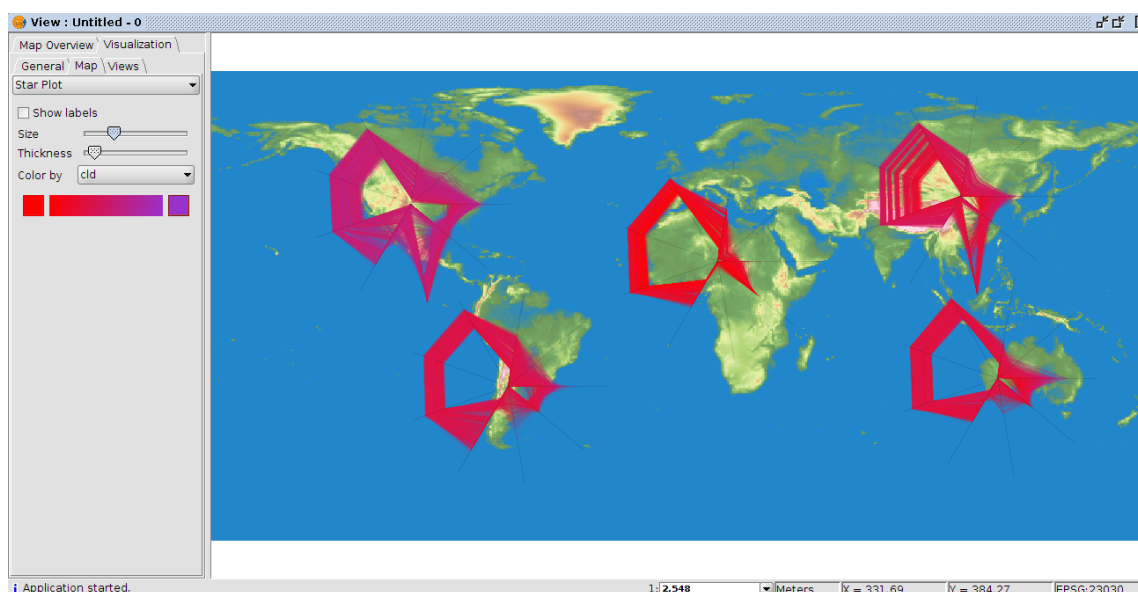


Figura 3.7: Visualización del gráfico en coordenadas de estrella.

te modificar esa asignación y además permite cambiar el tamaño y grosor de las figuras. La Figura 3.9 muestra un ejemplo de lo anteriormente expuesto.

A continuación se enumeran las opciones disponibles en el panel de configuración de la técnica:

Tamaño Configura el tamaño de los íconos a visualizar.

Grosor Establece el grosor de la figura.

Columnas Establece el número n de columnas utilizadas en la visualización. De esta manera se dispondrán filas de n íconos cada una.

Filas Alternativamente se puede elegir el número m de filas utilizadas en la visualización. De esta manera se dispondrán columnas de m íconos cada una.

Parámetros dependientes de las variables Existen 10 parámetros configurables que dependen del valor de las variables. El usuario indica mediante un cuadro de selección la variable asociada a un parámetro determinado. Los pará-

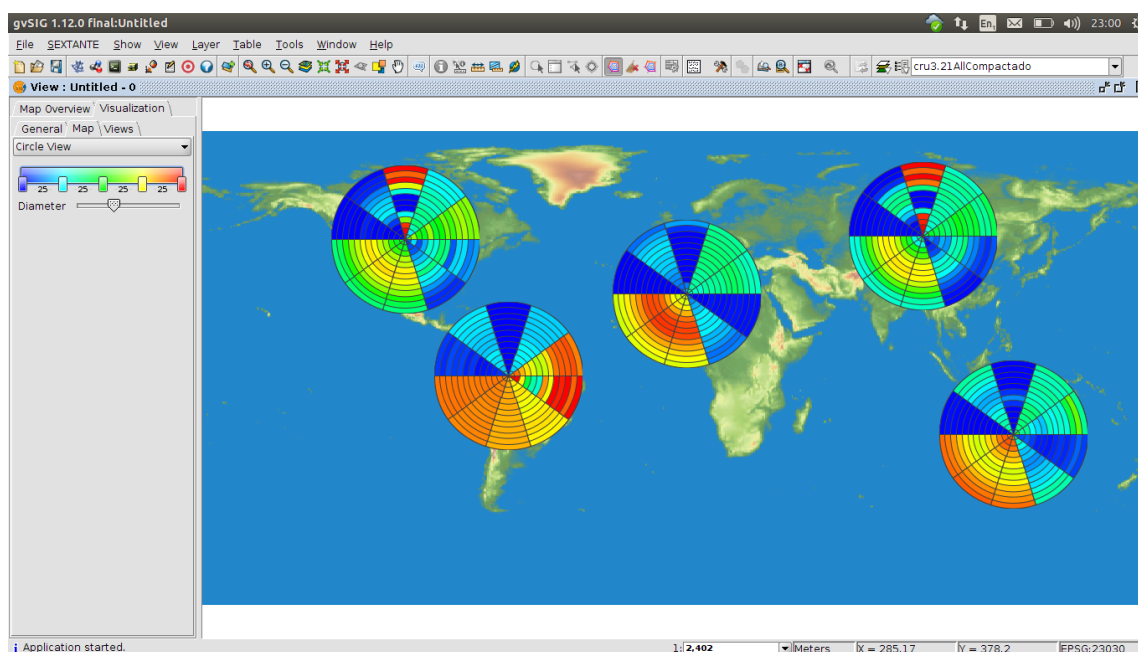


Figura 3.8: Visualización en vista circular.

metros utilizados están distribuidos en diferentes aspectos concernientes al tamaño e inclinación de brazos, piernas y cuerpo.

3.3. Caso de estudio: Datos climáticos mundiales

En este estudio se usa el conjunto de datos CRU TS 3.21 (Harris y col. 2013). Este conjunto de datos incluye 1344 registros mensuales de 10 variables climáticas durante 110 años para el período de 1901 hasta 2012. Cubre la superficie terrestre a una resolución de 0.5 grados. En la Tabla 3.1 se muestran dichas variables climáticas.

Será analizado un caso de estudio centrado en Cuba con el objetivo de mostrar las capacidades del sistema para la detección y estudio de fenómenos climáticos. Para la realización del análisis fueron seleccionadas las técnicas de patrones recursivos,

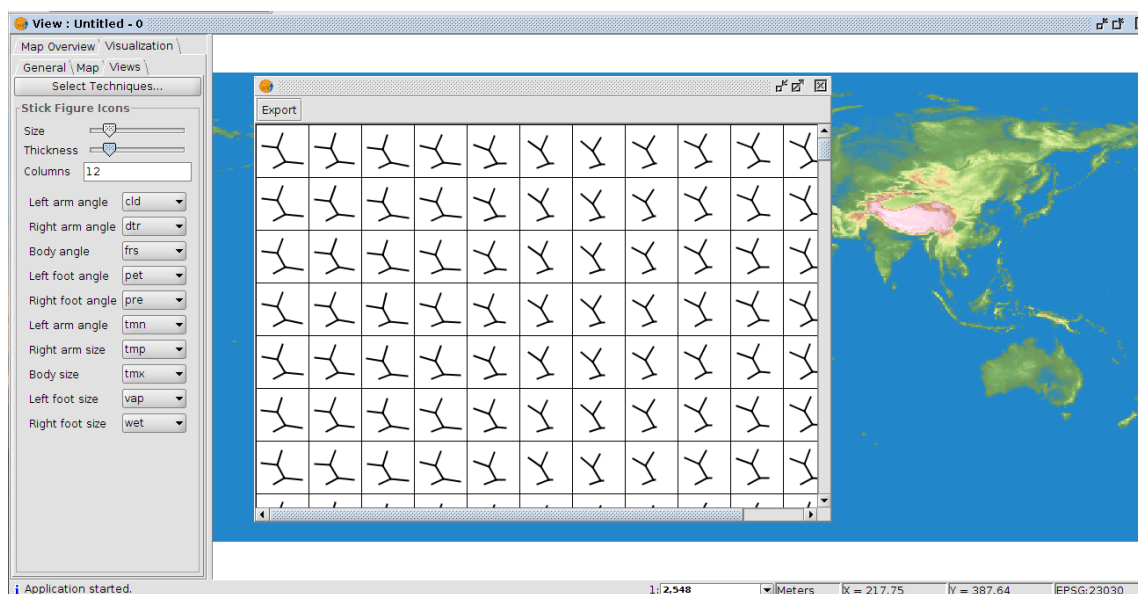


Figura 3.9: Visualización de figuras con palillos.

Table Lens y *Espiral de tiempo*. A la técnica de patrones recursivos se la aplicó el patrón $(12, 1)(1, 112)$ para la disposición de los datos de los 12 meses de izquierda a derecha y los 112 años de arriba hacia abajo. La técnica de visualización en espiral fue configurada para un periodo de 12. De esta manera se hace corresponder cada vuelta de la espiral con uno de los años analizados.

Se aplicaron las mencionadas técnicas de visualización sobre distintos puntos de la geografía cubana en la búsqueda de patrones o fenómenos «interesantes». Se mostrarán resultados obtenidos como parte del análisis del territorio oriental de la isla. En diversos puntos de la geografía oriental fue detectado un comportamiento peculiar. Al realizar el análisis con la técnica de visualización de patrones recursivos pudo observarse un valor anómalo en la variable correspondientes a las precipitaciones. El valor alterado está presente en varios puntos en el mismo espacio de tiempo, lo que indicaría un fenómeno climático que afectó gran parte de la región mencionada. En la Figura 3.10 se muestra la visualización en uno de los puntos analizados; la coloración amarilla del registro mencionado indica un valor inusualmente alto.

Etiqueta	Variable	Unidad
cld	Cobertura nubosa	%
dtr	Rango de temperatura diurna	°C
frs	Frecuencia diaria de escarcha	días
pet	Evotranspiración potencial	mm
pre	Precipitaciones	mm
tmp	Temperatura media diaria	°C
tmn	Promedio mensual de temperatura mínima diaria	°C
tmx	Promedio mensual de temperatura máxima diaria	°C
vap	Presión de vapor	hPa
wet	Frecuencia de humedad diaria	días

Tabla 3.1: Variables climáticas incluidas en el conjunto de datos analizado.

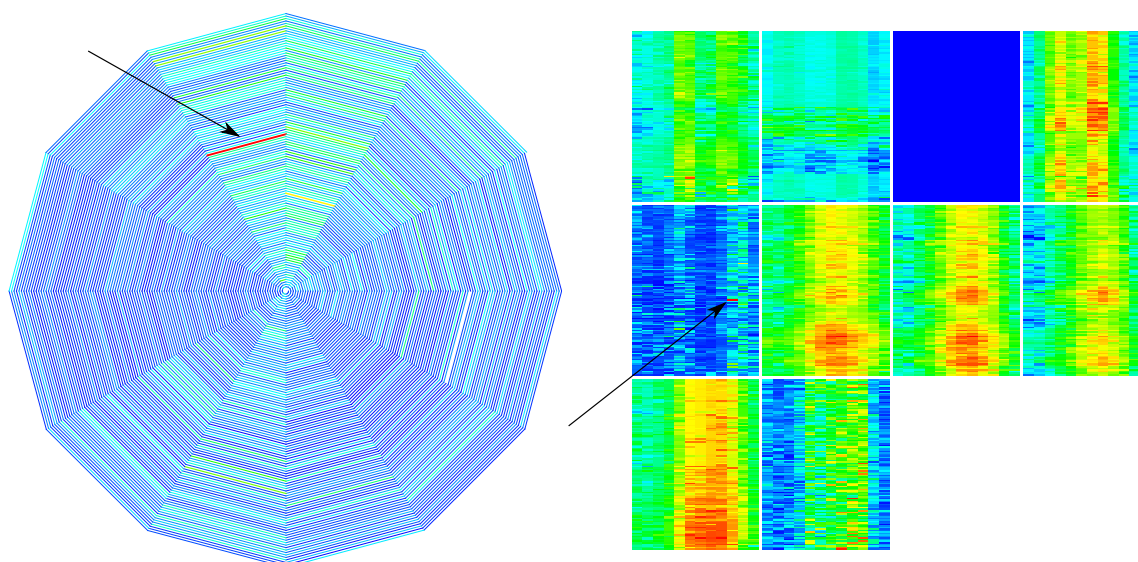


Figura 3.10: Visualización de registros climáticos de la región oriental de Cuba utilizando patrones recursivos y espiral de tiempo.

La realización del análisis de la misma región con la técnica de visualización *Table Lens* (ver Figura 3.11) permitió constatar que corresponde al mes de octubre de 1963 —registro 753—. Para el punto mostrado este mes registró un acumulado de precipitaciones de 747,9 mm y coincide con el paso del huracán «Flora» por la región oriental del país.

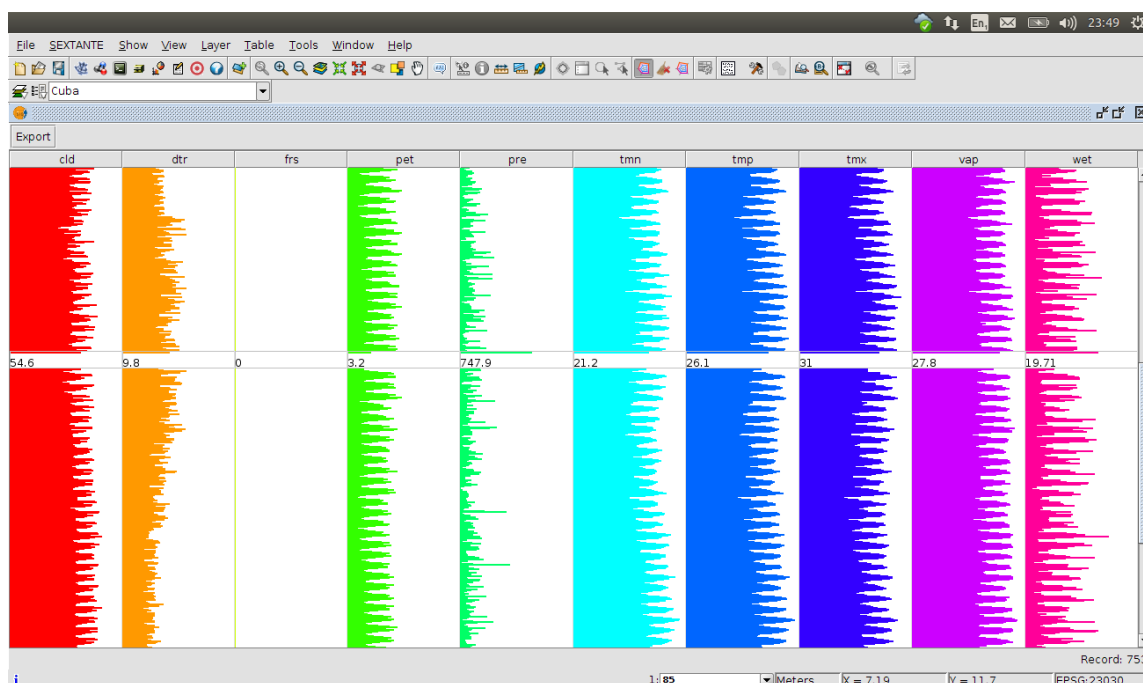


Figura 3.11: Visualización de registros climáticos de la región oriental de Cuba utilizando *Table Lens*. El foco se encuentra aplicado al registro 753 (octubre de 1963).

Como se puede observar mediante la técnica de espiral de tiempo (Figura 3.10) el valor es comparativamente superior a otros meses lluviosos del mismo año y destaca con respecto a registros del mismo mes durante los años analizados. Este valor inusual marca un indicio para la consideración del fenómeno meteorológico antes mencionado como causa fundamental de tales registros.

3.4. Conclusiones

En el presente capítulo se expusieron los aspectos fundamentales para el uso del módulo de visualización de datos multiparamétricos incorporado a gvSIG. Esto constituye una importante guía para el uso eficiente de la herramienta por parte de los usuarios. Se mostró un caso de estudio donde se evidencia la factibilidad del uso del módulo para el análisis visual de los datos climáticos mundiales recolectados por la unidad de investigación climática de la Universidad del Este de Anglia, en el Reino Unido. En este se pudo constatar las ventajas que brinda la herramienta para explorar datos y obtener información espacio-temporal relevante sobre los mismos.

CONCLUSIONES

Como resultado de esta investigación se implementaron nuevas técnicas de visualización científica al Sistema de Información Geográfica gvSIG para el análisis multivariado de grandes volúmenes de datos climáticos, cumpliéndose el objetivo general planteado. Además se presentaron los siguientes aspectos:

- Se seleccionaron un conjunto de técnicas para la visualización de datos teniendo en cuenta sus capacidades para el análisis visual de datos, de manera que facilitaran la exploración coordinada de grandes volúmenes de información espacio-temporal.
- Se incorporó una nuevo diseño para la adición de nuevas técnicas al módulo, para facilitar las futuras extensiones. Este diseño fue utilizado para la implementación e incorporación de las técnicas seleccionadas.
- Se realizó una extensión del soporte del módulo para los formatos de datos científicos mediante la definición de un conjunto de interfaces para la entrada de datos y la realización de una implementación para el formato HDF.
- Se presentó un caso de estudio donde se realiza el análisis visual de los datos climáticos mundiales publicados en la unidad de investigaciones climáticas de la universidad del Este de Anglia, en el Reino Unido, que evidencia la idoneidad del uso del módulo de visualización desarrollado para el análisis visual de datos.

Aunque el módulo desarrollado es probado con datos climáticos las herramientas implementadas son de propósito general, permitiendo su utilización en múltiples campos de aplicación.

RECOMENDACIONES

- Implementar nuevas técnicas de visualización para su adición al módulo de geovisualización de gvSIG.
- Perfeccionar el sistema de consultas y filtrado de datos para obtener una mayor flexibilidad y efectividad en la selección de los datos necesarios.
- Probar la efectividad del análisis visual de datos georreferenciados en otros dominios de aplicación utilizando la herramienta desarrollada.

BIBLIOGRAFÍA

- Abello, J. y J. Korn (2002). «Mgv: A system to visualize massive multi-digraphs». En: Citeseer.
- Aigner, Wolfgang (2011). *Visualization of time-oriented data*. Springer. ISBN: 0857290797.
- Andrews, D. F. (1972). «Plots of high dimensional data». En: *Biometric* 28.1, págs. 125-137. URL: <http://www.jstor.org/stable/2528964>.
- Andrews, Keith (2005). *Information Visualisation*. URL: <http://courses.iicm.edu/ivis/>.
- Andrienko, Gennady y col. (2010). «Space-in-Time and Time-in-Space Self-Organizing Maps for Exploring Spatiotemporal Patterns». En: *Computer Graphics Forum*. Vol. 29. 3. Wiley Online Library, págs. 913-922.
- Ankerst, M. y col. (1996). «'Circle Segments': A Technique for Visually Exploring Large Multidimensional DataSet». En: *Visualization*. San Francisco.
- Atkinson, Larry y Steve Hankin (1995). «Common Data Formats». En: *Oceanography-Washington DC-Oceanography Society*- 8, págs. 100-101. ISSN: 1042-8275.
- Beddow, J (1990). «Shape coding of multidimensional data on a microcomputer display». En: *1990 IEEE conference on visualization*. Los Alamitos, CA.: IEEE Computer Society Press, págs. 238-246.
- Chen, Chun-houh, Wolfgang Härdle y Antony Unwin, eds. (2008). *Handbook of Data Visualization*. Berlin Heidelberg. Berlin: Springer-Verlag.
- Chernoff, H. (1973). «The use of faces to represent points in k-dimensional space graphically». En: *J Am Stat Assoc* 68, págs. 361-368.

- Chrisman, Nicholas R y col. (1989). «Geographic information systems». En: *Geography in America*, págs. 776-790.
- Cleveland, W.S (1993). «Visualizing Data». En: *Hobart Press*. Summit New Jersey.
- Cui, Q, MO Ward y EA Rundensteiner (2006). «Enhancing scatterplot matrices for data with ordering or spatial attributes». En: vol. 6060. Citeseer, págs. 248-258.
- Eick, SG (2000). «Visualizing multi-dimensional data». En: *ACM SIGGRAPH computer graphics* 34.1, págs. 61-67. ISSN: 0097-8930.
- Furlanello, Cesare y col. (2003). «GIS and the random forest predictor: Integration in R for tick-borne disease risk assessment». En: *Proceedings of DSC*. Citeseer, pág. 2.
- Gahegan, Mark y col. (2002). «Introducing GeoVISTA Studio: an integrated suite of visualization and computational methods for exploration and knowledge construction in geography.» En: *Computers, Environment and Urban Systems* 26(4).267-292.
- Group, HDF y col. (2011). *HDF5 user guide*.
- Guo, Diansheng y col. (2006). «A visualization system for space-time and multivariate patterns (vis-stamp)». En: *Visualization and Computer Graphics, IEEE Transactions on* 12.6, págs. 1461-1474.
- Hanisch, Robert J y col. (2001). «Definition of the flexible image transport system (FITS)». En: *Astronomy and Astrophysics* 376.1, págs. 359-380. ISSN: 0004-6361.
- Hansen, Charles D. y Chris R. Johnson (2005). *The visualization handbook*. Elsevier. ISBN: 0-12-387582-X.
- Harris, I y col. (2013). «Updated high-resolution grids of monthly climatic observations—the CRU TS3. 10 Dataset». En: *International Journal of Climatology*.
- Hearnshaw, Hilary M. y David J. Unwin (1994). *Visualization in geographical information systems*. Chichester.
- Hearst, M.A. (1995). «TileBars: visualization of term distribution information in full text information access». En: *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM Press/Addison-Wesley Publishing Co., págs. 59-66. ISBN: 0201847051.

- Inselberg, A. y B. Dimsdale (1990). «Parallel coordinates: A tool for visualizing multi-dimensional geometry». En: *Visualization 90*. San Francisco, págs. 361-370.
- Jin, Hai y Diansheng Guo (2009). «Understanding climate change patterns with multivariate geovisualization». En: *Data Mining Workshops, 2009. ICDMW'09. IEEE International Conference on*. IEEE, págs. 217-222. ISBN: 1424453844.
- John, Mathias, Christian Tominski y Heidrun Schumann (2008). «Visual and analytical extensions for the table lens». En: *Electronic Imaging 2008*. International Society for Optics y Photonics, págs. 680907-680907.
- Kandogan, Eser (2000). «Star coordinates: A multi-dimensional visualization technique with uniform treatment of dimensions». En: *Proceedings of the IEEE Information Visualization Symposium*. Vol. 650. Citeseer.
- Kaski, Samuel, Janne Nikkilä y Teuvo Kohonen (1998). «Methods for interpreting a self-organized map in data analysis». En: *In Proc. 6th European Symposium on Artificial Neural Networks (ESANN98). D-Facto, Brugfes*. Citeseer.
- Keim, DA (2002a). «Designing pixel-oriented visualization techniques: Theory and applications». En: *IEEE Transactions on Visualization and Computer Graphics* 6.1, págs. 59-78. ISSN: 1077-2626.
- (2002b). «Information visualization and visual data mining». En: *IEEE transactions on Visualization and Computer Graphics* 8.1, págs. 1-8. ISSN: 1077-2626.
- Keim, D.A. y H.P. Kriegel (1994). «VisDB: Database exploration using multidimensional visualization». En: *Computer Graphics and Applications, IEEE* 14.5, págs. 40-49. ISSN: 0272-1716.
- Keim, Daniel A. (2000). «Designing Pixel-Oriented Visualization Techniques: Theory and Applications». En: *IEEE Transactions on Visualization and Computer Graphics* 6.
- Keim, Daniel A, Jörn Kohlhammer y col. (2010). *Mastering The Information Age-Solving Problems with Visual Analytics*. Florian Mansmann. ISBN: 3905673770.
- Keim, Daniel A., Hans-Peter Kriegel y Michael Ankerst (1995). «Recursive Pattern: A Technique for Visualizing Very Large Amounts of Data». En: *Visualization '95*. Atlanta, GA, págs. 279-286.

- Keim, Daniel A, Jörn Schneidewind y Mike Sips (2004). «CircleView: a new approach for visualizing time-related multidimensional data sets». En: *Proceedings of the working conference on Advanced visual interfaces*. ACM, págs. 179-182.
- Kohonen, Teuvo (1982). «Self-organized formation of topologically correct feature maps». En: *Biological cybernetics* 43.1, págs. 59-69. ISSN: 0340-1200.
- (1990). «The self-organizing map». En: *Proceedings of the IEEE* 78.9, págs. 1464-1480. ISSN: 0018-9219.
- Levkowitz, H. (1991). «Color icons-merging color and texture perception for integrated visualization of multiple parameters». En: *Visualization, 1991. Visualization'91, Proceedings., IEEE Conference on*. IEEE, págs. 164-170, 420. ISBN: 0818622458.
- MacEachren, Alan y col. (2003). «Exploring High-D Spaces with Multiform Matrices and Small Multiples». En: *Proceedings of the International Symposium on Information Visualization*. Seattle, WA, págs. 31-38.
- Mazza, Riccardo (2009). *Introduction to Information Visualization*. 1.^a ed. Springer Publishing Company, Incorporated. ISBN: 1848002181, 9781848002180.
- Neteler, Markus y col. (2012). «GRASS GIS: A multi-purpose open source GIS». En: *Environmental Modelling & Software* 31, págs. 124-130.
- North, C. y B Shneiderman (2000). «Snap-together visualization: can users construct and operate coordinated visualizations?» En: *International Journal of Human-Computer Studies* 53.5, págs. 715-739.
- Pickett, Ronald M y Georges G Grinstein (1988). «Iconographic displays for visualizing multidimensional data». En: *Proc. IEEE Conf. on Systems, Man and Cybernetics, IEEE Press, Piscataway, NJ*. Vol. 514, pág. 519.
- Pirolli, Peter y Ramana Rao (1996). «Table lens as a tool for making sense of data». En: *Proceedings of the workshop on Advanced visual interfaces*. ACM, págs. 67-80.
- Rao, Ramana y Stuart K. Card (1994). «The Table Lens: Merging Graphical and Symbolic Representations in an Interactive Focus+Context Visualization for Tabular Information». En: *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*.

- Rew, Russ y Glenn Davis (1990). «NetCDF: an interface for scientific data access». En: *Computer Graphics and Applications, IEEE* 10.4, págs. 76-82. ISSN: 0272-1716.
- Rew, Russ, Glenn Davis y col. (2011). «The NetCDF users guide, data model, programming interfaces, and format for self-describing, portable data». En: *NetCDF Version 4.1*.
- Rhyne, TM (1997). «Going virtual with geographic information and scientific visualization». En: *Computers & Geosciences* 23.4, págs. 489-491. ISSN: 0098-3004.
- Rhyne, T.M. (1997). «Going virtual with geographic information and scientific visualization». En: *Computers & Geosciences* 23.4, págs. 489-491. ISSN: 0098-3004.
- Space Physics Data Facility (2013). *CDF User's Guide*. NASA/Goddard Space Flight Center.
- Stankovic, Jan, Markus Neteler y Roberto Flor (2002). «Experimental mobile wireless GRASS based GIS for handheld computers running GNU/Linux». En: *Proceedings, Open Source GIS GRASS Users Conference. Trento, Italy*, págs. 11-13.
- Streit, Marc y col. (2006). «3D parallel coordinate systems—A new data visualization method in the context of microscopy-based multicolor tissue cytometry». En: *Cytometry Part A* 69.7, págs. 601-611.
- Symanzik, Jürgen y col. (2000). «Linking ArcView™ and XGobi: Insight behind the Front End». En: *Journal of Computational and Graphical Statistics* 9.3, págs. 470-490. ISSN: 1061-8600.
- Takatsuka, Masahiro y Mark Gahegan (2002). «GeoVISTA Studio: a codeless visual programming environment for geoscientific data analysis and visualization». En: *Computers & Geosciences* 28(10), págs. 1131-1144.
- Theisel, H. (2000). «Scientific Visualization». En: URL: http://www.aimatshape.net/s05_sciviz.zip.
- Ward, Matthew O. (2002). «A Taxonomy of Glyph Placement Strategies for Multidimensional Data Visualization». Tesis doct. Worcester Polytechnic Institute.
- Ward, M.O. (1994). «Xmdvtool: Integrating multiple methods for visualizing multivariate data». En: *Proceedings of the Conference on Visualization'94*. IEEE Computer Society Press, págs. 326-333. ISBN: 0780325214.

- Weber, Marc, Marc Alexa y Wolfgang Müller (2001). «Visualizing time-series on spirals». En: *Information Visualization, IEEE Symposium on*. IEEE Computer Society, págs. 7-7.
- Wong, Pak Chung y R Daniel Bergeron (1994). «30 Years of Multidimensional Multivariate Visualization.» En: *Scientific Visualization*, págs. 3-33.
- Xie, Z y col. (2006). «Exploratory visualization of multivariate data with variable quality». En: IEEE, págs. 183-190. ISBN: 1424405912.
- Zhao, Shichen y col. (2010). «GDAL-based extend ArcGIS Engine's support for HDF file format». En: *Geoinformatics, 2010 18th International Conference on*. IEEE, págs. 1-3. ISBN: 1424473012.

Colofón

Este informe fue preparado en $\text{\LaTeX}2_{\epsilon}$ con el editor GNU Emacs. La composición tipográfica fue realizada con el uso de \XeTeX . Los tipos de letra empleados son *TeXGyre Pagella*, basada en el diseño original de Hermann Zapf (*Palatino*) para el texto principal; *Linux Biolinum O* de Philipp H. Poll para los títulos y *Latin Modern Mono* basado en el diseño original (*Computer Modern*) de Donald Knuth para el código computacional. La lista de referencias bibliográficas fue generada por biber/biblatex con el gestor JabRef; los índices por *MakeIndex*. Las ilustraciones fueron preparadas con METAPOST y PGF/TikZ.