

UNIVERSIDAD CENTRAL Y UNIVERSIDAD JORGE
TADEO LOZANO



Caracterización y clasificación de patrones atmosféricos provenientes de simulaciones climáticas

TESIS

PARA OBTENER EL TÍTULO DE

MAGISTER EN MODELADO Y SIMULACIÓN

PRESENTA

JUAN FELIPE MOLINA ROJAS

DIRECTOR: HUGO FRANCO TRIANA, PhD

BOGOTÁ, D.C.

2015

AUTOR

FECHA

FIRMA

DEDICATORIA

Dedico este trabajo a las personas que estuvieron apoyándome

Agradecimientos

Quisiera agradecer al proyecto número 1225-569-34920 “*Diseño e implementación de un sistema de cómputo sobre recursos heterogéneos para la identificación de estructuras atmosféricas en predicción climatológica*” por la financiación dada por el departamento de ciencias, tecnología e innovación de Colombia (Colciencias), también agradezco al grupo COMMONS por el apoyo en la ejecución del trabajo, agradezco a ACOLMET por el apoyo en la última etapa del proyecto y agradezco al grupo MINDLAB por el apoyo en las experimentaciones y en la discusión de los descriptores. Igualmente, quisiera extender este agradecimiento al prof. José Daniel Pabón y el ing. Guillermo Armenta por compartir los resultados de sus trabajos en simulación del clima como insumo para la ejecución de este trabajo.

Agradezco a mi familia y amigos por el apoyo que se me presentó en el desarrollo de este proyecto a lo largo de la maestría.

Abstract

The atmosphere is a highly complex fluid system where multiple intrinsic and extrinsic phenomena superpose at the same spatial and temporal domains for different scales, making its characterization a challenging task. Despite the novel methods for pattern recognition and detection available present in the literature, most of climate data analysis and weather forecast processes rely on the ability of specialized personnel to visually detect atmospheric patterns from plots of climatic and meteorological data. This thesis presents a method for characterizing and classifying low-level wind flow configurations, namely: confluences, diffluences, vortices and saddle points. This method combines specialized image features to capture the particular structure of low-level wind flow configurations through a phase portrait characterization and a state-of-the-art machine learning classification method. The method was validated on a set of volumes extracted from climate simulations and manually annotated by experts. With this approach, the best result obtained within the independent test dataset was 0.89 of average accuracy among the four atmospheric structures.

Índice general

Índice de figuras	xI
Índice de cuadros	xv
1. Introducción	1
1.1. Caracterización del clima	4
1.1.1. Variables físicas de interés climático	4
1.1.2. Estructuras y patrones atmosféricos	7
1.2. Análisis de datos climáticos	12
1.2.1. Modelos del clima	13
1.2.2. Métodos analíticos y estadísticos	14
1.2.3. Análisis de datos climáticos mediante técnicas de apren- dizaje de máquina	15
1.2.4. Caracterización y detección de estructuras fluidas . .	15
1.3. Propuesta y estructura de la tesis	16
2. Objetivos	19
2.1. Objetivo general:	19
2.2. Objetivos específicos:	19
3. Materiales y métodos	21
3.1. Construcción de un conjunto de datos a partir de simulacio- nes WRF	21
3.1.1. Corrección de la posición de las componentes de la velocidad del viento	22
3.1.2. Herramienta y protocolo de anotación	23

3.2.	Caracterización de patrones atmosféricos	24
3.2.1.	Aproximación de Operadores Diferenciales sobre dominios discretos	24
3.2.2.	Descriptores basados en histogramas	27
3.2.3.	Descriptor basado en retratos de fase	30
3.3.	Clasificación y evaluación del desempeño de los descriptores	32
3.3.1.	SVM	32
3.3.2.	Random Forest	35
3.3.3.	K-Fold Stratified Cross-Validation	35
3.3.4.	Evaluación mediante matrices de confusión	36
4.	Resultados	39
4.1.	Herramienta de anotaciones de patrones atmosféricos	39
4.2.	Descripción del conjunto de datos con anotaciones	39
4.3.	Desempeño del proceso de clasificación	41
4.3.1.	Descriptor basado en histogramas	41
4.3.2.	Descriptor basado en retratos de fase	47
5.	Discusión y Conclusiones	51
	Bibliografía	57
A.	Proceso de extracción de datos	65
A.1.	Extracción de Variables	65
A.2.	Creación de Metadatos	70
B.	Documentación de la herramienta de anotaciones	79
B.1.	Casos de uso	79
B.2.	Diagrama de clases	80
B.2.1.	Diagrama de clases: Modelo del negocio	80
B.2.2.	Diagrama de clases: GUI	81
B.3.	Diagrama de actividades	82
B.4.	Diagrama de secuencias	83
B.5.	Formato del archivo de entrada (XML)	84
B.6.	Formato del archivo de salida (XML)	94

B.7. Repositorio en git	96
B.8. Manual de usuario	96
C. Experimentación propuesta	97
C.1. Extracción de ROIs	97
C.2. Experimentación	110
C.3. Validación	119
D. Publicaciones	127
D.1. Artículo publicado	127

Índice de figuras

1.1. Fenómenos atmosféricos en escala temporal y espacial. Adaptado de <i>Introducción a la meteorología</i> por Arlene Laing y Jenni-Louise Evans [25]. Copyright 2011 por The COMET Program, UCAR. Adaptado con permiso.	3
1.2. Vorticidad	7
1.3. Difluencia	8
1.4. Confluencia	8
1.5. Puntos de silla	9
3.1. Esquema de interpolación aplicada a las componentes U , V y W para obtener una visualización del campo vectorial, se observa a la izquierda una representación de la componente de vientos presente en el modelo WRF, en la parte central como es su representación de forma discreta y en la ultima el ajuste que se le hace a cada una de las componentes de vientos para que queden centradas	23
3.2. Visualización de los datos correspondiente a la variable de temperatura potencial y la magnitud de las componentes de vientos de forma vectorial	24
3.3. Caso representativos de las estructuras atmosféricas bajo estudio Confluencias, Difluencias, Vorticidades y puntos de silla	26
3.4. Los histogramas se extraen en un diseño piramidal de los diferentes mapas de características. Los 14 histogramas resultantes se integran adicionalmente en el descriptor max-pooling, que corresponde a un histograma con los valores máximos para todas las características.	28

3.5.	Esquema del Histograma orientada de flujo óptico (HOOF) gráfico tomado del paper de Chaudhry et al. Adaptado con permiso [8]	29
3.6.	Representación de la región de interés, junto con sus componentes U y V y sus respectivas derivadas parciales para las componentes correspondientes $(x$ y $y)$; se puede observar que cada derivada parcial representa de manera más pronunciada los cambios en los campos U y V que la distribución de valores de los propios campos.	31
3.7.	Esquema de SVM usando un kernel Lineal para 2 clases . .	34
3.8.	Esquema de <i>Random forest</i> para clasificación	36
3.9.	Esquema de Validación cruzada	37
3.10.	Esquema de matriz de confusión	37
4.1.	Herramienta de anotaciones visualizando temperatura y vientos	40
4.2.	Matrices de Confusión de evaluados en una máquina de aprendizaje SVM - RBF donde (a) es la concatenación de los histogramas extraídos de los operadores diferenciales divergencia y rotacional, (b) es el maxpooling de los 14 Histogramas extraídos con el esquema piramidal, (c) es la concatenación de los 14 histogramas de cada uno de los operadores diferenciales y (d) es el esquema piramidal usando HOOF como característica	45
4.3.	Matrices de Confusión de evaluados en una máquina de aprendizaje SVM Lineal donde (a) es la concatenación de los histogramas extraídos de los operadores diferenciales divergencia y rotacional, (b) es el <i>max-pooling</i> de los 14 Histogramas extraídos con el esquema piramidal y (c) es la concatenación de los 14 histogramas de cada uno de los operadores diferenciales	46

4.4.	Matrices de Confusión de evaluados en una máquina de aprendizaje Random Forest donde (a) es la concatenación de los histogramas extraídos de los operadores diferenciales divergencia y rotacional, (b) es el maxpooling de los 14 Histogramas extraídos con el esquema piramidal, (c) es la concatenación de los 14 histogramas de cada uno de los operadores diferenciales y (d) es el esquema piramidal usando HOOF como característica	47
4.5.	Resultados de la Matriz de confusión de los descriptores basados en retratos de fase	48
5.1.	Comparación entre el modelo analítico y el gradiente	53
5.2.	Configuración de frentes fríos y ondas del este. La imagen superior es una configuración para detectar frentes fríos donde se observan las variables de humedad específica, temperatura en superficie, velocidad de vientos en superficie y presión a nivel del mar. En la imagen inferior es una configuración para detectar ondas del este donde se observan los operadores diferenciales divergencia y rotacional a 850mb y 700mb. . .	54
B.1.	Diagrama de Casos de Uso	79
B.2.	Diagrama de Clases del Modelo del negocio	80
B.3.	Diagrama de clases de la GUI	81
B.4.	Diagrama de actividades	82
B.5.	Diagrama de secuencias	83

Índice de cuadros

- 3.1. Tamaño de las matrices 3D correspondientes a los valores para las variables extraídas en el proceso de ETL para cada instante de tiempo. La componente horizontal de la velocidad del viento (U), tiene una longitud adicional y la componente vertical (V) una latitud de más, debido a la representación particular de los componentes de viento en los archivos netCDF producto de las simulaciones (en las caras de los cubos y no centradas en los elementos de volumen, ver sección 3.1.1 22

- 4.1. División del dataset anotado por clase y mes de simulación 41
- 4.2. Resultados del Desempeño de la clasificación en el conjunto de prueba en términos de acierto observando la precisión por clase y la precisión promedio 44
- 4.3. Resultado del descriptor de retratos de fase donde se muestra la precisión de cada una de las clases y el desempeño promedio del descriptor con la máquina de aprendizaje . . . 49

Capítulo 1

Introducción

El clima, como expresión de la dinámica resultante de la interacción entre la atmósfera, el océano y la superficie terrestre, condiciona el desarrollo de las diferentes actividades humanas, particularmente aquellas relacionadas con procesos de producción, intercambio y consumo de bienes y servicios, desarrollo urbanístico y rural, entre otras. Tal influencia determina de forma directa el bienestar y la seguridad de la población, haciendo que el estudio de la dinámica de la atmósfera, tanto en su caracterización como en el predicción de su comportamiento se convierta en una necesidad práctica a escala económica y social.

La atmósfera es la capa gaseosa que rodea la superficie terrestre; de acuerdo a la masa presente en una zona dada y un momento determinado, los gases que constituyen la atmósfera se pueden clasificar en dos grupos: constantes y variables. Los gases constantes se mantienen en una proporción prácticamente fija, siendo los más abundantes el nitrógeno (78 %), el oxígeno (20.9 %) y el argón (0.9 %). Los gases variables tienen una presencia variable en la composición atmosférica; los más representativos en esta categoría son el vapor de agua y el dióxido de carbono, cuya masa presenta grandes oscilaciones a diferentes escalas temporales (geológicas, estacionales, horarias, etc.) [12].

El Tiempo es la manifestación concreta y de corto plazo de la dinámica atmosférica en un lugar e instante determinados. Los fenómenos asociados al transporte de momento, calor y masa asociados a las diferentes especies que componen la atmósfera (especialmente el vapor de agua) determinan

el comportamiento dinámico de las diferentes variables que describen el estado del sistema (temperatura, presión, humedad relativa, condensación y precipitación, velocidad del viento, etc.) cuyas variaciones se hacen perceptibles a través de los fenómenos atmosféricos (sensación térmica, humedad, nubosidad, condensación y precipitaciones, tormentas, granizo, nieve, vientos en sus diferentes manifestaciones, etc.) [12].

El Clima es el conjunto fluctuante de condiciones atmosféricas observable a escala global y grandes escalas de tiempo, determinadas éstas por fenómenos astronómicos, geológicos y, más recientemente, antropogénicos. El clima se describe mediante registros indirectos de variables como las temperatura media, máxima y mínima de determinados periodos, la humedad, precipitaciones, composición atmosférica, entre otras [29].

Los fenómenos atmosféricos se pueden clasificar de acuerdo a sus escalas espaciales y temporales, lo que permite un análisis adecuado del comportamiento de las distintas variables que describen la dinámica del clima. En Climatología y Meteorología, tales escalas son:

- **Escala planetaria:** más de 3.000 km. En ella se encuentran las ondas superiores de la troposfera y la zona de confluencia intertropical, entre otras 1.1.
- **Gran Escala o Sinóptica:** de 100 km a 3.000 km. A esta escala pertenecen los frentes fríos y cálidos, los ciclones de mayor tamaño, los conglomerados de nubes, la oscilación de “El Niño” (ENSO por sus siglas en inglés), etc.
- **Mesoescala:** de 3 km a 100 km. A ésta corresponden las tormentas marinas, las brisas de montaña, entre otros fenómenos.
- **Topoescala o Escala Local:** de 100 m a 3 km. Por ejemplo: Contaminación atmosférica, tornados
- **Microescala:** menos de 100 m. La agrometeorología y el estudio de la evaporación se enmarcan en esta escala.

Las escalas temporales, por su parte, suelen definirse dependiendo de cada fenómeno bajo estudio, ya que la duración del mismo depende de su escala espacial. A escalas cortas de tiempo, el estudio de los fenómenos atmosféricos y los eventos resultantes de su interacción se conoce como *Meteorología*,

siendo el pronóstico a corto plazo una de sus tareas más relevantes. El análisis de fenómenos a grandes escalas de tiempo corresponde a la *Climatología* y requiere la utilización de datos históricos (30 años o más) [33].

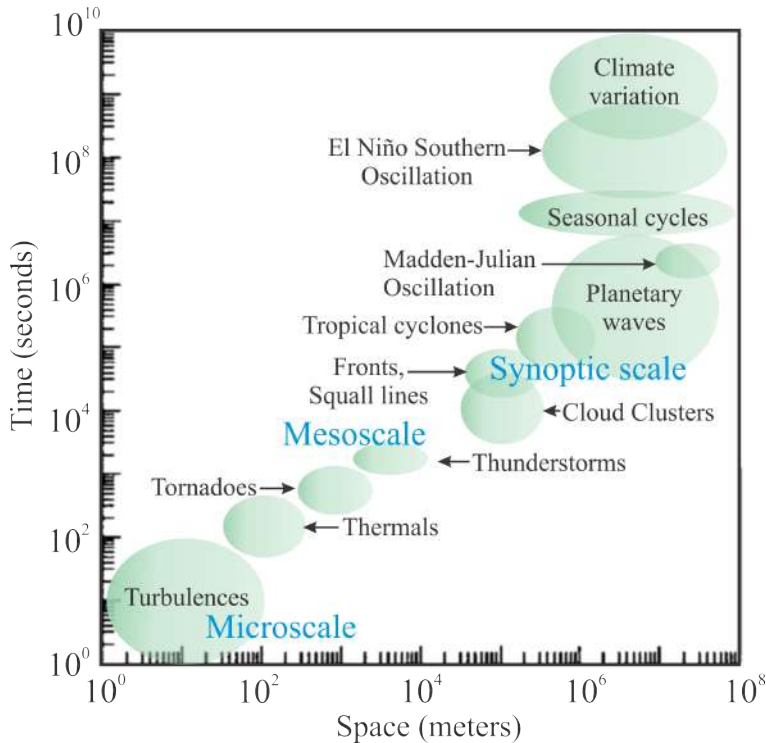


Figura 1.1: Fenómenos atmosféricos en escala temporal y espacial. Adaptado de *Introducción a la meteorología* por Arlene Laing y Jenni-Louise Evans [25]. Copyright 2011 por The COMET Program, UCAR. Adaptado con permiso.

La naturaleza de los fenómenos atmosféricos también varía con la ubicación geográfica, dadas las diferentes características térmicas observables a diferentes latitudes sobre la superficie terrestre. De acuerdo con la latitud, el planeta se divide en tres zonas geográficas:

- **Zona Tórrida:** es la franja localizada entre los 23° de latitud norte y los 23° de latitud sur. Esta zona presenta una temperatura media

mensual de 18°C , debido a la menor variación del ángulo de incidencia de la radiación solar respecto a otras zonas, cercano a 90° la mayor parte del año.

- **Zona Templada:** son las regiones localizadas entre 23° a 66° de latitud Norte y 23° a 66° latitud sur. Esta zona presenta estaciones definidas (veranos tórridos e inviernos fríos) debido a la alta variación del ángulo de incidencia de la radiación solar inclinación de la tierra es de 90° haciendo que los lugares que están en esta zona presenten un periodo caluroso y un periodo frío en distintos periodos que son inverso para ambos hemisferios.
- **Zona Polar:** corresponden a los extremos norte y sur del planeta. Se encuentran entre localizados a partir de los 66° de latitud norte y 66° de latitud sur. Estas regiones presentan bajas temperaturas y un periodos de día y noche notoriamente más largos debido a la inclinación de la Tierra.

1.1. Caracterización del clima

Para analizar el comportamiento del clima, científicos de diferentes disciplinas se valen de modelos matemáticos [40] que permiten dar explicación a cada uno de los fenómenos observables. Los expertos analizan diferentes variables escalares (temperatura, presión, humedad, precipitación) y vectoriales (vientos) a diferentes alturas o niveles de presión con el fin de determinar las variables más relevantes para cada una de las estructuras atmosféricas observadas.

1.1.1. Variables físicas de interés climático

Las variables anteriormente mencionadas permiten describir el comportamiento del clima en un instante de tiempo determinado en función de su comportamiento termodinámico [48]. Dentro de las variables de mayor relevancia en contextos aplicados y de investigación se encuentran las siguientes:

- *Temperatura atmosférica:* en meteorología, la temperatura es una magnitud relacionada con el calor específico en una región de la

atmósfera en un instante de tiempo y en un punto determinado que es generado por el movimiento de las partículas [21]. Esta variable se analiza en valores potenciales medidos en grados Kelvin (K°). En modelos de simulación es usual usar la perturbación de temperatura (T) relacionada con la temperatura total (T_t) con el siguiente modelo:

$$T_t = (T + 300)((PB + P)/100,000)^{2/7}$$

donde P es la perturbación de presión y P_B es la presión base.

- *Presión atmosférica*: en meteorología, es una magnitud física que se define como la fuerza ejercida por la gravedad sobre la capa aire presente en la atmósfera en un punto de la superficie, esta variable se ve afectada por la temperatura, humedad, la ubicación geográfica entre otras [21]. Esta variable se analiza en valores potenciales medidos en Pascales (Pa). En modelos de simulación es usual usar la perturbación de presión (P) y la presión base (P_B) relacionada con la presión total (P_t) como se muestra a continuación:

$$P_t = P + P_B$$

- *Humedad del aire*: es una magnitud física que determina la cantidad de agua presente en la atmósfera esta medida permite describir diferentes fenómenos atmosféricos como es caso de la nubosidad, precipitaciones, balances de radiación, entre otros. La humedad puede ser medida como humedad absoluta, humedad relativa y humedad específica [21].
 - *Humedad absoluta* (H_A): es la masa total de agua que se encuentra en la atmósfera, se mide por unidad de volumen, se expresa en gramos (g) sobre metros cúbicos (m^3).

$$H_A = \frac{m_w}{v_a}$$

donde m_w es la masa de vapor de agua y v_a es el volumen de aire presente.

- *Humedad relativa* (H_R): es la relación porcentual entre la cantidad de vapor de agua real que existe en la atmósfera y la máxima que contendría a la misma temperatura

$$H_R = \frac{p_{(H_2O)}}{p_{(H_2O)}^*} 100 \%$$

donde $p_{(H_2O)}$ es la presión parcial de vapor de agua que se encuentra en el aire medido en (Pa) y $p_{(H_2O)}^*$ es la presión de saturación de agua a la temperatura de la mezcla de aire medido en (Pa)

- *Humedad específica* (q): mide la masa de agua que se encuentra en estado gaseoso en un kilogramo de aire, se expresa como ($\frac{g}{Kg}$)

$$q = \frac{p_v}{p_v + p_d}$$

donde p_v representa la cantidad de vapor de agua, $p_v + p_d$ es la cantidad de agua contenida en la masa de aire húmedo.

- *Velocidad del vientos*: es una magnitud vectorial que representa el movimiento horizontal y vertical de masas de aire y otros componentes que se encuentran en la atmósfera. Los vientos se presentan por las diferencias de temperatura existentes en la atmósfera, las masas de aire caliente tienden a ascender mientras las masas de aire frío ocupan su lugar. La dirección de los vientos depende de la distribución y evolución de los centros isobáricos; este desplazamiento se presenta de altas a bajas presiones, su fuerza es directamente proporcional al gradiente de presión. El viento se define en términos de dirección en el plano horizontal y magnitud de velocidad. La dirección es dada por el ángulo en el cual se presente, y la velocidad es dada en nudos o en ($\frac{m}{s}$). Los vientos también son representados por sus componentes vectoriales U , V y W que representan los vientos longitudinales, latitudinales y su componente vertical y se definen a partir de:

$$\vec{v} = (U, V, W) = \left(\frac{\partial x_i}{\partial t}, \frac{\partial y_i}{\partial t}, \frac{\partial z_i}{\partial t} \right)$$

1.1.2. Estructuras y patrones atmosféricos

La atmósfera es un sistema fluido que presenta comportamientos altamente complejos, resultantes de la superposición de diferentes fenómenos a distintas escalas espaciales y temporales. Producto de estas interacciones, las variables que representan el estado del atmósfera adquieren patrones espaciales que pueden ser descritos mediante su distribución geométrica y características físicas, y cuya presencia determina la ocurrencia de diferentes tipos de eventos y fenómenos atmosféricos. Estos patrones podrían clasificarse, a su vez, de acuerdo a la complejidad de su configuración geométrica en *estructuras de bajo nivel* y *estructuras de alto nivel*. Las *estructuras de bajo nivel* se pueden definir como configuraciones espaciales de los valores de las variables del sistema (vientos, temperatura, presión, humedad, etc.) con una distribución geométrica bien definida. Para modelar tales configuraciones, de manera que pueda establecerse una clasificación, se pueden usar operadores diferenciales como gradientes, divergencias, rotacionales, derivadas materiales y operadores convectivos, entre otros [43, 10].

En efecto, dichos patrones suelen ser identificables mediante inspección visual a través del cálculo y visualización de isoclinas, líneas de corriente y distribuciones de vectores sobre los campos escalares y vectoriales asociados a las variables mencionadas. Es usando este tipo de representaciones que los especialistas (meteorólogos, climatólogos, geólogos, etc.). Dentro de las estructuras de bajo nivel de uso más común en tareas aplicadas y de investigación sobre el comportamiento de la atmósfera se encuentran:

- *Vorticidad*: es un patrón que se observa en regiones en el que el movimiento de un fluido se presenta alrededor de un punto fijo, por ende tiene un valor de rotacional alto, como se observa en la figura 1.2.

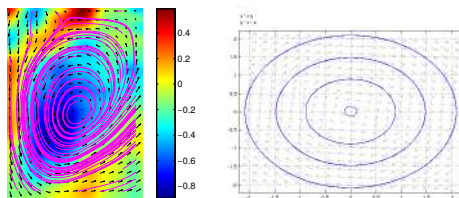


Figura 1.2: Vorticidad

- *Difluencia*: es un patrón que se observa en regiones en el que el movimiento de un fluido se presenta sale de un punto fijo como si fuera una fuente. por ende tiene valores de divergencia positivos, como se observa en la figura 1.3.

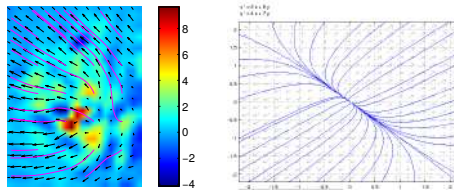


Figura 1.3: Difluencia

- *Confluencia*: es un patrón que se observa en regiones en el que el movimiento de un fluido se presenta sale de un punto fijo como si fuera un sumidero. por ende tiene valores de divergencia negativos, como se observa en la figura 1.4.

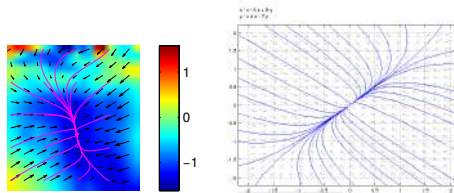


Figura 1.4: Confluencia

- *Puntos de Silla*: en un punto de silla, la distribución de las velocidades de un fluido es tal que se encuentran valores cercanos a cero en su punto fijo y valores mayores divergiendo del mismo, como se observa en la figura 1.5.

De otro lado, las estructuras de alto nivel corresponden a la ocurrencia conjunta e interrelacionada de dos o más estructuras de bajo nivel en un momento y lugar específicos. Este tipo de estructuras suele presentarse a escalas espaciales mayores y es el más usado en aplicaciones relacionadas

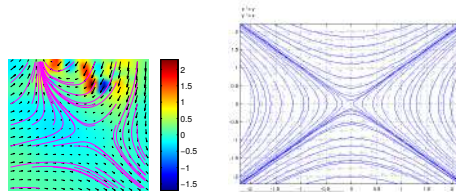


Figura 1.5: Puntos de silla

con el pronóstico del tiempo y la caracterización del clima. Algunas de las estructuras de alto nivel más relevantes son:

- *Zona de convergencia intertropical (ZCIT)*: es una franja de bajas presiones en la Zona Tórrida producto de la confluencia de vientos alisios procedentes de las Zonas Templadas del Norte y del Sur. La localización de la ZCIT oscila a lo largo del año, desplazándose de sur a norte entre los meses de enero y febrero y de norte a sur entre los meses julio y agosto. El paso de la ZCIT sobre una región determinada produce un incremento sensible en las precipitaciones, dando lugar a la sucesión de temporadas lluviosas y secas en regiones tropicales [41].
- *Ondas del Este*: las ondas del este son un fenómeno que se presenta en las cuencas del océano Atlántico proveniente de las corrientes que se presentan en noroeste de África que se propagan hacia el oeste. Es una de las principales causas de los ciclones tropicales en la cuenca del centro y el norte del Atlántico. Estas ondas se presentan como perturbación de tipo ondulatorio que se presenta como curvaturas ciclónicas que deforman los campos de presión [27].
- *Frentes Fríos*: un frente se considera como una discontinuidad en el campo de presión, temperatura y humedad que se da por la presencia de dos masas de aire, que se ven influenciadas por el comportamiento del viento empujando una masa sobre la otra, en el caso de frentes fríos es el desplazamiento de la masa de aire fría, se generan en latitudes medias en el hemisferio norte y sur, los cuales por su extensión inciden en la dinámica de la nubosidad en la región de Amazonia y en la región Caribe. Si se generan en el sur activan la celda amazónica

acentuando las lluvias hacia el trapecio y sur del país. Si se generan en el océano Atlántico arrastran humedad hacia el norte del país provocando lluvias en la región Caribe [35].

- *Sistemas de Alta Presión* son sistemas que se forman debido al flujo del aire que va de áreas de alta presión a áreas de baja presión tratando de encontrar un punto de equilibrio, aunque existe una fuerza causada por la rotación de la tierra que desvía este flujo haciendo que no vaya en línea recta. Una de las características es el patrón de los vientos en forma de espiral. Para Colombia estos fenómenos condicionan el comportamiento de la zona de confluencia intertropical, las Vaguadas extratropicales y las ondas del este haciendo que estas estructuras no sigan su curso. Estos sistemas son semipermanentes porque cambian su posición por cambios bruscos en la temperatura [18].
- *Vaguadas*: Se define como vaguada a un punto de baja presión que se presenta en medio de dos puntos de alta presión. Las vaguadas se produce cuando las masas de aire húmedo y cálido ascienden a través de este punto de baja presión, lo que provoca el desarrollo de nubes y precipitaciones. En Suramérica se presentan dos vaguadas que afectan el clima que son *las Vaguadas de latitudes medias y la Vaguada tropical de la alta tropósfera – (TUTT)* [18]
 - *Vaguadas de latitudes medias*: son ondulaciones de flujo de aire frío en la altura provenientes del oeste que se forman en latitudes medias y altas, presentan un desplazamiento de ida y vuelta hacia los trópicos en circulación en V. Esta estructura se encarga de producir nubosidad y precipitaciones constantes. Para Colombia este fenómeno se presenta cuando una vaguada interactúa con la zona de confluencia intertropical o con las ondas del Caribe, ocasionando periodos prolongados de nubosidad y precipitaciones [18].
 - *Vaguada tropical de la alta tropósfera – (TUTT)*: Es una vaguada que se presenta en los niveles altos de la tropósfera, que separa la cuña subtropical de la ecuatorial en el Pacífico norte y/o en el Atlántico Norte cerca de Suramérica, esta vaguada

se traslada desde las costas del noroccidente de África hasta el extremo suroccidental del Caribe [18].

- *La Zona de Convergencia del Atlántico Sur – ZCAS:* Es un fenómeno que se produce cuando convergen las masas de aire tropicales y las masas de aire subtropicales, se forma generalmente cuando los frentes fríos se debilitan y se estacionan al sudeste de Brasil, estas estructuras cambian de posición en el verano haciendo que se localice sobre el norte y noreste del continente americano [18].
- *Baja anclada de Panamá:* Es un sistema de baja presión que se ubica sobre la parte centro y oeste del Mar Caribe, generando una disminución de la presión de hasta 1008 milibares hacia el este, oscila entre 1010 y 1012 milibares de presión [18].
- *Chorros de baja altura:* Son flujos de aire rápidos y limitados a zonas bien localizadas, presentes usualmente en tropopausa. Para considerar que una corriente corresponde a un chorro de baja altura, debe tener una velocidad mínima de 120 Km/h y forma *tubular*, achata-da y es casi horizontal. Su detección y medida son importantes en Meteorología como soporte al pronóstico; en aeronáutica, es esencial saber de su presencia para establecer tiempos de vuelo y garantizar la seguridad de los vuelos, debido a que, dependiendo de su intensidad, pueden generar turbulencias. En Colombia y regiones aledañas, los chorros de baja altura más relevantes son:
 - *Chorro del Chocó:* el chorro del Chocó es un *jet* del oeste de bajo nivel, los vientos de este *jet* están casi ausentes durante febrero–marzo, y el viento máximo núcleo está presente durante octubre–noviembre ($8,6m/s$) situado en torno a los 5° N. Estos vientos provenientes del Océano Pacífico son más frías y húmedas de los vientos alisios que vienen del Atlántico y el Caribe. El chorro del Choco provoca fuerte advección de humedad sobre el norte de América del Sur para octubre, mostrando una gran cantidad de entrada de humedad desde el Océano Pacífico hasta el continente [42].
 - *Chorro de la Guajira:* la corriente en Chorro de niveles bajos en el Caribe (*Low Level Jet-LLJ*) es un fenómeno meteorológico que

se deriva de la influencia de la circulación Subtropical en niveles altos, que es generado por el movimiento de los vientos alisios en superficie con velocidades de 20 y 30 nudos. Está localizada en la ZCIT. Cuando esta corriente se encuentra con contenidos altos de humedad en las capas bajas genera una formación de nubes y precipitaciones [36].

1.2. Análisis de datos climáticos

La evaluación de los modelos matemáticos formulados para describir la dinámica del clima implica el contraste de los resultados arrojados por dichos modelos contra medidas experimentales que caractericen las condiciones atmosféricas presentes en la región bajo estudio para un determinado intervalo de tiempo. Estos valores experimentales son registrados mediante instrumentos específicos de medida [12] ubicados en estaciones meteorológicas, globos meteorológicos y boyas oceánicas, entre otros.

Los valores recogidos por los instrumentos de medida permiten la construcción de un conjunto de datos que representa la dinámica de la atmósfera según la evolución de sus variables representativas. Además de los registros de las variables relevantes antes citadas, tecnologías basadas en la teledetección han permitido extender y complementar la información disponible para la caracterización de los fenómenos atmosféricos. Tal es el caso de las imágenes satelitales, usualmente de carácter multiespectral, que permiten medir con mayor resolución espacial los valores de variables relacionadas con la atmósfera y la superficie terrestre [32].

Sin embargo, no siempre es factible una medición completa y exhaustiva de los valores relevantes de las variables de interés en la descripción de fenómenos en Meteorología y Climatología para todo el dominio bajo estudio, debido a limitaciones físicas tanto como a costos económicos propios de los sistemas de adquisición de datos. Las limitaciones físicas de los sistemas de adquisición se presentan porque se trabaja en un dominio 2D despreciando la dinámica que se observa a distintos niveles de altura con lo cual se pierde información que puede ser relevante. Por esta razón, un aspecto de creciente interés en el estudio del clima consiste en la elaboración y mejoramiento de modelos matemáticos para la simulación por computador de fenómenos climáticos y meteorológicos. Dichos modelos re-

producen el comportamiento la atmósfera en diferentes escenarios (reales o ideales) generando resultados que asignan valores a las variables de interés para todos los puntos el dominio, lo que equivale a todos los elementos de volumen en una aproximación discreta.

1.2.1. Modelos del clima

De acuerdo al tamaño del dominio bajo estudio y los fenómenos representados, los modelos para la descripción del clima se pueden clasificar en *modelos globales* y *modelos regionales*. Un modelo global permite analizar la dinámica atmosférica y oceánica en grandes escalas espaciales y temporales; este tipo de modelos es usado comúnmente en estudios de cambio climático y paleoclimatología, entre otros. Ejemplos de este tipo de modelos se encuentran en el trabajo de Tello et al. [55], que estudia la relación entre la temperatura superficial promedio y la temperatura al interior del océano, o en estudios sobre el monzón sudamericano [7], ambos basados en el modelo Ukmo–HadCM3 [15] (UK Model – Hadley Centre Coupled Model version 3).

A diferencia de los modelos globales, *los modelos regionales* permiten la ejecución de simulaciones en dominios espaciales correspondientes a zonas geográficas localizadas, con el objetivo de reproducir comportamientos atmosféricos observables a escalas más pequeñas. Dentro de los diferentes modelos regionales se encuentran:

- El Modelo Climático Regional (*RegCM*) del Instituto de Física Teórica de Trieste [38] que maneja una resolución espacial de 40×40 km, este modelo se ha usado para el análisis de los fenómenos de precipitación y de generación de nubes [39].
- El modelo PRECIS [23], que utiliza una resolución espacial mínima de 25×25 km [57], este modelo se ha usado para el estudio del cambio futuro de la temperatura y las precipitaciones extremas en América del Sur [30].
- El modelo Weather & Research Forecasting (WRF) [52] que maneja una resolución espacial mínima de 4×4 kilómetros como lo menciona [45, 1].

El modelo WRF ha sido utilizado recientemente para simular la variabilidad climática regional y el cambio climático, además de su impacto en diversos aspectos biogeofísicos y socioeconómicos. Este modelo es uno de los más extendidos por el amplio conjunto de *submodelos* provenientes orientados a la representación de los diferentes fenómenos e interacciones presentes en la atmósfera y en su relación con la superficie oceánica y continental.

Existe otro tipo de modelos climáticos, conocidos como *modelos de reanálisis*, que permiten combinar medidas reales de las variables climáticas (mediante estaciones meteorológicas e imágenes satelitales, entre otros) con resultados de simulaciones. Este tipo de modelo se caracteriza por incorporar a su vez un método para ajustar periódicamente las medidas físicas con los valores generados numéricamente. Los modelos de reanálisis son usados principalmente en procesos para el pronóstico y predicción. Un de los modelos de reanálisis de mayor uso es el *Climate Forecast System Reanalysis* (CFSR) [46], el cual ha sido empleado en estudios como el de Wang et al. [56], para caracterizar el comportamiento de la precipitación intraestacional en la Zona Tórrida. Este modelo cuenta con una resolución espacial mínima de 38×38 kilómetros.

1.2.2. Métodos analíticos y estadísticos

Para realizar un análisis analítico de los modelos propuestos se toma los diferentes valores obtenidos de manera experimental a través de los instrumentos específicos de medida, después de eso se tabulan estos valores y se realiza un proceso de regresión para entender el relación que puede tener los diferentes datos. en el caso del método estadístico se toman un periodo de tiempo mas largo que puede ser mensual, anual, decadal o centuria mediante periodogramas con el fin de determinar el comportamiento que se presente en ese periodo de tiempo, este tipo de estudio se hace para entender el comportamiento de los fenómenos de escala sinóptica y planetaria.

Para hacer este tipo de análisis se toma el promedio mensual de cada una de las variables, se analizan series temporales de cada una de las variables buscando un patrón que permita caracterizar el comportamiento del fenómeno. Un ejemplo de este se presenta en [47] que usa una transforma-

da de Wavelet para detectar la fenología en un cultivo usando imágenes MODIS, este análisis para calcular el índice de vegetación (*NDVI*) de formada dinámica analizando series temporales usando imágenes satelitales NOAA-AVHRR y MRA (Análisis de multiresolución basado en wavelets) [31].

1.2.3. Análisis de datos climáticos mediante técnicas de aprendizaje de máquina

La validez de los resultados obtenidos mediante la utilización de estos modelos matemáticos depende de la fiabilidad de los datos utilizados para su parametrización. Algunos trabajos [24, 14] proponen protocolos que ayuden a reducir la incertidumbre que presentan los resultados de las simulaciones basadas en este tipo de modelos, ya sea por la naturaleza no lineal del modelo o debido a la falta de rigor en la inicialización, en su relación con el escenario bajo estudio.

Algunos trabajos recientes abordan el estudio de la validez de los resultados en función de dicha parametrización empleando técnicas de aprendizaje de máquina, en el contexto de la predicción climática. El estudio de Monteleoni y Jaakkola [34] utiliza un algoritmo de aprendizaje (denominado α por los autores) consistente en una generalización de los modelos ocultos de Markov para el análisis de series de tiempo. A su vez, el trabajo de Lima et al. [28] abordó el problema de la predicción de los patrones asociados a la ocurrencia de eventos ENSO usando técnicas de reducción de la dimensionalidad (particularmente *Maximum Variance Unfolding* – MVU, como generalización del Análisis por Componentes Principales) para el análisis de sistemas no lineales. En el trabajo de Steinhäuser et al. [53] se usa una aproximación basada en redes complejas para la identificación de diferentes patrones atmosféricos.

1.2.4. Caracterización y detección de estructuras fluidas

La caracterización de estructuras en fluidos se ha abordado desde campos como la visión por computador y la Dinámica de Fluidos Computacional (CFD por sus siglas en inglés) [4]. Técnicas como el Flujo Óptico permiten medir el movimiento local de partículas fluidas en determinado tipo de datos, p.ej. vídeo digital [3]. Esta aproximación se utilizó en el

trabajo de Cheung et al. [9] para el análisis del clima a corto plazo utilizando un algoritmo de flujo óptico que combina medidas globales (Horn & Schunck) y locales (Lucas Kanade) como parte de la implementación del sistema SWIRL (Short-range Warning of Intense Rainstorms in Localized Systems) basado en imágenes de RADAR en la ciudad de Hong Kong. Igualmente, en el estudio de Nuwanthika et al. los vientos se han empleado medidas de flujo óptico combinadas con lógica difusa para la extracción de información relevante de datos reales [13].

Desde el punto de vista de la caracterización y análisis de la ocurrencia de determinados fenómenos atmosféricos, técnicas basadas en Wavelets (como en el trabajo de Amara [16]) permiten estudiar la variabilidad espacio-temporal del comportamiento atmosférico a diferentes escalas. Técnicas de este tipo se emplearon en el trabajo de Janicke et al. [22] para la propia visualización de datos provenientes de simulaciones (IPCC AR4 con el modelo ECHAM5/MPI-OM) como aporte al seguimiento del Fenómeno del Niño (ENSO). También en el campo de la visualización soportada en Wavelets se encuentra el trabajo de Gruchalla et al. [17], en el que se propone un marco de trabajo para la extracción y visualización 3D de estructuras turbulentas en fluidos mediante el realce de ciertas zonas de interés detectadas en el dominio Wavelet.

En algunos trabajos se ha empleado los resultados obtenidos mediante del flujo óptico para estimar el campo de vientos en imágenes de satélite, analizando dichos resultados a través de estudios basados en sistemas dinámicos, específicamente Retratos de Fase, como en el trabajo de Corpetti et al. [10]. Este método analítico permite representar los campos vectoriales como sistemas matriciales de ecuaciones diferenciales ordinarias. Las propiedades geométricas de los conjuntos de soluciones de dichos sistemas de ecuaciones permiten a su vez establecer una caracterización cualitativa del comportamiento del campo vectorial como se observa en el trabajo de Rao et al. [43].

1.3. Propuesta y estructura de la tesis

Esta tesis propone un método para caracterizar estructuras de bajo nivel como las descritas en la sección 1.1.2, de tal modo que permita en futuros trabajos la implementación y ejecución de procesos de detección

y extracción a partir de grandes volúmenes de datos climáticos, con el fin de aportar a la identificación de patrones atmosféricos (particularmente en Colombia) orientados al soporte de tareas de descripción y predicción climática, pronóstico del tiempo, etc.

Con tal objeto, este trabajo presenta un método basado en retratos de fase del gradiente de la velocidad del viento (*phase portraits* [54, 43], ver capítulo 3) para la caracterización y clasificación de patrones atmosféricos. Dichos métodos se evalúan mediante el uso de métodos del Estado del Arte en Aprendizaje de máquina. El capítulo 2 enumera los objetivos de la Tesis.

El capítulo 3 describe los diferentes modelos empleados para representar las distribuciones geométricas asociadas a los patrones atmosféricos de *bajo nivel* y la manera como se evaluará el desempeño de esta caracterización a través de los resultados de diferentes métodos de clasificación propuestos.

El capítulo 4 presenta los resultados ofrecidos por cada uno de los métodos de clasificación implementados mediante el uso de las características propuestas para la caracterización de las citadas estructuras atmosféricas.

El capítulo 5 analiza en detalle la naturaleza de dichos resultados y las consecuentes aportaciones de la Tesis en función de su aplicabilidad a la Meteorología y la Climatología, formular posibles vías de trabajo futuro en términos de tales aplicaciones.

Los Apéndices A, B, C y D se dedican respectivamente a detallar el proceso de Extracción, Transformación y Carga de datos WRF para su uso en los métodos propuestos, la documentación de desarrollo de la herramienta de anotaciones para la construcción del *dataset* empleado en este trabajo, los detalles técnicos de la experimentación propuesta y la primera publicación obtenida como resultado de esta línea de trabajo.

Capítulo 2

Objetivos

2.1. Objetivo general:

Proponer y validar un método para la clasificación de estructuras atmosféricas mediante aprendizaje de máquina a partir de su caracterización geométrica y física.

2.2. Objetivos específicos:

- Construir un conjunto de datos consistente en variables climáticas provenientes de simulaciones WRF en el dominio bajo estudio (norte de Sudamérica y el Caribe suroccidental).
- Formular un protocolo para la anotación de estructuras atmosféricas por parte de expertos (meteorólogos) sobre este conjunto.
- Proponer un modelo para la caracterización física y geométrica de las estructuras atmosféricas presentes en las anotaciones obtenidas.
- Diseñar e implementar un método basado en Aprendizaje de Máquina, a partir de las anotaciones de los expertos, para la clasificación automática de estructuras atmosféricas.
- Evaluar el método propuesto.

Capítulo 3

Materiales y métodos

3.1. Construcción de un conjunto de datos a partir de simulaciones WRF

En este trabajo se usan los resultados 4D (3D + tiempo) provenientes de simulaciones climáticas WRF generadas por Armenta y Pabón en su estudio exploratorio sobre cambio climático en Colombia y norte de Suramérica [1] para el periodo comprendido entre los años de 1981 y 2010 con resolución espacial de 10 km y resolución temporal de 3 horas. Para construir el *dataset*, se seleccionaron los meses con menor y mayor variabilidad en el comportamiento del fenómeno ENSO¹ [50], lo que corresponde a los resultados obtenidos para Mayo y Diciembre en el intervalo citado.

El proceso de Extracción, Transformación y Carga (ETL por sus siglas en Inglés, ver apéndice A), permitió convertir los archivos netCDF (versión WRF) resultantes de las simulaciones antes mencionadas y convertirlos a archivos crudos binarios, uno para cada variable en cada instante de tiempo y en unidades físicas.

Dado que este trabajo se concentra en estructuras atmosféricas de bajo nivel (ver sección 1.1.2, el proceso de ETL se aplicó a las componentes de vientos U , V y W y el campo de temperaturas, T para facilitar la exploración de los expertos en la herramienta de anotaciones (ver sección

¹Ver sitio web de la NOAA, consultado en http://www.cpc.ncep.noaa.gov/products/analysis_monitoring/ensostuff/ensoyears.shtml

4.1). El dominio discreto, dada la resolución espacial citada (que equivale a elementos de $10 \times 10 \text{ km}^2$ en la proyección paralela a la superficie de la Tierra) implica que para cada volumen (correspondiente a un paso de tiempo en la simulación) los cortes para cada uno de los 27 niveles isobáricos (en altura) tengan un tamaño de 287 elementos en latitud y 280 elementos en longitud. A su vez, cada mes consta de 273 tiempos dada la resolución temporal de 3 horas por muestra (ver Tabla 3.1).

Variables	Latitudes	Longitudes	Niveles
T	287	280	27
U	287	281	27
V	288	280	27
W	287	280	28

Cuadro 3.1: Tamaño de las matrices 3D correspondientes a los valores para las variables extraídas en el proceso de ETL para cada instante de tiempo. La componente horizontal de la velocidad del viento (U), tiene una longitud adicional y la componente vertical (V) una latitud de más, debido a la representación particular de los componentes de viento en los archivos netCDF producto de las simulaciones (en las caras de los cubos y no centradas en los elementos de volumen, ver sección 3.1.1

3.1.1. Corrección de la posición de las componentes de la velocidad del viento

Debido a que las componentes de la velocidad del viento están definidas sobre las caras de los elementos de volumen en la discretización en la simulación WRF, es necesario hacer una corrección para que los vectores correspondientes a la velocidad de viento –en los dominios crudos extraídos mediante el proceso de ETL– tengan sus componentes definidas siempre sobre el centro geométrico de los elementos de volumen.

Para esta labor, se hace uso de una interpolación bilineal, de modo que ésta genere valores para las componentes U y V de la velocidad en cualquier posición del dominio a partir de los valores definidos sobre las caras que delimitan el elemento (ver Figura 3.1).

3.1. CONSTRUCCIÓN DE UN CONJUNTO DE DATOS A PARTIR DE SIMULACIONE

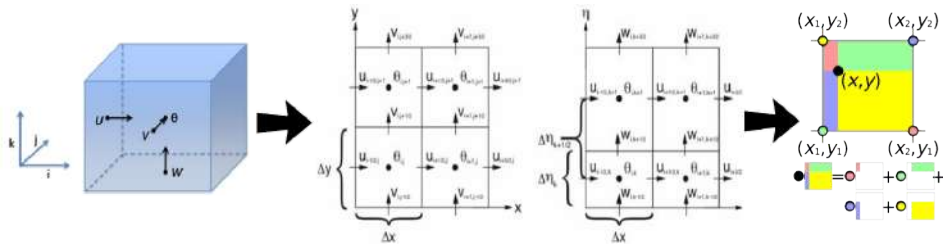


Figura 3.1: Esquema de interpolación aplicada a las componentes U , V y W para obtener una visualización del campo vectorial, se observa a la izquierda una representación de la componente de vientos presente en el modelo WRF, en la parte central como es su representación de forma discreta y en la ultima el ajuste que se le hace a cada una de las componentes de vientos para que queden centradas

3.1.2. Herramienta y protocolo de anotación

Para visualizar los archivos crudos se desarrollo una herramienta de anotaciones (ver apéndice B) que permite explorar la información extraída en el proceso ETL y tiene como objetivo permitir al experto seleccionar las regiones de interés que él considere como relevantes.

Para esta aplicación se planteo el uso de un campo escalar dado por la variable de temperatura y un campo vectorial calculado con las componentes U , V y W pero debido a un problema con sus dimensiones se debe realizar un proceso de interpolación bilinear para ajustar las componentes de vientos como se observa en la Figura 3.1.

En la aproximación utilizada en este trabajo, se usaron cortes 2D paralelos a la superficie terrestre, visualizados en la herramienta de anotaciones como una distribución bidimensional de temperatura en pseudocolor, sobre la cual se superpone una capa vectorial que contiene las líneas de costa y la proyección sobre dichos cortes del campo vectorial correspondiente a la velocidad del viento al mismo nivel de presión (componentes U y V).

En efecto, las estructuras atmosféricas son presentadas al experto encargado de la anotación como configuraciones geométricas del campo de vientos presentes en una región determinada (vorticidad, difluencias, confluencias y puntos de silla), de modo que éste pueda identificar y especificar

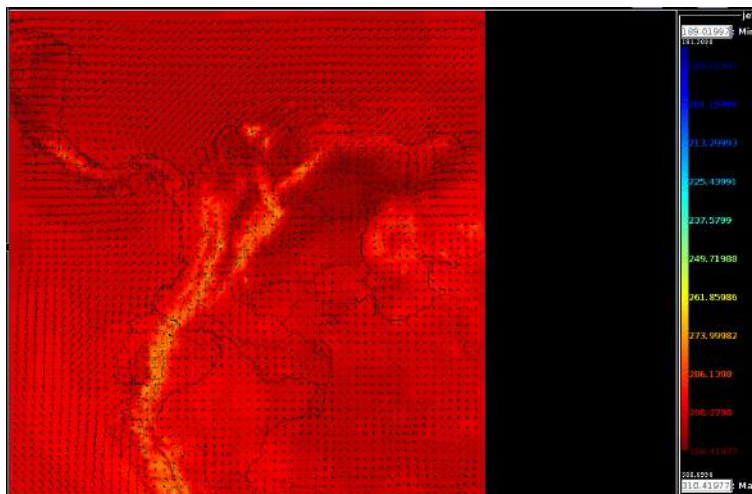


Figura 3.2: Visualización de los datos correspondiente a la variable de temperatura potencial y la magnitud de las componentes de vientos de forma vectorial

tales patrones mediante la selección de ROIs (regiones de interés), tal y como se presentan en la Figura 3.3 y siguiendo el protocolo descrito en el algoritmo 1.

El observar estas configuraciones resulta mas relevante para los expertos que la visualización del campo escalar propiamente ya que da más información la componente vectorial mostrando los diferentes comportamientos de la atmósfera a distintas alturas en diferentes tiempos.

3.2. Caracterización de patrones atmosféricos

3.2.1. Aproximación de Operadores Diferenciales sobre dominios discretos

Dado el operador $\nabla = \left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y} \right)$ y el vector $\mathbf{v} = (u, v)$ un campo vectorial que representa la proyección 2D del campo de vientos en un plano *paralelo* a la superficie de la Tierra para una capa isobárica determinada. A partir de esta formalización, los operadores *Divergencia* y *Rotacional*

Algoritmo 1 Protocolo de anotación de patrones atmosféricos de bajo nivel

- 1: Cargar el archivo de *metadatos* que define la configuración de los datos de entrada y la geometría del dominio de las variables por anotar (temperatura y componentes de la velocidad del viento).
 - 2: Establecer un nivel de altitud para la comenzar la búsqueda de ROIs que contengan patrones atmosféricos identificables
 - 3: Inicio de la anotación: al identificar un patrón atmosférico que corresponda a una de las estructuras de interés, seleccionar el tipo de estructura (vorticidad, confluencia, difluencia o punto de silla) en el botón de radio del panel izquierdo de la herramienta.
 - 4: Una vez seleccionado el tipo de estructura, enmarcar con el ratón una ROI que contenga un patrón atmosférico observado, usando la silueta mas conveniente de acuerdo a la geometría del patrón (para este estudio se recomendó usar únicamente rectángulos).
 - 5: Una vez se ha seleccionado la ROI, se presiona el botón de anotación y se introduce el texto descriptivo del patrón anotado.
 - 6: Presionar el botón de guardado para almacenar progresivamente en el archivo XML de salida las anotaciones definidas sobre la altitud actual.
 - 7: Volver al paso 2 para seleccionar una nueva altitud cuando se hayan anotado todas las estructuras observables en el nivel actual.
 - 8: Una vez se haya recorrido todo el dominio en altura para un tiempo dado, se presiona el botón de avanzar tiempo para iniciar en el nivel de altitud base en el tiempo inmediatamente posterior.
-

permiten describir el comportamiento local de un fluido por medio de sus valores sobre el dominio bajo estudio. Como los datos usados en este estudio tienen una naturaleza discreta (resultados de simulaciones computacionales), es necesario entonces formular dichos operadores en una versión igualmente discreta.

De este modo, \mathbf{v} se puede expresar como una matriz $V = [(u, v)_{ij}]_{M \times N}$ donde $i = 1, 2, \dots, M$ son los índices para cada fila y $j = 1, 2, \dots, N$ son los índices para cada columna.

Divergencia La divergencia de \mathbf{v} es definida como $\nabla \cdot \mathbf{v} = \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}$ para el caso bidimensional. Dado que es un producto punto, la divergencia es un campo escalar, que se puede obtener mediante una aproximación numérica usando diferencias finitas centradas para ambas derivadas parciales.

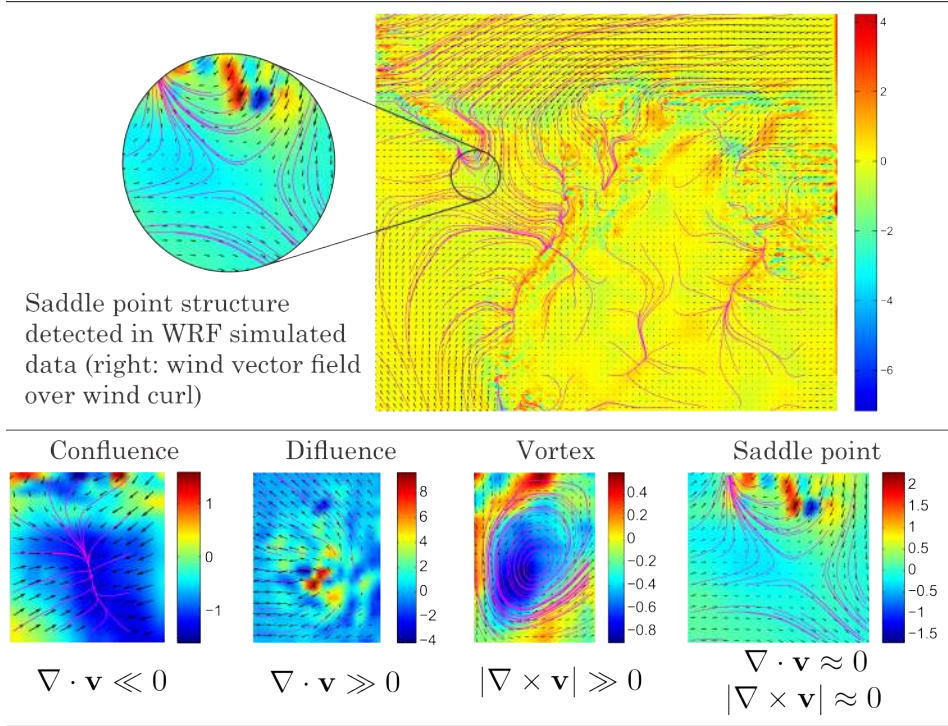


Figura 3.3: Caso representativos de las estructuras atmosféricas bajo estudio Confluencias, Difuencias, Vorticidades y puntos de silla

En el caso de 2D la divergencia [20] puede ser definida como

$$\nabla \cdot \mathbf{v} \approx \frac{(u_{i+1,j} - u_{i-1,j})}{2h} + \frac{(v_{i,j+1} - v_{i,j-1})}{2h}$$

donde i y j corresponden a los índices de las posiciones en la discretización del dominio y h es el cambio relativo (tamaño del paso) en la dirección de las componentes U y V .

Rotacional El rotacional de un campo vectorial determina la cantidad de los vectores dentro del dominio bajo estudio “giran” alrededor de cada posición particular. Una aproximación usual para calcular el rotacional de

un campo vectorial en el caso 3D –con $\mathbf{v} = (u, v, w)$ – utiliza la forma de un pseudo–determinante:

$$\nabla \times \mathbf{v} = \begin{vmatrix} \bar{x} & \bar{y} & \bar{z} \\ \frac{\partial}{\partial x} & \frac{\partial}{\partial y} & \frac{\partial}{\partial z} \\ u & v & w \end{vmatrix}$$

en el caso del 2D el rotacional [20] se define como $\nabla \times \mathbf{v} = \left(\frac{\partial v}{\partial x} - \frac{\partial u}{\partial y} \right) \bar{z}$ que es siempre perpendicular a ese plano y se puede aproximar por:

$$\nabla \times \mathbf{v} \approx \frac{(v_{i+1,j} - v_{i-1,j})}{2h} - \frac{(u_{i,j+1} - u_{i,j-1})}{2h}$$

donde i y j corresponden a los índices de las posiciones en la discretización del dominio y h es el cambio relativo (tamaño del paso) en la dirección de las componentes U y V .

3.2.2. Descriptores basados en histogramas

Tomando en cuenta que una estructura atmosférica puede ocurrir en distintos puntos del dominio discreto, en distinto nivel isobárico y a diferentes escalas, se utilizó un descriptor invariante a translación y escala usando una diseño piramidal (SPL por sus siglas en ingles *Spatial Pyramidal Layout*) aplicado a la ROI. El SPL es extraído de los campos calculados por los operadores diferenciales (Divergencia y Rotacional) y su representación es un histograma de 256 características de los operadores diferenciales normalizados. También se calculan otros descriptores basados en Max–Pooling y el histograma de orientación de flujo óptico (HOOF) aplicados a cada uno de los operadores diferenciales.

Representación espacial de diseño piramidal (SPL)

Esta representación se usó para capturar las estructuras atmosféricas en diferentes ubicaciones y tamaños como se muestra en la Figura 3.4. Este descriptor se extrae desde los *campos normalizados* de divergencia, divergencia negativa y rotacional, de la siguiente manera: a) se extrae su histograma de los campos extraídos de la ROI completa; b) la ROI se divide en 2×2 secciones y se extrae el histograma de cada uno los campos

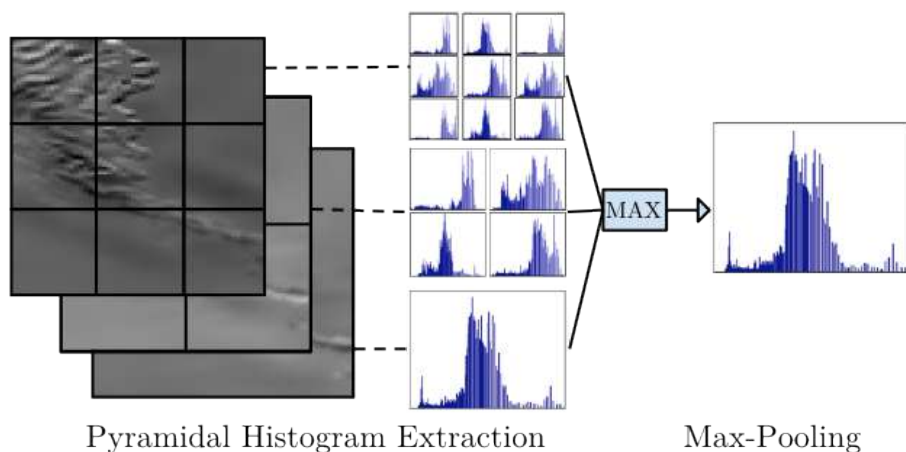


Figura 3.4: Los histogramas se extraen en un diseño piramidal de los diferentes mapas de características. Los 14 histogramas resultantes se integran adicionalmente en el descriptor max-pooling, que corresponde a un histograma con los valores máximos para todas las características.

calculados independientemente sobre dichas secciones y c) se divide la ROI en 3×3 secciones y, consecuentemente, se calculan los histogramas de los campos para esta nueva partición [26, 51].

De esta manera, el descriptor se construye como una concatenación de estos 14 histogramas por cada campo (equivalente a un vector de $256 \times 14 = 3584$ componentes). Así pues, el descriptor final corresponde a un vector con $3 \times 3584 = 10752$ características.

Max-pooling

La aproximación de *Max-pooling* para construir el descriptor utiliza una estrategia de selección de los valores máximos de cada *bin* del histograma en lugar de concatenar histogramas (ver Figura 3.4). En principio, esta estrategia permitiría detectar las estructuras atmosféricas independientemente su ubicación y escala (invarianza espacial [37]). En la implementación propuesta, para obtener el descriptor basado en Max-pooling, se aplica el operador *máximo* sobre los 14 histogramas de todas las particiones de la

ROI. Naturalmente, el tamaño de este descriptor es un vector de 256 características por campo, luego la representación final del descriptor consiste en un vector de 768 características.

Histograma de orientaciones del flujo óptico (HOOF)

Para construir este descriptor se toma como línea base el histograma de orientación de flujo óptico propuesto por Chaudry et al. [8] con el objeto de detectar los patrones de orientación del flujo de viento presentes en los resultados de las simulaciones. Adicionalmente, el esquema piramidal *SPL* se aplica a esta representación para la determinación de las características a diferentes escalas. Este histograma permite capturar las variaciones más grandes del campo vectorial, como se presenta en la Figura 3.5; en este caso de estudio, estas variaciones son visibles en cada una de las estructuras fluidas propuestas. De esta manera, cada partición se representa como un histograma de 256 características, dadas por la cuantización de las distintas orientaciones, obteniéndose finalmente un vector de tamaño de $256 \times 14 = 3584$ características para la representación piramidal.

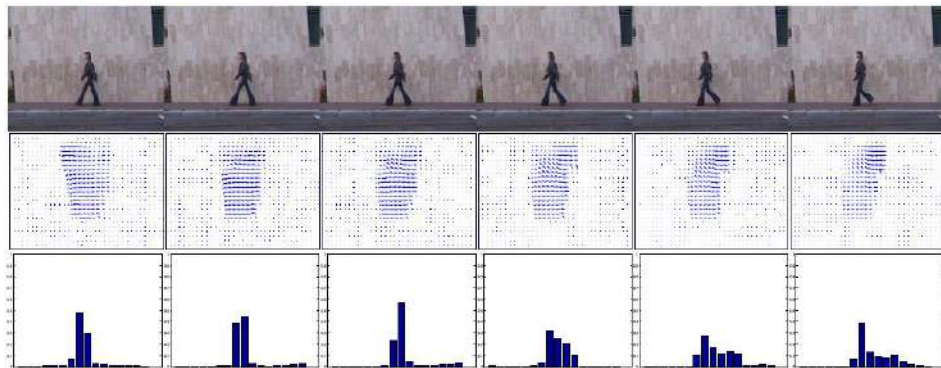


Figure 1. Optical flows and HOOF feature trajectories

Figura 3.5: Esquema del Histograma orientada de flujo óptico (HOOF) gráfico tomado del paper de Chaudhry et al. Adaptado con permiso [8]

3.2.3. Descriptor basado en retratos de fase

El método de **Retratos de Fase** permite obtener la solución de ecuaciones diferenciales de forma analítica usando una representación geométrica del comportamiento del sistema, este método se usa en sistemas de ecuaciones diferenciales de 2 o más variables y funciona de la siguiente manera, se plantea el sistema de ecuaciones diferenciales como un sistema matricial,

$$\frac{d\mathbf{x}}{dt} = \mathbf{v} = \mathbf{X}(\mathbf{x})$$

donde $\mathbf{X} : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ se asume como una transformación lineal que puede ser escrita de la siguiente manera

$$\begin{bmatrix} u'(x, y) \\ v'(x, y) \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}$$

donde \mathbf{b} es un vector de constantes, como lo muestra Rao et al. [43], la forma normalizada de Jordan J de la matriz A permite la clasificación de las distintas configuraciones que se pueden presentar en el campo vectorial para esto se calculan los autovalores y autovectores que terminan siendo la representación del comportamiento del sistema de ecuaciones diferenciales, esta descripción es dada en [43, 10, 54]

$$\dot{x} = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

Donde $\lambda_{1,2}$ son los coeficientes de la función x y y del sistema de ecuaciones diferenciales. El cálculo de los autovalores se da mediante el cálculo de la traza ($\tau = \lambda_1 + \lambda_2$) y el determinante ($\Delta = \lambda_1 \lambda_2$), ya con esto calculado se puede definir el comportamiento del sistema

$$\lambda_{1,2} = \frac{1}{2} \left(\tau \pm \sqrt{\tau^2 - 4\Delta} \right) \quad (3.1)$$

Existen formas de normalización de estos campos vectorial como lo menciona [6] donde se hablan de unos momentos de invarianza que permitiría una representación mas robusta de los patrones que se quieren detectar.

Implementación del descriptor

Para implementar este descriptor se extraen las componentes U y V de la ROI y se calcula su respectivo gradiente (∇), con el fin de observar cambios en las direcciones horizontal (longitudinal) y vertical (latitudinal) para ambas componentes vectoriales (ver Figura 3.6).

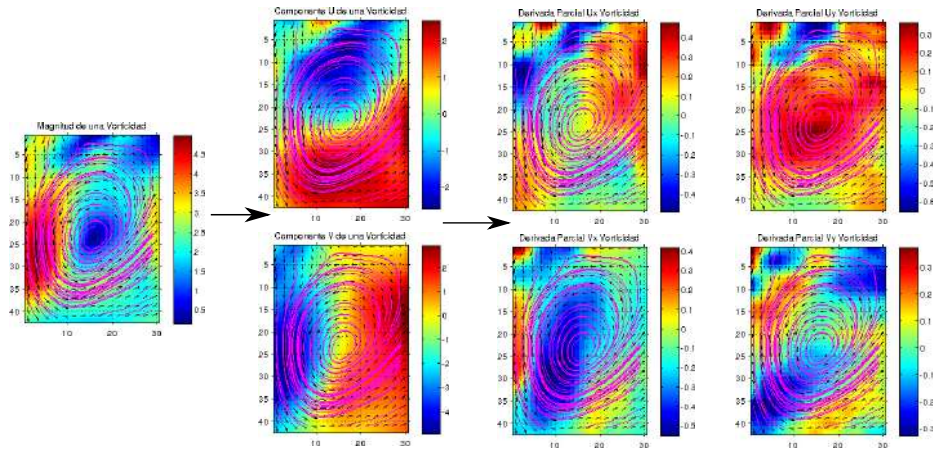


Figura 3.6: Representación de la región de interés, junto con sus componentes U y V y sus respectivas derivadas parciales para las componentes correspondientes (x y y); se puede observar que cada derivada parcial representa de manera más pronunciada los cambios en los campos U y V que la distribución de valores de los propios campos.

Después de obtener los ∇U y ∇V se crea el siguiente sistema de ecuaciones:

$$A = \begin{bmatrix} \frac{\partial U}{\partial x} & \frac{\partial U}{\partial y} \\ \frac{\partial V}{\partial x} & \frac{\partial V}{\partial y} \end{bmatrix}$$

donde $\frac{\partial U}{\partial x}$, $\frac{\partial U}{\partial y}$, $\frac{\partial V}{\partial x}$ y $\frac{\partial V}{\partial y}$ son las derivadas parciales de U y V con respecto a x y y , este calculo se realiza para resaltar que componentes de viento tiene mayor información de la estructura mediante las tasas de cambios presentes en sus componentes x y y .

La idea de calcular el gradiente es crear una matriz donde se puedan identificar los puntos fijos, usando la ecuación característica $\det(A - I\lambda) = 0$

donde I es la matriz identidad. Para resolver este sistema de ecuaciones se utiliza la ecuación cuadrática de la expresión 3.1 que permite obtener los valores de λ , donde Δ es el determinante de A y τ es la traza de A . A partir de la estimación de estas dos componentes, se obtendrán los puntos fijos $\lambda_{1,2}$.

Para la construcción del descriptor se calculan 3 estadísticos (media, desviación estándar y varianza) al determinante y la traza con el fin de obtener un comportamiento general de la ROI que es dado por los cambios de los campos de las derivadas parciales y reducir el campo que compone el determinante y la traza a un solo valor, con estos valores se calcula $\lambda_{1,2}$. El descriptor estará compuesto por los 6 valores de λ que contendrán un valor real y un valor complejo y los 3 estadísticos calculados en Δ y τ , esta representación busca aprovechar las reglas de clasificación propuestas en sistemas dinámicos [43, 54], el tamaño que obtiene esta representación es de 18 características que representan los 6 λ obtenidos además de los 3 estadísticos calculados al determinante (Δ) y los 3 estadísticos calculados a la traza (τ).

3.3. Clasificación y evaluación del desempeño de los descriptores

Para evaluar el desempeño de los descriptores bajo estudio se usó un esquema de *k-fold stratified cross-validation* (validación cruzada estratificada), que permite disminuir el inbalance de clases que se presentó en el proceso de anotaciones y obtener los mejores parámetros de los modelos de clasificación. Se utilizaron 2 máquinas de aprendizaje: SVM (Máquinas de vectores de soporte) [11] y *Random Forest* [5] con el fin de seleccionar la mejor configuración (descriptor y máquina de aprendizaje) que represente las ROIs.

3.3.1. SVM

Son algoritmos de aprendizaje de máquinas supervisados que representan un conjunto de datos en un espacio de características con el fin de separar las diferentes clases de forma óptima mediante el uso de un hiperplano [11]. El principio básico del SVM es buscar que el hiperplano tenga

3.3. CLASIFICACIÓN Y EVALUACIÓN DEL DESEMPEÑO DE LOS DESCRIPTORES

una separación máxima con los puntos que esté cerca de él, diferenciando las dos clases, De esta manera los valores etiquetados de una clase estarán a un lado del hiperplano mientras los valores restantes estarán al otro lado.

Para la tarea de clasificación, la SVM es definida como: dado un conjunto de entrenamiento \mathcal{D} y un conjunto de n puntos de la forma

$$\mathcal{D} = \{(\mathbf{x}_i, y_i) \mid \mathbf{x}_i \in \mathbb{R}^p, y_i \in \{-1, 1\}\}_{i=1}^n$$

Donde y_i son las dos diferentes clases $(1, -1)$ a las que pertenece el punto \mathbf{x}_i , cada uno de los \mathbf{x}_i pertenece a un espacio real p -dimensional, para encontrar el máximo hiperplano que divida los puntos con $y_i = 1$ de los puntos con $y_i = -1$, este hiperplano se puede escribir de la forma:

$$\mathbf{w} \cdot \mathbf{x} + b = 0$$

Donde \cdot representa el producto escalar y \mathbf{w} es el vector normal al hiperplano, la parametrización de $\frac{b}{|\mathbf{w}|}$ determina el desplazamiento del hiperplano desde el origen a lo largo del vector normal \mathbf{w} . Si el conjunto de entrenamiento es linealmente separable, se pueden crear dos hiperplanos que permitan una separación de los datos y con eso poder maximizar la distancia entre las dos clases, esta distancia se le conoce con el nombre de Margen, la forma de escribir estos dos hiperplanos es:

$$\mathbf{w} \cdot \mathbf{x} - b = 1 \quad \text{y} \quad \mathbf{w} \cdot \mathbf{x} - b = -1$$

Este modelo puede ser expresado como un una función de decisión:

$$\hat{y} = \text{sgn} \sum_{i=1}^n y_i \alpha_i K(x_i, x) + \rho$$

donde \hat{y} es la etiqueta dada por el clasificador binario usando el *kernel* $K(x_i, x)$ para una entrada dada por \mathbf{x} , siendo $y_i \alpha_i$ los pesos de las máquinas de soporte y ρ el intercepto.

En el caso ideal donde todas las clases se comportaran de forma lineal esto seria suficiente, pero en el mundo real pocos fenómenos presentan un comportamiento lineal es por esta razón que existen otras estrategias que permiten ajustar a los hiperplanos mediante el uso de una función de

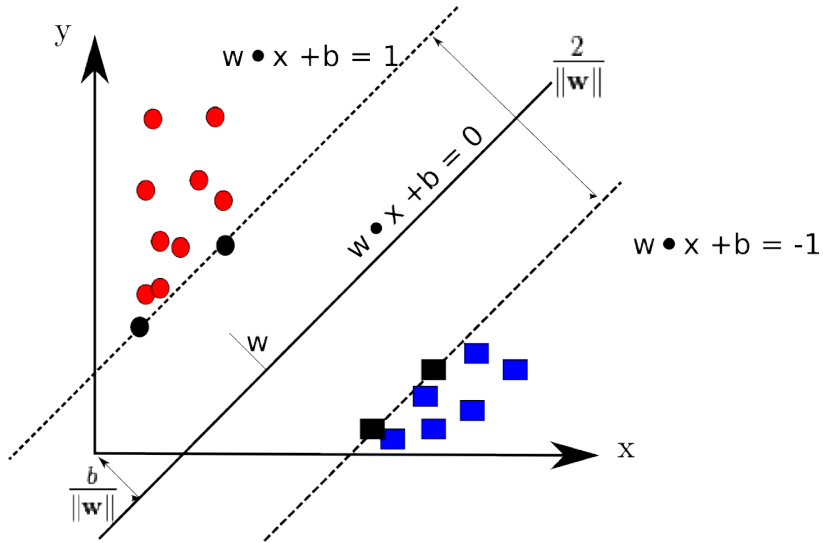


Figura 3.7: Esquema de SVM usando un kernel Lineal para 2 clases

kernel, que es una función de similitud que se busca ajustar el espacio de características a la distribución que presenta los datos. Se han diseñado diferentes *kernels* que se ajustan a la representación de los datos en el espacio de características. A continuación se presentarán la definición de algunos *kernels* que son usados en la tarea de clasificación:

- *Kernel Lineal*

$$K(x_i, x_j) = x_i \cdot x_j$$

donde x_i y x_j son vectores en el espacio de entrada

- *Kernel Polinomial*

$$K(x_i, x_j) = (x_i \cdot x_j + c)^d$$

donde x_i y x_j son vectores en el espacio de entrada y c es el intercepto de la función y d es el grado del polinomio que se quiere manejar debe ser mayor o igual que 1

3.3. CLASIFICACIÓN Y EVALUACIÓN DEL DESEMPEÑO DE LOS DESCRIPTORES

- *Kernel Gaussiano Función en base radial*

$$K(x_i, x_j) = \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right)$$

donde x_i y x_j es el espacio de características dos muestras en el espacio de entrada y σ es un parámetro libre.

- *Kernel de tangente hiperbólica*

$$K(x_i, x_j) = \tanh(\kappa x_i \cdot x_j + c)$$

donde x_i y x_j son los vectores del espacio de entrada, κ el peso de las máquinas de soporte y c es el intercepto.

Para la experimentación se usaron dos configuraciones de *kernel*: una usando un *kernel* Lineal y la otra con un *Kernel* en función de base radial [49, 19].

3.3.2. Random Forest

Esta técnica se basa en una combinación de varios arboles de decisión que aprenden a partir de varios subconjuntos independientes a través de un método de *bagging*, este método se utiliza para generar subconjuntos de datos, se caracteriza por su imparcialidad en la separación de clases, reducción de la varianza y la reducción del sobreajuste que se presenta en el conjunto [2].

El principio básico de *Random Forest* es crear diferentes subconjuntos independientes uno de otro y utilizar diferentes arboles de decisión que clasifiquen cada uno de los elementos de validación y/o entrenamiento. Después que clasifica cada uno de los arboles se hace un proceso de votación con el fin de determinar la clase a la que pertenece, este método es muy bueno en conjunto de datos grandes.

3.3.3. K-Fold Stratified Cross-Validation

Para evaluar el conjunto de entrenamiento se propone una estrategia de validación cruzada que permita evaluar los diferentes parámetros que deben tener las máquinas de aprendizaje para obtener el mejor desempeño. Este

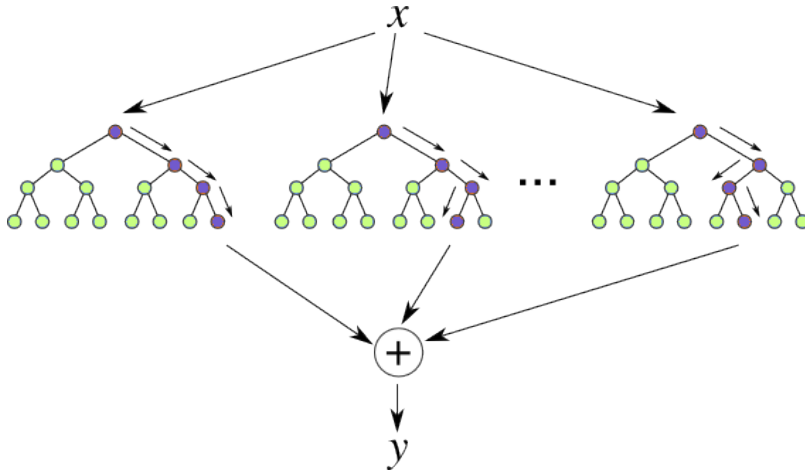


Figura 3.8: Esquema de *Random forest* para clasificación

proceso se hace a cada uno de los descriptores anteriormente descritos, para esto se seleccionó como estrategia una validación cruzada estratificada [44].

Este método utiliza k particiones de prueba de tamaño n y se selecciona el resto de entrenamiento, este conjunto de prueba tendrá la misma probabilidad de que aparezca cada una de las clases, la misma cantidad de veces con el fin de reducir el imbalance de clase presentado en el proceso de anotaciones realizado por el experto. El objetivo de esta técnica es encontrar los mejores parámetros de evaluación de cada una de las máquinas de aprendizaje.

3.3.4. Evaluación mediante matrices de confusión

Para calcular las medidas de desempeño de las máquinas de aprendizaje se utilizan las matrices de confusión que son una herramienta para la evaluación del desempeño de los algoritmos de aprendizaje de máquina supervisado. Se compone como una matriz cuadrada donde cada fila representa la clase real mientras que las columnas representa la clase que se predijo.

Esta representación funciona de manera binaria tomando *verdaderos positivos (TP)*, *verdaderos negativos (TN)*, *falsos positivos (FP)* y *falsos*

3.3. CLASIFICACIÓN Y EVALUACIÓN DEL DESEMPEÑO DE LOS DESCRIPTORES

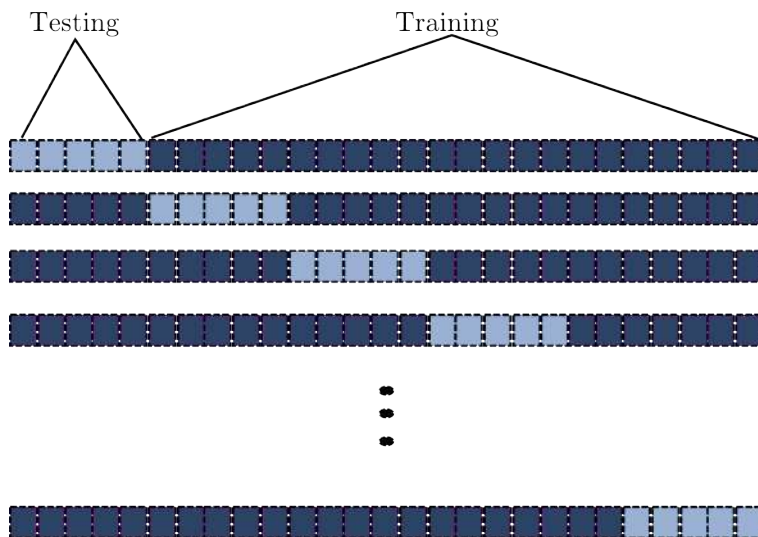


Figura 3.9: Esquema de Validación cruzada

negativos (FN) con estas medidas se puede calcular diferentes medidas de desempeño que permiten evaluar el comportamiento de la máquina de aprendizaje. También permite observar las diferentes clases que se confunden.

El esquema de la Figura 3.10 describe la presentación usual de diferentes medidas de desempeño para el proceso de clasificación.

	Condición positiva	Condición negativa
Predicción de condición positiva	TP	FP
Predicción de condición negativa	FN	TN

Figura 3.10: Esquema de matriz de confusión

Tasa de verdaderos positivos (Sensibilidad) (TPR)

Se encarga de medir la proporción de positivos que están correctamente identificados

$$TPR = \frac{TP}{P} = \frac{TP}{TP + FN}$$

Tasa de verdaderos negativos (Especificidad) (SPC)

Se encarga de medir la proporción de negativos que están correctamente identificados

$$SPC = \frac{TN}{N} = \frac{TN}{FP + TN}$$

Precisión (PPV)

Es la fracción de casos positivos correctos que se recuperan

$$PPV = \frac{TP}{TP + FP}$$

Precisión promedio (Average Accuracy)

Para evaluar los resultados de las secuencias experimentales propuestas en este trabajo se usará la precisión promedio (Average Accuracy), pues ésta permite obtener una medida descriptiva del desempeño general de cada proceso de clasificación a partir de los valores de precisión para cada una de las clases bajo estudio. La precisión promedio se puede calcular según la expresión

$$\text{Average Accuracy} = \frac{1}{n} \sum_{i=0}^{n-1} PPV \quad (3.2)$$

donde n representa el número de clases que tiene el problema.

Capítulo 4

Resultados

4.1. Herramienta de anotaciones de patrones atmosféricos

Se desarrollo una herramienta de anotaciones que permite además de visualizar los volúmenes extraídos del *WRF* usando un esquema de carpetas donde estaban cada una de las variables, y esta contenía archivos de cada una separadas por tiempo y un archivo XML que permitían la visualización de los datos, esta herramienta permite seleccionar regiones de interés (ROI) usando figuras geométricas regulares (Elipse y Rectángulo) o irregulares (Polígonos), ver anexo B.

El resultado de las marcaciones era almacenado en un archivo XML (como se muestra en el apéndice B) que contenía las 4 clases correspondientes y los niveles y tiempo anotados.

4.2. Descripción del conjunto de datos con anotaciones

El proceso de anotación se llevó a cabo utilizando el protocolo descrito (ver Algoritmo 1). Las descripción y características de la distribución geométrica de las estructuras fluidas bajo estudio fueron discutidas con un

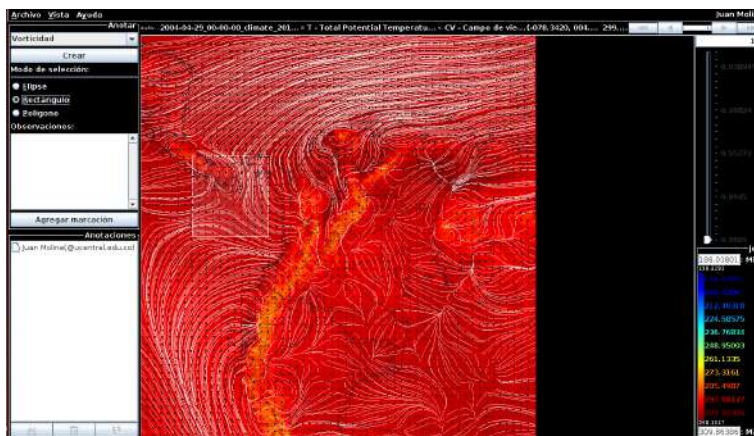


Figura 4.1: Herramienta de anotaciones visualizando temperatura y vientos

experto en Meteorología¹ quien además validó mediante inspección visual las anotaciones conseguidas mediante la aplicación del mencionado protocolo por parte de un asistente de investigación sobre la herramienta de anotaciones de patrones atmosféricos (sección 4.1). Para este se seleccionaron dos años diferentes que fueron los años 2004 y 2005 por ser un año donde se presentó el fenómeno del niño y un año neutro².

El conjunto de datos anotado contiene 793 anotaciones que están divididas como se muestra en la tabla 4.1; el tamaño promedio de de las ROI pertenecientes a este *dataset* es de $(37,12 \pm 14,62) \times (34,88 \pm 14,85)$ elementos de superficie (píxeles).

Por simplicidad, para el proceso de aprendizaje supervisado se dividió el conjunto de datos de la siguiente manera:

- Un subconjunto de entrenamiento con 594 ROIS anotadas, correspondientes a los meses de Mayo y Diciembre de 2004, y Diciembre de 2005.
- Un conjunto de pruebas, que contiene las 199 ROIs anotadas restan-

¹En este trabajo participó como asesora la prof. Diana Díaz, PhD (c) de la Universidad de Bogotá Jorge Tadeo Lozano, MSc en Meteorología

²Este registro lo hace la NOAA en la página: http://www.cpc.ncep.noaa.gov/products/analysis_monitoring/ensostuff/ensoyears.shtml

Clases	Confluencia	Difluencia	Puntos de silla	Vorticidad	Cantidad
Mayo de 2004	79	59	44	32	214
Diciembre de 2004	39	47	35	78	138
Mayo de 2005	30	29	40	39	199
Diciembre de 2005	84	31	58	69	242
Total	232	166	177	218	793

Cuadro 4.1: División del dataset anotado por clase y mes de simulación

tes, correspondientes al mes de Mayo de 2005.

4.3. Desempeño del proceso de clasificación

Como se menciona en la sección 4.2, el tamaño del conjunto de entrenamiento es de 594 y el de prueba es de 199. De este modo, se llevaron a cabo dos experimentaciones, una para los descriptores basados en histogramas y otra para el descriptor basado en retratos de fase. A continuación se presentan los detalles de cada secuencia experimental.

4.3.1. Descriptor basado en histogramas

Para evaluar el desempeño de cada descriptor se usó la estrategia propuesta en la sección 3.3.3, consistente en un proceso de *k-fold stratified cross-validation* con $k = 5$, entrenando respectivamente una SVM lineal, una SVM con funciones de base radial (SVM-RBF) y un método basado en *Random Forest*. Las SVM se evaluaron para los valores de los parámetros C y γ^3 . En el caso de *Random Forest*, se evaluó la cantidad de arboles predictores requeridos (n).

³ C es el parámetro que controla la influencia de cada uno de los vectores de soporte y suaviza el margen de la función de costo; γ define el comportamiento del Kernel Gaussiano usado.

En adelante, para cada descriptor se usará la siguiente notación:

- *pyrHOOF*: Histograma orientado de flujo óptico con la estrategia Piramidal.
- *histdivcurl*: Concatenación del histograma de divergencia, divergencia negativa y rotacional.
- *maxpooldivcurl*: Max-pooling a la concatenación del esquema piramidal extraído del histograma de de divergencia, divergencia negativa y rotacional.
- *pyrdivcurl*: Esquema piramidal aplicado a los histogramas de divergencia, divergencia negativa y rotacional.

En el proceso de entrenamiento se aplicó el método de validación cruzada estratificada, de modo que se encontraron los parámetros con el mejor desempeño (precisión promedio (AVG Accuracy) evaluada sobre el conjunto de entrenamiento propuesto), obteniéndose los siguientes valores para cada máquina de clasificación:

- SVM - Lineal
 - *histdivcurl*: $C = 0,0039$, AVG Accuracy = 0,85.
 - *pyrdivcurl*: $C = 0,0002$, AVG Accuracy = 0,80.
 - *maxpooldivcurl*: $C = 0,0039$, AVG Accuracy = 0,80.
- SVM - RBF
 - *histdivcurl*: $C = 2$, $\gamma = 0,0010$ AVG Accuracy = 0,83.
 - *pyrdivcurl*: $C = 4$, $\gamma = 3,0517e - 05$, AVG Accuracy = 0,83.
 - *maxpooldivcurl*: $C = 2$, $\gamma = 0,0010$, AVG Accuracy = 0,83.
 - *pyrHOOF*: $C = 2$, $\gamma = 0,0002$, AVG Accuracy = 0,69.
- *Random Forest* (RF)
 - *histdivcurl*: $n = 10000$, AVG Accuracy = 0,80.
 - *pyrdivcurl*: $n = 10000$, AVG Accuracy = 0,81.

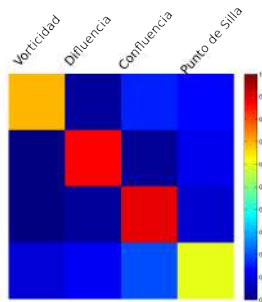
- *maxpooldivcurl*: $n = 10000$, AVG Accuracy = 0,81.
- *pyrHOOF*: $n = 10000$, AVG Accuracy = 0,67.

En la tabla 4.2 se muestra el desempeño de los métodos propuesto evaluado en términos de acierto promedio (ver ecuación 3.2) para diferentes representaciones con distintos modelos de clasificación. Observe que estos resultados se obtuvieron a partir de datos independientes correspondientes a un mes de la simulación como se demuestra en la tabla 4.1. La mejor medida de desempeño se logró mediante la representación SPL sin aplicar *max-pooling* (*pyrdivcurl*) y usando una máquina de vectores de soporte con un *kernel* RBF obtuvo una precisión promedio de 0.81, la clase de confluencia tiene el mejor desempeño reportado que es de 0.92 y la clase de puntos de silla tuvo el desempeño más bajo reportado que es de 0.77.

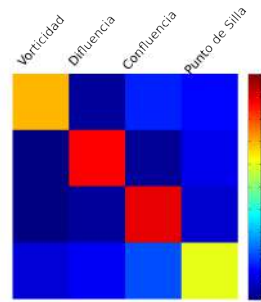
También se pueden observar el comportamiento de cada una de las clases se pueden observar las diferentes matrices de confusión que se observan en las figuras 4.2, 4.3, 4.4.

Descriptor	Clasificador	Vorticidad	Difluencia	Confluencia	Punto de Silla	Desempeño Promedio
pyrdiveurl	SVM-RBF	0.71	0.82	0.92	0.77	0.81
histdivcurl	SVM-RBF	0.69	0.87	0.89	0.6	0.765
maxpooldivcurl	SVM-RBF	0.69	0.87	0.89	0.6	0.765
pyrHOOF	SVM-RBF	0.76	0.87	0.56	0.48	0.672
pyrdiveurl	SVM-Linear	0.74	0.8	0.92	0.71	0.797
histdivcurl	SVM-Linear	0.75	0.78	0.87	0.6	0.753
maxpooldivcurl	SVM-Linear	0.75	0.78	0.87	0.6	0.753
pyrdiveurl	RF	0.73	0.8	0.94	0.6	0.771
maxpooldivcurl	RF	0.69	0.85	0.92	0.51	0.745
histdivcurl	RF	0.69	0.85	0.92	0.45	0.73
pyrHOOF	RF	0.62	0.97	0.58	0.34	0.634

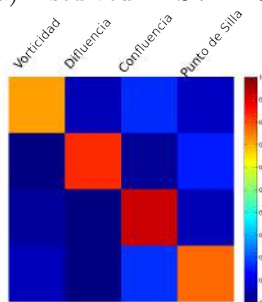
Cuadro 4.2: Resultados del Desempeño de la clasificación en el conjunto de prueba en términos de acierto observando la precisión por clase y la precisión promedio



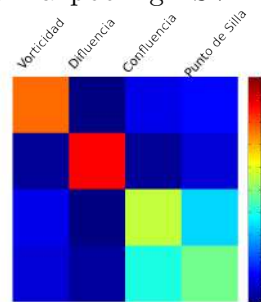
(a) histdivcurl - SVM RBF



(b) Maxpooling - SVM RBF

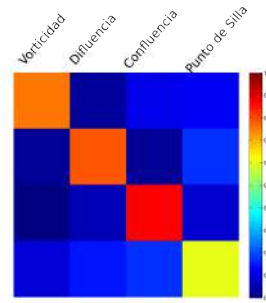
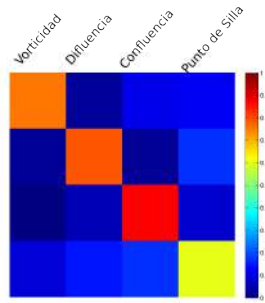


(c) pyrdivcurl - SVM RBF

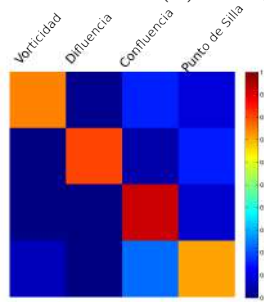


(d) Pyrhoof - SVM RBF

Figura 4.2: Matrices de Confusión de evaluados en una máquina de aprendizaje SVM - RBF donde (a) es la concatenación de los histogramas extraídos de los operadores diferenciales divergencia y rotacional, (b) es el maxpooling de los 14 Histogramas extraídos con el esquema piramidal, (c) es la concatenación de los 14 histogramas de cada uno de los operadores diferenciales y (d) es el esquema piramidal usando HOOF como característica



(a) histdivcurl - SVM Lineal (b) Maxpooling - SVM Lineal



(c) pyrddivcurl - SVM Lineal

Figura 4.3: Matrices de Confusión de evaluados en una máquina de aprendizaje SVM Lineal donde (a) es la concatenación de los histogramas extraídos de los operadores diferenciales divergencia y rotacional, (b) es el *max-pooling* de los 14 Histogramas extraídos con el esquema piramidal y (c) es la concatenación de los 14 histogramas de cada uno de los operadores diferenciales

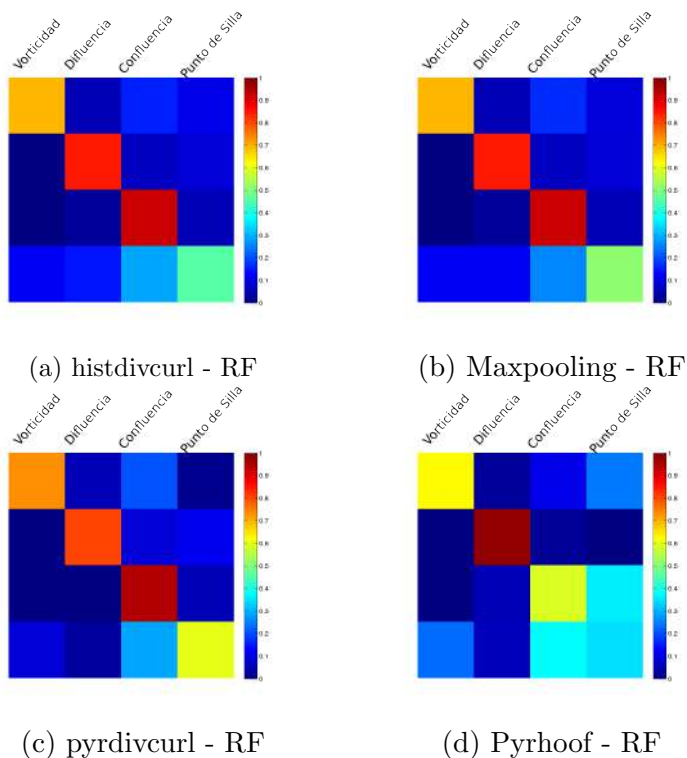


Figura 4.4: Matrices de Confusión de evaluados en una máquina de aprendizaje Random Forest donde (a) es la concatenación de los histogramas extraídos de los operadores diferenciales divergencia y rotacional, (b) es el maxpooling de los 14 Histogramas extraídos con el esquema piramidal, (c) es la concatenación de los 14 histogramas de cada uno de los operadores diferenciales y (d) es el esquema piramidal usando HOOF como característica

4.3.2. Descriptor basado en retratos de fase

Para la evaluación del descriptor basado en retratos de fase se usó un modelo de clasificación basado en una SVM Lineal sobre el conjunto de datos para entrenamiento y prueba descrito en la sección 4.2), lo que permite comparar su desempeño con los resultados obtenidos para la experimentación correspondiente a descriptores basados en histogramas (ver sección

4.3.1). La precisión promedio reportada para este descriptor fue de 0.89.

Los resultados que se observan en la tabla 4.3 muestran que las clases con mejor precisión fueron: vorticidad, difluencia y confluencia con un porcentaje superior al 0.9; aunque en el caso de los puntos de silla se obtuvo un porcentaje de 0.74. Para esta experimentación solo se usó la SVM lineal donde el parámetro $C = 1$.

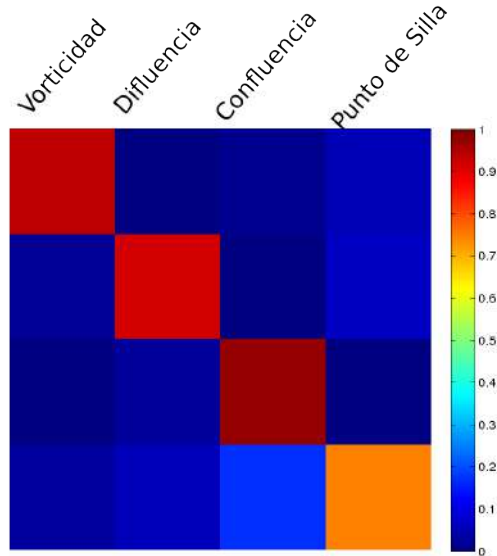


Figura 4.5: Resultados de la Matriz de confusión de los descriptores basados en retratos de fase

Descriptor	Clasificador	Vorticidad	Difluencia	Confluencia	Punto de Similitud	Desempeño promedio
retratos de fase	SVM-Linear	0.94	0.91	0.97	0.74	0.89
pyrdivcurl	SVM-RBF	0.71	0.82	0.92	0.77	0.81

Cuadro 4.3: Resultado del descriptor de retratos de fase donde se muestra la precisión de cada una de las clases y el desempeño promedio del descriptor con la máquina de aprendizaje

Capítulo 5

Discusión y Conclusiones

Esta tesis presenta un nuevo método para la caracterización de estructuras fluidas en la atmósfera terrestre resultantes de la ejecución de simulaciones climáticas.

En una de las primeras etapas del proyecto se presentan diferentes tipos de características basadas en operadores diferenciales y distribuciones de orientaciones (HOOF) que son empleados para alimentar procesos de clasificación basados en técnicas del estado del arte en aprendizaje de máquina (ver sección 3.2.2). Estos métodos se emplearon en la clasificación de las configuraciones del campo de vientos correspondientes a confluencias, difluencias, vorticidades y puntos de silla, en cortes paralelos a la superficie (contemplando las componentes U y V). Esta caracterización resulta satisfactoria, tal como se ve en los resultados de los procesos de clasificación aplicados a los conjuntos de datos anotados por expertos (ver sección 4.3.1).

Los métodos de caracterización general, basados en histogramas de flujo óptico o de la representación piramidal de operadores diferenciales aplicados sobre la divergencia y el rotacional de los campos vectoriales debidos a vientos, entregan resultados satisfactorios en el proceso de clasificación, especialmente para difluencias/confluencias y vorticidades. Los puntos de silla obtuvieron un desempeño más modesto. La precisión promedio que se reporta es de 0,81 que se obtuvo con la representación piramidal en los histogramas de divergencia y rotacional evaluados en una SVM - RBF. Este trabajo se presentó en el XX Congreso Iberoamericano de Reconocimiento

de Patrones¹, (ver Tabla 4.2). Además el clasificador basado en *Random Forest* fue entrenado con 10.000 estimadores, de hecho durante la validación cruzada se pudo determinar que el aumento del número de estimadores no mejoró los resultados.

Por otro lado, en una etapa posterior del proyecto se propuso la caracterización basada en retratos de fase obtuvo resultados significativamente mejores en la evaluación de la clasificación, usando una representación de las estructuras significativamente más simple y obteniendo mejores resultados con métodos de clasificación análogos a los empleados con las representaciones anteriormente descritas. La representación basada en retratos de fase tiene un tamaño de 18 valores mientras que la representación piramidal tiene un tamaño de 10,752, reduciendo en tres órdenes de magnitud el tamaño de la representación, sin emplear todavía métodos multiescala y, sin embargo, mejorando sensiblemente el desempeño de la clasificación. En efecto, la precisión promedio obtenida usando esta aproximación es de 0,89. Además, para la evaluación de desempeño del descriptor basado en retratos de fase se usó una SVM lineal (ver Tabla 4.3), con lo que se muestra que esta representación orientada a incorporar el comportamiento físico subyacente (flujos fluidos) obtiene mejores resultados.

Cabe anotar que la representación basada en retratos de fase usó el gradiente de la velocidad y no la velocidad en sí misma. Esto se debe a que dicho gradiente representa la tasa de cambio en cada una de las direcciones espaciales (dos para este estudio), reflejando de mejor manera la estructura del campo en entornos locales, lo que permite diferenciar de mejor manera la distribución del campo vectorial en las regiones de interés correspondientes a cada clase que se usaron como conjuntos de entrenamiento y prueba (dataset anotado sección 4.2), esta representación se inspiró en los modelos analíticos que se estudian en sistemas dinámicos [54, 43] observándose una clara similitud entre la distribución de orientaciones del retrato de fase y los valores del gradiente para cada una de las componentes (ver Figura 5.1).

A partir de los resultados anteriormente descritos se puede concluir que los datos provenientes de simulaciones atmosféricas/climáticas son sus-

¹Este trabajo se realizó con la colaboración del grupo MindLab, el trabajo se titula: "Classification of Low-Level Atmospheric Structures Based on a Pyramid Representation and a Machine Learning Method" [51] presentado en el CIARP 2015

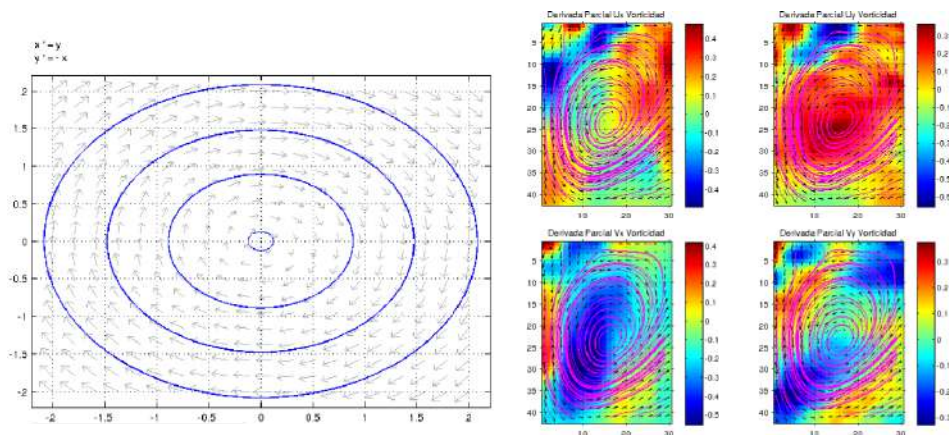


Figura 5.1: Comparación entre el modelo analítico y el gradiente

ceptibles de ser explorados de manera exhaustiva, ofreciendo un uso más completo y detallado que el ofrecido por las técnicas tradicionales Climatología y Meteorología. La caracterización propuesta basada en retratos de fase, principal aportación de esta Tesis, podría soportar métodos eficientes de exploración, con la posibilidad de dar soporte a procesos automáticos y semiautomáticos para la identificación, detección, seguimiento y caracterización dinámica de patrones atmosféricos complejos (conformados por conjuntos de estructuras fluidas) en grandes volúmenes de datos climáticos. Tales procesos serían aplicables, en conjunción con técnicas de análisis de series temporales, a tareas de pronóstico meteorológico y predicción climática.

Trabajo Futuro

La mencionada detección de patrones atmosféricos de *alto nivel* requiere el uso de diferentes variables termodinámicas y físico-químicas que permitan formular representaciones más rigurosas de cada estructura. Sin embargo, se puede anticipar que el viento será siempre una de las variables principales para el análisis. En la Figura 5.2 se presentan diferentes distribuciones espaciales de las variables que describen el estado de la atmósfera (escalares y vectoriales); dichas variables permiten caracterizar distin-

tos fenómenos atmosféricos presentes en los datos a manera de patrones, usualmente identificables por parte de expertos mediante inspección visual (frentes fríos, ondas del este, zonas de confluencia, corrientes de chorro de baja altura, etc.).

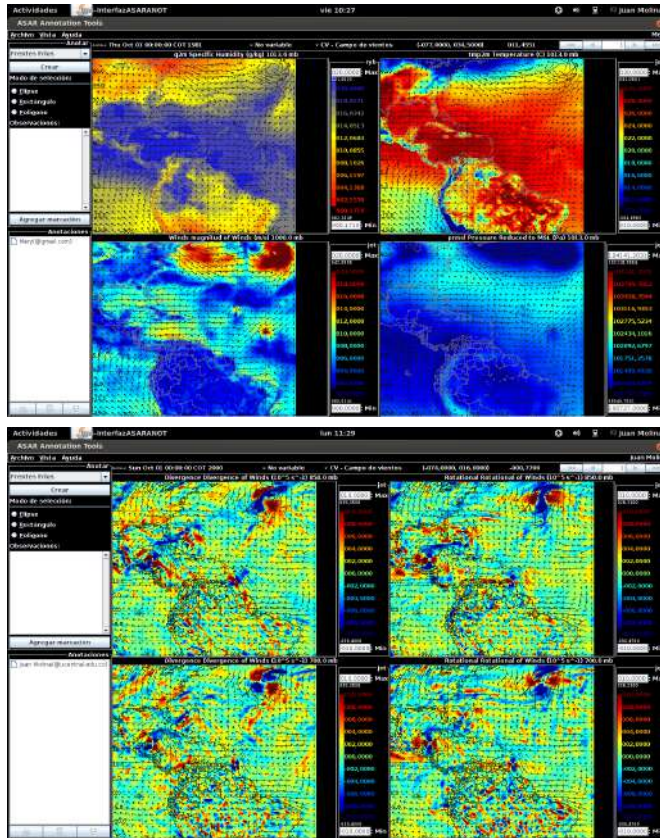


Figura 5.2: Configuración de frentes fríos y ondas del este. La imagen superior es una configuración para detectar frentes fríos donde se observan las variables de humedad específica, temperatura en superficie, velocidad de vientos en superficie y presión a nivel del mar. En la imagen inferior es una configuración para detectar ondas del este donde se observan los operadores diferenciales divergencia y rotacional a 850mb y 700mb.

Como trabajo futuro se incluye robustecer el conjunto de datos y realizar una exploración en la caracterización de estructuras más complejas y que tengan un significado meteorológico. También se busca explorar estrategias para la detección de estas estructuras en grandes volúmenes de información tomando en cuenta que los datos provenientes de simulaciones atmosféricas son susceptibles de ser explorados de manera más exhaustiva y detallada mediante técnicas de detección y clasificación que se adapten al problema.

Bibliografía

- [1] G. Armenta and J.D. Pabón. Modeling Northern South America and Caribbean climate using PRECIS and WRF for climate variability and change studies. In *Proceedings of the CORDEX-LAC1 Workshop – World Climate Research Programme*, Lima, Peru, 2013.
- [2] Javed A Aslam, Raluca A Popa, and Ronald L Rivest. On estimating the size and confidence of a statistical audit. In *Proc. 2007 USENIX/ACCURATE Electronic Voting Technology Workshop (EVT'07)*, 2007.
- [3] JL Barron and NA Thacker. Tutorial: Computing 2d and 3d optical flow. *Imaging Science and Biomedical Engineering Division, Medical School, University of Manchester*, 2005.
- [4] Marcelo Bertalmio, Andrea L Bertozzi, and Guillermo Sapiro. Navier-stokes, fluid dynamics, and image and video inpainting. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 1, pages I–355. IEEE, 2001.
- [5] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.
- [6] Roxana Bujack, Ingrid Hotz, Gerik Scheuermann, and Eckhard Hitzler. Moment invariants for 2d flow fields using normalization. In *Pacific Visualization Symposium (PacificVis), 2014 IEEE*, pages 41–48. IEEE, 2014.

- [7] Andrés W Burgoa Mariaca. Escenarios climáticos para el monzón sudamericano: determinados por los modelos de acoplamiento del ipcc ar4. *Revista Boliviana de Física*, 14(14):100–115, 2008.
- [8] R. Chaudhry, A. Ravichandran, G. Hager, and R. Vidal. Histograms of oriented Optical Flow and Binet-Cauchy kernels on nonlinear dynamical systems for the recognition of human actions. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1932–1939, June 2009.
- [9] P Cheung and HY Yeung. Application of optical-flow technique to significant convection nowcast for terminal areas in hong kong. In *The 3rd WMO International Symposium on Nowcasting and Very Short-Range Forecasting (WSN12)*, pages 6–10, 2012.
- [10] Thomas Corpetti, Etienne Memin, and Patrick Pérez. Extraction of singular points from dense motion fields: an analytic approach. *Journal of mathematical imaging and vision*, 19(3):175–198, 2003.
- [11] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.
- [12] Carlos Costa Posada and Instituto de Hidrologia Meterologia y Estudios Ambientales-Ministerio de Ambiente Vivienda y Desarrollo Territorial. Atlas climatologico de colombia. 2005.
- [13] Nuwanthika Fernando and Upul Sonnadara. Wind field analysis using optical flow and fuzzy logic. In *Proceedings of Technical Sessions*, volume 29, pages 71–76, 2013.
- [14] John C Fyfe, Nathan P Gillett, and Francis W Zwiers. Overestimated global warming over the past 20 years. *Nature Climate Change*, 3(9):767–769, 2013.
- [15] Chris Gordon, Claire Cooper, Catherine A Senior, Helene Banks, Jonathan M Gregory, Timonthy C Johns, John FB Mitchell, and Richard A Wood. The simulation of sst, sea ice extents and ocean heat transports in a version of the hadley centre coupled model without flux adjustments. *Climate dynamics*, 16(2-3):147–168, 2000.

- [16] Amara Graps. An introduction to wavelets. *Computational Science & Engineering, IEEE*, 2(2):50–61, 1995.
- [17] Kenny Gruchalla, Mark Rast, Elizabeth Bradley, John Clyne, and Pablo Mininni. Visualization-driven structural and statistical analysis of turbulent flows. In *Advances in Intelligent Data Analysis VIII*, pages 321–332. Springer, 2009.
- [18] J. F. Cadena M. Guzmán D., Ruíz. Regionalización de colombia según la estacionalidad de la precipitación media mensual, a través análisis de componentes principales (acp). 2014.
- [19] Chih-Wei Hsu, Chih-Chung Chang, Chih-Jen Lin, et al. A practical guide to support vector classification, 2003.
- [20] James M Hyman and Mikhail Shashkov. Natural discretizations for the divergence, gradient, and curl on logically rectangular grids. *Computers & Mathematics with Applications*, 33(4):81–104, 1997.
- [21] Mark Z Jacobson. *Fundamentals of atmospheric modeling*. Cambridge university press, 2005.
- [22] Heike Janicke, Michael Bottinger, Uwe Mikolajewicz, and Gerik Scheuermann. Visual exploration of climate variability changes using wavelet analysis. *Visualization and Computer Graphics, IEEE Transactions on*, 15(6):1375–1382, 2009.
- [23] Richard Jones. *Generating high resolution climate change scenarios using PRECIS*. Met office, 2004.
- [24] Richard W Katz, Peter F Craigmile, Peter Guttorp, Murali Haran, Bruno Sansó, and Michael L Stein. Uncertainty analysis in climate change assessments. *Nature Climate Change*, 3(9):769–771, 2013.
- [25] A Laing and JL Evans. Introduction to tropical meteorology, a comprehensive online and print textbook, 2010.
- [26] S. Lazebnik, C. Schmid, and J. Ponce. Beyond Bags of Features: Spatial Pyramid Matching for recognizing natural scene categories.

- In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 2, pages 2169–2178, 2006.
- [27] GE León, JA Zea, and JA Eslava. Ondas del este en colombia y algunos aspectos relevantes de los ciclones tropicales. *Meteorologia Colombiana*, (3):127–141, 2001.
- [28] Carlos HR Lima, Upmanu Lall, Tony Jebara, and Anthony G Barnston. Statistical prediction of enso from subsurface sea temperature using a nonlinear dimensionality reduction. *Journal of Climate*, 22(17):4501–4519, 2009.
- [29] Frederick K Lutgens, Edward J Tarbuck, and Dennis Tasa. *The atmosphere: an introduction to meteorology*. Prentice Hall New Jersey, 1998.
- [30] JA Marengo, R Jones, LM Alves, and MC Valverde. Future change of temperature and precipitation extremes in south america as derived from the precis regional climate modeling system. *international Journal of Climatology*, 29(15):2241–2255, 2009.
- [31] Beatriz Martínez and María Amparo Gilabert. Vegetation dynamics from ndvi time series analysis using the wavelet transform. *Remote Sensing of Environment*, 113(9):1823–1842, 2009.
- [32] Robert Mendelsohn, Pradeep Kurukulasuriya, Alan Basist, Felix Kogan, and Claude Williams. Climate analysis with satellite versus weather station data. *Climatic Change*, 81(1):71–83, 2007.
- [33] Francis John Monkhouse. *Diccionario de términos geográficos*. Oikos-tau, 1978.
- [34] Claire Monteleoni, Gavin A Schmidt, Shailesh Saroha, and Eva Asplund. Tracking climate models. *Statistical Analysis and Data Mining*, 4(4):372–392, 2011.
- [35] Gy Palomino Montoya. R. 2005. sistemas pluviogénéticos en Colombia: influencia de frentes fríos del hemisferio norte. *Meteorol. Colomb*, 9:75–82.

- [36] Ileana Mora and Jorge A. Amador. El ENSO, el IOS y la corriente en chorro de bajo nivel en el oeste del Caribe. *Tóp. Meteor. Oceanogr*, 7:1–20, 2000.
- [37] J. Nagi, F. Ducatelle, G.A. Di Caro, D. Ciresan, U. Meier, A. Giusti, F. Nagi, J. Schmidhuber, and L.M. Gambardella. Max-pooling convolutional neural networks for vision-based hand gesture recognition. In *Signal and Image Processing Applications (ICSIPA), 2011 IEEE International Conference on*, pages 342–347, Nov 2011.
- [38] Jeremy S Pal, Filippo Giorgi, Xunqiang Bi, Nellie Elguindi, Fabien Solomon, Sara A Rauscher, Xuejie Gao, Raquel Francisco, Ashraf Zakey, Jonathan Winter, et al. Regional climate modeling for the developing world: the ictp regcm3 and regcnet. *Bulletin of the American Meteorological Society*, 88(9):1395–1409, 2007.
- [39] Jeremy S Pal, Eric E Small, and Elfatih AB Eltahir. Simulation of regional-scale water and energy budgets: Representation of subgrid cloud and precipitation processes within regcm. *Journal of Geophysical Research: Atmospheres (1984–2012)*, 105(D24):29579–29594, 2000.
- [40] José P Peixoto and Abraham H Oort. *Physics of climate*. 1992.
- [41] Germán Poveda. La hidroclimatología de colombia: una síntesis desde la escala inter-decadal hasta la escala diaria. *Rev. Acad. Colomb. Cienc*, 28(107):201–222, 2004.
- [42] Germán Poveda and Oscar J Mesa. La corriente del chorro superficial del oeste, “Chorro del Chocó” y otras dos corrientes de chorro en colombia: climatología y variabilidad durante las fases del enso. *Revista Academia Colombiana de Ciencias*, 23:517–528, 1999.
- [43] A. Ravishankar Rao and Ramesh C Jain. Computerized flow field analysis: Oriented texture fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(7):693–709, 1992.
- [44] Payam Refaeilzadeh, Lei Tang, and Huan Liu. Cross-validation. In *Encyclopedia of database systems*, pages 532–538. Springer, 2009.

- [45] JF Ruiz. Cambio climático en temperatura, precipitación y humedad relativa para colombia usando modelos meteorológicos de alta resolución, 2011.
- [46] Suranjana Saha, Shrinivas Moorthi, Hua-Lu Pan, Xingren Wu, Jiande Wang, Sudhir Nadiga, Patrick Tripp, Robert Kistler, John Woollen, David Behringer, et al. The ncep climate forecast system reanalysis. *Bulletin of the American Meteorological Society*, 91(8):1015–1057, 2010.
- [47] Toshihiro Sakamoto, Masayuki Yokozawa, Hitoshi Toritani, Michio Shibayama, Naoki Ishitsuka, and Hiroyuki Ohno. A crop phenology detection method using time-series modis data. *Remote sensing of environment*, 96(3):366–374, 2005.
- [48] Murry L Salby. *Physics of the Atmosphere and Climate*. Cambridge University Press, 2012.
- [49] Bernhard Schölkopf, Kah-Kay Sung, Chris JC Burges, Federico Girosi, Partha Niyogi, Tomaso Poggio, and Vladimir Vapnik. Comparing support vector machines with gaussian kernels to radial basis function classifiers. *Signal Processing, IEEE Transactions on*, 45(11):2758–2765, 1997.
- [50] M. H. Shimizu and T. Ambrizzi. Influence of ENSO and MJO Phases on Extreme Events of Precipitation over Amazon. *AGU Fall Meeting Abstracts*, December 2014.
- [51] Sebastián Sierra, Juan F Molina, Angel Cruz-Roa, José Daniel Pabón, Raúl Ramos-Pollán, Fabio A González, and Hugo Franco. Classification of low-level atmospheric structures based on a pyramid representation and a machine learning method. In *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*, pages 19–26. Springer, 2015.
- [52] William C Skamarock, Joseph B Klemp, Jimy Dudhia, David O Gill, Dale M Barker, Wei Wang, and Jordan G Powers. A description of the advanced research wrf version 2. Technical report, DTIC Document, 2005.

- [53] Karsten Steinhaeuser, Nitesh V Chawla, and Auroop R Ganguly. An exploration of climate data using complex networks. *ACM SIGKDD Explorations Newsletter*, 12(1):25–32, 2010.
- [54] Steven H Strogatz. *Nonlinear dynamics and chaos: with applications to physics, biology, chemistry, and engineering (studies in nonlinearity)*. 2001.
- [55] JI Diaz L Tello. Sobre un modelo climatico de balance de energia superficial acoplado con un oceano profundo. *Actas del XVII CED-YA/VI CMA*, 2001.
- [56] Jiande Wang, Wanqiu Wang, Xiouhua Fu, and Kyong-Hwan Seo. Tropical intraseasonal rainfall variability in the CFSR. *Climate dynamics*, 38(11-12):2191–2207, 2012.
- [57] Simon Wilson, David Hassell, David Hein, Richard Jones, and Ruth Taylor. Installing and using the hadley centre regional climate modelling system, precis. *Version*, 1(1):157, 2008.

Apéndice A

Proceso de extracción de datos

Se realizó un proceso de extracción de datos que permitió extraer las variables de interés de un gran volumen de datos proveniente de una simulación WRF

A.1. Extracción de Variables

Se realizó un script en MATLAB para extraer estas variables

```
function wrf_extract(wrf_file, mainpath)
%% wrf Extract data
% wrf_file = file ntcdff with the simulation
% mainpath = locate of the file ntcdff
%

% Generate structure

mkdir(mainpath, '/simulation'); % puede ser el nombre del archivo
de salida wrf
mainpath = [mainpath '/simulation']; % puede ser el nombre del
archivo de salida wrf
mkdir(mainpath, '/simulation_Climate_data');
mkdir(mainpath, '/simulation_Studies');
```

```

mainpath;
wrf_file1=char(wrf_file)
creation_date = ncreadatt(wrf_file1,'/', 'START_DATE');
creation_dates=strprep(creation_date,',' ,'-');

eval(['!ls -l --time-style="+%Y-%m-%d %H-%M" ' wrf_file ' >datas
']);
f=fopen('datas');
fa=fread(f,123,'uint8=>char');
d=fa;
for i=1:9
[s d]=strtok(d);
da{i}=s;
end
fclose(f);
eval('!rm datas');
file_date = da{6}'
mainpath = [mainpath '/simulation_Climate_data/'];
mkdir(mainpath, [creation_dates '_climate_' file_date]);
mainpath = [mainpath creation_dates '_climate_' file_date];

path_t = [mainpath '/T'];
path_p = [mainpath '/P'];
path_wind = [mainpath '/Winds'];
path_pr = [mainpath '/Prate'];
path_rh = [mainpath '/RelHum'];
path_cf = [mainpath '/CloudFrac'];
% create the folders of the variables
mkdir(mainpath, '/T');
mkdir(mainpath, '/P');
mkdir(mainpath, '/Winds');
%mkdir(mainpath, '/Prate');
mkdir(mainpath, '/RelHum');
mkdir(mainpath, '/CloudFrac');

% Produce file dimensions
tmp = ncread(wrf_file, 'XLAT');
[nlon nlat ntime] = size(tmp);
tmp = ncread(wrf_file, 'ZNU');
nlev = size(tmp, 1);

```

```

disp('dimensions:')
disp(' Longs, Lats, levels, times')
disp([nlon nlat nlev ntime])
% write latitude in the file lats.dat
lats = ncread(wrf_file, 'XLAT', [1 1 1], [1 nlat 1]);
disp(['num_latitudes      = ' int2str(nlat)])
disp(['min_range_latitudes = ' num2str(min(lats))])
disp(['max_range_latitudes = ' num2str(max(lats))])
% write longitud in the file lons.dat
lons = ncread(wrf_file, 'XLONG', [1 1 1], [nlon 1 1]);
disp(['num_longitudes     = ' int2str(nlon)])
disp(['min_range_longitudes = ' num2str(min(lons))])
disp(['max_range_longitudes = ' num2str(max(lons))])
% write the number of levels in the file levs.dat
levs = ncread(wrf_file, 'ZNU', [1 1], [nlev 1]);
disp(['num_levels        = ' int2str(nlev)])
disp(['min_range_levels   = ' num2str(min(levs))])
disp(['max_range_levels   = ' num2str(max(levs))])
% write the number of times in the file times.dat
xtimes = ncread(wrf_file, 'Times');
xtimes = xtimes';
disp(['num_times          = ' int2str(ntime)])
disp(['min_range_times    = ' num2str(min(xtimes))])
disp(['max_range_times    = ' num2str(max(xtimes))])

% write latitude in the file latsU.dat
lats_U = ncread(wrf_file,'XLAT_U',[1 1 1],[1 (nlat) 1]);
lons_U = ncread(wrf_file,'XLONG_U',[1 1 1],[(nlon+1) 1 1]);
lats_V = ncread(wrf_file,'XLAT_V',[1 1 1],[1 (nlat+1) 1]);
lons_V = ncread(wrf_file,'XLONG_V',[1 1 1],[(nlon) 1 1]);
levs_W = ncread(wrf_file,'ZNV',[1 1],[nlev+1 1]);

num_lons=nlon;
num_lats=nlat;
num_levs=nlev;

maximoT=zeros(1,ntime);
minimoT=zeros(1,ntime);
maximoP=zeros(1,ntime);
minimoP=zeros(1,ntime);
maximoU=zeros(1,ntime);

```

```

minimoU=zeros(1,ntime);
maximoV=zeros(1,ntime);
minimoV=zeros(1,ntime);
maximoW=zeros(1,ntime);
minimoW=zeros(1,ntime);
maximorelatHum=zeros(1,ntime);
minimorelatHum=zeros(1,ntime);
maximocloudFrac=zeros(1,ntime);
minimocloudFrac=zeros(1,ntime);
for i=1:ntime

T = ncread(wrf_file1,'T',[1 1 1 i],[num_lons num_lats num_levs 1])
    ;% Temperature
P = ncread(wrf_file1,'P',[1 1 1 i],[num_lons num_lats num_levs 1])
    ;% Presion
PB = ncread(wrf_file1,'PB',[1 1 1 i],[num_lons num_lats num_levs
1]);%
U = ncread(wrf_file1,'U',[1 1 1 i],[num_lons+1 num_lats num_levs
1]);% U winds
V = ncread(wrf_file1,'V',[1 1 1 i],[num_lons num_lats+1 num_levs
1]);% V winds
W = ncread(wrf_file1,'W',[1 1 1 i],[num_lons num_lats num_levs+1
1]);% W winds
Q2= ncread(wrf_file,'Q2',[1 1 i],[num_lons num_lats 1]);%
t2= ncread(wrf_file,'T2',[1 1 i],[num_lons num_lats 1]);%
psfc= ncread(wrf_file,'PSFC',[1 1 i],[num_lons num_lats 1]);%
CloudFrac=ncread(wrf_file,'CLDFRA',...
[1 1 1 i],[num_lons num_lats num_levs 1]);%

% pq0
% Create files with temperature data
tp = (T+300).*((PB+P)/100000).^ (2/7); % temperature formula

maximoT(i)=max(max(max(tp)));
minimoT(i)=min(min((min(tp))));
str = [path_t '/' 'T' int2str(i) '.dat'];
pf = fopen(str, 'wb');
fwrite(pf, tp, 'single');
fclose(pf);
% Create files with presion data

```

```

p=P+PB; % Presion Formula
maximoP(i)=max(max(max(p)));
minimoP(i)=min(min(min(p)));
str = [path_p '/' 'P' int2str(i) '.dat'];
pf = fopen(str, 'wb');
fwrite(pf, p, 'single');
fclose(pf);
maximoU(i)=max(max(max(U)));
maximoV(i)=max(max(max(V)));
maximoW(i)=max(max(max(W)));

minimoU(i)=min(min(min(U)));
minimoV(i)=min(min(min(V)));
minimoW(i)=min(min(min(W)));

str = [path_wind '/' 'U' int2str(i) '.dat'];
pf = fopen(str, 'wb');
fwrite(pf, U, 'single');
fclose(pf);
%
str = [path_wind '/' 'V' int2str(i) '.dat'];
pf = fopen(str, 'wb');
fwrite(pf, V, 'single');
fclose(pf);
%
str = [path_wind '/' 'W' int2str(i) '.dat'];
pf = fopen(str, 'wb');
fwrite(pf, W, 'single');
fclose(pf);
pq0 = 379.90516;
a2 = 17.2693882;
a3 = 273.16;
a4 = 35.86;
h=(Q2./ ( pq0 ./ psfc) .* exp(a2 * (t2 - a3) ./ (t2 - a4)))) ;
str = [path_rh '/' 'RelHum' int2str(i) '.dat'];
pf = fopen(str, 'wb');
fwrite(pf, h, 'single');
fclose(pf);
maximorelatHum(i)=max(max(max(h)));
minimorelatHum(i)=min(min(min(h)));

```

```

str = [path_cf '/' 'CloudFrac' int2str(i) '.dat'];
pf = fopen(str, 'wb');
fwrite(pf, CloudFrac, 'single');
fclose(pf);

maximocloudFrac(i)=max(max(max(CloudFrac)));
minimocloudFrac(i)=min(min(min(CloudFrac)));

end
%% Calcula el maximo global de todos
maxT=max(maximoT)
minT=min(minimoT)
maxP=max(maximoP)
minP=min(minimoP)
maxU=max(maximoU)
minU=min(minimoU)
maxV=max(maximoV)
minV=min(minimoV)
maxW=max(maximoW)
minW=min(minimoW)
maxRH=max(maximorelatHum)
minRH=min(minimorelatHum)
maxCF=max(maximocloudFrac)
minCF=min(maximocloudFrac)

Maximum=[maxT maxP maxRH maxCF maxU maxV maxW ];
Minimum=[minT minP minRH minCF minU minV minW ];

metadatas(Maximum,Minimum,lons,lats,levs,lons_U,lats_U,lons_V,
    lats_V,levs_W,xtimes,{'T','P','Q2','CLDFRA','U','V','W'},[
    creation_dates '_climate_' file_date],mainpath,wrf_file);

```

A.2. Creación de Metadatos

Debido a la duplicidad de información se creo un nuevo de archivo de metadatos con la información global necesaria que permitiera representar las variables anteriormente extraídas

```
function metadatas(maximum,minimum,lons,lats,levs,lons_U,...
```



```

lats_U,lons_V,lats_V,levs_W,times,variables,name,namefile,wrf_file)
% parametros de entrada
% maximum ---- Vector de valores maximos por variable
% minimum ---- Vector de valores minimos por variable
% lons ---- Vector de longitudes
% lats ---- Vector de latitudes
% times---- Vector de tiempos
% levs ---- Vector de niveles
% variables ---- Cell de variables de estudio
% name ---- Nombre del estudio
% namefile --- ubicacion del archivo
% wrf_file --- archivo WRF
% salida es la generacion de un archivo XML
%% Creacion de XML
docNode = com.mathworks.xml.XMLUtils.createDocument('meta');
docRootNode = docNode.getDocumentElement;
docRootNode.setAttribute('simulation',name);
%thisExl=docNode.createElement('dimensions');
ncid = netcdf.open(wrf_file,'NC_NOWRITE');
dimensiones=cell(2,9);
for j=1:9
[diname,dimlen]=netcdf.inqDim(ncid,j-1);
dimensiones{1,j}=diname;
dimensiones{2,j}=dimlen;
end

dimensiones{1,3}='Longitude';
dimensiones{1,4}='Latitude';
dimensiones{1,5}='Level';
dimensiones{1,6}='Level_W';
dimensiones{1,8}='Longitude_U';
dimensiones{1,9}='Latitude_V';

%% Convertir vector numerico a cadena de caracteres
cadenalons=[];

for j=1:length(lons)
cadenalons=[cadenalons,' ',num2str(lons(j))];
end

cadenalats=[];

```

```
for j=1:length(lats)
cadenalats=[cadenalats,' ',num2str(lats(j))];
end

cadenalevs=[];
for j=1:length(levs)
cadenalevs=[cadenalevs,' ',num2str(levs(j))];
end
cadenatimes=[];

for j=1:length(times)
st=times(j,:);
cadenatimes=[cadenatimes,' ',times(j,:)];
end

cadenalonsU=[];

for j=1:length(lons_U)
cadenalonsU=[cadenalonsU,' ',num2str(lons_U(j))];
end

cadenalonsV=[];
for j=1:length(lons_V)
cadenalonsV=[cadenalonsV,' ',num2str(lons_V(j))];
end

cadenalatsU=[];
for j=1:length(lats_U)
cadenalatsU=[cadenalatsU,' ',num2str(lats_U(j))];
end

cadenalatsV=[];
for j=1:length(lats_V)
cadenalatsV=[cadenalatsV,' ',num2str(lats_V(j))];
end

cadenalevsW=[];
for j=1:length(levs_W)
```

```

cadenalevsW=[cadenalevsW,' ',num2str(levs_W(j))];
end
%% Crea los atributos para las dimensiones
thisElement = docNode.createElement('Dimensions');

docRootNode.appendChild(thisElement);
for i=1:9
if i == 2 || i == 7
disp(i);
else

thisEl= docNode.createElement(dimensiones{1,i});

thisElx=docNode.createElement('value');
thisElx.appendChild...
(docNode.createTextNode(sprintf('%s',num2str(dimensiones{2,i})))));
thisEl.appendChild(thisElx);
%
thisElx=docNode.createElement('data');
if i==1
thisElx.appendChild...
(docNode.createTextNode(sprintf('%s',cadenatimes)));
end
if i==3
thisElx.appendChild...
(docNode.createTextNode(sprintf('%s',cadenalons)));
end
if i==4
thisElx.appendChild...
(docNode.createTextNode(sprintf('%s',cadenalats)));
end
if i==5
thisElx.appendChild...
(docNode.createTextNode(sprintf('%s',cadenalevs)));
end
if i==6
thisElx.appendChild...
(docNode.createTextNode(sprintf('%s',cadenalevsW)));
end
if i==8
thisElx.appendChild...

```

```

(docNode.createTextNode(sprintf('%s',cadenalonsU)));
end
if i==9
thisElx.appendChild...
(docNode.createTextNode(sprintf('%s',cadenalatsV)));
end
thisEl.appendChild(thisElx);

thisElement.appendChild(thisEl);
end
end
netcdf.close(ncid);

thisElement = docNode.createElement('Palettes');
nombredelaspaletas={'jet','hsv','hot','cool','spring',...
'summer','autumn','bone','winter','copper','pink'};
thisEl= docNode.createElement('Palette');
thisElement.appendChild(thisEl);
thisEl1=docNode.createElement('name');
thisEl1.appendChild...
(docNode.createTextNode(sprintf('%s',' ')));
thisEl.appendChild(thisEl1);
thisEl1=docNode.createElement('Size');
thisEl1.appendChild...
(docNode.createTextNode(sprintf('%s',' ')));
thisEl.appendChild(thisEl1);
thisEl1=docNode.createElement('component_size');
thisEl1.appendChild...
(docNode.createTextNode(sprintf('%s',' ')));
thisEl.appendChild(thisEl1);
thisEl1=docNode.createElement('data');
thisEl1.appendChild...
(docNode.createTextNode(sprintf('%s',' ')));
thisEl.appendChild(thisEl1);
%
thisElement.appendChild(thisEl);
%
docRootNode.appendChild(thisElement);
%
% end

```

```
%% Crea los atributos para las variables
thisElement = docNode.createElement('Variables');
docRootNode.appendChild(thisElement);

for i=1:length(variables)
var=ncinfo(wrf_file,variables{i});
% Cambiar el nombre de la variable
if strcmp('Q2',variables{i})
variables{i}='RelHum';
end
if strcmp('CLDFRA',variables{i})
variables{i}='CloudFrac';
end
thisEl= docNode.createElement(variables{i});

thisEl1=docNode.createElement('Palette');

thisEl1.appendChild(docNode.createTextNode(sprintf('%s', ...
nombredelaspaletas{i})));

thisEl.appendChild(thisEl1);

thisEl1=docNode.createElement('dimension_name');
% If para verificar si posee 4 o 3 dimensiones
if length(var.Dimensions)==4

[name1,name2,name3,name4]=var.Dimensions.Name;
else
[name1, name2, name4]=var.Dimensions.Name;
name3='Level';

end
% If para cambiar los atributos del WRF a atributos propios
if(strcmp(name1,'west_east'))
name1='Longitude';
end
```

```

if(strcmp(name1,'west_east_stag'))
name1='Longitude_U';
end
if(strcmp(name2,'south_north'))
name2='Latitude';
end
if(strcmp(name2,'south_north_stag'))
name2='Latitude_V';
end
if(strcmp(name3,'bottom_top'))
name3='Level';
end
if(strcmp(name3,'bottom_top_stag'))
name3='Level_W';
end

thisEl1.appendChild...
(docNode.createTextNode(sprintf('%s',[name1 ' ' name2 ' ' name3...
' ' name4])));
thisEl.appendChild(thisEl1);
% Crea el tamaño que posee los datos
thisEl1=docNode.createElement('Size');
% if para mirar la longitud de var.Size si este posee 3 componentes
se
% debe agregar un 1 para que quede en términos de 4D
if length(var.Size)==3
valores=[num2str(var.Size(1)) ' ' num2str(var.Size(2))...
' ' num2str(1) ' ' num2str(var.Size(3))];
thisEl1.appendChild...
(docNode.createTextNode(sprintf('%s',valores)));
else
thisEl1.appendChild...
(docNode.createTextNode(sprintf('%s',num2str(var.Size))));

end
thisEl.appendChild(thisEl1);
% if para determinar si la variable es Times
if strcmp(variables{i},'Times')
else

thisEl1= docNode.createElement(var.Attributes(3).Name);

```

```

if strcmp('RelHum',variables{i})||strcmp('P',variables{i})...
||strcmp('T',variables{i})
if strcmp('RelHum',variables{i})
rhdescription='Relativity Humidity';
thisEl1.appendChild...
(docNode.createTextNode(sprintf('%s',rhdescription)));
end
if strcmp('P',variables{i})
rhdescription='Total Pressure';
thisEl1.appendChild...
(docNode.createTextNode(sprintf('%s',rhdescription)));
end
if strcmp('T',variables{i})
rhdescription='Total Potential Temperature';
thisEl1.appendChild...
(docNode.createTextNode(sprintf('%s',rhdescription)));
end
else
thisEl1.appendChild...
(docNode.createTextNode(sprintf('%s',var.Attributes(3).Value)));
end

thisEl.appendChild(thisEl1);
%
thisEl1= docNode.createElement(var.Attributes(4).Name);
if strcmp('',var.Attributes(4).Value)
thisEl1.appendChild...
(docNode.createTextNode(sprintf('%s',' ')));
else
thisEl1.appendChild...
(docNode.createTextNode(sprintf('%s',var.Attributes(4).Value)));
end
thisEl.appendChild(thisEl1);

%
thisEl1= docNode.createElement('Minimum');

thisEl1.appendChild...
(docNode.createTextNode(sprintf('%f',(minimum(i)))));

thisEl.appendChild(thisEl1);

```

```
%
thisEl1= docNode.createElement('Maximum');

thisEl1.appendChild...
(docNode.createTextNode(sprintf('%f', (maximum(i)))));

thisEl.appendChild(thisEl1);
end

thisElement.appendChild(thisEl);
end

xmlFileName = [namefile, '/', 'metadataWRF', '.xml'];
xmlwrite(xmlFileName, docNode);
type(xmlFileName);
end
```


Apéndice B

Documentación de la herramienta de anotaciones

Para este proyecto se desarrollo una herramienta de anotaciones para la selección de regiones de interés de las variables atmosféricas, estas regiones de interés serán representadas como coordenadas del volumen y almacenadas en un archivo XML para su experimentación

B.1. Casos de uso



Figura B.1: Diagrama de Casos de Uso

B.2. Diagrama de clases

En esta subsección se muestra los diagramas de clase que permitieron construir la herramienta de anotaciones

B.2.1. Diagrama de clases: Modelo del negocio

Modelo de Negocio de la herramienta de anotaciones

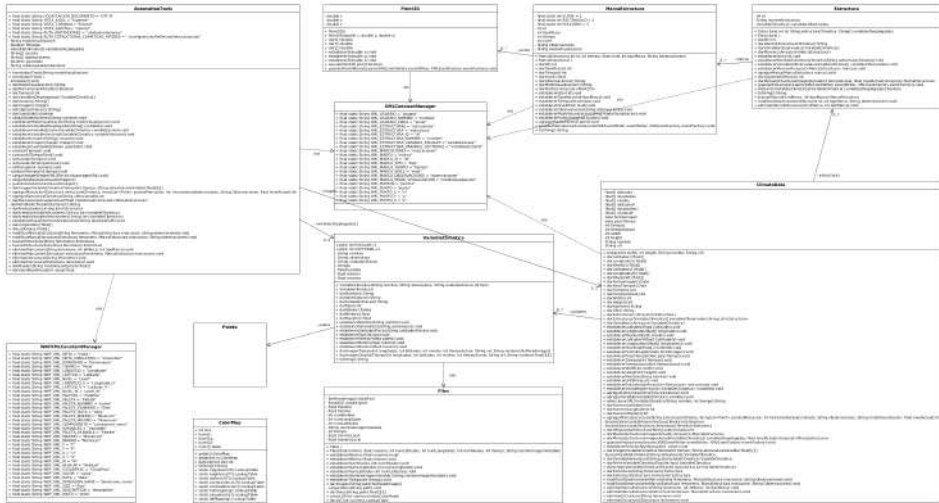


Figura B.2: Diagrama de Clases del Modelo del negocio

B.3. Diagrama de actividades

En esta subsección se muestra el diagramas de actividades que permitieron construir la herramienta de anotaciones

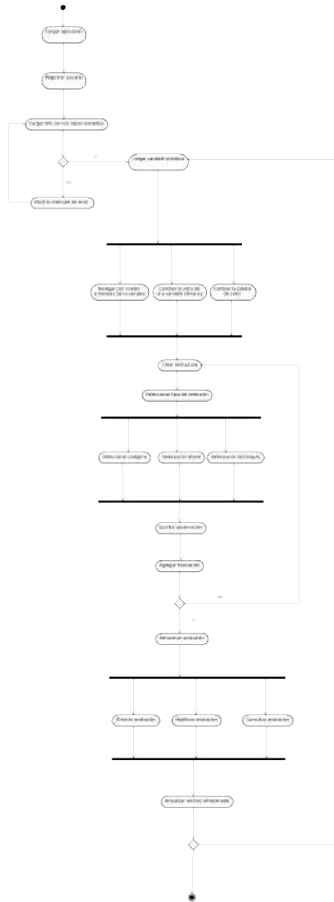


Figura B.4: Diagrama de actividades

B.4. Diagrama de secuencias

En esta subsección se muestra los diagramas de secuencia que permitieron construir la herramienta de anotaciones

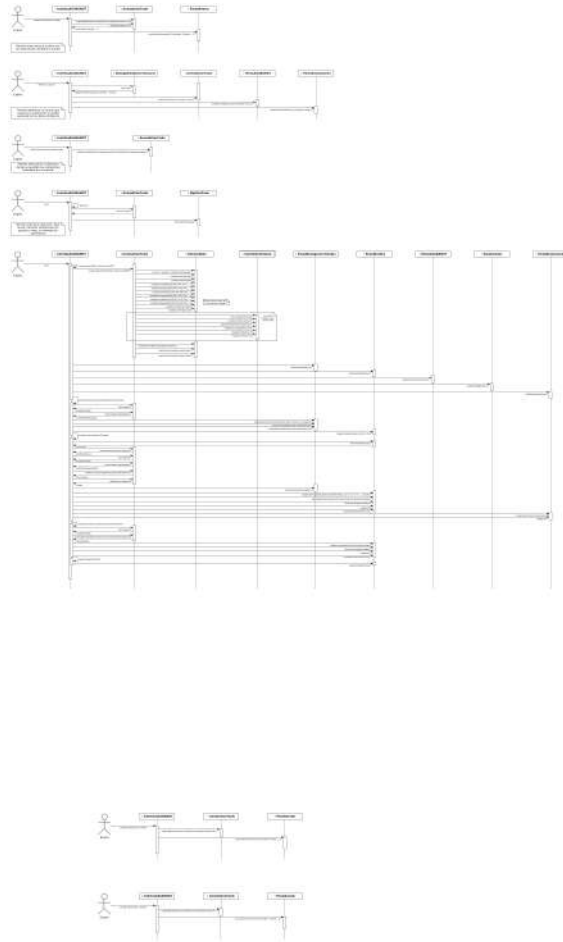


Figura B.5: Diagrama de secuencias

B.5. Formato del archivo de entrada (XML)

Se debe cargar este archivo de metadatos para hacer funcionar la herramienta de anotaciones

```
<?xml version="1.0" encoding="utf-8"?>
<meta simulation="2004-04-29_00-00-00_climate_2014-09-29">
  <Dimensions>
    <Time>
      <value>273</value>
      <data> 2004-04-29_00:00:00 2004-04-29_03:00:00 2004-04-29
        _06:00:00 2004-04-29_09:00:00
        2004-04-29_12:00:00 2004-04-29_15:00:00 2004-04-29_18
          :00:00 2004-04-29_21:00:00 2004-04-30_00:00:00
        2004-04-30_03:00:00 2004-04-30_06:00:00 2004-04-30_09
          :00:00 2004-04-30_12:00:00 2004-04-30_15:00:00
        2004-04-30_18:00:00 2004-04-30_21:00:00 2004-05-01_00
          :00:00 2004-05-01_03:00:00 2004-05-01_06:00:00
        2004-05-01_09:00:00 2004-05-01_12:00:00 2004-05-01_15
          :00:00 2004-05-01_18:00:00 2004-05-01_21:00:00
        2004-05-02_00:00:00 2004-05-02_03:00:00 2004-05-02_06
          :00:00 2004-05-02_09:00:00 2004-05-02_12:00:00
        2004-05-02_15:00:00 2004-05-02_18:00:00 2004-05-02_21
          :00:00 2004-05-03_00:00:00 2004-05-03_03:00:00
        2004-05-03_06:00:00 2004-05-03_09:00:00 2004-05-03_12
          :00:00 2004-05-03_15:00:00 2004-05-03_18:00:00
        2004-05-03_21:00:00 2004-05-04_00:00:00 2004-05-04_03
          :00:00 2004-05-04_06:00:00 2004-05-04_09:00:00
        2004-05-04_12:00:00 2004-05-04_15:00:00 2004-05-04_18
          :00:00 2004-05-04_21:00:00 2004-05-05_00:00:00
        2004-05-05_03:00:00 2004-05-05_06:00:00 2004-05-05_09
          :00:00 2004-05-05_12:00:00 2004-05-05_15:00:00
        2004-05-05_18:00:00 2004-05-05_21:00:00 2004-05-06_00
          :00:00 2004-05-06_03:00:00 2004-05-06_06:00:00
        2004-05-06_09:00:00 2004-05-06_12:00:00 2004-05-06_15
          :00:00 2004-05-06_18:00:00 2004-05-06_21:00:00
        2004-05-07_00:00:00 2004-05-07_03:00:00 2004-05-07_06
          :00:00 2004-05-07_09:00:00 2004-05-07_12:00:00
        2004-05-07_15:00:00 2004-05-07_18:00:00 2004-05-07_21
          :00:00 2004-05-08_00:00:00 2004-05-08_03:00:00
```

2004-05-08_06:00:00 2004-05-08_09:00:00 2004-05-08_12
:00:00 2004-05-08_15:00:00 2004-05-08_18:00:00
2004-05-08_21:00:00 2004-05-09_00:00:00 2004-05-09_03
:00:00 2004-05-09_06:00:00 2004-05-09_09:00:00
2004-05-09_12:00:00 2004-05-09_15:00:00 2004-05-09_18
:00:00 2004-05-09_21:00:00 2004-05-10_00:00:00
2004-05-10_03:00:00 2004-05-10_06:00:00 2004-05-10_09
:00:00 2004-05-10_12:00:00 2004-05-10_15:00:00
2004-05-10_18:00:00 2004-05-10_21:00:00 2004-05-11_00
:00:00 2004-05-11_03:00:00 2004-05-11_06:00:00
2004-05-11_09:00:00 2004-05-11_12:00:00 2004-05-11_15
:00:00 2004-05-11_18:00:00 2004-05-11_21:00:00
2004-05-12_00:00:00 2004-05-12_03:00:00 2004-05-12_06
:00:00 2004-05-12_09:00:00 2004-05-12_12:00:00
2004-05-12_15:00:00 2004-05-12_18:00:00 2004-05-12_21
:00:00 2004-05-13_00:00:00 2004-05-13_03:00:00
2004-05-13_06:00:00 2004-05-13_09:00:00 2004-05-13_12
:00:00 2004-05-13_15:00:00 2004-05-13_18:00:00
2004-05-13_21:00:00 2004-05-14_00:00:00 2004-05-14_03
:00:00 2004-05-14_06:00:00 2004-05-14_09:00:00
2004-05-14_12:00:00 2004-05-14_15:00:00 2004-05-14_18
:00:00 2004-05-14_21:00:00 2004-05-15_00:00:00
2004-05-15_03:00:00 2004-05-15_06:00:00 2004-05-15_09
:00:00 2004-05-15_12:00:00 2004-05-15_15:00:00
2004-05-15_18:00:00 2004-05-15_21:00:00 2004-05-16_00
:00:00 2004-05-16_03:00:00 2004-05-16_06:00:00
2004-05-16_09:00:00 2004-05-16_12:00:00 2004-05-16_15
:00:00 2004-05-16_18:00:00 2004-05-16_21:00:00
2004-05-17_00:00:00 2004-05-17_03:00:00 2004-05-17_06
:00:00 2004-05-17_09:00:00 2004-05-17_12:00:00
2004-05-17_15:00:00 2004-05-17_18:00:00 2004-05-17_21
:00:00 2004-05-18_00:00:00 2004-05-18_03:00:00
2004-05-18_06:00:00 2004-05-18_09:00:00 2004-05-18_12
:00:00 2004-05-18_15:00:00 2004-05-18_18:00:00
2004-05-18_21:00:00 2004-05-19_00:00:00 2004-05-19_03
:00:00 2004-05-19_06:00:00 2004-05-19_09:00:00
2004-05-19_12:00:00 2004-05-19_15:00:00 2004-05-19_18
:00:00 2004-05-19_21:00:00 2004-05-20_00:00:00
2004-05-20_03:00:00 2004-05-20_06:00:00 2004-05-20_09
:00:00 2004-05-20_12:00:00 2004-05-20_15:00:00

86 APÉNDICE B. DOCUMENTACIÓN DE LA HERRAMIENTA DE ANOTACIONES

2004-05-20_18:00:00 2004-05-20_21:00:00 2004-05-21_00:00:00
2004-05-21_03:00:00 2004-05-21_06:00:00
2004-05-21_09:00:00 2004-05-21_12:00:00 2004-05-21_15:00:00
2004-05-21_18:00:00 2004-05-21_21:00:00
2004-05-22_00:00:00 2004-05-22_03:00:00 2004-05-22_06:00:00
2004-05-22_09:00:00 2004-05-22_12:00:00
2004-05-22_15:00:00 2004-05-22_18:00:00 2004-05-22_21:00:00
2004-05-23_00:00:00 2004-05-23_03:00:00
2004-05-23_06:00:00 2004-05-23_09:00:00 2004-05-23_12:00:00
2004-05-23_15:00:00 2004-05-23_18:00:00
2004-05-23_21:00:00 2004-05-24_00:00:00 2004-05-24_03:00:00
2004-05-24_06:00:00 2004-05-24_09:00:00
2004-05-24_12:00:00 2004-05-24_15:00:00 2004-05-24_18:00:00
2004-05-24_21:00:00 2004-05-25_00:00:00
2004-05-25_03:00:00 2004-05-25_06:00:00 2004-05-25_09:00:00
2004-05-25_12:00:00 2004-05-25_15:00:00
2004-05-25_18:00:00 2004-05-25_21:00:00 2004-05-26_00:00:00
2004-05-26_03:00:00 2004-05-26_06:00:00
2004-05-26_09:00:00 2004-05-26_12:00:00 2004-05-26_15:00:00
2004-05-26_18:00:00 2004-05-26_21:00:00
2004-05-27_00:00:00 2004-05-27_03:00:00 2004-05-27_06:00:00
2004-05-27_09:00:00 2004-05-27_12:00:00
2004-05-27_15:00:00 2004-05-27_18:00:00 2004-05-27_21:00:00
2004-05-28_00:00:00 2004-05-28_03:00:00
2004-05-28_06:00:00 2004-05-28_09:00:00 2004-05-28_12:00:00
2004-05-28_15:00:00 2004-05-28_18:00:00
2004-05-28_21:00:00 2004-05-29_00:00:00 2004-05-29_03:00:00
2004-05-29_06:00:00 2004-05-29_09:00:00
2004-05-29_12:00:00 2004-05-29_15:00:00 2004-05-29_18:00:00
2004-05-29_21:00:00 2004-05-30_00:00:00
2004-05-30_03:00:00 2004-05-30_06:00:00 2004-05-30_09:00:00
2004-05-30_12:00:00 2004-05-30_15:00:00
2004-05-30_18:00:00 2004-05-30_21:00:00 2004-05-31_00:00:00
2004-05-31_03:00:00 2004-05-31_06:00:00
2004-05-31_09:00:00 2004-05-31_12:00:00 2004-05-31_15:00:00
2004-05-31_18:00:00 2004-05-31_21:00:00
2004-06-01_00:00:00 2004-06-01_03:00:00 2004-06-01_06:00:00
2004-06-01_09:00:00 2004-06-01_12:00:00
2004-06-01_15:00:00 2004-06-01_18:00:00 2004-06-01_21:00:00
2004-06-02_00:00:00</data>

</Time>


```
<Longitude>
  <value>280</value>
  <data> -86.4603 -86.3701 -86.2798 -86.1896 -86.0994
    -86.0092 -85.919 -85.8288 -85.7386 -85.6484 -85.5582
    -85.468 -85.3778 -85.2876 -85.1974 -85.1072 -85.017
    -84.9268 -84.8366 -84.7464 -84.6562 -84.566 -84.4758
    -84.3856 -84.2954 -84.2052 -84.115 -84.0248 -83.9346
    -83.8444 -83.7542 -83.664 -83.5738 -83.4836 -83.3934
    -83.3032 -83.213 -83.1228 -83.0326 -82.9424 -82.8521
    -82.7619 -82.6717 -82.5815 -82.4913 -82.4011 -82.3109
    -82.2207 -82.1305 -82.0403 -81.9501 -81.8599 -81.7697
    -81.6795 -81.5893 -81.4991 -81.4089 -81.3187 -81.2285
    -81.1383 -81.0481 -80.9579 -80.8677 -80.7775 -80.6873
    -80.5971 -80.5069 -80.4167 -80.3265 -80.2363 -80.1461
    -80.0559 -79.9657 -79.8755 -79.7853 -79.6951 -79.6049
    -79.5147 -79.4245 -79.3343 -79.244 -79.1538 -79.0636
    -78.9734 -78.8832 -78.793 -78.7028 -78.6126 -78.5224
    -78.4322 -78.342 -78.2518 -78.1616 -78.0714 -77.9812
    -77.891 -77.8008 -77.7106 -77.6204 -77.5302 -77.44
    -77.3498 -77.2596 -77.1694 -77.0792 -76.989 -76.8988
    -76.8086 -76.7184 -76.6282 -76.538 -76.4478 -76.3576
    -76.2674 -76.1772 -76.087 -75.9968 -75.9066 -75.8164
    -75.7262 -75.6359 -75.5457 -75.4555 -75.3653 -75.2751
    -75.1849 -75.0947 -75.0045 -74.9143 -74.8241 -74.7339
    -74.6437 -74.5535 -74.4633 -74.3731 -74.2829 -74.1927
    -74.1025 -74.0123 -73.9221 -73.8319 -73.7417 -73.6515
    -73.5613 -73.4711 -73.3809 -73.2907 -73.2005 -73.1103
    -73.0201 -72.9299 -72.8397 -72.7495 -72.6593 -72.5691
    -72.4789 -72.3887 -72.2985 -72.2083 -72.118 -72.0278
    -71.9376 -71.8474 -71.7572 -71.667 -71.5768 -71.4866
    -71.3964 -71.3062 -71.216 -71.1258 -71.0356 -70.9454
    -70.8552 -70.765 -70.6748 -70.5846 -70.4944 -70.4042
    -70.314 -70.2238 -70.1336 -70.0434 -69.9532 -69.863
    -69.7728 -69.6826 -69.5924 -69.5022 -69.412 -69.3218
    -69.2316 -69.1414 -69.0512 -68.961 -68.8708 -68.7806
    -68.6904 -68.6002 -68.5099 -68.4197 -68.3295 -68.2393
    -68.1491 -68.0589 -67.9687 -67.8785 -67.7883 -67.6981
    -67.6079 -67.5177 -67.4275 -67.3373 -67.2471 -67.1569
    -67.0667 -66.9765 -66.8863 -66.7961 -66.7059 -66.6157
    -66.5255 -66.4353 -66.3451 -66.2549 -66.1647 -66.0745
    -65.9843 -65.8941 -65.8039 -65.7137 -65.6235 -65.5333
```

88APÉNDICE B. DOCUMENTACIÓN DE LA HERRAMIENTA DE ANOTACIONES

```
-65.4431 -65.3529 -65.2627 -65.1725 -65.0823 -64.9921
-64.9018 -64.8116 -64.7214 -64.6312 -64.541 -64.4508
-64.3606 -64.2704 -64.1802 -64.09 -63.9998 -63.9096
-63.8194 -63.7292 -63.639 -63.5488 -63.4586 -63.3684
-63.2782 -63.188 -63.0978 -63.0076 -62.9174 -62.8272
-62.737 -62.6468 -62.5566 -62.4664 -62.3762 -62.286
-62.1958 -62.1056 -62.0154 -61.9252 -61.835 -61.7448
-61.6546 -61.5644 -61.4741 -61.3839 -61.2937</data>
</Longitude>
<Latitude>
  <value>287</value>
  <data> -8.543 -8.4538 -8.3646 -8.2753 -8.1861 -8.0968
    -8.0075 -7.9181 -7.8288 -7.7394 -7.65 -7.5606 -7.4712
    -7.3817 -7.2923 -7.2028 -7.1133 -7.0238 -6.9342 -6.8447
    -6.7551 -6.6655 -6.5759 -6.4863 -6.3967 -6.307 -6.2174
    -6.1277 -6.038 -5.9483 -5.8586 -5.7688 -5.6791 -5.5893
    -5.4995 -5.4097 -5.3199 -5.2301 -5.1403 -5.0504
    -4.9606 -4.8707 -4.7808 -4.6909 -4.601 -4.5111 -4.4212
    -4.3312 -4.2413 -4.1513 -4.0613 -3.9714 -3.8814 -3.7914
    -3.7014 -3.6113 -3.5213 -3.4313 -3.3412 -3.2512
    -3.1611 -3.071 -2.981 -2.8909 -2.8008 -2.7107 -2.6206
    -2.5305 -2.4404 -2.3502 -2.2601 -2.17 -2.0798 -1.9897
    -1.8995 -1.8094 -1.7192 -1.6291 -1.5389 -1.4487 -1.3585
    -1.2684 -1.1782 -1.088 -0.9978 -0.90762 -0.81743
    -0.72723 -0.63703 -0.54684 -0.45663 -0.36643 -0.27624
    -0.18604 -0.09584 -0.0056305 0.084572 0.17477 0.26498
    0.35518 0.44538 0.53558 0.62578 0.71597 0.80617 0.89635
    0.98655 1.0767 1.1669 1.2571 1.3473 1.4375 1.5276
    1.6178 1.708 1.7981 1.8883 1.9784 2.0686 2.1587 2.2489
    2.339 2.4291 2.5192 2.6093 2.6994 2.7896 2.8796 2.9697
    3.0598 3.1499 3.2399 3.33 3.42 3.5101 3.6001 3.6901
    3.7801 3.8701 3.9601 4.0501 4.1401 4.23 4.32 4.4099
    4.4999 4.5898 4.6797 4.7696 4.8595 4.9493 5.0392 5.1291
    5.2189 5.3087 5.3985 5.4883 5.5781 5.6679 5.7576
    5.8474 5.9371 6.0268 6.1165 6.2062 6.2958 6.3855 6.4751
    6.5647 6.6543 6.7439 6.8335 6.9231 7.0126 7.1021
    7.1916 7.2811 7.3706 7.46 7.5494 7.6389 7.7282 7.8176
    7.907 7.9963 8.0856 8.1749 8.2642 8.3535 8.4427 8.5319
    8.6211 8.7103 8.7994 8.8885 8.9777 9.0667 9.1558 9.2449
    9.3339 9.4229 9.5118 9.6008 9.6897 9.7786 9.8675
    9.9564 10.0452 10.134 10.2228 10.3115 10.4003 10.489
```

```

10.5777 10.6663 10.755 10.8436 10.9321 11.0207 11.1092
11.1977 11.2862 11.3746 11.4631 11.5514 11.6398 11.7281
  11.8164 11.9047 11.993 12.0812 12.1694 12.2575 12.3457
  12.4338 12.5218 12.6099 12.6979 12.7859 12.8738
12.9617 13.0496 13.1375 13.2253 13.3131 13.4009 13.4886
  13.5763 13.664 13.7516 13.8392 13.9268 14.0143 14.1018
  14.1893 14.2767 14.3641 14.4515 14.5388 14.6261
14.7134 14.8006 14.8878 14.9749 15.0621 15.1492 15.2362
  15.3232 15.4102 15.4971 15.584 15.6709 15.7577 15.8445
  15.9313 16.018 16.1047 16.1913 16.2779 16.3645 16.451
  16.5375 16.624 16.7104 16.7968 16.8831 16.9694</data>
</Latitude>
<Level>
  <value>27</value>
  <data> 0.9965 0.988 0.9765 0.962 0.944 0.9215 0.8945
    0.85479 0.80436 0.75394 0.70352 0.63502 0.55272 0.47857
    0.41191 0.35211 0.29859 0.2508 0.20824 0.17045 0.13699
    0.10747 0.08151 0.058776 0.038949 0.021739 0.0068766</
  data>
</Level>
<Level_W>
  <value>28</value>
  <data> 1 0.993 0.983 0.97 0.954 0.934 0.909 0.88 0.82958
    0.77915 0.72873 0.6783 0.59174 0.51369 0.44345 0.38038
    0.32385 0.27333 0.22827 0.18821 0.15269 0.12129
    0.093642 0.069378 0.048173 0.029725 0.013753 0</data>
</Level_W>
<Longitude_U>
  <value>281</value>
  <data> -86.5054 -86.4152 -86.325 -86.2347 -86.1445 -86.0543
    -85.9641 -85.8739 -85.7837 -85.6935 -85.6033 -85.5131
    -85.4229 -85.3327 -85.2425 -85.1523 -85.0621 -84.9719
    -84.8817 -84.7915 -84.7013 -84.6111 -84.5209 -84.4307
    -84.3405 -84.2503 -84.1601 -84.0699 -83.9797 -83.8895
    -83.7993 -83.7091 -83.6189 -83.5287 -83.4385 -83.3483
    -83.2581 -83.1679 -83.0777 -82.9875 -82.8973 -82.8071
    -82.7169 -82.6266 -82.5364 -82.4462 -82.356 -82.2658
    -82.1756 -82.0854 -81.9952 -81.905 -81.8148 -81.7246
    -81.6344 -81.5442 -81.454 -81.3638 -81.2736 -81.1834
    -81.0932 -81.003 -80.9128 -80.8226 -80.7324 -80.6422
    -80.552 -80.4618 -80.3716 -80.2814 -80.1912 -80.101

```

90 APÉNDICE B. DOCUMENTACIÓN DE LA HERRAMIENTA DE ANOTACIONES

```
-80.0108 -79.9206 -79.8304 -79.7402 -79.65 -79.5598
-79.4696 -79.3794 -79.2892 -79.199 -79.1087 -79.0185
-78.9283 -78.8381 -78.7479 -78.6577 -78.5675 -78.4773
-78.3871 -78.2969 -78.2067 -78.1165 -78.0263 -77.9361
-77.8459 -77.7557 -77.6655 -77.5753 -77.4851 -77.3949
-77.3047 -77.2145 -77.1243 -77.0341 -76.9439 -76.8537
-76.7635 -76.6733 -76.5831 -76.4929 -76.4027 -76.3125
-76.2223 -76.1321 -76.0419 -75.9517 -75.8615 -75.7713
-75.6811 -75.5909 -75.5006 -75.4104 -75.3202 -75.23
-75.1398 -75.0496 -74.9594 -74.8692 -74.779 -74.6888
-74.5986 -74.5084 -74.4182 -74.328 -74.2378 -74.1476
-74.0574 -73.9672 -73.877 -73.7868 -73.6966 -73.6064
-73.5162 -73.426 -73.3358 -73.2456 -73.1554 -73.0652
-72.975 -72.8848 -72.7946 -72.7044 -72.6142 -72.524
-72.4338 -72.3436 -72.2533 -72.1631 -72.0729 -71.9827
-71.8925 -71.8023 -71.7121 -71.6219 -71.5317 -71.4415
-71.3513 -71.2611 -71.1709 -71.0807 -70.9905 -70.9003
-70.8101 -70.7199 -70.6297 -70.5395 -70.4493 -70.3591
-70.2689 -70.1787 -70.0885 -69.9983 -69.9081 -69.8179
-69.7277 -69.6375 -69.5473 -69.4571 -69.3669 -69.2767
-69.1865 -69.0963 -69.0061 -68.9159 -68.8257 -68.7355
-68.6452 -68.555 -68.4648 -68.3746 -68.2844 -68.1942
-68.104 -68.0138 -67.9236 -67.8334 -67.7432 -67.653
-67.5628 -67.4726 -67.3824 -67.2922 -67.202 -67.1118
-67.0216 -66.9314 -66.8412 -66.751 -66.6608 -66.5706
-66.4804 -66.3902 -66.3 -66.2098 -66.1196 -66.0294
-65.9392 -65.849 -65.7588 -65.6686 -65.5784 -65.4882
-65.398 -65.3078 -65.2176 -65.1273 -65.0371 -64.9469
-64.8567 -64.7665 -64.6763 -64.5861 -64.4959 -64.4057
-64.3155 -64.2253 -64.1351 -64.0449 -63.9547 -63.8645
-63.7743 -63.6841 -63.5939 -63.5037 -63.4135 -63.3233
-63.2331 -63.1429 -63.0527 -62.9625 -62.8723 -62.7821
-62.6919 -62.6017 -62.5115 -62.4213 -62.3311 -62.2409
-62.1507 -62.0605 -61.9703 -61.8801 -61.7899 -61.6997
-61.6095 -61.5192 -61.429 -61.3388 -61.2486</data>
</Longitude_U>
<Latitude_V>
<value>288</value>
<data> -8.5876 -8.4984 -8.4092 -8.32 -8.2307 -8.1414
-8.0521 -7.9628 -7.8735 -7.7841 -7.6947 -7.6053 -7.5159
-7.4265 -7.337 -7.2475 -7.158 -7.0685 -6.979 -6.8895
```

-6.7999 -6.7103 -6.6207 -6.5311 -6.4415 -6.3519 -6.2622
-6.1725 -6.0828 -5.9931 -5.9034 -5.8137 -5.7239
-5.6342 -5.5444 -5.4546 -5.3648 -5.275 -5.1852 -5.0953
-5.0055 -4.9156 -4.8258 -4.7359 -4.646 -4.556 -4.4661
-4.3762 -4.2862 -4.1963 -4.1063 -4.0163 -3.9264 -3.8364
-3.7464 -3.6563 -3.5663 -3.4763 -3.3863 -3.2962
-3.2061 -3.1161 -3.026 -2.9359 -2.8458 -2.7557 -2.6656
-2.5755 -2.4854 -2.3953 -2.3052 -2.215 -2.1249 -2.0348
-1.9446 -1.8545 -1.7643 -1.6741 -1.584 -1.4938 -1.4036
-1.3135 -1.2233 -1.1331 -1.0429 -0.95271 -0.86252
-0.77232 -0.68214 -0.59193 -0.50174 -0.41154 -0.32133
-0.23113 -0.14094 -0.050728 0.039474 0.12968 0.21987
0.31007 0.40028 0.49048 0.58068 0.67088 0.76107 0.85126
0.94146 1.0316 1.1218 1.212 1.3022 1.3924 1.4825
1.5727 1.6629 1.753 1.8432 1.9334 2.0235 2.1136 2.2038
2.2939 2.384 2.4742 2.5643 2.6544 2.7445 2.8346 2.9247
3.0148 3.1048 3.1949 3.285 3.375 3.465 3.5551 3.6451
3.7351 3.8251 3.9151 4.0051 4.0951 4.1851 4.275 4.365
4.4549 4.5448 4.6347 4.7246 4.8145 4.9044 4.9943 5.0841
5.174 5.2638 5.3536 5.4434 5.5332 5.623 5.7127 5.8025
5.8922 5.9819 6.0716 6.1613 6.251 6.3407 6.4303 6.5199
6.6096 6.6992 6.7887 6.8783 6.9678 7.0574 7.1469 7.2364
7.3258 7.4153 7.5047 7.5941 7.6836 7.7729 7.8623
7.9517 8.041 8.1303 8.2196 8.3088 8.3981 8.4873 8.5765
8.6657 8.7549 8.844 8.9331 9.0222 9.1113 9.2003 9.2894
9.3784 9.4674 9.5563 9.6453 9.7342 9.8231 9.9119
10.0008 10.0896 10.1784 10.2672 10.3559 10.4446 10.5333
10.622 10.7106 10.7993 10.8878 10.9764 11.065 11.1535
11.242 11.3304 11.4188 11.5072 11.5956 11.684 11.7723
11.8606 11.9488 12.0371 12.1253 12.2135 12.3016 12.3897
12.4778 12.5659 12.6539 12.7419 12.8299 12.9178
13.0057 13.0936 13.1814 13.2692 13.357 13.4447 13.5325
13.6201 13.7078 13.7954 13.883 13.9705 14.0581 14.1455
14.233 14.3204 14.4078 14.4951 14.5825 14.6697 14.757
14.8442 14.9314 15.0185 15.1056 15.1927 15.2797 15.3667
15.4537 15.5406 15.6275 15.7143 15.8011 15.8879
15.9746 16.0613 16.148 16.2346 16.3212 16.4078 16.4943
16.5807 16.6672 16.7536 16.8399 16.9262 17.0125</data>

</Latitude_V>
</Dimensions>
<Palettes>

92 APÉNDICE B. DOCUMENTACIÓN DE LA HERRAMIENTA DE ANOTACIONES

```
<Palette>
  <name> </name>
  <Size> </Size>
  <component_size> </component_size>
  <data> </data>
</Palette>
</Palettes>
<Variables>
  <T>
    <Palette>jet</Palette>
    <dimension_name>Longitude Latitude Level Time</
      dimension_name>
    <Size>280 287 27 273</Size>
    <description>Total Potential Temperature</description>
    <units>K</units>
    <Minimum>188.038010</Minimum>
    <Maximum>309.863861</Maximum>
  </T>
  <P>
    <Palette>hsv</Palette>
    <dimension_name>Longitude Latitude Level Time</
      dimension_name>
    <Size>280 287 27 273</Size>
    <description>Total Pressure</description>
    <units>Pa</units>
    <Minimum>5375.970215</Minimum>
    <Maximum>101742.828125</Maximum>
  </P>
  <RelHum>
    <Palette>hot</Palette>
    <dimension_name>Longitude Latitude Level Time</
      dimension_name>
    <Size>280 287 1 273</Size>
    <description>Relativity Humidity</description>
    <units>kg kg-1</units>
    <Minimum>0.220087</Minimum>
    <Maximum>1.726528</Maximum>
  </RelHum>
  <CloudFrac>
    <Palette>cool</Palette>
```

```

    <dimension_name>Longitude Latitude Level Time</
      dimension_name>
    <Size>280 287 27 273</Size>
    <description>CLOUD FRACTION</description>
    <units> </units>
    <Minimum>0.000000</Minimum>
    <Maximum>1.000000</Maximum>
  </CloudFrac>
  <U>
    <Palette>spring</Palette>
    <dimension_name>Longitude_U Latitude Level Time</
      dimension_name>
    <Size>281 287 27 273</Size>
    <description>x-wind component</description>
    <units>m s-1</units>
    <Minimum>-36.097393</Minimum>
    <Maximum>48.602547</Maximum>
  </U>
  <V>
    <Palette>summer</Palette>
    <dimension_name>Longitude Latitude_V Level Time</
      dimension_name>
    <Size>280 288 27 273</Size>
    <description>y-wind component</description>
    <units>m s-1</units>
    <Minimum>-33.546371</Minimum>
    <Maximum>33.705818</Maximum>
  </V>
  <W>
    <Palette>autumn</Palette>
    <dimension_name>Longitude Latitude Level_W Time</
      dimension_name>
    <Size>280 287 28 273</Size>
    <description>z-wind component</description>
    <units>m s-1</units>
    <Minimum>-3.305846</Minimum>
    <Maximum>13.783591</Maximum>
  </W>
</Variables>
</meta>

```

B.6. Formato del archivo de salida (XML)

Es el formato de salida que tiene la herramienta de anotaciones

```
<?xml version="1.0" encoding="UTF-8"?>
<estructuras>
  <estructura id="1" nombre="Confluencia" variableEscalar="T"
    variableVectorial="CV">
    <marcaciones>
      <marca tipo="2" tiempo="1" modoVisualizacion
        ="Superior" id="1" nivel="0.9965">
        <observaciones>
          Patron de ZCIT
        </observaciones>
        <puntos>
          <punto x="-74.28289794921875" y
            ="8.17490005493164" z
            ="0.9965000152587891">
          </punto>
          <punto x="-79.0635986328125" y
            ="2.699399948120117" z
            ="0.9965000152587891">
          </punto>
        </puntos>
      </marca>
      <marca tipo="2" tiempo="1" modoVisualizacion
        ="Superior" id="2" nivel="0.962">
        <observaciones>
          Patron de ZCIT
        </observaciones>
        <puntos>
          <punto x="-79.15380096435547" y
            ="7.638899803161621" z
            ="0.9620000123977661">
          </punto>
          <punto x="-75.27510070800781" y
            ="2.4291000366210938" z
            ="0.9620000123977661">
          </punto>
        </puntos>
      </marca>
```



```
<marca tipo="2" tiempo="6" modoVisualizacion
  ="Superior" id="3" nivel="0.8945">
  <observaciones>
    Patron de la ZCIT
  </observaciones>
  <puntos>
    <punto x="-80.05590057373047" y
      ="8.888500213623047" z
      ="0.8945000171661377">
    </punto>
    <punto x="-74.73390197753906" y
      ="3.600100040435791" z
      ="0.8945000171661377">
    </punto>
  </puntos>
</marca>
<marca tipo="2" tiempo="6" modoVisualizacion
  ="Superior" id="4" nivel="0.8945">
  <observaciones>
    Divergencia en el meta
  </observaciones>
  <puntos>
    <punto x="-71.57679748535156" y
      ="4.859499931335449" z
      ="0.8945000171661377">
    </punto>
    <punto x="-76.08699798583984" y
      ="0.44538000226020813" z
      ="0.8945000171661377">
    </punto>
  </puntos>
</marca>
<marca tipo="2" tiempo="6" modoVisualizacion
  ="Superior" id="5" nivel="0.8945">
  <observaciones>
    Divergencias en el pacifico
  </observaciones>
  <puntos>
    <punto x="-83.03260040283203" y
      ="1.7980999946594238" z
      ="0.8945000171661377">
```

```

        </punto>
        <punto x="-86.37010192871094" y
            ="-4.870699882507324" z
            ="0.8945000171661377">
        </punto>
    </puntos>
</marca>
<marca tipo="2" tiempo="6" modoVisualizacion
    ="Superior" id="6" nivel="0.41191">
    <observaciones>
        Patron de la ZCIT
    </observaciones>
    <puntos>
        <punto x="-79.24400329589844" y
            ="7.281099796295166" z
            ="0.4119099974632263">
        </punto>
        <punto x="-74.73390197753906" y
            ="2.4291000366210938" z
            ="0.4119099974632263">
        </punto>
    </puntos>
</marca>
</marcaciones>
</estructura>
</estructuras>

```

B.7. Repositorio en git

```
git clone ssh://username@190.60.115.153:20022/var/repo/ASAR_Annotation_
Tools.git
```

B.8. Manual de usuario

Apéndice C

Experimentación propuesta

C.1. Extracción de ROIs

Después de realizar el proceso de anotado se debe cargar la salida de este proceso con el fin de extraer las ROI que hizo cada experto

Detalles de la experimentación

Se planteo un marco experimental que extraía las ROI y las almacenaba como archivos *mat*.

```
clc;clear all;close all
%
addpath([pwd '/Extraccion'])
addpath([pwd '/SVM'])
addpath([pwd '/SVM/libsvm-3.20'])
addpath([pwd '/SVM/libsvm-3.20/matlab'])
addpath([pwd '/Phase_Portrait'])
addpath([pwd '/Visualizacion'])
% %addpath('../data/')
%
% path = '/home/hpclab-su/Documents/Experimentacion/data/train/';
% path = '/home/jmolina/Experimentacion/data/train/';
% %path = '/home/juanmolina/data/WRF_DATA/data/train/';
% folder = dir(path);
% cd(path);
%
```

```

% tam = length(folder);
% %% Generacion de estructuras con las anotaciones y volúmenes
% for i = 3:tam
%     NombreCarpeta = folder(i).name;
%     cd(NombreCarpeta);
%     VolumenDatos{i-2} = visualizarDatos([NombreCarpeta]);
%     cd('..')
% end
% cd('.././../script')
% % if(isdir('fig'))
% % else
% %     mkdir('fig')
% % end
%
%
% %% Generar Regiones de Interés
% ROIS = [];
% conta = [];
% for jj = 1:length(VolumenDatos)
%     [Roi, RoiU, RoiV, Roiclass, RoiTiempo, RoiNiveles, ROIID, Nombre,
%     cont] = generarROI(VolumenDatos{jj});
%     ROI.Nombre = VolumenDatos{jj}.Nombre;
%     ROI.roi = Roi;
%     ROI.roiU = RoiU;
%     ROI.roiV = RoiV;
%     ROI.roiclass = Roiclass;
%     ROI.ROIID = ROIID;
%     ROI.Nombre = Nombre;
%     conta(jj) = sum(cont);
%
%
%     ROIS = [ROIS ROI];
%
%
%
% end
% jj
% sum(conta)
%% Obtener Nombres
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function [volumenreal]=generateVolume(file,tamano)

```

```

% file es el nombre del archivo
% el tamaño del volumen que se necesita
% Ejemplo
% tamañoU=[numlev numlonU numlat];
% tamañoV=[numlev numlon numlatV];
% tamañoW=[numlevW numlon numlat];
% [volumrealU]=generateVolume(file,tamañoU);
% [volumrealV]=generateVolume(file,tamañoU);
% [volumrealW]=generateVolume(file,tamañoU);
fT=fopen(file);
FcomponenteT=fread(fT, (tamaño(1)*tamaño(2)*tamaño(1)*tamaño(3)
), 'float');
g4f=zeros(tamaño(3),tamaño(2),tamaño(1));
k=tamaño(1);
for i=1:k

    g4 = reshape(FcomponenteT((i-1)*tamaño(2)*tamaño(3)+1:i*
        tamaño(2)*tamaño(3)), tamaño(2), tamaño(3));
    g4=g4';
    g4=flipud(g4);
    g4f(:,:,i)=g4;
end
volumenreal=g4f;
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function [VolumeData] = visualizarDatos(NombreCarpeta)
    import java.lang.*
    % Archivo XML generado en el proceso de extracción de
    variables
    archivodemetadatos = [NombreCarpeta '/metadataWRF.xml'];
    % Archivo XML generado en el proceso de anotaciones
    archivodedatosanotados=[NombreCarpeta '/' NombreCarpeta '-
        JuanMolina-jmolinar3@ucentral.edu.co.wsa'];

    entrada=[''];

    variables={'T' 'P' 'U' 'V' 'W'};

    metadataWRF=xml2structV(archivodemetadatos);
    latitudes=metadataWRF.meta.Dimensions.Latitude.data.Text;

```

```

latsC = String(latitudes);
longitudes=metadataWRF.meta.Dimensions.Longitude.data.Text;
lonsC = String(longitudes);
levels=metadataWRF.meta.Dimensions.Level.data.Text;
levsC = String(levels);
latitudesV=metadataWRF.meta.Dimensions.Latitude_V.data.Text;
latsVC = String(latitudesV);
longitudesU=metadataWRF.meta.Dimensions.Longitude_U.data.
    Text;
lonsUC = String(longitudesU);
levelsW=metadataWRF.meta.Dimensions.Level_W.data.Text;
levsWC = String(levelsW);

separador=' ';

lats=latsC.trim.split(separador);
lons=lonsC.trim.split(separador);
levs=levsC.trim.split(separador);

latsV=latsVC.trim.split(separador);
lonsU=lonsUC.trim.split(separador);
levsW=levsWC.trim.split(separador);

latsN=[];
lonsN=[];
latsVN=[];
lonsUN=[];
levsN=[];
levsWN=[];

for i=1:length(lats)
    latsN(i)=str2double(lats(i));
end
for i=1:length(lons)
    lonsN(i)=str2double(lons(i));
end
for i=1:length(latsV)
    latsVN(i)=str2double(latsV(i));
end
for i=1:length(lonsU)
    lonsUN(i)=str2double(lonsU(i));
end

```

```
end
for i=1:length(levs)
    levsN(i)=str2double(levs(i));
end
for i=1:length(levsW)
    levsWN(i)=str2double(levsW(i));
end

latsN=flipud(latsN');
latsVN=flipud(latsVN');

numlat=str2num(metadataWRF.meta.Dimensions.Latitude.value.
    Text);
numlon=str2num(metadataWRF.meta.Dimensions.Longitude.value.
    Text);
numlev=str2num(metadataWRF.meta.Dimensions.Level.value.Text)
;
numlatV=str2num(metadataWRF.meta.Dimensions.Latitude_V.value
    .Text);
numlonU=str2num(metadataWRF.meta.Dimensions.Longitude_U.
    value.Text);
numlevW=str2num(metadataWRF.meta.Dimensions.Level_W.value.
    Text);

tamanoT=[numlev numlon numlat];
tamanoU=[numlev numlonU numlat];
tamanoV=[numlev numlon numlatV];
tamanoW=[numlevW numlon numlat];

maximoP=str2num(metadataWRF.meta.Variables.P.Maximum.Text);
minimoP=str2num(metadataWRF.meta.Variables.P.Minimum.Text);
maximoT=str2num(metadataWRF.meta.Variables.T.Maximum.Text);
minimoT=str2num(metadataWRF.meta.Variables.T.Minimum.Text);
maximoU=str2num(metadataWRF.meta.Variables.U.Maximum.Text);
minimoU=str2num(metadataWRF.meta.Variables.U.Minimum.Text);
maximoV=str2num(metadataWRF.meta.Variables.V.Maximum.Text);
minimoV=str2num(metadataWRF.meta.Variables.V.Minimum.Text);
maximoW=str2num(metadataWRF.meta.Variables.W.Maximum.Text);
minimoW=str2num(metadataWRF.meta.Variables.W.Minimum.Text);

maximoF=[maximoT maximoP maximoU maximoV maximoW];
```

```

minimoF=[minimoT minimoP minimoU minimoV minimoW];

%% Generar los volumenes de la simulacion
%% Datos de las anotaciones
metadata=xml2structV(archivodedatosanotados);

estructura=metadata.estructuras.estructura;
estructurasTotales=[];
for i=1:length(estructura)
    % Extraccion de los metadatos
    Atributos=estructura{i}.Attributes;
    marcaciones=estructura{i}.marcaciones.marca;
    if length(marcaciones) == 1
        marcaciones = {marcaciones};
    end

    anotaciones=[];
    for j=1:length(marcaciones)
        anotacion.id=marcaciones{j}.Attributes.id;
        anotacion.modovisualizacion=marcaciones{j}.Attributes
            .modoVisualizacion;
        anotacion.nivel=marcaciones{j}.Attributes.nivel;
        tiempo(i,j)=str2num(marcaciones{j}.Attributes.tiempo)
            ;
    end
end
maximotiempo = max(tiempo(:));
for iii = 1:maximotiempo
    volumrealT = [];
    volumrealP = [];
    volumrealU = [];
    volumrealV = [];
    volumrealW = [];
    for kk=1:length(variables)
        if strcmp(variables{kk},'T')
            [volumrealT]=generateVolume([entrada
                variables{kk} '/' variables{kk} num2str(
                    iii) '.dat'],tamanoT);
        end
        if strcmp(variables{kk},'P')

```



```

        [volumrealP]=generateVolume([entrada
            variables{kk} '/' variables{kk} num2str(
            iii) '.dat'],tamanoT);
    end
    if strcmp(variables{kk},'U')
        [volumrealU]=generateVolume([entrada
            variables{kk} '/' variables{kk} num2str(
            iii) '.dat'],tamanoU);
    end
    if strcmp(variables{kk},'V')
        [volumrealV]=generateVolume([entrada
            variables{kk} '/' variables{kk} num2str(
            iii) '.dat'],tamanoV);
    end
    if strcmp(variables{kk},'W')
        [volumrealW]=generateVolume([entrada variables{kk}
            '/' variables{kk} num2str(iii) '.dat'],tamanoW);
    end

end

end

[volU,volV,volW]=imageinterp(volumrealU,volumrealV,
    volumrealW,[numlat,numlon],numlev);
[x y z] = size(volU);
Magnitud = sqrt((volU.*volU)+(volV.*volV));
for id = 1:z
    div(:,:,id) = divergence(volU(:,:,id),-volV(:,:,id));
    [curlz(:,:,id) cav(:,:,id)] = curl(volU(:,:,id),-volV
        (:,:,id));

end
divv{iii} = div;
curlzv{iii} = curlz;
volumenU{iii} = volU;
volumenV{iii} = volV;
magnitudWinds{iii} = Magnitud;

end

```

```

for i=1:length(estructura)
    % Extraccion de los metadatos
    Atributos=estructura{i}.Attributes;

    NombreEstructura = Atributos.nombre;
    marcaciones=estructura{i}.marcaciones.marca;
    if length(marcaciones) == 1
        marcaciones = {marcaciones};
    end

    anotaciones=[];
    for j=1:length(marcaciones)
        anotacion.id=marcaciones{j}.Attributes.id;
        anotacion.modovisualizacion=marcaciones{j}.Attributes
            .modoVisualizacion;
        anotacion.nivel=marcaciones{j}.Attributes.nivel;
        anotacion.tiempo=marcaciones{j}.Attributes.tiempo;
        anotacion.tipo=marcaciones{j}.Attributes.tipo;
        observ = String(marcaciones{j}.observaciones.Text);
        anotacion.observaciones=char(observ.trim);
        puntos=marcaciones{j}.puntos.punto;
        x=[];y=[];z=[];
        for k=1:length(puntos)
            x(k)=str2double(puntos{k}.Attributes.x);
            y(k)=str2double(puntos{k}.Attributes.y);
            z(k)=str2double(puntos{k}.Attributes.z);
            xn(k)=find(abs(lonsN-x(k))<1e-5);
            yn(k)=find(abs(latsN-y(k))<1e-5);
            zn(k)=find(abs(levsN-z(k))<1e-5);
        end

        if xn(1)<xn(2)
            anotacion.xnf=fliplr(xn);
            anotacion.xf=fliplr(x);
        else
            anotacion.xf=x;
            anotacion.xnf=xn;
        end

        if yn(1)<yn(2)

```

```

        anotacion.ynf=fliplr(yn);
        anotacion.yf=fliplr(y);
    else
        anotacion.ynf=yn;
        anotacion.yf=y;
    end
    anotacion.zf=z;
    anotacion.znf=zn;

    % Creacion de los volúmenes
    P=zeros([numlat numlon]);
    %Pim=zeros([numlat numlon]);
    T=zeros([numlat numlon]);
    %Tim=zeros([numlat numlon]);

    U=zeros([numlat numlonU]);
    %Uim=zeros([numlat numlon]);

    V=zeros([numlatV numlon]);
    %Vim=zeros([numlat numlon]);

    W=zeros([numlat numlon]);
    %Wim=zeros([numlat numlon]);
    if xn(1)>tamanoT(2)
        disp('mostrar longitud 1')
    end
    if xn(1)>tamanoT(2)
        disp('mostrar longitud 2')
    end

    if yn(1)>tamanoT(3)
        disp('mostrar latitud 1')
    end
    if yn(1)>tamanoT(3)
        disp('mostrar latitud 2')
    end
    end
    % for ii=xn(1):xn(2)
    %     for jj=yn(1):yn(2)
    %         P(jj,ii)=volumrealP(jj,ii,zn(1));
    %         T(jj,ii)=volumrealT(jj,ii,zn(1));
    %         U(jj,ii)=volU(jj,ii,zn(1));

```

```

%           V(jj,ii)=volV(jj,ii,zn(1));
%           W(jj,ii)=volW(jj,ii,zn(1));

%   end
% end

%
% FP=fopen([salida '/' Atributos.nombre '/'
%          anotacion.observaciones 'P' anotacion.tiempo '-'
%          anotacion.nivel '.dat'],'w');
% fwrite(FP,P,'single');
% fclose(FP);
% FT=fopen([salida '/' Atributos.nombre '/'
%          anotacion.observaciones 'T' anotacion.tiempo '-'
%          anotacion.nivel '.dat'],'w');
% fwrite(FT,T,'single');
% fclose(FT);
% FU=fopen([salida '/' Atributos.nombre '/'
%          anotacion.observaciones 'U' anotacion.tiempo '-'
%          anotacion.nivel '.dat'],'w');
% fwrite(FU,U,'single');
% fclose(FU);
% FV=fopen([salida '/' Atributos.nombre '/'
%          anotacion.observaciones 'V' anotacion.tiempo '-'
%          anotacion.nivel '.dat'],'w');
% fwrite(FV,V,'single');
% fclose(FV);
% FW=fopen([salida '/' Atributos.nombre '/'
%          anotacion.observaciones 'W' anotacion.tiempo '-'
%          anotacion.nivel '.dat'],'w');
% fwrite(FW,W,'single');
% fclose(FW);
% anotaciones=[anotaciones anotacion];
end

estructuras.anotaciones =anotaciones;
%estructuras.anotaciones =anotaciones;
estructuras.Atributos = NombreEstructura;
estructurasTotales=[estructurasTotales estructuras];
end
VolumeData.Nombre = NombreCarpeta;

```

```

    VolumeData.Magnitud = magnitudWinds;
    VolumeData.div = divv;
    VolumeData.curlz = curlzv;
    VolumeData.estructuras = estructurasTotales;
    VolumeData.componenteU = volumenU;
    VolumeData.componenteV = volumenV;
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function [ROIP,CUP,CVP,ClaseP,ROITiempos,ROINiveles,ROIDS,
    NombreData, cont] = generarROI(VolumenDatos)
% Carga la Magnitud y las componentes de Vientos U y V
MagnitudWinds = VolumenDatos.Magnitud;
ComponenteU = VolumenDatos.componenteU;
ComponenteV = VolumenDatos.componenteV;
div = VolumenDatos.div;
curlz = VolumenDatos.curlz;
[x y z] = size(MagnitudWinds{1});
xn = 1:4:x;
yn = 1:4:y;
[X Y] = meshgrid(yn,xn);
[sx sy]=meshgrid(1:10:y,1:10:x);
cont = 0;
NombreData = {};
for i = 1:length(VolumenDatos.estructuras)
    NombreEstructura = VolumenDatos.estructuras(i).Atributos;
    % Nombrecarpeta = ['/home/hpclab-su/Documents/
        Experimentacion/data/train/' VolumenDatos.Nombre '/'
        Anotaciones/' NombreEstructura];
    Nombrecarpeta = ['/home/jmolina/Experimentacion/data/train/'
        VolumenDatos.Nombre '/Anotaciones/' NombreEstructura];
    % Nombrecarpeta = ['/home/juanmolina/data/WRF_DATA/data/
        train/' VolumenDatos.Nombre '/Anotaciones/'
        NombreEstructura];
    mkdir(Nombrecarpeta)
    %mkdir(['fig/' VolumenDatos.Nombre '/' NombreEstructura])
    %mkdir(['fig/Results/' VolumenDatos.Nombre '/'
        NombreEstructura])

    for j = 1:length(VolumenDatos.estructuras(i).anotaciones)
        sid = VolumenDatos.estructuras(i).anotaciones(j).id;

```

```

tiempo = VolumenDatos.estructuras(i).anotaciones(j).
    tiempo;
nivel = VolumenDatos.estructuras(i).anotaciones(j).znf;
nivel = nivel(1);
xnf = VolumenDatos.estructuras(i).anotaciones(j).xnf;
ynf = VolumenDatos.estructuras(i).anotaciones(j).ynf;
%
%   h1 = figure('visible','off');
%   U = ComponenteU{str2num(tiempo)};
%   V = ComponenteV{str2num(tiempo)};
%   imshow(MagnitudWinds{str2num(tiempo)}(:, :, nivel), [])
%   colormap jet
%   colorbar
%   hold on
%   rectangle('Position', [min(xnf), min(ynf), abs(xnf(2)-
xnf(1)), abs(ynf(2)-ynf(1))])
%   quiver(X, Y, U(1:4:end, 1:4:end, nivel), V(1:4:end, 1:4:end,
nivel), 'k')
%   h=stream2(X, Y, U(1:4:end, 1:4:end, 1), V(1:4:end, 1:4:end
, 1), sx, sy);
%   g = streamline(h);
%   set(g, 'Color', 'magenta')
%   hold off
%   name=['fig/' VolumenDatos.Nombre '-' NombreEstructura
' Magnitud - ' num2str(tiempo) ' - ' num2str(nivel) ' - '
num2str(i) num2str(j)];
%   saveas(h1, name, 'png')
%   close all;
ROI Nivel{j} = (nivel);
ROI tiempo{j} = str2num(tiempo);
ROI{j} = MagnitudWinds{str2num(tiempo)}(min(ynf):max(ynf)
), min(xnf):max(xnf), nivel);
CU{j} = ComponenteU{str2num(tiempo)}(min(ynf):max(ynf),
min(xnf):max(xnf), nivel);
CV{j} = ComponenteV{str2num(tiempo)}(min(ynf):max(ynf),
min(xnf):max(xnf), nivel);
Div{j} = div{str2num(tiempo)}(min(ynf):max(ynf), min(xnf)
:max(xnf), nivel);
Curlz{j} = curlz{str2num(tiempo)}(min(ynf):max(ynf), min(
xnf):max(xnf), nivel);
Clase{j} = NombreEstructura;

```

```

ROID.id = str2num(sid);
ROID.Nivel = ROINivel{j};
ROID.tiempo = ROItiempo{j};
ROID.x = xnf;
ROID.y = ynf;
ROID.Magnitud = MagnitudWinds{str2num(tiempo)}(:, :, nivel
);
ROID.U = ComponenteU{str2num(tiempo)}(:, :, nivel);
ROID.V = ComponenteV{str2num(tiempo)}(:, :, nivel);
ROID.Divergences = div{str2num(tiempo)}(:, :, nivel);
ROID.Curlz = curlz{str2num(tiempo)}(:, :, nivel);
ROID.ROI = ROI{j};
ROID.CU = CU{j};
ROID.CV = CV{j};
ROID.Div = Div{j};
ROID.Curlzs = Curlz{j};
ROID.Clase = Clase{j};
ROID.psi = CU{j}'*CV{j}*(pinv(CV{j}'*CV{j}))*CV{j}'*CU{j
} - CU{j}'*CU{j};
name=[Nombrecarpeta '/' sid '-' NombreEstructura '-'
num2str(tiempo) '-' num2str(nivel)];
ROID.Name = [name '.mat'];
ROIDs{j} = ROID;
save([name '.mat'], 'ROID')
ROID.Lambdas = phasePortrait([name '.mat']);
Nombre{j} = [name '.mat'];

% saveAnnotations([name '.mat'], ['fig/' VolumenDatos.
Nombre '/' NombreEstructura], [sid '-'
NombreEstructura '-' num2str(tiempo) '-' num2str(
nivel)])
% saveResults([name '.mat'], ['fig/Results/' VolumenDatos
.Nombre '/' NombreEstructura], [sid '-'
NombreEstructura '-' num2str(tiempo) '-' num2str(
nivel)])
end
cont = cont + length(VolumenDatos.estructuras(i).anotaciones
);
save([VolumenDatos.Nombre '-' NombreEstructura '.mat'], '
Nombre', 'Clase', 'NombreEstructura')

```

```

NombreData{i} = Nombre;

ROITiempos{i} = ROItiempo;
ROINiveles{i} = ROINivel;
ROIP{i} = ROI;
CUP{i} = CU;
CVP{i} = CV;
ClaseP{i} = Clase;
ROIDS{i} = ROIDS;
Clase = {};
Nombre = {};
end
cont = sum(cont);
end

```

C.2. Experimentación

Para la experimentación se usaron los archivos *mat* anteriormente extraídos y se implementaron los diferentes descriptores que se propusieron en el capítulo 3

```

clc;clear all;close all
lista = dir('*.mat');

TestdatacsdDiv = [];
TestdatacsdCurl = [];
TestdatacsdMagnitud = [];
TraindatacsdDiv = [];
TraindatacsdCurl = [];
TraindatacsdMagnitud = [];

TestClases = [];
TrainClases = [];
TestdataCoocurrenceMatrix = [];
TraindataCoocurrenceMatrixNormalization = [];
TraindataCoocurrenceMatrix = [];
TraindataCoocurrenceMatrixNormalization = [];
Testlambdas = [];
Trainlambdas = [];
TestLambdas = [];

```



```

TrainLambdas = [];
TestLambdas1 = [];
TrainLambdas1 = [];
TestLambdas2 = [];
TrainLambdas2 = [];
TestLambdas3 = [];
TrainLambdas3 = [];
TrainLambdaHistogram = [];
TestLambdaHistogram = [];
TrainTraceDetHistogram = [];
TestTraceDetHistogram = [];
TraincsdLambdaHistogram = [];
TestcsdLambdaHistogram = [];
TraincsdTraceDetHistogram = [];
TestcsdTraceDetHistogram = [];
contar = [];
for i = 1:length(lista)
    i
    lista(i).name
    file = lista(i).name;
    load(char(file));

    contar(i) = length(Nombre);
    for j = 1:length(Nombre)
        filename = Nombre{j};

        load(filename)
        % Primer metodo
        normalize = (ROID.psi - min(min(ROID.psi)))/(max(max(ROID.
            psi)) - min(min(ROID.psi)));

        glcm = graycomatrix((normalize));
        normaglcm = (glcm - min(min(glcm))) / (max(max(glcm)) - min(
            min(glcm)));

        % Segundo Metodo
        S = entropyfilt(normalize);
        normalizates = (S - min(min(S)))/(max(max(S)) - min(min(S)))
        ;

        glcm1 = graycomatrix((normalizates));

```

```

normaglcm1 = (glcm1 - min(min(glcm1))) / (max(max(glcm1)) -
    min(min(glcm1)));

%     dataset.Class = Clase{j};
%     dataset.data = normaglcm1(:);
%
%     dataset1.Class = Clase{j};
%     dataset1.data = normaglcm1(:);
% Tercer Metodo
normalizeROI = (ROID.ROI - min(min(ROID.Magnitud)))/(max(
    max(ROID.Magnitud)) - min(min(ROID.Magnitud)));
normalizeDiv = (ROID.Div - min(min(ROID.Divergences)))/(
    max(max(ROID.Divergences)) - min(min(ROID.Divergences)))
;
normalizeCurl = (ROID.Curlzs - min(min(ROID.Curlz)))/(max(
    max(ROID.Curlz)) - min(min(ROID.Curlz)));

csdROI = extract_csd(uint8(255*normalizeROI),256);
csdDiv = extract_csd(uint8(255*normalizeDiv),256);
csdCurl = extract_csd(uint8(255*normalizeCurl),256);

%     data = [data;csdROI];
%     data1 = [data;csdDiv];
%     data2 = [data;csdCurl];

% Cuarto Metodo
[dUx dUy] = gradient(ROID.CU);
[dVx dVy] = gradient(-ROID.CV);
detA = (dUx .* dVy) - (dVx .* dUy);
traceA = dUx + dVy;

%detA = ROID.Curlzs;
%traceA = ROID.Div;

detAA = mean2(detA);
traceAA = mean2(traceA);

```

```

stddetAA = std2(detA);
stdtraceAA = std2(traceA);

vardetAA = std2(detA).^2;
vartraceAA = std2(traceA).^2;

maxdetAA = max(max(detA));
maxtraceAA = max(max(traceA));

mindetAA = min(min(detA));
mintraceAA = min(min(traceA));

lambda1 = 0.5 * (mean2(traceA) - sqrt((mean2(traceA) .*
    mean2(traceA) - 4 * mean2(detA))));
lambda2 = 0.5 * (mean2(traceA) + sqrt((mean2(traceA) .*
    mean2(traceA) - 4 * mean2(detA))));

lambda3 = 0.5 * (std2(traceA) - sqrt((std2(traceA) .* std2(
    traceA) - 4 * std2(detA))));
lambda4 = 0.5 * (std2(traceA) + sqrt((std2(traceA) .* std2(
    traceA) - 4 * std2(detA))));

lambda5 = 0.5 * (vartraceAA - sqrt((vartraceAA .* vartraceAA
    - 4 * vardetAA)));
lambda6 = 0.5 * (vartraceAA + sqrt((vartraceAA .* vartraceAA
    - 4 * vardetAA)));

lambda7 = 0.5 * (max(max(traceA)) - sqrt((max(max(traceA))
    .* max(max(traceA)) - 4 * max(max(detA))));
lambda8 = 0.5 * (max(max(traceA)) + sqrt((max(max(traceA))
    .* max(max(traceA)) - 4 * max(max(detA))));

lambda9 = 0.5 * (min(min(traceA)) - sqrt((min(min(traceA))
    .* min(min(traceA)) - 4 * min(min(detA))));
lambda10 = 0.5 * (min(min(traceA)) + sqrt((min(min(traceA))
    .* min(min(traceA)) - 4 * min(min(detA))));

% dataset2.Class = Clase{j};
dataset2.detA = detAA;
dataset2.traceA = traceAA;
dataset2.stddetA = stddetAA;

```

```

dataset2.stdtraceA = stdtraceAA;
dataset2.vardetA = vardetAA;
dataset2.vartraceA = vartraceAA;
dataset2.lambda1 = lambda1;
dataset2.lambda2 = lambda2;
dataset2.lambda3 = lambda3;
dataset2.lambda4 = lambda4;
dataset2.lambda5 = lambda5;
dataset2.lambda6 = lambda6;
dataset2.lambda7 = lambda7;
dataset2.lambda8 = lambda8;
dataset2.lambda9 = lambda9;
dataset2.lambda10 = lambda10;

%      lambdas = [lambdas dataset2];
vect = phasePortraitMoments(ROID.CU,ROID.CV);
vec1 = phasePortraitMoments2(ROID.CU,ROID.CV);
vec = [real(lambda1) imag(lambda1) ...
       real(lambda2) imag(lambda2)...
       real(lambda3) imag(lambda3)...
       real(lambda4) imag(lambda4)...
       real(lambda5) imag(lambda5)...
       real(lambda6) imag(lambda6)...
       detAA traceAA stddetAA stdtraceAA maxdetAA maxtraceAA
       mindetAA mintraceAA...
       vardetAA vartraceAA mean2(ROID.Curlzs) mean2(ROID.Div)
       ...
       std2(ROID.Curlzs) std2(ROID.Div) ...
       (std2(ROID.Curlzs).*std2(ROID.Curlzs))...
       (std2(ROID.Div).*std2(ROID.Div)) max(max(ROID.Div)) max(
       max(ROID.Curlzs))...
       min(min(ROID.Div)) min(min(ROID.Curlzs))];
vec3 = [real(lambda1) imag(lambda1) ...
       real(lambda2) imag(lambda2)...
       real(lambda3) imag(lambda3)...
       real(lambda4) imag(lambda4)...
       real(lambda5) imag(lambda5)...
       real(lambda6) imag(lambda6)...
       real(lambda7) imag(lambda7)...
       real(lambda8) imag(lambda8)...
       real(lambda9) imag(lambda9)...

```

```

real(lambda10) imag(lambda10)...
detAA traceAA stddetAA stdtraceAA maxdetAA maxtraceAA
    mindetAA mintraceAA...
vardetAA vartraceAA mean2(ROID.Curlzs) mean2(ROID.Div)
...
std2(ROID.Curlzs) std2(ROID.Div) ...
(std2(ROID.Curlzs).*std2(ROID.Curlzs))...
(std2(ROID.Div).*std2(ROID.Div)) max(max(ROID.Div)) max(
    max(ROID.Curlzs))...
min(min(ROID.Div)) min(min(ROID.Curlzs))];

%   vec = [real(lambda1) imag(lambda1) ...
%         real(lambda2) imag(lambda2)...
%         real(lambda3) imag(lambda3)...
%         real(lambda4) imag(lambda4)...
%         real(lambda5) imag(lambda5)...
%         real(lambda6) imag(lambda6)...
%         detAA traceAA stddetAA stdtraceAA ...
%         vardetAA vartraceAA mean2(ROID.Curlzs) mean2(ROID.Div)
...
%         std2(ROID.Curlzs) std2(ROID.Div) ...
%         (std2(ROID.Curlzs).*std2(ROID.Curlzs))...
%         (std2(ROID.Div).*std2(ROID.Div)) ];%...
%         %entropydetAA entropytraceAA];
lambdaG1 = 0.5 * (traceA) - sqrt((traceA).*(traceA) - 4 .*
    detA);
lambdaG2 = 0.5 * (traceA) + sqrt((traceA).*(traceA) - 4 .*
    detA);

lambdaG1R = real(lambdaG1);
lambdaG2R = real(lambdaG2);

lambdaG1I = imag(lambdaG1);
lambdaG2I = imag(lambdaG2);

% normalizeROI = (ROID.ROI - min(min(ROID.Magnitud)))/(max
    (max(ROID.Magnitud)) - min(min(ROID.Magnitud)));
normalizelambdaG1R = (lambdaG1R - min(min(lambdaG1R)))/(
    max(max(lambdaG1R)) - min(min(lambdaG1R)));
normalizelambdaG2R = (lambdaG2R - min(min(lambdaG2R)))/(
    max(max(lambdaG2R)) - min(min(lambdaG2R)));

```

```

normalizatelambdaG1I = (lambdaG1I - min(min(lambdaG1I)))/(
    max(max(lambdaG1I)) - min(min(lambdaG1I)));
normalizatelambdaG2I = (lambdaG2I - min(min(lambdaG2I)))/(
    max(max(lambdaG2I)) - min(min(lambdaG2I)));

normalizedetA = (detA - min(min(detA)))/(max(max(detA)) -
    min(min(detA)));
normalizatetraceA = (traceA - min(min(traceA)))/(max(max(
    traceA)) - min(min(traceA)));
[h1] = imhist(normalizedetA)/(size(ROID.CU,1)*size(ROID.
    CU,2));
[h2] = imhist(normalizatetraceA)/(size(ROID.CU,1)*size(ROID
    .CU,2));
h = [h1;h2];
h1csd = extract_csd(uint8(255*normalizedetA),256);
h2csd = extract_csd(uint8(255*normalizatetraceA),256);

hcsd = [h1csd h2csd];

[hs1] = imhist(normalizatelambdaG1R)/(size(ROID.CU,1)*size(
    ROID.CU,2));
[hs2] = imhist(normalizatelambdaG2R)/(size(ROID.CU,1)*size(
    ROID.CU,2));
[hs3] = imhist(normalizatelambdaG1I)/(size(ROID.CU,1)*size(
    ROID.CU,2));
[hs4] = imhist(normalizatelambdaG2I)/(size(ROID.CU,1)*size(
    ROID.CU,2));
hs = [hs1;hs2;hs3;hs4];
h1csd = extract_csd(uint8(255*normalizatelambdaG1R),256);
h2csd = extract_csd(uint8(255*normalizatelambdaG2R),256);
h3csd = extract_csd(uint8(255*normalizatelambdaG1I),256);
h4csd = extract_csd(uint8(255*normalizatelambdaG2I),256);
hscsd = [h1csd h2csd h3csd h4csd];
if i > 4 && i <= 8
    if strcmp(Clase{j},'Confluencia')
        TestClases = [TestClases 1];
    end

    if strcmp(Clase{j},'Divergencia')
        TestClases = [TestClases 2];
    end
end

```

```

end

if strcmp(Clase{j},'Vorticidad')
    TestClases = [TestClases 3];
end

if strcmp(Clase{j},'Puntos de Silla')
    TestClases = [TestClases 4];
end
TestdataCoocurrenceMatrix = [TestdataCoocurrenceMatrix
    normag1cm(:)];
TestdataCoocurrenceMatrixNormalization = [
    TestdataCoocurrenceMatrixNormalization normag1cm1(:)
];
TestdatacsdMagnitud = [TestdatacsdMagnitud;csdROI];
TestdatacsdDiv = [TestdatacsdDiv;csdDiv];
TestdatacsdCurl = [TestdatacsdCurl;csdCurl];
Testlambdas = [Testlambdas dataset2];
TestLambdas = [TestLambdas;vec];
TestLambdas1 = [TestLambdas1;vect];
TestLambdas2 = [TestLambdas2;vec1];
TestLambdas3 = [TestLambdas3;vec3];
TestLambdaHistogram = [TestLambdaHistogram hs];
TestTraceDetHistogram = [TestTraceDetHistogram h];
TestcsdLambdaHistogram = [TestcsdLambdaHistogram;hscsd];
TestcsdTraceDetHistogram = [TestcsdTraceDetHistogram;
    hcsd];
else
if strcmp(Clase{j},'Confluencia')
    TrainClases = [TrainClases 1];
end

if strcmp(Clase{j},'Divergencia')
    TrainClases = [TrainClases 2];
end

if strcmp(Clase{j},'Vorticidad')
    TrainClases = [TrainClases 3];
end

if strcmp(Clase{j},'Puntos de Silla')

```

```

        TrainClases = [TrainClases 4];
    end
    TraindataCooccurrenceMatrix = [TraindataCooccurrenceMatrix
        normaglcm(:)];
    TraindataCooccurrenceMatrixNormalization = [
        TraindataCooccurrenceMatrixNormalization normaglcm1
        (:)];
    TraindatacsdMagnitud = [TraindatacsdMagnitud;csdROI];
    TraindatacsdDiv = [TraindatacsdDiv;csdDiv];
    TraindatacsdCurl = [TraindatacsdCurl;csdCurl];
    Trainlambdas = [Trainlambdas dataset2];
    TrainLambdas = [TrainLambdas;vec];
    TrainLambdas1 = [TrainLambdas1;vect];
    TrainLambdas2 = [TrainLambdas2;vec1];
    TrainLambdas3 = [TrainLambdas3;vec3];
    TrainLambdaHistogram = [TrainLambdaHistogram hs];
    TrainTraceDetHistogram = [TrainTraceDetHistogram h];
    TraincsdLambdaHistogram = [TraincsdLambdaHistogram;hscsd
    ];
    TraincsdTraceDetHistogram = [TraincsdTraceDetHistogram;
    hcsd];
end
% data = [data normaglcm(:)];
% data = [data normaglcm1(:)];
%Clases = {Clase{j}};
% dataCooccurrenceMatrix = [dataCooccurrenceMatrix dataset];
% dataCooccurrenceMatrixNormalization = [
    dataCooccurrenceMatrixNormalization dataset1];

end

end
%
save('Extraccion/dataTrain.mat','TrainClases',
    TraindataCooccurrenceMatrix',...
    'TraindataCooccurrenceMatrixNormalization','TraindatacsdCurl',
    TraindatacsdDiv',...

```



```

'TraindatacsdMagnitud','Trainlambdas','TrainLambdas','
    TrainLambdas1','TrainLambdas2','TrainLambdas3'...
,'TrainLambdaHistogram'...
,'TrainTraceDetHistogram','TraincsdLambdaHistogram','
    TraincsdTraceDetHistogram','TrainClases')
save('Extraccion/dataTest.mat','TestClases','
    TestdataCoocurrenceMatrix',...
'TestdataCoocurrenceMatrixNormalization','TestdatacsdCurl','
    TestdatacsdDiv',...
'TestdatacsdMagnitud','Testlambdas','TestLambdas','TestLambdas1
    ','TestLambdas2','TestLambdas3'...
,'TestLambdaHistogram'...
,'TestTraceDetHistogram','TestcsdLambdaHistogram','
    TestcsdTraceDetHistogram','TestClases')

```

C.3. Validación

Para la validación se uso las librerías de aprendizaje de máquina de MATLAB y algunas librerías extras como la LIBSVM

```

%% SVM Lineal Experimentation
clear all;clc;close all
% Load data
load('dataTest.mat')
load('dataTrain.mat')

[TestPyramidalVectorHistogramClass
    TrainPyramidalVectorHistogramClass...
    TestPyramidalVectorHistogram TrainPyramidalVectorHistogram] =
    generateData('PyramidalVectorHistogram.mat');
[TestPyrHOOFClass TrainPyrHOOFClass...
    TestPyrHOOF TrainPyrHOOF] = generateData1('pyrHOOF.mat');
[TestmaxPyramidalVectorHistogramPoolClass
    TrainmaxPyramidalVectorHistogramPoolClass...
    TestmaxPyramidVectorHistogramPool
    TrainmaxPyramidVectorHistogramPool] = generateData2('
    maxPyramidVectorHistogramPool.mat');
[TestmeanPyramidVectorHistogramPoolClass
    TrainmeanPyramidVectorHistogramPoolClass...

```

```

TestmeanPyramidVectorHistogramPool
    TrainmeanPyramidVectorHistogramPool] = generateData3('
        meanPyramidVectorHistogramPool.mat');
[TesthistHOOFFClass TrainhistHOOFFClass...
    TesthistHOOFF TrainhistHOOFF] = generateData4('histHOOFF.mat');
[TestmaxHOOFFClass TrainmaxHOOFFClass...
    TestmaxHOOFF TrainmaxHOOFF] = generateData2('maxHOOFF.mat');
% Validation
ResultSVMTrain = [];
partition = 'partition';
vect = 0.5:0.5:15;
for i = 0.5:0.5:15
    configuration = ['-s 0 -c ' num2str(i) ' -t 0 -h 0'];

    % Labels
    y = TrainClases';
    ya = TrainhistHOOFFClass';

    % Features
    X1 = TraindataCoocurrenceMatrix';
    X2 = TraindataCoocurrenceMatrixNormalization';
    X3 = TraindatacsdMagnitud;
    X4 = TraindatacsdDiv;
    X5 = TraindatacsdCurl;
    X6 = TrainLambdas;
    X7 = TrainLambdas1;
    X8 = TrainLambdaHistogram';
    X9 = TrainTraceDetHistogram';
    X10 = TrainhistHOOFF;
    X11 = TrainmaxHOOFF;
    X12 = TrainmaxPyramidVectorHistogramPool;
    X13 = TrainmeanPyramidVectorHistogramPool;
    X14 = TrainPyramidalVectorHistogram;
    X15 = TrainLambdas2;
    X16 = TraincsdLambdaHistogram;
    X17 = TraincsdTraceDetHistogram;
    X18 = TrainLambdas3;

    % Partitions
    cp = cvpartition(y,'k',5); % Stratified cross-validation
    cp1 = cvpartition(ya,'k',5); % Stratified cross-validation

```

```
f = @(xtr,ytr,xte,yte) confusionmatStats(yte,svmpredict(yte,xte
    ,svmtrain(ytr,xtr,configuration)));
% 5 - Fold Stratified cross-validation and confusion matrix
cfMat1 = crossval(f,X1,y,partition,cp);
cfMat2 = crossval(f,X2,y,partition,cp);
cfMat3 = crossval(f,X3,y,partition,cp);
cfMat4 = crossval(f,X4,y,partition,cp);
cfMat5 = crossval(f,X5,y,partition,cp);
cfMat6 = crossval(f,X6,y,partition,cp);
cfMat7 = crossval(f,X7,y,partition,cp);
cfMat8 = crossval(f,X8,y,partition,cp);
cfMat9 = crossval(f,X9,y,partition,cp);
cfMat15 = crossval(f,X15,y,partition,cp);
cfMat16 = crossval(f,X16,y,partition,cp);
cfMat17 = crossval(f,X17,y,partition,cp);
cfMat18 = crossval(f,X18,y,partition,cp);
cfMat10 = crossval(f,X10,ya,partition,cp1);
cfMat11 = crossval(f,X11,ya,partition,cp1);
cfMat12 = crossval(f,X12,ya,partition,cp1);
cfMat13 = crossval(f,X13,ya,partition,cp1);
cfMat14 = crossval(f,X14,ya,partition,cp1);
for ii = 1:length(cfMat1)
    d1(ii) = cfMat1(ii).accuracy;
    d2(ii) = cfMat2(ii).accuracy;
    d3(ii) = cfMat3(ii).accuracy;
    d4(ii) = cfMat4(ii).accuracy;
    d5(ii) = cfMat5(ii).accuracy;
    d6(ii) = cfMat6(ii).accuracy;
    d7(ii) = cfMat7(ii).accuracy;
    d8(ii) = cfMat8(ii).accuracy;
    d9(ii) = cfMat9(ii).accuracy;
    d10(ii) = cfMat10(ii).accuracy;
    d11(ii) = cfMat11(ii).accuracy;
    d12(ii) = cfMat12(ii).accuracy;
    d13(ii) = cfMat13(ii).accuracy;
    d14(ii) = cfMat14(ii).accuracy;
    d15(ii) = cfMat15(ii).accuracy;
    d16(ii) = cfMat16(ii).accuracy;
    d17(ii) = cfMat17(ii).accuracy;
    d18(ii) = cfMat18(ii).accuracy;
```

```

end
% mean Accuracy
resultTrain = [mean(d1) mean(d2) mean(d3) mean(d4) mean(d5)
              mean(d6) mean(d7)...
              mean(d8) mean(d9) mean(d10) mean(d11) mean(d12) mean(
                d13) mean(d14)...
              mean(d15) mean(d16) mean(d17) mean(d18)];

ResultSVMTrain = [ResultSVMTrain;resultTrain];
end

% Selection parameters
[posx posy val] = find(ResultSVMTrain == max(max(ResultSVMTrain)));

configuration = ['-s 0 -c ' num2str(vect(posx)) ' -t 0 -h 0'];

% Test and confusion Matrix
model1 = svmtrain(TrainClases',TraindataCoocurrenceMatrix',
                 configuration); %-g 0 -r 1 -d 3 ');
[predicted_label1, accuracy1, decision_values1] = svmpredict(
    TestClases', TestdataCoocurrenceMatrix', model1);
statSVM1 = confusionmatStats(TestClases,predicted_label1);

model2 = svmtrain(TrainClases',
                 TraindataCoocurrenceMatrixNormalization', configuration); %-g 0
                 -r 1 -d 3 ');
[predicted_label2, accuracy2, decision_values2] = svmpredict(
    TestClases', TestdataCoocurrenceMatrixNormalization', model2);
statSVM2 = confusionmatStats(TestClases,predicted_label2);

model3 = svmtrain(TrainClases', TraindatacsdMagnitud, configuration
                 ); %-g 0 -r 1 -d 3 ');
[predicted_label3, accuracy3, decision_values3] = svmpredict(
    TestClases', TestdatacsdMagnitud, model3);
statSVM3 = confusionmatStats(TestClases,predicted_label3);

model4 = svmtrain(TrainClases', TraindatacsdDiv, configuration); %-
                 g 0 -r 1 -d 3 ');
[predicted_label4, accuracy4, decision_values4] = svmpredict(
    TestClases', TestdatacsdDiv, model4);

```

```
statSVM4 = confusionmatStats(TestClasses,predicted_label4);

model5 = svmtrain(TrainClasses', TraindataCurl, configuration);
    %-g 0 -r 1 -d 3 ');
[predicted_label5, accuracy5, decision_values5] = svmpredict(
    TestClasses', TestdataCurl, model5);
statSVM5 = confusionmatStats(TestClasses,predicted_label5);

model6 = svmtrain(TrainClasses', TrainLambdas1, configuration); %-g
    0 -r 1 -d 3 ');
[predicted_label6, accuracy6, decision_values6] = svmpredict(
    TestClasses', TestLambdas1, model6);
statSVM6 = confusionmatStats(TestClasses,predicted_label6);

model7 = svmtrain(TrainClasses', TrainLambdas, configuration); %-g 0
    -r 1 -d 3 ');
[predicted_label7, accuracy7, decision_values7] = svmpredict(
    TestClasses', TestLambdas, model7);
statSVM7 = confusionmatStats(TestClasses,predicted_label7);

model8 = svmtrain(TrainClasses', TrainLambdaHistogram',
    configuration); %-g 0 -r 1 -d 3 ');
[predicted_label8, accuracy8, decision_values8] = svmpredict(
    TestClasses', TestLambdaHistogram', model8);
statSVM8 = confusionmatStats(TestClasses,predicted_label8);

model9 = svmtrain(TrainClasses', TrainTraceDetHistogram',
    configuration); %-g 0 -r 1 -d 3 ');
[predicted_label9, accuracy9, decision_values9] = svmpredict(
    TestClasses', TestTraceDetHistogram', model9);
statSVM9 = confusionmatStats(TestClasses,predicted_label9);

model10 = svmtrain(TrainhistHOOFFClass', TrainhistHOOFF,
    configuration); %-g 0 -r 1 -d 3 ');
[predicted_label10, accuracy10, decision_values10] = svmpredict(
    TesthistHOOFFClass', TesthistHOOFF, model10);
statSVM10 = confusionmatStats(TesthistHOOFFClass,predicted_label10);

model11 = svmtrain(TrainhistHOOFFClass', TrainmaxHOOFF, configuration
    ); %-g 0 -r 1 -d 3 ');
```

```

[predicted_label11, accuracy11, decision_values11] = svmpredict(
    TesthistHOOFFClass', TestmaxHOOFF, model11);
statSVM11 = confusionmatStats(TesthistHOOFFClass,predicted_label11);

model12 = svmtrain(TrainhistHOOFFClass',
    TrainmaxPyramidVectorHistogramPool, configuration); %-g 0 -r 1
    -d 3 ');
[predicted_label12, accuracy12, decision_values12] = svmpredict(
    TesthistHOOFFClass', TestmaxPyramidVectorHistogramPool, model12)
;
statSVM12 = confusionmatStats(TesthistHOOFFClass,predicted_label12);

model13 = svmtrain(TrainhistHOOFFClass',
    TrainmeanPyramidVectorHistogramPool, configuration); %-g 0 -r 1
    -d 3 ');
[predicted_label13, accuracy13, decision_values13] = svmpredict(
    TesthistHOOFFClass', TestmeanPyramidVectorHistogramPool, model13
);
statSVM13 = confusionmatStats(TesthistHOOFFClass,predicted_label13);

model14 = svmtrain(TrainhistHOOFFClass',
    TrainPyramidalVectorHistogram, configuration); %-g 0 -r 1 -d 3
    ');
[predicted_label14, accuracy14, decision_values14] = svmpredict(
    TesthistHOOFFClass', TestPyramidalVectorHistogram, model14);
statSVM14 = confusionmatStats(TesthistHOOFFClass,predicted_label14);

model15 = svmtrain(TrainClasses', TrainLambdas2, configuration); %-g
    0 -r 1 -d 3 ');
[predicted_label15, accuracy15, decision_values15] = svmpredict(
    TestClasses', TestLambdas2, model15);
statSVM15 = confusionmatStats(TestClasses,predicted_label15);

model16 = svmtrain(TrainClasses', TraincsdLambdaHistogram,
    configuration); %-g 0 -r 1 -d 3 ');
[predicted_label16, accuracy16, decision_values16] = svmpredict(
    TestClasses', TestcsdLambdaHistogram, model16);
statSVM16 = confusionmatStats(TestClasses,predicted_label16);

model17 = svmtrain(TrainClasses', TraincsdTraceDetHistogram,
    configuration); %-g 0 -r 1 -d 3 ');

```

```

[predicted_label17, accuracy17, decision_values17] = svmpredict(
    TestClases', TestcsdTraceDetHistogram, model17);
statSVM17 = confusionmatStats(TestClases,predicted_label17);

model18 = svmtrain(TrainClases', TrainLambdas3, configuration); %-g
    0 -r 1 -d 3 ');
[predicted_label18, accuracy18, decision_values18] = svmpredict(
    TestClases', TestLambdas3, model18);
statSVM18 = confusionmatStats(TestClases,predicted_label18);

% Results
PrecisionTestSVM = [statSVM1.precision' statSVM2.precision'
    statSVM3.precision'...
    statSVM4.precision' statSVM5.precision' statSVM6.precision'
    statSVM7.precision'...
    statSVM8.precision' statSVM9.precision' statSVM10.precision'
    statSVM11.precision'...
    statSVM12.precision' statSVM13.precision' statSVM14.precision'
    statSVM15.precision'...
    statSVM16.precision' statSVM17.precision' statSVM18.precision
    '];

AccuracyTestSVM = [statSVM1.accuracy statSVM2.accuracy statSVM3.
    accuracy...
    statSVM4.accuracy statSVM5.accuracy statSVM6.accuracy statSVM7.
    accuracy...
    statSVM8.accuracy statSVM9.accuracy statSVM10.accuracy
    statSVM11.accuracy...
    statSVM12.accuracy statSVM13.accuracy statSVM14.accuracy
    statSVM15.accuracy...
    statSVM16.accuracy statSVM17.accuracy statSVM18.accuracy];

AVGAccuracyTestSVM = [avgAccuracy(statSVM1) avgAccuracy(statSVM2)
    avgAccuracy(statSVM3)...
    avgAccuracy(statSVM4) avgAccuracy(statSVM5) avgAccuracy(
    statSVM6) avgAccuracy(statSVM7)...
    avgAccuracy(statSVM8) avgAccuracy(statSVM9) avgAccuracy(
    statSVM10) avgAccuracy(statSVM11)...
    avgAccuracy(statSVM12) avgAccuracy(statSVM13) avgAccuracy(
    statSVM14) avgAccuracy(statSVM15)...

```

```
avgAccuracy(statSVM16) avgAccuracy(statSVM17) avgAccuracy(  
    statSVM18)]  
  
save('ResultsSVM.mat', 'ResultSVMTrain', 'PrecisionTestSVM', '  
    AccuracyTestSVM', 'AVGAccuracyTestSVM', 'configuration')
```


Apéndice D

Publicaciones

A lo largo del proyecto se logró publicar el siguiente Artículo:

D.1. Artículo publicado

Classification of low-level atmospheric structures based on a pyramid representation and a machine learning method

Sebastián Sierra*², Juan F. Molina*¹, Angel Cruz-Roa², José D. Pabón, Raúl Ramos-Pollán³, Fabio A. González², and Hugo Franco¹

¹ Computer Eng. Dept., Universidad Central, Bogotá D.C. 110321, Colombia,
hfrancot@ucentral.edu.co,

² MindLab Research Group, Universidad Nacional de Colombia,
Ciudad Universitaria, 111321, Bogotá, Colombia

³ Centre for Supercomputing and Scientific Computing, Universidad Industrial de Santander, Bucaramanga, Colombia

Abstract. The atmosphere is a highly complex fluid system where multiple intrinsic and extrinsic phenomena superpose at same spatial and temporal dominions and different scales, making its characterization a challenging task. Despite the novel methods for pattern recognition and detection available in the literature, most of climate data analysis and weather forecast rely on the ability of specialized personnel to visually detect atmospheric patterns present in climate data plots. This paper presents a method for classifying low-level wind flow configurations, namely: confluences, diffluences, vortices and saddle points. The method combines specialized image features to capture the particular structure of low-level wind flow configurations through a pyramid layout representation and a state-of-the-art machine learning classification method. The method was validated on a set of volumes extracted from climate simulations and manually annotated by experts. The best results into the independent test dataset was 0.81 of average accuracy among the four atmospheric structures.

1 Introduction

The superposition of an intricate set of interactions between air, ocean and land components and states yields both regular (periodic or quasiperiodic) and chaotic patterns at different scales for atmospheric relevant variables –i.e. temperature, pressure, humidity, air velocity, density, chemical composition, etc. The dynamics of such patterns and their corresponding physical manifestations (cloud coverage, winds, rain, hail, snow, etc.) configure a continuous from climate (long time and space atmospheric scales) to weather (in short time scale and local domains).

Behind the specific occurrence of each observable atmospheric pattern, there are several physical processes converging at different spatial and temporal scales, determining their distinctive features. Indeed, the relevant structures for both

climate and weather description are a compound of several atmospheric variable configurations present in a particular region at the same time.

Weather forecast is carried out by the coupling of mechanistic models, such as WRF (Weather Research and Forecasting Model [6]) and statistical processing of real data. Furthermore, computer simulations and data acquisition (satellite imagery, weather stations, RADAR) are just the first part of the process: a detailed empirical description from classical meteorological approaches allow specialists to visually detect atmospheric structures and manually track their evolution over time to *qualitatively* describe their dynamics and, then, predict the most probable scenarios according to each particular spatio-temporal configuration obtained [7]. Then, low level structure characterizations of the atmosphere are implicitly carried out by meteorologists on a daily basis, with the aid of computer systems that enhance data visualization (isolines, streamlines, data fusion on scalar and vector fields, etc.).

Moreover, the automatic detection of atmospheric structures, even for *low level* configurations (particularly, fluid flow patterns) is a challenging problem, given the high local and global variability of climatic data, no matter if it comes from experimental measurements or modeling and simulation results. Recent works are still exploring new techniques for structure representation and identification [4].

In the data visualization context, the works of Tzeng and Ma [10] and Gruchalla et al. [3] proposed frameworks for 3D fluid structure extraction and rendering by detecting regions of interests (ROIs) containing such patterns in the wavelet domain. Given that low level fluid structures are expected to have quite regular patterns, Rao et al. [9] presented a region-based detection and extraction method based in phase portraits, where the salient ROIs are those where the geometrical pattern of the vector field best matches the solution of the differential equation associated to the analytic structure description. Even the use of such methods in climate and weather data visualization, state of the art climate data analysis and weather forecast depends mainly on the ability and experience of the specialist to detect patterns constituting relevant atmospheric structures.

This work proposes a machine learning approach to support the automatic detection of such *low level* patterns, specifically those related to wind dynamics. The method takes as input a volume, which includes information of atmospheric variables (temperature, pressure, humidity and air velocity) at a given time in a particular region; next, different features based on divergence and curl differential operators are extracted from the volume represented as a spatial pyramid layout (SPL) and fed to a support vector machine classifier, which has been previously trained with a set of annotated volumes for four classes: confluence, diffluence, vortex and saddle point (as it is shown in Figure 1); finally, the classifier outputs a prediction that indicates the probability of the presence of a particular low-level wind flow configuration. The method was evaluated over a set of volumes extracted from climate simulations and manually annotated by expert meteorologists.

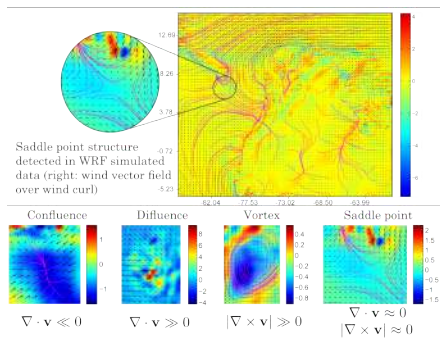


Fig. 1. Representative instances of confluence, diffluence, vortex and saddle point wind structures in WRF simulated data for northern South America and the Caribbean. Winds (in m/s) are plotted over this geographical domain and its corresponding latitude and longitude coordinates (upper map) for a spatial resolution of 40 km.

The rest of paper is organized as follows: Section 2 explains the details of the proposed classification method. Section 3 describes the experimental evaluation—including the dataset description, the experimental setup and results—. Finally, Section 4 concludes with the main remarks and establishes the future work.

2 Automatic classification of *low level* atmospheric structures

An overview of the method herein proposed is depicted in Figure 2, which starts from a database of *low level* atmospheric structures manually annotated by experts meteorologists. For each annotated sample, an approximation of the differential operators curl and divergence is applied to capture characteristics of the wind velocity vector field into three atmospheric feature maps (curl, divergence, negative of divergence). Then, several histogram representations are built at different resolutions following a SPL. The final concatenation of the resulting histograms is used to train a machine learning classifier (support vector machine, SVM, or random forest, RF) to distinguish between four atmospheric structures (vortex, diffluence, confluence and saddle point). The prediction is performed using the trained classifier over a particular ROI. The details of each step are presented in the following subsections.

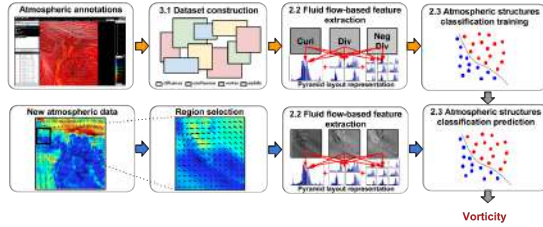


Fig. 2. Overall method description for automatic detection of atmospheric structures.

2.1 Approximation of differential operators from wind velocity field

Given $\nabla = \left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y} \right)$ and $\mathbf{v} = (u, v)$, a vector field representing two dimensional components of winds *parallel* to Earth surface within an isobaric layer, it is possible to derive expressions for both *divergence* and *curl* operators, to support further design of a fluid flow-based vector field descriptor. A simple discrete version of a two-dimensional \mathbf{v} , is the matrix $V = \{(u, v)_{ij}\}_{M \times N}$ where $i = 1, 2, \dots, M$ is the index pointing to each row and $j = 1, 2, \dots, N$ the index pointing to each column of the matrix.

Divergence. The divergence of \mathbf{v} is defined as $\nabla \cdot \mathbf{v} = \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}$ for the two-dimensional case. Since this is an interior product, divergence is a scalar field. Using the basic centered finite difference approximation to each partial derivative, the 2D divergence can be estimated as $\nabla \cdot \mathbf{v} \approx (u_{i+1,j} - u_{i-1,j})/2h + (v_{i,j+1} - v_{i,j-1})/2h$.

Curl. The curl of a vector field determines how much the vectors within the dominion under study “rotate” around each particular position. In the 2D case, curl is easily build by taking into account only the rotation on the xy plane. Then $\nabla \times \mathbf{v} = \left(\frac{\partial v}{\partial x} - \frac{\partial u}{\partial y} \right) \hat{\mathbf{z}}$, which, therefore, is always perpendicular to that plane and can be approximated by $\nabla \times \mathbf{v} \approx (v_{i+1,j} - v_{i-1,j})/2h - (u_{i,j+1} - u_{i,j-1})/2h$.

2.2 Fluid flow-based feature descriptor

Taking into account that an atmospheric structure can occur in different spatial locations and each phenomenon can have different scales, we propose a set of atmospheric-based feature descriptors able to support translational and scale invariance by using a SPL representation of the region of interest. Over the SPL, we extract features from the differential curl and divergence operators, max-pooling and the histogram of oriented optical flow as explained below.

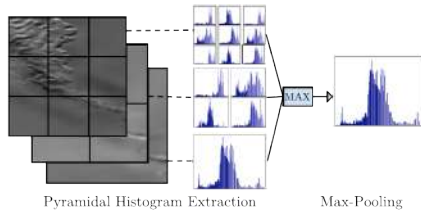


Fig. 3. Fluid flow-based feature descriptors. Histograms are extracted in a pyramid layout from the different feature maps. The resulting 14 histograms are further integrated in the max-pooling descriptor, which corresponds to a histogram with the maximum values for all the bins.

Spatial pyramid layout (SPL) representation This representation allows to capture the atmospheric structures in different spatial locations and scales. As it is shown in Figure 3, the image is divided in different regions following a pyramidal layout organization. The first layer of the pyramid corresponds to the whole image, the second to a 2×2 division, and the third to a 3×3 division [5]. The image is represented by a concatenation of the resulting 14 normalized histograms for divergence, negative-divergence and curl feature maps.

Max-pooling The max-pooling works to detect atmospheric structure independently of its location and scale. Max-pooling applies a maximum operator among the 14 histograms of all spatial layouts. This pooling function is typically used for detection tasks exploiting spatial invariances [8].

Histogram of oriented optical flow As a baseline, we decided to use histograms of oriented optical flow (HOOF) [2] attempting to detect the patterns in the wind flow that better describe each phenomena. Additionally, the SPL was also applied to this image representation.

2.3 Atmospheric structure classifier

For automatic prediction of the atmospheric structures, a Support Vector Machine (SVM) classifier and a Random Forest (RF) classifier were trained. For addressing the multiclass problem of classifying the atmospheric structures (vortex, diffluence, confluence and saddle points), the SVM uses a one-vs-all strategy. The kernels evaluated for SVM were linear and radial basis function (RBF).

3 Experimental Evaluation

3.1 Atmospheric Structure Dataset construction

The 4D (3D + time) data used in this study come from WRF regional climate simulations from the climate change study developed by Armenta and Pabón [1]

for the period between 1981 and 2010. These simulations were performed at a spatial resolution of $h = 10$ km and temporal resolution of 3 hours.

A monthly highest and lowest ocean surface temperature criterion was applied to focus on May and December data. An Extraction-Transformation and Load (ETL) process was performed to extract and transform to real units 3D fields for temperature, T , and wind velocity components, U , V and W , and then store them in raw files. For the region under study (northern South America and western Caribbean Sea), 27 isobaric slices were extracted, each of 287×280 uniform surface elements.

An annotation tool was developed specifically for these data. The tool loads simulation data and renders temperature and wind velocity for slices at different isobaric levels (up to 27 for each volume). Experts navigate the WRF data using the annotation software by exploring the 3D volumes along a specific time period, and are asked to manually segment the ROI containing one structure and to specify its particular type. The ROI bounding box is then stored in a XML file along with the corresponding class and related information, such as simulation time, isobaric level and divergence and rotational computations on the ROI. The resulting annotated dataset contains 793 annotations, 232 for confluences, 166 for divergences, 177 for saddles and 218 for vortices.

3.2 Experimental design and performance measures

For evaluation purposes, we split the original dataset into two parts for training and testing. The testing dataset comprises a complete and independent 4D data from a simulation of a month with region annotations for all four climate phenomena (vortex = 78, diffluence = 47, confluence = 39, saddles = 35). The training set corresponds to remaining simulation 4D data with annotations from four months (vortex = 140, diffluence = 119, confluence = 193, saddles = 142). The performance measure used to evaluate the classification was the average accuracy, i.e. the average of the accuracy per class.

For parameter selection, a stratified 5-fold cross-validation scheme was applied over the training dataset for each strategy (combinations of features and classifiers). The same folds distribution was used for all strategies in order to compare them in the same conditions. The parameter combination that obtained the best performance was used to train a model using the whole training dataset. The final performance measure was reported over the independent testing dataset for each strategy for comparison.

The representations evaluated were defined as follows: *pyrHOOF* (baseline) by concatenating the HOOF descriptor for each region obtained from SPL, *histdivcurl* is the concatenation of histogram distribution of the three feature maps of atmospheric operators (curl, divergence and negative divergence), *pyrdivcurl* is the histogram concatenation of the three feature maps for each region obtained from SPL, and *maxpooldivcurl* is the corresponding histogram obtained by applying max-pooling over all concatenated histograms from regions obtained from the SPL.

3.3 Results

Table 1 shows the classification performance of the proposed method evaluated in terms of accuracy for the different representations and classification methods. Notice that these results were obtained from independent data corresponding to a month of simulation. The best performance measure was achieved by the complete SPL representation without max-pooling (*pyrdivcurl*) and using a Support Vector Machine with an RBF kernel obtaining an average accuracy of 0.81. In general, the confluence class shows the highest accuracy for the most configurations whereas saddle point class have the worst performance. Interestingly max-pooling applied over SPL does not help because most of the region annotations are centered in the atmospheric structure. However, preliminary work with larger regions where the atmospheric structure was not at the center, the max-pooling over SPL obtained the best results. Table 1 also shows that performance achieved using SVM with a linear kernel is very close to the reported one using SVM with RBF kernel. Furthermore Random Forest classifiers were trained with 10,000 estimators, indeed during cross validation we could determine that increasing the number of estimators did not improve our results.

Table 1. Classification performance in test dataset in terms of accuracy.

Feature	Classifier	Vortex	Diffuence	Confluence	Saddle pt	Avg. Accuracy
pyrdivcurl	SVM-RBF	0.71	0.82	0.92	0.77	0.81
histdivcurl	SVM-RBF	0.69	0.87	0.89	0.6	0.765
maxpooldivcurl	SVM-RBF	0.69	0.87	0.89	0.6	0.765
pyrHOOF	SVM-RBF	0.76	0.87	0.56	0.48	0.672
pyrdivcurl	SVM-Linear	0.74	0.8	0.92	0.71	0.797
histdivcurl	SVM-Linear	0.75	0.78	0.87	0.6	0.753
maxpooldivcurl	SVM-Linear	0.75	0.78	0.87	0.6	0.753
pyrdivcurl	RF	0.73	0.8	0.94	0.6	0.771
maxpooldivcurl	RF	0.69	0.85	0.92	0.51	0.745
histdivcurl	RF	0.69	0.85	0.92	0.45	0.73
pyrHOOF	RF	0.62	0.97	0.58	0.34	0.634

4 Concluding remarks

This paper presents a successful application of a novel method which combines features based on differential operators and machine learning classifiers to discriminate *low level* wind structures. The feature maps obtained from approximate differential operators help to highlight the relevant atmospheric structures. The SPL show the best classification performance since it includes multi-resolution information, despite the complexity of the structures.

Future work includes to increase the dataset with more annotations of *low level* and new *high level* atmospheric structures, adding more atmospheric variables, enhancing the *low level* structure characterization, as well as to develop a detection method to efficiently analyze whole 4D volumes.

5 Acknowledgements

This work was funded by project number 1225-569-34920 “*Diseno e implementación de un sistema de cómputo sobre recursos heterogéneos para la identificación de estructuras atmosféricas en predicción climatológica*” through Administrative Department of Science, Technology and Innovation of Colombia (Colciencias). We want to thank Diana Díaz and Darwin Martínez for their valuable hints and discussions.

References

1. Armenta, G., Pabón, J.: Modeling northern South America and Caribbean climate using PRECIS and WRF for climate variability and change studies. In: Proceedings of the CORDEX-LAC1 Workshop – World Climate Research Programme, Lima, Peru (2013)
2. Chaudhry, R., Ravichandran, A., Hager, G., Vidal, R.: Histograms of oriented Optical Flow and Binet-Cauchy kernels on nonlinear dynamical systems for the recognition of human actions. In: Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on. (June 2009) 1932–1939
3. Gruchalla, K., Rast, M., Bradley, E., Clyne, J., Mininni, P.: Visualization-driven structural and statistical analysis of turbulent flows. In: Advances in Intelligent Data Analysis VIII. Springer (2009) 321–332
4. Holmén, V.: Methods for vortex identification. Master’s thesis, Lund University (2012)
5. Lazebnik, S., Schmid, C., Ponce, J.: Beyond Bags of Features: Spatial Pyramid Matching for recognizing natural scene categories. In: Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on. Volume 2. (2006) 2169–2178
6. Michalakes, J., Chen, S., Dudhia, J., Hart, L., Klemp, J., Middlecoff, J., Skamarock, W.: Development of a next generation regional weather research and forecast model. In: Developments in Teracomputing: Proceedings of the Ninth ECMWF Workshop on the use of high performance computing in meteorology. Volume 1., World Scientific (2001) 269–276
7. Murphy, A.H.: What is a good forecast? an essay on the nature of goodness in weather forecasting. *Weather and forecasting* **8**(2) (1993) 281–293
8. Nagi, J., Ducatelle, F., Di Caro, G., Ciresan, D., Meier, U., Giusti, A., Nagi, F., Schmidhuber, J., Gambardella, L.: Max-pooling convolutional neural networks for vision-based hand gesture recognition. In: Signal and Image Processing Applications (ICSIPA), 2011 IEEE International Conference on. (Nov 2011) 342–347
9. Rao, A.R., Jain, R.C.: Computerized flow field analysis: Oriented texture fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **14**(7) (1992) 693–709
10. Tzeng, F.Y., Ma, K.L.: Intelligent feature extraction and tracking for visualizing large-scale 4d flow simulations. In: Proceedings of the 2005 ACM/IEEE conference on Supercomputing, IEEE Computer Society (2005) 6