

UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS
Facultad 9

*TÍTULO: Representación de Modelos Digitales
del Terreno en 3D sobre la Web.*

TRABAJO DE DIPLOMA PARA OPTAR POR EL TÍTULO DE INGENIERO EN CIENCIAS
INFORMÁTICAS

AUTOR: Yennis Muñoz Aldana

*TUTOR: Ing. Rolando Toledo Fernández
Ing. Antonio Membrides Espinosa*

RESUMEN

El siguiente trabajo tiene como objetivo fundamental la propuesta de un método que permita realizar la representación de Modelos Digitales del Terreno en 3D sobre la Web, en la plataforma GeneSIG del departamento Geoinformática del centro GEySED. Los Modelos Digitales del Terreno permiten la representación digital de la topografía de un territorio y una especialización de estos se denomina Modelos Digitales de Elevaciones, permitiendo identificar diferentes datos de interés por concepto de elevaciones, tales como cuencas visuales, curvas de nivel, entre otros.

Para generar estos modelos es necesario de una muestra del terreno a visualizar, de donde se extraen los puntos de interés para cargar en memoria una estructura de datos espaciales a través de algoritmos de triangulación, cada uno de los identificados en la representación cartográfica.

A lo largo de la investigación se estudiaron diferentes modelos, métodos matemáticos, algoritmos y herramientas que de alguna forma permitieran generar, representar y visualizar estos modelos en 3D, para obtener una propuesta a ser aplicada en el proyecto que permita la representación de superficies del terreno de forma eficiente en la Web.

PALABRAS CLAVES

Modelos Digitales del Terreno, GeneSIG, GEySED, Modelos Digitales de Elevaciones.

Introducción.....	1
CAPÍTULO 1: Fundamentación Teórica	6
1.1 Introducción.....	6
1.2 Conceptos asociados al dominio del problema	6
1.3 Proceso de representación de los Modelos Digitales del Terreno en 3D sobre la Web.....	8
1.4 Situación problemática	9
1.5 Modelos Digitales del Terreno	10
1.5.1 MDE	10
1.5.2 Cuencas visuales, visualización del relieve y modelos de reflectancia	12
1.5.3 Líneas de flujo, Cuencas hidrológicas y Modelo de caudales máximos.....	14
1.5.4 Sombras, Insolación y Modelo de irradiancia.....	16
1.6 Otras soluciones existentes.....	19
1.7 Conclusiones.....	22
CAPÍTULO 2: Tendencias y Tecnologías para la representación de MDT	23
2.1 Introducción.....	23
2.2 Métodos Matemáticos para representar MDT	24
2.2.1 Modelo de Spline Bilineal.....	24
2.2.2 Modelo de Spline bicuadráticas.....	25
2.2.3 Modelo de Spline bicúbico.....	26
2.3 Algoritmos para la representación y visualización de los MDT	30
2.3.1 Quadtree.....	32
2.3.2 Mallas Regulares R-Trianguladas y Bitrees no Restringidos.....	35
2.3.3 Mallas Adaptables Óptimas en Tiempo Real.....	36
2.3.4 Geomipmapping.....	43
2.3.5 Hilbert-Quadtree.....	45
2.4 Herramientas y tecnologías.....	48
2.4.1 Librerías y Motores en JavaScript.....	48
2.4.2 Herramientas para Adobe Flash	51
2.4.3 Plugin para navegadores.....	54

2.4.4 Tecnologías.....	55
2.5. Estándares para la representación en 3D.....	61
2.5.1 VRML.....	61
2.5.2 JAVA3D.....	61
2.5.3 X3D	62
2.5.4 Comparación entre VRML/X3D y Java3D	63
CAPÍTULO 3: Propuesta de la solución	65
3.1 Introducción.....	65
3.2 Justificación de la solución propuesta	65
3.3 Algoritmos.....	66
3.4 Herramientas y tecnologías	67
3.5 Conclusiones parciales	71
Conclusiones.....	72
Recomendaciones.....	72
Referencias Bibliográficas	73
Bibliografía.....	75

Introducción

El mundo actual se encuentra en constante cambio, la necesidad humana de vivir en mejores condiciones le ha dado lugar al surgimiento de las TIC¹. Esta tecnología desempeña un papel importante en la sociedad, donde se hace necesario un mejor manejo y gestión de la información, así como la informatización de diferentes sectores de la misma. La informática constituye la ciencia fundamental del desarrollo de las TIC y los avances de estas han sido muy utilizadas para crear herramientas que faciliten un mejor almacenamiento y control de los datos. Algunas de estas herramientas son muy utilizadas en nuestros días como por ejemplo los SIG².

El tratamiento geográfico de la información se ha vuelto imprescindible y sus posibilidades de aplicación se hacen cada vez mayor gracias a los SIG, los cuales permiten llevar a cabo la representación cartográfica, realizar evaluaciones ambientales y de recursos naturales, estudiar y evaluar las redes de servicios de electricidad, telefonía, emergencias médicas, transportes, la ubicación geoespacial, entre otras. El impacto de los SIG ha logrado la versatilidad y desarrollo en cuanto a la toma de decisiones en las diferentes esferas de la sociedad, permitiendo que la herramienta cumpla con las funcionalidades atribuidas y ofrezca los servicios de georeferenciación y análisis de los datos según la entidad que lo utilice.

En 1950 surge el software de cartografía automatizada CAD³, CAM⁴ y las primeras bases de datos para tener un control de datos de los atributos en la computadora; pero no fue hasta 1960 que surgen sistemas que permitían integrar las bases de datos con las figuras, que más adelante se fueron ampliando para un mejor manejo de la información. El primer SIG utilizado y que mostró eficiencia fue el SIG-Canadá, con el objetivo de almacenar, analizar y manipular datos relacionados con la información cartográfica de este país.

Los SIG han alcanzado mayor aceptación y uso en la sociedad en cuanto al tratamiento de datos espaciales. La mayor parte de las representaciones que se realizan teniendo en cuenta los datos

¹ **TIC:** acrónimo de Tecnologías de la Información y las Comunicaciones

² **SIG:** acrónimo de Sistemas de Información Geográfica

³ **CAD:** acrónimo de *Computer Aided Design*

⁴ **CAM:** acrónimo de *Computer Aided Modeling*

espaciales se desarrollan en 2D⁵; sin embargo el amplio mundo de la computación ha llegado más allá de una simple representación en dos dimensiones, logrando una mejor visualización con herramientas realizadas en 3D⁶. El uso de 3D en los SIG, permite una mayor libertad en cuanto a la visualización de datos globales y espaciales. La diferencia entre las dos representaciones es que a través de tres dimensiones, se puede ilustrar de forma más definida los rasgos topográficos o variabilidad espacial de los datos, y el usuario puede tener una mejor imagen de las cualidades y atributos del objeto representado.

Los SIG permiten llevar a cabo la cartografía digital, obteniendo los datos para el desarrollo de los MDT⁷, poseen un gran número de aplicaciones entre ellas: levantamiento de mapas geográficos, la ingeniería civil, la planificación urbana y regional y en actividades militares. El objetivo del modelado que se realiza, es crear una similitud entre el modelo y el objeto, llevando a dicho modelo la propiedad que en un futuro permita reconstruir esa parte de la realidad que se quiere representar del terreno. Los MDT junto con los SIG son un factor importante para el conocimiento de la información geográfica a través de mapas y representaciones, hechas de distintas partes del mundo para la realización y análisis geográficos del mismo.

La inclusión de los MDT como soporte básico en la investigación de las ciencias ambientales supone un incremento notable de las posibilidades de trabajo e introduce frecuentemente un cambio cualitativo en el enfoque del mismo. (1)

En la UCI⁸ se desarrolla la plataforma GeneSIG la cual forma parte del departamento GeoInformática del centro GEySED de la facultad 9, donde se crean proyectos productivos relacionados con los sistemas geográficos. Surgió debido a la necesidad de crear un software que utilizara tecnologías libres, y que permitiera cumplir con la política del estado de migrar a software libre. La plataforma se desarrolla para ser utilizada como soporte para aplicaciones de Sistemas de Información Geográfica. Actualmente GeneSIG desarrolla productos en 2D, por lo que su representación está limitada y las posibilidades de visualizar con más detalle el terreno son menores debido a que la misma no permite representar un tercer dato en las

⁵ **2D**: acrónimo de Dos Dimensiones

⁶ **3D**: acrónimo de Tres Dimensiones

⁷ **MDT**: acrónimo de Modelos Digitales del Terreno

⁸ **UCI**: acrónimo de Universidad de las Ciencias Informáticas

coordenadas. El colectivo de trabajo no cuenta con el conocimiento para llevar estas aplicaciones a 3D y de esta manera poder visualizar la altura de la superficie. Se desconocen los algoritmos y herramientas que se puedan utilizar para una mejor visualización de los componentes. La plataforma se desarrolla sobre la Web, sin embargo, los desarrolladores no poseen ningún tipo de información que les permita llevarla a una representación en la Web pero en 3D. Además, el colectivo de trabajo no posee el tiempo suficiente para llevar a cabo la investigación y desarrollo de un método que le permita esta representación.

Debido a lo planteado anteriormente se tiene como **Problema a Resolver**: ¿Cómo representar Modelos Digitales del Terreno en 3D sobre la plataforma GeneSIG?, Teniendo como **Objetivo General**: Documentar las técnicas y algoritmos necesarios para la representación de MDT en 3D sobre la Web; y de este objetivo se marcaría el **Objeto de Estudio** como: Proceso de representación de los MDT en 3D sobre la Web.

Enmarcado en el **Campo de Acción**: Algoritmos y herramientas que permitan la Representación de los MDT en 3D sobre la Web. Luego de los objetivos trazados se tiene como **Idea a defender**: Una vez obtenida una propuesta de las técnicas y algoritmos para la representación de MDT en 3D sobre la Web, se potenciará el módulo de análisis de la plataforma GeneSIG, logrando una mejor visualización de los datos espaciales.

Para cumplir el objetivo general se trazaron las siguientes **Tareas de investigación**:

- Describir los MDT en 3D.
- Delimitar el estado del arte de la representación de los Modelos Digitales del Terreno en 3D sobre la Web.
- Describir los modelos Matemáticos para la representación de los MDT.
- Caracterizar los algoritmos informáticos y las estructuras Geoespaciales que existen dirigidos a la representación de dichos modelos.
- Comparar las herramientas que permiten la representación de los MDT en 3D sobre la Web.
- Identificar las posibles herramientas a utilizar en la plataforma, que permitan la representación de dichos modelos.
- Comparar los estándares para la generación y representación de MDT en 3D.

- Definir los estándares a utilizar para la representación de dichos modelos.
- Identificar posibles mejoras para las herramientas, complementándolas con los algoritmos seleccionados.
- Documentar la investigación con vistas a su socialización.

Luego de haber realizado todas las tareas se tendrá como resultado:

Toda la documentación necesaria para la futura representación de MDT en la plataforma GeneSIG en 3D sobre la Web.

Para darle cumplimiento a las tareas especificadas se emplearon los siguientes métodos científicos:

Métodos teóricos.

Análítico – sintético: Este método da la posibilidad del estudio e investigación de las herramientas y algoritmos que permitan una modelación, seleccionando lo mejor de cada uno, basándose en la documentación existente.

Análisis histórico – lógico: Los cuales permiten el estudio de las soluciones existentes ya sea en otro país o en Cuba que puedan servir de base y ayuda para llegar a la conclusión final. Realizando un estudio de las diferentes herramientas que existen en el mundo y la evolución de estos métodos.

Inductivo – Deductivo: La aplicación de este método permite que se pueda llegar a la conclusión de que es lo mejor y cómo llegar a un resultado final para la modelación digital de terrenos sobre la Web y así poder utilizarlo en la plataforma GeneSIG.

Métodos empíricos.

Entrevista: A través de las entrevistas que serán realizadas a lo largo del trabajo, a personas que posean algún tipo de conocimiento que pueda servir de ayuda para la realización del mismo.

Para realizar este método se escogieron 2 personas para realizarle la entrevista.

Máster Yusnier Valle Martínez

Ing. Dwuan Vázquez Rodríguez

El documento está estructurado en 3 capítulos:

Capítulo 1. Fundamentación Teórica: en este capítulo se hace mención a los conceptos asociados al dominio del problema. Se describe de forma clara y sencilla la situación problemática, se argumenta el objeto de estudio y se mencionan algunos de los tipos de MDT.

Capítulo 2. Tendencias y Tecnologías para la representación de MDT: en el capítulo se abordará sobre los algoritmos, herramientas o técnicas que son usados para la representación de MDT, se hará un análisis de las utilidades de cada uno, para escoger uno o la combinación de ambos que permita llevar a cabo la representación en la plataforma GeneSIG.

Capítulo 3. Propuesta de la solución: en este capítulo se dará una breve explicación del porque de la solución, así como las desventajas por las cuales fueron descartados los otros métodos dentro de la investigación.

CAPÍTULO 1: Fundamentación Teórica

1.1 Introducción

La representación de la geografía es algo imprescindible en cualquier país, para ello se utilizan los SIG, que proveen de toda la información necesaria para realizar múltiples aplicaciones, que junto con los MDT permiten la visualización de cualquier parte del mundo.

En el capítulo se aborda sobre los MDT, haciendo un pequeño énfasis en los SIG. Se dará una breve explicación de los diferentes MDT. Abordando conceptos como por ejemplo: que es una curva de nivel, estructura de datos, cartografía entre otros de importancia para la investigación.

1.2 Conceptos asociados al dominio del problema

Modelo: Resultado del proceso para definir los fenómenos del mundo real o las características geográficas de interés en términos de sus atributos y sus relaciones. (29)

Modelo es una representación simplificada de la realidad en la que aparecen algunas de sus propiedades.

Cuenca: Se entiende por cuenca u hoya hidrográfica, el área de aguas superficiales o subterráneas, que vierten a una red natural con uno o varios cauces naturales, de caudal continuo o intermitente, que se unen en un curso mayor que, a su vez, puede desembocar en un río principal, en un depósito natural de aguas, en un pantano o directamente en el mar. (30)

Representación en 3D: Es la representación de un objeto de visualización en 3 dimensiones, que pueden ser el largo, ancho y profundidad de esta, proyectado en una pantalla bidimensional. Los objetos en 3D permiten una movilidad que no se puede realizar en 2D. (2)

La principal diferencia entre un objeto 2D y otro 3D proyectado en una pantalla bidimensional es la adición de una tercera dimensión en el objeto. Esta permite que el objeto se acerque y se aleje del punto de vista del usuario.

SIG: sistema informático que facilita la introducción, almacenamiento, recuperación, transformación, modelado, análisis, representación y salida de datos espaciales y tabulares del mundo real, para garantizar la toma de decisiones en problemas de planificación y gestión. (25)

Se puede definir como una tecnología de manejo de información geográfica que integra hardware, software, datos geográficos y espaciales, a un equipo científico y de trabajo que en un conjunto crean lo que puede definirse como un modelo de la parte de la realidad en lo que refiere a las coordenadas terrestres. Los SIG pueden relacionar por medio de capas o tipos de datos la información de una localización geográfica.

Sistemas de Información Geográfica, definidos como sistemas informáticos diseñados para el manejo, análisis y cartografía de información espacial referenciada internamente. (3)

Los SIG funcionan como una base de datos donde recogen la información de cada uno de los objetos gráficos que son representados en un mapa digital, donde es guardada la información espacial que junto con la cartografía conforman la base de datos geográfica. Esta base de datos junto con la descripción espacial de los objetos y las relaciones que se establecen entre ellos también llamada topología, es lo que marca la diferencia de otros sistemas de gestión de la información.

MDT: Estructura numérica de datos que representa la distribución espacial de una variable cuantitativa y continua. Por lo tanto, son modelos simbólicos, ya que las relaciones de correspondencia que se establecen con el objeto real tienen la forma de algoritmos o formalismos matemáticos. (26)

Un Modelo Digital del Terreno es la representación de una forma más clara de la topología del terreno, lo realiza a través de una estructura interna de datos numéricos, como son una distribución espacial de variables que podrían ser la temperatura, o la presión atmosférica. Las variables utilizadas deben ser cuantitativas y continuas, donde se mantienen estáticas los valores de estas.

MDT es una estructura numérica de datos que representa la distribución espacial de una variable cuantitativa y continúa. (1)

Curvas de nivel: Es la unión de todos los puntos que poseen un mismo valor z en un conjunto de coordenadas x, y, z ; es decir los puntos están a una misma altura en el plano x, y . Desarrollan una trayectoria horizontal y representa la intersección de la superficie de nivel sobre el plano. (4)

Cartografía: Representación convencional, generalmente plana y con posiciones relativas; gráfica de una porción de superficie terrestre, de fenómenos concretos o abstractos localizables en el espacio y que se elabora según a una determinada escala y proyección. (4)

Estructura de datos: es una forma de organizar un conjunto de datos elementales (mínima información que se tiene en el sistema) con el objetivo de facilitar la manipulación de estos datos, como un todo o individualmente. (2)

Interpolación: Acción y efecto de interpolar: determinar e insertar unos valores intermedios a partir de unos puntos fijos conocidos. (4)

Acimut o azimut: Ángulo comprendido en un plano horizontal y medido, generalmente, en el sentido de las agujas del reloj formado por la dirección a determinar y una dirección fija que se toma como referencia, habitualmente es el norte. Es la demora de un objeto en rotación de 360 grados. (4)

1.3 Proceso de representación de los Modelos Digitales del Terreno en 3D sobre la Web

Para la representación de los MDT es necesario tener en cuenta que todos ellos se relacionan entre sí, y es lo que da lugar a una representación de la parte del terreno que se quiere visualizar. Para crear un modelo de un lugar determinado, se requiere de datos como la iluminación, los principales puntos de altura, la representación del relieve, la vegetación e hidrología. La representación de la vegetación y la cartografía se realizan normalmente sobre papel, sin embargo, la digitalización de estos datos permite un mejor entendimiento de la información.

Para llevar a cabo la digitalización del terreno se requiere conocer la información referente a las coordenadas(x, y) en el espacio y tener los datos de la altura de cada una de ellas. A través de estos

datos se pueden construir las líneas o curvas de nivel, mediante estas se puede obtener información sobre el relieve y accidentes naturales (presas, ríos, valles, montañas).

La generación de un MDE⁹ por medio de la cartografía, requiere de la digitalización de las curvas de nivel, además de la validación y depuración del MDE mediante la generación del relieve sombreado, para tener un mejor detalle del relieve y en su caso eliminar inconsistencias de altura. También se realiza la unión de los modelos que conforman el proyecto y se verifica la continuidad y consistencia lógica de los datos mediante una visualización de los mismos.

Para el proceso de representación de MDT en 3D es necesario tener presente que el MDE es un factor fundamental, así como la captura de los datos, y saber corregir los errores de captura. Entre los aspectos importantes dentro del MDE están la identificación y cálculo de las líneas de flujo, cuencas visuales, definir los modelos hidrológicos, irradiancia y reflectancia; realizar análisis de la insolación y sombra, para realizar estos cálculos se utilizan métodos de interpolación. La representación de MDT sobre la Web se realiza utilizando imágenes de todo lo que se quiere visualizar y no solamente cargando aquellas imágenes que se tienen guardadas en ficheros. A través de las estructuras de datos se calculan los valores en el espacio de aquellas variables de importancia para la representación.

1.4 Situación problemática

La utilización de los SIG es muy importante en nuestros días, en la UCI se encuentra del departamento Geoinformática del centro GEySED, el cual se dedica a la producción de Software, enmarcados en la Geografía, donde se guarda y almacena información sobre los datos geográficos y la información espacial. En este Centro se desarrolla actualmente el proyecto GeneSIG el cual tiene como objetivo crear una plataforma soberana basada en tecnologías libres. El proyecto cuenta con varios módulos, que conforman la estructura de este. La plataforma se desarrolla sobre el SO¹⁰ GNU/Linux en su distribución Ubuntu v8.10, la documentación es llevada por un control de versiones y así los usuarios tienen un registro restringido sobre lo que se está desarrollando. Para la gestión de recursos se utilizó el *MS Project* y el *MS*

⁹ **MDE**: acrónimo de Modelo Digital de Elevaciones

¹⁰ **SO**: acrónimo de Sistema Operativo

Excel. Entre los lenguajes de programación que se utilizan en la plataforma está PHP 5.0.1, Gd que permite generar imágenes dinámicamente, el framework Symfony, XSL, HTML, entre otros.

La visualización en la plataforma actualmente se realiza en 2D, lo que no permite la representación del modelado completo de los mapas. Además, no se puede realizar el análisis completo en los tres planos del espacio, pues hay datos que no son visibles en 2D. En el Centro existe el desconocimiento por parte de los integrantes de cómo llevar a cabo representaciones geográficas a 3D y así crear una visualización más completa y definida. No cuentan con documentación, así como herramientas ni algoritmos que le permitan modelar los escenarios en 3D sobre la Web.

1.5 Modelos Digitales del Terreno

1.5.1 MDE

Un MDE forma parte del conjunto de MDT. Es una estructura de datos numérica que representa los datos de las elevaciones de un territorio dado, a través de una ecuación en función de x , y , z , donde le atribuye a z el valor de la altura unido a los valores de x , y en las coordenadas. Constituye una capa del Modelo Digital del Terreno que representan las características de la superficie terrestre.

Ecuación en función de la altura:

$z=f(x, y)$, donde x , y constituyen las coordenadas de un punto, mientras que z representa la altura.

El MDE está dividido en dos modelos de datos que posibilitan la representación: el modelo vectorial y el ráster. Los vectoriales están definidos por sus coordenadas que representan puntos o líneas y las ráster que representan los valores que ocupan las variables en la localización espacial del terreno.

Estructuras de datos:

- Estructura vectorial:
 - ◆ Contornos
 - ◆ TIN
- Estructura ráster
 - ◆ Matrices regulares

Contornos: Utiliza una polilínea que consta con un vector de varios pares de coordenadas(x,y) que conforman la trayectoria de las curvas de nivel, que son representadas mediante una serie de puntos que se sitúan sobre el modelo. Este Modelo Digital consta de varias curvas de nivel que pasan por encima de la zona a representar.

TIN o red de triángulos irregulares: La representación se realiza utilizando una serie de triángulos irregulares adosados¹¹ que son construidos usando un plano y seleccionando tres puntos subyacentes y que no sean colineales¹², donde cada triángulo puede representar una parte de la superficie y así caracterizar el relieve.

Matrices regulares: En esta estructura se superpone una retícula¹³ sobre el terreno, que toma la forma de una red de mallas cuadrada de donde se extrae una serie de datos para precisar la altura de cada celda. Donde el valor de cada dato está precisado por su localización espacial en la matriz con respecto a las filas y columnas.

Variables derivadas

Para realizar la representación del relieve se utilizan variables derivadas, como son: la pendiente, la orientación y la curvatura, a este proceso se le llama parametrización del relieve, que no es más que el conjunto de medidas que son definidas para la caracterización geométrica del terreno.

La pendiente de un punto del terreno está definida por el ángulo del vector normal de la superficie que forma dicho punto con la vertical. En el MDE se utiliza para el cálculo de la pendiente de un plano.

Un Modelo Digital de pendientes se realiza a través del plano y puede representar la información relativa de la distancia y vecindad de los valores de altitud.

La orientación en un punto puede definirse como el ángulo existente entre el vector que señala el Norte y la proyección sobre el plano horizontal del vector normal a la superficie en ese punto. Como en el caso de la pendiente, el valor de orientación se estima directamente a partir del MDE. (1)

¹¹ **adosados:** Triángulos adyacentes, que se encuentren uno al lado del otro

¹² **Colineales:** cuándo al pasar una recta todos los puntos están dentro de esta si hay alguno fuera de esta no son colineales

¹³ **Retícula:** red de puntos sobre el terreno

La curvatura en un punto puede definirse como la tasa de cambio en la pendiente, que depende de las derivadas de la altitud, de los cambios de pendiente en el entorno del punto. La curvatura tiene especial interés como variable influyente en fenómenos como la escorrentía superficial, canalización de aludes, erosión y flujos en general.

MDM¹⁴

Cuando un conjunto de capas constituye un modelo digital del terreno, como por ejemplo en la primera capa vendría el modelo de elevaciones y en la segunda capa la pendiente.

1.5.2 Cuencas visuales, visualización del relieve y modelos de reflectancia

Las cuencas visuales y los modelos de reflectancia se utilizan para la visualización del relieve fácilmente interpretado.

Cuencas visuales

Las cuencas visuales son un conjunto de información que permite la visibilidad de lugares, que utilizan una serie de puntos de donde se escogen los que permiten una mejor visualización del terreno en el área de estudio. Para el análisis de las cuencas visuales se requiere una serie de datos entre ellos, las curvas de nivel, densidad de la vegetación, y datos del relieve. Los puntos son creados sobre curvas de nivel de mediano y bajo tamaño, aquellas que son muy grandes dificultarían la visualización de áreas muy bajas en el terreno.

Una cuenca visual de un punto o foco determinado no es más que la conexión que existe entre este punto y el conjunto de puntos de un modelo que están conectados visualmente, para que esto suceda debe de existir la intervisibilidad¹⁵ entre el punto o foco y cada uno de los puntos que forman el modelo. Esto no es más que al trazar una recta sobre estos pares de puntos se cumpla que la recta posea una altura superior a la del terreno sobre la proyección del plano sin tener en cuenta el inicio y final de la recta. Luego de

¹⁴ **MDM:** acrónimo de Modelo Digital Multivariable

¹⁵ **Intervisibilidad:** es la accesibilidad visual de un punto desde el resto de puntos del modelo.

obtener el modelo de puntos y de haber realizado el análisis de intervisibilidad entre cada par de puntos se realiza la cartografía que no es más que la representación de estos lugares.

Las cuencas visuales tienen gran importancia para las aplicaciones prácticas de visibilidad de lugares como el análisis del impacto visual para representar los efectos negativos sobre el paisaje, y puede ser tomado como una base objetiva para tomar decisiones en cuanto a fiabilidad de la incidencia visual.

Visualización del relieve

La representación del relieve en un MDE, es muy importante por lo que permite una mejor explicación sobre los diferentes fenómenos sobre la superficie terrestre. La representación del relieve tiene que facilitar una mejor interpretación visual del relieve y está muy relacionado con los patrones de iluminación, donde utilizan valores que de acuerdo con los valores de altitud son representados en una escala de grises, donde el menor valor corresponde al negro y el mayor al blanco, esto permite una mejor explicación de la apariencia visual pero dificulta localización de los elementos del relieve.

Modelo de reflectancia

El modelo de reflectancia es considerado un caso particular de la visualización del relieve debido a que realiza una simulación de forma realista de las condiciones reales de la iluminación.

El proceso de iluminación es muy importante en este modelo y para esto posee dos propiedades muy importantes. (5)

- La vista cenital: el punto de vista está a una altitud infinita sobre el centro del modelo.
- El valor correspondiente a cada pixel o elemento del modelo se asigna mediante algoritmos que simulan la apariencia real de la superficie frente a condiciones específicas de iluminación (brillo aparente), de donde se deriva la denominación MDB que se adopta aquí para estos modelos.

La iluminación depende de algunos factores que inciden sobre él, entre ellos se encuentran que es definida una sola fuente de iluminación, esta tiene que ser directa con un ángulo de incidencia solar por encima del modelo, que en casos muy particulares se define otra fuente de luz hemisférica para simular la luz difusa que depende de la absorción atmosférica, la incidencia solar y la altitud.

1.5.3 Líneas de flujo, Cuencas hidrológicas y Modelo de caudales máximos

La línea de flujo es el trayecto que sigue un punto inicial, toda la escorrentía superficial sobre el terreno, que puede terminar en el desemboque del mar o al final de un modelo. Este se puede construir siguiendo el sentido de la máxima pendiente hasta llegar a un sumidero, a la costa o al borde del modelo.

A través de las líneas de flujo se puede determinar la red hidrológica, el área subsidiaria de una celda y las cuencas hidrológicas. El área subsidiaria de una celda no es más que el conjunto de celdas cuyas líneas de flujo convergen en ella, y una cuenca hidrológica está formada por el área subsidiaria de una celda singular, que actúa como sumidero. (5)

El agua que escurre en un río es captada en un área determinada, por lo general por la conformación del relieve. A esta área se le llama cuenca hidrológica.

El área subsidiaria puede definirse para cualquier punto del territorio que se está analizando, sin embargo, no todos los puntos pueden ser considerados sumideros de una cuenca hidrológica. Un punto del modelo es considerado un sumidero de la cuenca si es el de menor altura de una concavidad, si se encuentra al borde del modelo y los demás puntos drenen hacia él, o si está situado en la costa con drenaje al mar.

El trazado de las líneas de flujo puede realizarse de acuerdo con criterios muy simples: partiendo de un punto del modelo, se construye siguiendo el sentido de máxima pendiente hasta llegar a un sumidero, al borde del modelo o a la costa.

El proceso de construcción de una línea de flujo a partir de un punto inicial, es iterativo y consta de tres fases elementales. (5)

1. Se fija el punto inicial de la línea, $P(i)$.
2. Se calculan las pendientes hacia sus 8 vecinos más próximos.

Pueden darse 3 casos:

- 2.1. Todas las pendientes son negativas (se trata de una concavidad): fin de la línea.
- 2.2. Todas las pendientes son negativas y el punto está en el borde del MDE (la cuenca continúa probablemente fuera de los límites del MDE): fin de la línea.
- 2.3. Se localiza al menos un punto con pendiente positiva: se elige el punto con pendiente máxima.

3. El punto elegido (pendiente máxima) se incorpora a la línea flujo y se toma como base para volver al paso 2.

El sentido de flujo de un punto de un determinado modelo se mantiene de forma constante, por lo tanto, es posible construir una matriz que almacena un archivo de códigos o CDF¹⁶, que son los posibles sentidos de la escorrentía superficial, que se reduce a 9 valores, el 1 identifica los flujos en dirección norte; del 2 al 8, en sentido horario, del noreste al noroeste, y el 9 identifica las concavidades y zonas planas, en las que el flujo no es posible o queda indefinido.

Cuencas de drenaje

La identificación y delimitación de las cuencas puede considerarse una generalización de las líneas de flujo. Se crea una matriz del mismo tamaño que el MDE, donde a cada punto le corresponde un número indicando la cuenca a la que pertenece. El área de una cuenca se puede determinar por el número de puntos que la conforman por la distancia entre ellos.

Para determinar el límite de las cuencas de drenaje se sigue una serie de pasos donde inicialmente se le asigna como valor cuenca 0 a todos los puntos y se repite los pasos hasta que todos los puntos posean un número de cuenca. (6)

1. Se elige un punto P (i) del MDE y se comprueba si está asignado a alguna cuenca. En caso afirmativo se continúa con el siguiente punto del modelo; en caso negativo se calcula su línea de flujo, compuesta por n puntos.
2. Se comprueba el valor de cuenca del último punto de la línea. Pueden presentarse dos casos:
 - 2.1. El punto final de la línea no está asignado aún a ninguna cuenca (mantiene el valor 0): se asigna a todos los puntos de la línea un nuevo valor de cuenca y se inicia el proceso con el siguiente punto del MDE.
 - 2.2. El punto final de la línea es diferente de cero, lo que implica que la línea de flujo comunica con una cuenca preexistente: todos los puntos pertenecientes a la línea actual se asignan a esa cuenca y se inicia el proceso en 1 con un nuevo punto del modelo.

¹⁶ CDF acrónimo de códigos de flujo de un punto

MDF¹⁷

La magnitud del área subsidiaria de una celda del MDE está directamente relacionada con el CMP¹⁸. El caudal que puede circular en un momento dado en un punto del terreno depende, entre otros factores, de la magnitud del área subsidiaria, de las precipitaciones sobre ella y de la pendiente de la zona, que permite la circulación con menor o mayor rapidez.

Cuando se realiza el trazado de las líneas de flujo, se puede calcular la extensión del área subsidiaria de cada punto del modelo. Se realiza una línea de flujo para cada punto del MDE, donde cada elemento del modelo obtiene la cantidad de líneas de flujo que pasan por él, es decir, que al final cada valor se corresponde con su área subsidiaria, que serían los puntos que drenan hacia él, desde las zonas de mayor altura. La mayor parte de los puntos tendrán valores bajos, especialmente los que se encuentran situados en zonas de cumbres y crestas, y en menor medida los constituyentes de laderas. Los flujos convergerán en el fondo de los valles y los valores van a ir aumentando hasta que alcanza el máximo valor en el punto que ejerce como sumidero de la cuenca.

Este modelo representa los valores de caudal máximo de cada punto, donde el área subsidiaria es proporcional a este, para alcanzar un equilibrio en el proceso. La situación de equilibrio llega cuando el número de ciclos elementales (paso de un punto a otro del modelo), ha sido suficiente para completar la línea de flujo más larga, y también cuando después de un tiempo de lluvia suficientemente largo, la escorrentía superficial, originada en el punto más alejado del sumidero de la cuenca, ha tenido tiempo de pasar por el mismo.

1.5.4 Sombras, Insolación y Modelo de irradiancia

El relieve también se puede representar cartográficamente mediante un mapa de sombra, donde la variable utilizada es una simulación del nivel de luz o sombra, reflejado por el relieve al ser iluminado por el sol, situado en una situación geográfica. La cantidad de luz reflejada depende de la posición del sol y de la pendiente del relieve. Las zonas con más pendientes, que estén expuestas directamente al sol, reflejan

¹⁷ **MDF** acrónimo de Modelo de caudales máximos

¹⁸ **CMP** acrónimo de caudal máximo potencial

mucha luz y son más visibles que aquellas que se encuentran en laderas que no están iluminadas directamente por el sol, por lo tanto, no reflejan luz y están oscuras. (10)

Trayectoria solar

La radiación solar es una variable de gran interés, constituye la única entrada de energía en los sistemas físicos y biológicos, y condiciona una parte de los factores climáticos como la temperatura. La localización aparente del sol depende de una serie de parámetros, donde los más importantes son: latitud del punto problema (positiva en el hemisferio Norte y negativa en el Sur), declinación solar (variable según la época del año) y ángulo horario (dependiente de la hora del día).

En el ángulo horario suele tomarse en 0 cuando es mediodía, cuando el azimut solar es de 180° , el sol está situado al Sur. Los ángulos son negativos hacia el Este y positivos hacia el Oeste, con intervalos de 15° por hora: por ejemplo, cuando faltan dos horas para el mediodía, el ángulo horario es -30° . (5)

Ocultamiento topográfico

La existencia de zonas de sombras es una variable de interés en las zonas montañosas, donde el relieve puede ser el factor determinante más importante del clima local. Un punto se encuentra en sombra debido a dos alternativas:

- ✓ Autoocultamiento, se produce cuando el vector normal a la superficie forma un ángulo superior a los 90° con el vector solar; como sería el caso, por ejemplo, de una ladera orientada al Norte con pendiente de 45° cuando el sol ilumina desde el Sur, elevado solamente 30° sobre el horizonte.
- ✓ Ocultamiento por relieve circundante; las fases del análisis son: (5)
 - Se elige el punto problema P (i), definido por su fila y columna.
 - Se construye el perfil topográfico entre P (i) y un punto P (j), inmediatamente exterior al MDE situado sobre la línea del azimut solar.
 - A P(j) se le asigna la altitud correspondiente a la línea de trayectoria solar en ese punto (función de la distancia a P(i) y de la altura del sol sobre el horizonte).
 - Realiza un análisis de intervisibilidad entre P (i) y P (j). Pueden darse dos casos.
 - Existe intervisibilidad entre los dos puntos: P (i) no está en sombra.

- La línea visual entre P (i) y P (j) está interrumpida: existe ocultamiento topográfico de P (i) por algún punto de su entorno.

Para el levantamiento del perfil: (5)

- ✓ Se definen los puntos inicial y final del segmento rectilíneo sobre el que se levantará el perfil topográfico.
- ✓ Se fija el número de puntos del perfil (su resolución en el plano xy).
- ✓ Se calculan las coordenadas (x, y) de cada punto del perfil.
- ✓ Se estima por interpolación la altitud z de cada punto (x, y).

La insolación potencial de un punto no es más que el tiempo al que está sometido ese punto a la radiación solar directa en ausencia de nubosidad. La diferencia de insolación entre dos puntos en un mismo día, está determinado por el relieve y el ocultamiento topográfico. Suponiendo que se desee conocer la insolación en un punto del MDE a lo largo de un día concreto del año, el método de análisis se plantea como sigue:

1. Se especifican los parámetros básicos (latitud del lugar y declinación solar) del punto problema, cuyo valor de insolación es inicialmente cero.
2. Se generan las posiciones del sol a lo largo del día (en función de los parámetros anteriores) a intervalos predefinidos de tiempo. El resultado es un conjunto de n posiciones solares definidas por pares de valores de azimut y elevación: S (i). Comenzando por i=1 (primera posición solar):
3. Se analiza la posición solar comprobando los siguientes aspectos:
 - 3.1 Si el valor de altura sobre el horizonte es positivo, la posición corresponde al periodo diurno: se continúa en el paso 4.
 - 3.2 Si es de noche e $i < n$, se toma la siguiente posición solar, S (i+1) y se repite el paso 3.
 - 3.3 Si es de noche e $i = n$ (última posición), el análisis ha finalizado y el tiempo de insolación está almacenado en la variable correspondiente.
4. Se realiza el análisis de ocultamiento para la posición solar actual. Pueden ocurrir dos casos:
 - 4.1 Existe ocultamiento topográfico (el lugar está en sombra y no recibe luz solar directa): se toma la siguiente posición solar y se continúa en el paso 3.
 - 4.2 No existe ocultamiento (la luz solar incide directamente sobre el punto analizado): se añade una unidad al tiempo de insolación, se toma la siguiente posición solar y se continúa en el paso 3.

Modelo de Irradiancia

Es posible calcular la radiación solar que una superficie puede obtener potencialmente en ausencia de nubosidad. Sobre una superficie perpendicular a la radiación solar, la energía incidente o irradiancia I_n , está dada:

$$I_n = I_0 * p^m$$

Donde I_0 representa una constante solar (1367 Wm^{-2} ¹⁹), p la transmisividad²⁰ media cenital, y m la masa óptica del aire.

Los índices de radiación determinados para cada punto del MDE están definidos como el cociente entre la radiación solar incidente sobre la superficie del problema y una superficie horizontal, usada como referencia. Los índices pueden elaborarse, considerando que no existe ocultamiento topográfico, analizando la localización específica de cada punto del modelo. Los modelos de irradiancia son construidos a través de los índices de radiación.

1.6 Otras soluciones existentes

En el mundo se han creado programas y herramientas, que posibilitan la representación de MDT. En el siguiente punto se abordarán algunas de estas soluciones existentes en algunos países y se hará un análisis de que se podría tomar como referencia para la investigación que se lleva a cabo.

3Dem

3Dem es un software de visualización de MDE gratuito, que presenta grandes funcionalidades para generar vistas y animaciones en 3 dimensiones. Es útil para crear mapas temáticos o visualizar imágenes satelitales en 3D y vuelos animados, a distintas altitudes y por rutas predefinidas por el usuario.

¹⁹ Wm^{-2} : densidad de flujo de radiación solar, recibida, transmitida reflejada y absorbida por el planeta.

²⁰ **Transmisividad:** no es más que la facilidad del agua de circular horizontalmente por una formación geológica. Cantidad de agua por unidad de ancho.

El programa 3DEM es un software libre creado para Windows. Los formatos soportados son: USGS DEM, SRTM, GLOBE, Mars Viking Orbiter, GEOTIFF, Terrain Matrix, GTOPO30 y Mars MOLA. Además de visualizar MDE, permite también exportar estos modelos a otros formatos (USGS ASCII DEM, Geotiff y Terrain Matrix), capturar imágenes de pantalla, generar vistas en 3D, entre otros.

DbMAP Web 3D

Es un software que puede ser utilizado para la representación de un escenario virtual en 3D; que permite utilizar información cartográfica, proveniente de los IDE (infraestructura de datos espaciales). Permite el uso de datos espaciales de los IDE y posee la ventaja de renovar la información automáticamente en el modelo virtual, siempre y cuando se hagan actualizaciones en el IDE. Los usuarios no necesitan instalar software cliente para poder utilizar la información en 3D. El software permite visualizar en la Web y en 3D información cartográfica recogida.

GRASS GIS

Es un software de SIG totalmente gratuito, está basado en licencia libre, que soporta las estructuras de datos ráster y vectorial. Además, contiene herramientas que permiten procesar imágenes digitales.

Posee una base de datos que almacena toda la información de GRASS. En esta BD los proyectos son organizados por áreas de subdirectorios que son llamadas *location*. Un *location* es definido por un sistema coordenado, una proyección cartográfica y unas fronteras geográficas. Los subdirectorios y archivos que definen un *location* son creados automáticamente cuando GRASS es iniciado la primera vez con un nuevo *location*. Permiten almacenar mapas relacionados con los proyectos que son guardados en la BD. Este software puede importar capas, ya sea ráster o vectorial, o cargar mapas.

CEM

El CEM²¹ es un producto digital derivado en el Instituto Nacional de Estadística, Geografía e Informática INEGI que se ofrece de manera pública como un servicio más por Internet. A comparación de otros Modelos Digitales de Elevación del terreno, posee ventajas de homogeneidad en sus datos con cobertura nacional y sin huecos de información. El CEM es el modelo digital de elevaciones del terreno con

²¹CEM acrónimo de Continuo de Elevaciones Mexicano

cubrimiento en 100% del territorio nacional, con fuente de información Topográfica a escala 1:50,000. En una sola red sin faltantes de información éste modelo abre un nuevo servicio con la habilidad de dar a quien lo solicite, datos de cualquier zona del país con cobertura de libre elección, sin necesidad de ajustarse al mosaico original.

El MDE del Ecosistema Sabana-Camagüey (Cuba) forma parte de la base de dato del SIESC²², como soporte de mapas temáticos. Este por su esencia digital, admite la realización de simulaciones de procesos dinámicos, lo cual tiene gran importancia para el estudio de los procesos naturales que resultarán útiles en la toma de decisiones y otros tipos de aplicaciones, debido a que permite la visualización de este modelo utilizando algunas herramientas.

El ecosistema Sabana-Camagüey se extiende en una franja de 465 km a lo largo de la zona norte y central de Cuba, entre la Punta Hicacos y la bahía de Nuevitas, de oeste a este. El área de estudio abarca 75 000 km² y comprende la vertiente o cuenca hidrográfica norte de las provincias de Matanzas, Villa Clara, Sancti Spíritus, Ciego de Ávila y Camagüey, incluyendo todo el conjunto de cayos desde el sur al norte. Donde se estudia:

- La cuenca hidrográfica (19 630 km²) y sus respectivas subcuencas, incluye toda el área terrestre de la isla principal. (14)
- La plataforma marina (8 311 km²) incluye todos los hábitats de aguas marinas poco profundas, hasta 50 metros en el borde exterior de la plataforma. Sus 2 517 cayos representan el 60% de todos los cayos de Cuba y tienen 3 400 km².
- La zona económica exclusiva (zona oceánica), que abarca hasta 200 Millas náuticas, más allá del borde de la plataforma marina, con aproximadamente 43 800 km².

Algunos de los proyectos mencionados anteriormente constituyen herramientas que se han creado en diferentes lugares o países para la representación de los MDT. Estas soluciones sirven de guía para tener una idea sobre cómo se está desarrollando en el mundo la representación del terreno, sin embargo, es necesario continuar con la investigación debido a que se necesita un método, algoritmo o herramienta que

²² **SIESC**: acrónimo de Sistema de Información del Ecosistema Sabana-Camagüey

permita llevar a cabo el trabajo realizado por el proyecto GeneSIG, actualmente en 2D a 3D y así mejorar la visibilidad de lo que se quiere representar.

1.7 Conclusiones

En este capítulo se realizó un pequeño análisis de los MDT, se dio una fundamentación teórica, así como la explicación del objeto de estudio, situación problemática. Luego de realizado el primer capítulo del trabajo se tiene una noción de que es MDT, así como cuales son los diferentes modelos dentro de este y una breve explicación de algunos programas que han sido realizados hasta el momento.

CAPÍTULO 2: Tendencias y Tecnologías para la representación de MDT

2.1 Introducción

En este capítulo se abordará sobre los métodos matemáticos que permiten una representación de los MDT, para ello es necesario la utilización de polinomios de interpolación. Son muchos los tipos de interpolación que existen, pero los utilizados para la representación del terreno son los Spline, unido a los polinomios de Lagrange de dos variables, debido a que si se utiliza solamente el polinomio de Lagrange al interpolar los datos, los puntos pueden estar lejos de la realidad y ofrecer una representación de la curva poco realista; utiliza una Base de Datos donde se encuentran almacenados todos los puntos, que son obtenidos a través de un mapa topográfico. Ya que los Spline en vez de utilizar un polinomio utiliza trozos de polinomios por tramos para interpolar los datos.

Polinomio de Lagrange

$$P_n(x) = y_0l_0(x) + y_1l_1(x) + \dots + y_jl_j(x) + \dots + y_nl_n(x) = \sum_{j=0}^n y_jl_j(x)$$

Para esto se deben de cumplir una serie de restricciones entre las que están:

Se tiene que cumplir que $P(x_0)=y_0$ si y solo si $l_0(x_0)=1$ y $l_i(x_0)=0 \quad i \neq 0$

Se tiene que cumplir que $P(x_1)=y_1$ si y solo si $l_1(x_1)=1$ y $l_i(x_1)=0 \quad i \neq 0$

Estas restricciones son para cada uno de los valores de n. (17)

La ecuación final para $L_j(x)$ es:

$$L_j(x) = \frac{(x-x_0)(x-x_1)\dots(x-x_{j-1})(x-x_{j+1})\dots(x-x_n)}{(x_j-x_0)(x_j-x_1)\dots(x_j-x_{j-1})(x_j-x_{j+1})\dots(x_j-x_n)} = \prod_{\substack{k=0 \\ k \neq j}}^n \frac{(x-x_k)}{(x_j-x_k)}, \quad j = 0,1,\dots,n$$

Ejemplo para $L_0(x)$
$$L_0(x) = \frac{(x-x_1)(x-x_2)\dots(x-x_n)}{(x_0-x_1)(x_0-x_2)\dots(x_0-x_n)}$$

2.2 Métodos Matemáticos para representar MDT

2.2.1 Modelo de Spline Bilineal

Spline: No es más que la representación de las curvas a través de polinomios de una determinada forma. Curva de Spline se refiere a cualquier curva compuesta que se forma con secciones polinómicas que satisfacen condiciones específicas de continuidad. Una curva de Spline se especifica a partir de un conjunto de posiciones de coordenadas, que se conocen como puntos de control, los cuales indican la forma general de la curva. Dado un conjunto de puntos de control, los métodos de interpolación generan una curva que pasa por todos los puntos de control. En cambio, los métodos de aproximación generan una curva que normalmente no pasa por todos los puntos de control, excepto, tal vez, por los puntos extremos.

Para llevar a cabo la interpolación se tienen los valores de una función $f(x)$ evaluada en x_0, \dots, x_n , la interpolación consiste en hallar otra función $g(x)$ que cumpla que $g(x_0)=f(x_0)$

x_0, \dots, x_n : nodos de interpolación

$g(x)$: función interpoladora

Existen $n+1$ números reales diferentes y se cumple que $g(x)$ tiene grado menor o igual que n si $y_0=f(x_0)$ se cumple que $g(x_0)=y_0$ de donde se obtiene la ecuación:

$$g(x) = a_0 + a_1x + \dots + a_nx^n$$

Para

$$g(x_0) = a_0 + a_1x_0 + \dots + a_nx_0^n = y_0$$

$$g(x_1) = a_0 + a_1x_1 + \dots + a_nx_1^n = y_1$$

⋮

$$g(x_n) = a_0 + a_1x_n + \dots + a_nx_n^n = y_n$$

Se forma una matriz con los valores de

$$\begin{bmatrix} 1 & x_0 & \dots & x_0^n \\ 1 & x_1 & \dots & x_1^n \\ \vdots & \vdots & & \vdots \\ 1 & x_n & \dots & x_n^n \end{bmatrix} \neq 0 \mapsto \text{si } x_0, x_1, \dots, x_n \text{ Son diferentes}$$

Es necesario interpolar porque no se conocen con exactitud el valor de $f(x)$ sino algunos valores aproximados.

La fragmentación más simple es la interpolación lineal fragmentaria o Spline lineal, que consiste en unir los puntos mediante segmentos de rectas, por lo que sería continuo pero no derivable lo cual es una condición necesaria para la suavidad del polinomio.

Como es un Spline bilineal la función polinómica tiene grado uno, ejemplo:

$$FP(x) = ax + b$$

Para llevar a cabo las funciones de interpolación se requiere conocer las coordenadas espaciales de los nodos, estas son tomadas de un mapa topográfico.

En el caso del modelo bilineal los subconjuntos tomados para la interpolación segmentaria son cuadrículas de cuatro nodos, por lo que al interpolar con el polinomio de Lagrange las ecuaciones obtenidas sobre cada subconjunto son funciones bilineales, en estas funciones se realiza la interpolación de los valores de Z de aquellos puntos que se encuentran en la región interior de la poligonal determinada por los datos originales:(18)

$$Z = a_0 + a_1x + a_2y + a_3x$$

Ecuación para el Spline de grado 1 por cada par de intervalos:

$$s_i(x) = y_i + \frac{y_{i+1} - y_i}{x_{i+1} - x_i}(x - x_i), x \in [x_i, x_{i+1})$$

2.2.2 Modelo de Spline bicuadráticas

La interpolación cuadrática asegura que la función es continua; para llevarlo a cabo tiene que cumplir algunas condiciones:

- Que las partes de la función a trozos $P(x)$ pasen por ese punto. Es decir, que las dos $P_n(x)$ que rodean al $f(x)$ que se quieren aproximar, sean igual a $f(x)$ en cada uno de estos puntos.

- Que la derivada en un punto siempre coincida para ambos lados de la función definida a trozos que pasa por tal punto común.

En el caso del Spline bicuadráticas la función polinómica sería de grado 2:

$$FP(x) = ax^2 + bx + c$$

Cada uno de los n polinomios poseen 3 coeficientes por lo tanto existen 3*n coeficientes.

Para este modelo los subconjuntos tomados para la interpolación segmentaria son cuadrículas de nueve nodos, por lo que al interpolar con el polinomio de Lagrange las ecuaciones obtenidas sobre cada subconjunto son funciones bicuadráticas de la forma:

$$Z(x, y) = a_0 + a_1 x + a_2 y + a_3 xy + a_4 x^2 + a_5 y^2 + a_6 yx^2 + a_7 xy^2 + a_8 x^2 y^2$$

Cuando esta superficie se intercepta con los planos paralelos a los planos coordenados se producen curvas parabólicas.

2.2.3 Modelo de Spline bicúbico

Función polinómica:

$$FP(x) = ax^3 + bx^2 + cx + d$$

Cada uno de los n polinomios contiene 4 coeficientes por lo tanto el Spline posee 4n coeficientes.

S(x): polinomio

$$s(x) = \begin{cases} a_1 x^3 + b_1 x^2 + c_1 x + d_1 \rightarrow x_0 \leq x < x_1 \\ a_2 x^3 + b_2 x^2 + c_2 x + d_2 \rightarrow x_1 \leq x < x_2 \\ \vdots \\ a_n x^3 + b_n x^2 + c_n x + d_n \rightarrow x_{n-1} \leq x < x_n \end{cases}$$

Se deben de cumplir las siguientes condiciones:

Interpolación: $S(x_i) = y_i$ $i=0, 1, 2, \dots, n$

Continuidad: S(x) es continua en x_i $i=0, 1, 2, \dots, n-1$

Suavidad: S'(x) es continua en x_i $i=0, 1, 2, \dots, n-1$

S''(x) es continua en x_i $i=0, 1, 2, \dots, n-1$

Se escoge una notación:

$$M_i = S''(x) \quad i=0,1,2,\dots,n$$

$h_i = x_{i+1} - x_i \quad i=0,1,2,\dots,n-1$ h_i representa la longitud de tramo de los polinomios del Spline.

$$\begin{aligned} \frac{h_0 + h_1}{3} M_1 + \frac{h_1}{6} M_2 &= \frac{y_2 - y_1}{h_1} - \frac{y_1 - y_0}{h_0} \\ \frac{h_1}{6} M_1 + \frac{h_1 + h_2}{3} M_2 + \frac{h_2}{6} M_3 &= \frac{y_3 - y_2}{h_2} - \frac{y_2 - y_1}{h_1} \\ \frac{h_2}{6} M_2 + \frac{h_2 + h_3}{3} M_3 + \frac{h_3}{6} M_4 &= \frac{y_4 - y_3}{h_3} - \frac{y_3 - y_2}{h_2} \\ &\vdots \\ \frac{h_{n-2}}{6} M_{n-2} + \frac{h_{n-2} + h_{n-1}}{3} M_{n-1} &= \frac{y_n - y_{n-1}}{h_{n-1}} - \frac{y_{n-1} - y_{n-2}}{h_{n-2}} \end{aligned}$$

La ecuación del Spline para cada valor de i es:

$$s(x) = \frac{(x_{i+1} - x)^3 M_i + (x - x_i)^3 M_{i+1}}{6h_i} + \frac{(x_{i+1} - x)y_i + (x - x_i)y_{i+1}}{h_i} - \frac{h_i [(x_{i+1} - x)M_i + (x - x_i)M_{i+1}]}{6}$$

Luego llevándolo a la forma matricial

$$HM = Y$$

Donde

$$H = \begin{bmatrix} \frac{h_0 + h_1}{3} & \frac{h_1}{6} & & & & \\ \frac{h_1}{6} & \frac{h_0 + h_1}{3} & \frac{h_2}{6} & & & \\ & \frac{h_2}{6} & \frac{h_2 + h_3}{3} & \ddots & & \\ & & \ddots & \ddots & \frac{h_{n-2}}{6} & \\ & & & & \frac{h_{n-2}}{6} & \frac{h_{n-2} + h_{n-1}}{3} \end{bmatrix} \quad M = \begin{bmatrix} M_1 \\ M_2 \\ M_3 \\ \vdots \\ M_{n-1} \end{bmatrix} \quad Y = \begin{bmatrix} \frac{y_2 - y_1}{h_1} - \frac{y_1 - y_0}{h_0} \\ \frac{y_3 - y_2}{h_2} - \frac{y_2 - y_1}{h_1} \\ \frac{y_4 - y_3}{h_3} - \frac{y_3 - y_2}{h_2} \\ \vdots \\ \frac{y_n - y_{n-1}}{h_{n-1}} - \frac{y_{n-1} - y_{n-2}}{h_{n-2}} \end{bmatrix}$$

Métodos de ajuste

Cuando se lleva a cabo la interpolación de curvas es necesario llevar a cabo métodos de ajustes de curvas que permite que la función pase por todos los puntos.

Para que la curva sea lo más cercano a la realidad es necesario que se escojan varios grupos de puntos para crear una curva suave, de donde se escogen varios pares de puntos para realizar la interpolación lineal.

Algunos métodos para el ajuste son:

- Ajustes por mínimos cuadrados

Es la técnica de optimización matemática más utilizada mundialmente para ajustar superficies, pues brinda soluciones eficientes y elegantes.

Se realiza utilizando distancias verticales u horizontales, es decir, distancias en x o y.

Polinomio:

$$g(x) = c_1 x^n + c_2 x^{n-1} + \dots + c_{n+1}$$

La desviación de la curva es:

$$r_i = y_i - g(x_i) \quad i = 1, 2, \dots, L \quad R = \sum_{i=1}^L r_i^2$$

L: número de puntos de datos.

y,x: coordenadas de los puntos.

n: cantidad de puntos.

R: derivaciones.

c_j: coeficientes del polinomio de ajuste.

Para minimizar R

$$\frac{\partial R}{\partial c_j} = 0 \rightarrow j = 1, 2, \dots, n+1 \quad \text{O lo que es lo mismo} \quad \sum_{j=1}^{n+1} \left(\sum_{i=1}^L x_i^{2n+2-j-k} \right) c_j = \sum_{i=1}^L x_i^{n+1-k} y_i \rightarrow k = 1, 2, \dots, n+1$$

Llevándolo a la forma matricial

$$\begin{bmatrix} x_1^n & x_1^{n-1} & \dots & 1 \\ x_2^n & \dots & \dots & 1 \\ \dots & \dots & \dots & \dots \\ x_L^n & \dots & \dots & 1 \end{bmatrix}, c = \begin{bmatrix} C_1 \\ C_2 \\ \dots \\ C_{n+1} \end{bmatrix}, y = \begin{bmatrix} y_1 \\ y_1 \\ \dots \\ y_L \end{bmatrix}$$

Cuando $L > n + 1$, la ecuación esta sobredeterminada porque el número de ecuaciones es mayor que el número de coeficientes no determinados.

➤ Ajustes de puntos infinitos

Cuando se realiza por un polinomio de grado 2 en el espacio el cual representa dos paraboloides:

$$Z = A + Bx + Cy + Dxy + Ex^2 + Fy^2$$

Se tienen los pares de puntos (x, y) que corresponden a una Z que tiene un error E , es decir, cada $Z_i = Z_v + E$.

Se definen como sigue a continuación los coeficientes:

$$R_0 = 1, R_1 = x, R_2 = y, R_3 = xy, R_4 = x^2, R_5 = y^2$$

$$(R_0 \cdot R_1)A + (R_0 \cdot R_2)B + (R_0 \cdot R_3)C + (R_0 \cdot R_4)D + (R_0 \cdot R_5)E + (R_0 \cdot R_0)F = R_0 Z$$

$$(R_1 \cdot R_1)A + (R_1 \cdot R_2)B + (R_1 \cdot R_3)C + (R_1 \cdot R_4)D + (R_1 \cdot R_5)E + (R_1 \cdot R_0)F = R_1 Z$$

$$(R_2 \cdot R_1)A + (R_2 \cdot R_2)B + (R_2 \cdot R_3)C + (R_2 \cdot R_4)D + (R_2 \cdot R_5)E + (R_2 \cdot R_0)F = R_2 Z$$

$$(R_3 \cdot R_1)A + (R_3 \cdot R_2)B + (R_3 \cdot R_3)C + (R_3 \cdot R_4)D + (R_3 \cdot R_5)E + (R_3 \cdot R_0)F = R_3 Z$$

$$(R_4 \cdot R_1)A + (R_4 \cdot R_2)B + (R_4 \cdot R_3)C + (R_4 \cdot R_4)D + (R_4 \cdot R_5)E + (R_4 \cdot R_0)F = R_4 Z$$

$$(R_5 \cdot R_1)A + (R_5 \cdot R_2)B + (R_5 \cdot R_3)C + (R_5 \cdot R_4)D + (R_5 \cdot R_5)E + (R_5 \cdot R_0)F = R_5 Z$$

Y haciendo uso del método de mínimos cuadrados quedaría como:

$$A \sum x_i + B \sum y_i + C \sum x_i y_i + D \sum x_i^2 + E \sum y_i^2 + F \sum 1 = \sum z_i$$

$$A \sum x_i^2 + B \sum x_i y_i + C \sum x_i^2 y_i + D \sum x_i^3 + E \sum y_i^2 x_i + F \sum x_i = \sum z_i x_i$$

$$A \sum x_i y_i + B \sum y_i^2 + C \sum x_i y_i^2 + D \sum x_i^2 y_i + E \sum y_i^3 + F \sum y_i = \sum z_i y_i$$

$$A \sum x_i^2 y_i + B \sum x_i y_i^2 + C \sum x_i^2 y_i^2 + D \sum x_i^3 y_i + E \sum y_i^3 x_i + F \sum x_i y_i = \sum z_i y_i x_i$$

$$A \sum x_i^3 + B \sum x_i^2 y_i + C \sum x_i^3 y_i + D \sum x_i^4 + E \sum y_i^2 x_i^2 + F \sum x_i^2 = \sum z_i x_i^2$$

$$A \sum x_i y_i^2 + B \sum y_i^3 + C \sum x_i y_i^3 + D \sum x_i^2 y_i^2 + E \sum y_i^4 + F \sum y_i^2 = \sum z_i y_i^2$$

De este sistema de ecuaciones se hallan los valores de las variables A, B, C, D, E, F .

2.3 Algoritmos para la representación y visualización de los MDT

Para la generación de los MDT es necesario tener una serie de datos entre ellos (Topografía, Fotogrametría, o Cartografía existente), que será una nube de puntos con coordenadas tridimensionales (x, y, z) , que permiten representar la superficie topográfica. Esta nube de puntos, posee una distribución irregular, que serán los datos de partida, que mediante algoritmos de cálculo, se utilicen para la formación de los MDT. Los algoritmos que se utilizan para la formación de la malla de triángulos irregular, se basan fundamentalmente en la triangulación de Delaunay, que se trata de estructuras computacionales, que permiten la construcción de una triangulación óptima para la representación del terreno. Con estos algoritmos se cumple que los triángulos formados son lo más regulares posibles, la longitud de los lados de los triángulos es mínima, y la triangulación formada es única, dando lugar a la red irregular de triángulo.

➤ Triangulación de Delaunay

Una triangulación es la subdivisión de un área en triángulos. Una triangulación de una nube de puntos del plano es el conjunto de triángulos de interiores disjuntos cuyos vértices son puntos de la nube y en cuyo interior no hay ningún punto de la nube. Se puede triangular mientras se pueda realizar la unión de puntos de la nube con segmentos rectilíneos que no atraviesen los segmentos anteriores.

De una nube de puntos se puede obtener más de una triangulación siempre y cuando la unión de los puntos no sean los mismos. Existen muchas formas de triangulación pero la más eficiente para el terreno es la que forma triángulos más regulares, para obtener una mejor imagen del terreno real. Dada una nube de puntos del plano, hallar una triangulación en la que puntos próximos estén conectados entre sí por una arista, o dicho de otro modo, en la que los triángulos sean lo más regulares posible. (7)

Se trata de obtener una triangulación óptima, que permita una interpolación coherente entre los valores asociados (cotas o altitudes) a cada uno de los vértices; y así construir una red de triángulos irregulares (TIN), para la generación de los MDE. Una triangulación T_1 , es mejor que otra T_2 , cuando el menor ángulo de los triángulos de T_1 es mayor que el menor ángulo de los triángulos de T_2 . Es decir, la triangulación óptima, es la que maximiza el ángulo mínimo de los triángulos. (7)

Características de la triangulación de Delaunay

Se tiene un área P que contiene varios puntos (P_1, \dots, P_n) , P tiene que cumplir las siguientes propiedades:

1. Si se tienen tres puntos P_i , P_j , P_k que pertenecen a P y forman un triángulo, son vértices de la misma cara de la triangulación de Delaunay si el círculo que pasa por los tres puntos no contiene puntos en su interior (Figura 2).

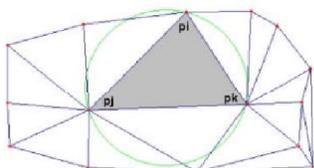


Figura 1 Propiedad 1 (Triangulación de Delaunay)

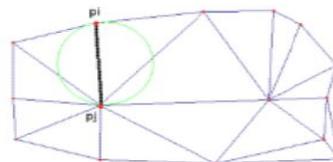


Figura 2 Propiedad 2 (Triangulación de Delaunay)

2. Dos puntos P_i , P_j que pertenecen a P , forman un lado de la triangulación de Delaunay de P , si existe un círculo que contiene a los puntos P_i , P_j y no contiene puntos en su interior (Figura 3).

Según estas dos propiedades mencionadas anteriormente se puede decir que la característica de la Triangulación de Delaunay es: sea P un conjunto de puntos en el plano y T una triangulación de P , T es una triangulación de Delaunay de P , si y solamente si, la circunferencia circunscrita de cualquier triángulo de T no contiene puntos de P (7). Este algoritmo es óptimo y de mínima complejidad, además de ser muy rápido y está implementado en numerosos programas para la generación de MDT.

Existen otros tipos de algoritmos para representar un terreno, algunos de ellos utilizan como representación gráfica un archivo de imagen en escala de grises en el que, por ejemplo, los colores claros representan valores de mayor altura, donde a cada píxel de la imagen le hacen corresponder un valor de altura. El problema en estas técnicas está que cuando es necesario realizar mapas muy grandes, se requiere dibujar un gran número de triángulos. Debido a esto se han creado los algoritmos CLOD²³ que tiene como objetivo reducir el número de polígonos enviados a la tarjeta a costa de trabajo extra para la CPU.

Existen varios algoritmos CLOD entre ellos: (8)

- ✓ Algoritmo Quadtree por Stefan Roettger.
- ✓ Algoritmo ROAM²⁴ por Mark Duchaineau.

²³ CLOD acrónimo de *Continuous Level Of Detail*

²⁴ ROAM acrónimo de *Real-time Optimally Adapting Meshes* o Mallas Adaptables Óptimas en Tiempo Real

- ✓ Algoritmo Geomipmapping por Willem H. Boer.

CLOD

Esta técnica usa el enfoque del LOD, que se basa en usar diferentes niveles de detalle dependiendo de la distancia y otros factores. La diferencia respecto al LOD estriba en que en este tipo de algoritmos es la misma aplicación la que va generando las mallas que se verán por pantalla. Esto ofrece varias ventajas, como por ejemplo el ahorro de espacio, al no tener que almacenar varias copias del mismo modelo en la memoria. Como siempre, se encuentran varios acercamientos; uno de ellos consiste en usar triángulos aproximadamente equiláteros e irlos dividiendo (cualquier triángulo se puede dividir en otros 2 ó 4), de manera que se obtienen más triángulos y por lo tanto, más detalles.

2.3.1 Quadtree

Quadtree se utiliza para describir una clase de estructuras de datos jerárquicas que tienen la propiedad de descomponer recursivamente el modelo. Pueden representar diferentes tipos de datos entre los que están puntos, regiones, curvas, entre otros. La descomposición está orientada en niveles hasta obtener la partición más pequeña donde se encuentran los datos que se quieran visualizar. Para el proceso de construcción del Quadtree se utilizan una serie de puntos de un plano, que conforman una malla rectangular regular que contiene las medidas de la altura mediante un arreglo bidimensional.

Quadtree es una estructura que divide recursivamente una región a modelar hasta que quedan celdas muy pequeñas. Permiten organizar el conjunto de texturas que se utiliza para la representación del terreno, y obtener que parte de la textura está siendo representada mediante la visibilidad de los nodos que conforman el árbol. Los métodos de triangulación del terreno mediante Quadtree son uno de los más eficientes. Puede representar diferentes tipos de datos como son puntos, regiones, curvas, volúmenes, entre otras. Cada una de las regiones que son particionadas conforman un nodo del árbol, cada nodo padre tiene cuatro nodos hijos. Un Quadtree deja de dividirse cuando todos los elementos en su interior tienen un mismo valor o cuando se halla alcanzado el máximo número de divisiones prefijado.

- **Quadtree restringido**
-

En esta estructura el proceso de triangulación mediante el proceso de división de los cuadrantes debe tener en cuenta algunos aspectos como por ejemplo que un cuadrante que se relaciona con otro cuadrante de otro nivel, en el proceso de división la arista que comparte con la celda de otro nivel no puede ser mayor que la unidad del cuadrante de mayor nivel. (Figura 4)(9)

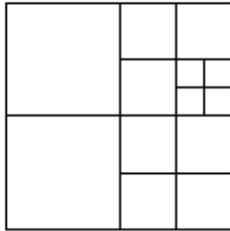


Figura 3 Triangulación mediante proceso de división

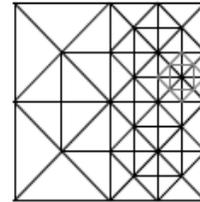


Figura 4 División de los cuadrantes

Después que se termina el proceso de división se comienza a realizar la triangulación, donde cada cuadrante se divide en ocho triángulos, a cada arista le corresponden dos triángulos, cuando una arista de un cuadrante coincide con un cuadrante de un nivel inferior, en este caso se forma un solo triángulo a lo largo de esta. (Figura 5)(9)

Pasos para el proceso de triangulación

La malla es un cuadrante básico que posee cuatro puntos, uno en cada una de sus aristas, se unen dos de sus puntos y forman dos triángulos, se une el punto medio del cuadrante con sus otros dos puntos y conforman 4 triángulos, después se lleva a cabo la triangulación uniendo los puntos medios de las aristas con el punto medio del cuadrante. (Figura 6)

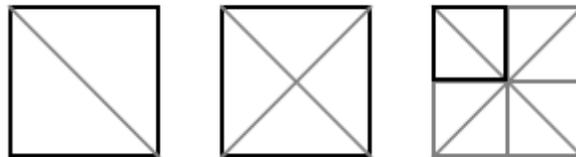


Figura 5 Pasos para el proceso de triangulación

Para lograr la representación mediante un Quadtree se utiliza un arreglo bidimensional de 0 y 1, donde le atribuye 1 si el vértice va a formar parte de la triangulación. (Figura 7)(9)

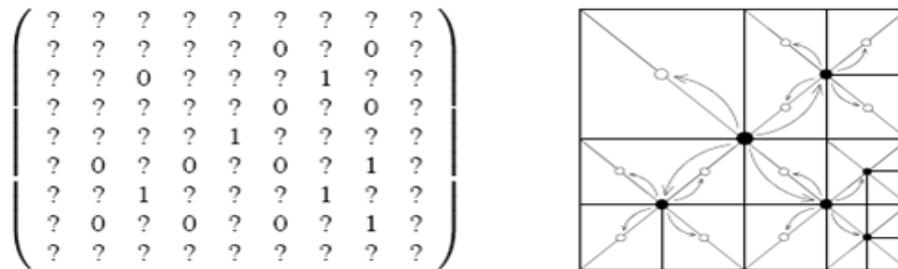


Figura 6 Arreglo bidimensional de 0 y 1, 1 si el vértice forma parte de la triangulación

Para evitar el proceso de grietas en los cuadrantes es necesario tener en cuenta la dependencia entre vértices. Los Quadtree restringido son muy eficientes en la visualización de la superficie del terreno, sin embargo, posee una dificultad ya que cuando va a seleccionar la estructura de un cuadrante para ser triangulado provoca el efecto cascada cuando se selecciona sus vecinos, esto provoca que puedan ser seleccionados cuadrantes que no influyan en la calidad de visualización y sería mayor el número de triángulos y mayor lentitud en la velocidad de representación.

➤ **Quadtree no restringido**

En esta estructura el proceso de división de cada cuadrante se lleva a cabo por separado de sus vecinos, esto permite que no esté balanceada la estructura.

Para llevar a cabo el proceso de división se adiciona el punto medio del cuadrante y en adelante se lleva a cabo la unión de este punto con los que se encuentran en la frontera del cuadrante (Figura 9) (9)

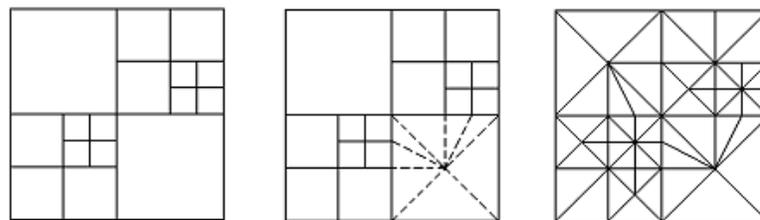


Figura 7 Proceso de división

Para esta estructura se puede utilizar también un método donde se utilizan listas que están ordenadas de acuerdo con el orden espacial que tienen los vértices (9). La primera lista contiene los vértices que poseen una misma coordenada y, están ordenadas por la coordenada z y la segunda, las que poseen una misma

coordenada x, ordenada por la coordenada y. Para insertar los vértices en las listas se definen con $O(1)$. (Figura 10)

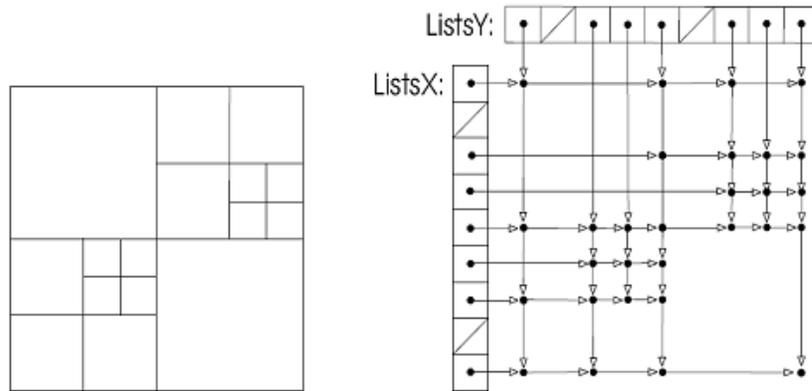


Figura 8 Representación por listas

2.3.2 Mallas Regulares R-Trianguladas y Bitrees no Restringidos

Otra de las técnicas muy utilizada son las mallas regulares, que son una estructura de datos muy eficiente para representar modelos en memoria, al realizar la triangulación los triángulos que se obtienen son rectángulos o isósceles. Los triángulos son representados en un árbol binario, donde se le da valor 0 o 1 a la codificación de su ancestro. (Figura 12) (9)

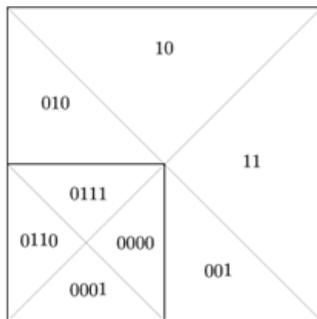


Figura 9 Asignar valores a la triangulación

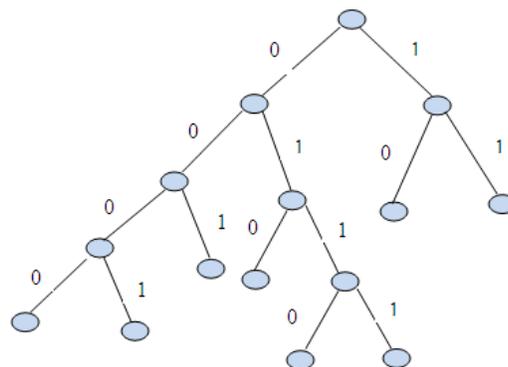


Figura 10 Representación de la triangulación en forma de árbol

El proceso de triangulación se realiza trazando las diagonales del cuadrante principal de la malla, luego se adicionan puntos en la malla a las hipotenusas de cada uno de los triángulos. (Figura 13)

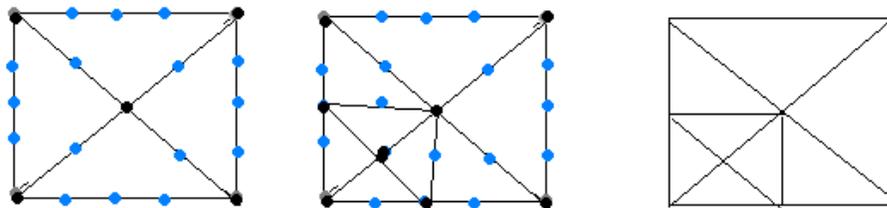


Figura 11 Proceso de triangulación

Al igual que en los Quadtree restringido la desventaja fundamental es que en el proceso de división de los triángulos se genera una gran cantidad de polígonos para procesar, además de que se produce un efecto cascada de división en su vecino por la hipotenusa, lo cual provoca una disminución en la velocidad del proceso representación. (9)

➤ Bitrees no Restringidos

Una alternativa del RTIN²⁵ es la NRBT²⁶ que minimiza el número de triángulos; debido a que solo realiza una etapa del proceso de subdivisión de los triángulos. En este método se presenta un triángulo que es subdividido, adicionando el punto medio de su hipotenusa como en el caso de RTIN. (Figura 14)

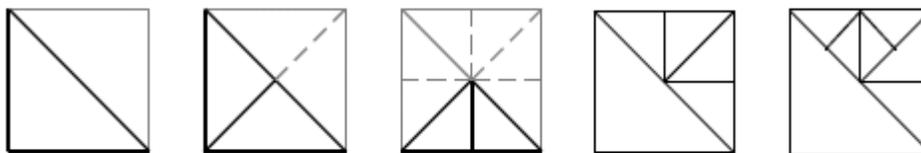


Figura 12 Proceso de subdivisión de los triángulos

En el proceso de subdivisión cada triángulo puede ser subdividido independiente de sus vecinos y así se elimina el efecto cascada en la triangulación. (9)

2.3.3 Mallas Adaptables Óptimas en Tiempo Real

Es un algoritmo para construir mallas triangulares que optimiza métricas de error flexibles dependientes de la vista. Una coherencia cuadro a cuadro para operar a altas velocidades mostrando miles de triángulos.

²⁵ RTIN acrónimo de Mallas Regulares R-Trianguladas

²⁶ NRBT acrónimo de Triangulación sobre Bitrees no Restringidos

El método ROAM usa dos colas de prioridad para operaciones de división y fusión que mantienen triangulaciones continuas a partir de árboles binarios de triángulos previamente procesados. (10)

- El algoritmo ROAM consiste de un componente para pre-procesamiento y distintos componentes para tiempo de ejecución.
- El componente de pre-procesamiento produce límites de error, anidados e independientes, de la lista, para un árbol binario de triángulos, de abajo hacia arriba.
- En tiempo de ejecución, se realizan cuatro fases por cada cuadro desplegado, y son las siguientes.

Fases de ejecución (12)

- Actualizar recursiva e incrementalmente el recorte del frustum²⁷, o área visible.
- Actualizar la prioridad para los triángulos que se dibujarán y con posibilidades de ser fusionados o divididos en la fase 3.
- Actualizar la triangulación usando pasos de algoritmos *avaros* de división y fusión ejecutados por las respectivas colas de prioridad.
- Actualizar las tiras de triángulos afectadas por los cambios de recorte de la fase 1 y las uniones y divisiones de la fase 3.

Recorte de Frustum (12)

- Se debe eliminar aquellas áreas que no sean visibles el frustum.
- Se asume que el área visible está definida como la intersección de seis medios espacios.
- Cada medio espacio es dividido por un plano.
- A cada triángulo se le asigna una bandera dentro para cada medio espacio.
- La bandera se marca cuando su cuña se encuentra completamente dentro del correspondiente medio espacio.
- Dependiendo de en qué parte se encuentre la cuña en estos espacios se le asigna una etiqueta al triángulo.

La etiqueta:

Se dice que esta fuera cuando la cuña se encuentra fuera de algún medio espacio.

²⁷ **Frustum:** porción de una figura geométrica, simula el comportamiento de una cámara virtual.

- Todo-dentro se asigna cuando la cuña se encuentra completamente dentro de todos los medios espacios.
- Quien-sabe en otro caso, lo cual sucede cuando la cuña no está completamente dentro de algún medio espacio.

Ventajas

- Su desempeño es independiente del tamaño y resolución de la base de datos de terreno.
- El uso de un árbol binario para el cálculo, y la generación de mallas adaptativas y continuas a través de pasos incrementales granulares produce siempre triángulos rectángulos isósceles.
- El algoritmo es simple de entender y de implementar.
 - Está basado en operaciones de división y fusión de un árbol binario de triángulos.
 - Esta estructura evita la proliferación de casos complejos y reglas de validación para casos especiales.
 - No necesita pasos adicionales para evitar discontinuidades.
 - El mecanismo avaro para determinar la prioridad de las colas permite extender mejoras en la métrica de errores de una forma simple.
- Usando ROAM, es posible producir mallas con un número específico de triángulos.
- Permite aumentar el número de polígonos que se puede mostrar.

Representación de la malla (12)

- Un árbol binario de triángulos, similar a los árboles binarios de rectángulos, parte de un triángulo isósceles raíz, el cual se dice que se encuentra en el nivel 0 del triángulo.
- En el siguiente nivel de refinamiento, los hijos se definirán dividiendo la raíz a lo largo del eje formado entre el eje ápice y el punto medio del borde base.
- Este proceso formará dos nuevos triángulos, que compartirán estos puntos. (Figura 15)
- El proceso se repite recursivamente, repitiendo esta división.

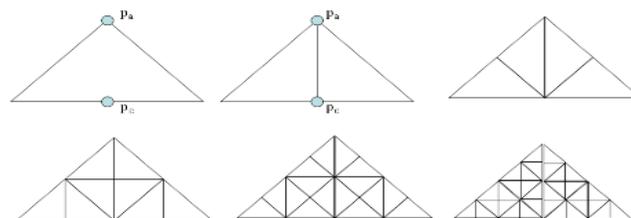


Figura 13 Proceso de triangulación para la malla

Las mallas se conforman, asignando las posiciones de los vértices a cada nodo del árbol binario.

Un conjunto de triángulos del árbol forma una malla continua cuando cualquier par de triángulos

- ✓ No coincide entre sí.
- ✓ Coincide en un vértice común.
- ✓ Coincidir en una arista común.

Un triángulo es considerado vecino de la base cuando comparte la base. (Figura 16)

Es vecino izquierdo cuando comparte la arista izquierda. (Figura 17)

Es vecino derecho cuando comparte la arista derecha. (Figura 18)

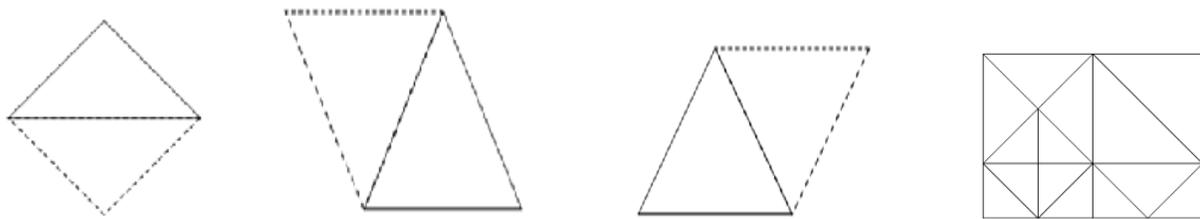


Figura 14 Vecino de la base Figura 15 Vecino izquierdo Figura 16 Vecino derecho Figura 17 Diferentes niveles

En la triangulación los vecinos pueden estar en un mismo nivel, en el siguiente nivel con vecinos izquierdos y derechos y en un nuevo nivel menos detallado con vecinos base. (Figura 19)

- Cuando un par de vecinos base está en el mismo nivel se le llama diamante.
- La división reemplaza ambos triángulos por sus hijos e introduce un nuevo vértice en el centro del diamante.
- Esto resulta en una nueva triangulación continua.
- Si un triángulo no tiene vecino base, sólo se divide ese triángulo.
- Para fusionar un diamante, todos los hijos de los dos vecinos base deben ser parte de la triangulación, refiriéndonos en este caso a un diamante fusionable.
- Una propiedad importante de las operaciones de división y fusión es que cualquier triangulación puede obtenerse a partir de una secuencia de divisiones y fusiones. (Figura 20)

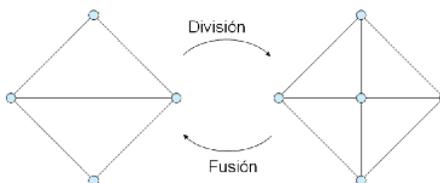


Figura 18 Proceso de división y fusión

- Una malla base de más de un triángulo se puede utilizar para representar superficies arbitrarias.
- Si esta malla base puede organizarse en diamantes, entonces las operaciones de división y fusión pueden usarse libremente como en el caso de un solo triángulo raíz.
- En el caso del terreno, una malla base típica puede representarse como un diamante único.

Un triángulo no puede dividirse automáticamente si su vecino base es de un nivel menos detallado. (Figura 21).

En este caso, primero debe forzarse la división del vecino menos detallado, lo cual puede causar más divisiones. Esto se requiere para que funcione el algoritmo de optimización. (Figura 22)

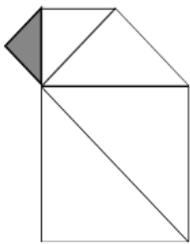


Figura 19 Condición de división

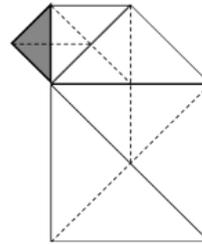


Figura 20 Forzar la división

Proceso de división y fusión (12)

- ✓ Las operaciones de dividir y fusionar conforman un marco de trabajo flexible para hacer actualizaciones precisas a una triangulación, evitando huecos en la malla.
- ✓ El proceso de división y fusión utiliza un algoritmo avaro cuya idea es simple: mantener prioridades para cada triángulo en la triangulación, empezando con el triángulo base, y repetidamente haciendo una división forzada en el triángulo de mayor prioridad.
- ✓ Este proceso crea una secuencia de triangulaciones que minimizan la máxima prioridad, que normalmente es un límite de error, en cada paso.
- ✓ El único requisito para garantizar que sea óptimo es que las prioridades deben mantener el orden; es decir, que la prioridad de un hijo no debe ser mayor que la de su padre.
- ✓ Si se agrega una segunda cola de prioridades, para diamantes que se puedan fusionar, se le permite al algoritmo avaro iniciar de una triangulación previamente óptima cuando las prioridades han cambiado.
- ✓ Esto permite aprovechar las similitudes entre un cuadro y el siguiente.

Cola de división

En la cola de división cada triángulo tiene una prioridad entre 0 y 1 donde la triangulación se construye de arriba hacia abajo. Se mantiene una cola de prioridades con todos los triángulos de la triangulación actual, se inicia con la triangulación base.

Algoritmo de división

Se insertan los triángulos de la triangulación base en la cola. Cuando la triangulación es muy pequeña o imprecisa, se identifican los triángulos de mayor prioridad, se lleva a cabo la división de este y se actualiza en la cola.

Cola de fusión

En la cola de fusión las prioridades son variables dependiendo del cuadro. El problema consiste en construir triangulaciones óptimas para cada cuadro. Si las prioridades cambian lenta y gradualmente, las triangulaciones óptimas para cada par de cuadros consecutivos serán muy similares entre sí. Se puede utilizar la triangulación del cuadro anterior como base para el siguiente triángulo.

Proceso de Fusión

En el proceso de fusión, se mantiene una segunda cola de prioridades que contiene todos los diamantes a fusionar para la triangulación actual. La prioridad para un diamante fusionable se define como el máximo de las prioridades de sus dos triángulos.

Algoritmo de fusión/división (12)

- ✓ Si es el primer cuadro
 - Borrar ambas colas
 - Para cada diamante y triángulo de la triangulación base.
 - Calcular su prioridad.
 - Insertar en la cola correspondiente.
- ✓ Si es otro cuadro
 - Para cada diamante y triángulo de la triangulación anterior calcular su prioridad.
 - Actualizar en la cola correspondiente

- ✓ Mientras que la triangulación sea muy pequeña o imprecisa, o la prioridad máxima de división sea mayor que la prioridad mínima de fusión.
- ✓ Si la triangulación es muy grande o precisa
 - Buscar el diamante a fusionar de menor prioridad.
 - Fusionar este diamante.
 - Actualizar las colas como sigue.
 - Quitar los hijos fusionados de la cola de división
 - Agregar los padres fusionados a la cola de división.
 - Quitar el diamante a fusionar de la cola de fusión.
 - Agregar nuevos diamantes fusionables a la cola de fusión.
 - Si la triangulación no es muy grande o es imprecisa
 - Buscar el triángulo de mayor prioridad
 - Forzar la división de dicho triángulo
 - Actualizar las colas como sigue
 - Quitar los triángulos divididos de la cola de división.
 - Agregar los nuevos triángulos a la cola de división.
 - Agregar nuevos diamantes fusionables a la cola de fusión
- ✓ Actualizar la triangulación actual

Métrica de error

Métrica de error se utiliza para calcular las prioridades de las colas. Se pueden utilizar para campos o mapas de alturas. Un mapa de alturas se caracteriza porque la altura de un punto puede definirse en función de sus coordenadas en el plano.

Envoltentes anidados en el espacio del mundo (10)

Es una cuña que se define como el volumen de espacio del mundo contenido los puntos (x, y, z) . La diferencia entre z y z_t correspondiente a las coordenadas (x, y) , sea menor que c_t .

En un triángulo padre el grosor de la cuña e_t se define en términos de los grosores de cuña de sus hijos, c_{t_0} y e_{t_1} , y de la altura en el centro del segmento base $z(vc)$. (Figura 23)

z_t : valor de z en la triangulación

ct : ancho de la cuña

et : grosor de la cuña

vc : centro del segmento

$$ct = \max(ct_0 + ct_1) + |z(vc) - zt(vc)|$$

$$zt(vc) = (z(v_0) + z(v_1)) / 2$$

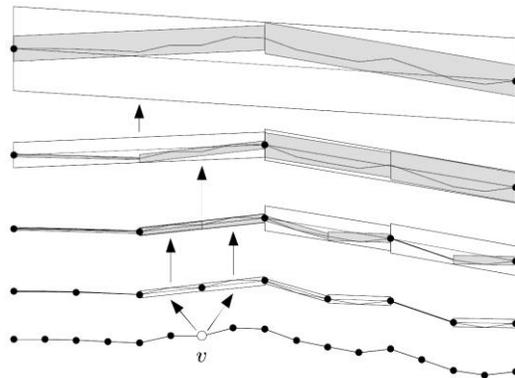


Figura 21 Grosor de la cuña

Distorsión geométrica en pantalla

Una distorsión geométrica no es más que la distancia entre donde una superficie debe aparecer en la pantalla y donde aparece el punto debido a la triangulación. Debido a esto en la imagen producida, se medirán los máximos de estas distorsiones, lo cual constituye la métrica base para ROAM. Esta métrica se calcula como el límite superior a partir de la distorsión máxima. Luego para cada triángulo, se obtiene un límite superior de distorsión proyectando la cuña de dicho triángulo en la pantalla. El límite se define como la longitud máxima de los segmentos de grosor proyectados para cada vértice de la triangulación. La máxima prioridad de la cola de división provee un límite superior en la máxima distorsión.

2.3.4 Geomipmapping

Desarrollado por Willem H. de Boer, es un algoritmo que es amigable con la GPU, haciendo uso intensivo de esta y liberando la carga de trabajo del CPU, una tendencia a seguir debido a la mayor capacidad de proceso de las nuevas tarjetas gráficas. Es también un algoritmo simple que es perfecto para tu transición en el mundo de los terrenos CLOD.

Es un algoritmo para gestionar el nivel de detalle del terreno. Que persigue como objetivo que las zonas del terreno que están más alejadas de la cámara no necesitan tantos polígonos para dibujarse como las que están cerca, ya que de todos modos se van a ver más pequeños y no se van a apreciar. Se basa en usar un mecanismo parecido al que se usa normalmente en las texturas: mipmapping. El primer nivel de la cadena de mipmaps sería la textura en sí, mientras que cada nivel por debajo sería una simplificación de la misma (menor resolución), creando así varios modelos hasta que se llega a una profundidad satisfactoria. Cuando un trozo de la malla se encuentra a una determinada distancia de la cámara, se carga el nivel adecuado. Estos conceptos son lo mismo, excepto que en lugar de tratar con texturas, se trata con vértices de un bloque de terreno. (11)

Si se aplica esto a un mapa de altura se pueden crear bloques de puntos de un tamaño dado ($n \times n$). En un nivel de detalle máximo (nivel 0) se dibujarían todos los vértices mientras que si la cámara se aleja, el detalle cambiaría a un nivel más bajo en el que se dibujan menos vértices. (Figura 24). (11)

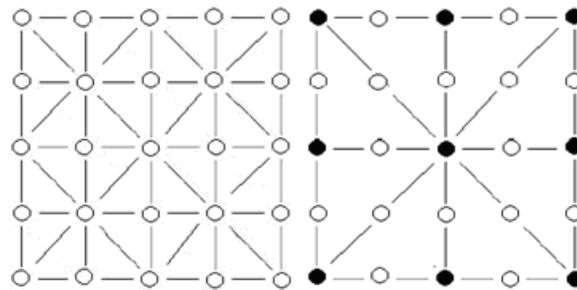


Figura 22 Bloques de puntos de diferentes niveles

En la imagen se aprecia un bloque de 5×5 . En la izquierda se pintan todos los vértices, pues es el nivel 0 o de máximo detalle. En el dibujo de la derecha solo se pintan los vértices coloreados debido a que el nivel sería menor.

Cuando dos bloques contiguos son de distinto nivel de detalle forman agujeros en sus fronteras. Esto se produce porque el bloque de más detalle dibuja vértices en sus extremos sin embargo el otro no lo dibuja. Estos se dibujan a su altura determinada en el mapa de alturas, pero en el bloque con menos detalle el triángulo formado por los vértices que sí se dibujan no tiene porque pasar por ese vértice. Los dos triángulos del bloque con más resolución estarían en planos distintos, y entre estos y el triángulo del otro bloque quedarían huecos. (Figura 25)(11)

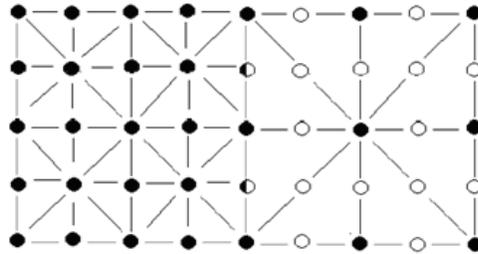


Figura 23 Modo de evitar huecos

La solución a este problema es omitir este vértice en los bloques que se encuentren situados junto a un bloque de menor resolución. Para esto es necesario realizar una serie de comprobaciones antes de dibujar cada bloque.

Si se almacena los bloques en un arreglo, cada vez que la cámara se mueva se tendría que recorrer el arreglo para actualizar todos los valores del nivel de detalle y comprobar antes de dibujar si los vértices frontera forman huecos. Para esto se pueden usar cuatro valores booleanos, uno para cada posible vértice a omitir (superior, inferior, izquierdo y derecho), si la variable tiene el valor false, el vértice correspondiente no se dibuja. La función de dibujar iría recorriendo el arreglo de bloques y actualizando los valores booleanos comprobando el nivel de detalle de los bloques adyacentes. Si uno de estos bloques tiene menos nivel de detalle, entonces el valor de la variable correspondiente debe ser false y ese vértice no debe dibujarse.

2.3.5 Hilbert-Quadtree

Es la combinación de la curva de recorrido del espacio de Hilbert con la estructura de datos Quadtree. Curva de recorrido del espacio: es el camino descrito por el movimiento repetido de un punto; son muy utilizadas para las búsquedas, como por ejemplo la búsqueda de vecinos en las estructuras de datos. Se tiene una malla rectangular de orden $N \times N$ de donde es definida SFC²⁸ como una numeración de las celdas de la malla con valores desde 1 hasta N^2 , para $c \geq 0$. Una numeración de la malla consiste en uno a uno de la forma: $P: N \times N \rightarrow \{1, \dots, N^2\}$ (9)

Esta curva se dice que es recursiva, cuando se puede dividir la malla SFC en cuatro RSFC²⁹ cuadradas de igual tamaño. (Figura 26)

²⁸SFC: acrónimo de *Space-filling Curve* o Curva de recorrido del espacio

²⁹RSFC: acrónimo de *Recursive Space-filling Curve* o Curva de recorrido del espacio recursiva

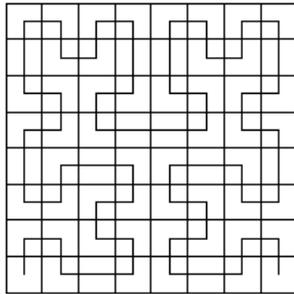


Figura 24 División de la malla SFC

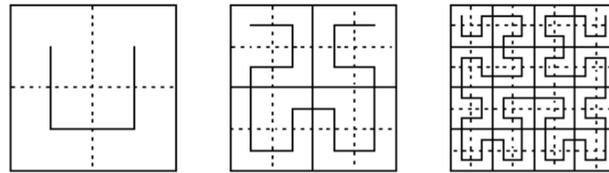


Figura 25 Diferentes niveles de la curva de Hilbert

A pesar de la complejidad en la manipulación de las RSFC, son muy utilizadas en el diseño de las estructuras de datos espaciales (Figura 27) (9), ya que permite de forma eficiente la modelación de superficies de terrenos y la generación de cadenas de triángulos para su visualización.

En esta estructura de datos no se requiere almacenar la información de la malla sobre nodos hijos, padres o vecinos como los ejemplos vistos anteriormente. (Figura 28)

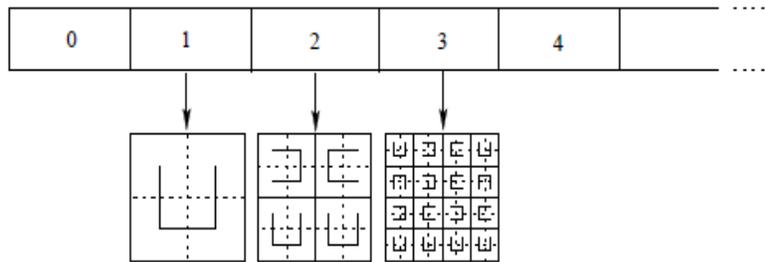


Figura 26 Representación de los tres primeros niveles de la Hilbert-Quadtree

El punto de partida es la malla rectangular la cual se va dividiendo recursivamente a cada paso del algoritmo. Cada nodo del Quadtree es representado como un segmento o patrón de la curva de Hilbert. (Figura 29)



Figura 27 Patrones de Hilbert: abajo, derecha, izquierda, arriba

Como se puede apreciar en la (Figura 28) el Quadtree es almacenado en un arreglo unidimensional, donde cada posición almacena otro arreglo bidimensional de patrones de Hilbert. Cada índice del arreglo unidimensional representa los niveles del Quadtree, donde la posición 0 permanece vacía, y el caso de arreglo bidimensional cada uno almacena la información correspondiente a los nodos que corresponden a ese nivel. (9)

Una ventaja importante que posee esta representación es que cada patrón corresponde a un nuevo punto introducido en cada paso del proceso de división y representa cuatro cuadrantes, por lo que la información de los cuadrantes de máxima resolución no es almacenada implícitamente en la estructura. (9)

➤ Generación de cadenas de triángulos

La obtención de cadenas de triángulos a través de curvas de recorrido del espacio ha sido muy utilizada para generar de manera eficiente las triangulaciones y así ganar mayor rapidez, todo esto a partir de estructuras de datos jerárquicas.

La triangulación mediante Hilbert-Quadtree se basa en la curva de Hilbert y la información que necesita para la construcción de las cadenas se encuentra almacenada en el modelo (Figura 30). El procedimiento recursivo de creación de la estructura siempre comienza con un patrón de Hilbert Abajo, el inicio de la curva de recorrido del espacio siempre se encuentra en la región noroeste, mientras que el final se localiza en el área noreste. Siguiendo la curva, para cada nodo en la estructura se adicionan sus vértices a la cadena de triángulos siguiendo algunas reglas. (Figura 31)(9)

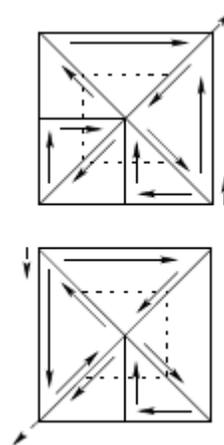
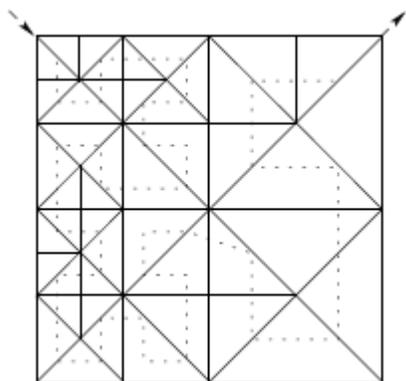


Figura 28 Superficie triangulada a través de una curva de Hilbert. Figura 29 Triangulación de los patrones de Hilbert

La triangulación que es llevada a cabo por un patrón de Hilbert:

- Abajo: comienza (termina) en el cuadrante superior-izquierdo (superior-derecho).
- Derecha: son triangulados comenzando en el cuadrante superior-izquierdo y terminando en el inferior-izquierdo.
- Izquierdo: en este caso comienza y termina en el cuadrante inferior-derecho (superior-derecho).

- Arriba: son triangulados partiendo del cuadrante inferior-derecho y terminan en el inferior-izquierdo.

Los cuadrantes que van a ser visualizados son seleccionados de forma que la diferencia de niveles entre cuadrantes adyacentes sea menor o igual a uno, como en el caso de la triangulación sobre Quadtree restringido. El punto de partida de la curva de Hilbert es encontrado descendiendo recursivamente por la región noroeste del Quadtree hasta que un cuadrante indivisible es encontrado. Una vez encontrado el inicio, siguiendo las direcciones almacenadas en la estructura, cada cuadrante es triangulado en forma continua. (9)

2.4 Herramientas y tecnologías

Las aplicaciones Web hacen uso de JavaScript, el cual cuenta con varias librerías o framework para facilitar la programación de estas aplicaciones y la representación en 3D. JavaScript se ha vuelto imprescindible en el concepto de las Apis Web 2.0, ofrece una interfaz desde el lado del cliente para usar un complejo funcionamiento desde el lado del servidor.

2.4.1 Librerías y Motores en JavaScript

2.4.1.1 JS3D

JS3D es una librería libre de JavaScript bajo la licencia GPL que permite crear objetos 3D interactivos en nuestra Web. Requiere conocimientos de JavaScript y HTML a la hora de utilizarla. Ha sido probado en Firefox y Safari, y no es compatible con Internet Explorer. La librería es descargada y guardada en un directorio de la página para luego importarla para el cuerpo de la página, solo brinda la opción de visualizar objetos a través de puntos y líneas y con estas crear formas como por ejemplo cubos, dentro del cuerpo de la página se encuentra una página que funciona como tapiz donde se pinta los puntos o líneas que formaran la figura a representar en 3D, esta capa viene como un archivo JS3D que se encuentra entre los elementos de la página. La transformación y rotación de la escena se lleva a cabo mediante variables que van tomando valores entrados por pantalla y luego se pinta la escena a través de estos valores.

Dentro del código de la página se añade una función que es la encargada del proceso de dibujar, para eso se especifica los valores de las coordenadas x, y, z y un último valor que sería el símbolo con el que voy a dibujar, cuenta con otras funcionalidades para rotar, donde a cada valor de las coordenadas le

corresponde una funcionalidad especificando el valor y ángulo. Luego que se obtienen todos los valores y se llaman las funcionalidades se lleva a cabo el proceso de pintar que se realiza a través de Canvas. (13)

2.4.1.2 SVG-VML-3D

Es una librería libre de JavaScript que puede ser empleada para extraer y manipular objetos en 3D en la Web y escenas de gráfico, el código JavaScript que se escribe en la página HTML es la misma para SVG y VML, estos se utilizan según el navegador para el caso de Internet Explorer sería VML, en otro caso sería SVG. El archivo tiene que ser incluido en la biblioteca, además puede utilizar también gráficas 3D que deben ser incluidas. (14)

SVG³⁰: Es un lenguaje para describir gráficos que se generan de forma rápida y eficaz. Está basado en XML³¹ lo que permite que las compañías de software puedan ampliarlo y aplicarlo según sus necesidades, sin perder el prototipo elemental. Utiliza el modo texto que facilita el control sobre el modelado de formas. Permite incluir código JavaScript y referencias a hojas de estilo en cascada y el flujo de trabajo de diseño se puede ejecutar desde una sola aplicación. Su desarrollo está a cargo del consorcio W3C por lo tanto las compañías pueden utilizarlo sin necesidad de un costo de patente. Permite a los navegadores proporcionar funciones integradas de gráficos, pero necesita de un plugin para poder visualizar el contenido de la página. El visor SVG se integra con el navegador Web como un plugin que le permite interactuar con las imágenes SVG. (14)

Tiene un visor gratuito y sencillo que se trata de un plugin desarrollado por Adobe, que se encuentra en su versión 3 y está disponible en www.adobe.com/svg; en el caso del Netscape, el plugin debe ser incorporado de manera manual a la carpeta correcta, ya que Netscape sufre un error que lo bloquea cuando debe interpretar un gráfico escalable que se encuentra en la página, pero es necesario instalarle un plugin para poder ver el contenido de la página.

Existen otros visores que utilizan SVG: (14)

³⁰ **SVG**: acrónimo de Gráficos vectoriales escalables

³¹ **XML**: acrónimo de *Extensible Markup Language*

- ✓ Batik SVG browser, de la fundación Apache. Un navegador SVG que funciona bajo plataforma Java.
- ✓ SdVG viewer, de Bitflash, que incluye soporte para dispositivos móviles como los organizadores personales, como las Palm.
- ✓ SVG en Mozilla *project*, para dotar compatibilidad con el formato en navegadores de la familia Mozilla.

El SVG permite representar tres tipos gráficos: las gráficas de vectores: pueden representar caminos a través de rectas y curvas; las imágenes de mapa de bits: utiliza la estructura de datos ráster y la de texto.

VML³²: Fue propuesto por la el W3C³³ como un estándar para gráficos vectoriales en la Web y es una especificación de XML, se utiliza por Microsoft Internet Explorer y no requiere de un plugin para su navegador. Dentro del contenido se compone de rutas descritas usando líneas y curvas conectadas. Está escrito con la sintaxis de XML como HTML utilizando la sintaxis de SGML³⁴ (con la norma ISO 8879). (15)

Dada la estructura global de VML puede resumirse en las definiciones XML de los dos elementos principales la forma y el grupo. Un elemento forma se utiliza para definir un elemento gráfico de un vector visible. La mayoría de las formas tienen una definición de camino, una secuencia de líneas rectas y curvas cúbicas que define un esquema, el mismo puede ser trazado, según las especificaciones de los atributos de la forma y los estilos de sus subelementos. Participa plenamente en el modelo de representación visual CSS2 y permite la rotación, donde cada elemento tiene un espacio de coordenadas.

Beneficios de VML: (15)

- ✚ VML hace más fácil la autoría para la productividad de usuarios y autores. Facilita el intercambio (a través de cortar y pegar) y la edición de gráficos vectoriales entre una amplia variedad de aplicaciones de productividad y diseño.

³² **VML**: acrónimo de Lenguaje de Marcado Vectorial.

³³ **W3C**: acrónimo de *World Wide Web*. Consorcio internacional.

³⁴ **SGML**: acrónimo de *Standard Generalized Markup Language*.

- ✚ VML ofrece la descarga de gráfico de forma más rápida y una mejor experiencia de usuario. Permite una entrega de alta calidad que integra los gráficos de vector escalables a la Web, en un formato abierto basado en texto. En lugar de hacer referencia a los gráficos como archivos externos, los gráficos VML se entregan en línea con la página HTML, lo que le permite una mejor interacción con el usuario. (15)

2.4.1.3 JavaScript Raytracer

JavaScript Raytracer es un motor de representación 3D escrito totalmente en JavaScript. Implementa el algoritmo de trazado de rayos para crear imágenes de formas definidas matemáticamente como esferas. Es computacionalmente costosa y el lenguaje interpretado de JavaScript no es la mejor opción para las aplicaciones de trazado de rayos. Para evitar que sea muy larga la demora a la hora de visualizar la imagen, el trazador de rayos lo que hace es ir mostrando la imagen por un nivel de resolución, primero lo hace con un nivel más bajo y luego se va incrementando la resolución.

Existen dos formas de representación, la representación de una imagen de mayor calidad, pero puede causar lentitud en los navegadores, debido a que la representación no se hace solamente en base de tres por tres cuadrados de píxel. Para la compatibilidad máxima de la imagen se compone de un generador de visualización para cada elemento <div> dentro del cuerpo de la página. Lo que hace este motor es arrojar un rayo de luz sobre la escena (conjunto de formas y luces de todo lo representado en el espacio 3D), cada píxel de la imagen sería el efecto del rebote del rayo sobre una parte de cada objeto de la escena, estas operaciones son muy costosas.

2.4.2 Herramientas para Adobe Flash

2.4.2.1 PaperVision 3D

PaperVision 3D: Es un API para flash de código abierto, el cual se puede incorporar tanto en aplicaciones Web como en videojuegos. Este framework es muy utilizado para proyectos pequeños y medianos donde se quiere mostrar algunos elementos 3D incidentales o juegos de un solo jugador que no requieran de un servidor multiusuario. Tiene soporte integrado Quadtree, que divide la pantalla en regiones cada vez más pequeños para resolver los posibles conflictos entre triángulos, no es el primer motor que incorpora Quadtree en 3D Flash dado que otros también lo utilizan, como por ejemplo Away3D. Permite hacer

escenas en 3D que integran luces, texturas, sombras entre otros para obtener objetos totalmente interactivos. (16)

En el 2009 junto con otros dos proyectos de la PV3D fue nominado para mejor premio Web en .Net Magazine³⁵ ha sido seleccionado también como finalista para la aplicación de código abierto del año. (16)

✚ Ventajas:

- ✓ Integra modelos 3D rápidamente, sobre todo si se utiliza modelos collada³⁶ o directamente exportados de Autodesk 3d Max Studio³⁷ o maya³⁸.
- ✓ La integración de texturas a los modelos es rápida.
- ✓ Las librerías para interactuar con el objeto 3D permite crear interacciones comunes rápidamente.
- ✓ Hay muchísima documentación, sitios, ejemplos, tutoriales, etc.

✚ Desventajas:

- ✓ El manejo del orden de la malla y del orden de z en un modelo integrado tiende a dar problemas a partir del nivel 3.
- ✓ El manejo de varias instancias de modelos dentro del cliente realizando alguna interacción individual, como moverse o rotar hace que el cliente se vuelva lento, se trabe, se cierre, explote y que los recursos de la maquina se consuman por completo.
- ✓ Combinar más de un modelo y un video. flv de más de 350 px en el cliente realizando interacciones individuales, consume todos los recursos de la maquina, por lo que todo se vuelve excesivamente lento.
- ✓ La orientación en el plano cartesiano de un objeto 3D con elementos 2D o objetos de visualización común se vuelve un poco difícil debido a que debe utilizar un plano 3D o un 3dDisplayobject³⁹ Container para poder combinarlos, por lo que a la hora de concebir donde esta cada elemento fuera de un plano 3D, es decir 2D, se vuelve más difícil si hay que rotar, mover o similares y puede explotar el programa a hora de ver las animaciones.

³⁵ **.Net Magazine:** revista.net de la compañía de las aplicaciones net ejemplo c#.

³⁶ **collada:** proyecto de la familia libre de XML que permite el intercambio de activos digitales dentro de la industria interactiva 3D.

³⁷ **Autodesk 3d Max Studio:** programa de creación de gráficos y animación 3D desarrollado por Autodesk.

³⁸ **Maya:** programa para el modelado y diseño gráfico en 3D de altas prestaciones, desarrollado por la empresa Alias System Corporation (adquirida por Autodesk en 2005).

³⁹ **3dDisplayobject:** objeto de visualización en 3D.

- ✓ Su aceleración por hardware de gráficos 3D es muy pobre.

2.4.2.2 Away 3D

Es un framework de código abierto que tiene una gran potencia y simplicidad en cuanto a la representación en 3D en tiempo real para su uso con Flash y Flex, es una librería de código abierto que permite hacer aplicaciones usando 3D con Flash Builder y ActionScript. Se recomienda para proyectos pequeños y medianos que no utilicen servidores multiusuario. Requiere la instalación de un cliente de svn⁴⁰. (17)

+ Ventajas:

- ✓ El rendimiento es excelente aún con varias instancias de objetos 3D corriendo al mismo tiempo y realizando interacciones individuales, mucho mejor que lo que ofrece PaperVision.
- ✓ Las librerías para interactuar con los objetos 3D ofrecen mayores alternativas que las que tiene PaperVision.
- ✓ El código de las librerías está mucho más depurado que el de PaperVision por lo que utiliza menos recursos.

+ Desventajas:

- ✓ No es fácil importar modelos al proyecto y el comportamiento de estos es errático.
- ✓ El manejo de la cámara no es tan certero como PaperVision.
- ✓ Hay poca documentación y tutoriales.

2.4.2.3 Alternativa Platform

Es un motor 3D para Flash que soporta la conexión multiusuario en tiempo real de los clientes y jugadores, debido a su alta productividad en Java del servidor, provee la oportunidad de probar y de trabajo para los desarrolladores. Permite mostrar mundos tridimensionales, juegos, visitas virtuales o los objetos en el navegador. Es recomendable para crear proyectos medianos o grandes. (18)

+ Ventajas:

⁴⁰ **svn**: cliente gráfico para la versión de código abierto sistema de control.

- ✓ Completa vista 3D en el navegador de proyectos de cualquier tipo.
- ✓ La creación de complejas escenas en 3D en Flash: visualización de objetos, mecanismos, edificios, objetos de importación de formatos de 3D, la carga de la textura, incluso animados queridos.
- ✓ Simulación física básica (deslizamiento, colisión, accidente).
- ✓ Interacción de 5000 usuarios en tiempo real en un servidor físico con la oportunidad de mejora en cuanto a conexión.
- ✓ Fácil interfaz de usuario para Flash.
- ✓ Metodología de desarrollo disponible (completo y con profundidad de la información, foro de participación activa, libre curso básico de apoyo).

✚ Desventajas:

- ✓ El framework viene dentro de por lo que no se puede modificar las clases y por tanto no permite resolver errores o problemas que se tengan en el proyecto de índole del framework.
- ✓ Su GUI está en construcción y no se puede utilizar.
- ✓ La solución para la interacción cliente servidor para juegos multiusuario, está en construcción
- ✓ La documentación es muy pobre.

2.4.3 Plugin para navegadores

FreeWRL: es un plugin de navegador bastante sencillo y totalmente gratuito bajo la licencia GPL que permite la manipulación de la escena 3D con tan solo el teclado y el ratón, ofreciendo un control sobre los objetos. Funciona con la mayoría de los navegadores Web y también se puede acceder mediante interfaces EAI⁴¹ o SAI⁴² para activar el control de contenido visual. Es multiplataforma entre las que se encuentra Linux, Mac OS / X y otras plataformas al estilo Unix. Para la visualización utiliza los ficheros en formato VRML 2.0, VRML 97 y X3D, para GNU/Linux y OSX. Permite representar de forma rápida y sencilla modelos tridimensionales de objetos, sin la necesidad de desarrollar un sistema de visualización que haga uso de OpenGL u otro. FreeWRL fue desarrollado por los contribuyentes de código abierto, dirigido por John Stewart y sigue el Consorcio Web3D de las normas ISO. (24)

⁴¹ **EAI:** acrónimo de *Enterprise Application Integration* o Integración de Aplicaciones de Empresa

⁴² **SAI:** acrónimo de Interfaz de Sensores y Actuadores

Requiere instalar un conjunto de paquetes, y en el caso de Ubuntu puede darse el caso de que en el proceso de instalación de estos paquetes desinstale algunas aplicaciones importantes como puede ser el caso del navegador Firefox 3.5.

Web3D hace referencia a cualquier lenguaje de programación, protocolo, formato de archivo o tecnología que pueda ser usado para la creación y presentación de universos tridimensionales interactivos a través de Internet. (24)

2.4.4 Tecnologías

2.4.4.1 Unity3d

Es una herramienta multiplataforma para el desarrollo de juegos tanto en 3D como 2D. Una buena opción para cualquier tipo de desarrollo, desde el más simple gadget⁴³ para un micro sitio pequeño, hasta el más complejo juego MMORPG. Está basado en JavaScript y se debe descargar el Web Player plugin (3.2 Mb) para poder ver el contenido en cualquier browser. Permite hacer juegos multijugador y mediante algunas licencias no libres se pueden hacer aplicaciones para Iphone y para Wii. Inicialmente fue desarrollada para Mac, pero se hace un tiempo se ha adentrado en el mundo de Windows. Luego de obtener los modelos 3D (No gratis), se exportan de uno de los programas de manejo 3D (3ds máx, maya, blender, cinema3d) a formato FBX, no posee un buen manejo con las texturas. (19)

🚦 Ventajas:

- ✓ Tiene un buen rendimiento independientemente de la maquina o de la cantidad de cosas que se tengan abiertas.
- ✓ Se puede programar en C#, JavaScript, AJAX, .net y se puede comunicar mediante sockets al flash player por lo que se puede unir con tecnologías adobe como MXML, y as3.
- ✓ Puede utilizar el editor, que trae Unity, aunque permite usar cualquier otro editor para JavaScript, llámese dreamweaver, eclipse, etc., ya que estas librerías se pueden integrar a un proyecto JavaScript.

⁴³ **Gadget:** dispositivo que tiene un propósito y una función específica, con un diseño más ingenioso que el de la tecnología corriente.

- ✓ Cuenta con un editor exclusivo para 3D con el cual se puede ver y testear sin necesidad de compilar, esto es una gran ventaja, ofrece la posibilidad de ver cómo se comporta en entorno 3D y los modelos en este modo.
- ✓ Cuenta con una gran variedad de modelos de sombras y texturas para terrenos. El que más luce es el *cartoon shader*.
- ✓ Cuenta con un *Physic Engine*⁴⁴ que facilita el comportamiento de los modelos en el cliente.
- ✓ Hay muchos tutoriales, ejemplos en la Web.

✚ Desventajas:

- ✓ Presenta algunos *crash*⁴⁵, aunque no son muy comunes.
- ✓ Se dice que la programación es orientada a objetos, aunque al codificar no luce del todo así, se parece más a la programación tipo ActionScript 2.
- ✓ La documentación de las clases no es muy clara y muchas veces hay que perder mucho tiempo experimentando para entender el comportamiento de una clase en particular.
- ✓ Se puede bajar la versión Trial para desarrollar, lo demás es bajo licencia, aunque los costos son accesibles.

2.4.4.2 O3D

O3D es un API Web de software libre bajo la licencia BSD, fue creado por Google para crear aplicaciones en 3D completas e interactivas con el navegador sin la necesidad de flash, funciona en múltiples sistemas Operativos y navegadores como Firefox, Safari, Google Chrome, e Internet Explorer. Se ejecuta como un plugin de navegador y así los usuarios no tienen que superar los obstáculos de descargar y ejecutar código de aplicación independiente en sus sistemas. Está diseñado para ser incorporado al navegador, sistema operativo, aplicaciones y plataforma de aplicaciones. La API de O3D maximiza el rendimiento de la GPU para la programación del lenguaje de sombreado de forma directa, una ventaja sobre el software puro de visualización. (20)

Google O3D funciona a tres niveles (20)

⁴⁴ **Physic Engine:** motor para la representación de modelos físicos en 3D.

⁴⁵ **Crash:** significa Choque, estrellarse

- ✓ Sistema: proporciona un plugin para el navegador que añade capacidades gráficas dentro del motor de representación.
- ✓ Contenido: aparte de trabajar con texto, imágenes o vídeo, O3D proporciona nuevas herramientas para crear contenido en 3 dimensiones. Existe una guía para el desarrollador donde se puede encontrar los conceptos que se deben conocer para programar con O3D.
- ✓ Código: O3D extiende el código disponible para los programadores en JavaScript con un API para gráficos 3D, que usa el sistema de eventos y procedimientos típico de JavaScript.

Al igual que Chrome⁴⁶, O3D usa GYP⁴⁷ para generar ficheros de construcción para cada una de las plataformas. En Windows produce el fichero. sln y archivos. Vcproj, en Mac OS X produce. Paquetes xcodeproj⁴⁸, y en Linux que produce archivos SConscript.

En Windows, GYP produce los archivos de solución que se van a cargar en Microsoft Visual Studio. La solución principal se encuentra dentro del archivo que se llama "o3d.sln", y aparecerá en el directorio de construcción. Cargar este archivo en Visual Studio, y construir como lo haría con cualquier otro archivo de Visual Studio. El resultado aparecerá en build\directorio de versión, en función de la construcción que hace.

O3D tiene dos representadores diferentes en Windows, un D3D⁴⁹ basado en procesador, y un procesador basado en OpenGL. El cambio entre representadores requiere una reconstrucción, ya que utilizan los directorios de la misma salida que en la actualidad. (21)

El tamaño del plugin a la hora de descargarlo es aproximadamente: (20)

- ✓ Windows y Mac: en función de si se ha ejecutado antes de las aplicaciones 3D en el sistema, entre 1.4MB y 1.7MB (en Windows) y alrededor de 5 MB en el Mac
- ✓ Linux: se debe descargar y compilar el código fuente.

⁴⁶ **Chrome:** navegador Web compilado por componentes de código abierto desarrollado por Google.

⁴⁷ **GYP:** acrónimo de *Generate Your Projects*.

⁴⁸ **Xcodeproj:** extensión de archivo asociado a los programas de **xcode:** entorno de desarrollo integrado de Apple Inc

⁴⁹ **D3D:** acrónimo de Direct3D API creada para el gráfico y programación de gráficos en 3 dimensiones.

O3D no funciona en VMware o Ubuntu sin embargo, en el sistema operativo Mac si funciona, siempre y cuando se habilita la aceleración DirectX⁵⁰.

Es fundamentalmente un sistema basado en sombra, por lo que tiene definido un solo tipo de luz, si se desean más luces en la muestra se deberá modificar el código de O3D que está relacionado con el efecto de la luz en el modelo. Se decidió no utilizar el lenguaje de sombreado de OpenGL (GLSL) por algunas razones, como por ejemplo los controladores de OpenGL no se ajustan a GLSL y a la hora de compilar pueden quedar inconsistencias en el sombreado, algunos gráficos de hardware tampoco son compatibles con GLSL. Este API no tiene la habilidad de devolver un mensaje de error al mismo tiempo que detiene el JavaScript. (21)

2.4.4.3 WebGL

Es un motor de representación que permite crear gráficos 3D en un navegador Web, está basado en la versión 2.0 de OpenGL ES. Contiene plugin-3D libre que son implementados en el navegador. Utiliza el lenguaje de sombreado de OpenGL (GLSL), y permite una familiarización con la norma API del OpenGL. Está integrado directamente en el navegador, y sus aplicaciones pueden utilizar la infraestructura de JavaScript y el DOM, especialmente para documentos HTML. WebGL es manejado por el consorcio de tecnología Khronos Group, e integra Canvas, debido a esto puede combinar elementos HTML5 y otros contenidos Web en la representación 3D. El uso del DOM le posee muchas ventajas entre las que se encuentra: (22)

Carga de imágenes. Existen numerosas formas de cargar imágenes que han sido desarrolladas por OpenGL, las aplicaciones WebGL utilizan la carga de imágenes directamente del navegador. Contiene una función que es la encargada de cargar una textura para formar el objeto en 3D y luego representarlo en la Web.

Gestión de eventos. WebGL utiliza el mecanismo de manejo del estándar del navegador. El código JavaScript programa de flujo de WebGL está completamente integrado con el navegador de manejo de eventos del sistema. En estos casos el elemento controlador de cámara utiliza el navegador estándar de

⁵⁰ **DirectX:** colección de API para facilitar las tareas de la multimedia, programación de juegos entre otras.

eventos que se ejecutan en cualquier navegador Web, en cualquier plataforma, se utilizan valores de las coordenadas de los eventos para actualizar las coordenadas actuales de la cámara para representar la escena.

Perfecta composición de contenido Web. WebGL utiliza el estándar Canvas elemento, que se integran automáticamente con los otros elementos en la página Web.

La gestión automática de memoria. En OpenGL, la memoria está destinada de forma explícita. En WebGL, la gestión de memoria es manejada automáticamente.

Las aplicaciones de WebGL son muchas, como por ejemplo implementa juegos en 3D que funcionan directamente en el navegador y así permite un buen rendimiento, especialmente si se implementa con aceleración del 3D por hardware. Es compatible con varios navegadores entre los que esta firefox, safari, cromo entre otros. Los principales proveedores de navegadores de Google (Chrome), Opera (Opera), Mozilla (Firefox) y Apple (Safari) son miembros de WebGL, junto con otros desarrolladores de gráficos 3D. Requiere de un soporte preliminar disponible en Linux, Mac OS X y Windows.

Debido a que está basado en OpenGL y se integra varios navegadores populares, WebGL ofrece una serie de ventajas, entre ellas: (22)

- ✓ Una API que se basa en un conocido y ampliamente aceptado estándar de gráficos 3D.
- ✓ Cross-navegador y multi-plataforma de compatibilidad.
- ✓ La estrecha integración con contenido HTML, incluidos los mecanismos de composición de capas, la interacción con otros elementos HTML, y el uso de la manipulación de eventos estándar HTML.
- ✓ Aceleración gráfica en 3D a nivel de hardware para entornos o ambientes de navegadores Web.
- ✓ Un entorno de programación que facilita el prototipo de gráficos 3D, no es necesario para compilar y enlazar antes de poder ver y depurar los gráficos prestados.

Algunas bibliotecas en desarrollo que se están incorporando WebGL incluyen el C3DL y el WebGLU.

C3DL

C3DL es una librería JavaScript que hace que sea más fácil escribir aplicaciones 3D usando WebGL. Proporciona un sistema de matemáticas, escena, y clases de objetos 3D que hace WebGL más accesibles para los desarrolladores que quieren desarrollar el contenido 3D en el navegador. Fue originalmente parte de la red de investigación CATGames y fue desarrollado con lienzo 3D (el precursor de WebGL). Permite cargar geometría, transformaciones, materiales y texturas, aunque hasta el momento no soporta animación. (23)

Es compatible con Chrome, Firefox, Safari y permite hacer animaciones collada que es un formato XML abierto que es compatible con muchas de las herramientas 3D que funcionan como plugin. WebGLU es un motor de alto nivel para desarrollar aplicaciones basadas en WebGL, que se centra en la solución final con poco código.

Algunas de las características de WebGLU son:

- ✓ Diseñado con el rendimiento en mente. Las críticas se toman muy en serio.
- ✓ Para sombras y programas de sombreado. Se puede cargar directamente desde .frag / .vert / .vp / archivos .fp (Y automáticamente asignar el tipo de sombreado).
- ✓ Crear en anexo - cargas y establecer en marcha el estilo de sombra dentro del programa.
- ✓ Configura automáticamente cualquier proyección para el *rendering*.
- ✓ Proporciona proyección y *model view*⁵¹, que aproxima el comportamiento de OpenGL al modo de matriz.
- ✓ Automáticamente crea matriz / buffer elemento según sea necesario y los une, según sea necesario durante la representación.
- ✓ A cada objeto se puede asignar un programa de sombreado diferente
- ✓ Es compatible con las jerarquías de objetos.
- ✓ Texturas de la imagen estática y vídeo.
- ✓ Las animaciones de procedimiento y fotogramas clave son tan fáciles como imperiosas.
- ✓ Se pueden mezclar los tipos de animación dentro de una jerarquía de objetos.

⁵¹ **model view:** Modelo de vista, que define conjunto de puntos de vista para el análisis de los sistemas.

2.5. Estándares para la representación en 3D

2.5.1 VRML

Es un formato de fichero para describir objetos 3D interactivos con efectos de iluminación, texturas, detección de colisiones, etc. Diseñado especialmente para ser usado a través de Internet o un cliente local. Se creó con el objetivo de realizar escenas y modelos vectoriales en 3D con las que el usuario pudiera tener cierta interacción. El estándar VRML fue creado y desarrollado en 1994 por el consorcio VRML y fue la primera tecnología reconocida oficialmente por la ISO⁵² como estándar para la creación, distribución y representación de elementos 3D a través de Internet. La filosofía de VRML es muy similar a la de HTML⁵³. Al ser un estándar abierto no está limitado a ninguna aplicación o plataforma, y existen varios visores gratuitos. Al utilizar un formato de texto plano, no es necesaria ninguna herramienta para crear un mundo VRML. No se limita a la descripción de escenas estáticas, sino que permite un dinamismo en las escenas proporcionándole al observador la interacción con los objetos de la misma. Permite abrir ficheros de formato WRL, WRZ, VRML; solo que para utilizarlo se requiere de un módulo en el navegador.

Fue diseñado para cumplir con los siguientes principios básicos. (24)

- ✓ Dar la posibilidad del desarrollo de programas para crear, editar y mantener archivos VRML, así como la creación de programas para la importación y exportación del formato VRML a otros formatos gráficos tridimensionales.
- ✓ Permitir la opción de utilizar y combinar objetos dinámicos tridimensionales dentro de un mismo mundo VRML.
- ✓ Incorporar la capacidad de crear nuevos tipos de objetos no definidos.
- ✓ Resaltar la importancia del funcionamiento interactivo en una amplia variedad de plataformas existentes.
- ✓ Permitir la creación de mundos tridimensionales de cualquier tamaño.

2.5.2 JAVA3D

⁵² ISO: acrónimo de *International Organization for Standardization* o estándar de la organización internacional.

⁵³ HTML: acrónimo de *HyperText Markup Language*

El API Java 3D es un interface para escribir programas que muestran e interactúan con gráficos tridimensionales. Proporciona un conjunto de clases para crear aplicaciones y applets⁵⁴ con elementos 3D, además, de las funciones para la creación de imágenes, visualizaciones, animaciones y programas de aplicaciones gráficas 3D interactivas. Le permite a los desarrolladores la posibilidad de manipular geometrías complejas en tres dimensiones. La principal ventaja que presenta este API 3D frente a otros entornos de programación 3D es que permite crear aplicaciones gráficas 3D independientes del tipo de sistema. Es parte de la API JavaMedia y por tanto puede hacer uso de la versatilidad del lenguaje Java, así como soportar un gran número de formatos como VRML, CAD entre otros. (24)

El conjunto de clases, interfaces y librerías de alto nivel que posee permiten aprovechar la aceleración gráfica por hardware que incorporan muchas tarjetas gráficas, las llamadas a los métodos de Java 3D son transformadas en llamadas a funciones de OpenGL o Direct3D. Aunque no necesariamente soporta todas las necesidades 3D, proporciona la capacidad de implementarlo a través de código Java y en otros casos requiere de cargadores (VRML, X3D) que traducen ficheros de ese formato en objetos apropiados en Java3D. Proporciona una interfaz de programación de alto nivel basado en el modelo orientado a objetos, por lo que permite un desarrollo de aplicaciones rápido y simple. (24)

2.5.3 X3D

X3D es un estándar abierto XML bajo la norma ISO, un formato de archivo 3D que permite la creación y transmisión de datos 3D entre distintas aplicaciones, especialmente en red. Sus características son: (24)

- X3D está integrado en XML: esto representa un paso fundamental a la hora de conseguir una correcta integración en:
 - Servicios Web.
 - Redes Distribuidas.
 - Sistemas multiplataforma y transferencia de archivos y datos entre aplicaciones.
- X3D es Modular (tiene componentes): esto permite la creación de un núcleo 3D más ligero ajustado a las necesidades de los desarrolladores.
- X3D es Extensible: permite añadir componentes para ampliar las funcionalidades según las necesidades del mercado.

⁵⁴ **Applets:** componente de una *aplicación* que se ejecuta en el contexto de otro programa, en estos casos un navegador Web.

- X3D es Perfilado: se pueden escoger distintos grupos de extensiones apropiadas según las necesidades específicas de la aplicación.
- X3D es Compatible con VRML: se mantiene el desarrollo, el contenido y la base de VRML97.

En vez de mantener una especificación amplia y estática como VRML, tiene una arquitectura basada en componentes que da soporte para la creación de diferentes perfiles, con esto se pretende que el desarrollo en un área no afecte ni retrase la evolución en su conjunto. (24)

2.5.4 Comparación entre VRML/X3D y Java3D

Una de las principales diferencias entre VRML/X3D y Java3D, es que Java3D se define como un lenguaje de programación de escenarios 3D a bajo nivel. La creación de objetos y elementos tridimensionales en Java3D requiere no sólo la formación de los elementos 3D, sino también la definición de todos los aspectos relacionados con la visualización y control de las capacidades del escenario. Para la creación del escenario más simple, el código en Java3D es superior al necesario en VRML/X3D, pero por otro lado el control de los distintos elementos presentes en el sistema es superior y más natural en Java3D. Otro aspecto que se destaca es la pérdida de velocidad y prestaciones en el caso de Java3D frente a otros visores de VRML/X3D desarrollados en C/C++ y empleando directamente Direct3D u OpenGL. También es posible emplear Java3D como visor de archivos VRML/X3D. Para ello sólo es necesario utilizar alguno de los cargadores⁵⁵ de VRML/X3D desarrollados para Java3D.

Las ventajas principales son la capacidad de ejecución en distintas plataformas y evitar la necesidad de instalar un plugin específico para el navegador. X3D ofrece soporte para múltiples codificaciones de archivos: VRML97, XML. X3D emplea una arquitectura modular para dar una mayor extensibilidad y flexibilidad. La gran mayoría de las aplicaciones no necesitan de todo el poder de X3D, como tampoco el soporte para todas las plataformas y funcionalidades definidas en la especificación. A diferencia de VRML97, el cual requiere tener un soporte total por parte de la implementación, X3D permite tener un soporte particular para cada necesidad. El mecanismo de componentes X3D también permite a las empresas implementar sus propias extensiones de acuerdo con un riguroso grupo de reglas. Además, X3D permite una mayor flexibilidad del ciclo de vida del estándar, ajustándose a la evolución del mismo.

⁵⁵ **Cargadores:** se le suele llamar también *loaders*.

Entre los formatos más utilizados para la representación en 3D se encuentran WRL, WRZ, OBJ, SCN, DXF, VRML, DWG, 3Ds.

WRL: Es un lenguaje de descripción de escenarios virtuales para navegadores, nacido de la necesidad de presentar modelos 3D en la Web, se deriva del componente WSML⁵⁶.

2.6 Conclusiones

Hasta el momento en el marco del capítulo se abordaron temas como por ejemplo los diferentes métodos matemáticos que se pueden llevar a cabo para la representación de MDT a través de polinomios de interpolación, así como los diferentes métodos de ajuste que son necesarios aplicar para obtener una mejor representación del polinomio y que los valores sean los más cercanos posibles a la curva. Se mencionaron los diferentes algoritmos y métodos de cómo visualizar los MDT, que requiere de datos que son obtenidos a través de distintos medios como por ejemplo Topografía, Fotogrametría, o Cartografía del lugar a visualizar, las diferentes tecnologías y herramientas que facilitan la representación 3D en la Web y por último los estándares para la representación.

⁵⁶ **WSML**: acrónimo *Web Service Modeling Language*.

CAPÍTULO 3: Propuesta de la solución

3.1 Introducción

En el presente capítulo se describe la propuesta de la solución para aplicar a la plataforma GeneSIG, haciendo una breve explicación del por qué no de cada uno de los métodos, herramientas, algoritmos y estándares que no fueron seleccionados para formar parte de la propuesta.

3.2 Justificación de la solución propuesta

La construcción de los MDT requiere la generación de grandes volúmenes de datos para la representación de los mismos. Para manejarlos se han creado algoritmos y modelos matemáticos que permiten la representación interactiva de la superficie, ajustado al campo de visión del espectador; reducir el procesamiento y acceso a los datos; obtener el mejor desempeño entre la GPU y CPU, así como conservar las características del terreno. Los modelos matemáticos estudiados en la investigación utilizan los métodos de interpolación a través de una muestra de valores para elaborar los modelos, estos métodos al igual que los de ajuste son empleados para representar curvas de nivel. Éstos deben de ser empleados con técnicas de solución adecuada que requiere de un gran número de análisis por lo cual se vuelven costosos en cuanto al uso de recursos de la PC y difíciles de desarrollar.

Debido a la necesidad del proyecto GeneSIG de encontrar una solución que le permita representar MDT en 3D sobre la Web y teniendo en cuenta la investigación desarrollada, se decidió emplear para la representación una tecnología y un algoritmo que permitan eficientemente el almacenamiento y visualización de los modelos. Con esta propuesta se minimizará el costo por concepto de tiempo de desarrollo, teniendo en cuenta que el proyecto cuenta con un colectivo de trabajo reducido, donde cada uno tiene asignado tareas dentro del desarrollo y producción. Además, en el mismo se tienen muchos aplicativos que realizar, y no se cuenta con el personal suficiente para llevar a cabo el desarrollo de una técnica o herramienta que facilite esta representación.

El algoritmo será el encargado de llevar a cabo la modelación de los MDT con un mínimo de memoria y la tecnología permitirá representarlos en 3D sobre la Web. En resumen, el algoritmo almacenará el modelo que será extraído por la tecnología para aplicarle textura y luego representarlo en la Web.

3.3 Algoritmos

- ✚ Quadtree: permite extraer de forma efectiva la triangulación de múltiples resoluciones que se visualizan; sin embargo, posee el problema de ocasionar el efecto cascada debido a que cuando se selecciona el cuadrante a triangular puede darse el caso de seleccionar uno que no influya en la visualización, provocando lentitud en el mismo.
- ✚ Mallas Regulares R-Trianguladas: es una técnica de representación de terrenos para aplicaciones, realiza el proceso de triangulación solo que en vez de realizarlo por cuadrantes, divide la malla en triángulos. Entre sus ventajas se puede destacar que a diferencia del Quadtree evita los huecos debido a que entre regiones adyacentes no existe diferencia de resolución, la representación se realiza en forma de árbol binario dando lugar a un algoritmo muy eficiente de búsqueda de vecinos. Su principal desventaja consiste en provocar un efecto cascada donde el proceso de división del vecino por la hipotenusa genera gran cantidad de triángulos, por lo tanto, el número de polígonos procesados por el sistema es mayor, minimizando la rapidez del proceso de visualización.
- ✚ Mallas Adaptables Óptimas en Tiempo Real: Es un algoritmo para construir mallas triangulares que minimiza las métricas de error dependientes de la vista. Usa dos colas de prioridad para el proceso de división y fusión, por lo que su almacenamiento requiere de un mínimo de memoria. Sin embargo, posee gran cantidad de pasos en el análisis del algoritmo, aumentando el tiempo de ejecución debido a que continuamente debe de actualizar el área que se está analizando. Además de comprobar las relaciones entre vecinos para llevar a cabo el proceso de división y fusión de triángulos.
- ✚ Geomipmapping: Es un algoritmo para gestionar el nivel de detalle del terreno. Cada nivel de detalle es representado como los vértices de un bloque del terreno, estos son almacenados en un arreglo. Este algoritmo requiere un gran análisis ya que debe recorrer el arreglo para comprobar los bloques adyacentes y evitar que se generen huecos entre bloques de diferentes niveles. Este proceso es repetido entre cada par de bloques lindantes tantas veces como sea movida la cámara.

Luego de haber analizado las desventajas de los algoritmos antes mencionados se decidió utilizar el **Hilbert-Quadtree**.

Como parte de una investigación realizada por el Máster Yusnier Valle Martínez sobre Modelación y visualización de superficies de terrenos en tres dimensiones para obtener una propuesta de un algoritmo

que permitiese la modelación y visualización de la superficie terrestre; fue probada y publicada como un algoritmo eficiente Hilbert-Quadtree, el cual permite cargar de forma eficiente la información espacial del modelo, así como la navegación de forma eficaz sobre superficies modeladas. En este algoritmo la búsqueda de vecinos de igual tamaño de una región, con respecto a otros algoritmos constituye una optimización importante en cuanto a estructuras de datos espaciales. Requiere de un mínimo de cantidad de memoria para su almacenamiento, facilitando el proceso de visualización; a diferencia de Quadtree que necesita de apuntadores para los nodos vecinos, aumentando así el espacio en memoria.

Hilbert-Quadtree: Este algoritmo es la unión de la curva de Hilbert con la estructura de dato Quadtree. Con respecto a la triangulación de las superficies posee patrones que le facilita el manejo de la estructura, por lo cual no requiere de gran cantidad de memoria debido a que utiliza dos arreglos donde almacena los datos de cada cuadrante en cada nivel, facilitando el almacenamiento del modelo, ya que no requiere información sobre los nodos vecinos como otros de los algoritmos estudiados. No es necesario realizar análisis sobre la diferencia entre niveles, y le atribuye la ventaja de no contar con inconsistencia en la representación. En su proceso de triangulación aplica el criterio de la representación en forma de árbol binario donde los nodos representan los patrones reduciendo el número de polígonos que deben de ser enviados a la memoria.

La triangulación de las regiones debe de realizarse de forma eficiente sobre superficies de múltiple resolución ya que esto influye directamente en la calidad y rapidez de la visualización. Los algoritmos estudiados de una forma u otra poseen desventaja en cuanto a este nuevo algoritmo que no requiere de gran exactitud en las superficies modeladas, al contrario del Geomipmapig que requiere de niveles de detalle para su representación. Además, que este fue creado con la intención de reducir la información que debía de ser enviada a memoria en comparación con el Quadtree y el de Mallas Regulares R-Trianguladas quienes pueden provocar una sobrecarga en la memoria si se produce el efecto cascada. Con respecto al ROAM posee la ventaja de ejecutarse en menor tiempo ya que está almacenado en dos arreglos donde se guarda toda la información y el mismo no requiere de un análisis entre vecinos.

3.4 Herramientas y tecnologías

Librerías de JavaScript

Permiten la representación de aplicaciones en 3D. JavaScript es un lenguaje interpretado que está diseñado para ser ejecutado por medio de un intérprete, que también se le conoce como lenguaje de Script. El intérprete no es más que aquella aplicación que es capaz de analizar y ejecutar programas escritos en un lenguaje de alto nivel. Los programas interpretados suelen ser más lentos que los programas compilados, debido a la necesidad de traducir el código mientras se están ejecutando, son menos eficientes en cuando al tiempo y ejecución, dado que cada instrucción del código debe ser interpretado antes de cada ejecución del programa, por lo que las aplicaciones de gran cantidad de información se pueden volver lentas y tediosas, así como disminuir el procesamiento.

Plugin para navegadores

Son pequeños programas de software unido al navegador que amplían las capacidades de un programa, donde son ejecutados por la aplicación principal por medio de las API. Son muy utilizados en las páginas Web donde permiten ampliar las funcionalidades para ver contenidos interactivos, videos, gráficos, entre otros. En los sistemas de gestión de contenidos permiten cambiar la apariencia de la página, incluir botones y otro tipo de contenido a las páginas Web. Sin embargo, ninguno de los plugin que existen actualmente, han sido creados por la compañía Mozilla y a la hora de integrarlos con el navegador puede que no se tenga el control total y estricto sobre las funcionalidades para realizar una representación en 3D, tampoco permite hacer consultas o análisis dentro del modelo y no todos los navegadores soportan la integración de plugin.

Adobe Flash

Permiten crear aplicaciones interactivas en la Web, tienen una serie de ventajas y desventajas que serán expuestas a continuación:

❖ **Ventajas**

- Adobe Flash está basado en imágenes vectoriales lo que permite que al ajustar sus tamaños éstas no pierdan calidad.
- Una vez que se conozca el software y con la práctica, se vuelve relativamente sencillo y rápido crear páginas Web.
- Permite lograr presentaciones excelentes, de altísima calidad, que cautivan a los visitantes y que llamen mucho la atención.

❖ Desventajas

- Puede haber problemas con algunos navegadores, pues si no se tiene instalado el la Web desaparece.
- El contenido de Flash no es indexado por los buscadores por lo que se vuelve difícil encontrar la Web en Google, Yahoo, Ask, etc.
- Si la animación no está bien programada puede resultar pesada cargarla. Se debe tener en cuenta hacia qué destinatarios se está apuntando y si los visitantes tendrán el suficiente ancho de banda para navegarla y que no demore mucho tiempo en los cambios de página.
- Mucha animación o sonido es molesto. Caso típico son aquellas que al cargarlas reproducen un MP3 y no tienen botón desde dónde silenciarlo. Otras en donde son tantas las animaciones que el visitante se marea. Del punto anterior y de este se desprende que si se cometen estos errores el visitante abandonará la Web rápidamente.
- Producir cambios en páginas realizadas con Flash puede significar largo tiempo de espera por parte del cliente porque en muchos casos es casi lo mismo que hacerla de nuevo, ya que requiere de análisis profundo.
- Para hacer una Web flash de manera fácil hay que estudiar ActionScript, un lenguaje de programación propio del sistema.
- Adobe Flash es un programa de pago y si bien los plugin o reproductores de este tipo de archivos son gratuitos, para crearlos se necesita comprar el software.
- El desarrollo Web en Flash requiere de muchas horas de trabajo.

Por lo expuesto antes y debido a que Flash no es un estándar abierto, se decidió no utilizar ninguna herramienta de la investigación relacionada con Adobe Flash, se requiere de proyectos de licencia libre, gratuito y de fácil manejo para el cliente.

Tecnologías

Unity3d es una herramienta que ofrece muy buenos resultados pero fue descartado como solución por no ser libre, y además, no posee un buen manejo con texturas.

Luego del estudio de las herramientas y tecnologías se tenían como propuesta O3D y WebGL debido a que son las tecnologías libres que actualmente se encuentran en la lucha por el mercado mundial de la

representación en 3D sobre la Web. Para definir cuál de los dos sería la mejor tecnología se hizo un análisis basado en las ventajas y desventajas que poseía uno con respecto al otro.

WebGL es un motor para crear aplicaciones con representación en 3D que contiene implementados plugin libres que están directamente en el navegador y permite habilitar gráficos 3D en la Web. Utiliza OpenGL obteniendo una familiarización con este, sus aplicaciones pueden utilizar JavaScript y el DOM lo cual le permite una serie de ventajas, entre ellas cargar imágenes de forma rápida y segura. Está implementado en la versión de desarrollo de los principales navegadores por lo que las implementaciones están disponibles en Mozilla Firefox, Google Chrome, Safari y Opera.

O3D es un API de licencia libre que fue creado por Google para crear aplicaciones en la Web con contenido 3D, posee ciertas desventajas, requiere de tarjetas gráficas específicas, como las Nvidia GeForce 8 o las ATI Radeon HD lo que traería consigo que se necesitara potenciar el equipamiento de desarrollo que se tiene en el proyecto en estos momentos, a la hora de realizar grandes proyectos se producen *crash* y no permite la integración de flash player. Solo tiene definido un solo tipo de iluminación, por lo que si se desea otro, se requiere cambiar todo el código relacionado con el modelo de sombra. Entre sus ventajas se puede decir que permite representar gráficos tridimensionales con la misma eficiencia que una aplicación de escritorio o una consola de videojuegos, esto gracias a que tiene acceso directo por medio de una API al GPU de la tarjeta de video. Entre los formatos mencionados anteriormente O3D puede utilizar 3Ds, dwg, DXF, VRML y OBJ.

De estas dos propuestas se puede decir que el objetivo de ambas es llevar los gráficos 3D a una mejor representación en la Web, y ambos utilizan JavaScript como base para generar estos gráficos. WebGL utiliza el lenguaje de sombreado de OpenGL por lo que permite el manejo de texturas del mismo, sin embargo, no se puede decir que O3D también pueda utilizarlo debido a que los controladores de este no son compatibles.

Al final se llegó a la conclusión de que la mejor propuesta sería **WebGL**. Entre las desventajas por las cuales fue descartado O3D se puede decir que la necesidad de una tarjeta de video diferente de las que cuenta el equipamiento del proyecto, traería consigo el cambio masivo en cada una de las maquinas,

donde cada desarrollador debe de hacer uso de la plataforma GeneSIG. No permite el manejo de texturas de OpenGL, y posee una sola fuente de iluminación, al intentar cambiarle esta fuente se tendría que cambiar todo el código, por lo tanto, se podrían producir errores a la hora del cambio.

WebGL permite utilizar y combinar elementos de HTML5 y la etiqueta Canvas, ofreciéndole la opción de una interactividad muy buena en lo que se refiere a los videojuegos, debido a que permite jugar en línea partidas entre varios jugadores. Es compatible con la mayoría de los navegadores y hace uso de OpenGL por lo que posee varias ventajas, está más enfocado en la representación gráfica ofreciendo gran aceleración en el proceso de la misma. Permite representar en las páginas Web gráficos 3D de forma eficiente y es una buena alternativa para cargar imágenes y llevarlos a diferentes formatos de ficheros.

Las especificaciones de esta tecnología aprovechan la evolución reciente de la tecnología Web que contiene el elemento lienzo definido como parte de la especificación HTML 5 y el aumento marcado en el rendimiento de JavaScript en todos los principales navegadores. Debido a que soporta OpenGL Es 2.0 el mismo puede mediante JavaScript acceder directamente a este para así optimizar aun más el proceso de representación. Entre el consorcio de desarrollo de WebGL se encuentran las compañías Nokia y Motorola, debido a esto en un futuro los dispositivos móviles podrán acceder a gráficos 3D en sus navegadores. El motor de representación Webkit, usado en Google Chrome y Safari, incorpora en su última versión a WebGL y por tanto será soportada en las próximas versiones de desarrollo de estos navegadores, una vez que actualicen al nuevo motor.

3.5 Conclusiones parciales

En este capítulo se hizo referencia a las herramientas, algoritmos, modelos matemáticos y estándares del capítulo anterior y se explicó porque algunas de estas no son una buena solución para el proyecto. Como respuesta final se decidió escoger una herramienta y un algoritmo que van a permitir la representación de los MDT facilitando la visualización de los datos espaciales dentro del proyecto GeneSIG del departamento Geoinformática del centro GEySED.

Conclusiones

En el transcurso de la investigación que se realizó con el objetivo de determinar cuáles serían los algoritmos, técnicas o herramientas que harían posible la modelación, visualización y representación de los MDT en 3D sobre la Web para aplicar a la plataforma GeneSIG, se encontraron un conjunto de técnicas que se utilizan para llevar a cabo esta representación. Dentro de las principales tecnologías que en estos momentos actuales luchan por liderar la Web en este sentido se encontraron O3D la cual es una tecnología que se está desarrollando en la actualidad por una de las compañías que más desarrollo tiene en los temas relacionados con representaciones de estos modelos, Google. Los mismos han puesto todos sus esfuerzos por lograr que esta sea el futuro de los modelos digitales en la web. Además de esto, se encontró WebGL que es otra de las tecnologías que actualmente al igual que O3D están en la lucha por ser el soporte de las aplicaciones en 3D en la Web. De ellas después de haber realizado un estudio minucioso de las ventajas y desventajas del mismo se llegó a la conclusión de que para las acciones que posteriormente a llevar a cabo para la representación de estos modelos en la plataforma, la propuesta sería WebGL debido a que es mucho más estable para la representación 3D que flash, posee una buena gestión de eventos con el navegador y una completa integración de contenido Web. Por otra parte, el algoritmo seleccionado para la propuesta permitirá cargar de forma eficiente los modelos que serán visualizados por WebGL.

Recomendaciones

Se recomienda:

- ✚ Implementación e Integración de la propuesta con la plataforma GeneSIG para lograr una representación tridimensional de los datos espaciales del terreno.
- ✚ Implementar e integrar métodos y algoritmos que se fusionen de forma eficiente, para obtener un resultado óptimo en cuanto a la representación y visualización de MDT en 3D en la Web.
- ✚ Continuar el estudio de la tecnología WebGL y O3D debido a que actualmente se encuentran en desarrollo para su perfeccionamiento.

Referencias Bibliográficas

- 1 Felicísimo, Angel Manuel. Simulación de procesos:modernización climática. 1999.
- 2 Babylon Ltd. Babylon Programa de Traducción. [Online] 2009. De <http://diccionario.babylon.com/3D>.
3. Berry, J.K. (1987):"Computer assisted map analysis: potential and pitfalls". Photogrammetric Engineering and Remote Sensing, 53(10): 1405-1410.
4. Besalú, Miquel J. Pavón i. DICCIONARIO DE TERMINOLOGÍA CARTOGRÁFICA. [Online] <http://www.hyparion.com/Web/diccionari/dics/cartografia.htm#mapa>
5. Felicísimo, Angel Manuel. 1999, p. 16.
6. Felicísimo, Angel Manuel. Simulación de procesos: cuencas hidrológicas y cuencas visuales. p. 16.
7. Jose Enrique P, Maria Joaquina,Priego de los Santos. *LA Triangulación de Delaunay aplicada a los Modelos Digitales del Terreno*. Valencia : Departamento de Ingeniería Cartográfica, Geodesia y Fotogrametría.
- 8.Daily Ibarreche Delgado, Manuel Macías Martínez. 2007. *Módulo de seguimiento de terrenos para sistemas de realidad virtual*. Universidad de las Ciencias Informáticas,Cuba : s.n., 2007.
- 9.Martínez, Lic.Yusnier Valle. 2009. *Modelación y visualización de superficies del terrenos en tres dimensiones*. Universidad de las Ciencias Informática,Ciudad de la Habana,Cuba : s.n., 2009.
- 10.Pomeraz, Alex A. 200. *Uso ROAM superficie trinagular*. California : s.n., 200.
- 11.Roberto Blanco Ancos, José Antonio García Gimeno,Gustavo de Santos García. 2004-2005. *Motor 3D con soporte de red aplicado a un videojuego*. Madrid : s.n., 2004-2005.
12. David E. Sigeti, Mark B. Mineev-Weinstein,Mark Duchaineau. *ROAMing terrain: real-time optimally adapting meshes*. Arizona : ISBN:1-58113-011-2, 1997.
13. Gutiérrez, Manuel. TuFunción. [En línea] 20 de abril de 2007. <http://www.tufuncion.com/>.
- 14.W3C. (2009, abril 30). *Gráficos Vectoriales Escalables*. Retrieved abril 2010
15. BIBLIOGRAPHY Consortium, W. W. (1998, mayo 13). (*VML*), *Vector Markup Language*. Retrieved abril 2010, de <http://www.w3.org/TR/NOTE-VML>.
16. Carlos Ulloa, Tim Knip Tim Knip. *BLOG.PAPERVISION3D.ORG*. [En línea] 2006. <http://blog.papervision3d.org/about/>.
17. RIAHispano. [Online] 2009. <http://www.riahispano.org/>.
18. AlternativaPlafom (2007-2010) <http://blog.alternativaplatform.com/en/>

19. Moraleda, Miquel. Unity3D & Actionsript examples. [En línea] 2009. <http://as3.miguelmoraleda.com/es/tag/unity3d/>.
20. Puente, Enrique. The O3D API Blog. [En línea] 2007.
21. Panzuela, Manuel Fernández. Manuel F. [En línea] 2006-2009. <http://mfernandez.es/wordpress/2009/07/23/Web-en-3d-o3d-api/>.
22. Grupo de Trabajo WebGL. Wiki pública WebGL. [En línea] diciembre de 2009. http://translate.googleusercontent.com/translate_c?hl=es&sl=en&u=http://www.khronos.org/webgl/wiki/Main_Page&prev=/search%3Fq%3DWebGI%26hl%3Des&rurl=translate.google.com.cu&usg=ALkJrhi3JrDxMKcbjsQGiHGWSnoc7oUJgw.
23. Leung, Catherine. 3DTest. [En línea] febrero de 2010.
24. Jiménez Macías, Emilio; Pérez de la Parte, Mercedes. Simulación con VRML, JAVA3D y X3D. España : s.n., 2006.
25. Aronoff, S. (1989). Geographical Information Systems: A management perspective. Ottawa, WDL Pub.
26. Sarasola, Irene .Simulador de pérdidas de programación basado en modelos digitales de elevaciones.<http://www.galeon.com/proplab/>
27. Gutiérrez, Javier J. ¿Qué es un framework Web?. http://www.lsi.us.es/~javierj/investigacion_ficheros/Framework.pdf
28. Mellado, José Espinoza. Topografía Docente. [Online] julio 2009. <http://topografiadocente.over-blog.es/article-34307781.html>.
29. Días., Ing. Juan Javier Durán. Modelo digitales de elevación para el continuo de elevaciones mexicanas.
30. Miliarium.com.2001-2004. http://www.miliarium.com/Proyectos/RestauracionAmbiental/RestauracionRiberas/Hidrologia/cuenca_fluvial

Bibliografía

1. ARANGO, A. P. (2002). Patente nº 1729 DE 2002. De Bogotá.com/plantilla-ante.asp?id_articulo=1051
2. Besalú, M. J. (s.f.). DICCIONARIO DE TERMINOLOGÍA CARTOGRÁFICA. Obtenido de <http://www.hyparion.com/Web/diccionari/dics/cartografia.htm#mapa>
3. Babylon Ltd. (2009). Babylon Programa de Traducción. Obtenido de <http://diccionario.babylon.com/3D>
4. Berry, J.K. (1987): "Computer assisted map analysis: potential and pitfalls". Photogrammetric Engineering and Remote Sensing, 53(10): 1405-1410.
5. Cayuela, L. (marzo de 2009). Actividades de investigación y docencia. Obtenido de <http://luiscayuela.blogspot.com/2009/03/bienvenido-grass.html>
6. Díaz, I. J. (s.f.). Modelos Digitales de Elevación para el continuo de elevaciones Mexicano.
7. Felicísimo, (1999) A. M. Modelos Digitales del Terreno
8. Felicísimo, (1999) A. M. (1999). La utilización de los MDT en los estudios del medio físico
9. Felicísimo, (1999) A. M. Simulación de procesos: cuencas hidrológicas y cuencas visuales.
10. Felicísimo, (1999) A. M. Simulación de procesos: modernización climática.
11. INEGI (Instituto Nacional de estadística y geografía). (s.f.). Recuperado el 2009, de <http://mapserver.inegi.org.mx/DescargaMDEWeb/informacion.cem?id=sirve>
12. Joly, F. (1988): La cartografía. Oikos-Tau. Barcelona.
13. La Torre, R. F., & Roffe, T. G. (mayo de 2004). MAPPING (Revista Internacional de Ciencias de la Tierra). Obtenido de MODELO DIGITAL DE ELEVACIÓN DE LA ZONA EMERGIDA DEL ECOSISTEMA SABANA CAMAGÜEY, CUBA: Obtenido de http://www.mappinginteractivo.com/plantilla-ante.asp?id_articulo=509.
14. M. Gallego Priego, R. B. Escenario 3D Proyecto SIGN II: una solución para la representación 3D de los datos territoriales bidimensionales
15. Ortiz, G. (2009). GabrielOrtiz.com. Obtenido de <http://www.gabrielortiz.com/>
16. Parimbelli, M. H. (2005). Crear visualizaciones y animaciones 3D con 3dem. Cátedra Técnicas Espaciales de Análisis.
17. Rodríguez, D. V. (2007). Algoritmos para la Modelación Digital del Terreno en un Sistema de Información Geográfico. Ciudad de La Habana Cuba.

18. Santos, M. Á. MODELOS DIGITALES DE ELEVACIÓN PARA TODOS LOS PÚBLICOS. Madrid: Universidad complutense de madrid.
19. Silva, D. L. (s.f.). Aplicación de Sistemas de Información geográfica en Cuba. Recuperado el Noviembre de 2005, de http://www.mappinginteractivo.com/plantilla-ante.asp?id_articulo=1051
20. Zubrinic, F. Modelos Digitales del Terreno(MDT). Facultad de Ciencias Forestales-UNSE.
21. Burmister, Adán. Flog Web super-hero training. 2007.
22. librosWeb.es <http://www.librosweb.es/javascript/capitulo1.html>