



**ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO**  
**FACULTAD DE CIENCIAS**  
**CARRERA MATEMÁTICA**

**MODELOS MATEMÁTICOS DE COMPUTACIÓN GRÁFICA PARA  
SIMULAR TEXTURAS DE SUPERFICIES REALES DE MADERA,  
ROCAS Y AGUA**

**Trabajo de Integración Curricular**

Tipo: Proyecto de Investigación

Presentado para optar al grado académico de:

**MATEMÁTICO**

**AUTOR:** JOSÉ ANDRÉS CIFUENTES FLORES  
**DIRECTOR:** DR. ALONSO WASHINGTON ÁLVAREZ OLIVO

Riobamba – Ecuador

2024

**©2024, José Andrés Cifuentes Flores**

Se autoriza la reproducción total o parcial, con fines académicos, por cualquier medio o procedimiento, incluyendo la cita bibliográfica del documento, siempre y cuando se reconozca el Derecho de Autor.

Yo, José Andrés Cifuentes Flores, declaro que el presente Trabajo de Integración Curricular es de mi autoría y los resultados del mismo son auténticos. Los textos en el documento que provienen de otras fuentes están debidamente citados y referenciados.

Como autor asumo la responsabilidad legal y académica de los contenidos de este Trabajo de Integración Curricular; El patrimonio intelectual pertenece a la Escuela Superior Politécnica de Chimborazo.

Riobamba, 13 de mayo de 2024

A handwritten signature in blue ink, appearing to read 'J. Cifuentes', with a red dot above the 'i'.

**José Andrés Cifuentes Flores**

**0605653112**

**ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO**  
**FACULTAD DE CIENCIAS**  
**CARRERA MATEMÁTICA**

El Tribunal del Trabajo de Integración Curricular certifica que: el Trabajo de Integración Curricular; Tipo: Proyecto de Investigación. **MODELOS MATEMÁTICOS DE COMPUTACIÓN GRÁFICA PARA SIMULAR TEXTURAS DE SUPERFICIES REALES DE MADERA, ROCAS Y AGUA**, realizado por el señor: **JOSÉ ANDRÉS CIFUENTES FLORES**, ha sido minuciosamente revisado por los Miembros del Tribunal del Trabajo de Integración Curricular, el mismo que cumple con los requisitos científicos, técnicos, legales, en tal virtud el Tribunal Autoriza su presentación.

	<b>FIRMA</b>	<b>FECHA</b>
Mtr. Alex Eduardo Pozo Valdiviezo <b>PRESIDENTE DEL TRIBUNAL</b>		2024-05-13
Dr. Alonso Washington Álvarez Olivo <b>DIRECTOR DEL TRABAJO DE INTEGRACIÓN CURRICULAR</b>		2024-05-13
Dr. Luis Marcelo Cortez Bonilla <b>ASESOR DEL TRABAJO DE INTEGRACIÓN CURRICULAR</b>		2024-05-13

## DEDICATORIA

El presente trabajo de investigación está dedicado a mi padres, José Cifuentes y Ligia Flores, quienes con su amor incondicional, su comprensión, sus consejos y su sacrificio infinito han sido mi mayor inspiración y sostén durante esta travesía, ustedes son mis pilares, me enseñaron a ser la persona que soy, mis valores, mis principios, mi perseverancia y el coraje para conseguir mis objetivos, lo aprendí de ustedes; por ello este logro es también suyo, ya que cada página de este documento lleva impresa la huella de su amor y dedicación.

A mis hermanos, quienes han sido mis cómplices en los momentos alegres y tristes, gracias por ser mis compañeros de aventuras, y por estar a mi lado sin importar las circunstancias, son un regalo invaluable que lo atesoro en mi corazón.

A mis compañeros de la carrera y amigos, quienes sin esperar nada a cambio han compartido sus conocimientos, sus momentos felices y tristes, y han hecho de este camino académico un emocionante viaje, gracias por alentarme y por celebrar conmigo cada logro que he alcanzado.

A mi familia, quienes me alentaron a perseguir mis sueños, gracias por estar siempre apoyándome en las buenas y en las malas, por aconsejarme y enseñarme que toda experiencia vivida me deja una valiosa lección, todo lo que he conseguido es gracias a su apoyo inquebrantable.

Con todo mi amor y admiración,

José

## AGRADECIMIENTO

Quisiera expresar mi más profundo agradecimiento a todas las personas que contribuyeron de alguna manera al desarrollo y culminación de este trabajo de investigación. En primer lugar, quiero agradecer a mi tutor, al Doctor Washington Alonso Álvarez Olivo, por su experta orientación, su paciencia inagotable, por su constante motivación y su compromiso inquebrantable. Su sabiduría, consejos y sus comentarios perspicaces, fueron fundamentales para la realización de este trabajo y para mi crecimiento académico y personal.

Quiero agradecer a todos los docentes que tuve durante la carrera, pues han dejado una huella imborrable en mi formación académica, les quiero agradecer por transmitirme los conocimientos, su pasión y compromiso con el desarrollo de los estudiantes, han sido una inspiración constante para mí a lo largo de este maravilloso viaje. Gracias por desafiarme a pensar de manera crítica, por alentarme a explorar nuevas ideas y por ayudarme a alcanzar mi máximo potencial.

Quiero expresar mi enorme gratitud a la Escuela Superior Politécnica de Chimborazo, por proporcionarme los recursos y el ambiente para poder llevar a cabo no solo el proceso de investigación de este trabajo, sino también por brindarme los conocimientos que he adquirido en esta prestigiosa institución. Agradezco el compromiso con la excelencia académica y el apoyo brindado a lo largo de este proceso.

Gracias infinitas a mis padres y a mi familia, quienes estuvieron a mi lado brindándome aliento, comprensión y motivación en todo momento. Su fe en mí, incluso en los momentos más difíciles, ha sido el pilar de este logro. Su amor y sacrificio han sido la luz que me guió en mi camino a través de este viaje académico. A todos ustedes, mi más sincero agradecimiento.

Finalmente agradezco a todos mis amigos y compañeros que estuvieron conmigo en los momentos de estrés y alegría durante este retador camino. Su apoyo, confianza y cariño han sido invaluable. Gracias por ser mi punto de apoyo, mi equipo de aliento y, lo más importante, la familia que yo elegí.

En resumen, agradezco de corazón a todas las personas que contribuyeron y fueron parte de este proceso, ya sea de manera directa o indirecta. Sus contribuciones han dejado una marca indeleble en este trabajo y en mi trayectoria académica. Que este pequeño gesto de gratitud sirva como muestra de mi profundo aprecio por todo su apoyo y amabilidad a lo largo de esta travesía académica.

¡Mil gracias!

José

## ÍNDICE DE CONTENIDOS

ÍNDICE DE ILUSTRACIONES . . . . .	viii
RESUMEN . . . . .	ix
ABSTRACT . . . . .	x
INTRODUCCIÓN . . . . .	1

### CAPÍTULO I

1. PROBLEMA DE INVESTIGACIÓN . . . . .	3
1.1. Antecedentes . . . . .	3
1.2. Planteamiento del problema . . . . .	4
1.3. Objetivos . . . . .	5
1.3.1. <i>Objetivo general</i> . . . . .	5
1.3.2. <i>Objetivos específicos</i> . . . . .	5
1.4. Justificación . . . . .	6

### CAPÍTULO II

2. MARCO TEÓRICO . . . . .	7
2.1. Referencias teóricas. . . . .	7
2.2. Antecedentes teóricos. . . . .	8
2.2.1. <i>Gráficos tridimensionales</i> . . . . .	8
2.2.2. <i>Segmentación y mallado</i> . . . . .	8
2.3. Aplicación del Modelo de los colores RGB . . . . .	10
2.4. Fundamentación teórica . . . . .	10
2.4.1. <i>Conceptos de .Net y C #</i> . . . . .	10
2.4.2. <i>Visual Studio</i> . . . . .	11

### CAPÍTULO III

3. MARCO METODOLÓGICO . . . . .	12
3.1. Descripción de enfoque, alcance, diseño, tipo, métodos, técnica e instrumentos de investigación . . . . .	12

## CAPÍTULO IV

4.	MARCO DE ANÁLISIS E INTERPRETACIÓN DE RESULTADOS . . . .	14
4.1.	Resultado . . . . .	14
4.1.1.	<i>Implementación de las funciones y ecuaciones matemáticas</i> . . . . .	14
4.1.2.	<i>Crear modelos matemáticos para la simulación</i> . . . . .	16
4.1.3.	<i>Modelo de textura de madera</i> . . . . .	18
4.1.4.	<i>Modelo de textura de roca</i> . . . . .	21
4.1.5.	<i>Modelo de textura de agua</i> . . . . .	24
4.1.6.	<i>Otros modelos de textura de madera</i> . . . . .	29
4.1.7.	<i>Otros modelos de textura de roca</i> . . . . .	32
4.1.8.	<i>Otros modelos de textura de agua</i> . . . . .	34

## CAPÍTULO V

5.	CONCLUSIONES Y RECOMENDACIONES . . . . .	36
	BIBLIOGRAFÍA . . . . .	38



## ÍNDICE DE ILUSTRACIONES

<b>Ilustración 1-1:</b> Segmentación de imágenes por textura. . . . .	4
<b>Ilustración 2-2:</b> Paraboloides . . . . .	9
<b>Ilustración 2-3:</b> Imagen RGB . . . . .	10
<b>Ilustración 4-1:</b> (Códigos de color HTML, 2024 ) . . . . .	14
<b>Ilustración 4-1:</b> Madera 1 . . . . .	21
<b>Ilustración 4-1:</b> Roca 1 . . . . .	23
<b>Ilustración 4-1:</b> Agua 1 . . . . .	26
<b>Ilustración 4-1:</b> Madera 2 . . . . .	29
<b>Ilustración 4-1:</b> Madera 3 . . . . .	30
<b>Ilustración 4-1:</b> Madera 4 . . . . .	31
<b>Ilustración 4-1:</b> Roca 2 . . . . .	33
<b>Ilustración 4-1:</b> Roca 3 . . . . .	34
<b>Ilustración 4-1:</b> Agua 2 . . . . .	35

## RESUMEN

Este trabajo de investigación aborda la creación de modelos matemáticos para simular texturas reales de madera, rocas y agua, utilizando como herramienta principal la computación gráfica; se busca resolver la falta de precisión que se encuentra en la simulación de estas texturas, pues los detalles presentes en la realidad son prácticamente infinitos, mientras que la representación por computadora es finita, por lo cual, al usar estas simulaciones en un estudio, generará errores y resultados imprecisos. El objetivo es proponer modelos matemáticos con la ayuda de la computación gráfica, en la creación de funciones y ecuaciones que permitan simular texturas de las superficies propuestas, para mejorar la calidad visual de las texturas en diferentes aplicaciones industriales. La investigación utiliza una metodología mixta, además es de tipo experimental y descriptiva, pues se busca entender las características de las texturas y experimentar con diversas ecuaciones, diseñadas para cada superficie, y los resultados son obtenidos dentro de la plataforma Visual Studio, empleando diversos instrumentos como los modelos de colores RGB y funciones matemáticas específicas para cada tipo de textura planteada. Los resultados incluyen algoritmos que generan texturas detalladas y realistas, manteniendo las características presentes en cada textura de las superficies. Finalmente, la propuesta de generar los modelos de las texturas de madera, rocas y agua, contribuyó en el avance de la representación gráfica manteniendo las características presentes en cada textura de las superficies.

**Palabras clave:** <SIMULACIONES DE SUPERFICIES>, <COMPUTACIÓN GRÁFICA>, <MODELOS DE COLORES EN RGB>, <SIMULACIÓN DE TEXTURAS DE SUPERFICIES>, <GEOMETRIA PROYECTIVA> .

0598-DBRA-UPT-2024



## ABSTRACT

This research work addresses the development of mathematical models to simulate real textures of wood, rocks, and water, using computer graphics as the primary tool; it seeks to solve the lack of precision found in the simulation of these textures, since the details present in reality are infinite, in contrast to the finite nature of computer representations. This discrepancy leads to errors and imprecise results when these simulations are used in studies. The objective is to propose mathematical models, supported by computer graphics, to create functions and equations that accurately simulate the textures of the proposed surfaces, thus enhancing the visual quality of textures in various industrial applications. The research employs a mixed-methods approach and is both experimental and descriptive in nature. The study seeks to understand the characteristics of the textures and to experiment with various equations designed for each surface. The results are obtained within the Visual Studio platform, using various tools such as RGB color models and specific mathematical functions for each type of texture. The outcomes include algorithms that generate detailed and realistic textures, preserving the characteristics present in each surface texture. Finally, the proposal to generate models for the textures of wood, rocks, and water contributed to advancements in graphical representation, maintaining the characteristics present in each surface texture.

**Keywords:** <SURFACE SIMULATIONS>, <COMPUTER GRAPHICS>, <RGB COLOR MODELS>, <SURFACE TEXTURE SIMULATION>, <PROJECTIVE GEOMETRY>

A handwritten signature in blue ink, appearing to read 'María Eugenia Camacho', with a horizontal line underneath.

**Lic. María Eugenia Camacho, M.Sc.**

**0601609597**

## INTRODUCCIÓN

La computación como herramienta, permite realizar procesos que en el pasado eran inimaginables, desde calcular y analizar bases de datos gigantes de los ingresos de una empresa, hasta realizar simulaciones de situaciones, como puede ser generar una simulación de texturas reales; para este último caso, se necesita de una modelación matemática, y utilizando fórmulas y ecuaciones complejas, puedan simular los comportamientos y características de las superficies; pero, generar estos modelos puede ser un gran desafío para cualquier individuo, excepto para los matemáticos, ya que son capaces de identificar y reconocer cuáles son las variables y casos que se debe tomar en cuenta para estudiar las cualidades y estructuras, se podrá realizar un bosquejo de los posibles modelos que se pueden aplicar empleando la llamada “teoría de modelos”.

Aquí es donde la herramienta de la tecnología es usada en el proceso de la simulación, pues facilitará el trabajo para probar los modelos planteados por el matemático, utilizando la herramienta de la programación matemática, la cual se conoce como computación gráfica. Esta va ligada al diseño asistido por ordenadores comúnmente conocidos por las siglas CAD (*Computer-Aided Design*), el cual consiste en crear y modificar diferentes representaciones visuales en segunda y tercera dimensión (Admin, 2021).

Generar imágenes de superficies a base de simulaciones es una tarea complicada para las personas si es que su conocimiento sobre estas es escaso, pues no solo se busca dibujar las diferentes formas y figuras que se encontrarían en las superficies reales, también se busca generar una ilusión de texturización dentro de la imagen que se asemeje a los que se aprecian en la naturaleza. De no lograr este objetivo al momento de simular, se producirá un resultado erróneo y poco fiable para su uso en diversos campos. Este es el problema principal de las simulaciones por computadora, las cuales se han generado obviando los detalles importantes de la texturización, concentrándose en generar las figuras, formas, circunferencias, entre otras representaciones que hallamos en la realidad, y producen imágenes sin realismo ante el espectador.

Para realizar el texturizado de las superficies se necesita de una recolección de datos, con los cuales generen una representación gráfica en la simulación, en donde la percepción generada por los colores y la iluminación, destaque una ilusión visual de las texturas regulares e irregulares halladas en las superficies reales. Por lo tanto, el primer paso para obtener los datos respectivos de la superficie, dependerá principalmente de los colores que la componen, es decir, se necesitará de una pelta de colores capaz de combinar los diferentes tonos, que pintarán los pixeles de las formas y figuras generadas por las ecuaciones empleadas en el modelo, dentro de la pantalla gráfica.

Hoy en día la simulación de texturas de las superficies es muy importante, debido a su uso en varios campos industriales o de entretenimiento, ya que están orientados a mejorar su calidad visual para

el usuario. Por lo cual proponer un modelo matemático que sea capaz de generar una simulación que presente las texturas de las superficies, y que sean visualmente agradables para el observador, basándose en las características esenciales de la combinación de colores presentes en la vida real, será una herramienta muy útil para diversos trabajos en la industria.

Para la redacción de este documento, el trabajo de investigación se dividirá en 5 capítulos, los cuales presentarán definiciones, los datos, las características y como se hallaron los modelos matemáticos que están centrados en el comportamiento de la texturización de las superficies de madera agua y roca, y mostrando el resultado de la simulación obtenida del uso de la computación gráfica.

El primer capítulo servirá como introducción para el trabajo de investigación, en él se mostrarán los antecedentes sobre la computación gráfica, en qué consiste y cómo es aplicada para realizar los procesos de simulación gráfica, también contiene la problemática, el motivo por el cual se realiza este trabajo, los objetivos planteados para el trabajo de integración curricular y finalmente la justificación del mismo.

En el segundo capítulo se presenta el marco teórico referente al trabajo de investigación, en él se narra las referencias y antecedentes teóricos, además se expone otras definiciones importantes para entender el comportamiento de los gráficos por computadora, la segmentación y su mallado, por último, se explica brevemente las herramientas a usar en el proceso de simulación de las texturas, como lo es la aplicación de los colores RGB, y conceptos básicos del programa Visual Studio.

En el tercer capítulo se halla el marco metodológico, donde se describe el enfoque, alcance, diseño, tipo, los métodos y las técnicas que se usaron para este trabajo de investigación, con el fin de generar la simulación de las 3 texturas de las superficies anteriormente expuestas.

El cuarto capítulo comprenderá de los resultados, los pasos usados para generar los diversos modelos matemáticos que simulan las texturas de madera, agua y roca, además se presentará como se implementó estos modelos dentro del programa Visual Studio, y obteniendo las imágenes de las superficies hallas.

Finalmente, el capítulo 5 se dividirá en 2, conclusiones y recomendaciones, las cuales fueron obtenidas tras analizar los resultados de las simulaciones y comprobando el cumplimiento de los objetivos propuestos por este trabajo de investigación, adicionalmente, adicionalmente se menciona en las recomendaciones otras texturas que se pueden realizar utilizando este modelo matemático.

## CAPÍTULO I

### 1. PROBLEMA DE INVESTIGACIÓN

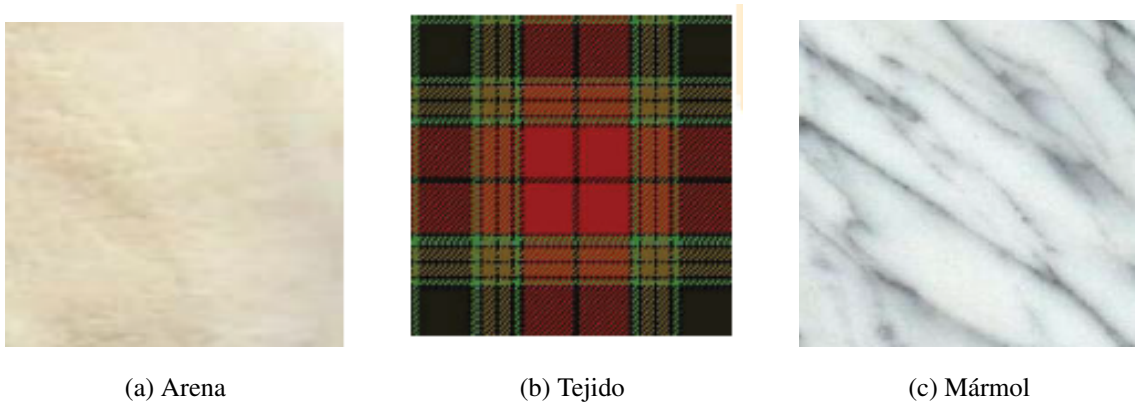
#### 1.1. Antecedentes

Para la creación de modelos matemáticos que permitan realizar simulaciones implementando computación gráfica, y que simulen la textura de elementos, no se encuentra en textos bibliográficos, por tal razón debemos investigar en trabajos, ponencias, escritos que estén relacionados en describir esta rama de la ciencia. Existen algunos programas matemáticos que utilizan lenguajes poco comunes, los cuales son empleados para realizar un estudio de las características importantes de la imagen, la texturización, para aplicarlas en las imágenes base del proceso de simulación; aprender a utilizar los programas para aprovechar sus herramientas, aplicaciones y resultados, que conjuntamente con la matemática pueden generar otras expectativas y nuevos productos, por lo que será un aporte para la industria tanto gráfica como productiva, ya que con una simulación se puede ahorrar materiales que tal vez son costos o difíciles de conseguir.

Las texturas se utilizan en diversas áreas como la ingeniería, el arte, diseño gráfico; un ejemplo de su aplicación es en imágenes satelitales, se las emplea para identificar y describir diferentes objetos, sus características, tales como el agua, la vegetación, montañas, valles y ciudades; en la medicina, al diagnosticar enfermedades; o en el estudio del suelo, ya sea fallas, grietas o cualquier otro análisis de las superficies.

Dar una definición de qué es la textura, no es muy sencillo, “esto se debe a que las personas tienen un conocimiento intuitivo acerca de ella dado por su experiencia” (Pinto, 2006, pág. 2), además, si queremos definir que es una textura, debemos tener en cuenta la gran gama que existe, como son las finas, granuladas, ásperas, etc. Es por ello que, al tratar de detectar la textura de una imagen, esta se puede identificar por sus características, las cuales pueden ser definidas como fina, suave, gruesa o tosca según corresponda.

Berti (2003) expone que algunos autores proponen definir a la textura como algo consistente de elementos mutuamente relacionados, y se los llama elementos “**primitivas**” o “**TEXEL**” (*texture elements*); así representamos en la textura de arena los elementos primitivos, estos serán los granos de arena **Ilustración 1-1:a**. La forma de las “primitivas” pueden ser regulares como es el caso del tejido **Ilustración 1-1:b**, o no tan regular como la en el caso de la tierra o mármol **Ilustración 1-1:c**.



**Ilustración 1-1::** (PINTO, 2006. Figura 1.2)

Así también tenemos definiciones de este concepto expresada por varios autores:

- Pickett: “Textura, término usado para describir arreglos bidimensionales de variaciones. Los elementos y reglas de espaciado o arreglos pueden ser arbitrariamente manipulados, supuesto que aparezca una característica de repetitividad.”
- Irons y Petersen: “La textura visual se refiere a la impresión de rugosidad o suavidad por las variaciones de tono o repetición de patrones a través de una superficie.”
- Harlick y Shapiro: “Una textura está definida por la uniformidad, densidad, grosor, rugosidad, regularidad, intensidad y direccionalidad de medidas discretas del tono y de sus relaciones espaciales.”

Finalmente, consideraremos como la definición principal la planteada por Pinto (2006) “Un arreglo de píxeles cuya relación es la variación espacial de los tonos grises” (pág. 4) .

Esta definición nos servirá para entender que existe una relación entre sí de los distintos elementos de una imagen, por lo que se puede describir algún patrón con el fin de identificar si tiene un aspecto liso, rugoso, homogéneo, heterogéneo o granulado.

## **1.2. Planteamiento del problema**

La modelación matemática de texturas de superficies reales es una herramienta fundamental en la resolución de problemas en diferentes áreas, su aplicación y estudio aún enfrenta una serie de desafíos y limitaciones que deben ser abordados para mejorar su eficacia y utilidad.

La información existente sobre texturas de superficies resulta no ser confiable, ya que el problema radica en los detalles reales que encontramos en la naturaleza, pues son prácticamente infinitas, mientras que la representación y almacenamiento de esa realidad utilizando ordenadores es finita, en consecuencia las mediciones y observaciones de estas texturas pueden ser imprecisas, incompletas y

generará errores, lo que producirá modelos inexactos y poco fiables, "por lo que es necesario extraer una serie de elementos característicos, los cuales se recogerán como valores numéricos que maneja un ordenador, y así poder interpretar la modelación matemática"(Olaya, 2014) , estableciendo una entidad llamada término para denotar el objeto que le interesa al sistema, el mismo que permite observar atributos y propiedades de esta entidad, y así proceder a las actividades para generar el proceso que provoca cambios en el sistema.

Otro desafío importante es la complejidad de las ecuaciones matemáticas necesarias para describir algunas superficies reales, lo que puede requerir técnicas avanzadas y sofisticadas para su resolución. Por tal motivo, tratar de realizar esta simulación de forma manual o de manera empírica logrará llegar a un fracaso total, por ende el utilizar simulación matemática apoyado de la tecnología que en estos momentos está en boga, facilitará el entendimiento y sobre todo la elaboración de diferentes texturas, superficies y otros elementos que permitan mejorar la aplicación de fórmulas y ecuaciones al modelo matemático, y así tratar de asemejarlo a la realidad, en consecuencia encontrar esta alternativa en cosas ya elaboradas se hace difícil, ya que en el mercado de la informática o la realidad virtual, no existe componente alguno que nos ayude a facilitar el diseño gráfico de texturas de superficies de madera, agua y rocas.

### **1.3. Objetivos**

#### **1.3.1. *Objetivo general***

Proponer modelos matemáticos con la ayuda de la computación gráfica en la creación de diferentes funciones y ecuaciones, que permitan simular texturas de superficies de madera, agua y rocas, para ser utilizadas en diversos campos de la industria.

#### **1.3.2. *Objetivos específicos***

- Identificar cuáles son los elementos y características básicas, e implementar funciones y ecuaciones, para poder asemejarlo con las imágenes reales de las superficies de madera, agua y rocas.
- Crear modelos matemáticos, utilizando funciones y ecuaciones para poder simular texturas parecidas a la superficie.
- Implementar modelos matemáticos con la ayuda de la computación gráfica que permitan elaborar texturas similares a la superficie, empleando el programa Visual Studio.



#### 1.4. Justificación

"Hoy en día es difícil no encontrar una disciplina científica que no pueda sacar partido de los modelos matemáticos de superficies, y que no contemple a estos como herramientas de primera línea"(Olaya, 2014) . "La modelación de superficies describe generalmente el proceso de representación física y artificial de una superficie, mediante un modelo geométrico"(Petrie, 2014), pues se basa en la representación de estas mediante ecuaciones matemáticas, que describen su forma y características. Estas ecuaciones pueden ser polinómicas, trigonométricas, sinusoidales, exponenciales, entre otras, y pueden ser ajustadas a los datos obtenidos a partir de mediciones o simulaciones.

Uno de los principales beneficios de la modelación matemática de superficies reales es su capacidad para proporcionar una representación precisa y detallada de estas superficies, lo que permite analizar su comportamiento y sus propiedades en diferentes condiciones sin necesidad de realizar pruebas físicas costosas y laboriosas. Por lo que se convierte en una herramienta fundamental para la resolución de problemas en diversas áreas, como la ingeniería, la física, la química, la medicina, la biología, entre otras.

Incluso fuera del ámbito científico, los modelos de superficies son parte de nuestra vida diaria, es una parte fundamental en muchos procesos creativos, desde el cine y la animación hasta la arquitectura y el diseño de interiores, también son claves en el diseño y desarrollo de videojuegos. (Tokio, 2023).

En resumen, la modelación matemática de texturas de superficies reales es una herramienta esencial para la resolución de problemas en diferentes áreas, que permite obtener modelos precisos y detallados, capaces de analizar su comportamiento y propiedades, simplificar el proceso de diseño y optimización. Por tal motivo se busca desarrollar modelos matemáticos capaces de simular texturas de superficies de madera, agua y rocas, que resulten fundamentales en el desarrollo de la ciencia y la tecnología.

## CAPÍTULO II

### 2. MARCO TEÓRICO

#### 2.1. Referencias teóricas.

Para el proceso de modelación es importante contar con la herramienta de la computación, para este tema en particular, se empleará la computación gráfica, pues pertenece al área de la informática vinculada a la generación y creación de imágenes, se encarga de la edición de contenidos gráficos y multimedios a través de ordenadores.

“Este proceso permite mejorar las visualizaciones, hacer un diseño más sólido y limpio, reducir el tiempo de elaboración, tener una mejor precisión en el trabajo realizado y así mismo disminuir errores”. (Admin, 2021), por ello la simulación por ordenador será una gran herramienta con la cual buscará usar el modelo matemático para estudiar su comportamiento, con ello, “ podemos prever cuál será el estado final del sistema modelado al cambiar las variables que lo afectan, evitando los problemas que pueden conllevar al trabajar directamente con el sistema real” (Cayado, 2014, págs. 263-264).

En la actualidad, modelar objetos 3D usando la computación, se ha convertido en una herramienta fundamental para el desarrollo de la mayoría de las ciencias. Al generar una superficie tridimensional con el computador, cuyo despliegue es una pantalla gráfica bidimensional, presenta algunos desafíos como el de simular la profundidad en la pantalla gráfica, o el identificar y remover las partes ocultas de la superficie. Para superar este inconveniente se propone utilizar el Análisis Diferencial Vectorial (Geometría Diferencial), ya que, mediante el cálculo del vector normal a la superficie, se puede eliminar secciones escondidas y diferenciar caras externas de las internas para texturizar de forma diferente. “De igual manera aprovechando las propiedades del Vector Gradiente se logra simular intensidades de luz sobre las superficies, pues la iluminación, el sombreado, la textura, y el color son parte fundamental en la representación de la superficie” (Álvarez, 2018, pags 255-258).

Según Álvarez (2018), desde una perspectiva física, una superficie puede emitir luz por su propia emisión, como focos de luz, o reflejar luz de otras superficies que la iluminan. El color que se ve en un punto de un objeto, está determinado por las múltiples interacciones entre las fuentes de luz y superficies reflectivas. Para comprender el proceso de iluminación, se puede comenzar siguiendo los rayos de luz de un punto fuente, donde el observador ve solamente la luz que emite la fuente y que llega a los ojos (pags. 255-258).

## 2.2. Antecedentes teóricos.

### 2.2.1. Gráficos tridimensionales

Para la representación gráfica de objetos (curvas, superficies, . . .) tridimensionales, es preciso utilizar técnicas de geometría proyectiva para determinar la perspectiva y conseguir la impresión de tridimensionalidad.

Adicionalmente, aparece la necesidad de utilizar algoritmos y técnicas complejas para determinar partes ocultas y, aún más, iluminación, transparencias, aplicación de texturas, etc. Por lo que primero se presentara un estudio rápido de la representación gráfica de “objetos” matemáticos tridimensionales. “Una superficie en tres dimensiones se puede definir como el grafo cartesiano de una función continuamente diferenciable de dos variables” (Glyn, 2012).

$$z = F(x, y).$$

$$F : \Omega \subseteq \mathbb{R}^2 \rightarrow \mathbb{R}.$$

O como el grafo de una función paramétrica, también llamada función vectorial, donde tenemos a  $t \in [a, b]$  y  $h \in [c, d]$ .

$$\alpha(t, h) = (x(t, h), y(t, h), z(t, h)).$$

Así se puede presentar un modelo matemático, pues, se puede considerar como la representación de cualquier fenómeno real utilizando estructuras matemáticas. Con estos conceptos establecidos, se pretende llegar a la sección de segmentación y mallado, para poder plantear las fórmulas que serán empleadas en la elaboración del texturizado de las superficies.

### 2.2.2. Segmentación y mallado

Para la segmentación y el mallado, suponemos que  $\Omega$  sea un espacio rectangular.

$$\Omega = \{(x, y) \in \mathbb{R}^2 | a \leq x \leq b, c \leq y \leq d\}.$$

Si la superficie está definida por  $z = F(x, y)$  donde  $(x, y) \in \Omega$  y  $F$  es una función continua y diferenciable.

Sean  $n, m$  el número de particiones en el eje  $x$  y en el eje  $y$  respectivamente entonces, si para cada  $i = 1$  hasta  $n$  y para cada  $j = 1$  hasta  $m$  tenemos las siguientes expresiones.

$$h_x = |b - a|/n.$$

$$h_y = |d - c|/m.$$

Y definimos tres matrices con valores reales,  $V_x, V_y, V_z$ , de la siguiente forma.

$$V_x[i, j] = a + i * h_x.$$

$$V_y[i, j] = c + i * h_y.$$

$$V_z[i, j] = F(V_x[i, j], V_y[i, j]).$$

Ahora, la transformación 3D a 2D que se realiza físicamente en el sistema visual humano, se tiene que realizar mediante un proceso matemático en un sistema de graficación por computadora. Este proceso se le conoce como Axonométrico.

$$Axometria(x, y, z) \rightarrow (ax, ay).$$

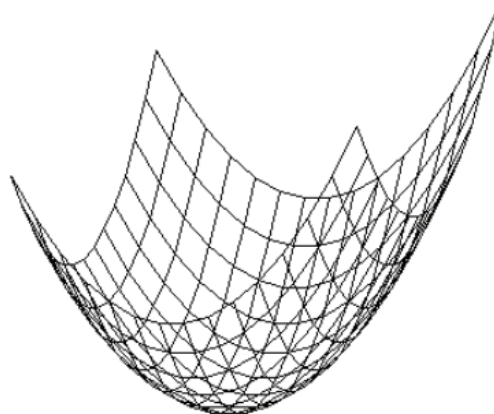
$$ax = y - p * x * \cos(\alpha).$$

$$ay = z - p * x * \sin(\alpha).$$

Donde  $p = 0.5; \alpha = \pi/4$ , así usando este proceso se puede transformar las matrices.

$$(V_x, V_y, V_z) \rightarrow (A_x[i, j], A_y[i, j]).$$

Y finalmente se construye la aproximación de la superficie mediante el mallado (Álvarez, A., Et al, 2018. pags. 255-258).



**Ilustración 2-2::** (Álvarez, A., Et al, 2018. Figura 1)

### 2.3. Aplicación del Modelo de los colores RGB

Para generar una imagen en la pantalla gráfica se necesita definir los colores que la conformarán, para esto se emplean modelos matemáticos que pueden describir los colores que vemos y con los cuales trabajaremos, entonces, para la generación del modelo de texturas, se utilizará el modelo RGB.

Para entender en qué consiste este modelo, describimos la definición de RGB. Cuando se habla de RGB (*Red, Green, Blue*), se hace referencia a un sistema de composición de colores basado en la adición de los colores primarios de la luz. Al combinarlos en un modelo, se pueden definir los colores que percibe el ojo humano ante las diferentes frecuencias de la luz y sus interferencias. La aplicación de estos colores del modelo RGB se encuentra en el uso de televisores, en las pantallas de los computadores, y equipos informáticos, ya que las pantallas se componen de píxeles, los cuales se dividen en tres subpíxeles. Cada subpíxel representa un color primario de la luz. Para el uso de este modelo en los sistemas informáticos, se asigna un valor numérico a la intensidad de cada una de las componentes de color (Efectoled, 2023).



**Ilustración 2-3::** (Efectoled, 2023)

### 2.4. Fundamentación teórica

#### 2.4.1. *Conceptos de .Net y C #*

.NET es una herramienta de desarrollo de aplicaciones presentado por Microsoft, tanto para el entorno de Windows, como para otros entornos; y que sea un modelo estándar para el Internet y otros dispositivos que se conecten a ella. El objetivo principal de .NET en Internet es ser la base de un sistema operativo distribuido sobre el cual se ejecutarán aplicaciones que estarán relacionadas entre sí. Para proporcionar a los usuarios información un lenguaje consistente, uniforme, configurable y fácil de manejar. Así se puede deducir que en un futuro la programación se hará en un sistema operativo dentro de la Red.

En la plataforma .Net se puede escribir códigos en muchos lenguajes como pueden ser Visual Basic.NET o JScrip.NET pero, C# (pronunciado en inglés *C Sharp*) es el único diseñado para ser utilizado en esta plataforma, por lo cual, empleando este lenguaje, el proceso de programación

se hace más sencillo e intuitivo. Es por esta razón que C# es el lenguaje nativo de .Net, creado por Microsoft para la plataforma, diseñada por Scott Wiltamuth y Anders Hejlsberg, conocido por diseñar el lenguaje Turbo Pascal y la herramienta RAD Delphi (Villa, R. 2002, pág. 14).

#### **2.4.2. *Visual Studio***

Visual Studio es una herramienta muy popular, utilizada por programadores a nivel mundial; es un entorno de desarrollo integrado, generado por Microsoft que permite a los programadores crear aplicaciones para diferentes plataformas, utilizando diversas herramientas como la depuración del código, integración con sistemas de control de versiones, pruebas automáticas y asistencia en la escritura del código.

La historia de Visual Studio inicio en 1989, cuando Microsoft lanzó su primer entorno de desarrollo integrado, conocido como Visual Studio Basic, esta herramienta se convirtió en una de las más populares para el desarrollo de aplicaciones de Windows. En 1997 se lanzó Visual Studio 97, esta versión conto con un entorno de desarrollo integrado para diversas plataformas, incluyendo Windows y su lenguaje de programación. Visual Studio cuenta con múltiples ediciones, y es compatible con distintos lenguajes de programación como C++, C#, F#, Visual Basic, Python, entre otros ( Urrutia, D. 2023).

## CAPÍTULO III

### 3. MARCO METODOLÓGICO

#### 3.1. Descripción de enfoque, alcance, diseño, tipo, métodos, técnica e instrumentos de investigación

Generar modelos matemáticos capaces de realizar el proceso de simulación de las texturas de superficies, brindará una ayuda importante con su implementación en diversos trabajos de las industrias, pues será una herramienta de estimación y pronóstico que ahorrará no solo dinero, si no, también tiempo al evitar construir el sistema en la realidad.

Así la investigación que se propone usará una metodología mixta, es decir, cuantitativa y cualitativa, con el fin de proporcionar modelos matemáticos, empleando funciones y ecuaciones, capaces de cumplir, satisfacer y resolver los inconvenientes y problemas sobre las características y formas que se deben generar en la simulación de texturas de las superficies de madera, agua y rocas.

Uno de los principales problemas es la estimación de las formas y figuras que se encuentran en dichas superficies, pero la más importante es la caracterización de la textura que se presenta en la simulación, la cual se la puede identificar por sus patrones, es decir, su aspecto liso, la rugosidad, la homogeneidad, entre otras; las cuales son características cualitativas, por lo que se debe cuantificar esas características, analizar la frecuencia de los pixeles en las imágenes resultantes.

Finalmente se empleará los modelos obtenidos dentro del programa Visual Studio Code, el cual usa el lenguaje de *C-Sharp*; es ahí donde se obtendrá la presentación de las imágenes simuladas con sus texturas, también se podrá comprobar visualmente el comportamiento de las funciones y ecuaciones empleadas en el modelo, y si es necesario se ira interpolando dichas funciones con otros datos, con el fin de formularlo de la manera más efectiva.

La investigación es de tipo experimental y descriptiva, puesto que, para realizar el proceso de simulación, se necesita comprender las características de las texturas, por lo que se experimentó con diferentes ecuaciones, simples y complejas, con el fin de observar los resultados de su comportamiento en la implementación del modelo matemático generado dentro del entorno Visual Studio.

Luego se podrá analizar cada imagen generada a través de esta simulación, finalmente aplicando diferentes colores dentro de una paleta, se podrá describir si el resultado se asemeja a algo real presente en la naturaleza, y de ser necesario se cambiarán los valores escogidos entre las variables, para de obtener diferentes resultados presentes dentro de la simulación de las texturas.

Para generar las texturas de las superficies planteadas, es necesario manipular las ecuaciones dentro de los modelos propuestos en el programa, así se interpolará las ecuaciones dentro del modelo, con el fin de prever los cambios que presenta el resultado final al cambiar dichos valores obtenidos para las 3 superficies.

Durante el proceso de experimentación de la simulación, se tomará en cuenta las características de las 3 superficies, en especial, la aleatoriedad de las formas y figuras que se generan en una superficie, así, se considera la repetición de dichas formas y figuras para nuestra simulación, y que se presenten de forma aleatoria en la imagen final; a continuación, se necesita visualizar esta simulación, por lo que se requiere de un espacio para presentar dicha simulación, en otras palabras, se empleará esta simulación dentro de una pantalla gráfica. Para el resultado final se espera obtener una imagen que, al observarla, garantice al espectador una sensación de estar observando una imagen real y no una imagen generada por computadora, así se garantizará una similitud a las texturas presentes en la naturaleza.

Adicionalmente, este trabajo presenta una propuesta de investigación de campo, basada en la recopilación de información a través de la observación directa e indirecta, aplicando diferentes fórmulas y ecuaciones capaces de representar de forma visual la información presente de las texturas generadas, considerando la aleatoriedad de los patrones en las superficies a simular, las cuales se diferencian visualmente por los tonos y colores que se mezclan con naturalidad, aquí es donde se emplea una paleta de colores implementada mediante los modelos RGB, las cuales combinarán estos tonos con naturalidad, así se obtiene una distribución aleatoria de cada pixel, pintándolo de un color seleccionado de acuerdo al modelo de colores en RGB, y al combinarlas ejecutando las ecuaciones, se podrá simular los patrones aleatorios de la superficie dentro del entorno implementado para la simulación.

La mayoría de superficies se generan empleando de forma aleatoria ecuaciones de segundo y tercer grado, sinusoidales, logarítmicas y exponenciales, estas ecuaciones generan gráficas de ondas con curvas pronunciadas y simétricas, las cuales presentan longitudes, frecuencias y amplitudes, que pueden alterar sus valores, según los requisitos del usuario, por ende serán muy útiles en la simulación de cada textura, y al combinarlas con los modelos RGB, se espera crear un modelo matemático, que simule gráficas complejas, y al presentarlas dentro de la pantalla gráfica, la cual pintará a cada pixel de forma natural y sin presencia de ruido, obtenga una imagen de textura con formas reconocibles y similares presentes en las superficies a simular.



## CAPÍTULO IV

### 4. MARCO DE ANÁLISIS E INTERPRETACIÓN DE RESULTADOS

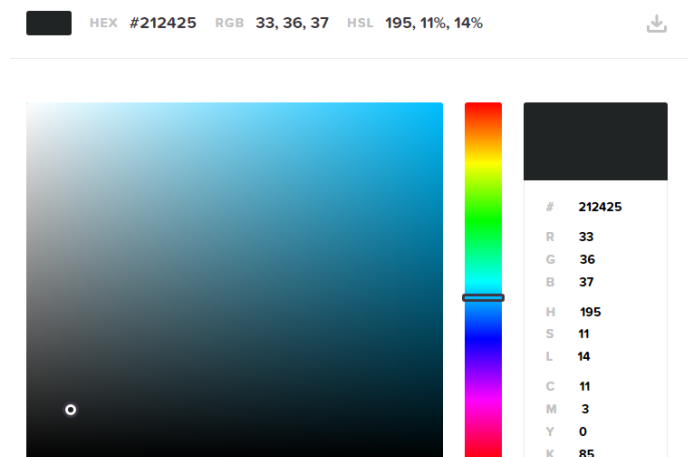
#### 4.1. Resultado

##### 4.1.1. Implementación de las funciones y ecuaciones matemáticas

Para la recolección de datos sobre las texturas de las superficies, se debe tener en cuenta la definición de las texturas visuales, las cuales consisten en una representación gráfica sobre una superficie plana en 2 dimensiones. La percepción y la capacidad expresiva de una textura depende de la iluminación y el color, pues destaca la sensación visual de una superficie regular e irregular, las cuales pueden ser lisas o rugosas, y al presentarse con una combinación de colores produce un efecto estático o de movimiento visual.

Por ende, el primer paso para obtener los datos respectivos de la superficie dependerá principalmente de los colores que la componen, es decir, se necesitará de una paleta de colores capaz de combinar los diferentes tonos que pintarán las formas y figuras generadas por las ecuaciones empleadas en el modelo.

La paleta de colores será escogida dentro de la página “Códigos de Color HTML”, en ella se presentan todas combinaciones de colores dependiendo del tono escogido, estos colores se tomarán en formato RGB, las cuales se conforman de 3 números representadas en un vector de 3 componentes, cada valor representa la intensificación del tono de colores en rojo, verde y azul respectivamente. Para el caso de este modelo de computación gráfica se usó una paleta de 32 colores, pues esta permitirá una amplia paleta con diferentes tonos de un mismo color.



**Ilustración 4-1:** (Códigos de color HTML, 2024 )

Luego de escoger los colores para la simulación de la superficie se necesita que estos colores sean seleccionados por los valores que tome las variables de las funciones planteadas, pero la combinación de los 32 colores debe pintar con armonía la simulación presente en la pantalla gráfica, por lo que se generará un bucle empelando funciones lineales para cada color en RGB dentro de la paleta escogida.

Los valores R, G, B serán calculados empleando las siguientes funciones lineales de variable  $i$ , la cual servirá para orientar la ubicación de los colores dentro de la pantalla gráfica.

La relación lineal se expresa de la siguiente forma para cada componente en RGB.

**Componente rojo.**

$$r = c_1 - \frac{d_1}{16} * i.$$

Donde  $c_1$  y  $d_1$  son constante y representan la componente en rojo de los 2 colores seleccionados que conformaran la paleta de 32 colores.

**Componente verde.**

$$g = c_2 - \frac{d_2}{16} * i.$$

Donde  $c_2$  y  $d_2$  son constante y representan la componente en verde de los 2 colores seleccionados que conformaran la paleta de 32 colores.

**Componente azul.**

$$b = c_3 - \frac{d_3}{16} * i.$$

Donde  $c_3$  y  $d_3$  son constante y representan la componente en azul de los 2 colores seleccionados que conformaran la paleta de 32 colores. La obtención de las constantes para las componen se obtendrán empleando la siguiente ecuación.

$$d_n = \frac{(c_n - e_n)}{2}; \quad \forall n = \{1, 2, 3\}.$$

donde  $c_n$  son los componentes del color más claro de la paleta y  $e_n$  son los componentes del color más oscuro de la paleta. Usando estas funciones dentro del código, se podrá pintar la textura de la superficie a gusto del usuario.

El proceso de simulación de las texturas necesita de la implementación de diferentes funciones matemáticas, estas desempeñan un papel esencial al proporcionar las herramientas para realizar el proceso de modelaje, con el propósito de lograr resultados visuales que den la sensación del texturizado.

Las funciones que se emplearon en este proceso son las exponenciales, logarítmicas, trigonométricas

y polinómicas. Estas funciones son usadas por su comportamiento al momento de graficarlas, por ejemplo, al emplear las funciones exponenciales da la posibilidad de manipular la luz que interactúa con la superficie, ofreciendo una representación realista de efectos como el brillo y la refracción en la textura. De la misma forma, las funciones logarítmicas desempeñan un papel crucial en el mapeo de las variaciones de la opacidad y densidad de las texturas, permitiendo los efectos de transparencia y sombreado. En cambio, el uso de las funciones trigonométricas, como son el seno, coseno y tangente, son fundamentales para recrear los patrones y detalles presentes en la textura; su representación gráfica de ondulación garantiza figuras presentes dentro de varias superficies, como lo es en el caso del agua y madera. Su capacidad de producir repeticiones periódicas y variaciones a lo largo de la imagen contribuye a la creación de las superficies.

Mientras el uso de las funciones polinómicas, permite modelar transiciones suaves entre las diversas texturas, creando un efecto de mezcla que añade una profundidad y genera la sensación de realismo en la representación visual, este tipo de funciones fueron utilizadas anteriormente, pues proporcionó una de la paleta de colores en RGB para el proceso de modelaje.

La combinación de estas funciones matemáticas nos ofrece un amplio abanico de posibilidades, como crear efectos visuales realistas, y lograr capturar la complejidad de las texturas de las superficies de madera, agua y rocas. Además, el uso de las funciones en la simulación de las texturas no solo proporciona un enfoque preciso y detallado, si no también permite optimizar la eficacia computacional, pues aprovecha su naturaleza predictiva y puede lograr resultados impactantes con un menor costo computacional. Por lo que se puede decir que la integración inteligente de funciones matemáticas en el proceso de la simulación de texturas, es fundamental para lograr un equilibrio entre el realismo visual y eficaz en el uso de la herramienta de la computación gráfica para diversos artistas y programadores, que deseen dar vida a sus ideas de nuevos mundos virtuales.

#### **4.1.2. *Crear modelos matemáticos para la simulación***

Como primer paso para empezar con el modelo matemático, se necesita del uso del entorno Visual Studio, en él se requiere del uso de herramientas que permitan la presentación de la simulación usando una pantalla gráfica, dicha pantalla se obtendrá utilizando la aplicación de Windows Forms, esta aplicación proporciona una forma más efectiva al momento de crear aplicaciones de escritorio basadas en el diseño visual de imágenes por computadora proporcionado por Visual Studio.

En esta aplicación se escoge 2 herramientas, las cuales serán importantes en la presentación de la simulación, el "Picture Box" y los "Buttons", ellos permitirán dividir el código principal en varias partes, las cuales permitirían manejar de forma más sencilla la manipulación del código a generar. El Picture Box presentará la imagen final simulada, además que combinará los colores presentes

en la paleta de colores RGB, por lo que el resultado presentará una armonía en la combinación de colores, dándoles tonos que se pueden hallar en la vida real.

Los Buttons serán usados para generar los códigos de cada textura, en ellos se implementará los diversos códigos, donde las ecuaciones seleccionadas harán el trabajo de calcular diversos valores a los cuales serán tomados por los pixeles dentro de la pantalla gráfica del Picture Box.

Además, se debe tener en cuenta que estas ecuaciones deben estar relacionadas con el modelo de colores RGB, por lo que este modelo debe ser manipulado, escogiendo la paleta de colores para cada textura, un ejemplo en esta manipulación puede considerarse el tartar de generar una superficie con los colores gris y verde, en este caso, se debe manipular el modelo de colores, así se obtendrá una paleta que contenga dichos colores, y no los confunda con colores diferentes al momento de ir pintando la pantalla con los pixeles, y así se obtendrá los colores seleccionados para dicha simulación.

#### Listing 1: Código en C# para SetPixel en Bitmap

```
void Bitmap.SetPixel(int x, int y, System.Drawing.Color color);
```

A continuación, para iniciar con el proceso de la simulación, se ingresa dentro de un Button, en su sección se introduce y se modifica la paleta de colores dentro del código, también se puede ingresar dentro de la paleta de colores un solo color, para este caso se puede colocar simplemente dentro del código cuales son los números que conforman al color dentro del modelo de colores en RGB. Para nuestro caso se necesita que exista una paleta de colores, pues al emplearla dentro del código se debe ir cambiando los colores de los pixeles dependiendo de su posición, además que este cambio de colores debe presentar una armonía, pues el cambio brusco de tonos dañaría la manipulación de los tonos presentes en la imagen, por ello se usará un bucle capaz de ir alternando los colores seleccionados dentro de la paleta de colores. Esta paleta de colores en RGB se debe ingresar empleando el siguiente código.

#### Listing 2: Código en C# para generar la paleta de colores

```
Color[] palette = new Color[32];  
int modFactor = palette.Length;  
Color color;  
int r, g, b, colorT;  
palette[i] = Color.FromArgb(r, g, b);
```

Empleando este código de la paleta de colores dentro del código del modelo de las texturas de las superficies, se puede realizar el proceso de coloreo de los pixeles, dándole valores a cada uno sin importar su ubicación dentro de la pantalla gráfica, finalmente, para dar estos valores a los pixeles,

se requiere el uso de ecuaciones con 3 variables, i, j y de las variables que define a cada textura; por lo que ahora dependerá el comportamiento de las formas y figuras únicamente de las ecuaciones planteadas en cada caso.

El siguiente paso será explicar cómo se generaron los modelos de cada textura, seleccionando primero los colores y luego colocar las ecuaciones necesarias para simular la textura.

#### 4.1.3. Modelo de textura de madera

Para el modelo de madera se planea generar betas, una característica primordial de esta textura, y con ellas combinar las diferentes formas que podemos hallar, tomando en cuenta los colores y los cambios de tono de color característico de la superficie. Por lo que se plantea el siguiente modelo de colores en RGB, en el cual se colocan los diferentes tonos relacionados al color café claro, estos comprenden desde (81, 26, 0) hasta (255, 144, 26), con ellos se crea la paleta de colores, y se utiliza estos colores usando el código de la paleta anteriormente explicado (Listing 2), luego, empleando un bucle en este código se puede intercambiar los colores dentro de la Picture Box.

Después, se genera el código, en el cual se presentan las ecuaciones seleccionadas para la simulación de la textura, para este caso se escogieron las siguientes ecuaciones que al emplearlas se crea dicha simulación.

$$Ma_1 = \frac{\sqrt{\tan(i * j) * (i + 1400)}}{15}$$
$$Ma_1 = 1.03 * \sqrt{((i^2 * 25) + ((j + 100)^3 + 25))}$$
$$Ma_1 = \frac{\sqrt{\tan((i + 100) * (j + 100)) * (i + 500)}}{76}$$

Cabe recalcar la importancia del uso de la raíz cuadrada, la cual acortan los resultados obtenidos dentro de la ecuación, el propósito de acortar los resultados se debe a que los valores que obtienen permitirán reducir la simulación dentro de la pantalla gráfica, y evitar una dispersión de colores al momento de pintar con la paleta de colores. Se emplearon 3 ecuaciones con el propósito de combinar las figuras de forma aleatoria, y así garantizar una ilusión de realismo a la mezcla dentro de la simulación.

Para poder visualizar dicha imagen se debe emplear el código Listing 1, así se puede observar dicha simulación en la pantalla de la Picture box. Por último, se presenta el código del modelo planteado para la simulación de la imagen que presenta la textura de madera, la cual fue llamada Madera 1.

Listing 3: Código en C# para la simulación de superficie de Madera 1

```
Color[] palette = new Color[32];  
int modFactor = palette.Length;
```

```

Color color;
int r, g, b, Ma_1;

for (int i = 0; i < 32; i++) \\Bucle para generar los cambios de
colores en la paleta
    r = (int)(-i * (87 / 16) + 255);
    g = (int)(-i * (59 / 16) + 144);
    b = (int)(-i * (13 / 16) + 26);
palette[i] = Color.FromArgb(r, g, b);

for (int i = 0; i < pbViewPort.Width; i++)
    for (int j = 0; j < pbViewPort.Height; j++)
        if (j < 50 || (j > 500 && j < 600))
Ma_1 = (int)(Math.Sqrt(Math.Tan(j * j) * (i + 1400)) / 15) % 32;
color = palette[Ma_1];
viewport.SetPixel(i, j, color);
continue;

    Ma_1 = (int)((Math.Sqrt(((i * i) * 25) + (Math.Pow((j + 100),
3) * 25)) * 1.03));

    if (Ma_1 % modFactor > 15)
Ma_1 = (int)(Math.Sqrt(Math.Tan((j + 100) * ((i * j) + 150))
* (i + 500)) / 76) % 32;
color = palette[Ma_1];
viewport.SetPixel(i, j, color);
    continue;
color = palette[Ma_1 % 32];

viewport.SetPixel(i, j, color);
pbViewPort.Image = viewport;

```

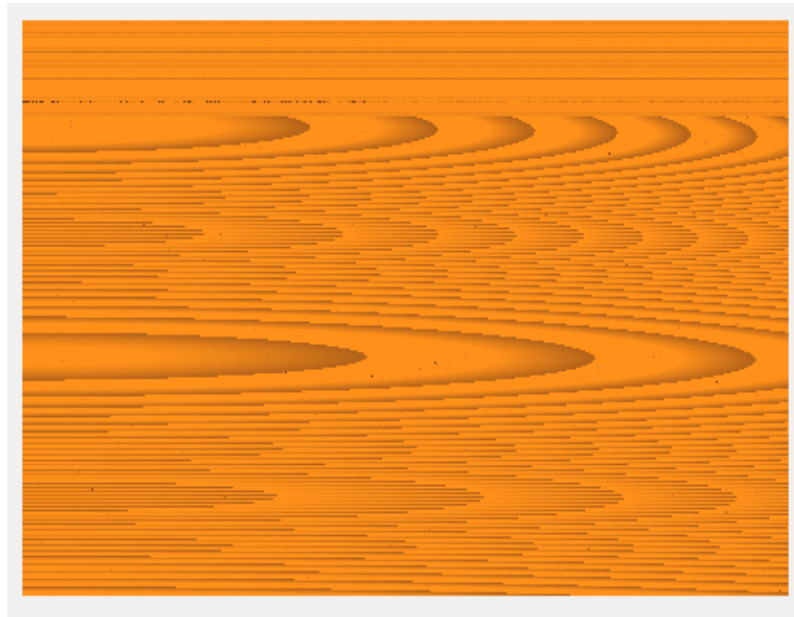
Para apreciar el comportamiento de las ecuaciones planteadas en el modelo matemático, se presenta una tabla de valores con las variables  $i$ ,  $j$  y  $Ma_1$  (Tabla 1), la misma que solo toma los primeros 100 valores como muestra, pues al ser una imagen con dimensiones de 500 x 400 píxeles, contiene una cantidad enorme de valores que toma la variable  $Ma_1$  en toda la simulación.

Tabla 1: Valores obtenidos de la simulación Madera 1

<i>i</i>	<i>j</i>	valor	<i>i</i>	<i>j</i>	valor	<i>i</i>	<i>j</i>	valor	<i>i</i>	<i>j</i>	valor	<i>i</i>	<i>j</i>	valor
0	0	0	0	20	3	0	40	2	0	60	0	0	80	0
0	1	3	0	21	3	0	41	1	0	61	0	0	81	0
0	2	2	0	22	1	0	42	6	0	62	0	0	82	12644
0	3	0	0	23	4	0	43	0	0	63	0	0	83	12749
0	4	1	0	24	3	0	44	2	0	64	10816	0	84	0
0	5	0	0	25	0	0	45	0	0	65	10915	0	85	0
0	6	6	0	26	1	0	46	0	0	66	11014	0	86	13064
0	7	0	0	27	0	0	47	1	0	67	11114	0	87	0
0	8	3	0	28	0	0	48	4	0	68	11214	0	88	0
0	9	0	0	29	0	0	49	2	0	69	0	0	89	13381
0	10	0	0	30	9	0	50	0	0	70	0	0	90	13487
0	11	0	0	31	0	0	51	0	0	71	0	0	91	0
0	12	0	0	32	0	0	52	0	0	72	11617	0	92	13701
0	13	0	0	33	0	0	53	0	0	73	11718	0	93	0
0	14	4	0	34	0	0	54	0	0	74	11820	0	94	0
0	15	0	0	35	0	0	55	0	0	75	0	0	95	14023
0	16	12	0	36	0	0	56	0	0	76	0	0	96	0
0	17	0	0	37	0	0	57	0	0	77	0	0	97	0
0	18	1	0	38	0	0	58	0	0	78	12230	0	98	14348
0	19	0	0	39	1	0	59	0	0	79	12333	0	99	0

Finalmente se presenta la imagen resultante de la simulación de la textura de madera, la cual se la puede obtener dentro de la pantalla del PictureBox, al momento de reproducir el código en el programa Visual Studio.

**Ilustración 4-1:** Madera 1



#### **4.1.4. Modelo de textura de roca**

Empleando el mismo código Listing 2 se hace el respectivo cambio para seleccionar los colores relacionados a tonos en grises, así se presenta la nueva paleta de colores que servirá para pintar las texturas de la superficie de roca; para la siguiente simulación se emplea la paleta de colores que comprende los colores desde (33, 36, 37) hasta (87, 100, 103); el siguiente paso es ingresar esta paleta de colores dentro del bucle, entonces se podrá pintar de forma aleatoria en la pantalla Picture Box para simular gráfica.

Usando las siguientes ecuaciones dentro del código de la simulación se puede generar la textura de roca, está se caracteriza por la enorme aleatoriedad de sus formas y sus colores, por lo que puede ser rápida de generar, esto se debe a que una superficie rocosa se compone de una mezcla de varios minerales y vidrio volcánico en una sola masa sólida, entonces, para simular la texturización de la superficie se puede ocupar diversas ecuaciones que generen formas aleatorias, las ecuaciones escogidas para esta superficies son siguientes ecuaciones.

$$Ro_1 = \sqrt{\frac{5}{2}(i * j + 2 * j)^3 + (-10 - 2 * i)^3 - Tan(4 * i)^2}.$$
$$Ro_1 = \sqrt{(12 * i - 25)^2 + \frac{(j + 100)^6}{j^2}}.$$
$$Ro_1 = \frac{\sqrt{(20 + j) * i * j + 200 + (i + 150)^3}}{70}.$$

Nuevamente se usa el arreglo de colocar las ecuaciones dentro de raíces cuadradas, así se acorta la cantidad de valores que van a tomar los pixeles para ser coloreados, dentro de las ecuaciones



se puede observar la presencia de polinomios de primer y segundo grado, y además elevados a diferentes grados, se realiza este cambio en las ecuaciones pues se requiere una dispersión aleatoria de los resultados, y al estar ubicados dentro de un solo intervalo, el cual comprende toda la pantalla gráfica, genera una simulación con bastante ruido, pero a la vez, está mezclada con consonancia, y produce al espectador una sensación agradable. Teniendo en cuenta estas características, se generó el siguiente código para simular la textura de la superficie rocosa, empleando la paleta de colores con tonos grises, esta textura se llama Roca 1.

Listing 4: Código en C# para la simulación de superficie de Roca 1.

```

Color[] palette = new Color[32];
int modFactor = palette.Length;
Color color;

int r, g, b, Ro_1;

for (int i = 0; i < 32; i++)
    //la paleta de colores esta comprendida por (54, 64, 66) hasta
    los colores (87, 100, 103).
    r = (int)(-i * (27 / 16) + 87);
    g = (int)(-i * (32 / 16) + 100);
    b = (int)(-i * (33 / 16) + 103);
    palette[i] = Color.FromArgb(r, g, b);

for (int i = 0; i < pbViewPort.Width; i++)
    for (int j = 0; j < pbViewPort.Height; j++)
        if (j < 300 || (j > 300 && j < 600))
            Ro_1 = (int)(Math.Sqrt((5 / 2) * Math.Pow((j * i + 2 * j), 3)
            + Math.Pow((-10 - 2 * i), 3) - Math.Pow(Math.Tan(4 * i), 2))) % 32;
            color = palette[Ro_1];
            ViewPort.SetPixel(i, j, color);
        continue;

Ro_1 = (int)(Math.Sqrt(Math.Pow((12 * i * i - 25), 2)
+ (Math.Pow((j + 100), 6) / j * j)));

```

```

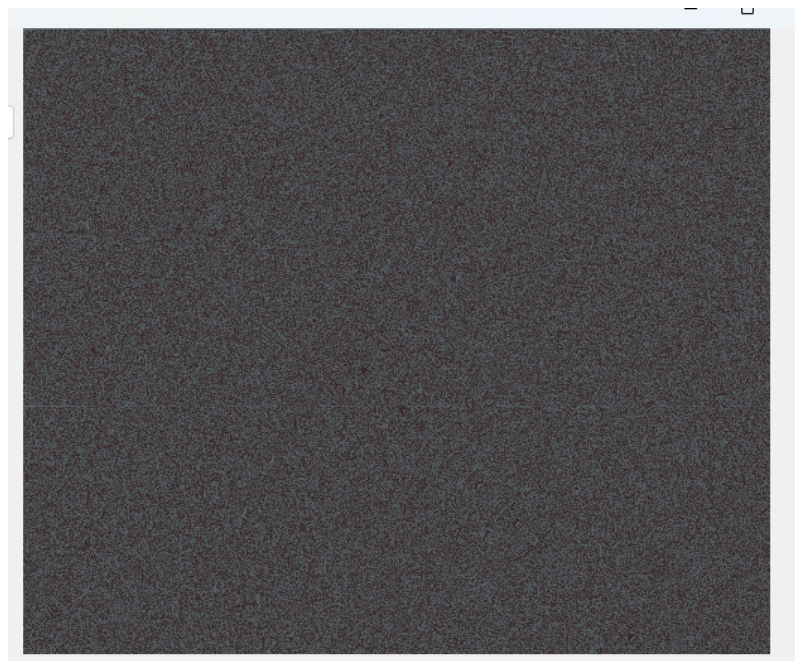
if (Ro_1 % modFactor > 15)
    Ro_1 = (int)(Math.Sqrt(((j + 20) * (i * j) + 200)
    + Math.Pow((i + 150), 3)) / 70) % 32;
color = palette[Ro_1];

Viewport.SetPixel(i, j, color);
    continue;
color = palette[Ro_1 % 32];
    Viewport.SetPixel(i, j, color);

```

Tras la implementación del código en el programa, se obtuvo la simulación del texturizado de la superficie de roca, adicionalmente se presenta la tabla de valores con las variables  $i$ ,  $j$  y  $Ro_1$ , así se puede observar no solo el resultado de la simulación, sino también los valores que toman los píxeles, dependiendo de la posición en la pantalla gráfica. En la Tabla 2 se observa una pequeña muestra de 100 valores, que toman los píxeles, para luego ser pintados por la paleta de colores anteriormente planteada.

El resultado de esta simulación se la puede visualizar a continuación.



**Ilustración 4-1:** Roca 1

Tabla 2: Valores obtenidos de la simulación Roca 1

$i$	$j$	$Ro_1$	$i$	$j$	$Ro_1$	$i$	$j$	$Ro_1$	$i$	$j$	$Ro_1$	$i$	$j$	$Ro_1$
0	0	0	0	20	4	0	40	19	0	60	2	0	80	13
0	1	0	0	21	31	0	41	25	0	61	17	0	81	3
0	2	0	0	22	27	0	42	0	0	62	0	0	82	25
0	3	0	0	23	24	0	43	7	0	63	15	0	83	16
0	4	4	0	24	21	0	44	15	0	64	31	0	84	7
0	5	31	0	25	18	0	45	23	0	65	15	0	85	30
0	6	17	0	26	17	0	46	31	0	66	0	0	86	21
0	7	2	0	27	16	0	47	8	0	67	17	0	87	13
0	8	20	0	28	15	0	48	17	0	68	2	0	88	5
0	9	7	0	29	15	0	49	27	0	69	20	0	89	30
0	10	26	0	30	16	0	50	5	0	70	6	0	90	23
0	11	14	0	31	17	0	51	16	0	71	24	0	91	16
0	12	3	0	32	19	0	52	27	0	72	11	0	92	9
0	13	24	0	33	21	0	53	7	0	73	30	0	93	3
0	14	15	0	34	24	0	54	18	0	74	18	0	94	29
0	15	6	0	35	27	0	55	31	0	75	5	0	95	23
0	16	30	0	36	31	0	56	11	0	76	26	0	96	18
0	17	22	0	37	3	0	57	25	0	77	14	0	97	13
0	18	15	0	38	8	0	58	6	0	78	3	0	98	8
0	19	9	0	39	13	0	59	20	0	79	24	0	99	4

#### 4.1.5. Modelo de textura de agua

Por último, usamos este código del modelo matemático para simular la textura de agua, la cual será pintada manejando la paleta de colores en RGB comprendida desde (2, 151, 151) hasta (99, 238, 238) en tonos de azul celeste, la cual empleando el bucle del código podrá colorear la pantalla gráfica con la simulación.

Así, empleado este código dentro del modelo y ejecutando las ecuaciones halladas para la simulación podemos generar la textura de agua, dando prioridad a la presencia de ondas de diferentes amplitudes las cuales generen una sensación de movimiento en la superficie. La ecuación hallada para este modelo es la siguiente.

$$Ag_1 = \sqrt{250 * j * j - Tan(200 * i) * j * j + Sen(100 * i) - Log_{10}(150 * j) + 150}.$$

Cabe señalar que para este modelo Agua 1, se utilizó una ecuación, esto se debe a la decisión de repetir la onda planteada en diferentes posiciones dentro de la pantalla gráfica, así se obtendrá uniformidad para el desarrollo de la superficie, y aplicando los diferentes tonos de colores en los pixeles, genero una sensación visual de marea en el resultado, finalmente, aplicamos el siguiente código llamado Listing 5, el cual presentó la siguiente simulación.

Listing 5: Código en C# para la simulación de superficie de Agua 1.

```

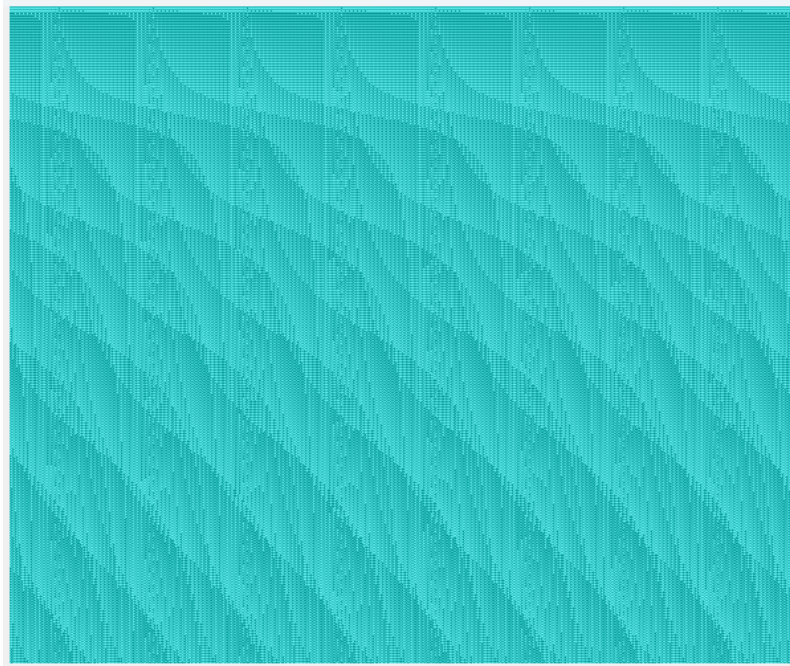
Color[] palette = new Color[32];
int modFactor = palette.Length;
Color color;
int r, g, b, Ag_1;

for (int i = 0; i < 32; i++)
    r = (int)(-i * (48.5 / 16) + 99);
    g = (int)(-i * (43.5 / 16) + 238);
    b = (int)(-i * (43.5 / 16) + 238);
    palette[i] = Color.FromArgb(r, g, b);

for (int i = 0; i < pbViewPort.Width; i++)
for (int j = 0; j < pbViewPort.Height; j++)
    if (i <= 20 || (i > 20 && i < 700))
Ag_1 = (int)(Math.Sqrt(250*j*j - Math.Tan(200*i)*j*j +
    Math.Sin(100 * i) - Math.Log10(150*j)+150)) % 32;
color = palette[Ag_1];
    ViewPort.SetPixel(i, j, color);
    continue;

pbViewPort.Image = ViewPort;

```



**Ilustración 4-1::** Agua 1

Adicionalmente se muestra una tabla de valores (Tabla 3), con las variables  $i$ ,  $j$  y  $Ag\_1$ , en la cual se podrán observar los valores que toman los pixeles dentro de la pantalla gráfica, para luego ser coloreados por la paleta anteriormente planteada para este modelo de textura de agua. Esta tabla de valores contiene los primeros 100 valores de la variable  $Ag\_1$ , estos valores servirán como una muestra de todos los posibles valores que se obtienen dentro de la simulación en la pantalla gráfica.

Tabla 3: Valores obtenidos de la simulación Agua 1

<i>i</i>	<i>j</i>	Ag <sub>1</sub>	<i>i</i>	<i>j</i>	valor	<i>i</i>	<i>j</i>	Ag <sub>1</sub>	<i>i</i>	<i>j</i>	Ag <sub>1</sub>	<i>i</i>	<i>j</i>	Ag <sub>1</sub>
0	0	0	0	10	30	0	20	28	0	30	26	0	40	24
0	1	19	0	11	14	0	21	12	0	31	10	0	41	8
0	2	1	0	12	30	0	22	28	0	32	26	0	42	24
0	3	16	0	13	13	0	23	11	0	33	9	0	43	7
0	4	0	0	14	29	0	24	27	0	34	25	0	44	23
0	5	15	0	15	13	0	25	11	0	35	9	0	45	7
0	6	31	0	16	29	0	26	27	0	36	25	0	46	23
0	7	15	0	17	13	0	27	11	0	37	9	0	47	7
0	8	31	0	18	28	0	28	26	0	38	24	0	48	23
0	9	14	0	19	12	0	29	10	0	39	8	0	49	6

<i>i</i>	<i>j</i>	Ag <sub>1</sub>	<i>i</i>	<i>j</i>	Ag <sub>1</sub>	<i>i</i>	<i>j</i>	Ag <sub>1</sub>	<i>i</i>	<i>j</i>	Ag <sub>1</sub>	<i>i</i>	<i>j</i>	Ag <sub>1</sub>
0	50	22	0	60	20	0	70	18	0	80	16	0	90	15
0	51	6	0	61	4	0	71	2	0	81	0	0	91	30
0	52	22	0	62	20	0	72	18	0	82	16	0	92	14
0	53	6	0	63	4	0	73	2	0	83	0	0	93	30
0	54	21	0	64	20	0	74	18	0	84	16	0	94	14
0	55	5	0	65	3	0	75	1	0	85	0	0	95	30
0	56	21	0	66	19	0	76	17	0	86	15	0	96	13
0	57	5	0	67	3	0	77	1	0	87	31	0	97	29
0	58	21	0	68	19	0	78	17	0	88	15	0	98	13
0	59	4	0	69	3	0	79	1	0	89	31	0	99	29

Para mejorar la experiencia al correr el código del modelo matemático dentro del programa Visual Studio, se decidió agregar 2 botones, los cuales proporcionarán una interacción más amigable con el usuario. El primer botón se lo nombró Descargar, este botón posee un código que pueda capturar la imagen resultante de la simulación, con las mismas dimensiones establecidas anteriormente en la pantalla gráfica con el Picture Box. Al momento de activar este botón, aparecerá una ventana en donde se escogerá el destino de descarga, la opción de nombrar la imagen y el tipo de archivo con el que se desea guardar la imagen resultante de la simulación.

Listing 6: Código en C# para descarga la simulación de superficie en una imagen.

```
if (pbViewPort.Image != null)
```

```

// Abre un di\’alogo para guardar archivo
SaveFileDialog guardarDialogo = new SaveFileDialog();
guardarDialogo.Filter = "Archivos de imagen (*.png)|*.png|
Todos los archivos (*.*)|*.*";
guardarDialogo.FilterIndex = 1;
guardarDialogo.RestoreDirectory = true;

if (guardarDialogo.ShowDialog() == DialogResult.OK)
// Guarda la imagen en el formato seleccionado
pbViewPort.Image.Save(guardarDialogo.FileName, System.Drawing.
Imaging.ImageFormat.Png);
MessageBox.Show("La imagen se ha guardado correctamente.",
"\’Exito", MessageBoxButtons.OK, MessageBoxIcon.Information);

else
MessageBox.Show("No hay una imagen para guardar.", "Advertencia",
MessageBoxButtons.OK, MessageBoxIcon.Warning);

```

El segundo botón agregado se llamó Apagar, este botón tendrá la tarea de desactivar la simulación; generará una pantalla blanca, la cual detendrá la simulación, hasta que el usuario decida iniciar otra simulación activando otros modelos establecidos dentro de la simulación.

Listing 7: Código en C# para apagar la simulación de superficie.

```

using (Graphics g = Graphics.FromImage(ViewPort))

g.Clear(Color.White);
pbViewPort.Image = ViewPort;

```

Utilizando el modelo para generar las texturas de las superficies propuestas, se ha decidido probar más simulaciones, pues la cantidad de diseños que se pueden realizar para crear estas superficies son prácticamente infinitas, debido a la enorme gama de combinaciones que se pueden aplicar al plantear las ecuaciones, ya sea para dar diferentes formas o variar en la mezcla de colores seleccionados, y obtener nuevas texturas de dichas superficies, por lo tanto, se plantean varios ejemplos de modelos matemáticos para simular nuevas texturas, las cuales presentaron diversos resultados muy inusuales.

#### 4.1.6. Otros modelos de textura de madera

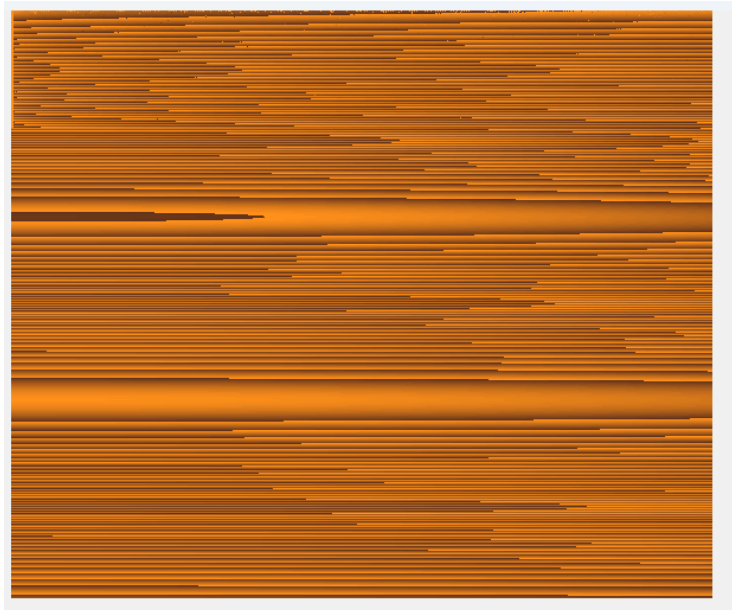
##### Madera 2

El modelo planteado a continuación se lo llamó Madera 2, este modelo matemático contiene una paleta de colores de 32 tonos de café claro, los cuales comprenden desde el color (81, 26, 0) hasta (255, 144, 26) en RGB. Para este caso, se emplearon 3 ecuaciones, las cuales principalmente están contenidas dentro de una raíz cuadrada, con el fin de reducir acortar los resultados para colorear los píxeles.

$$Ma_2 = 1.03 * \sqrt{j^3 - \text{Tan}(-100 + 2 * i - \text{Tan}(5 * i * j))} + i / \sqrt{\frac{i}{2}}.$$
$$Ma_2 = \sqrt{\frac{25 * i^2}{12} + 25 * (j + 100)^3}.$$
$$Ma_2 = \frac{\sqrt{\text{Tan}((j + 100) * (i * j + 150)) * i + 500}}{76}.$$

Estas ecuaciones nuevamente están ubicadas dentro de un solo intervalo, el cual tiene el tamaño de la pantalla gráfica, así garantizamos una mezcla de colores con armonía y que sea agradable a la vista.

Además, se puede observar el uso de las funciones trigonométricas como la función tangente, la cual nos garantiza la presencia de curvas y ondas de diferentes amplitudes dependiendo de la posición que se encuentra dentro de la pantalla, y así se obtiene una simulación de la superficie de madera que presenta betas de diferentes tamaños. Por lo que tenemos el siguiente resultado.



**Ilustración 4-1:** Madera 2



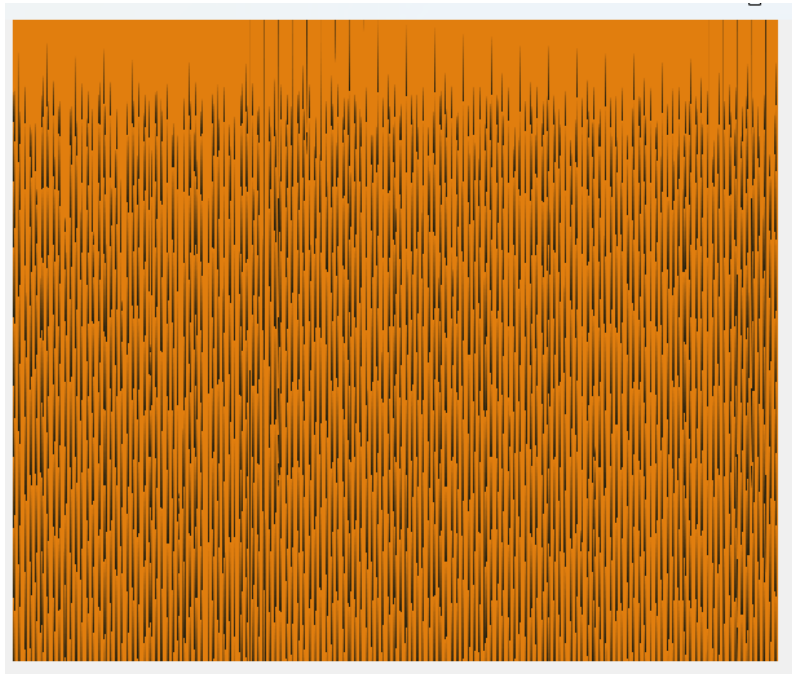
### Madera 3

En otro modelo propuesto para simular textura de madera, se le realizó un cambio su formulación, el cual consiste en utilizar una sola ecuación dentro del código planteado, el objetivo es hallar una textura homogénea de madera, la cual repita su patrón, es decir, el diseño esperado debe presentar las mismas formas y figuras en toda la imagen simulada, además, debe inducir una ilusión de naturalidad, la cual produzca una sensación de realidad al observarla.

Este modelo comprende de los colores en tonos café, desde (33, 18, 2) hasta (225, 126, 14) los cuales se emplearán dentro de la paleta de colores en RGB. Aquí es donde entra el cambio de los anteriores modelos para simular la textura de madera, en este caso se usará solo una ecuación, esta comprende de funciones sinusoidales, las cuales reproducirán un patrón por toda la imagen, por último, el resultado de la ecuación será reducido por una raíz cuadrada, con el fin de obtener una imagen con textura homogénea.

$$Ma_3 = \left( \sqrt{\cos\left(\frac{10 \cdot i}{7}\right) \cdot (\cos(j^2) + 2 \cdot j)^2 / 12 + (-10 - \tan(2 \cdot i))^3 - \frac{8}{7} \cdot j + \sin\left(\pi \cdot \frac{i}{5}\right)} \right)$$

El resultado de la simulación se lo llamo Madera 3, y es una superficie homogénea de textura de madera planteada con una sola ecuación la cual produce betas de madera orientadas de abajo hacia arriba.



**Ilustración 4-1:** Madera 3

#### **Madera 4**

Por último, el modelo planteado a continuación tendrá el mismo cambio con las ecuaciones, nuevamente se busca generar una simulación con un resultado uniforme y homogéneo, esta imagen debe presentar una texturización basada en betas de madera, el objetivo es que la orientación de las betas sea de una sola dirección, por lo que las formas y figuras generadas deben tener el mismo comportamiento, y presentar una simulación similar a una superficie real de madera.

Para este modelo se planteó una paleta de colores con diferentes tonos de café oscuro, comprendidos desde (90, 42, 4) hasta (35, 17, 2). Nuevamente el cambio que se realizó con este modelo es la manipulación de una sola ecuación la cual dará valores a los píxeles para ser pintados por el modelo de la paleta de colores en RGB. Esta ecuación busca generar betas de madera orientadas en una sola dirección, además la manipulación de ecuaciones sinusoidales dentro de la ecuación principal, produce curvas con ciertos patrones distinguibles a la vista, que son relacionados a una superficie de madera.

$$Ma_4 = \sqrt{(\tan(300 \cdot j^2) \cdot i^2 + 120) + 1500}.$$

La imagen obtenida de la manipulación del modelo se la denomino Madera 4, esta superficie es homogénea, y con detalles en las betas de madera, las cuales en este caso están orientadas de izquierda a derecha.



**Ilustración 4-1:: Madera 4**

#### 4.1.7. Otros modelos de textura de roca

##### Roca 2

Para la siguiente simulación se usaron 2 ecuaciones, con el fin de obtener 2 diferentes texturas de roca dentro de la imagen resultante, es decir, el proceso consistirá en combinar los resultados obtenidos en intervalos para obtener una amalgama de texturas rocosas en la simulación, el motivo radica en darle importancia a la característica principal de esta superficie, la aleatoriedad. Así las ecuaciones que generen formas aleatorias rocosas serán presentadas dentro de los intervalos que dividen a la imagen y, combinadas con la paleta de colores, producirán una mezcla natural de la textura dentro de la pantalla gráfica, garantizando el montaje de la simulación.

La paleta de colores en RGB está comprendida por diferentes tonos de gris, desde (49, 52, 54) hasta (101, 101, 102). Esta paleta presenta un pequeño arreglo, el cual será implementado en el código Random. Este código servirá para generar aleatoriedad a las constantes dentro de la ecuación principal del modelo, y al momento en que el programa utilice esta ecuación al simular la textura, reconocerá la función *Random* dentro de la misma, por lo que el resultado de la simulación será variado.

Listing 8: Código en C# para invocar la función *Random* en el código de la ecuación

```
Random random = new Random ();
```

Para la formulación de las ecuaciones se tomará en cuenta el uso de intervalos, los cuales dividirán en 4 partes la pantalla gráfica. En el primer intervalo se utilizó la primera ecuación y se estableció como referencia a la variable  $j$ , la cual comprenderá para  $j$  menor o igual a 75, y  $j$  mayor o igual a 100 hasta  $j$  menor o igual a 600, mientras que la segunda ecuación estará en el intervalo desde  $j$  mayor a 75 hasta 100, y  $j$  mayor a 600 y menor a 1000.

Estas ecuaciones tienen funciones sinusoidales como son las tangentes, y funciones polinómicas elevadas a segundo y tercer grado. Aquí es donde entra el uso de la función *Random*, la cual manipuló pequeños intervalos establecidos para las constantes que acompañan a las variables  $j$  e  $i$ . Por último, se le añadió el arreglo de la raíz cuadrada en las ecuaciones, la cual nuevamente, acertó los valores obtenidos por el modelo, y esto permite darle más cercanía a la distribución de colores de los píxeles dentro de la pantalla gráfica.

$$Ro_2 = \sqrt{\text{random}(45, 50) \cdot j^3 - \tan((-100 - 2 \cdot i) - \tan(5 \cdot i \cdot j)) + i^2 \cdot j / \sqrt{i \cdot j / 2}}$$
$$Ro_2 = \sqrt{\left( \frac{(100 \cdot i^4) \cdot 25}{\text{random}(9, 12)} + ((j + 100)^3 \cdot 25) \right) \cdot 1.03}$$

El resultado que se observa de la simulación es parcial, pues el uso de la función *Random* dentro de

la ecuación principal genera cambios cada vez que se invoca la imagen dentro del programa Visual Studio. A esta imagen de la simulación se la denominó Roca 2.



**Ilustración 4-1::** Roca 2

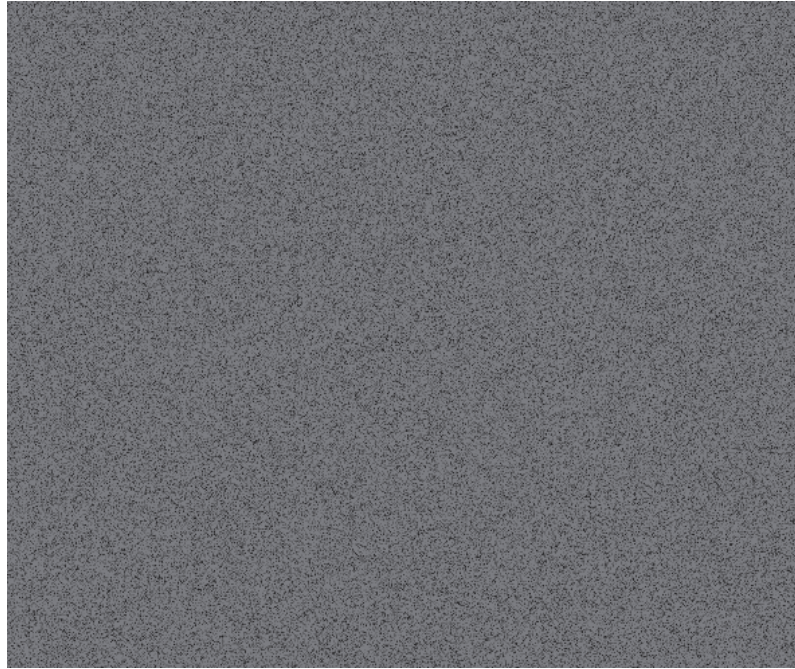
### **Roca 3**

En este nuevo modelo se utilizó nuevamente la herramienta de la función *Random* para obtener la aleatoriedad de la textura rocosa en la simulación, pero la diferencia será en la no manipulación del modelo utilizando diferentes intervalos y ecuaciones dentro de la imagen; para este modelo se intentó generar la textura de roca empleando una sola ecuación, así el resultado generado será homogéneo, pero como se mantendrá la función *Random*, continuará la característica principal de los relieves rocosos.

Al emplear la paleta de colores en este modelo, también se colocó el código Listing 8, esta simulación nuevamente usó la función *Random*; para la paleta de colores en RGB, se usaron los tonos en gris claro que comprenden desde (33, 32, 33) hasta (124, 125, 130). Por último, en la formulación de la ecuación empleada en el modelo, se utilizó una combinación de funciones sinusoidales dentro de un binomio al cuadrado, y éste fue operado en función de una tangente con las variables  $i, j$ . Además, ingresamos una función *Random* que manipula un intervalo de constantes operadas dentro de una función seno; el propósito de esta composición es dar una distribución casual al momento de obtener valores para los píxeles dentro de la pantalla gráfica. Finalmente le damos el arreglo de la raíz cuadrada, acortando los valores y reduciendo el tamaño de la distribución que escogerá los colores de los píxeles.

$$Ro_3 = \sqrt{\sin(\text{random}(20, 25) \cdot i) + 100 \cdot \tan(\text{random}(10, 14) \cdot j \cdot i - (100 \cdot \cos(i) + \sin(10 \cdot j))^2)}.$$

El resultado obtenido se lo nombró Roca 3, como se observa en la imagen, el modelo de esta superficie mantiene el comportamiento *Random*, y adicionalmente esta textura es homogénea, pues los patrones presentan composiciones y estructuras iguales, pero al invocarla en el programa, cambia su ubicación y comportamiento manteniendo la aleatoriedad, la característica principal de una textura rocosa.



**Ilustración 4-1::** Roca 3

#### **4.1.8. *Otros modelos de textura de agua***

Para generar otro modelo de textura de agua, se requerirá de la manipulación del código, se necesita una simulación homogénea, por lo que se usó una sola ecuación en el modelo, empleando principalmente funciones sinusoidales, la base de este modelo es generar ondas de diferentes amplitudes y frecuencias, para obtener diferentes longitudes de ondas. Adicionalmente, esta textura debe presentar uniformidad en toda la simulación, además, considerando el movimiento del agua, priorizaremos la orientación, esta debe presentarse hacia una dirección.

La paleta de colores en RGB tomó tonos de color celeste y blanco, los cuales comprenden desde (4, 139, 202) hasta (161, 215, 241), nuevamente estos colores fueron utilizados dentro del modelo de la paleta de colores Listing 2. Por último, en la formulación de las ecuaciones, se emplearon funciones sinusoidales como lo es la función tangente, la cual está manipulando a la variable  $j$ , en las 4 veces que se la nombro, esto garantiza la uniformidad en la textura y orientación. A continuación, se presenta la ecuación principal para generar esta simulación.



$$Ag_2 = \sqrt{400 \cdot j^2 + \left( \tan(300 \cdot j^2) \cdot i^2 + 120 + \tan\left(\frac{500 \cdot j^2}{25}\right) \cdot i^2 - \tan(\tan(300 \cdot j^2) + i^2) \right) + 1500}.$$

El resultado obtenido se la denominó Agua 2, y describiremos los detalles hallados en el resultado, presenta una orientación de izquierda a derecha, también, la mezcla de colores genera una ilusión de movimiento producido por olas, además, la simulación presentó un detalle particular, el cual tiene un efecto visual de profundidad en la imagen, por último, resaltamos otro detalle, la similitud de la simulación a una pintura en lienzo.



**Ilustración 4-1::** Agua 2

## CAPÍTULO V

### 5. CONCLUSIONES Y RECOMENDACIONES

El capítulo V consiste en presentar las conclusiones y recomendaciones que se obtuvieron tras finalizar el trabajo de investigación, las conclusiones exponen los resultados del cumplimiento de los objetivos planteados, mientras que las recomendaciones redactan un aporte adicional que se puede considerar al aplicar los resultados obtenidos tras la investigación, mostrando más consejos para mejorar la simulación obtenida y posibles nuevas simulaciones de texturas de superficies.

#### **Conclusiones**

- La propuesta de generar modelos matemáticos para simular texturas de superficies de madera, agua y roca, empelando la herramienta de la computación gráfica, representa un avance significativo en la creación de contenido visual, pues emplear diferentes funciones y ecuaciones adaptadas a cada tipo de superficie nos permite tener una mayor flexibilidad y precisión en la simulación de la textura, con lo cual podemos aportar para el avance de la productividad en diferentes campos dentro de la industria gráfica.
- El proceso de identificación y análisis de los elementos y características básicas; la combinación de colores, formas, texturas y modelos que se requieren en las superficies resulto crucial, pues proporciono un entendimiento del comportamiento visual y estructural, lo que facilitó el proceso de diseño al momento de crear los modelos matemáticos de las superficies de madera, agua y roca.
- Empleando diferentes funciones y ecuaciones, como las polinómicas, logarítmicas y las sinusoidales, se pudo desarrollar modelos matemáticos, capaces de realizar la simulación de texturas, y la utilización de herramientas computacionales en específica la gráfica, nos garantizó la simulación de las texturas parecidas a las superficies planteadas.
- La implementación de modelos matemáticos, en el programa Visual Studio, permitió desarrollar una herramienta versátil para crear modelos que facilitan la simulación de texturas que se asemejan a las superficies reales de madera, agua y roca, y dando apertura a nuevas posibilidades de creación en los diversos campos de la industria.

#### **Recomendaciones**

- El desarrollar variedad de funciones y ecuaciones matemáticas nos abre el campo de experimentación para mejorar el proceso de modelaje y así, obtener resultados más confiables y

precisos en la simulación, presentando una mayor flexibilidad al crear diversas texturas que se encuentran en la naturaleza, por lo que recomendamos se combinen elementos sencillos para conseguir objetos únicos.

- Es importante continuar con el análisis y la investigación de las características básicas de las superficies, para tener más herramientas al momento de comprender el comportamiento visual y estructura de la superficie, y mejorar los modelos matemáticos planteados que servirán para mejorar los productos en la industria.
- Mantenerse al tanto de los avances tecnológicos relacionados a la simulación de las texturas de superficies, como técnicas de renderización para mejorar los colores desvanecidos, o el uso de técnicas de escaneo en 3D, con los cuales se puede capturar detalles presentes en las superficies naturales, y mejorar la calidad y eficacia de las simulaciones.
- Documentar los procesos de desarrollo y los resultados obtenidos de la simulación, pues facilitará la revisión, la replicación y la retroalimentación por parte de los lectores, promoviendo el avance y la difusión de esta área de la computación gráfica, que servirá de base para el desarrollo de otras aplicaciones.
- Este modelo matemático puede ser utilizado para generar texturas de otras superficies, como pueden ser de tela o de baldosa, ya que el modelo brinda una comodidad al momento de plantear patrones que se repitan en toda la pantalla gráfica, por lo cual, estas nuevas texturas se pueden obtener sin problemas.



## BIBLIOGRAFÍA



1. **ÁLVAREZ, A. Et al**, Mathematical algorithms for the texture and lighting of 3D surfaces in a computer. *KnE engineering* [en línea], 2018, (Ecuador), págs. 255â269. [Consulta: 10 de mayo del 2023]. Disponible en: <https://knepublishing.com/index.php/KnE-Engineering/article/view/3659/7638toc>.
2. **BERTI, M.**, Detección de Fallas en soldaduras utilizando análisis de textura en imágenes radiográficas, Universidad de Santiago de Chile. Facultad de Ingeniería. Chile. 2003.
3. **CAYADO, L. Et al**. La simulación por ordenador como herramienta de aprendizaje de procesos reales en prácticas de planta piloto. *Dialnet* [en línea], 2014, págs. 261-267. [Consulta: 13 de mayo del 2023]. Disponible en: <https://dialnet.unirioja.es/servlet/articulo?codigo=6441091>.
4. **EFECTOLED**, Qué significa RGB - Blog efectoLED.com. efectoLED Blog [en línea], 2023. [Consulta: 20 de diciembre del 2023] Disponible en: <https://www.efectoled.com/blog/es/que-significa-rgb/:text=Cuando>
5. **GLYN, J.** “Matemáticas Avanzadas para Ingeniería”, Pearson, Madrid. 2012.
6. **ADMIN**, Historia, objetivos y aplicaciones de la computación gráfica. Ilet Toluca [en línea], 2021. Disponible en: <https://ilet.mx/toluca/concepto-computacion-grafica/>.
7. **OLAYA, V.**, Sistemas de Información Geográfica. S.l.: *GitHub Inc* [en línea] 2014. [Consulta: 20 de mayo del 2023]. Disponible en: <https://volaya.github.io/libro-sig/>
8. **PETRIE G. & KENNIE T. J.** Terrain Modelling in Surveying and Civil Engineering. Glasgow U. K.: McGraw-Hill. 2014.
9. **PINTO A.** Segmentación de imágenes por textura. Informe de Memoria de Título. Universidad de Concepción, Facultad de Ingeniería, Departamento de Ingeniería Eléctrica. Chile. 2006. págs. 1 - 4.
10. **TOKIO R.**, . 5 tipos de modelado 3D que debes conocer. Tokio School [en línea], 2023, Disponible en: <https://www.tokioschool.com/noticias/tipos-modelado-3d/>.
11. **URRUTIA, D.** *Qué es Visual Studio. Definición, historia y ventajas.* [blog]. Arimetrics, 2023. [Consulta: 03 de enero del 2024]. Disponible en: <https://www.arimetrics.com/glosario-digital/visual-studio>.
12. **VILLA, R.** Programación en Visual Studio.NET bajo C# de Aplicaciones Gráficas. Ingeniero Técnico en Informática de Sistemas.Universidad de Málaga. Departamento de Lenguas y Ciencias de la Computación. [ En línea], 2002, España, pág. 14. [Consulta: 03 de enero del 2024]. Disponible en: <http://info.iaia.lcc.uma.es/lcc/pfc/37.pdf>





**ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO**  
**CERTIFICADO DE CUMPLIMIENTO DE LA GUÍA PARA**  
**NORMALIZACIÓN DE TRABAJOS DE FIN DE GRADO**

**Fecha de entrega:** 19/07/2024

<b>INFORMACIÓN DEL AUTOR</b>
<b>Nombres – Apellidos:</b> José Andrés Cifuentes Flores
<b>INFORMACIÓN INSTITUCIONAL</b>
<b>Facultad:</b> Ciencias
<b>Carrera:</b> Matemática
<b>Título a optar:</b> Matemático
 <b>Dr. Alonso Washington Alvarez Olivo</b> <b>Director del Trabajo de Integración Curricular</b>  <b>Dr. Luis Marcelo Cortez Bonilla</b> <b>Asesor del Trabajo de Integración Curricular</b>