



**FACULTAD DE INGENIERÍA Y CIENCIAS AGROPECUARIAS
ESCUELA DE INGENIERÍA DE SONIDO Y ACÚSTICA**

**SISTEMA DE GENERACIÓN DE IMÁGENES Y PATRONES VISUALES A
PARTIR DE UNA SEÑAL DE AUDIO; CONSIDERANDO AMPLITUD,
FRECUENCIA Y ESPECTRO, DESTINADO PRINCIPALMENTE A
REFUERZO SONORO**

Autor:

Hugo Fernando Jácome Andrade

Año

2009



FACULTAD DE INGENIERÍA Y CIENCIAS AGROPECUARIAS
ESCUELA DE INGENIERÍA DE SONIDO Y ACÚSTICA

**SISTEMA DE GENERACIÓN DE IMÁGENES Y PATRONES VISUALES A
PARTIR DE UNA SEÑAL DE AUDIO; CONSIDERANDO AMPLITUD,
FRECUENCIA Y ESPECTRO, DESTINADO PRINCIPALMENTE A
REFUERZO SONORO**

**Trabajo de Titulación presentado en conformidad a los requisitos
establecidos para optar por el título de:**

Ingeniero de Sonido y Acústica

Profesor Guía:

Ing. Marcelo Lazzati

Autor:

Hugo Fernando Jácome Andrade

Año

2009

DECLARACIÓN DEL PROFESOR GUÍA

“Declaro haber dirigido este trabajo a través de reuniones periódicas con el estudiante, orientando sus conocimientos para un adecuado desarrollo del tema escogido, y dando cumplimiento a todas las disposiciones vigentes que regulan los Trabajos de Titulación.”

Marcelo Lazzati

Ingeniero en Ejecución de Sonido

CI: 171163573-8

DECLARACIÓN DE AUTORÍA DEL ESTUDIANTE

“Declaro que este trabajo es original, de mi autoría, que se han citado las fuentes correspondientes y que en su ejecución se respetaron las disposiciones legales que protegen los derechos de autor vigentes. Las muestras de audio usadas en este proyecto, *samples* de instrumentos y obras musicales, son estrictamente utilizadas con fines académicos, respetando plenamente los derechos de autor y las leyes vigentes.”

Hugo Fernando Jácome Andrade

CI: 100312035-7

AGRADECIMIENTO

Agradezco de todo corazón a las siguientes personas:

María Cristina Breilh por inculcar el interés en las herramientas informáticas
aquí usadas,

Marcelo Lazzati por la guía para que este trabajo llegue a buen término,

José Rafael Subía por su ayuda y orientación a la distancia,

Mi padre, mi madre, mis hermanas y familiares por sus innumerables
sugerencias y comentarios a lo largo del desarrollo de esta tesis,

Jessie Peñafiel Vallejos por su valiosa ayuda en la parte visual, gráfica y de
color.

Muchas gracias por ser parte de la consecución satisfactoria de este proyecto.

DEDICATORIA

Dedico este trabajo a:

Mi madre Mónica Andrade y mi padre Gustavo Jácome por su amor, entrega, apoyo, dedicación, enseñanzas y valores impartidos, tesoros invaluable que guardaré conmigo el resto de mi vida,

Mis hermanas por todo el amor, y las hermosas vivencias desde siempre,

Mis abuelos maternos Saady y Hugo; paternos Gloria y Fernando (†) que han guiado desde siempre a toda la familia con la sabiduría que sólo el tiempo y el trajinar por la vida pueden brindar,

Jessie Peñafiel Vallejos por todo su amor, comprensión y enseñanzas,

Judith Mendoza y Rubén Montenegro por acogerme en su hogar al estar yo lejos del mío,

Mi familia, mi extensa, querida y bella familia.

A todos dedicado, porque han sabido apoyarme, comprenderme y llenar mi vida de felicidad.

RESUMEN

El presente trabajo plantea la necesidad de implementar una herramienta tipo *software* que construya imágenes a partir de elementos del sonido.

Se usa a lo largo de este proceso, el entorno o lenguaje de programación Max/MSP/Jitter, nacido con el nombre "Patcher" bajo la mano de Miller S. Puckette y actualmente propiedad de la compañía Cycling '74.

Se relaciona los distintos elementos mediante entradas de audio digital, analizadores de audio en tiempo real y generadores de figuras geométricas, con el objetivo de construir a la salida una sola imagen que corresponda en tiempo real al modelo de trabajo del *software*.

Los resultados o salidas del programa, concuerdan con los objetivos al relacionar los parámetros de amplitud del sonido con tamaño de la imagen, frecuencia del sonido con el color de la imagen y el espectro sonoro con la textura de la imagen.

Además, se incluye la construcción de una interfaz de usuario, en ella se puede activar la entrada de audio, con lo cual, automáticamente se enciende un analizador básico de audio que proporciona datos de amplitud, frecuencia, forma de onda, espectro, y fase de la señal de entrada; incluyendo control de origen de la señal (archivo, entrada de línea o micrófono), y el nivel con que ingresa, logrado mediante *faders* y controles similares a los físicos.

De igual forma, existen controles para el generador de imágenes y figuras geométricas, con opciones de encendido/apagado, tipo de figura geométrica, tipo de unión entre los puntos de su estructura, colores de fondo y opciones de composición de nuevas imágenes mediante superposición de figuras.

Este trabajo permite llegar a establecer una relación entre elementos sonoros y visuales por medio de un modelo planteado y mostrarlos mediante una aplicación *software* que cumple con los objetivos del proyecto.

ABSTRACT

The present work considers the necessity to implement a software tool that constructs images from elements of the sound.

It will be used throughout this process, the Max/MSP/Jitter programming language, been born with the name "Patcher" under the hand of Miller S. Puckette and at the moment property of the company Cycling '74.

The different elements are related using digital audio entrances, real time audio analyzers and geometric figures generators, with the aim of constructing at the exit of the program, a single image that corresponds in real time to the working model of the software.

The results or exits of the program, agree with the objectives when relating the parameters of amplitude of the sound to size of the image, acoustic frequency with the color of the image and the sonorous spectrum with the texture of the image.

In addition, the construction of an user interface is included, in it the entrance of audio can be activated, consequently, automatically an audio basic analyzer activates; which provides amplitude data, frequency, waveform, spectrum, and phase of the entrance signal; including options to choose the origin of the signal (computer file, entrance of line or microphone), and the level with which it enters, obtained by using faders and controls similar to the real ones.

In the same way, they are controls for the geometric figures generator, with options of turning on/off, type of geometric figure, type of union between the points of its structure, bottom colors and options of composition of new images by superposition of figures.

This work allows to establish a relation between sonorous and visual elements by using of a raised model and to show them by a software application that fulfills the objectives of the project.

INDICE

Contenido	Página
1. INTRODUCCIÓN.....	14
1.1 Descripción del problema.....	14
1.2 Antecedentes.....	15
1.3 Justificación.....	15
1.4 Objetivo General.....	16
1.5 Objetivos Específicos.....	16
1.6 Hipótesis.....	16
2. ANÁLISIS BÁSICO DE SEÑALES.....	17
2.1 Repaso histórico.....	17
2.1.1 Orígenes del análisis de señales continuas.....	17
2.1.2 Orígenes del análisis de señales discretas.....	18
2.2 Conversión del sonido físico al mundo digital.....	19
2.3 Parámetros principales de una señal.....	23
2.3.1 Amplitud.....	23
2.3.2 Frecuencia.....	25
2.3.3 Espectro.....	26
2.4 La audición humana.....	30
2.5 Percepciones psicoacústicas de los distintos parámetros físicos del sonido.....	36
2.5.1 Intensidad.....	36
2.5.2 Altura tonal.....	37
2.5.3 Timbre.....	38
2.6 La visión humana.....	39

2.7 Modelos de composición del color.....	40
2.7.1 Modelo de color RGB.....	41
3. PROGRAMACIÓN.....	42
3.1 Breve historia del ordenador o computador.....	42
3.2 Algunos lenguajes de programación.....	44
3.3 Software Max/MSP/Jitter.....	44
4. DESARROLLO, PRUEBAS Y ANÁLISIS DE RESULTADOS.....	47
4.1 Analogías y asociaciones entre audio e imagen.....	47
4.1.1 Modelo de implementación del programa.....	49
4.2 Proceso de construcción y estructuración del programa.....	50
4.2.3 Etapa de análisis de amplitud, frecuencia y espectro....	52
4.2.3.1 Herramientas usadas.....	52
4.2.3.2 Proceso de construcción del <i>sub-patch</i>	56
4.2.4 Etapa de generación de partículas o figuras geométricas.....	60
4.2.4.1 Herramientas usadas.....	60
4.2.4.2 Proceso de construcción del <i>sub-patch</i>	67
4.2.5 Conexión del analizador de audio con el generador de imágenes.....	76
4.2.5.1 Herramientas usadas.....	76
4.2.5.2 Proceso de construcción del <i>sub-patch</i>	76
4.2.6 Interfaz de usuario.....	79
4.2.6.1 Herramientas usadas.....	79
4.2.6.2 Proceso de construcción del <i>patch</i>	79
4.3 Pruebas y análisis de resultados.....	82

4.3.1 Amplitud vs. Tamaño.....	83
4.3.2 Frecuencia vs. Color.....	85
4.3.3 Espectro vs. Textura.....	88
4.4 Diferentes figuras geométricas y sus principales modos de construcción.....	91
4.4.1 Esferas (sphere).....	91
4.4.2 Toroide (torus).....	94
4.4.3 Cilindros (cylinder).....	94
4.4.4 Cubo (cube).....	95
4.4.5 Planos cuadrados (plane).....	95
4.5 Composición de nuevas imágenes mediante superposición.....	96
4.6 Imágenes resultantes de timbres definidos.....	97
4.7 Imágenes resultantes de estilos musicales definidos.....	101
5. ESTUDIO ECONÓMICO.....	106
5.1 Estudio de costos.....	106
5.2 Relación costo beneficio.....	107
6. PROYECCIONES Y POSIBILIDADES DE EXPANSIÓN.....	109
6.1 Proyecciones.....	109
6.2 Posibilidades de expansión.....	109
7. CONCLUSIONES Y RECOMENDACIONES.....	111
7.1 Conclusión General.....	111
7.2 Conclusiones Específicas.....	111

7.3 Conclusiones técnicas y	
de desempeño del programa.....	112
7.3.1 Conclusiones Específicas.....	112
7.3.1.1 Amplitud vs. Tamaño.....	112
7.3.1.2 Frecuencia vs. Color.....	113
7.3.1.3 Espectro vs. Textura.....	114
7.3.1.4 Imágenes resultantes	
de timbres definidos.....	114
7.3.1.5 Imágenes resultantes	
de estilos musicales definidos.....	114
7.3.2 Conclusiones generales	
de desempeño del programa.....	115
7.4 Conclusiones económicas	
y de relación costo/beneficio.....	116
7.5 Conclusiones de proyecciones	
y posibilidades de expansión.....	116
7.6 Recomendaciones.....	117
7.6.1 Recomendaciones Generales.....	117
7.6.2 Recomendaciones técnicas	
y de operación del programa.....	118
8. BIBLIOGRAFÍA.....	120
8.1 Libros.....	120
8.2 Ficheros electrónicos (PDF).....	120
8.3 Artículos de Internet.....	121
9. ANEXOS.....	125
9.1 Imágenes del Desarrollo.....	125

9.2 Imágenes de las Pruebas y Análisis de resultados.....	135
10. GLOSARIO.....	153

CAPÍTULO 1. INTRODUCCIÓN

1.1 Descripción del problema

La necesidad de combinar sonido e imagen para lograr diferentes resultados de comunicación en un oyente siempre ha sido de marcado interés, especialmente en el área objeto de estudio de esta tesis, como es el refuerzo sonoro y la producción de audio y eventos. Se puede enriquecer la información que un ser humano recibe cuando se conjuga adecuadamente elementos audiovisuales como un todo, esto nos lleva a una apreciación de la realidad en nuestro país respecto al tema mencionado, no existen esquemas de trabajo completos y con ideas generalizadas acerca de la conjugación de estos elementos de una manera altamente tecnicada, estudiada, razonada, planificada y justificada para lograr resultados diferentes y en ciertos casos más interesantes.

Por la dificultad de acceso y tal vez más preponderantemente por la limitación económica, algunos sistemas que conjugan luz y sonido sencillamente no llegan a nuestro mercado porque una demanda casi nula así lo provoca, y el factor económico lamentablemente puede llegar a entorpecer la producción de un programa de buenas condiciones. Es por esto que los conciertos, eventos con música en vivo, recitales, etc. que se hacen en nuestro país no han alcanzado a gozar de un adecuado estándar de calidad (tomando como referencia lo que en otros lugares se realiza) y, por lo tanto no provocan en el oyente aquel estado de máxima recepción de información.

Por lo tanto el pensar en acceder a este tipo de dispositivos por el momento es un tanto dificultoso, este problema plantea la necesidad de crear un sistema que tenga prestaciones útiles en la generación de imágenes a partir de parámetros de una onda eléctrica de audio que presente facilidades de acceso e implementación.

1.2 Antecedentes

Gracias al surgimiento de la era digital, la masificación y sofisticación de distintos dispositivos ha sido posible realizar análisis más minuciosos de señales, descomponerlas en sus distintos parámetros, analizarlos, implementarlos en otra información y sintetizarlos.

El tema de la generación de patrones visuales a partir del sonido es relativamente reciente. No existe abundante información o fuentes de consulta para la generación de patrones visuales a partir de una señal de audio (objeto de este proyecto) debido a que sus utilidades no han sido explotadas en su totalidad ni existen muchas facilidades de acceso a este tipo de dispositivos que dicho sea de paso existen, principalmente en el campo del refuerzo sonoro. Se han visto por ejemplo utilidades de estos sistemas en lo que respecta a conciertos donde la conjugación entre luces, imagen y sonido bajo ciertas condiciones crea ambientes y estados que logran en el público un impacto relevante, con programaciones que están a la vanguardia de la tecnología y no sólo constituyen un evento de índole musical, sino un conjunto de elementos audiovisuales que hacen de una aquello una experiencia inolvidable, jugando mucho con la psicoacústica y la forma en que reacciona el cuerpo humano a los diferentes patrones de imágenes y elementos visuales.

1.3 Justificación

Se puede aportar a la calidad del mensaje que recibe el oyente, con la conjugación de diferentes elementos. Al ser una herramienta diseñada para fácil acceso y relativamente bajo costo los usuarios pueden empezar a usar este sistema con el propósito de enriquecer algunos elementos sonoros con los visuales en un público que favorablemente en la actualidad está empezando a ser cada vez más exigente respecto a contenidos y elementos de juicio dentro de una programación, independientemente de la índole de ésta.

Se justifica entonces este proyecto por la necesidad de incrementar las herramientas para lograr una mejor transmisión de información en el público, por el enriquecimiento con elementos visuales de las experiencias sonoras y por la facilidad de acceso e implementación que el sistema posee para nuestro medio.

1.4 Objetivo General

Desarrollar un *software* que provea la capacidad de generar imágenes a partir de una única señal de audio, guardando relación de coherencia entre los elementos comunes de ambos espectros.

1.5 Objetivos Específicos

- Encontrar similitudes entre los parámetros de amplitud, frecuencia y espectro tanto de luz visible como de sonido que permitan desarrollar el proyecto.
- Formar patrones o elementos comunes a ambas partes (luz visible y sonido) que obedezcan a un modelo de trabajo planteado.
- Generar una aplicación capaz de manejar esa relación teniendo como interfaz de entrada los puertos de entrada de audio comunes de una computadora.

1.6 Hipótesis

Se plantea como hipótesis principal la factibilidad de realizar un *software* que posea la capacidad de generar una imagen a partir de una onda de energía eléctrica que lleve información de audio, bajo los siguientes parámetros:

- Frecuencia relacionada con el rango de colores de los patrones visuales.
- Espectro o timbre relacionado con la textura de las imágenes.
- Amplitud relacionada con el tamaño de formas o patrones visuales a generar.

CAPÍTULO 2. ANÁLISIS BÁSICO DE SEÑALES

El análisis de señales ha permitido aprovechar y sacar máximo partido a los fenómenos especialmente ondulatorios que rigen nuestro universo. A través de un análisis exhaustivo de una señal es posible obtener sus distintos parámetros, manejarlos, estudiarlos, modificarlos de manera independiente unos de otros y así comprender de mejor manera sus distintos comportamientos.

2.1 Repaso histórico [17]

2.1.1 Orígenes del análisis de señales continuas

Las señales continuas en el tiempo, presentes en la naturaleza son de alta complejidad, es por esto que nació el interés por analizar y estudiar esas señales pero desde el punto de vista de la descomposición de sus elementos, para estudiarlos de manera independiente.

Es a inicios del siglo XIX, cuando se originaron las bases matemáticas para el análisis de señales continuas mediante las Transformadas de Fourier y Laplace.

Fue Jean Baptiste Joseph, Barón de Fourier quien inició una serie de estudios e investigaciones científicas, sobre aspectos termodinámicos y los flujos de calor, los cuales originaron el desarrollo de su teoría de las series, ahora conocidas como Series de Fourier; a pesar de que su aplicación puede ser generalizada a diversos campos de la ciencia, fue en el análisis de señales donde se encontró una utilidad trascendental.

Estas series son muy útiles al momento de analizar señales periódicas (que poseen patrones de repetición uniformes en el tiempo). Sin embargo para señales aperiódicas (que no poseen un patrón de repetición en el tiempo, y hasta son aleatorias), la Serie de Fourier tiene ciertas limitaciones y entra en uso la Transformada o Integral de Fourier para análisis de estas señales.

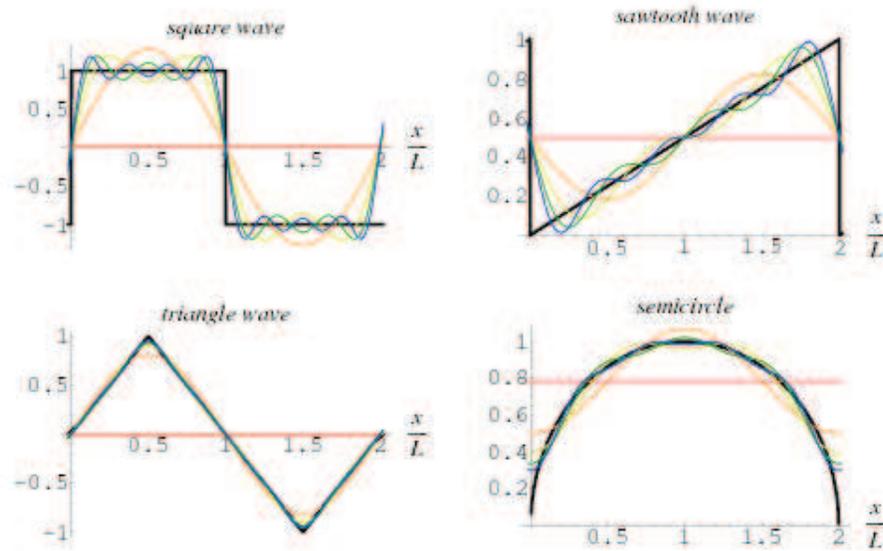


Figura 2.1: Series de Fourier, donde se aprecia la forma como las formas de onda son compuestas por varias componentes senoidales, o que se aproximan a aquella condición [28].

2.1.2 Orígenes del análisis de señales discretas

Tal vez un par de décadas antes de la publicación de los estudios de Fourier, aparece la Transformada de Laplace. Es una herramienta muy útil creada por Pierre Simon, Marqués de Laplace. Sus estudios matemáticos fueron destinados al estudio astronómico de la época, auge de descubrimientos planetarios. Al igual que los descubrimientos matemáticos de Fourier estos avances fueron encontrando, cientos de años después, mucha utilidad especialmente en el campo de las señales digitales.

Estos descubrimientos matemáticos cobran verdadera importancia en el mundo de las señales y su estudio, en el siglo XX cuando aparecen los computadores digitales alrededor de la década del 40, pasando a formar una de las piedras angulares de la construcción de filtros digitales y demás herramientas afines usadas hasta el día de hoy.

Para señales digitales o discretas en el tiempo la Transformada de Fourier también tiene aplicaciones. Como se mencionó anteriormente, con el apareamiento de los computadores digitales se empezó a utilizar la Transformada Discreta de Fourier (DFT por las siglas en inglés de *Discrete Fourier Transform*), pero no por propósitos de audio, sino por aspectos de procesamiento de información computacional.

La Transformada Rápida de Fourier (FFT por las siglas en inglés de *Fast Fourier Transform*) aparece proveniente de un estudio realizado por Cooley y Tukey en 1965, basado en trabajos del matemático alemán Runge que datan de 1903. La FFT salió a la luz como un nuevo algoritmo prometedor que luego supuso una revolución en la computación de la época debido a que reducía aproximadamente en un factor de 100 las multiplicaciones de algunos procesos computacionales (problema también recurrente en procesamiento digital de señales), y requeridas para el cálculo de una Transformada de Fourier.

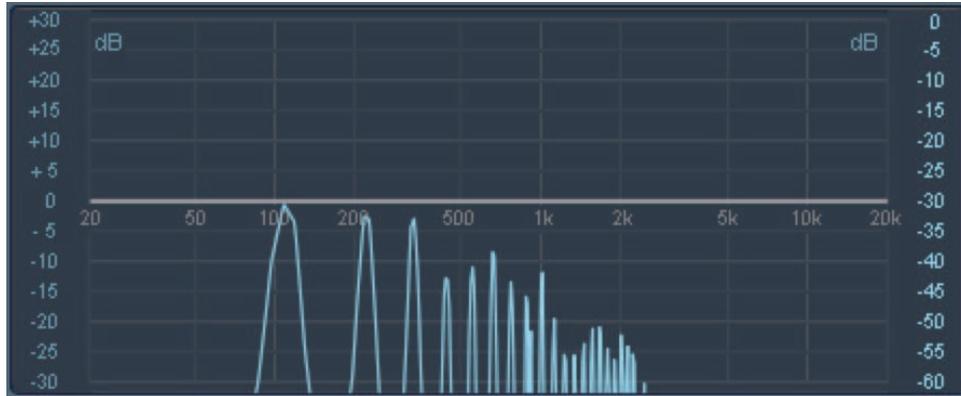


Figura 2.2: Diagrama de la FFT de la señal de audio de un violonchelo, donde se observa las múltiples componentes de frecuencia que puede tener una señal [29].

2.2 Conversión del sonido físico al mundo digital [1], [4]

La digitalización del sonido ha jugado un papel preponderante en las últimas décadas, como elemento fundamental de los nuevos medios de almacenamiento y procesamiento de audio. Una de sus ventajas es la inalterabilidad de los datos, así

como fiabilidad y fidelidad. Con esta serie de avances se logró pasar de procesos que en el mundo analógico resultan demasiado costosos a procedimientos digitales de gran complejidad y bajísimo costo, por lo tanto con fácil acceso a la mayoría del público.

El centro de estas operaciones, es el circuito electrónico, los hay de propósito general y también los más específicos para la tarea de procesamiento de señales, como son los procesadores de señales digitales (DSP por las siglas en inglés de *Digital Signal Processor*).

La principal herramienta matemática de la conversión digital del audio es la numeración binaria donde sólo existen dos dígitos, 0 y 1. Ellos conforman el sistema de numeración que consiste en representar con una secuencia de ceros y unos, a diferentes números, valores o estados que conocemos en otros sistemas numéricos como el decimal, por ejemplo el número decimal 86 se lo puede representar como una secuencia binaria de 8 *bits* (los que conformarían un *byte*), mediante la siguiente operación:

$$\begin{aligned}
 01010110 &= (0 \times 2^7) + (1 \times 2^6) + (0 \times 2^5) + (1 \times 2^4) + (0 \times 2^3) + (1 \times 2^2) + (1 \times 2^1) + (0 \times 2^0) \\
 &= (0) + (64) + (0) + (16) + (0) + (4) + (2) + (0) \\
 &= 86
 \end{aligned}$$

El sistema binario al poseer sólo dos valores es idóneo para su representación eléctrica dentro de un circuito, además de ser tremendamente insensible al ruido, ya que para representar el 1 se usa un voltaje alto (5 V) y para el 0 un voltaje bajo (0 V), lo que hace muy complicado que el sistema tome en cuenta valores de "ruido" fuera del sistema binario como por ejemplo 3 V. Se usan niveles de tolerancia en virtud de que los sistemas eléctricos tardan una fracción de tiempo en lograr la generación de esos voltajes para representar los dos dígitos.

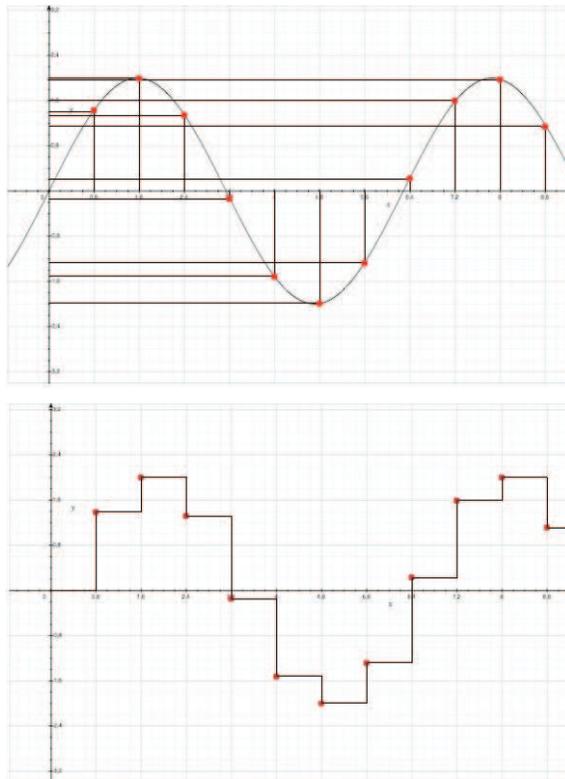


Figura 2.3: Hipotética representación de la conversión análogo/digital (A/D) de una señal continua a una señal con valores discretos, representación con frecuencia de muestreo y profundidad de bits extremadamente baja, se aprecian errores de reconstrucción entre la señal senoidal original y su representación digital.

Las señales analógicas son representaciones continuas de un fenómeno físico tal y como lo es el sonido, por lo que constan de una cantidad infinita de valores independientemente del instante de tiempo en que se analiza la señal, sin embargo el oído humano no es capaz de distinguir una diferencia en esa cantidad de valores a lo largo del tiempo por lo que se usa una tasa de muestreo que extrae muestras de la señal continua, una especie de toma de "fotografías de la amplitud" de la señal a intervalos de tiempo muy cortos.

Haciendo una analogía, tal y como los cuadros por segundo de una filmación que constituyen una imagen en movimiento que es aceptada por nuestros ojos. Esa

toma de muestras se la realiza con una frecuencia dada, conocida como frecuencia de muestreo y se establece mediante el Teorema de Nyquist que debe ser, por lo menos, el doble de la máxima frecuencia presente en la señal que va a ser convertida.

Se dice que por lo menos debe ser el doble porque tomando como punto de partida las altas frecuencias, con dos “fotografías” o muestras por ciclo de la señal en todo su rango de frecuencia, se garantiza una posterior reconstrucción en “buenos términos” dentro del proceso inverso de conversión digital/analógica (D/A); las señales de alta frecuencia se distorsionan pero en la práctica este fenómeno es imperceptible.

Conjuntamente, en la etapa de la conversión A/D se usa el concepto de resolución o profundidad de bits, en el cual la cantidad de representaciones posibles de la amplitud del sonido está dada por la expresión 2^n , donde n es el número de bits. El estándar de 16 bits que se maneja en el *Compact Disc* (CD) produce 65536 (2^{16}) posibles valores para representar la amplitud de la señal, lo que equivale aproximadamente a un rango dinámico de 96 dB ($20\log 65536$), adecuado para gran parte del material de audio presente en el mercado, sin embargo existen dispositivos que manejan resoluciones mayores (24, 48 bits), ellos producen rangos dinámicos aproximados de 144 y 288 dB, respectivamente; lo que es difícil alcanzar como niveles sonoros, semejantes escalas se usan con el fin de ser más fieles y asegurar más holgura de trabajo en la conversión A/D, respecto al rango dinámico de la señal original.

Una vez que el sonido ha sido convertido a datos digitales, se pueden llevar a cabo a continuación una infinidad de procesos dentro de un ordenador o DSP.

2.3 Parámetros principales de una señal

2.3.1 Amplitud

La amplitud de una onda es la desviación máxima de la misma respecto al punto de equilibrio [20]. En términos prácticos en una onda de audio equivale al máximo valor de voltaje que alcanza la onda, tomando en cuenta valores positivos y negativos, alrededor del eje horizontal sobre el que se halla representado aquel voltaje. En ondas acústicas se refiere al desplazamiento máximo alrededor del punto de equilibrio de las partículas de aire por la perturbación producida, la misma que consta de diminutas variaciones de presión atmosférica positivas y negativas, es decir compresión y rarefacción del aire circundante.

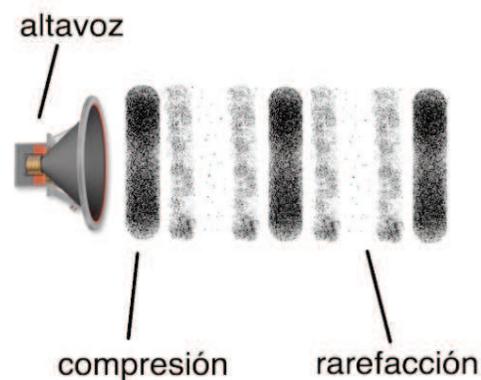


Figura 2.4: Compresión-rarefacción del aire, las áreas más oscuras indican mayor agrupamiento o compresión de moléculas de aire y las más claras rarefacción debido a los cambios de presión.

Las variaciones de presión pueden llegar a tener rangos enormes, por ejemplo para los sonidos más débiles 2×10^{-4} dinas/cm² y para los más fuertes alcanza valores de 280 dinas/cm², ambas medidas por encima y por debajo de la presión atmosférica. Y en términos de movimiento de partículas el desplazamiento causado por el sonido más débil perceptible en promedio alcanza valores de 10^{-9} cm, tomando en cuenta que el diámetro de una molécula es de aproximadamente

10^{-8} cm, se tiene una idea de la extrema sensibilidad del oído humano para captar variaciones de presión y por consiguiente el desplazamiento en las moléculas de aire que estas producen [2].

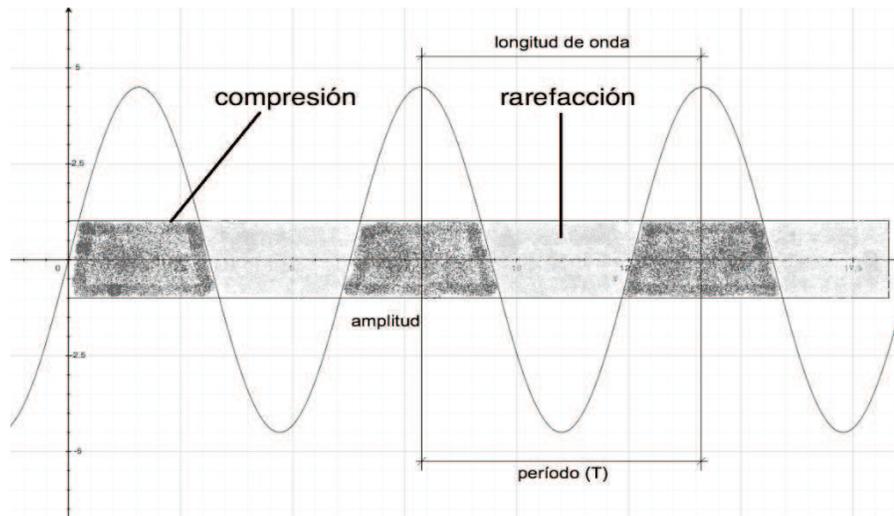


Figura 2.5: Compresión y rarefacción del aire y su relación con la onda acústica, las áreas oscuras indican compresión del aire y las áreas más claras indican rarefacción del medio por los cambios de presión.

Para trabajar de mejor manera con el enorme rango de variación de presión que supone el rango dinámico de la audición humana, se trabaja de forma logarítmica lo que supone una reducción de la escala a valores más manipulables. Y se origina una magnitud denominada Nivel de Presión Sonora, medido en la unidad relativa dB, unidad relativa porque para su correcta expresión se necesita un valor de referencia sobre el cual se relaciona el otro, aquel valor de referencia es 10^{-5} Pa. La fórmula se expresa de la siguiente manera:

$$\text{NPS} = 20 \log \frac{p}{p_0} \quad \text{Ec. 1: F\acute{o}rmula del nivel de presi\acute{o}n sonora NPS o SPL (siglas en ingl\acute{e}s de Sound Pressure Level).}$$

3.3.2 Frecuencia

La frecuencia de un sonido es el número de oscilaciones completas por segundo del elemento generador de sonido. Se mide en Hertz (Hz) en honor al físico alemán Heinrich Rudolf Hertz (1857 – 1894), pionero en la demostración de la existencia de la radiación electromagnética. Las frecuencias en la naturaleza van desde las muy bajas tales como ciertos cambios en la presión barométrica de la atmósfera terrestre (alrededor de 10^{-5} Hz), hasta las frecuencias de la radiación en el universo exterior (10^{23} Hz) [23].

Sin embargo los seres humanos percibimos un rango relativamente muy pequeño de frecuencias, lo que denominamos audible, y para un oído promedio que goza de buena salud aquel rango está entre los 20 Hz como límite inferior y 20000 Hz o 20 KHz como límite superior. Es en este pequeño rango de frecuencias donde suceden todos los fenómenos concernientes a nuestra audición.

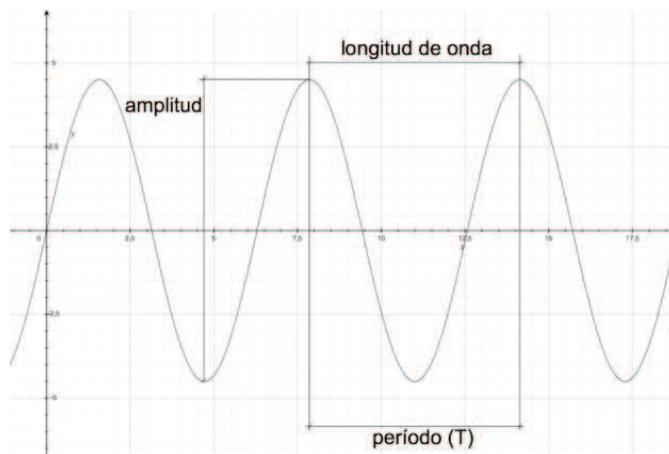


Figura 2.6: Amplitud, longitud de onda y período de una onda senoidal.

En este apartado, merece también atención el estudiar brevemente dos características del sonido muy estrechamente ligadas a la frecuencia como son la longitud de onda y el período.

La longitud de onda es la distancia que viaja una onda mientras realiza un ciclo completo de oscilación, se la denota con la letra griega *lambda* (λ). En el caso del

sonido, debido a que su velocidad de propagación es relativamente constante a una temperatura promedio de 21° C (344 m/s), a 0° C recorre el medio a 331 m/s [19]; se puede calcular la longitud de onda de un sonido dividiendo la velocidad de propagación del mismo por su frecuencia. Las longitudes de onda pueden variar desde 17.2 metros para una frecuencia de 20 Hz, hasta 1.72 centímetros para la frecuencia de 20 KHz.

El período se denota como T. Se define como el tiempo que demora la onda en completar un ciclo y es inversamente proporcional a la frecuencia [20].

Un parámetro que no es usado en el modelo de desarrollo e implementación de esta tesis, salvo como información adicional en el *software*, pero no por ello menos importante es la fase. Una onda se dice que está completamente en fase con otra (0°) si sus ciclos positivos y negativos coinciden perfectamente, en este caso se producirá un aumento de 6 dB en la amplitud de la onda resultante, producto de la suma de las dos anteriores.

Si entre esas mismas ondas hay una oposición total entre sus ciclos positivos y negativos se dice que están completamente desfasadas (180°) y al sumarlas pueden generarse disminuciones de alrededor de -100 dB en la amplitud de la onda resultante, lo que se traduce como una cancelación total [44], [45].

2.3.3 Espectro

El espectro de un sonido es el resultado de una superposición de distintas ondas que forman una emisión sonora que es percibida como un conjunto único de sonidos [22], por ejemplo los instrumentos musicales a pesar de estar emitiendo una misma nota simultáneamente pueden ser distinguidos claramente entre algunos de ellos, esto se da porque tienen distintos espectros.

El comportamiento de estas ondas superpuestas, su presencia o ausencia, su amplitud respecto a la fundamental y su comportamiento en el tiempo son factores decisivos para definir el espectro [21].

Los sonidos se componen de frecuencias múltiplos enteros de la fundamental denominados armónicos, estos armónicos pueden tener distintas amplitudes respecto a la fundamental y comportarse en el tiempo también de manera distinta. También existen frecuencias que no son necesariamente múltiplos enteros de la fundamental y se los denomina inarmónicos, los mismos que se los encuentra en sonidos de gran complejidad como en algunos instrumentos musicales y muchas veces en ruidos propios de la naturaleza así como generados por el hombre.

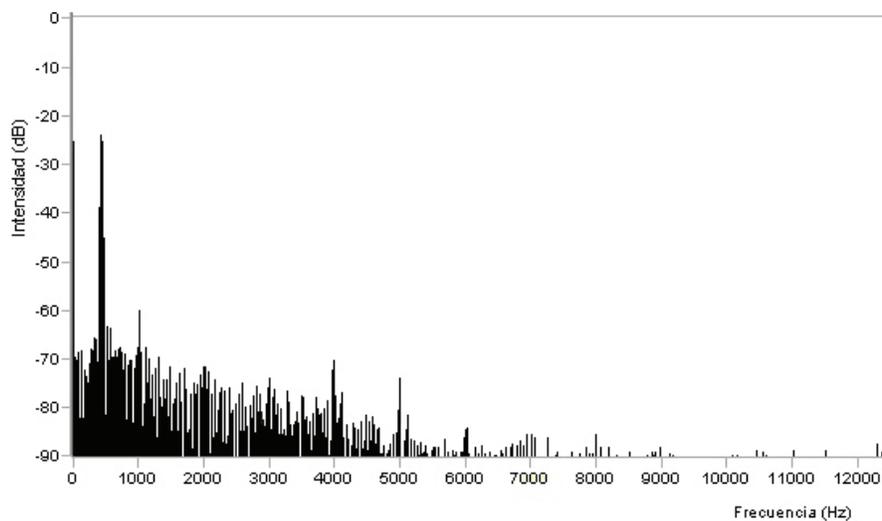


Figura 2.7: Espectro de un diapasón emitiendo la nota A de 440 Hz, se puede observar la presencia de diferentes frecuencias en un instante de tiempo dado, los denominados armónicos [30].

Otro factor es la envolvente dinámica, que hace referencia al comportamiento en el tiempo de la intensidad de cada uno de los armónicos presentes en una onda compleja. Existen envolventes primarias que indican el comportamiento de toda la onda a lo largo del tiempo en términos de amplitud; éstas se encuentran fuertemente ligadas a la forma en que un sonido se produce, las características de su fuente y la manera en que la misma es inducida a producir la onda sonora.

Las envolventes secundarias son similares a las primarias pero analizan el comportamiento de amplitud a lo largo del tiempo de los armónicos, y están

ligadas al grado de atenuación de las distintas frecuencias conforme pasa el tiempo.

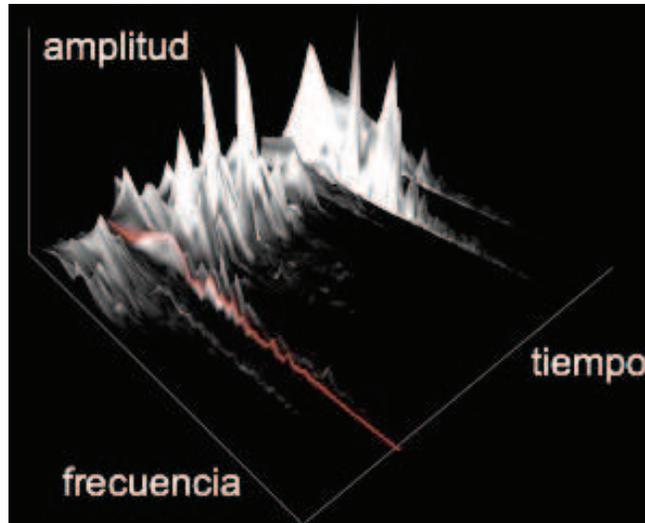


Figura 2.8: Espectro de un sonido, la tercera dimensión corresponde al comportamiento en el tiempo de dicho espectro [31].

Por ejemplo los armónicos superiores (muy agudos) se atenúan más pronto que los de baja frecuencia, esto es claramente notable cuando se escucha el estruendo posterior a un relámpago en una tormenta eléctrica, si el rayo cae cerca se logrará escuchar una especie de “crash”, debido que ha pasado poco tiempo y lograron llegar a nosotros esos armónicos, pero si se está lejos del lugar donde el rayo golpea se podrá escuchar ya un sonido “sordo” de baja frecuencia debido a que el aire y el tiempo atenuaron esos armónicos agudos y dejaron sólo los armónicos inferiores (graves).

El espectro es la presencia de determinados armónicos respecto a otros, existen sonidos en donde predominan los armónicos pares (2, 4, 6, 8, etc.) y otros donde predominan los armónicos impares (3, 5, 7, 9, etc.). En el sistema musical temperado occidental se los estudia mediante la Serie de Armónicos Naturales, donde se muestran los armónicos respecto a una nota fundamental y los intervalos

que ellos constituyen, en la siguiente tabla se toma como frecuencia fundamental al C1 (264 Hz) de la escala musical temperada:

Tabla 2.1: Se observa los armónicos de la Serie Natural de Armónicos y una clasificación muy básica de los fenómenos de consonancia o disonancia que los mismos producen [27].

Armónico N°	Frecuencia (Hz)	Nota	Intervalo
1	264	C1	tono fundamental
2	528	C2	1 octava (consonante)
3	792	G2	1 octava + 1 quinta (consonante)
4	1056	C3	2 octavas (consonante)
5	1320	E3	2 octavas + 1 tercera mayor (consonante)
6	1584	G3	2 octavas + 1 quinta (consonante)
7	1848	Bb3	2 octavas + 1 séptima menor (disonante)
8	2112	C4	3 octavas (consonante)
9	2376	D4	3 octavas + 1 segunda mayor (disonante)
10	2640	E4	3 octavas + 1 tercera mayor (consonante)
11	2904	F#4	3 octavas + 1 cuarta aumentada (disonante)
12	3168	G4	quinta justa (consonante)
13	3432	A4	3 octavas + 1 sexta mayor (consonante)
14	3696	Bb4	3 octavas + 1 séptima menor (disonante)
15	3960	B4	3 octavas + 1 séptima mayor (disonante)
16	4224	C5	4 octavas (consonante)

Un sonido puede llegar a tener los mismos armónicos que otro y aún seguir siendo diferente para nuestro oído. Esto se explica porque cada armónico es capaz de presentar un nivel de energía particular y único, estos niveles de potencia de cada

componente de frecuencia pueden ser claramente observados en representaciones gráficas cuyo resultado se conoce como densidad espectral [22]. Pues cada armónico puede poseer una amplitud diferente y así variar la relación entre las amplitudes de los mismos.

Es necesario hacer una breve generalización del rol de los armónicos en la calificación subjetiva final que un humano asocia.

El primer armónico es la fundamental y es el que da la sensación primaria de altura tonal de un sonido, a veces esta frecuencia no está presente y son sus múltiplos quienes ayudan al oído a reconstruir esa frecuencia fundamental, creando el fenómeno de la "fundamental fantasma". Esto demuestra una calidad impresionante del oído en lo que concierne a análisis de espectros armónicos y reconstrucción de elementos faltantes.

Los armónicos 2, 4, 8 y 16, es decir los que guardan relación de duplicación de la fundamental suman fuertemente a la sensación de altura de la fundamental. Los armónicos relacionados con el 3, tales como 3, 6, 9 y 12, aportan sonoridades de tipo "nasal" y un poco de "calidez" al conjunto.

La "profundidad", "redondez" y demás calificaciones similares provienen de la presencia de los armónicos 5 y 10. Y por último los armónicos 7, 11, 13, 14 y 15 producen disonancias creando sonoridades "duras" y hasta "ásperas" [27].

Por lo tanto el espectro de un sonido es dependiente de la cantidad de armónicos presentes y de sus amplitudes correspondientes, así como su comportamiento en el tiempo.

2.4 La audición humana [3]

Es un proceso complicado que consta de tres etapas funcionales y de estudio principales: oído externo, medio e interno.

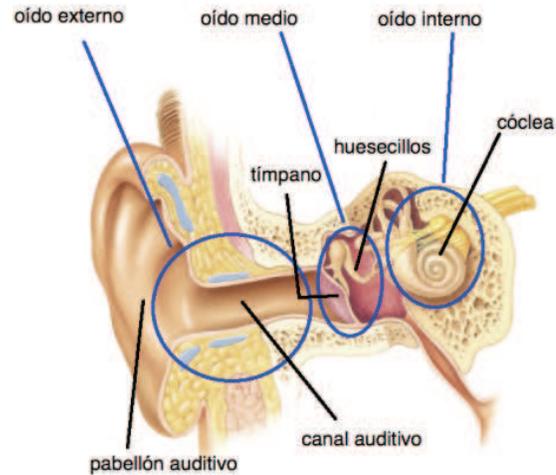


Figura 2.9: Anatomía de las tres etapas del oído humano [3].

El oído externo se compone principalmente de:

- **Pabellón auditivo:** También conocido como oreja. Es una porción de cartílago con intrincadas formas y complicada estructura que se encuentra a los lados del cráneo y tiene la finalidad de recoger y conducir las ondas sonoras hacia el interior de nuestro cráneo.

También es esencial para analizar la localización del lugar de donde provienen las ondas acústicas.

- **Canal auditivo:** Es un canal de forma aproximadamente cilíndrica cuya función es conducir las ondas acústicas hacia el tímpano, y también la amplificación ante sonidos muy débiles así como el cierre parcial del conducto por medio de cerumen ante sonidos muy fuertes.

Una vez que el canal auditivo conduce las ondas sonoras hacia el interior del cráneo, éstas llegan al tímpano, que se encuentra dentro de una cavidad llena de aire denominada caja timpánica, punto de partida del oído medio.

- **Tímpano:** Es una membrana muy delgada pero a la vez muy resistente que se encarga de recoger las ondas sonoras y transformarlas en ondas mecánicas para continuar así con el proceso de audición.
- **Cadena de huesecillos:** Cuando el tímpano recibe las ondas acústicas las transforma en vibración mecánica, la misma que es conducida hacia el interior por una cadena de huesecillos, cuyos miembros son el martillo, yunque y estribo. Estas pequeñas estructuras óseas tienen la función de amplificar y transmitir el sonido hacia la siguiente etapa.

Una vez que los huesecillos han recibido la vibración proveniente del tímpano, estos los conducen por medio de un pequeño receptor llamado ventana oval a un conjunto de intrincadas estructuras que conforman el oído interno:

- **Cóclea:** La cóclea es una estructura con forma de un espiral enrollado, similar a un caracol, en cuyo interior se encuentran unidades especializadas de detección de vibraciones, las mismas cumplen funciones de recopilación y envío de información auditiva al cerebro y de equilibrio al cerebelo. Alberga en su interior a la membrana basilar y el órgano de Corti.
- **Membrana basilar:** Similar a una larga cinta enrollada al interior de la cóclea, soporte directo para el órgano de Corti y la estructura responsable del análisis espectral que realiza el oído, posee varias zonas que varían en rigidez y masa; en su extremo más cercano a la ventana oval es rígida y ligera lo que le provee de una frecuencia de resonancia alta.

En su extremo más remoto es poco rígida y pesada, resonando en frecuencias bajas. De aquí nace nuestro rango de audición en cuanto a frecuencia (20Hz a 20KHz), ya que las ondas acústicas hacen resonar distintas porciones de la membrana basilar, provocando una respuesta de frecuencia característica.

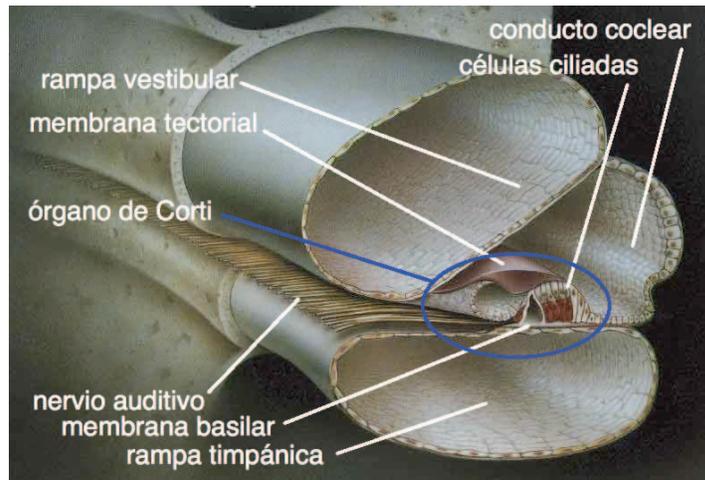


Figura 2.10: Vista en tres dimensiones de una sección de la cóclea y la membrana basilar [36].

- **Órgano de Corti:** El órgano de Corti es una serie de superficies suspendidas sobre cuerdas provenientes de la membrana basilar donde descansan miles de células que poseen cilios o pequeñas vellosidades que están rodeados de líquido perilinfático y endolinfático, el momento en que los huesecillos transmiten una vibración desde el exterior golpean la ventana oval, la misma que excita a este medio líquido con ondas que constituyen una analogía de la onda acústica original. Las células ciliadas se encuentran confinadas entre la membrana basilar y la membrana tectorial, cuando éstas vibran generan entre ellas las diferencias de presión y distancia que las células ciliadas con sus pequeños pelillos o cilios recogen y las convierten a impulsos nerviosos que luego van al cerebro por medio del nervio auditivo para ser analizados e interpretados, formando un verdadero y estupendo analizador de espectro armónico.

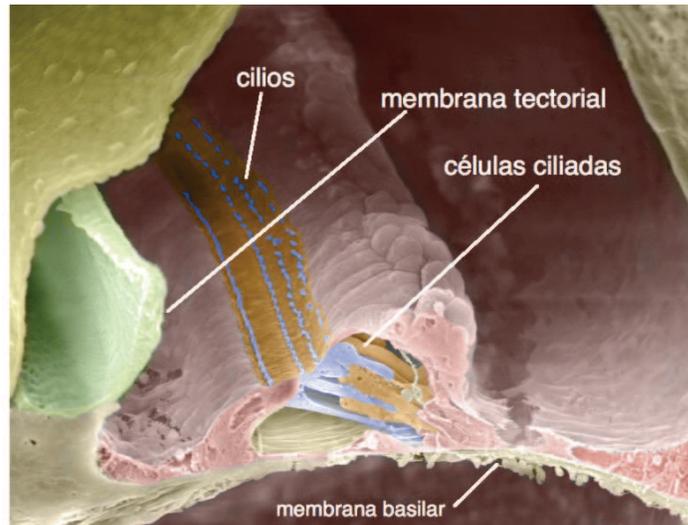


Figura 2.11: Órgano de Corti [37]

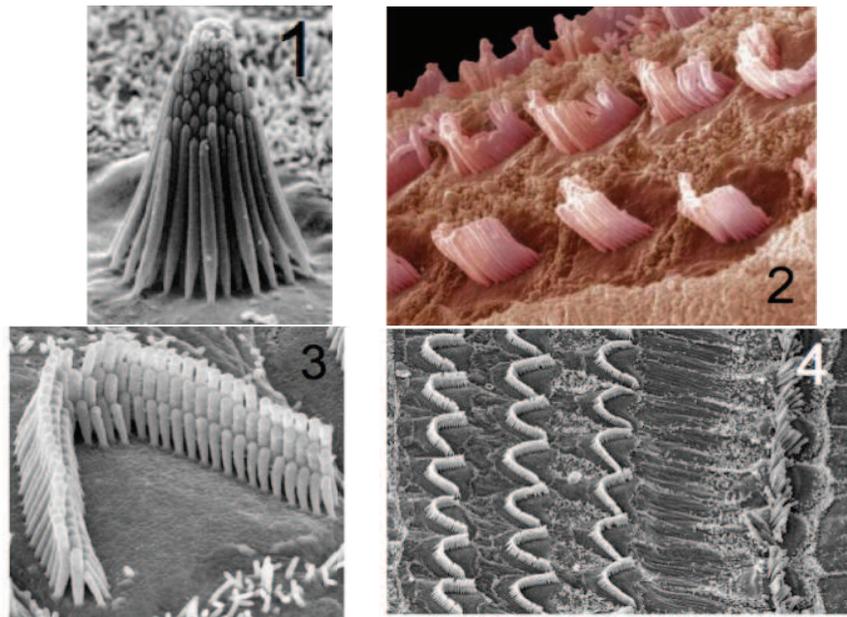


Figura 2.12: Diferentes vistas de las células y sus cilios, en el órgano de Corti.

- 1) Célula ciliada [38]. 2) Grupos de cilios en movimiento [39]. 3) Disposición en V de los cilios [40]. 4) Formación en hileras de los grupos de los cilios [41].

Una característica física del oído humano es la que determina que el rango audible se divida en 31 bandas denominadas “críticas”. La membrana basilar dentro de la cóclea se puede dividir físicamente en 24 segmentos los cuales reaccionan ante un rango de frecuencias determinado, por lo tanto se asumen 24 divisiones igualmente en el espectro audible.

Sin embargo estas 24 bandas abarcan desde los 50 Hz hasta los 15.5 KHz, y para abarcar el rango completo de la audición humana (20 Hz a 20 KHz) se hacen combinaciones lineales entre las 24 bandas ya existentes [5], [6]; obteniendo así las 31 bandas muy comúnmente usadas en el audio profesional.

La clasificación de frecuencias a continuación expuesta, corresponde a la manera como se trabaja en algunos aspectos referentes a la post producción de audio para video, todas las frecuencias se encuentran en Hertz (Hz) [46].

- **Frecuencias extremadamente bajas:** 16 a 20, 20 a 25, 25 a 31.5, 31.5 a 40, 40 a 50.
- **Frecuencias muy bajas:** 50 a 63, 63 a 80, 80 a 100.
- **Frecuencias bajas:** 100 a 125, 125 a 160, 160 a 200, 200 a 250, 250 a 315.
- **Frecuencias medias bajas:** 315 a 400, 400 a 500, 500 a 630.
- **Frecuencias medias:** 630 a 800, 800 a 1000, 1000 a 1250.
- **Frecuencias medias altas:** 1250 a 1600, 1600 a 2000, 2000 a 2500.
- **Frecuencias altas:** 2500 a 3150, 3150 a 4000, 4000 a 5000.
- **Frecuencias muy altas:** 5000 a 6300, 6300 a 8000, 8000 a 10000.
- **Frecuencias extremadamente altas:** 10000 a 12500, 12500 a 16000, 16000 a 20000.

2.5 Percepciones psicoacústicas de los distintos parámetros físicos del sonido

El cerebro humano interpreta de distintas maneras los parámetros anteriormente expuestos como sensaciones subjetivas no sujetas a medición. Son la manera en que se construye una percepción sensorial completa de aquellas características físicas de las señales; y establecer una relación lineal por ejemplo entre amplitud e intensidad no es posible debido a que nuestro cerebro relaciona también frecuencia para determinar la intensidad, lo que nos da cuenta de la complejidad de los procesos humanos de identificación de estímulos sensoriales.

2.5.1 Intensidad

La intensidad está ligada a la amplitud y también a la frecuencia. Una onda sonora con gran amplitud necesariamente causará en nuestros oídos una gran sensación de intensidad, pero como se expresó anteriormente la relación no es del todo lineal ya que está presente el elemento de la frecuencia que es crucial para determinar la intensidad de algunos sonidos; esto se debe a que nuestro oído es más sensible en ciertas frecuencias a diferencia de otras. Dos sonidos de igual frecuencia y diferente amplitud van a producir una diferencia notable en la intensidad sólo por cuestiones de amplitud de onda.

Sin embargo, cuando comparamos a un sonido de cierta amplitud y frecuencia con otro de menor amplitud pero mayor frecuencia (frecuencias sensibles para el oído humano), el sonido de menor amplitud pero mayor frecuencia se percibirá como más intenso [4].

Esto es debido a la cualidad de realzar ciertas frecuencias y atenuar otras a lo largo del espectro que es capaz de percibir el oído humano, la audición dista de ser lineal lo cual se demuestra con las curvas de Fletcher y Munson, tan populares en el mundo del sonido.

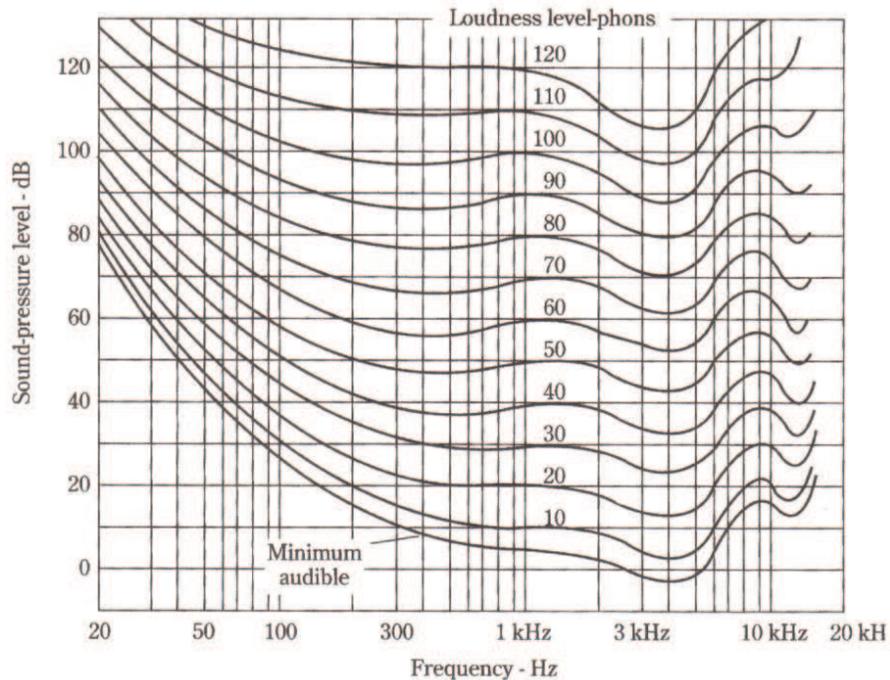


Figura 2.13: Curvas de Igual Sonoridad de Fletcher y Munson [32].

Se observa que en frecuencias bajas el oído tiende a atenuar las ondas (es necesario más nivel para percibir las), ascendiendo en frecuencia hasta llegar al rango de las frecuencias medias se observa como el oído no realiza atenuación alguna e incluso se realiza un refuerzo en frecuencias cercanas a 4 KHz.

Ya en la zona de alta frecuencia de nuevo se producen atenuaciones que confieren a la audición humana características muy importantes a tomar en cuenta para análisis más profundos.

2.5.2 Altura tonal

La altura tonal tiene una relación con la frecuencia mucho más directa de la que tiene la intensidad con la amplitud por ejemplo, sin embargo también se ve influenciada por fenómenos como la intensidad y el timbre.

Dos sonidos de igual frecuencia pero diferente amplitud se los percibe con una ligera diferencia en su altura.

Y sonidos cuyo timbre es brillante pueden dar la sensación de que el sonido tiene una altura superior, igualmente con timbres opacos que dan la sensación de alturas tonales menores [4].

Existe una relación matemática entre los cambios de frecuencia y la sensación de cambio de altura tonal que el oído humano percibe, medida en semitonos de la escala musical temperada, y se refiere a que cada vez que se añade un semitono la frecuencia debe multiplicarse por $\sqrt[12]{2}$, lo que resulta aproximadamente 1,05946; y para bajar un semitono se divide la frecuencia original por la misma raíz [2].

2.5.3 Timbre

Para explicar el término timbre, se puede suponer el hecho de una cuerda o una lámina inducida a vibración con suficiente amplitud para ser audible, lo que se escucha no es una sola frecuencia, más bien es un conjunto simultáneo de las mismas.

A excepción de un diapasón que se acerca a emitir una oscilación senoidal, se podría llegar a aseverar que todos los sonidos en la naturaleza se componen de ondas complejas.

Ninguno constituye un único tono puro, sino una superposición de los mismos que componen ese sonido tan característico que emite cada elemento en la naturaleza y que permite identificar claramente entre dos fuentes que emiten un sonido con una misma frecuencia e intensidad, por ejemplo dos instrumentos musicales emitiendo la misma nota y con similar intensidad son perfectamente diferenciables incluso para un oído no entrenado [2].

El timbre es una característica del sonido relacionada con el espectro y es la manera en que el cerebro distingue entre dos fuentes de similares características en cuanto a frecuencia y amplitud pero diferentes en espectro. La diferente presencia e intensidad de los armónicos hacen que el oído interprete esas

variaciones como cambios en el “color”, “textura”, “timbre”, entre otras calificaciones, del sonido.

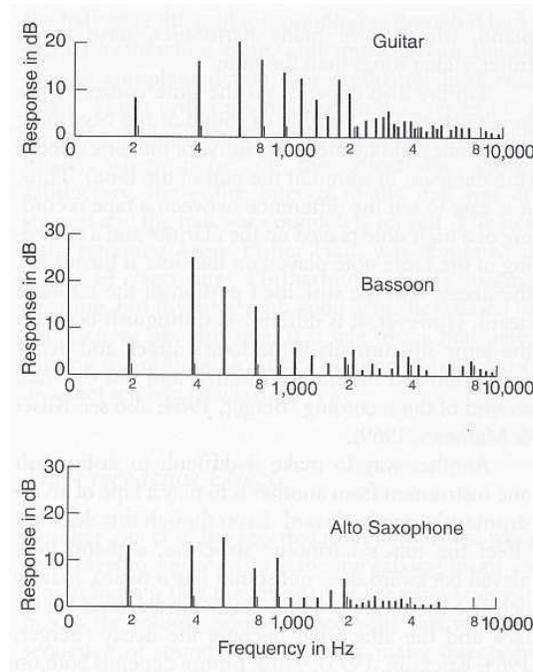


Figura 2.14: Espectro de una guitarra, un fagot y un saxofón alto, todos tocando una nota fundamental G3 (196 Hz) [33].

Por lo tanto es la presencia de los diferentes armónicos los que proporciona información valiosa sobre las características de la fuente, especialmente por asociaciones predeterminadas en nuestro cerebro se identifica rápidamente una onda con términos subjetivos como: "profundo", "redondo", "cálido", "suave", "brillante", "opaco", "melodioso", "áspero", entre otros.

2.6 La visión humana [3]

La visión humana es igualmente un sistema de gran complejidad y precisión que permite captar un segmento del espectro electromagnético, percibido como luz visible, su órgano principal es el ojo. En primera instancia se sitúa la córnea, membrana fibrosa que gracias a su transparencia permite la entrada de luz al ojo. Luego el iris, que es el diafragma del sistema de visión, lo controla un sistema de

músculos que amplía o reduce su abertura y por consiguiente la cantidad de luz que entra al globo ocular.

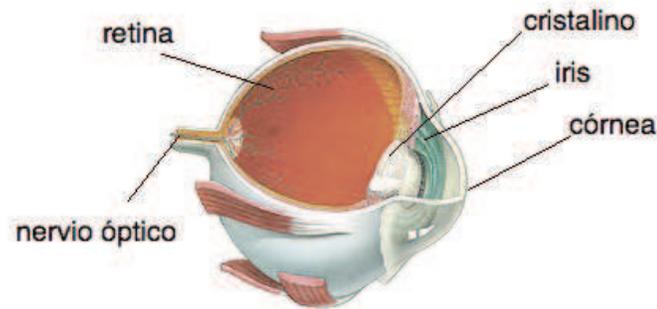


Figura 2.15: Anatomía del ojo humano [3].

El cristalino es una lente biconvexa que es perfectamente transparente, tiene la función de conducir las ondas luminosas hacia la región central de la retina donde se encuentran la mayor parte de los receptores y transductores de la visión.

Una vez que la onda de luz ha sido refractada y regulada en su entrada, atraviesa el interior líquido del ojo hasta llegar a un manto lleno de células receptoras de luz que se encuentra en su parte posterior.

Es la retina, en ella se encuentran dos diferentes tipos de células especializadas en recibir información de luz visible, los conos y bastones que se encargan de transducir la información óptica en impulsos nerviosos que son conducidos a través del nervio óptico hacia el cerebro, para su análisis e interpretación.

2.7 Modelos de composición del color

Existen varias maneras de componer color en distintas condiciones, tales como la composición que realiza una impresora al momento de plasmar imágenes en un papel, o la composición que realizan los computadores y pantallas para mostrar color mediante la proyección de imágenes, usan el método RGB (*Red, Green, Blue*).

2.7.1 Modelo de color RGB [34]

El modelo RGB es un modelo de síntesis aditiva de colores, es decir por medio de la adición de los colores primarios (rojo, verde y azul) se puede crear nuevos colores. Se denomina aditiva porque la suma de los tres colores a su máxima intensidad da como resultado el blanco, que se entiende como el exceso de luminosidad o intensidad y la ausencia de los tres produce la oscuridad total.

Los circuitos electrónicos codifican el color en código binario, al igual que el sonido, por lo que la cantidad de valores asignables dependen de una expresión de tipo 2^n . Por ejemplo, al trabajar con 8 *bits* (2^8), son 256 las posibilidades de luminosidad o intensidad de cada uno de los tres colores (rojo, verde, azul); y por lo tanto se tienen 256^3 (16'777.216) posibles combinaciones de colores por *píxel*. Combinando esas posibilidades se obtiene la gama de aproximadamente 16.7 millones de colores RGB.



Figura 2.16: Interacción de los colores primarios RGB [35].

En la figura se puede observar claramente cómo la interacción de los tres colores produce el blanco cuando los tres convergen al máximo, y el negro u oscuridad total cuando no se juntan en ninguna proporción, de igual manera podemos observar los colores cian, magenta y amarillo; producto de la interacción de dos colores RGB a su máxima luminosidad y un tercero con valor 0; de aquí nace el sistema de colores CMYK (*Cyan, Magenta, Yellow, Black*) que es muy usado en impresoras y dispositivos similares.

CAPÍTULO 3. PROGRAMACIÓN

La programación es un mecanismo por medio del cual se generan procesos que nos pueden ayudar a resolver problemas cotidianos, su principal medio de creación y/o herramienta operativa es el ordenador.

3.1 Breve historia del ordenador o computador

Un ordenador es simplemente en la actualidad un conjunto de circuitos y cables que necesitan instrucciones para realizar las acciones que el usuario requiere.

Anteriormente los ordenadores no contaban con la sofisticación electrónica con que ahora cuentan, producto del desarrollo de la tecnología; eran grandes máquinas incluso con partes mecánicas que cumplían el propósito de agilizar los cálculos, tales como la máquina de sumar de 10 dientes mecánicos de Blaise Pascal en 1642, y en 1670 la máquina de multiplicar de Gottfried Wilhelm Leibniz.

Posteriormente vienen a finales del siglo XIX la utilización de las placas perforadas con contactos eléctricos para almacenar información, los responsables de aquello fueron Joseph Marie Jacquard y Herman Hollerith. Los británicos Charles Babbage y Augusta Byron crean una máquina analítica, considerada como la verdadera precursora del computador digital moderno, capaz de realizar cálculos matemáticos complejos e incluía funcionamiento con electricidad, entrada de datos con tarjetas perforadas, memoria de almacenamiento de datos, procesador independiente para los cálculos e impresión de datos de salida [25].

El siglo XX empieza a ver el desarrollo de los sistemas de ordenadores bajo las presiones de las guerras mundiales por necesidades de cálculo de trayectorias balísticas, vienen máquinas como el totalmente programable Z1 en 1936, el Colossus con sus 1500 tubos al vacío, y en 1945 el ENIAC nace como el ordenador más avanzado de su época pues tenía 18000 tubos al vacío y realizaba varias centenas de cálculos por minuto [26].

Con el apareamiento del transistor a finales de la década del 50 se hace posible crear equipos más pequeños y con mayores capacidades de procesamiento, además de mayor eficiencia térmica y menores costos de fabricación. Esto fue el antecedente para los circuitos integrados, los mismos que aparecieron en la década del 60, reduciendo aún más el tamaño de los circuitos. Los circuitos a gran escala LSI (*Large Scale Integrated*) y los VLSI (*Very Large Scale Integrated*) aparecen en los años 70, posteriormente surgen los GLSI (*Giga Large Scale Integrated*) logrando incluir en una sola placa de silicio varios millones de transistores [25].

La empresa IBM lanza en 1981 el primer computador personal usando unidades de almacenamiento de 10 *Megabytes (MB)*, velocidades de reloj de procesador de 4.77 *Megahertz (MHz)* y memoria *RAM* de hasta 64 *Kilobytes (KB)* [24], [43]; desde entonces estas máquinas han tenido un vertiginoso desarrollo en cuanto capacidad de procesamiento y almacenamiento.

Actualmente, al año 2009 no es extraño encontrar computadores de uso doméstico que posean unidades de almacenamiento del orden de cientos de *Gigabytes (GB)* e incluso un par de *Terabytes (TB)*; en cuanto a la rapidez de sus procesadores hoy en día se bordean velocidades de reloj en torno al orden de un par de *Gigahertz (GHz)* y sus memorias *RAM* están en el orden de un par de *GB*; además incorporan elementos como tarjetas de procesamiento de sonido, de gráficos, entre otros; todos ellos con cada vez capacidades más grandes.

Con su imparable avance, estas máquinas se han ido introduciendo en todos los aspectos de la vida humana, han facilitado tareas que antes se realizaban manualmente y han influenciado desde el entretenimiento y las comunicaciones hasta en los aspectos de diseño y simulación, lo que ahorra grandes cantidades de tiempo y dinero en campos como la investigación.

3.2 Algunos lenguajes de programación [18]

La programación aparece conjuntamente con el ordenador y a mediados del siglo XX es donde empieza un vertiginoso desarrollo producto de la investigación y las nuevas tecnologías.

A lo largo de la historia se han creado más de 2000 lenguajes de programación, algunos de los primeros y más populares antes de la década del 60 fueron Plankalkul, Short Code, Math-Matic, Autocode, Fortran, Lisp, Algol, Cobol.

Es sin embargo, a partir de la década del 60, que el desarrollo de estos lenguajes se acelera para ir acorde con las nuevas tecnologías en construcción de microchips y aumento de capacidades de los ordenadores, algunos de los principales son Basic, Pascal, C, C++, Perl, Python, Java.

Constantemente crece la cantidad de lenguajes, entornos y aplicaciones que se ajustan a necesidades cada vez más específicas en todos los campos del conocimiento. Este trabajo de titulación ha sido desarrollado utilizando el entorno de programación Max/MSP/Jitter.

3.3 Software Max/MSP/Jitter [42]

Nace originalmente como un programa denominado “Max” de la mano de Miller Puckette, su principal aplicación es formar un entorno gráfico para la composición musical por computadora, actualmente existe bajo el nombre de Max/MSP/Jitter, siendo adquirido en 1999 por la compañía Cycling '74, propiedad de David Zicarelli.

Sus orígenes se remontan a los años 80, cuando Miller Puckette crea un *software* llamado “Patcher”, que fue diseñado para un computador Macintosh del organismo francés *IRCAM (Institut de recherche et coordination acoustique/musique*, o en español Instituto de investigación y coordinación acústica/música) de aquella época, el objetivo principal de Patcher era brindar a los músicos del *IRCAM* una

herramienta capaz de ayudar en la composición musical por medio de ordenadores, su primera aplicación práctica fue una obra para teclado y computador denominada Plutón, en la que se lograba sincronizar un computador a dicho teclado y a la par trabajar con un procesador de audio de la época llamado Sogitec 4X.

En 1989 el *IRCAM* desarrolla una versión de Max para sistemas operativos predecesores de los actuales sistemas Linux, esta es denominada Max/FTS y fue la precursora de Max/MSP, MSP consta de una serie de extensiones que amplían el campo de aplicación del programa original que sólo trabajaba con protocolo MIDI, al dominio del audio.

En principio Max/MSP fue diseñado como un sistema de control para sintetizadores, *samplers*, máquinas de efectos, entre otros. Pero paulatinamente fue implementando herramientas y aditamentos que lo hacen aparte de una herramienta de control, un entorno muy poderoso para el procesamiento de información en tiempo real.

Posteriormente, en el año 2003 se añade una sección para procesamiento de video, imagen y matrices denominada Jitter, con la que Max/MSP/Jitter se convierte en una aplicación capaz de manejar simultáneamente datos MIDI, información de audio y además video, todo en tiempo real.

El entorno fue desarrollado en sus inicios bajo el lenguaje C, su modo de programación es por objetos debido a que utiliza unidades de trabajo bien definidas. Las mismas que cuentan con características propias y que pueden interconectarse entre sí para formar estructuras más complejas.

Estos objetos se los puede reunir en el espacio de trabajo de Max/MSP/Jitter denominado *patcher*, para conformar archivos tipo *patch*, una especie de conexión virtual de cables y objetos que conforman un flujo de señal. Igualmente existen los

sub-patch que son la misma modalidad de trabajo pero “condensada” en objetos que van conformando estructuras mayores hasta llegar al *patch* final.

Max/MSP/Jitter además puede crecer añadiendo módulos, *externals* o *plug-in* para su funcionamiento, que consisten en extensiones que amplían las funciones del programa sin interrumpir las que actualmente existen, los diferentes *externals* pueden ser creados por los mismos usuarios según sus propios requerimientos.

Incluso es posible programar en Max/MSP/Jitter muchos de los *plug-in* usados en algunas aplicaciones de grabación y edición de audio, así como de diferentes programas que procesan video.

En el mercado existen otras aplicaciones y entornos de programación similares, tales como MAM Player, Reaktor, vvvv Meso, entre otros. Sin embargo individualmente no combinan las distintas posibilidades que por sí solo ofrece Max/MSP/Jitter.

CAPÍTULO 4. DESARROLLO, PRUEBAS Y ANÁLISIS DE RESULTADOS

Para efectos de mayor tamaño y detalle de algunas imágenes que conforman este capítulo, se referirá a ellas como figuras numeradas dentro del capítulo Anexos, Secciones Imágenes del Desarrollo e Imágenes de las Pruebas y Análisis de resultados.

4.1 Analogías y asociaciones entre audio e imagen

El sonido y la imagen conforman gran parte de la percepción sensorial humana, gracias a estos dos sentidos mayoritariamente y sin desmérito de los demás es que la realidad física del entorno es percibida por el ser humano.

Los órganos sensoriales están conectados unos con otros en cierta manera, lo que puede producir sensaciones derivadas de un sentido en otro. Por ejemplo una de ellas, adquirida por la cotidianidad, es el hecho de escuchar el sonido potente de un avión y enseguida el individuo se agacha o ve hacia arriba, es lógico y normal suponer que los aviones que generan ese tipo de ondas de sonido siempre estarán arriba.

Entonces se puede hacer una simple relación de los elementos visuales y auditivos, que será la base del proyecto de tesis:

- **Frecuencia vs. Rango de Color:** El rango de colores que el ser humano puede distinguir se encuentra en una parte relativamente pequeña del espectro electromagnético (aproximadamente de 400 a 790 THz), por lo tanto se compone de ondas con distintas frecuencias y características similares a las ondas sonoras. El sonido también se compone de ondas que tienen un rango limitado de frecuencias, en el caso de la audición humana de 20 Hz a 20 KHz, por lo tanto es posible hacer una especie de analogía entre ambos espectros y relacionar por ejemplo los colores con menor frecuencia a los sonidos de menor frecuencia y así durante todo el rango de frecuencias hasta llegar al límite superior de ambos espectros energéticos. A continuación se

presenta una tabla donde se aprecian algunos colores y sus longitudes de onda aproximadas, además de sus frecuencias:

Tabla 4.1: Longitud de onda en nm, Frecuencia en THz y Temperatura en grados Kelvin de los colores más básicos del espectro visible para el ser humano [23].

Color	Longitud de onda (nm)	Frecuencia (THz)	Temperatura (°K)
Rojo	780 - 622	384 - 482	1600
Anaranjado	622 - 597	482 - 503	2000
Amarillo	597 - 577	503 - 520	3000
Verde	577 - 492	520 - 610	5500
Azul	492 - 455	610 - 659	18000
Violeta	455 - 390	659 - 769	34000

- **Amplitud vs. Tamaño:** El tamaño de una imagen, figura o patrón visual puede estar directamente relacionado con la amplitud o nivel de un sonido. Se puede entonces asociar una escala de tamaño en la generación de imágenes con la escala de amplitud o la del nivel de presión sonora (NPS).
- **Timbre vs. Textura:** La textura de una imagen corresponde principalmente a una conjugación de luz y sombra, esto puede hacer que ciertos detalles de la imagen se resalten haciendo que se torne “dura” o en tal caso “áspera” a la vista, o caso contrario suavizarla. Los sonidos poseen una serie de cualidades y entre ellas la presencia de ciertos armónicos, con diferentes relaciones de amplitud entre ellos hacen que un sonido sea asociado a términos subjetivos como los antes mencionados. Entonces mediante un análisis principalmente de la presencia de ciertos armónicos y sus relaciones de amplitud, así como de su comportamiento en el tiempo, se puede llegar a asociar sonidos

de ciertas calificaciones subjetivas con imágenes que compartan esas mismas calificaciones.

4.1.1 Modelo de implementación del programa

El modelo que se plantea inicialmente en esta tesis es un *software* que genera imágenes conformadas por figuras geométricas en su salida a partir del análisis de una señal de audio en su entrada. Principalmente relaciona los parámetros de amplitud, frecuencia y espectro en lo concerniente a audio con los parámetros tamaño de las figuras, rango de colores y textura de las imágenes.

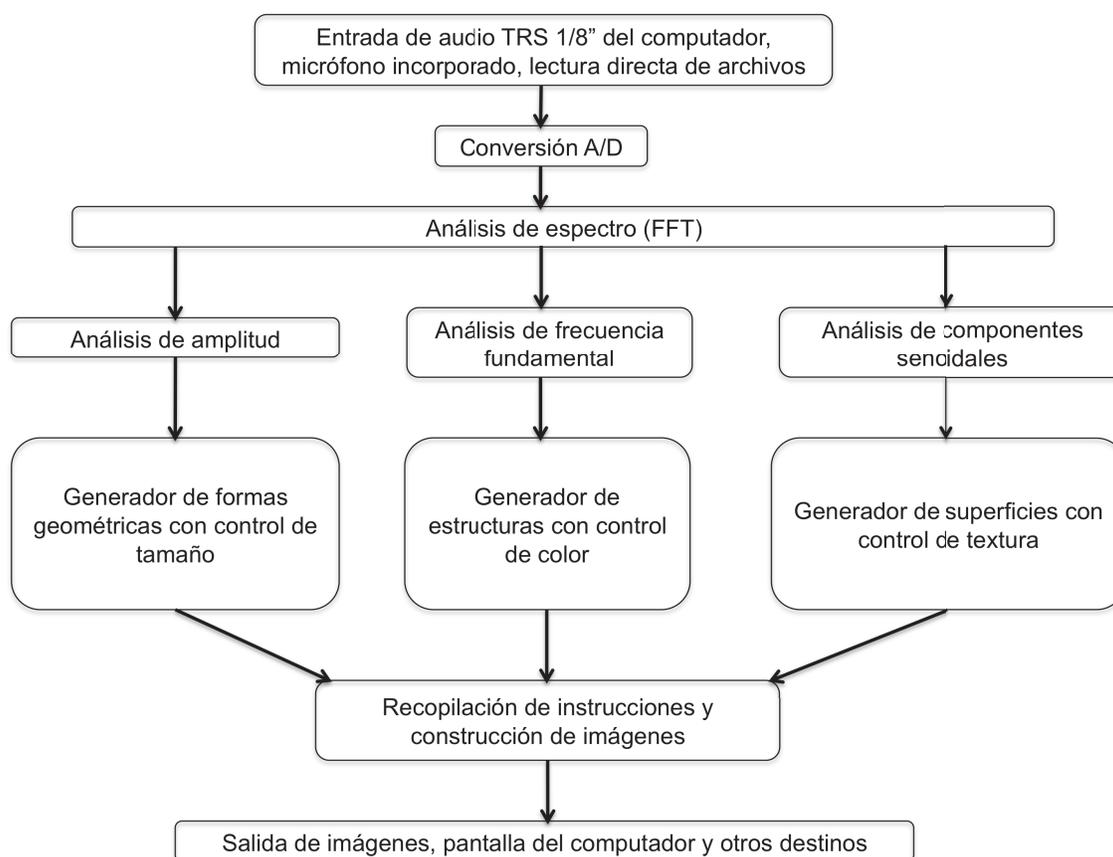


Figura 4.1: Esquema general de la forma de trabajo y las etapas principales de la aplicación.

Las analogías que realiza el programa aquí se las explica con más detalle:

- **Amplitud vs. Tamaño:** El programa analiza la amplitud de la onda continuamente y se obtiene un valor que es enviado a una caja de

números la cual controla otra sección que comanda la generación de formas geométricas básicas con un tamaño controlado que corresponde a la amplitud de la señal de audio en la entrada.

- **Frecuencia vs. Color:** El rango de la audición humana está en un segmento de frecuencias específico (20 Hz a 20 KHz), al igual que el rango de la luz visible (400 a 790 THz). El *software* analizará la frecuencia fundamental de la señal de audio en la entrada (un seguidor de *pitch* o altura tonal) y la asociará a un rango de colores específico. Sonidos de baja frecuencia tenderán a propiciar la generación de imágenes con colores tendiendo hacia el rojo y altas frecuencias irán aproximándose al azul y al violeta.
- **Espectro vs. Textura:** Las imágenes o patrones de figuras geométricas serán generadas mediante estructuras, las mismas que al poseer una superficie pueden adoptar diferentes texturas que guardan relación con el espectro de una señal, muy especialmente con la presencia o ausencia de ciertos armónicos y su amplitud. Cuando un sonido tenga presencia de ciertos armónicos con diferentes amplitudes produce ciertas sensaciones subjetivas como “suave, profundo, áspero”, etc. Esas calificaciones serán asociadas a un generador de imágenes que producirá principalmente texturas que correspondan a la calificación anteriormente mencionada.

4.2 Proceso de construcción y estructuración del programa

La explicación del proceso de construcción y estructuración del programa es realizada primero viendo los distintos elementos que se usan, además de mencionar sus principales prestaciones, para luego pasar a describir la manera en que se conectan, su interacción y como van formando la estructura del programa desde lo más pequeño hasta llegar a lo más grande o complejo, sin embargo el *software* trabaja en una forma diferente, su flujo de señal es totalmente inverso al orden de esta descripción del proceso, pues toma las señales de audio y las analiza, aquellos datos son entregados al generador de imágenes y es donde se realiza la parte medular del procesamiento; pero para

efectos de una explicación clara y concisa se empezará desde lo más sencillo hasta llegar a la estructura final.

Es importante manifestar que los objetos que se exponen en la lista de herramientas usadas en cada *patch*, se encuentran en su estado inicial y por lo tanto no gozan de las capacidades, características y configuraciones de procesamiento, ruteo de señal, manejo de señales, etc. a las que fueron sometidos en el proceso de construcción por lo que al final de cada etapa es incluida una imagen ilustrando el estado final de cada una de ellas, además de la interfaz de usuario que constituye el último eslabón de la cadena de programación.

Además todos los objetos que Max/MSP/Jitter posee, tienen por defecto y protocolo de flujo de información, la particularidad de implementar las entradas en su parte superior y sus salidas en la parte inferior [14], con lo que se logra un camino de datos, que es explicado posteriormente en la construcción de cada *sub-patch* hasta llegar al *patch* final.

Las conexiones entre los objetos (cables), se encuentran en grandes cantidades para componer un flujo de señales capaz de responder a los estímulos sonoros y generar las imágenes al instante, por lo que en esta explicación del proceso de construcción, algunas imágenes ilustrativas pueden aparecer repletas de estos elementos, estos cables o conexiones poseen tres variantes:

- **Cable negro:** Sirve para conducir tan sólo datos o información numérica o de caracteres [7].
- **Cable amarillo:** Conduce señales de audio entre objetos que manejan este tipo de información [8], [10].
- **Cable verde:** Maneja información visual o acerca de matrices relacionadas con Jitter y el procesamiento de imágenes [9].

Las etapas de análisis de audio, generación de imagen y la de interconexión entre las dos anteriores están “encerradas” o “condensadas” en *sub-patches* y pasan a formar parte de una estructura principal mediante una especie de

entradas y salidas, por lo que ciertos elementos pueden estar presentes de manera "virtual", debido a que su raíz está en esas etapas pero mediante aquellas entradas y salidas se proyectan a estructuras superiores del programa.

4.2.3 Etapa de análisis de amplitud, frecuencia y espectro

Como se explicó en el esquema general de trabajo (Figura 4.1), el *software* realiza un análisis por medio de la Transformada Rápida de Fourier (FFT por sus siglas en inglés). Se describen los objetos y procesos en el orden que fueron usados e implementados, respectivamente.

4.2.3.1 Herramientas usadas [7], [8], [11]

En esta etapa se usan los siguientes objetos:

- **inlet/outlet:** Estas pequeñas herramientas permiten la interconexión de sistemas, estructuras, *patches*, objetos, etc. Y son de gran utilidad al momento de ir construyendo módulos que luego forman estructuras mayores, ya que permiten conectar lo que está dentro de ese módulo y también sacar datos que ese módulo produce hacia el exterior.



- **bang:** Un emisor de pulsos que en Max/MSP/Jitter se los denomina "*bang*", son pulsos que generalmente son gatillados por el mismo objeto o por otros elementos y se usan para gatillar otras acciones.



- **ezadc~:** El objeto *ezadc~* tiene como propósito establecer una entrada digital de audio, luego por supuesto de la conversión A/D que realiza la tarjeta de sonido del computador. Es entonces la puerta de entrada para futuros procesos con las señales. Su ícono en el *patcher* de

Max/MSP/Jitter es la imagen de un micrófono muy sencillo de color gris. Es importante acotar que según la tarjeta de sonido que posea la máquina donde se está ejecutando Max/MSP/Jitter el objeto permite trabajar con una o dos señales simultáneas, generalmente gracias a la masificación en los últimos años de tarjetas de sonido de relativamente buen desempeño se puede decir que en la totalidad de los casos el entorno permite con su objeto ezadc~ trabajar con dos canales de entrada, es decir de manera estéreo.



- **sfplay~**: El objeto permite obtener una señal de audio desde archivo y anexarle controles de típicos de un reproductor físico de audio.



- **gain~**: Es un potenciómetro deslizable virtual y actúa como *fader* logarítmico que se lo implementa a manera de control de nivel de la señal de entrada, trabaja con valores configurables perfectamente por el usuario o el programador, hay dos de ellos para controlar el nivel proveniente de ambas salidas del objeto ezadc~.



- **+~**: Este objeto realiza una suma entre sus dos entradas, para a la salida obtener el producto de la adición de las dos señales de entrada.



- **pitch~**: Gran herramienta del Max/MSP/Jitter creada por Tristan Jehan en base al objeto fiddle~, autoría de Miller Puckette. Es un objeto capaz de realizar un análisis de una señal de audio, bajo ciertos parámetros de configuración iniciales y luego de definir algunos parámetros de funcionamiento, a su salida se pueden obtener valores de amplitud en dB, frecuencia principal o fundamental de un sonido complejo, o bien un conjunto simultáneo de ellos, análisis usando FFT por hasta 4096 bandas de frecuencia, además de un detector de picos de amplitud de señal.

```
pitch~ 2048 512 16 blackman70 0 116 16
```

- **number/flonum**: La caja de números nos sirve para publicar los valores que arrojan las salidas de ciertos objetos o bien para introducir valores de operación y/o configuración inicial a ciertos objetos, number acepta números enteros y flonum números decimales. Se encuentran a las salidas del objeto pitch~ para conocer los valores de amplitud y frecuencia fundamental de la señal, hay muchas de ellas presentes en todos los niveles de estructura del programa.

```
>0   >0.
```

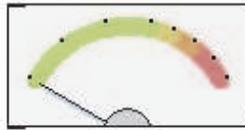
- **number~**: Es un medidor de nivel de señal que entrega datos de forma numérica y que permite también el paso de señal a través de si mismo hacia una salida.

```
>0.
```

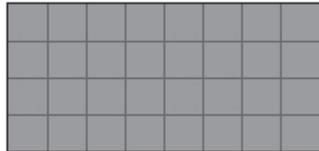
- **meter~**: Es un medidor basado en los medidores *LED meter* físicos que nos entrega una medida de la amplitud de una señal, es posible configurar todos los rangos dinámicos que maneja y la manera en que los representa, incluyendo tiempos de respuesta.



- **levelmeter~:** Este medidor trabaja de manera muy similar a un medidor de tipo *VU meter* físico real. Se lo puede configurar al igual que el medidor anterior en lo que respecta a los rangos en los que trabaja y al tiempo de respuesta de operación.



- **scope~:** Aquí podemos observar la forma de onda de la señal que proviene de la entrada de audio seleccionada en el computador, es similar a un osciloscopio real, aunque con opciones que permiten modificar su tamaño y otras propiedades de funcionamiento.



- **spectroscope~:** Una herramienta del sistema que consiste en un espectrómetro gráfico, en este caso luego de un proceso de Transformada Rápida de Fourier (FFT), además nos permite luego de ciertos ajustes de configuración conocer la respuesta de fase de una señal, etc., todo en forma gráfica y con opciones de presentación de información en escala de grises o a colores.



- **split:** Este objeto es un selector de datos que estén dentro del rango establecido por el usuario, por su entrada van toda la información y luego es clasificada, por su salida izquierda todo lo que cumple con el rango y en la salida derecha todo lo que queda fuera.



4.2.3.2 Proceso de construcción del *sub-patch*

Se configura una entrada de audio digital con el objeto `ezadc~` de manera “virtual”, porque la conexión real se encuentra hecha por medio de un `inlet`, pues en el *patch* principal del programa está realmente el ícono.

Además un objeto `sfplay~` que permite cargar un archivo de audio y enviar esa señal para que el programa la analice y genere imagen, el objeto `sfplay~` posee los controles típicos de un reproductor de audio.

Aquí han sido utilizados de forma básica. Sin embargo en estas primeras etapas, hasta no llegar a la estructura final o interfaz de usuario muchos objetos no aparecen físicamente, sino como observamos aquí: con `inlets` y `outlets`.



Figura 4.2: Entradas tipo `inlet/outlet` para conectar la entrada de audio por micrófono, línea o carga directa de archivos en el computador; con las estructuras superiores del programa.

Por aquí ingresamos la señal estéreo a dos objetos `gain~` que nos sirven para controlar el nivel de entrada de la señal de audio en cada uno de sus canales, estos objetos son ajustables en su escala por lo que deben ser ajustados de acuerdo a las necesidades de la aplicación.

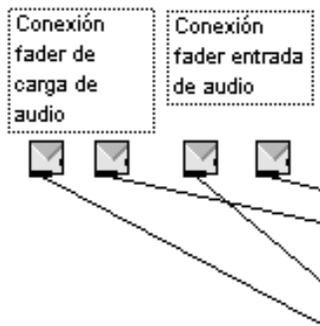


Figura 4.3: Conexión de tipo inlet/outlet con los *faders* que conforman ambas opciones de ingreso de señal estéreo.

Posteriormente se pasa por un sumador que convierte ambas señales en una sola para su procesamiento global como única señal.

De este sumador la señal sale hacia diferentes elementos de análisis de sonido tales como el `scope~` y el `spectroscope~`, los cuales se encuentran presentes virtualmente pues su verdadera salida está en un outlet hacia la interfaz final de usuario.



Figura 4.4: Salida desde el sumador de señales hacia los objetos `scope~` y `spectroscope~` en la interfaz de usuario.

El objeto `spectroscope~` ha sido configurado para que muestre un espectrograma (Amplitud vs. Frecuencia) para poder ver más claramente una "radiografía" del sonido y también a manera de sonograma (Frecuencia vs. Tiempo vs. Amplitud) para poder ver un "historial" del sonido durante un período de tiempo. La respuesta de fase es igualmente mostrada considerando ambas con disposición logarítmica de la frecuencia. Estos dos objetos se encuentran con outlets en la interfaz final de usuario, sin embargo nacen en esta etapa. Algunos objetos no siguen una secuencia de conexión e interacción lineal por lo que pueden paralelamente obtener datos de diferentes lugares y

procesarlos o enviar información en una formación que se aleja del procesamiento en serie.

Se conduce la señal al objeto `pitch~` que realiza la parte del análisis de la señal de audio, este objeto tiene distintas salidas que nos permiten conocer mediante conexiones de cajas de números, los distintos parámetros de una señal como su frecuencia fundamental, amplitud en decibeles, espectro, respuesta de fase, etc.

Este objeto tiene argumentos de funcionamiento interno como el tamaño de los *buffers*, el número de *samples* a usar, el tipo de ventana para el análisis de espectro que influyen en el tipo de filtro digital que se crea para analizar las señales, cantidad de frecuencias y picos de amplitud con que va a trabajar el objeto, entre otros.

Como condiciones iniciales de trabajo se puso además mensajes a la entrada de `pitch~`. Mensajes que determinan la respuesta ante sonidos de bajo NPS, variaciones muy pequeñas de frecuencia, tiempos de respuesta ante picos de señal y comportamiento respecto a ellos, así como la atenuación de algunas frecuencias altas para optimizar el reconocimiento en tiempo real de las frecuencias fundamentales que en la gran mayoría de los casos ocupan rangos bajos del espectro audible.

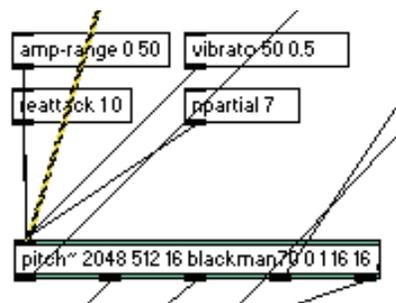


Figura 4.5: Objeto `pitch~` con algunas de sus conexiones y mensajes de control, así como sus argumentos internos de funcionamiento.

La respuesta del espectro es generada por una descomposición senoidal, el objeto analiza la señal y la descompone, por configuración personal, en 16 componentes senoidales de frecuencia, en virtud de que el análisis de textura

se basa principalmente en la Serie de Armónicos Naturales, debido a que esta tesis es destinada principalmente al campo de complementos visuales para refuerzo sonoro, donde predominan elementos como la voz, los instrumentos musicales y la computadora como generador de sonidos.

Esta información sirve para que la siguiente etapa pueda construir las imágenes a partir del sonido. Se establecen cinco cualidades tímbricas de la onda repartidas entre las 16 bandas consideradas:

- La primera toma tan sólo la frecuencia fundamental y la relaciona directamente con figuras que no poseen ningún tipo de textura, una especie de superficies enteramente lisas, asemejándolas a la pureza de una única componente senoidal.
- La siguiente muestra toma en cuenta los armónicos que tienen relación de duplicación de la fundamental (2, 4, 8, 16), que refuerzan la sensación de altura de la misma. Estos disparan la generación de una textura donde se añade profundidad a las figuras, ahora tienen un aspecto de ser completamente tridimensionales, reforzando la figura.
- Posteriormente los armónicos 5 y 10, que aportan a la sensación de calidez del sonido, se relacionan con la adición de características de iluminación a las figuras, con esto se logra más cambios en el aspecto de las mismas, haciendo su superficie con mayor relieve pero muy suavizada.
- Frecuencias múltiplos de la fundamental en el orden del 3 (3, 6, 9, 12) que generan sensaciones de sonidos nasales generan una disminución del proceso de suavizado del *render*, con lo que la superficie se deforma hacia un aspecto menos suave.
- Los armónicos 7, 11, 13, 14 y 15, encargados mediante disonancias de brindar aspereza al sonido, le indican a la siguiente etapa la necesidad de deformar en mayor medida las figuras, asemejándose al efecto que producen los armónicos relacionados al tres.

Ante la presencia de un sonido de gran complejidad o cantidad de componentes senoidales como por ejemplo un ruido blanco o cualquier sonido

aleatorio de la naturaleza, el *software* deforma las figuras al punto de crear incluso nuevas formas a partir de las ya existentes entre las opciones que brinda el programa.

Los datos que genera este análisis son exteriorizados por medio de objetos tipo inlet/outlet hacia la estructura inmediatamente superior o bien hacia los *sub-patches* con el propósito de interconectar todas las etapas entre sí para formar una sola estructura final.

Para poder apreciar de mejor manera esta estructura de análisis armónico y el *patch* completo del analizador de audio, por favor recurrir a la Figura 4.6 y 4.7, respectivamente, en el capítulo Anexos.

4.2.4 Etapa de generación de partículas o figuras geométricas

La etapa de generación gráfica se compone de varias partes que conforman un conglomerado de estructuras de interconexión entre el analizador de audio y el generador primigenio de partículas o formas.

Los objetos usados y los procesos implementados son descritos en el mismo orden en que se construyeron los *patches*.

4.2.4.1 Herramientas usadas [12], [13], [15], [16]

Se usan muchos objetos de etapas anteriores por tener una utilidad importante, tales como cajas de números, mensajes, etc. Los objetos que podrían decirse exclusivos de esta etapa son los siguientes:

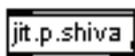
- **metro:** Este objeto es una especie de cronómetro que emite pulsos (en Max se los llama “*bangs*”), los mismos que son generados de acuerdo al argumento del objeto que se da en milisegundos.



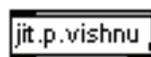
- **jit.matrix:** El objeto que genera, almacena y trabaja con matrices de Jitter, es capaz de almacenar datos tipo matriz y extraerlos luego para usos posteriores.



- **jit.p.shiva:** Es un generador de partículas que las crea de acuerdo a las especificaciones del manejo de datos tipo matriz de Jitter, genera matrices de orden (n x 2), donde n es el número de partículas a generar, y el 2 representa las dos dimensiones de creación de las mismas en su configuración más básica, pues es capaz de generar partículas en tercera dimensión con distintas características de iluminación, desplazamiento, rotación, etc.



- **jit.p.vishnu:** Es el responsable de las aparentes fuerzas de movimiento que rigen a las partículas creadas, a su entrada están las partículas como datos tipo matriz y este objeto aplica modelos de fuerzas que rigen el desplazamiento de partículas, con lo que se logra modificar el comportamiento de las mismas.



- **pak:** Es un objeto que recibe en su entrada varios datos y los dirige todos hacia una misma salida, una especie de multiplexor de información.



- **led:** Es una especie de interruptor luminoso que trabaja en dos posiciones, encendido y apagado, emite 1 y 0, respectivamente según el estado en que se encuentre, uno de ellos es usado para encender el generador de imágenes y se encuentra virtualmente en esta etapa mediante un inlet.



- **jit.p.bounds:** Este objeto sirve para delimitar las partículas a la región del espacio donde uno quiera que se encuentren. Con jit.p.bounds se las puede confinar a áreas establecidas o también dejarlas vagar por todo el espacio donde se generan las formas geométricas.



- **ubumenu:** Este objeto es un menú que no causa interrupciones en las actividades que controla por efecto de un cambio de opción, se le puede asignar varios valores que son detonados por un simple *click* del usuario para elegir una de las opciones.



- **preset:** Este objeto permite almacenar información a manera de casilleros gráficos pequeños de donde haciendo posteriormente un solo *click* con el *mouse* se puede recuperar esa información.



- **loadbang:** Este objeto genera un *bang* cuando un *patcher* es cargado.



- **trigger:** Es capaz de enviar la información que recibe en su entrada a muchos destinos a la vez de izquierda a derecha.



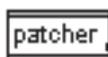
- **print:** Imprime cualquier mensaje manera de reporte en la ventana Max, una ventana que Max/MSP/Jitter posee a manera de bitácora de todos los reportes de procesos realizados por la herramienta.



- **switch:** Su comportamiento es similar a un *switch* físico real, pues tiene muchas entradas de información dependiendo del argumento que el usuario escoja, pero le brinda la capacidad de elección sobre cuál de esas entradas quiere tener a la única salida del objeto.



- **patcher:** Es una opción muy poderosa de Max/MSP/Jitter que permite incluir un *patch* entero dentro de un solo objeto, con el objetivo de ir construyendo estructuras mayores de programación a partir de la reunión y conexión de otras estructuras más pequeñas pero de niveles de complejidad igualmente muy elevados, no existen restricciones en aquello. Se lo conecta con el “mundo exterior” del *patch* general mediante inlets y outlets.



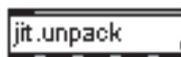
- **jit.split:** Divide a una matriz de Jitter en dos matrices.



- **select:** Este objeto arroja a su salida un dato cuando a la entrada coincide la misma información.



- **jit.unpack:** Puede dividir una matriz de varios planos o de datos superpuestos en varias matrices, de un solo plano cada una.



- **jit.pack:** Hace justamente lo contrario jit.unpack.



- **jit.iter:** Realiza iteraciones con las matrices presentes y las maneja como listas de valores o datos de Max.



- **prepend:** Este objeto inserta listas de caracteres antes de otras, para poder tener una sucesión de listas de datos.



- **zl:** Es una herramienta de trabajo para listas de datos, puede agrupar, desagrupar, distribuir, entre otras funciones.



- **expr:** Es un evaluador de expresiones lógicas, de condicionamiento, etc.



- **jit.gl.handle:** Poderosa herramienta de Jitter que permite rotar con el *mouse* una figura en 3D en cualquier dirección dentro de los 3 ejes dimensionales (x, y, z) y moverla al gusto del usuario dentro del campo de generación de imágenes en tiempo real y mientras los demás procesos suceden.



- **jit.gl.render:** Este objeto es un constructor de estructuras que tienen diferentes tipos de uniones entre sus puntos, lo que resulta útil para en este caso construir múltiples figuras y variantes de las mismas.

jit.gl.render

- **jit.gl.gridshape:** La generación de imágenes y formas geométricas es posible mediante este objeto, el mismo que trabaja con una especie de grilla que obtiene sus datos de las matrices que Jitter suele manejar, tiene muchas opciones de configuración como todos los objetos de Max/MSP/Jitter que se las ingresa mediante cajas de mensajes. Los resultados de los procesos que jit.gl.gridshape genera se los puede extraer como datos hacia otro objeto, directamente realizar un *render* o construir las formas en un objeto tipo pantalla dentro del mismo *patcher*.

jit.gl.gridshape

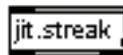
- **gate:** Es una puerta, cuando se le asigna mediante una caja de mensajes un número diferente a cero deja pasar la información de entrada, cuando tiene varias entradas el mensaje de apertura de puerta debe corresponder al número de entrada que se quiere dejar pasar.

gate

- **key:** Es un identificador de códigos *ASCII* y de código de teclado para reportar eventos que se generan al presionar las teclas e identificar o asignar esos eventos a otras acciones.

key

- **jit.streak:** Este objeto maneja probabilidades para determinar la oportunidad de que las celdas de una matriz se superpongan a otras, para realizar una especie de aleatoriedad en la generación de imágenes para superponerlas sin orden aparente. Esto arroja como resultado una deformación de las figuras originales hacia formas parecidas a estrellas producto de la superposición de muchas figuras.



- **jit.tiffany:** Realiza un proceso de *re-sampleo* en los ejes (x, y) de la matriz con el objetivo de aleatorizar el alcance de hasta dónde recorren las figuras el espacio de generación en el sentido horizontal y vertical, y también hasta dónde el generador extiende las aristas y formas de las figuras.



- **jit.window:** El sitio ideal para publicar datos a manera de figuras geométricas, dibujos, etc., es la plataforma final de visualización de todos los objetos previamente explicados, es la pantalla de todos los procesos ya que permite interconectarse con todos esos objetos que procesan imagen y es en donde se manifiestan todas sus prestaciones y características. Es capaz también bajo algunas opciones de configuración diferentes, brindar interacción con el usuario como reportar todos los movimientos del *mouse* en coordenadas y con ellos controlar otros eventos o procesos que puedan estar vinculados.



4.2.4.2 Proceso de construcción del *sub-patch*

Para la construcción de la etapa de generación de imágenes se empieza a hacer un recuento desde la parte más pequeña e inferior en el nivel de escala de la estructura del programa.

Se comienza con un *patch* en donde se encuentran las principales piezas generadoras de la imagen que se usará posteriormente.

Este *patch* está basado en una herramienta que en un origen fue creada exclusivamente para generar figuras geométricas pero sin ningún tipo de control externo respecto a tamaño, colores ni texturas; como un pequeño tutorial de Max/MSP.

Con una serie de modificaciones, adaptaciones y opciones de control es posible hacer que reaccione ante estímulos sonoros y así cambiar profundamente su comportamiento.

Se inicia con un loadbang que emite un “bang” o impulso cuando el *patch* es cargado e inicia la activación del objeto jit.window. Luego con un mensaje de control se inicia el objeto jit.split que divide una matriz en dos matrices ya que distintos datos de control provienen de estructuras superiores del generador de imágenes y se hace necesario diferenciarlas para su direccionamiento hacia partes muy específicas de esta primera etapa.

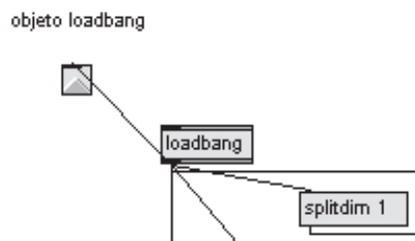


Figura 4.8: Objeto loadbang con una conexión tipo inlet/outlet y una salida hacia un mensaje de control.

Posteriormente una salida de `jit.split` va hacia un objeto trigger para crear un mensaje de control hacia `jit.gl.render` que es quien crea el espacio para la construcción de las imágenes.

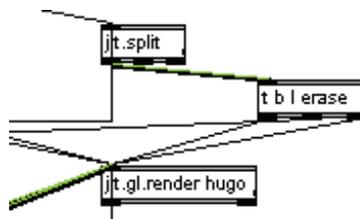


Figura 4.9: El objeto `jit.gl.render` crea y nombra la pantalla y recibe datos de control desde múltiples lugares.

Un objeto que trabaja con datos empaquetadas, `jit.unpack` recoge información proveniente del objeto trigger, gracias a que su argumento de trabajo fue establecido como “5”, reparte esa señal hacia otro objeto de compresión de información como es `jit.pack` con argumento “4”.

Por lo que podría actuar como un reductor de flujo y una especie de puerta de clasificación ya que de los cinco datos originales ahora se escogen cuatro para seguir trabajando.

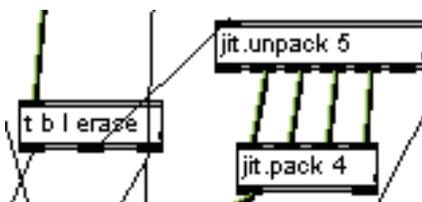


Figura 4.10: Separación y agrupamiento posterior de datos por medio de objetos `jit.unpack` y `jit.pack`.

Esa lista sufre una serie de iteraciones por medio del objeto `jit.iter` para luego pasar al objeto `zl` que bajo la configuración de “slice” selecciona el primer valor de la lista que resulte de esa iteración, estos valores cambian constantemente ya que la estructura superior a este generador arroja diversos datos de control cada 0.001 ms o 1 μ s.



Figura 4.11: Iteración y selección de datos.

El objeto `expr` recibe la información y decide su paso o no, debido a que es un evaluador de expresiones lógicas o de valores que cumplen con ciertos requerimientos de lógica matemática.

Toda la información anterior viene a desembocar en un “*bang*” que activa o no un mensaje de control de las posiciones por donde se desplazarán inicialmente las figuras y su comportamiento a lo largo de la pantalla de `jit.window`.

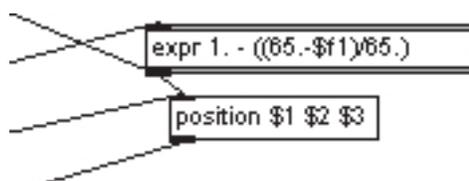


Figura 4.12: Evaluación de datos y activación de mensaje de control de posición.

La generación de las formas geométricas como estructuras iniciales tipo grilla está a cargo de `jit.gl.gridshape`, este objeto genera distintas formas geométricas, las que pueden ser escogidas mediante un menú tipo `ubumenu` conectado a la entrada de `jit.gl.gridshape`.

Con distintos mensajes se puede controlar literalmente todo aspecto de la generación de las imágenes como colores, tamaños, texturas, posiciones, duración en el tiempo, etc.

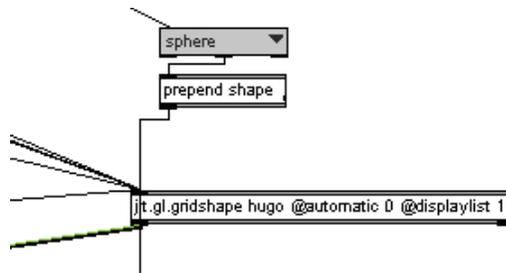


Figura 4.13: Objetos involucrados en la generación de las primeras estructuras sobre las cuales se construye formas sólidas.

Se controla en este *patch*, con los parámetros de las ondas de sonido analizadas en la primera etapa, los tres aspectos fundamentales de esta tesis: amplitud, color y textura; con el *patch* en su estado original sin opción a control de los procesos, los dos primeros se controlan con simples mensajes, se explicará luego como esos tres aspectos son controlados por los datos que arroja el analizador de audio.

La textura es modificada en términos de suavizado, iluminación y profundidad del proceso de *render* de las figuras, ya que las mismas pueden ser modificadas más profundamente con las herramientas que ahora entran a operar.

Los objetos *jit.streak* y *jit.tiffany* trabajan con probabilidad y *re-samplero* de los ejes (x, y), respectivamente. El primero modifica la imagen haciendo superposiciones de muchas figuras obedeciendo un orden de probabilidades y lograr así cambio de texturas, también controlando la escala de aquellas superposiciones.

El segundo “aleatoriza” el alcance de hasta dónde se extienden la figuras en su desplazamiento dentro de *jit.window* en los ejes vertical y horizontal (x, y), cuántas columnas y filas de las matrices que contienen la información de las figuras geométricas van a ser procesadas y además hasta dónde ellas extienden sus formas, aristas, ángulos, etc.

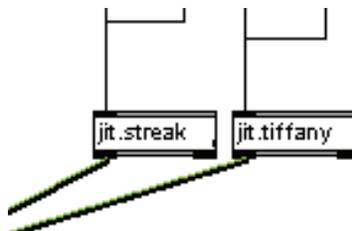


Figura 4.14: Objetos que modifican el desplazamiento y la superposición entre las figuras.

Estos reciben los datos de un objeto gate que es una puerta, el objeto así obtiene información del mismo `jit.gl.gridshape` o también directamente del usuario al activarlo o no mediante las opciones de control que han sido implementadas.

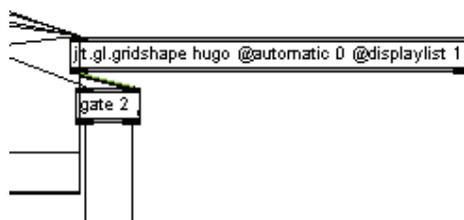


Figura 4.15: Objeto gate actuando como puerta.

El objeto `jit.gl.render` también es capaz de ofrecer control sobre la manera en como son conectados entre sí los puntos que conforman la malla de la estructura de cada figura, puede ser por líneas, puntos, polígonos diversos, etc.

Con esta opción se logra modificar la visualización de las mismas.

A la par, se conecta la herramienta de `jit.gl.handle` que permite al usuario rotar la posición de la imagen dentro de `jit.window` en los tres planos dimensionales (x, y, z); con esto se da más libertad para que el usuario escoja distintas posiciones y poder modificar también de esta forma vía *mouse* del computador por ejemplo la incidencia de la luz al cambiar la rotación. Todos estos procesos pueden ser realizados en tiempo real por el usuario u operador del sistema.



Figura 4.16: Control del tipo de unión entre los puntos de las estructuras y control de rotación tridimensional de los gráficos en jit.window.

Se incluyen a esta etapa opciones como el color de fondo de jit.window y también si las formas geométricas se extinguen luego de un tiempo o forman superposiciones entre ellas. Y con el objeto key se puede asignar una tecla para hacer la imagen a pantalla completa, se escogió la tecla de escape.

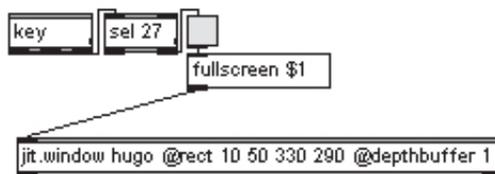


Figura 4.17: Asignación de la tecla de escape o ESC para ampliación a pantalla completa de la imagen.

Una vez que esta etapa inicial de generación de figuras geométricas está lista, es encerrada en un objeto tipo *patcher* para componer así una estructura superior, en donde se encuentran los controles y las conexiones que vinculan la parte de la imagen con la parte del sonido.

El *patch* completo del generador primigenio de figuras geométricas, se encuentra en el capítulo Anexos, Figura 4.18.

Una vez completada esta primera parte de la etapa de generación de imágenes se pasa a la siguiente donde un objeto metro que emite pulsos cada 0.001 ms o 1 μ s controla el ritmo de trabajo del objeto matrix, responsable de generar la matriz con la cual se construirá posteriormente las imágenes. Estos pulsos sin emitidos como *bangs*, que es la manera como Max/MSP/Jitter entiende los pulsos de control para todos sus procesos.

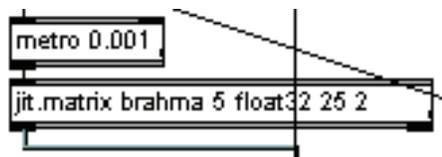


Figura 4.19: Configuración del reloj interno del generador de imagen, establecido en 0.001 ms ó 1 μ s.

Aquellos datos son enviados hacia un objeto jit.p.shiva, para así crear los códigos de generación básicos donde las figuras adquieren características como profundidad, iluminación, etc.

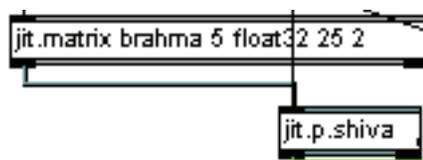


Figura 4.20: Aumento de características de profundidad e iluminación a las imágenes.

Algunas instrucciones son ingresadas a jit.p.shiva tal como números mediante las cajas respectivas que conforman mensajes de control para la vida o duración en el tiempo de las figuras y para su cantidad de apariciones por cuadro de tiempo, y la variabilidad a lo largo del tiempo de cada una de estas instrucciones.

A continuación jit.p.vishnu recibe aquella información y aplica una especie de fuerza o patrón de movimiento y desplazamiento de partículas, con esto se logra un comportamiento más aleatorio de las figuras cuando el generador se enciende. Y eso se logra con algunas instrucciones provenientes de cajas de números que son ingresadas como mensajes de control para manejar la posición en el espacio tridimensional, la fuerza que empuja como una especie de “viento” a las figuras, la rotación, la velocidad a la que son emitidas y también la variabilidad con que cambian estos parámetros de control.



Figura 4.21: Estos objetos se complementan para dar profundidad y movimiento a la imagen.

La información fluye ahora hacia jit.p.bounds para confinar las figuras a desplazarse por espacios limitados o bien para dejarlas recorrer por todo el espacio de jit.window libremente. Al igual que anteriores objetos, jit.p.bounds recibe instrucciones por medio de mensajes a su entrada, que vienen dados por cajas de números.

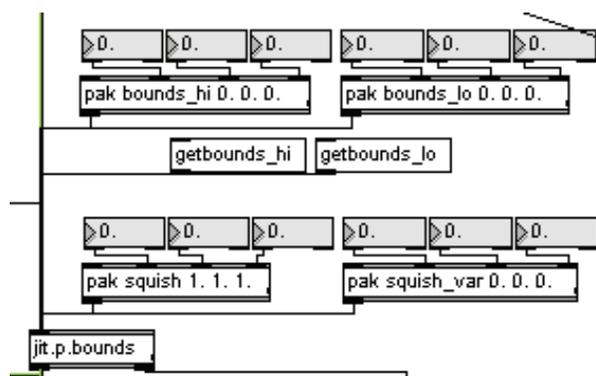


Figura 4.22: Se observa la cantidad de mensajes que controlan a jit.p.bounds.

Estas instrucciones que han sido procesadas y reunidas antes de jit.p.bounds ahora fluyen desde ahí directamente al generador primigenio, específicamente el objeto jit.gl.render con el objetivo de controlar a esa etapa inicial de generación.

La manera en que esta etapa modifica los colores, tamaños y texturas de las figuras geométricas está asociada directamente a una división entre rangos o intervalos de datos y una posterior clasificación para gatillar diferentes eventos de acuerdo a la misma.

Los colores van asociados en una escala desde el rojo hasta el violeta; pasando por anaranjados, verdes, amarillos y azules. Esto va asociado a una

división de frecuencias por 31 bandas de tercio de octava, comenzando desde los 16 Hz hasta los 20 KHz.

Una vez que el programa hace aquella división de bandas de frecuencia, determina cuál mensaje gatillar, estos mensajes contienen información sobre el color que van a adquirir las figuras geométricas y son activados directamente hacia objeto `jit.gl.gridshape` para determinar el color en tiempo real.

El generador de imágenes y su etapa de interacción entre las diferentes analogías se ilustra en tres imágenes en el capítulo Anexos, la analogía frecuencia/color se presenta en la Figura 4.23.

En el caso de la amplitud, se maneja de manera muy similar al de la frecuencia y el color. El analizador de audio divide en 29 intervalos de 6 dB de NPS cada uno a la onda de audio, desde 0 dB hasta 174 dB; el programa determina cuál mensaje activar, el mismo que también va directamente a `jit.gl.gridshape` con información clave para determinar el tamaño de las figuras desde una escala relativa desde 0.00001 hasta 167.772156, estos números son manejados por el objeto `jit.gl.gridshape` como información de tamaño para las figuras geométricas.

Se usó un rango dinámico de tal magnitud para asegurar holgura en el trabajo del *software*, sin embargo se considera muy difícil que existan tales condiciones de trabajo respecto a NPS tan grandes como 174 dB donde el programa se utilice.

La relación amplitud/tamaño es mostrada justamente como se desarrolló en el *patch* del generador completo de imágenes, se encuentra en Anexos, Figura 4.24.

De igual forma sucede con la textura, el programa analiza cuáles de las 16 componentes senoidales exclusivas para el área de textura están presentes en todo momento y luego determina qué mensajes dar respecto al tipo de superficie que se les da a las figuras, se manejan 5 texturas diferentes asociadas a 5 grupos con diferente presencia de armónicos.

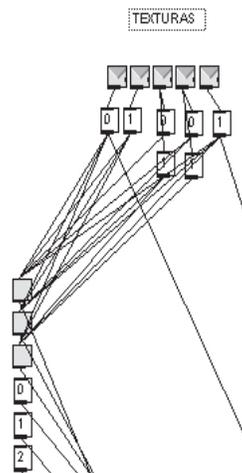


Figura 4.25: Conexión entre las etapas de análisis de audio y generación de imágenes correspondiente a la analogía espectro/textura.

Existe en esta parte del programa una lista de mensajes referentes al color de fondo sobre el cual se generan las imágenes y de un efecto visual que constituye en una superposición continua de las figuras, logrando una especie de mosaico.

Estas opciones están presentes en la interfaz final de usuario, pero nacen en esta etapa y se las conecta con las estructuras superiores mediante inlets.

En la Figura 4.26 del capítulo Anexos se observa la estructura de la herramienta que permite cambiar el color de fondo y controlar la superposición de imágenes, se encuentra dentro del generador gráfico.

El *patch* del generador completo de imágenes se aprecia en la Figura 4.27 del capítulo Anexos.

4.2.5 Conexión del analizador de audio con el generador de imágenes

4.2.5.1 Herramientas usadas

Los objetos y herramientas aquí utilizados son los mismos que en las anteriores etapas, simplemente su disposición es diferente.

4.2.5.2 Proceso de construcción del *sub-patch*

La interconexión entre el analizador de audio y el generador de imágenes es uno de los pasos sustanciales en este programa, pues conforma el vínculo real que hace posible desarrollar esta herramienta.

Esta etapa está repleta de objetos tipo split con el propósito de obtener los datos de salida del analizador de audio, clasificarlos o separarlos en rangos establecidos e ingresar esa información a las diferentes unidades del generador de imágenes.

Un objeto split que selecciona datos entre cierto rango es situado luego de las cajas de números que contienen datos de amplitud y frecuencia, existen en este *sub-patch*, dos grupos de objetos split.

El primero está dedicado a dividir la información de amplitud de la señal de audio en rangos de 6 dB, comenzando desde 0 dB hasta 174 dB.

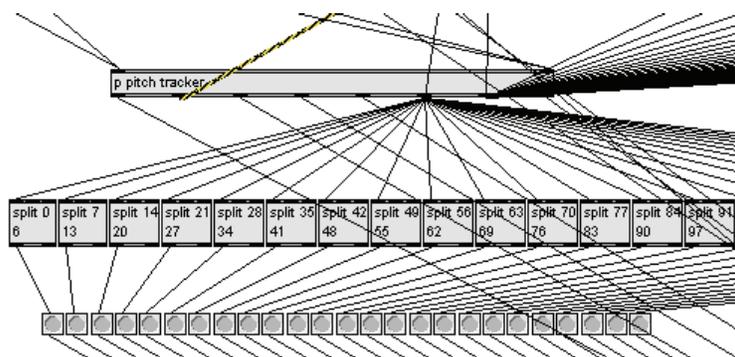


Figura 4.28: Parte de la división de los 29 rangos de amplitud entre 0 dB y 174 dB, con objetos split.

El segundo grupo se enfoca en clasificar la información de frecuencia fundamental del audio en rangos que van desde 16 Hz, hasta 20 KHz, siguiendo la distribución de 31 bandas de frecuencia, comúnmente usada en el audio profesional.

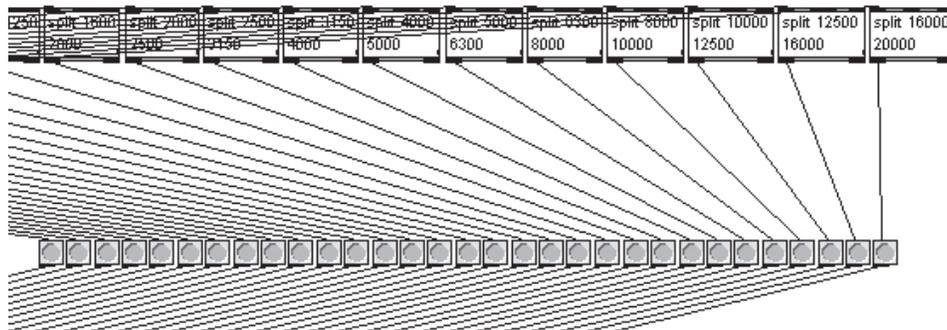


Figura 4.29: Parte de la división de las 31 bandas de frecuencia desde 16 Hz hasta 20 KHz, con objetos split.

Una vez que split confirma que los datos se encuentran en su rango de trabajo emite un pulso que activa un objeto bang, este pulso o "*bang*" es transmitido justamente hacia los mensajes de tamaño, color y textura que controlan el generador de imágenes o figuras geométricas, por medio de las entradas o inlets que contiene el *sub-patch* encargado de la imagen. Para observar la conexión entre los objetos split y el *sub-patch* del generador de partículas o imágenes, Figura 4.30, Anexos.

Se incluye la opción de ingresar datos para modificar los colores de fondo de jit.window y además la de este cambio conjuntamente con la superposición de las figuras que se van generando para componer una especie de mosaico cambiante que irá acorde a las nuevas formas geométricas que se van construyendo, esto nace en la etapa anterior, sin embargo las estructuras se van interconectando entre ellas, aparentando redundancias en las conexiones que se realizan con objetos inlet/outlet.

La forma cómo se van interconectando opciones por medio de objetos inlet/outlet hasta llegar a la etapas finales es ilustrada en Anexos, Figura 4.31, la misma que muestra la opción de los cambios de color de fondo y el control de la superposición de imágenes en el *patch* de interconexión entre audio e imagen.

El *patch* completo de la etapa de interconexión entre el analizador de audio y el generador de imágenes se presenta en la Figura 4.32, Anexos.

4.2.6 Interfaz de usuario

4.2.6.1 Herramientas usadas

Las herramientas y objetos usados en esta etapa son exactamente los mismos que se usaron en etapas anteriores, tan sólo hay un nuevo elemento:

- **panel:** El objeto panel que permite poner una superficie de fondo con colores y efectos de sombra u otras opciones estéticas para armar en este caso una interfaz de usuario.



4.2.6.2 Proceso de construcción del *patch*

La interfaz es el paso final, en donde todos los objetos que en etapas anteriores se describieron con aquella especie de presencia "virtual", aparecen para formar parte de los controles que se le brinda al usuario sobre la fuente de audio, su amplitud y los controles básicos del generador de imágenes.

El objeto panel que conforma la superficie de fondo donde se anidan todos los elementos se encuentra configurado en su color y en el efecto de sombra en sus contornos.

La disposición de los elementos en la interfaz de usuario provisional corresponde a un ordenamiento de acuerdo a las entradas de audio y su análisis, para posteriormente incluir todos los controles del generador de imágenes en una sección independiente, pero todo dentro de la misma interfaz.

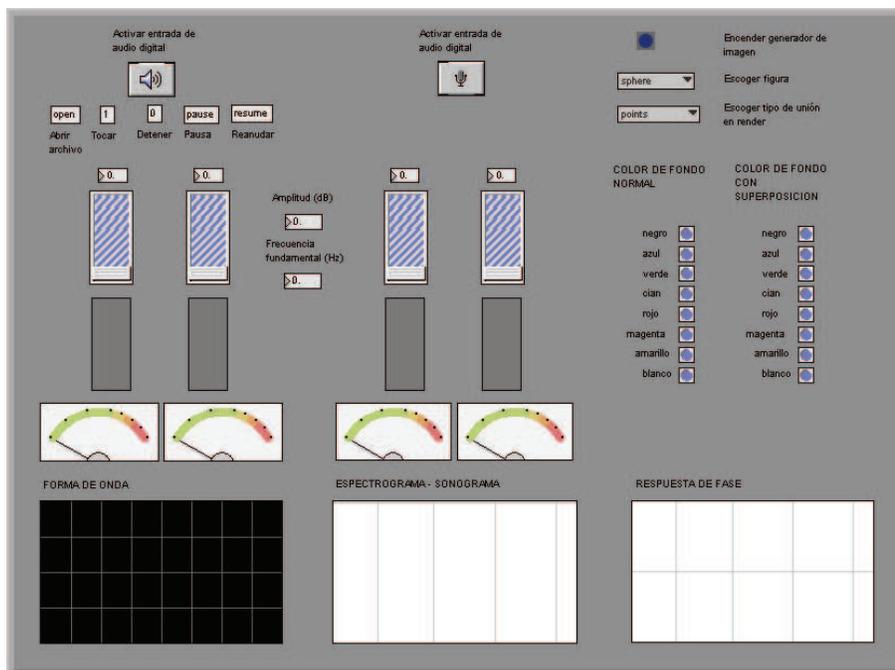


Figura 4.33: Etapa final en su estado inicial de trabajo, interfaz de usuario provisional usada para las pruebas del *software*.

Una opción muy importante en esta interfaz es la capacidad de Max/MSP/Jitter de esconder al momento de correr el programa todos los elementos que el usuario desee.

Así se logra obtener un ambiente visual limpio de objetos y cables, extremadamente sencillo a diferencia de las anteriores etapas o *sub-patches* donde la tónica del ambiente visual era una innumerable cantidad de objetos y conexiones que cruzaban por todos lados el espacio de trabajo.

El usuario puede activar o desactivar esta opción en el *patcher* de Max/MSP/Jitter, en el modo de edición del programa, la "aparición" o "desaparición" de todos y cada uno de los objetos, desde cables hasta los objetos más complejos, según su conveniencia para determinar qué objetos aparecen cuando el *patcher* sale del modo de edición. Esta opción resulta muy práctica para construir justamente interfaces de usuario con los elementos justos y necesarios para mostrar, caso contrario sería mucho más difícil operar el programa desde sus distintas etapas de construcción y desarrollo.

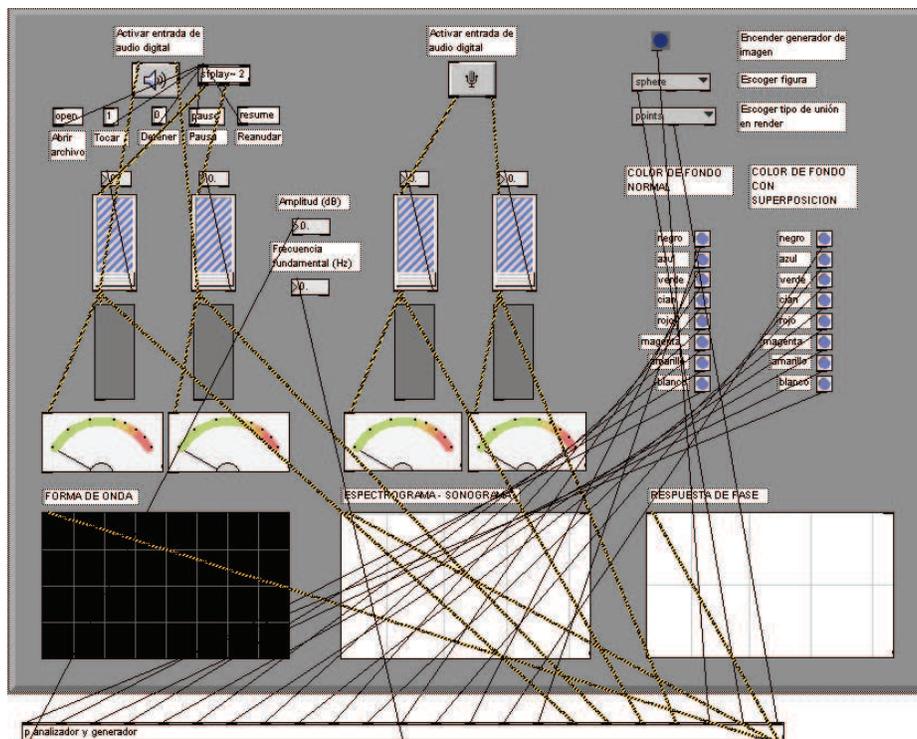


Figura 4.34: En comparación con la anterior imagen, se aprecia la opción descrita para esconder o mostrar elementos no deseados por el usuario el momento de correr el programa.

La interfaz utilizada para el análisis de resultados es una interfaz de prueba que es susceptible a cambios, los cuales surgen a partir de necesidades específicas de aspecto funcional, estético entre otros.

Razón por la cual luego de las pruebas surgió una interfaz final de usuario con una disposición diferentes de sus elementos.

El programa fue desarrollado en Max/MSP/Jitter funcionando bajo la plataforma Windows de la empresa Microsoft; sin embargo para ilustrar la compatibilidad del programa y el *patch* en cualquier sistema operativo que soporte el entorno y sus *externals*, se muestra a continuación la interfaz final de usuario en una ventana de Max/MSP/Jitter bajo la plataforma Mac OSX de la empresa Apple.

El programa una vez realizada la instalación del entorno y sus objetos externos necesarios puede desempeñarse en cualquier sistema operativo que soporte Max/MSP/Jitter.

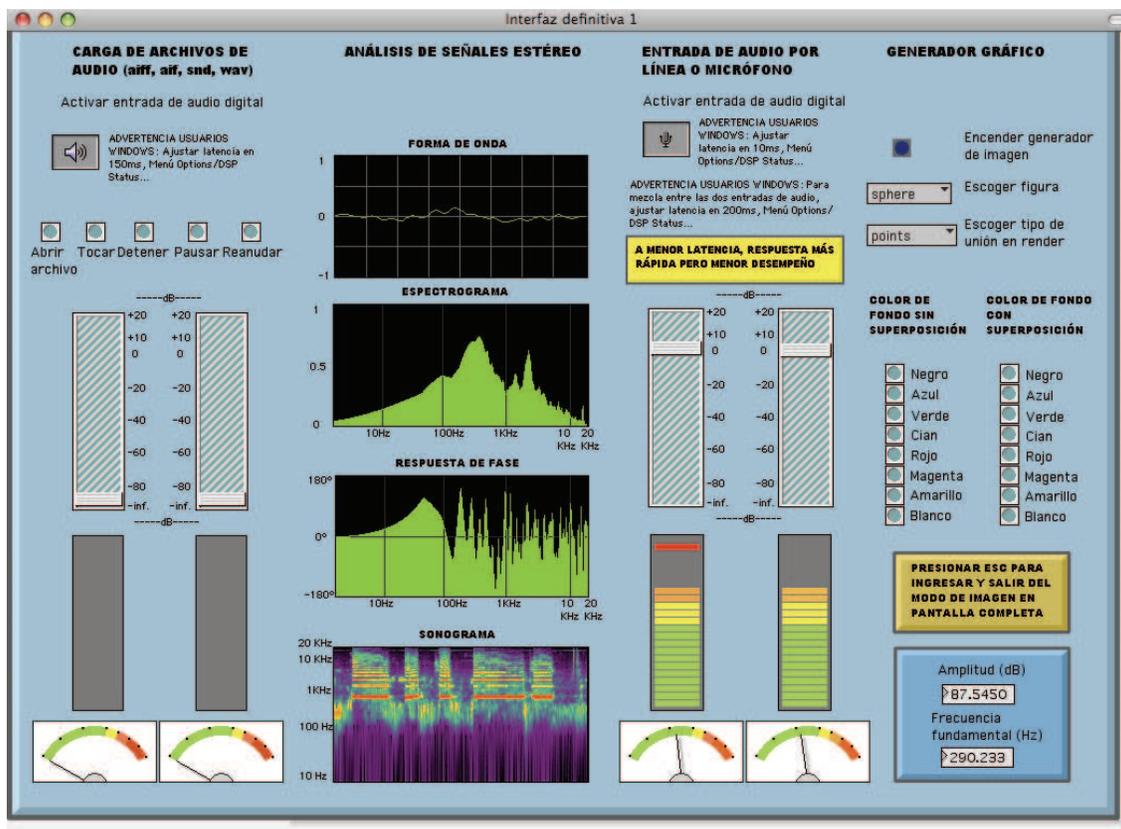


Figura 4.35: Interfaz de usuario definitiva bajo el sistema operativo Mac OSX.

4.3 Pruebas y análisis de resultados

Se procede a continuación a analizar los resultados tomando en cuenta las analogías planteadas en el modelo de esta tesis (Figura 4.1), que conforman la parte medular de este *software*. Es importante recalcar que las pruebas son realizadas con las figuras que aparecen en la configuración por defecto con que el programa inicia, la cual es una forma de circunferencia, existiendo muchas otras opciones para escoger.

Y las señales de audio usadas son ingresadas por la opción de apertura de archivos de audio, los mismos que fueron previamente generados. Las imágenes pueden sufrir modificaciones a partir de su estado original en el *patch* o interfaz de usuario debido a que se encuentran en movimiento algunos elementos el instante de la captura de la imagen. De igual manera todas las pruebas fueron realizadas en una interfaz de usuario provisional, presente en todas las imágenes de análisis de resultados, pero que puede sufrir cambios

luego del proceso de pruebas al que será sometida, para adquirir luego una configuración definitiva.

4.3.1 Amplitud vs. Tamaño

La amplitud de la onda de entrada controla eficientemente el tamaño de las figuras a proyectarse, se encuentran en una relación de escalas en donde el sonido es evaluado cada 6 dB y el tamaño de las figuras empieza con un tamaño que se podría decir que son prácticamente invisibles, avanzando en tamaño mientras duplica sus medidas hasta llegar a figuras extremadamente grandes que llenan por completo la pantalla de proyección del programa. La amplitud de la onda se observa en el cuadro referente a amplitud en dB, a pesar de esto puede ser tan baja que los medidores visuales de amplitud en ocasiones no arrojen medidas de la misma. Se realiza esta prueba con una onda senoidal de 1 KHz que varía su amplitud para estos propósitos ilustrativos, lo que da las siguientes imágenes:

- **0 dB:** Claramente en estas condiciones no existe tamaño alguno de las figuras, este es el umbral de audición humana.

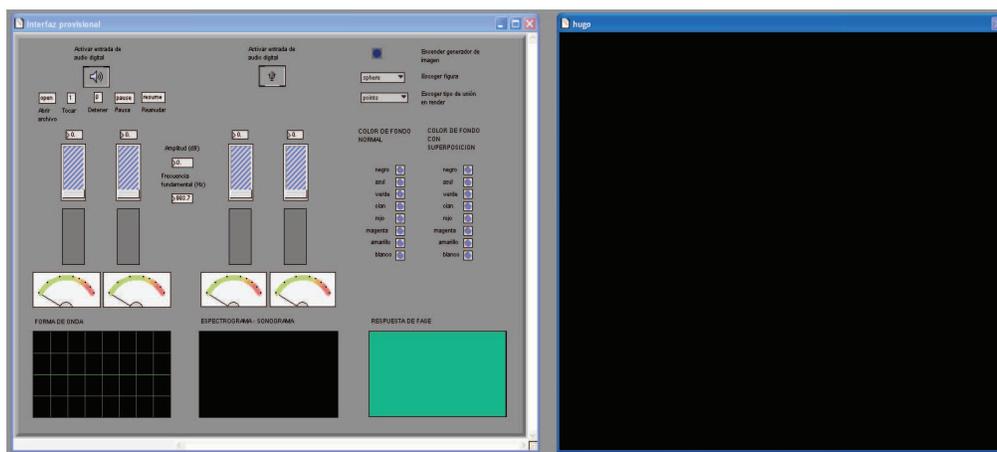


Figura 4.36: Imagen en respuesta a 0 dB de NPS.

- **20 dB:** Amplitud aún poco apreciable por el oído humano, genera formas que son prácticamente invisibles a pesar de ya poseer un tamaño asignado para esta amplitud. Se hace alusión a los 20 dB, que a pesar de tener ya una diferencia con la presión atmosférica de

referencia es aún demasiado pequeña para ser captada por la mayoría de los seres humanos en condiciones normales. Figura 4.37, Anexos.

- **40 dB:** Es posible ver muy pequeñas circunferencias verdes que son generadas a partir de esta amplitud, si bien la forma de onda no aún es visible en el osciloscopio de la interfaz el espectrograma. Figura 4.38, Anexos.
- **60 dB:** A este nivel ya es visible la generación de circunferencias con mayores dimensiones de acuerdo al NPS que maneje el *software* a su entrada. Figura 4.39, Anexos.
- **80 dB:** Se observa incluso que la forma de onda ya aparece en el objeto scope~ y activa los medidores visuales.

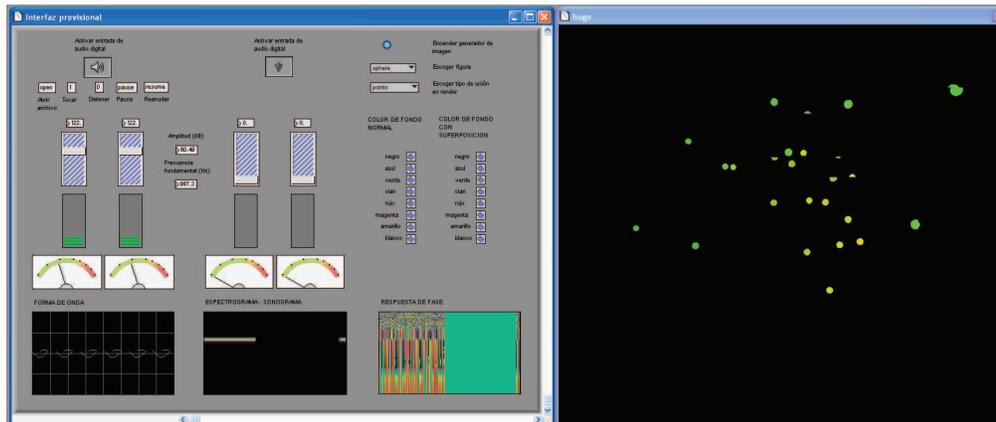


Figura 4.40: Imagen en respuesta a 80 dB de NPS.

- **100 dB:** Ahora es ya perfectamente apreciable las reacciones del *software* ante esta amplitud de onda, los medidores refieren a un nivel que está cerca de causar distorsión digital. Figura 4.41, Anexos.
- **120 dB:** La distorsión digital es muy fuerte, sin embargo el programa es capaz de determinar limpiamente una frecuencia fundamental y dibujar una superficie lisa correspondiente a una onda senoidal. Por medio de otros *software* generadores de onda resulta muy complicado ir más allá de los 120 dB, por lo que la medición de amplitud termina aquí, sin embargo el programa responde hasta los 174 dB en caso de ser necesario. A esos niveles de presión sonora tan grandes ya es visible

sólo partes de los contornos de las figuras ampliadas a escalas enormes.

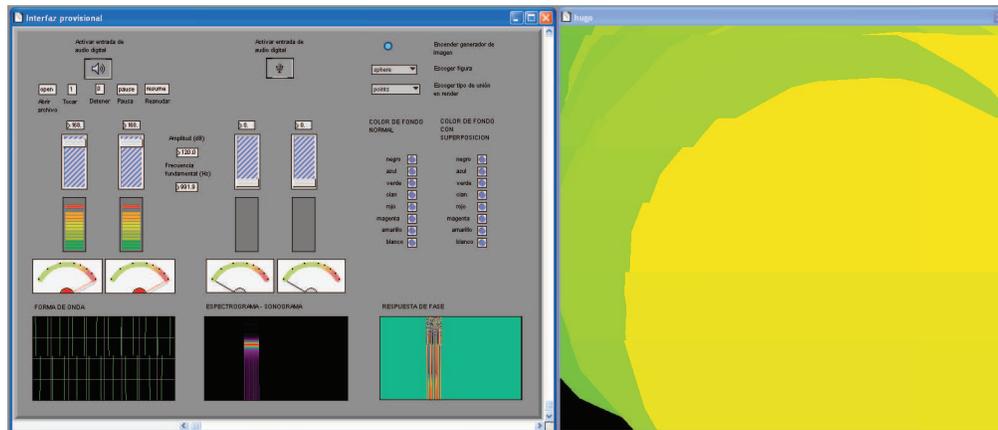


Figura 4.42: Imagen en respuesta a 120 dB de NPS.

4.3.2 Frecuencia vs. Color

La frecuencia está dividida en 31 bandas, tomando en consideración la división por tercio de octava comúnmente usada en ecualizadores profesionales. Las frecuencias más bajas están relacionadas con los colores más apegados al rojo y las frecuencias altas al violeta como está planteado en las analogías iniciales del programa, sin embargo en el rango de frecuencias muy bajas o de las primeras bandas es complicado con generadores de onda convencionales generar un tono puro de tan baja frecuencia, y el programa responde con imágenes que muestran que aparte de ese tono puro que se quiere lograr hay otras componentes presentes que no posibilitan un tono puro de baja frecuencia.

Los tonos de muy baja frecuencia pueden ser captados por un sistema de altavoces y microfónia que trabaje con aquellos rangos de frecuencia, más no con un sencillo programa generador de tonos; por lo que esta medición empezará ascendiendo por octavas desde los 125 Hz, frecuencia a la cual el generador no produce componentes indeseadas aparte de su frecuencia de oscilación. Sin embargo al trabajar con intervalos de octavas superiores el *software* puede tender a generar colores similares a ciertos instantes, lo que se

puede producir por elementos de frecuencia indeseados que en ocasiones superen a la fundamental y que estén presentes en una onda que no es perfectamente senoide pero se aproxima a ello.

- **125 Hz:** Es posible observar cómo se generan figuras circulares lisas que tienen contenido de colores anaranjados, correspondientes a este rango de frecuencias.

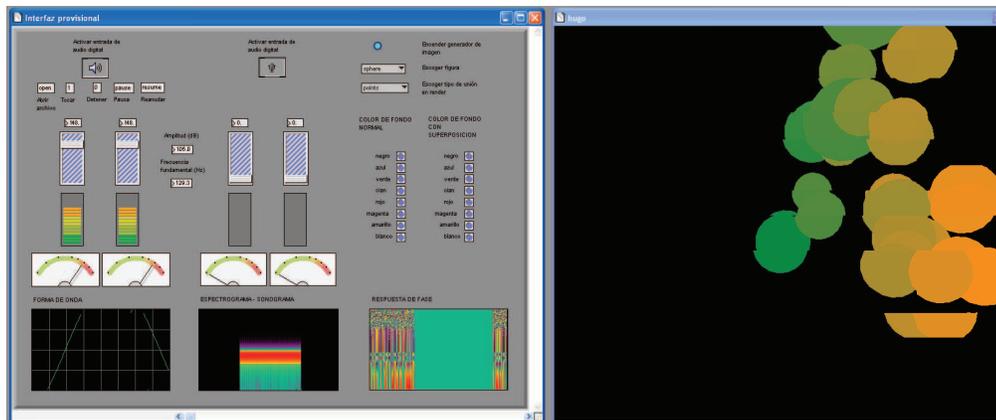


Figura 4.43: Imagen en respuesta a 125 Hz.

- **250 Hz:** A esta frecuencia se nota cómo la tendencia general de los colores va cambiando hacia el verde, en frecuencias relativamente bajas los generadores usados para estas pruebas no son capaces de producir una onda de perfecta pureza por lo que existen ligeras combinaciones de los colores correspondientes a rangos de frecuencias con tonalidades diferentes como por ejemplo los verdes, que responde a la presencia de "algo más" en cuanto a componentes de frecuencia. Figura 4.44, Anexos.
- **500 Hz:** En esta onda, el color amarillo resalta entre las figuras que es el color perteneciente a este rango, sin embargo los verdes son cada vez más pronunciados al ascender en frecuencia. Figura 4.45, Anexos.
- **1000 Hz (1 KHz):** A esta frecuencia los colores se definen ya hacia el verde con muy pocos rasgos de amarillo o colores ajenos a este rango.

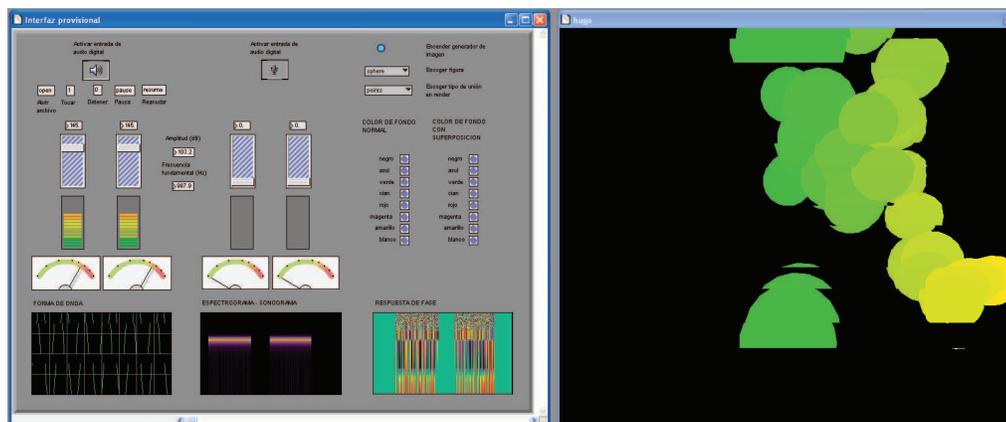


Figura 4.46: Imagen en respuesta a 1000 Hz.

- **2000 Hz (2 KHz):** En esta imagen se observa cómo el intervalo de frecuencias entre octavas es más grande mientras se asciende en ellas. A este rango se le asignan colores donde predomina el azul y tonalidades similares. Figura 4.47, Anexos.
- **4000 Hz (4 KHz):** Los azules y algunos colores relacionados se van acentuando mientras avanza la frecuencia hasta este rango.

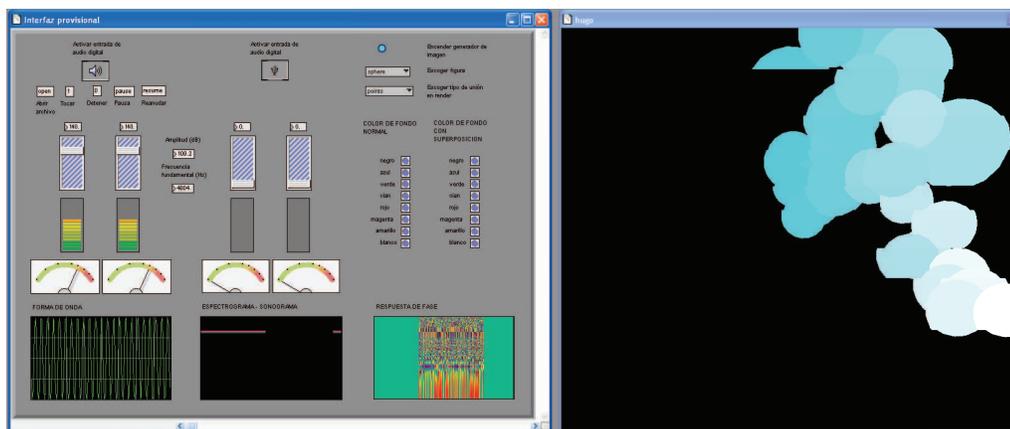


Figura 4.48: Imagen en respuesta a 4000 Hz.

- **8000 Hz (8 KHz):** El azul empieza a perder fuerza y muy levemente cambia su tonalidad, sin embargo se asemeja a la octava anterior por posibles componentes ajenas a la onda que no es perfectamente senoidal. Al trabajar con intervalos de frecuencia que no guarden

relaciones de octavas es más fácil observar los cambios de color. Figura 4.49, Anexos.

- **16000 Hz (16 KHz):** Se observa claramente lo abrupto del cambio entre la anterior octava y la actual, este rango de colores violetas corresponde al último escalón de la relación frecuencia/color.

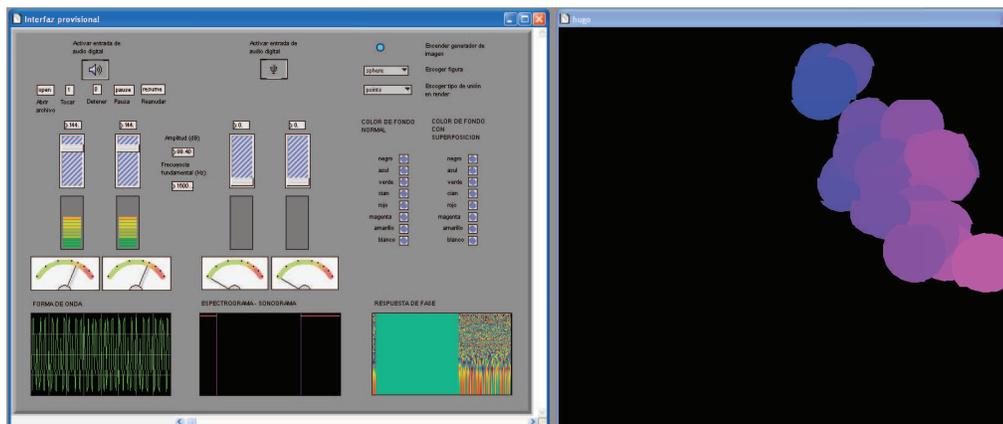


Figura 4.50: Imagen en respuesta a 16000 Hz.

4.3.3 Espectro vs. Textura

En el rango de frecuencias medias e incluso hasta los límites superiores del espectro audible no existe mayor problema para generar tonos senoidales de hasta 20 KHz con generadores tipo *software*. De manera similar a la medición de amplitud, se trabajó con una onda de frecuencia fundamental de 1 KHz, sobre la cual se fue construyendo ondas complejas mediante superposición de componentes senoidales, con los armónicos correspondientes a cada relación espectro/textura:

- **Frecuencia fundamental (1 KHz):** Las figuras adquieren superficies completamente lisas, asemejando la pureza de una onda perfectamente senoidal, se asemejan a membranas circulares en dos dimensiones, sin embargo pueden ser manipuladas por el usuario con el *mouse* en cualquier dirección dentro de los tres ejes de coordenadas (x, y, z).

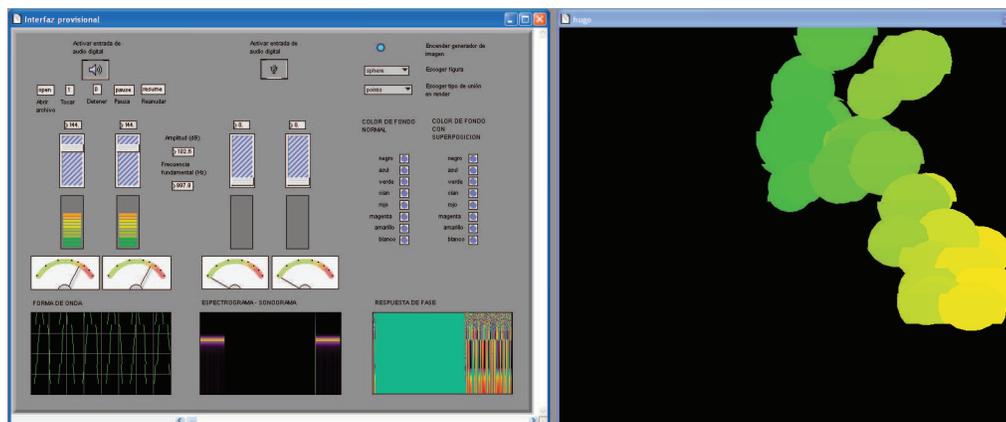


Figura 4.51: Imagen en respuesta a un tono puro de 1000 Hz.

- **Armónicos 2, 4, 8, 16 (2 KHz, 4 KHz, 8 KHz, 16 KHz):** Las formas adquieren profundidad y características tridimensionales que refuerzan la figura cuando la señal presenta armónicos en relación de duplicación con la fundamental, se da también un cambio de colores debido a que intencionalmente los armónicos tienen igual o mayor amplitud que la fundamental de 1 KHz, por lo que el programa realiza modificaciones al color al asumir ese fenómeno de similitud de amplitudes entre los diferentes armónicos.

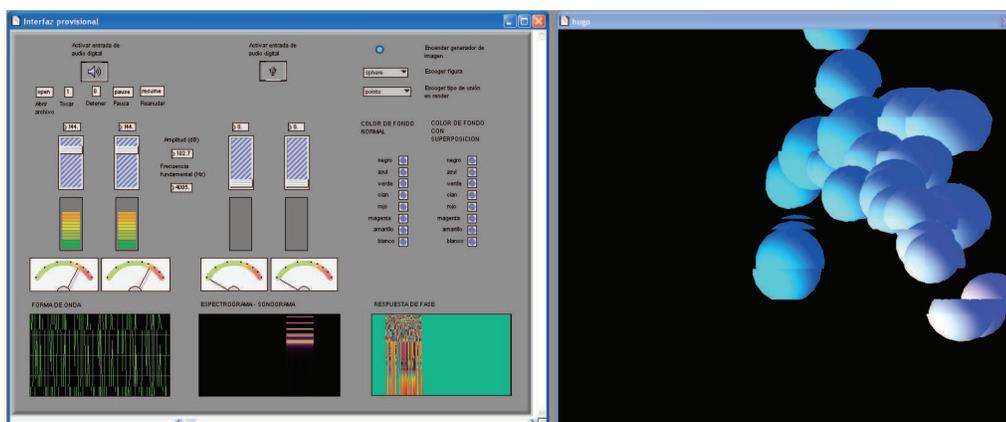


Figura 4.52: Imagen en respuesta a un tono puro de 1000 Hz, más sus armónicos 2, 4, 8 y 16.

- **Armónicos 5, 10 (5 KHz, 10 KHz):** La iluminación en las figuras proviene de ondas que contienen mayor presencia de los armónicos 5 y

10 que refuerzan la “calidez” del sonido y esto tienen relación con el hecho de que las figuras adquieren mayor “profundidad”. Figura 4.53, Anexos.

- **Armónicos 3, 6, 9, 12 (3 KHz, 6 KHz, 9 KHz, 12 KHz):** Al trabajar con ondas donde los armónicos 3, 6, 9, 12 están presentes se aprecia claramente como las figuras se suavizan creando la ilusión de ser objetos enteramente tridimensionales y de suaves superficies, lo que corresponde a esa sonoridad un tanto "nasal" que puede crear una cierta "suavidad" sonora. Figura 4.54, Anexos.
- **Armónicos 7, 11, 13, 14, 15 (7 KHz, 11 KHz, 13 KHz, 14 KHz, 15 KHz):** Las figuras se deforman completamente cuando el analizador de audio detecta la presencia de los armónicos 7, 11, 13, 14, 15. Igualmente el software escoge el color para una onda que está formada por diferentes componentes de idéntica amplitud cada uno, por lo que el resultado de color puede variar ligeramente respecto a la repetición de la medición. Figura 4.55, Anexos.

Se puede observar por ejemplo cómo las imágenes en presencia de una onda compleja que varía en el tiempo proporciona diferentes patrones de figuras geométricas y texturas al igual que cambios de color.

- **Ruido blanco:** La imagen se distorsiona en extremo, al punto de no poder reconocer con claridad la figura geométrica inicial de una circunferencia o esfera, debido a la cantidad de componentes de frecuencia que posee este tipo de onda. Figura 4.56, Anexos.
- **Ruido rosa:** Debido al mayor contenido energético en bandas de baja frecuencia la imagen resultante del ruido rosa tiende un poco más hacia los colores anaranjados. Figura 4.57, Anexos.

La interfaz de usuario incluye opciones de cambio de figuras, la manera cómo el *render* construye las estructuras y la opción de construir imágenes mediante la superposición de figuras con diferentes colores de fondo. Estas características son puestas a prueba y descritas a continuación.

4.4 Diferentes figuras geométricas y sus principales modos de construcción

Ahora se describen las principales opciones gráficas del programa, sin embargo existen otras que el usuario podrá escoger al momento de operar el programa.

Estas figuras son construidas a partir de diferentes estructuras presentes en los objetos del generador de imágenes, y la unión que se realiza entre los puntos de esas estructura es posible cambiarla mediante control por parte del usuario, modificando profundamente la apariencia de la imagen resultante; es importante recalcar que el programa por defecto une las imágenes mediante una superficie continua como se observó en el análisis de resultados entre las analogías amplitud/tamaño, frecuencia/color y espectro/textura.

Esto se modifica a unión por puntos, líneas, triángulos, cuadrados, polígonos, entre otros. Se describirá esta característica del *software* de acuerdo a la figura y sus principales opciones de unión entre sus puntos de construcción.

Se asumirá que estas figuras son generadas por una onda compleja, es decir con armónicos variables y hasta aleatorios, anteriores pruebas al programa fueron realizadas en su mayor parte con tonos puros que generaban formas lisas y planas en dos 2D, ahora para apreciar los cambios planteados en esta parte del análisis de resultados se trabajará con señales provenientes del micrófono para obtener mayor aleatoriedad.

4.4.1 Esferas (sphere)

El *software* realiza la construcción de esferas con superficies lisas y continuas de unión, esta opción viene por defecto, sin embargo las principales opciones incluyen:

- **Puntos (points):** Se observa las estructuras pero conformadas por puntos.

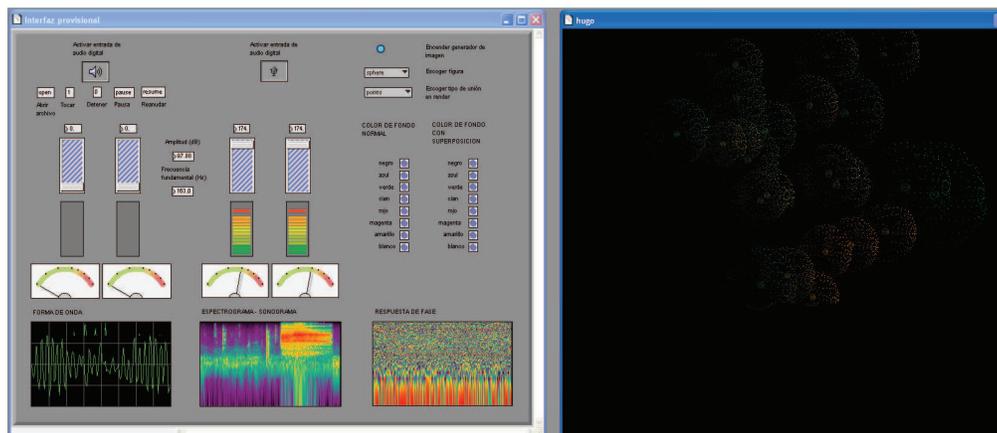


Figura 4.58: Ilustración de una estructura esférica con unión conformada por puntos.

- **Líneas (lines):** Las figuras son construidas con líneas, y existen algunas variantes como line_strip (tira de líneas) y line_loop (bucle de líneas).

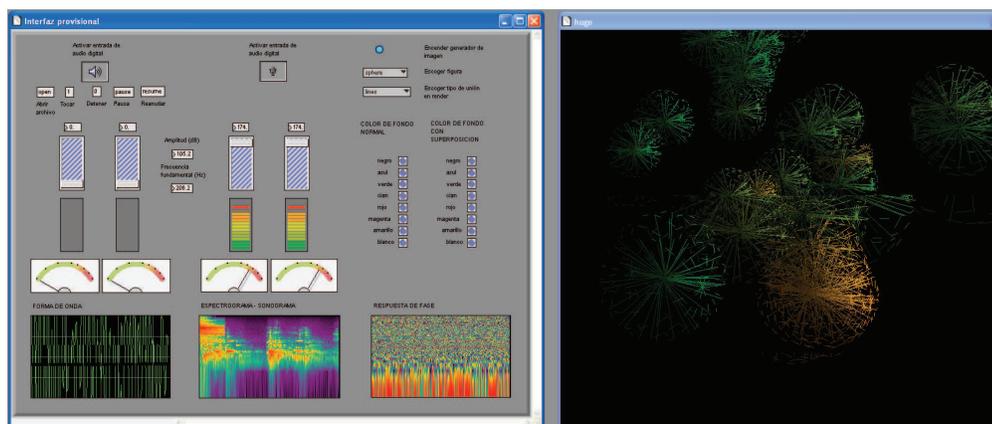


Figura 4.59: Ilustración de una estructura esférica con unión conformada por líneas.

- **Triángulos (triangles):** Se usa triángulos para unir los puntos de las estructuras visuales, existen variantes como tri_strip (tira de triángulos), tri_fan (hélice de triángulos) y tri_grid (grilla de triángulos).

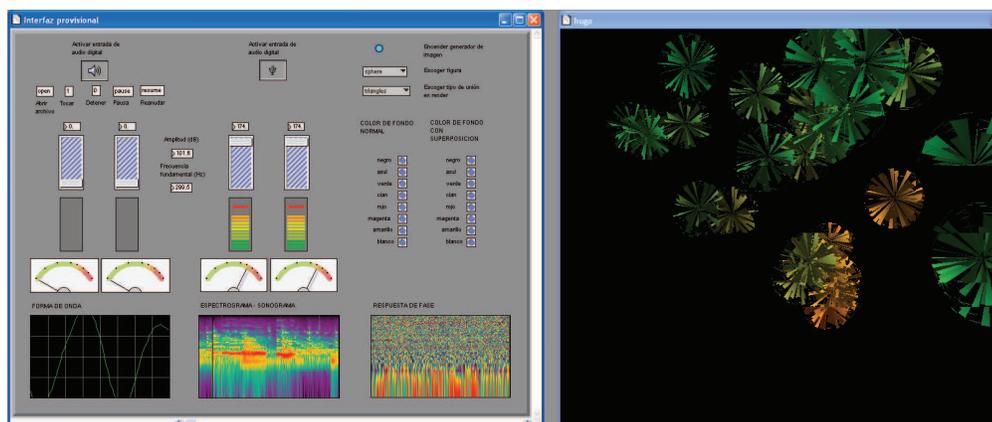


Figura 4.60: Ilustración de una estructura esférica con unión conformada por triángulos.

- **Cuadrados (quads):** Las figuras son construidas mediante una especie de cuadrados y rectángulos, también una variante quad_strip (tira de cuadrados).

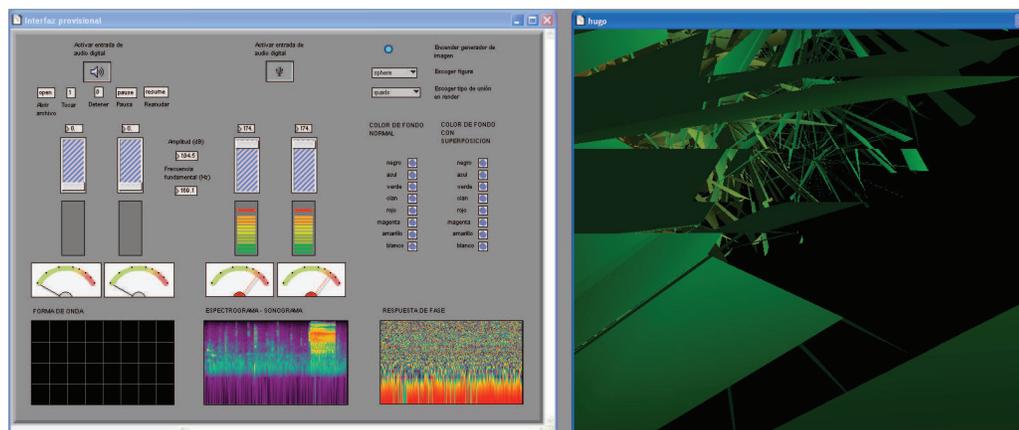


Figura 4.61: Ilustración de una estructura esférica con unión conformada por cuadrados.

- **Polígonos (polygon):** La unión se realiza mediante polígonos de más de 5 lados, aleatoriamente. Lo que confiere un comportamiento poco predecible en cuanto a la manera de cómo se van uniendo las figuras mientras el programa genera diferentes texturas dependiendo del contenido armónico diverso que una señal pudiera tener.

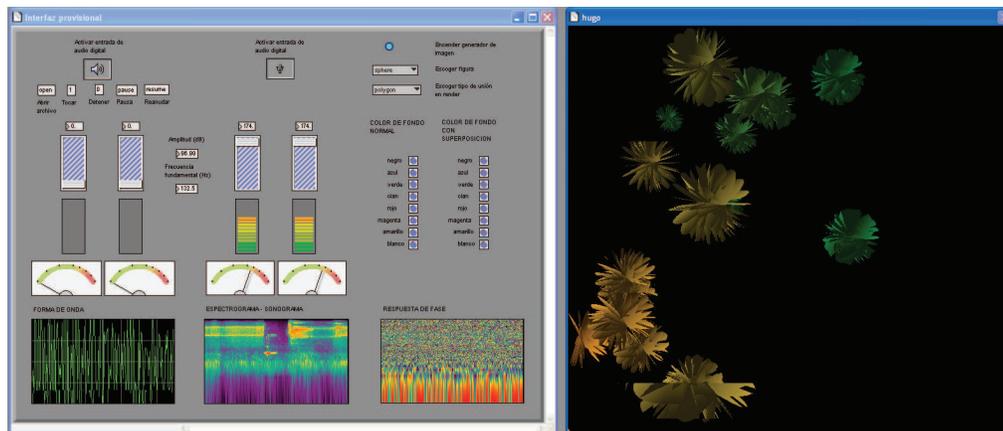


Figura 4.62: Ilustración de una estructura esférica con unión conformada por polígonos aleatorios.

4.4.2 Toroide (torus)

La figura toroidal conforma una especie de anillo o aro, que es parte de las opciones del *software*, y las imágenes derivadas por las opciones de unión son:

- **Puntos (points):** Figura 4.63, Anexos.
- **Líneas (lines):** Figura 4.64, Anexos.
- **Triángulos (triangles):** Figura 4.65, Anexos.
- **Cuadrados (quads):** Figura 4.66, Anexos.
- **Polígonos (polygon):** Figura 4.67, Anexos.

4.4.3 Cilindros (cylinder)

Los cilindros que el programa es capaz de generar son principalmente sólidos, aunque viene la opción secundaria de realizar cilindros huecos, a lo que el *software* llama "opencylinder".

Las opciones de unión son las mismas que las anteriores figuras sin embargo los resultados son diferentes:

- **Puntos (points):** Figura 4.68, Anexos.
- **Líneas (lines):** Figura 4.69, Anexos.
- **Triángulos (triangles):** Figura 4.70, Anexos.

- **Cuadrados (quads):** Figura 4.71, Anexos.
- **Polígonos (polygon):** Figura 4.72, Anexos.

4.4.4 Cubo (cube)

Los cubos igualmente incluyen la posibilidad de ser sólidos o de manera opcional, vacíos; a lo que el *software* denomina "opencube". Las opciones son las siguientes:

- **Puntos (points):** Figura 4.73, Anexos.
- **Líneas (lines):** Figura 4.74, Anexos.
- **Triángulos (triangles):** Figura 4.75, Anexos.
- **Cuadrados (quads):** Figura 4.76, Anexos.
- **Polígonos (polygon):** Figura 4.77, Anexos.

4.4.5 Planos cuadrados (plane)

Los planos son formados como cuadrados de dos dimensiones, pero al rotar con el *mouse* el campo de generación de imágenes pueden adquirir características tridimensionales a manera de membranas o láminas que también recorren los tres ejes del espacio.

Las opciones de cambio de tipo de unión entre los puntos de la estructura son los mismos que anteriores formas geométricas:

- **Puntos (points):** Figura 4.78, Anexos.
- **Líneas (lines):** Figura 4.79, Anexos.
- **Triángulos (triangles):** Figura 4.80, Anexos.
- **Cuadrados (quads):** Figura 4.81, Anexos.
- **Polígonos (polygon):** Figura 4.82, Anexos.

4.5 Composición de nuevas imágenes mediante superposición

Como ha sido mencionado a lo largo del desarrollo del proyecto, también se incluye la opción de crear imágenes diferentes mediante superposición de las figuras geométricas que el programa genera.

Esto se logra haciendo que las figuras "antiguas" no se extingan en algún tiempo, hasta que las nuevas se dibujen encima y así sucesivamente.

Se han realizado arreglos para que esas superposiciones sean también vistas sobre varios colores de fondo, se muestra cada color con imagen normal y posteriormente superpuesta, donde el fondo tiene el efecto de desvanecimiento en el tiempo hacia el color negro, por lo que al momento de capturar la imagen puede aparecer más oscuro el fondo.

Igualmente para acceder a un fondo de cualquier color con superposición se debe, en primer lugar escoger el fondo sin superposición de imágenes.

Se trabajará por propósitos estéticos con formas toroides con tipo de unión por puntos y líneas y serán mostradas sólo algunas de las opciones de color de fondo en virtud de la cantidad de posibilidades existentes en la interfaz de usuario.

- **Fondo negro:**

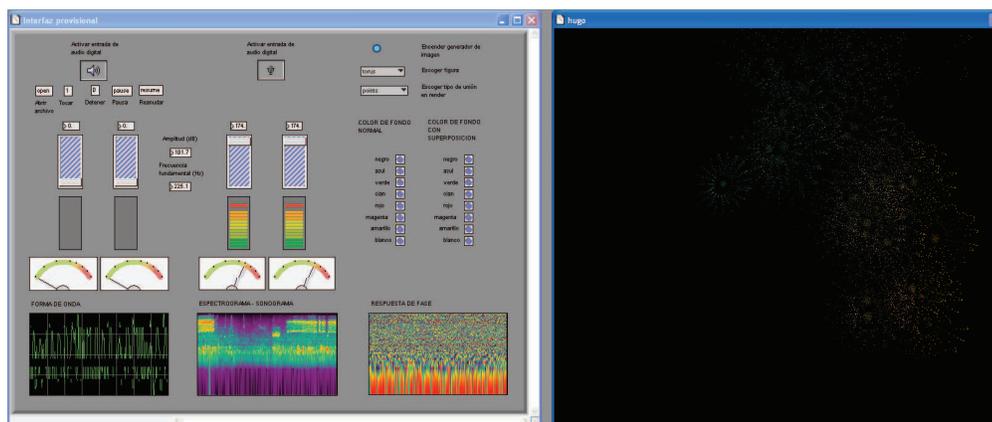


Figura 4.83: Ilustración de la opción de superposición de imágenes desactivada y con un fondo de color negro.

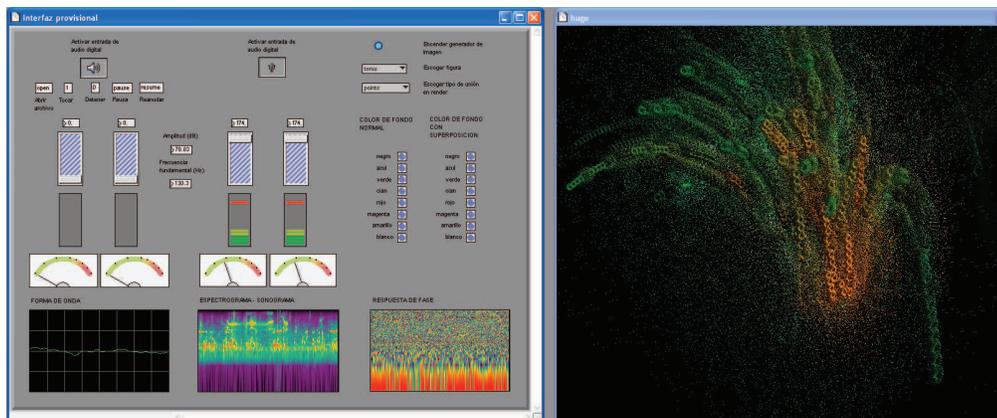


Figura 4.84: Ilustración de la opción de superposición de imágenes activada y con un fondo de color negro.

Fondo azul: Figura 4.85 y 4.86, Anexos.

4.6 Imágenes resultantes de timbres definidos

Al hablar de timbre definido en este subcapítulo, se hace referencia de manera directa a un instrumento musical. En virtud de lo familiar que le resulta al ser humano encontrar sonoridades específicas cuando los mismos son interpretados y hacer calificaciones subjetivas relacionadas a ese sonido. Los archivos usados para esta prueba del programa se encuentran formato WAV, frecuencia de muestreo de 44.1 KHz, 16 y 24 bits; todos archivos estéreo.

- **Flauta Traversa:** Pequeña muestra de audio de una flauta traversa interpretando un pasaje, ingresando al momento de la captura de imagen con amplitud de 87 dB y frecuencia fundamental de 904 Hz.

Se puede notar muy claramente la presencia de ciertos armónicos relativos al número 5 (fundamental, 5º armónico, 10º), por la textura de la superficie un tanto rugosa de las esferas, a la que está asociada esa presencia armónica evidenciando sonoridades llenas de "calidez" y sonidos "redondos" y "suaves".

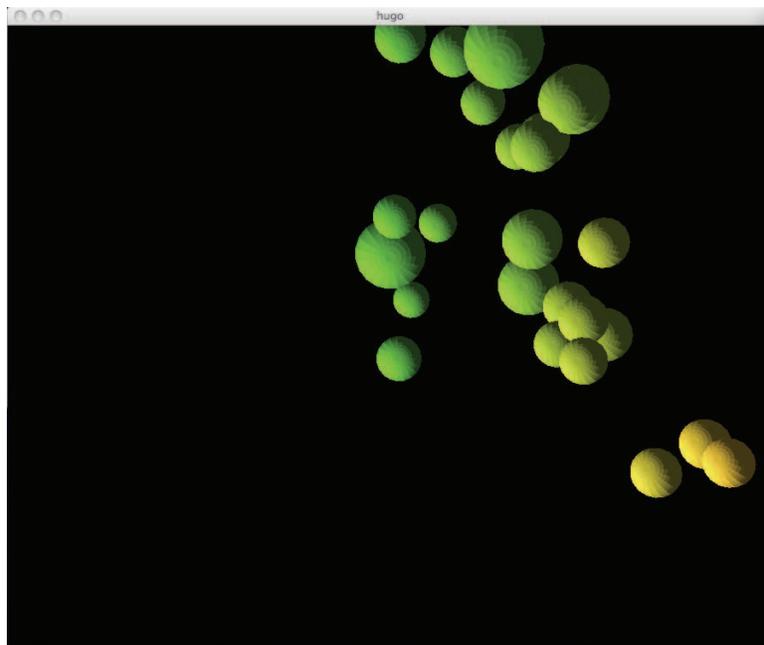


Figura 4.87: Imagen generada por la acción de la interpretación de una flauta travesa como instrumento solista.

- **Trompeta:** Esta muestra es de una trompeta interpretando un pequeño pasaje, al momento de la captura de imagen el instrumento emite una onda con amplitud de 88 dB y frecuencia fundamental de 805.6 Hz.

La forma tridimensional con grandes segmentos de superficie lisa de los toroides sugiere presencia de ciertos armónicos relacionados al número 2 (fundamental, 2º armónico, 4º, 8º, 16º), los cuales refuerzan el carácter tonal de la frecuencia fundamental y generan estructuras que refuerzan su forma con tridimensionalidad.

Las pequeñas irregularidades en los contornos de las toroides sugieren la muy sutil existencia en ciertos instantes de la interpretación de componentes senoidales distintas a los armónicos 2, 4, 8 y 16; principalmente debido a sonidos propios de la interpretación producidos por elementos como la embocadura del instrumento y los pistones.

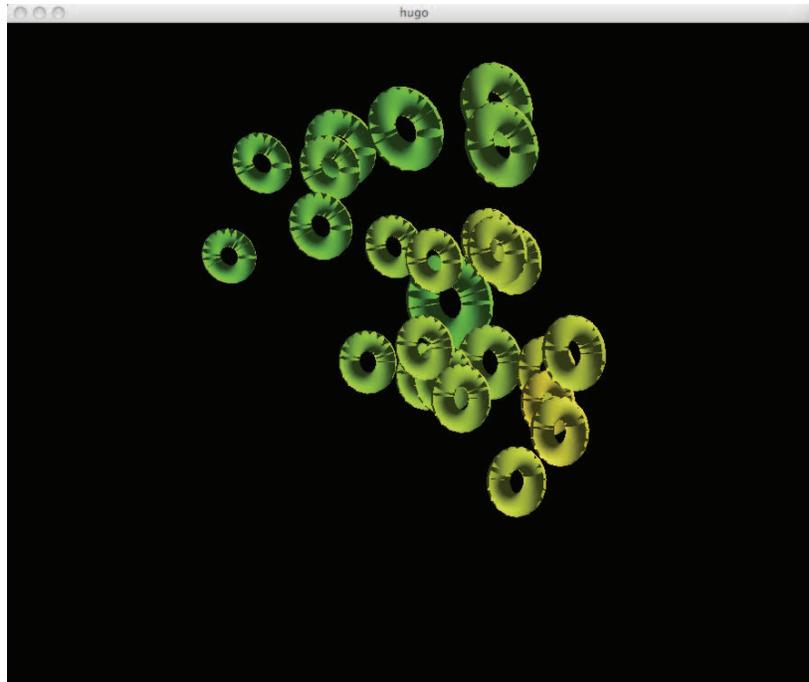


Figura 4.88: Imagen generada por la acción de la interpretación de una trompeta como instrumento solista.

- **Redoblante:** Instrumento emitiendo al momento de la captura de imagen un sonido con amplitud de 92 dB y frecuencia de 289.6 Hz.

La muestra es de un redoblante con un sonido agudo por lo que posee gran cantidad de componentes armónicas de frecuencia alta y de allí su coloración que tiende al verde.

Sin embargo la frecuencia que predomina se manifiesta en el color más anaranjado del cilindro con superficie irregular de la parte inferior de la imagen.

Al tratarse de un instrumento de percusión puede tener en un mismo instante muchas componentes armónicas, de ahí surge el hecho de que la imagen se torne de superficie muy irregular.

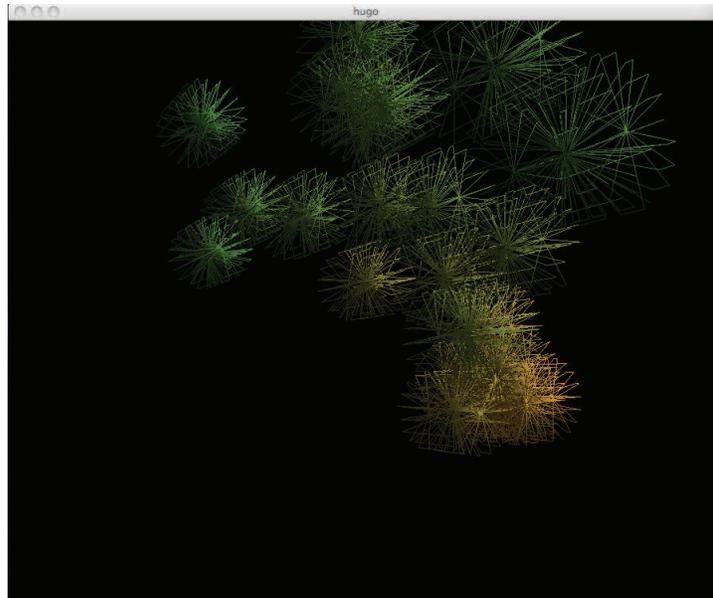


Figura 4.89: Imagen generada por la acción de la interpretación de un redoblante solo.

- **Violín:** Al momento de la captura de la imagen el violín emite una nota G6 con amplitud de 92 dB y frecuencia 1562.4 Hz. Una presencia de armónicos pares e impares (parecido a una onda diente de sierra), causa que la imagen se modifique y la forma de un cubo no se observe con estricta claridad.

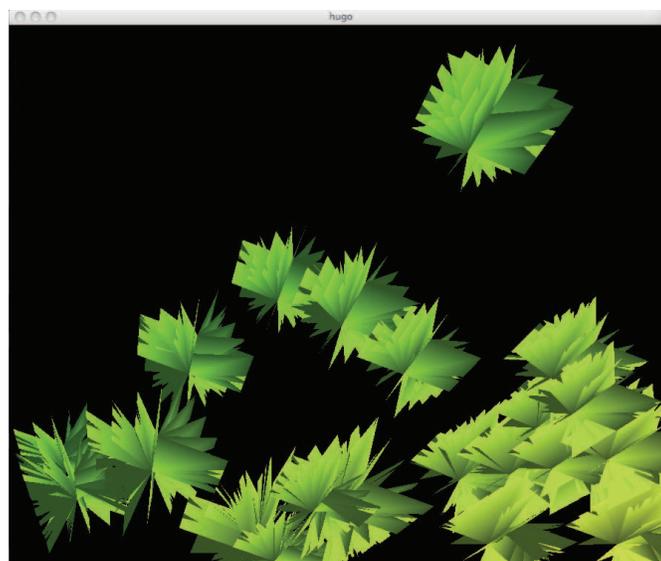


Figura 4.90: Imagen generada por la acción de la interpretación de un violín como instrumento solista.

4.7 Imágenes resultantes de estilos musicales definidos

A continuación se ingresa al programa tres archivos de audio que contienen música de diferentes estilos, para observar la reacción del generador de imágenes ante cambios respecto a género musical. Estas imágenes que a continuación se presentan cambian constantemente a lo largo de la música, por lo tanto no constituyen imágenes decisivas sobre la respuesta del programa ante estilos musicales diferentes, sino que dependen de factores diversos como número de instrumentos o voces en la interpretación, amplitud, frecuencia y espectro del instante en que fue capturada la imagen, entre otros. Los archivos de audio usados se encuentran en el formato WAV, frecuencia de muestreo de 48 KHz, 16 bits; todos archivos estéreo.

- **Guitarra clásica (Música Latinoamericana):** Imagen obtenida cuando ingresa una guitarra solista interpretando música latinoamericana (Seis por Derecho del compositor venezolano Antonio Lauro, interpretado por John Williams), se generan formas suaves y de poco tamaño evidenciando la fuerza relativamente pequeña del sonido y la suavidad de la interpretación de ciertos momentos de esta obra en la guitarra clásica. La imagen está generada sin el modo de superposición de figuras.

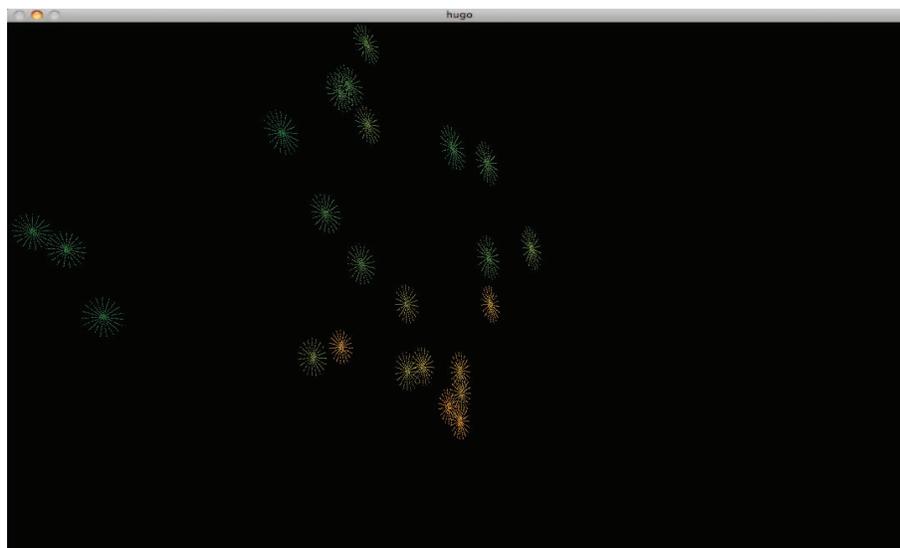


Figura 4.91: Imagen generada por la acción de la interpretación de música latinoamericana en una guitarra solista.

- **Conjunto instrumental (Flamenco):** Al existir una variedad de instrumentos en escena se multiplican los sonidos presentes y en cualquier instante las condiciones armónicas de la señal son muy diferentes, al momento de la captura de la imagen se realiza un solo de guitarra usando notas del registro alto del instrumento. En esta imagen se aprecia colores diferentes que resultan del ingreso de una señal que contiene audio de un grupo musical conformado por guitarra flamenca, cantaores, bajo eléctrico, percusión variada, entre otros. En esta imagen es posible ver los colores que genera la guitarra cuando su ejecutante toca notas agudas; este instrumento incluso emitiendo notas graves tiene componentes agudas de frecuencia debido a la técnica del guitarrista de pulsar y en ocasiones golpear levemente las cuerdas con las uñas, en el rasgueo flamenco se lo hace con potencia y extrema rapidez. La obra usada fue Ciudad de las Ideas, compuesto e interpretado por Vicente Amigo y su conjunto. La imagen está generada sin el modo de superposición de figuras.

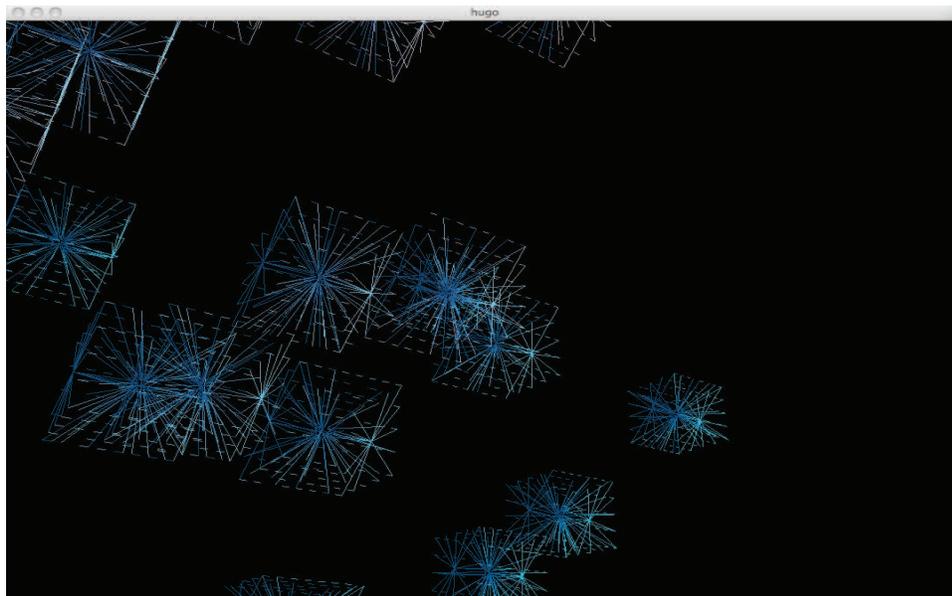


Figura 4.92: Imagen generada por la acción de la interpretación de música flamenca por parte de un conjunto instrumental y voces.

- **Orquesta Sinfónica (Música española):** Esta imagen es generada a partir del Allegro con Spirito del Concierto de Aranjuez, del compositor español Joaquín Rodrigo, interpretado en la guitarra por Paco de Lucía y la Orquesta Sinfónica de Cadaqués, dirigida por Edmon Colomer. Un movimiento de la obra caracterizado por la fuerza de algunos de sus momentos de interpretación en donde el instrumento solista dialoga con la orquesta en pasajes de extrema complejidad y virtuosismo, además de gran intensidad sonora. Al momento de la captura de la imagen se da una especie de cúspide en la interpretación, por lo que se aprecia imágenes que llenan el espacio dedicado para la imagen y gran cantidad de colores superpuestos, debido a que fue activada la opción de superposición de figuras en la interfaz de usuario.

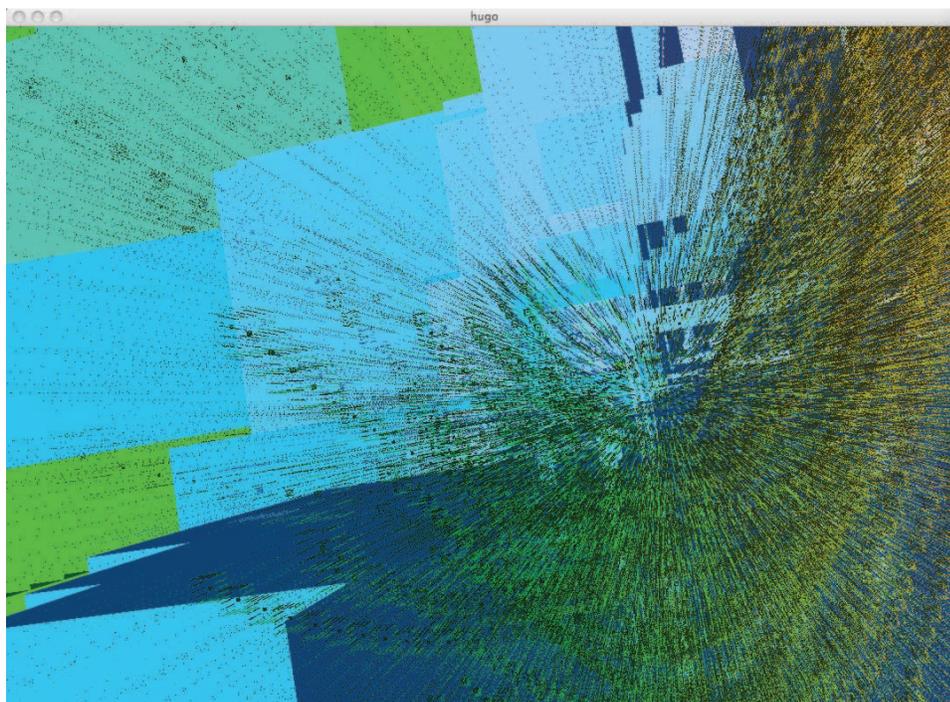


Figura 4.93: Imagen generada por la acción de la interpretación de música española por parte de una Orquesta Sinfónica e instrumento solista.

- **Banda (Metal):** El audio que generó esta imagen corresponde al tema First Commandment de la agrupación Soulfly, de su álbum homónimo. Interpretan música del género metal por lo que la imagen posee gran fuerza en términos de tamaño de las figuras, sin embargo se observa que la gama de colores no es tan extensa como el anterior ejemplo debido a que en aquel instante las guitarras y las voces comparten zonas cercanas del rango de frecuencias, el bajo eléctrico tiene cierta influencia en una de las pocas zonas de la imagen que difiere en color (zona superior). Los elementos centrales más oscuros que parecen nacer de un punto en especial son las figuras geométricas severamente deformadas por el contenido armónico de este tema del género metal. Opción de superposición de figuras activada.

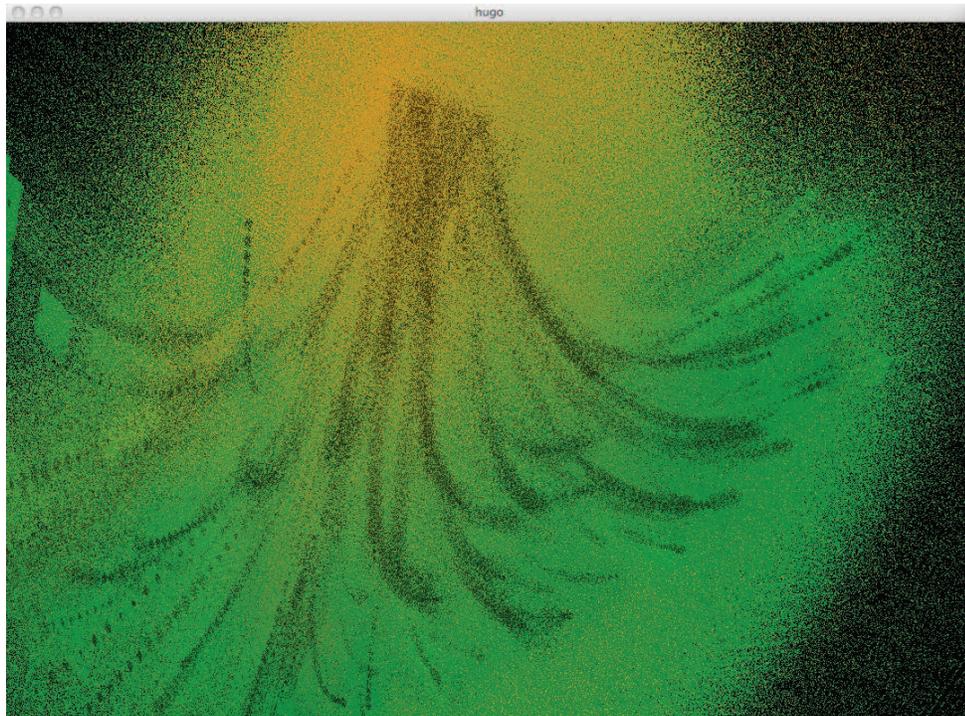


Figura 4.94: Imagen generada por la acción de la interpretación de música metal por parte de una banda instrumental y voces.

- **Banda (Indy Rock):** La agrupación ecuatoriana Totem interpreta el tema Dejar de ver. Es claramente apreciable que la amplitud se mantiene relativamente constante al momento de la captura de la imagen dando una sensación visual de profundidad, sin embargo se observan diferentes colores que muestran contenido armónico diverso en aquel instante de la interpretación. Los colores se hallan en una especie de perspectiva hacia el fondo, de donde nace la imagen; y se observa que las figuras geométricas están seriamente deformadas hasta llegar a una especie de "estrellas" que se agrupan unas con otras en un conjunto denso de aristas y puntas. Opción de superposición de figuras activada.

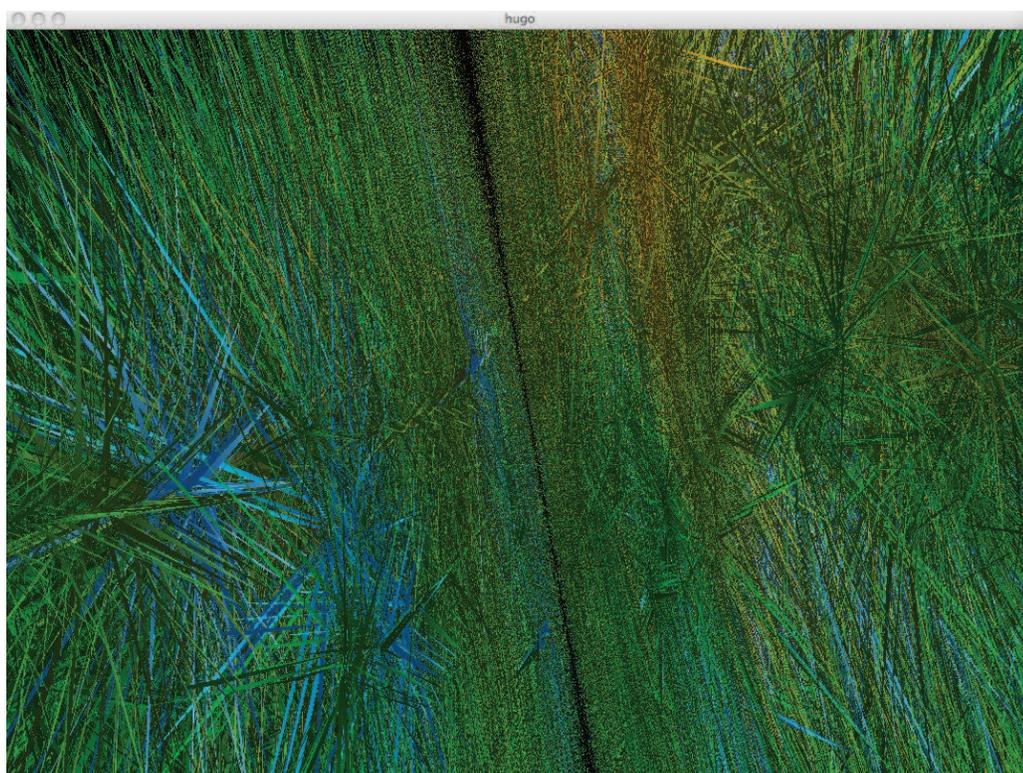


Figura 4.95: Imagen generada por la acción de la interpretación de música indy rock por parte de una banda instrumental y voces.

CAPÍTULO 5. ESTUDIO ECÓNOMICO

5.1 Estudio de costos

Este trabajo de titulación fue principalmente estructurado usando como herramienta principal un computador, textos o ficheros electrónicos y una conexión básica a Internet. No requiere mayores gastos de movilización, alquiler de equipos y materiales, entre otros; razón por la cual este presupuesto general es de considerable sencillez, el mismo que a continuación se detalla.

Tabla 5.1: Detalle de costos generales para la realización de este trabajo de tesis.

RUBRO	UNIDAD	CANTIDAD	PRECIO	TOTAL
Computador	u	1	700.00	700.00
Conexión a Internet	mes	6	18.00	108.00
Capacitación en entorno Max/MSP/Jitter	hora	80	3.00	240.00
<i>Software</i> Max/MSP/Jitter	u	700	1.00	700.00
Material bibliográfico	u	4	40.00	160.00
Asesoría adicional	hora	10	3.00	30.00
Impresión y conformación de documentos	hoja	300	0.15	45.00
Gastos varios	glb	1	100	100
			TOTAL	2083.00

La información es detallada a continuación:

- **Computador:** La totalidad del proyecto usa como herramienta una computadora, por lo que es imprescindible al momento de emprender un trabajo de esta naturaleza. Para asegurar un desempeño en tiempo real del programa se necesita un computador con características básicas

como procesador de 2 GHz, memoria RAM de 2 GB, tarjeta de video de 128 MB, tarjeta de sonido compatible con la norma ASIO.

- **Conexión a Internet:** La mayor parte de la información respecto a tópicos relacionados en el tema de audio, programación, imagen y *software* en general se encuentra en Internet por lo que se necesita una conexión a la red, de una velocidad promedio como las que actualmente se manejan en el mercado (256 Kbps, 500 Kbps).
- **Capacitación en entorno Max/MSP/Jitter:** El uso de Max/MSP se facilita al contar con un período de capacitación, esto reduce considerablemente el tiempo de construcción e implementación de un proyecto de esta naturaleza. La capacitación se centra básicamente en el conocimiento previo de los diferentes objetos y sus funciones, así como los campos donde el entorno se puede desempeñar. Una labor de capacitación incluiría el uso de los diferentes tutoriales que vienen anexos al instalar Max/MSP/Jitter en un computador, para de esta forma ir conociendo las capacidades del entorno mediante ejercicios prácticos respaldados por una adecuada documentación.
- **Software Max/MSP/Jitter:** Se considera la versión más actual y su respectivo costo para realizar el trabajo.
- **Material bibliográfico:** Algunos textos y ficheros electrónicos referentes a temas de sonido, imagen, procesamiento digital de señales, programación, entre otros.
- **Asesoría adicional:** En ocasiones, además de una adecuada capacitación se presentan circunstancias donde un profesional puede brindar valiosa ayuda técnica.
- **Impresión y conformación de documentos:** Se estima un total de hojas a ser impresas para la presentación de los distintos documentos.
- **Gastos varios:** Actividades anexas a la tesis, movilización, entre otros.

5.2 Relación costo beneficio

El costo de producción de este programa incluye algunos rubros que posiblemente podrían ya estar cubiertos como por ejemplo una computadora

de mediano desempeño que ahora es muy común tener en casa, además de una conexión a Internet básica que igualmente es común tenerla en el hogar; sin embargo fueron tomados en cuenta como una condición para construir el software asumiendo el costo de todos ellos.

Una posible limitación del programa es su condición de archivo no ejecutable y que por lo tanto requiera del entorno Max/MSP/Jitter y sus *externals* o *plug-in* para funcionar.

El beneficio que este *software* posee está básicamente concentrado en las prestaciones que ofrece respecto a otra aplicación relacionada. Pues es capaz de trabajar con señales de audio en tiempo real, sean éstas cargadas desde el disco duro o cualquier dispositivo de almacenamiento externo, o bien directamente adquiridas de la entrada de audio del computador por línea o micrófono; y no sólo con señales MIDI como por ejemplo el programa Music Animation Machine (MAM Player) de Stephen Malinowski, que posee muchas opciones de visualización gráfica muy interesantes pero está limitado a cargar archivos exclusivamente bajo el protocolo MIDI y la extensión *.mamx*. Además el programa producto de esta tesis brinda también la posibilidad de ser un analizador básico de audio conjuntamente con un generador de imágenes 3D y sus respectivas opciones gráficas.

El costo de desarrollo de este programa no es alto, considerando incluso el partir desde cero con todos los rubros detallados, algunos de los cuales podrían estar ya cubiertos. Y los beneficios radican en sus capacidades, prestaciones, opciones gráficas, opciones de análisis de audio, trabajo con diferentes entradas de audio y las posibilidades de inclusión de más herramientas que brinda el entorno en que fue creado.

CAPÍTULO 6. PROYECCIONES Y POSIBILIDADES DE EXPANSIÓN

6.1 Proyecciones

El *software* de este proyecto de tesis guarda relación directa con ambos mundos del sonido y la imagen, en el mercado nacional no es posible encontrar aplicaciones dedicadas similares; por lo que el programa puede proyectarse en el futuro hacia programaciones referentes al refuerzo sonoro, a la producción de eventos, entre otras actividades, como complemento a la parte lumínica o de escenografía, además de brindar paralelamente un analizador de audio básico que puede ser usado en cualquier momento.

Merecen mención los temas de la salud humana y la investigación de las formas de interactuar entre los diferentes sentidos, tal es el caso del oído y la vista que pueden tener potenciales nexos de conexión para influir en el uno por medio del otro y viceversa, con diferentes propósitos.

6.2 Posibilidades de expansión

El *software* posee posibilidades de expandirse en sus prestaciones o características, principalmente en su etapa de generación de imágenes y en la rapidez de los procesos. Al poder desarrollarse mejores métodos de construcción de imágenes y simplificación de pasos de programación se puede lograr un mejor desempeño del programa en la globalidad de sus procesos.

Métodos como la simplificación u optimización de la etapa de generación de imágenes, la división de los rangos de amplitud y frecuencia con otros objetos que puedan ser implementados, la reducción del número de objetos emisores de pulsos para así manejar un cuerpo más compacto de clasificación de información de audio y que trabaje de manera optimizada.

Prestaciones prácticas para el refuerzo sonoro como un asistente para determinar parámetros o niveles de compatibilidad mono-estéreo de la señal en tiempo real e incluso la posibilidad de un ecualizador dedicado para que al modificar la respuesta de frecuencia de la señal se afecte el comportamiento

del generador de imágenes en términos de color, al igual que actualmente se realiza en términos de amplitud/tamaño con los *faders*.

La adición de una salida de video independiente de la interfaz de control es una opción a considerar por la utilidad que puede brindar al momento de proyectar y querer modificar una imagen.

Una pantalla pequeña en la misma interfaz con el propósito de monitorear la imagen que se proyecta en el exterior es una posibilidad.

Opciones distintas a partículas tridimensionales que se desplazan por un espacio: tal como superficies oscilantes a manera de membranas, gráficos generados por fractales, imágenes que el usuario pueda cargar de sus archivos personales y modificarlas con opciones que van desde efectos tipo caleidoscopio hasta la mezcla de efectos de procesamiento de video.

CAPÍTULO 7. CONCLUSIONES Y RECOMENDACIONES

7.1 Conclusión General

Mediante la aplicación Max/MSP/Jitter que conforma un entorno para la programación ha sido posible desarrollar el *software* planteado que recoge como elemento principal de trabajo la vinculación existente entre el sonido y la imagen. El desarrollo del programa es capaz de guardar una relación constante entre los elementos comunes entre el espectro de luz visible y el rango de la audición humana. La relación de coherencia planteada en el objetivo general hace alusión a la construcción de un modelo que cumpla con las relaciones entre elementos comunes a ambos espectros de energía.

7.2 Conclusiones específicas

- Se encontraron las similitudes entre los aspectos de ambos tipos de energía (sonido y luz visible), y fueron relacionadas para poder construir el modelo de tesis planteado, dividiendo cada uno en tres aspectos fundamentales de su naturaleza.
- La relación amplitud/tamaño fue desarrollada dividiendo la amplitud de la señal de audio en intervalos de 6 dB, los mismos que fueron asociados a una escala relativa de tamaño de las figuras. La relación de aumento de tamaño de imagen frente a la del incremento del NPS es de 1:2, cada 6 dB la onda de audio aproximadamente cuadruplica su presión sonora, mientras que por cada aumento de 6 dB la imagen dobla su tamaño.
- La relación frecuencia/color se desarrolló dividiendo el dato proveniente del análisis de frecuencia fundamental en 31 bandas, y lo mismo fue planteado para el color, 31 colores que van desde el rojo relacionado con bajas frecuencias hasta el violeta que fue asignado a la última banda de frecuencias altas, con esto se concluye la relación casi linealmente a diferencia de amplitud/tamaño.
- La analogía de espectro/textura fue desarrollada tomando en cuenta las primeras 16 componentes que plantea la Serie Natural de Armónicos, agrupadas en conjuntos que determinan entre ellos distintas

sonoridades subjetivas, éstas son relacionadas directamente con texturas, también subjetivas, que el generador de imágenes construye. Se cumple con el objetivo de relacionar estos dos elementos de espectro/textura mediante la asignación directa de los datos sonoros al *patch* o módulo de generación de imágenes.

- Al contar con una relación de aspectos entre sonido e imagen como la planteada en esta tesis, es posible establecer una especie de interés por parte del usuario hacia las reacciones visuales que el programa tenga en presencia de ondas de audio, pues el modelo planteado obedece a una similitud técnica entre ambos tipos de energía, sonora y de luz visible, con lo que se cumple los objetivos del modelo de trabajo.
- La aplicación es capaz de manejar estas relaciones en tiempo real y con un buen desempeño, brindando la posibilidad de ingresar la señal de audio a la computadora desde cualquier destino y por ende al programa por medio de las entradas de audio normales o bien por un micrófono, cumpliendo con el propósito de ser una herramienta destinada al refuerzo sonoro dando algunas facilidades para la conexión entre los múltiples equipos o dispositivos y la computadora como generador.

7.3 Conclusiones técnicas y de desempeño del programa

Las conclusiones técnicas están directamente derivadas del análisis de resultados, por lo que serán divididas asemejando la estructura del análisis de resultados.

7.3.1 Conclusiones específicas

7.3.1.1 Amplitud vs. Tamaño

- La relación de aumento de amplitud con el aumento de tamaño de las figuras no es lineal. Y esto tiene como razón el hecho de que la pantalla donde se generan las figuras no es lo suficientemente grande como para abarcar un rango de variaciones de NPS tan grande como el que maneja el oído humano. Se concluye que el rango de audición es demasiado grande frente al rango de tamaños que el programa puede

manejar, por lo que se desarrolló una relación de duplicación entre el aumento de tamaño de las figuras geométricas y el aumento de NPS.

- El *software* en presencia de una distorsión digital por exceso de amplitud de señal, es capaz de determinar la frecuencia fundamental de dicha señal y por consiguiente su amplitud de manera solvente. Sin embargo cerca de los 120 dB de amplitud de la onda de audio el programa maneja figuras muy grandes como para ser apreciadas en su mayor medida, a niveles de presión más altos se siguen construyendo figuras hasta los 174 dB, donde se aprecia una ampliación enorme de los elementos visuales. Por lo tanto se concluye que el programa es capaz de discernir en condiciones adversas datos específicos de una señal para enviarlos a procesamiento posterior.

7.3.1.2 Frecuencia vs. Color

- Cuando trabaja con el micrófono interno del computador o con una señal generada en un *software*, por debajo de los 120 Hz, el programa tiene problemas para extraer parámetros pues ninguno de las dos fuentes de sonido anteriores garantiza la captación y la pureza de una señal a frecuencias demasiado bajas, por lo que el programa podría en eventuales ocasiones no generar los colores correspondientes a rangos de frecuencias demasiado bajos.
- Al ascender en frecuencia hasta el rango de 2 KHz a 8 KHz, el programa en ciertas ocasiones puede generar imágenes de similar color, por lo tanto se concluye que la aplicación en ocasiones compara tonos puros que pueden tener otras componentes senoidales implícitas e indeseadas, y al tener relación de octavas entre algunos tonos el programa puede generar en muy especiales condiciones, colores muy similares a pesar de una diferencia de frecuencias.

7.3.1.3 Espectro vs. Textura

- Haciendo un análisis del comportamiento del *software* respecto a la relación espectro/textura se llega a la conclusión de que es la más lineal de las relaciones a pesar de la mayor subjetividad con que fue construida, esto se debe a que el programa determina la presencia o ausencia de ciertas componentes senoidales y con ello determina las instrucciones para que el generador de imágenes modifique la superficie de las figuras geométricas.
- En presencia de señales compuestas por una gran cantidad de ondas superpuestas, tales como los sonidos aleatorios de la naturaleza o generados artificialmente como los ruidos blanco y rosa, el programa cambia con gran rapidez la textura sin darse fenómenos como en las anteriores analogías, por lo que se concluye un desempeño sólido de esta etapa del proyecto.

7.3.1.4 Imágenes resultantes de timbres definidos

- Los instrumentos provocan imágenes que demuestran contenido armónico acorde a sus sonoridades. El *software* es capaz de discernir información en *samples* que poseen formas de onda compleja, alejados de los tonos puros usados anteriormente.
- Sin embargo las imágenes no son una constante, incluso al repetir el mismo *sample* el programa puede generar imágenes que guardan diferencias unas respecto a otras, pues algunos de los procesos del generador primigenio de figuras son de naturaleza aleatoria.

7.3.1.5 Imágenes resultantes de estilos musicales definidos

- Cuando se trabaja con señales de extrema complejidad como el audio de una interpretación de un instrumento solista, un grupo musical e incluso una orquesta sinfónica, el programa construye imágenes que son muy cambiantes de acuerdo al momento y la intención de la interpretación, por lo que son un tanto esquivas de obtener en un

instante dado y no representan un comportamiento definitivo del generador de figuras ante un género o estilo musical determinado.

- La opción de superposición de figuras brinda imágenes más elaboradas, por lo tanto es una opción a tomar en cuenta para componer imágenes que vayan más allá de figuras o partículas desplazándose por un espacio confinado.

7.3.2 Conclusiones generales de desempeño del programa

- La aplicación finaliza su desarrollo con la implementación de las múltiples opciones gráficas y herramientas de distinta índole, por lo que se concluye que se superan los objetivos planteados inicialmente de construir un *software* que tan sólo relacione las analogías base de esta tesis, pues la aplicación ha sido dotada de una serie de elementos adicionales que propenden hacia un mejor desempeño y un mejor ambiente de trabajo con el usuario.
- El uso de la interfaz de usuario no presenta complicaciones pues los elementos están dispuestos de acuerdo a las secciones de sonido e imagen. Por lo tanto es una aplicación de uso sencillo y se concluye mediante su desarrollo que es posible construir una aplicación que puede tener elementos complicados de trabajo, pero que al usuario puede brindarle un ambiente amigable de trabajo.
- Algunas opciones requieren pasos previos, como por ejemplo el hecho de escoger un fondo con superposición de figuras requiere primero escoger ese color sin superposición, sin embargo esto no complica la operación de la aplicación, y refuerza la conclusión acerca de un ambiente amigable.
- Sin embargo luego de estudiar las posibilidades del programa se concluye que para trabajar de manera óptima requiere que la entrada de audio sea de calidad garantizada en cuanto a respuesta de frecuencia, rango dinámico, ausencia de ruido, entre otros; esto se puede lograr implementando excelente microfónica o dispositivos de audio de buena

calidad y características en los puntos indicados, que alimenten con su señal al *software*.

7.4 Conclusiones económicas y de relación costo/beneficio

- El desarrollo de este trabajo de titulación es principalmente una tarea sin movilizaciones a lugares distantes, adquisición o alquiler de equipos o herramientas complejas o de difícil disponibilidad.
- Su presupuesto es reducido aún si se considera el empezar de cero con todos los rubros, pues como se dijo anteriormente muchos de ellos podrían estar ya cubiertos.
- Resulta un tanto difícil acceder a capacitación de este tipo de herramientas principalmente por escasez de gente conocedora del tema y también por falta de público interesado en este tipo de entornos de programación, muchas personas aprenden por experimentación propia y con el uso de una conexión a Internet.
- La herramienta creada para este trabajo de titulación tiene considerables ventajas sobre programas relacionados, en muchos campos empezando desde su capacidad para trabajar con audio externo en tiempo real, pasando por su opción de analizador de audio, hasta la facilidad para generar imágenes en 3 dimensiones y su libre manipulación espacial.

7.5 Conclusiones de proyecciones y posibilidades de expansión

- El programa posee proyecciones en el ámbito de la producción de eventos donde se quiera combinar audio e imagen, sin embargo este es un programa piloto que necesita revisión de sus prestaciones para desempeñarse en un uso intensivo a nivel comercial.
- Es posible también por su importante vinculación entre dos órganos sensoriales del ser humano, ampliar sus proyecciones de uso y mejoramiento a campos muy diversos como la medicina y la investigación, entre muchos otros.
- Por medio de la experiencia generada al momento de crear este programa para el trabajo de titulación, se concluye la enorme posibilidad

de expansión que puede tener el programa, sólo dependiente de las necesidades del usuario y de las capacidades cada vez mayores de los computadores como plataformas de procesamiento digital de señales.

7.6 Recomendaciones

Las recomendaciones serán divididas en cierta manera como las conclusiones, desde recomendaciones generales, hasta las más específicas de aspectos técnicos y operación del *software*.

7.6.1 Recomendaciones generales

- Este *patch* debe ser usado ingresando a la aplicación Max/MSP/Jitter de la compañía Cycling '74, que debe ser instalado aparte de sus objetos por defecto, con los *plug-in* u objetos externos que a continuación se nombran: WinPitch~ (para el caso de usuarios Windows) y pitch~ (para usuarios Macintosh).
- El programa producto de esta tesis no es una aplicación ejecutable y por lo tanto necesita del entorno Max/MSP/Jitter, ya que es un archivo de tipo *patch*, con extensión .pat y que para poder funcionar de manera correcta debe contar con los objetos externos mencionados anteriormente.
- Para obtener resultados de mejor visualización es preferible contar con una pantalla de buen tamaño que garantice la visión simultánea de la interfaz de usuario y del espacio donde se generan las figuras. Para de esta manera poder modificar las imágenes y observar el trabajo del analizador de audio en tiempo real. Esto sin embargo, no quiere decir que no se pueda usar en computadores de pantalla pequeña, se lo puede usar sin ningún inconveniente aparte de la visualización óptima de todos los elementos que componen el programa.
- Los requerimientos técnicos del programa no son definidos completamente ya que puede trabajar en cualquier computador que posea estos elementos instalados (Max/MSP/Jitter y sus objetos externos), la única diferencia es la rapidez con que se realiza el análisis

del audio y la generación de la imagen. Cualquier computador que soporte un sistema operativo compatible con las diferentes versiones de Max/MSP que incluyan Jitter y sus objetos es potencial candidato a usar el *software*, con tiempos de respuesta en función de su rapidez.

- Sin embargo luego de probar el programa en diferentes computadores se llegó a la conclusión de una serie de requerimientos técnicos de velocidad y memoria para que el programa trabaje con tiempos de respuesta mínimos, asegurando un verdadero desempeño en tiempo real. Esto no quiere decir que el programa no trabaje en computadores por debajo de estos parámetros técnicos. Se recomienda para computadores de cualquier plataforma un procesador de 2 GHz de velocidad y 2 GB de memoria RAM, tarjeta de video de 128 MB, tarjeta de sonido compatible con la norma ASIO.

7.6.2 Recomendaciones técnicas y de operación del programa

Aquí se recogen las sugerencias y recomendaciones de operación que eventualmente pudieron surgir a lo largo de los capítulos del desarrollo o las pruebas del programa.

- El *software* maneja dos tipos de entrada de audio, la carga de archivos y el ingreso directamente por un micrófono o la entrada de audio del computador, al activar cualquiera de los dos íconos de entrada de audio automáticamente se activan los dos, por lo que se recomienda la apertura del conjunto de *faders* de la entrada con la que se desee trabajar.
- Los archivos de audio que puedan ser cargados deben estar en formatos específicos (.aiff, .aif, .snd, .wav) para que el programa pueda trabajar con ellos.
- Al situar el *fader* al máximo hay que cuidar mucho la distorsión digital que esto pueda causar, es mejor trabajar con un nivel inferior cuidando la distorsión pero intentando guardar un buen nivel de señal para que las figuras tengan el tamaño suficiente para ser apreciadas de acuerdo a la amplitud de la onda; de igual manera cuando la entrada de audio es

configurada al través del micrófono de la computadora o su entrada de audio digital. Opciones futuras como por ejemplo el uso de ecualizadores dedicados para modificar la respuesta de color requerirían el mismo cuidado.

- Al encender el generador de imágenes se aprecia su configuración por defecto que consta en esferas y un tipo de unión *tri_strip*. Posteriormente el usuario puede modificar la imagen con todas las opciones que fueron descritas en el proceso de desarrollo y pruebas.
- Cuando se quiere escoger un color de fondo diferente, en la modalidad sin superposición de figuras el *software* igualmente empieza con un tipo de unión *tri_strip*. Pero al momento de escoger la modalidad con superposición de imágenes es importante que el usuario primero escoja el color de fondo sin superposición para luego pasar a superposición.
- Cuando se encuentra en modalidad de superposición resulta un poco menos práctico aplicar la herramienta de rotación libre con el *mouse*, debido a que al momento de rotar se dibuja un sistema de ejes tridimensionales a manera de guía y los mismos también sufren superposición entre ellos mientras rotan.
- Lo contrario sucede cuando se trabaja sólo con colores de fondo y figuras de rápida extinción, se puede rotar libremente sin que el sistema de ejes guías sufra deformaciones en el proceso de rotación.
- La pantalla donde se generan las imágenes se la puede modificar libremente en términos de tamaño y de forma (cuadrada a rectangular, y viceversa); por lo que se recomienda asignarle el mayor tamaño en la medida de lo posible, si el usuario quiere obtener una vista de pantalla completa de las imágenes debe presionar la tecla ESC.
- Al entrar a este modo es posible que ciertas preferencias previamente escogidas por el usuario, principalmente un color de fondo con superposición se convierta en un fondo negro con superposición, por lo que el usuario debe estar atento a un cambio de esta naturaleza el momento de ampliar su imagen a modo de pantalla completa.