

UNTREF

UNIVERSIDAD NACIONAL
DE TRES DE FEBRERO

Regiones vacías en el mapa de AudioStellar

Estrategias de interpolación, síntesis e interfaz de usuario

Máximo Signiorini

Trabajo Final de Grado

Licenciatura en Artes Electrónicas

Índice

• Abstract	02
• Agradecimientos	03
• Introducción	04
○ Sampler	04
○ AudioStellar	05
○ Motivación y Pregunta	07
○ Antecedentes y Estado del Arte	07
• Materiales y Métodos	13
○ Dataset	13
○ Métodos	15
○ Morph Unit	15
○ Experimentación	19
○ Machine Learning	24
• Conclusiones	27
○ Futuro	28
• Referencias	29
• Anexo	30

Abstract

En el presente trabajo final de grado mi objetivo es realizar un desarrollo técnico para el software AudioStellar del proyecto de investigación “Análisis y representación sensible de datos” dirigido por Leandro Garber y radicado en MUNTREF Centro de Arte y Ciencia. Este desarrollo técnico propone investigar áreas o ideas inexploradas hasta el momento en el software, expandiendo sus posibilidades sonoras y performáticas.

Mi trabajo explora las regiones sin sonidos. Regiones que hasta el momento han sido inexploradas tanto por lxs usuarixs como por lxs mismxs desarrolladores. El desarrollo consta de una nueva *Unit* en AudioStellar, la cual incorpora técnicas como la del crossfade de volumen para sintetizar sonidos en regiones vacías del mapa.

Este trabajo también está enmarcado en el proyecto de investigación acreditado "Estéticas Latentes" dirigido por Laurence Bender y que integro junto a Leandro Garber, Leonardo Pepino e Ivan Pupkin.

Agradecimientos

En primer lugar quisiera agradecer a mi tutor, Leandro Garber, quien me acompañó y guio en todo el proceso. Fue un gran pilar a toda hora para ayudarme a no perder el foco del trabajo y siempre traer aportes o ideas nuevas.

También agradecer a Damian Anache, con quien mantuvimos un intercambio a lo largo del trabajo y brindó valiosas opiniones y devoluciones acerca del mismo.

Agradecer a Ernesto Romeo quien brindó su Estudio/Laboratorio “La Siesta del Fauno” para la grabación de sintetizadores analógicos y con quien mantuvimos un intercambio muy valioso en torno a técnicas de *morphing* y su implementación en sintetizadores.

Gracias a Gaston Duchovny, Ivana Budmann y Camila Godoy quienes brindaron grabaciones de sus voces para la construcción del dataset.

Gracias a Hernán Kerlleñevich, Mene Savasta, Nicolas Varchausky y Ezequiel Abregu por haber brindado valiosas referencias de artistas y bibliografía.

Gracias a mi familia por apoyarme constantemente en mi carrera y trabajo, sin su confianza, amor y apoyo esto no hubiera sido posible.

Gracias a la Universidad Nacional de Tres de Febrero en donde me formé y me llevo valiosos aprendizajes y recuerdos con docentes y amigxs.

Introducción

Sampler

Un sampler es un instrumento electrónico capaz de grabar, cargar o reproducir audios. Se considera al “*Mellotron*” como el primer sampler, siendo este un instrumento de tipo electromecánico fabricado por primera vez en 1963. El mismo era capaz de reproducir audios grabados en cinta a través de una interfaz de tipo teclado musical. Cada tecla disparaba la reproducción de un cabezal de cinta, posibilitando así la reproducción de audios por cada tecla. Este instrumento rápidamente se popularizó en bandas musicales de la época como *The Beatles*, *The Rolling Stones*, *King Crimson*, etc.



Figura 1: El Mellotron M400 de 1970 por Alison Stout, 2017, Reverb¹

Con el avance de los años y el surgimiento de la era digital en los 80, vieron la luz los primeros samplers digitales. Los primeros como el *EMS* o el *Fairlight CMI* contaban con una lógica idéntica a la del mellotron. Su interfaz constaba de un teclado musical capaz de disparar sonidos. La diferencia entre estos últimos y el mellotron radica en que su fuente de sonido no proviene de cintas analógicas, sino de una computadora digital. Esto facilitó el uso del sampler, pudiendo cargar mediante diskettes cualquier sonido grabado rápidamente. Ya para 1988 sale al mercado uno de los samplers más importantes en el mundo de la música, el Akai MPC60. Su novedosa interfaz estaba orientada hacia lo que hoy se conoce como “*finger drumming*”, una forma de crear ritmos con los dedos. Su interfaz no se restringe a tecladistas, sino que abre una nueva posibilidad de performatividad en el mundo de los samplers. A su vez, contaba con un poderoso secuenciador que permite construir detalladamente secuencias de sonidos y manipulación de sus

¹ <https://reverb.com/news/how-to-repair-a-mellotron-m400-a-look-inside-the-early-sampler-keyboard>

parámetros. El impacto de la MPC60 fue tal, que inspiró y propulsó géneros como el Hip Hop.



Figura 2: El Akai MPC60 de 1988, por Equipboard ²

Pero hasta acá, todos estos desarrollos comparten una limitación, y esta es la cantidad de sonidos que pueden alojar. Estos instrumentos están diseñados para poder reproducir como máximo entre 10 y 20 sonidos a la vez. Si bien el surgimiento de las computadoras hogareñas y los softwares de producción y edición de audio rompieron este límite, el manejo de una gran cantidad de sonidos es un problema que persiste hasta el día de hoy en el mundo del audio. Un problema común en el área de la producción musical es la curaduría o elección de sonidos. Sucede que, muchas veces, productores tienen miles de sonidos disponibles para utilizar en sus producciones y la única interfaz para escucharlos o elegir entre ellos es el navegador de archivos de la computadora. Rápidamente la elección de sonidos se vuelve tediosa, teniendo que navegar entre miles y miles de sonidos sin orden evidente. De esta problemática y del mundo de los samplers es que nace AudioStellar.

AudioStellar

AudioStellar es un *sampler* experimental, se puede comprender como “*un instrumento musical de código abierto basado en datos para el descubrimiento de estructuras sonoras latentes y la experimentación musical*” (Garber et al., 2020). Es un proyecto *open source* o de código abierto, esto quiere decir que su código está libre y disponible a la comunidad para que esta pueda contribuir, colaborar o simplemente aprender del mismo.

² <https://equipboard.com/items/akai-mpc60>

AudioStellar posee las características básicas que reúnen a un sampler, y las lleva a terrenos inexplorados en el área de los softwares de producción sonora. Según Leandro Garber, ya no está presente la idea del sample aislado como lo estaba en el Akai, por ejemplo. En AudioStellar existe la idea del *sample* como parte de un grupo, no como un sample por sí solo.

Esto se logra a partir de la capacidad de descubrir estructuras sonoras latentes, relaciones entre los sonidos. Es capaz de organizar o agrupar en un mapa bidimensional sonidos según distintos atributos (timbre, envolvente de amplitud o frecuencias). De esta manera, la exploración de sonidos y la relación entre los mismos se evidencia en la interfaz que propone AudioStellar.

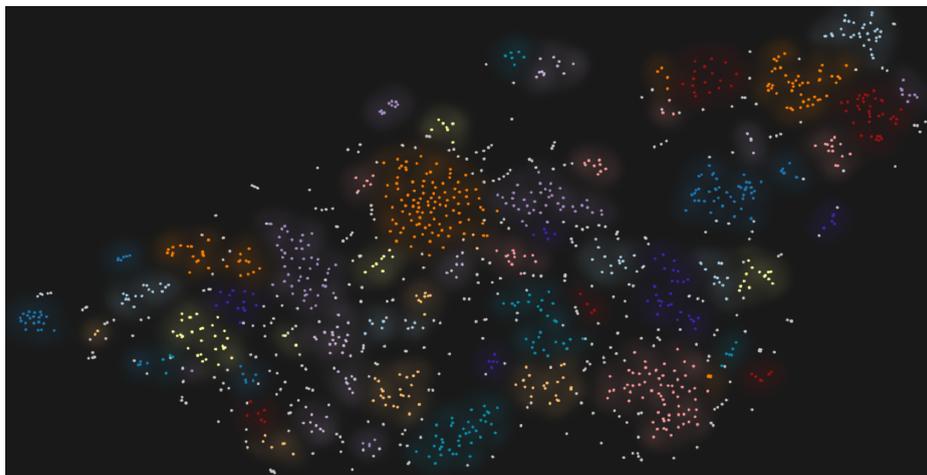


Figura 3: Mapa bidimensional de AudioStellar

Cada punto en el mapa representa un sonido el cual es parte de un grupo o *cluster* de sonidos. Estas agrupaciones de sonidos están dadas según distintas características sonoras. Este espacio o agrupación es construido a través de técnicas de Machine Learning y se lo conoce como *espacio latente*. Este espacio latente “Es una representación de los clips de audio de entrada que están codificados para lograr un espacio significativo semánticamente” (Garber et al., 2020).

De manera premonitoria, en 1996 Trevor Wishart imaginaba una forma de trabajo casi idéntica a la de AudioStellar:

¿Tiene el espacio tímbrico una topología? Al trabajar con instrumentos musicales existentes, podemos construir un mapa de las posibilidades tímbricas del instrumento. Para hacer esto, en lugar de simplemente enumerar todos los posibles tipos de sonido que podría producir un instrumento como un violín, intentaríamos colocarlos en un mapa (que podría ser multidimensional) en el que se colocarían objetos de sonido similares.

cerca unos de otros y los objetos sonoros que son bastante diferentes entre sí se colocarían a una mayor distancia”.

Parece una premonición de lo que finalmente hoy en día es el software.

AudioStellar es capaz de generar cierta expresividad o discurso sonoro a través de las distintas herramientas e interfaces que nos provee. Estas herramientas las llamamos “Units” o “Unidades”. Cada Unit es capaz de explorar este espacio latente de diversas formas y al día de hoy cuenta con cuatro: “Explorer Unit”, “Particle Unit”, “Sequence Unit” y “OSC Unit”.

Motivación

Mi motivación para el trabajo parte de explorar nuevas funcionalidades que posibiliten nuevos sonidos y expresividades en el software. Particularmente aquellas que impulsen aún más esta idea de explorar sonoridades latentes.

La posibilidad de contribuir a un proyecto tan innovador y que empuja los límites de lo que se considera un instrumento es una gran inspiración. Sobre todo, teniendo en cuenta que dicha contribución será implementada en el software y utilizada por muchas personas para la creación o experimentación sonora.

Estos intereses desembocan en una serie de preguntas que son las que guían este trabajo. ¿Cómo podemos vincular los sonidos en AudioStellar? ¿Cómo suenan las regiones sin sonidos en el mapa? ¿Cómo puedo hacerlas sonar? ¿Con qué herramientas?

Para responder estas incógnitas es necesario comprender los antecedentes y el estado del arte que involucran las distintas técnicas, herramientas y aplicaciones que vinculan sonidos.

Antecedentes y Estado del Arte

Existen muchas maneras de relacionar, combinar y mezclar audios. Muchas provienen de técnicas o herramientas tan elementales como el control de volumen de una fuente sonora.

La primera técnica es quizás la más conocida y utilizada en el mundo del audio. Me refiero a el de *crossfade* de volumen. Esta técnica consiste en generar un cambio gradual o brusco de volumen en el tiempo entre dos o más sonidos. Mientras un sonido comienza a decrecer en amplitud, el otro comienza a crecer. Esto genera una transición o mezcla gradual entre ambos sonidos. Este cambio puede variar desde uno muy veloz hasta uno más lento, dependiendo de la curva que se utilice.

Si bien se comienza a popularizar con la creación de las primeras consolas analógicas, esta técnica es totalmente realizable a través de cualquier instrumento o fuente sonora capaz de regular su amplitud. La novedad de estas consolas o “mixers” es que permitieron un control de volumen (entre otros parámetros) a través de perillas o “faders”. De esta manera permitiendo disminuir o aumentar el volumen de múltiples sonidos manualmente y en simultáneo. Con el advenimiento de las computadoras y los softwares de edición de audio, esta técnica se vio tremendamente potenciada. Dichos entornos digitales permiten controlar y automatizar el volumen de centenares o miles de fuentes sonoras simultáneamente y variar o personalizar cada cambio de volumen.

Aunque muy simple, esta técnica puede generar ricos resultados tímbricos si es utilizada correctamente. Un ejemplo muy claro de la potencialidad del crossfade de volumen lo escuchamos en la obra de Trevor Wishart, “Red Bird”. En ella Wishart hace un uso extensivo del crossfade de volumen para generar un “*morphing*” o una combinación dinámica entre sonidos tímbricamente muy compleja. El verbo *morphing* refiere a “la ilusión de que un sonido se ha convertido de forma continua y perfectamente en otro” (Miller, 2015).

Para realizar esta operación, Wishart hizo una selección de sonidos tímbricamente similares entre sí (voces humanas, animales y grabaciones de campo). Esto le permitió combinarlos y mezclarlos sutilmente para que su contenido espectral sea similar, de modo que un fundido cruzado entre los dos crea la ilusión de transformación.

El concepto de *morphing* no solo es algo central en dicha obra de Wishart, sino en casi la totalidad de su trabajo como artista sonoro e investigador. Su escrito “*On Sonic Art*” publicado en 1996 ha influenciado mayormente mi trabajo final de grado. En él, Wishart pone en discusión la forma de analizar y componer música que se venía utilizando hasta el momento. Una concepción de la música y el sonido basada en un sistema que Wishart define como “*entramado*”.

(...) no necesitamos tratar con un conjunto finito de posibilidades. La idea de que la música tiene que construirse sobre un entramado finito y la idea relacionada de que los procedimientos permutacionales son una forma válida de proceder serán criticadas aquí y se desarrollará una metodología musical para tratar con un continuo utilizando el concepto de transformación.
(Wishart, 1996)

Y lleva más allá esta idea, proponiendo este concepto de transformación de sonidos como un problema práctico que debe abordar el arte sonoro.

Paradójicamente su obra Red Bird fue publicada muchísimo antes de que surgieran las tecnologías digitales capaces de hacer un proceso de *morphing* del cual Wishart se explaya en su texto. Lo cual hace todavía más interesante su trabajo, ya que logra concretar estos efectos de *morphing* haciendo uso de grabadoras de cinta

analógicas y utilizando técnicas de cortado, pegado y fundido manuales entre ellas. Pero una de las técnicas más poderosas que despliega Wishart en *Red Bird* es la de la contextualización aural. Para poder realizar una transformación o un *morphing* exitoso entre dos sonidos, muchas veces no alcanza solo con transformar alguna de sus características sonoras en el tiempo. Se necesitan de sonidos auxiliares que nos permitan correctamente identificar las fuentes sonoras, para que este cambio o transformación entre un sonido al otro sea percibido claramente. Wishart nos da un claro ejemplo de dicho proceso en *Red Bird* en su libro *On Sonic Art*:

En el primer ejemplo escuchamos el sonido de un libro siendo arrojado sobre una superficie (...) cambiando gradualmente hacia el sonido de un portazo. Esta transformación depende fuertemente del uso de las señales o pistas de contexto. Por lo tanto, como se mencionó previamente, el sonido del golpe del libro, si es escuchado aisladamente, no es fácilmente asignable a la fuente "libro", sin importar qué tan realista sea la grabación. La interpretación de la imagen aural "libro" es por lo tanto impuesta sobre el sonido cuando primero escuchamos la pista contextualizante "sonidos de páginas de un libro al pasar". Los sonidos de portazos, sin embargo, pueden o no ser fácilmente asignables a la fuente "puerta" como se mencionó más arriba. La introducción de ruidos de la manija de la puerta hacen esta tarea más sencilla. De esta manera podemos efectuar una transición desde un portazo sin señales previas a un portazo con ruidos de manija, introduciendo gradualmente el sonido de la manija y por lo tanto haciendo la identificación de "puerta" más definida. En la imagen acústica que interviene entre estos dos extremos, la asignación del sonido 'slam' se vuelve difícil (...) La ambigüedad de la fuente sólo puede resolverse mediante pistas contextuales que nos permiten pasar de la interpretación de "libro" a la interpretación de "puerta". (Wishart, 1996).

Otra posible aproximación para vincular sonidos es a través del *morph* espectral o en palabras de Wishart "*interpolación espectral*" (1994). A diferencia del crossfade de volumen, el *morph* espectral se encarga de variar y hacer un fundido del espectro de dos o más sonidos.

Esta técnica nace con la creación del "*vocoder*" y continúa su desarrollo con el advenimiento del audio digital en la década del 90. El *vocoder* o "*voice coder*" fue creado alrededor de la década del 50. Originalmente surgió en el campo de las telecomunicaciones para comprimir la voz humana y así transferirla de manera más eficiente. Mark Jenkins describe el funcionamiento del *vocoder* como un dispositivo que divide el sonido en distintas bandas de frecuencia y utiliza la salida de cada banda para alimentar un amplificador aplicado a solo una banda de frecuencia de una señal de entrada alternativa. En otras palabras, el *vocoder* opera directamente sobre el espectro de un sonido para aplicar información de este a otro. Si bien los primeros *vocoders* no estaban pensados para un uso artístico, en 1968 Robert Moog crea el primer *vocoder* a base de transistores pensado para un uso musical.



Figura 4: Vocoder fabricado por Robert Moog en 1970. Fotografía por Stephen Ruk, Pinterest ³

Ya para fines de los 80 y comienzos de los 90, con las primeras computadoras hogareñas y los softwares de edición y creación sonora, llegaron los primeros desarrollos capaces de implementar técnicas como la interpolación espectral. Esto se vio posibilitado por la aplicación de las transformadas de fourier o *fft* en el plano digital.

Tanto en el *vocoder* como en técnicas de *morphing* espectral, el resultado es un tanto más complejo que en técnicas de *crossfade* de volumen. Esto se debe a que el timbre, a diferencia de la amplitud, es un aspecto del sonido multidimensional, así lo considera Robert Erickson “*Claramente, el timbre es un estímulo multidimensional: no puede correlacionarse con ninguna dimensión física única*” (1975).

Diversos son los artistas explorando con técnicas en torno a la interpolación espectral en su obra, pero uno a destacar particularmente es Amon Tobin con su disco “*ISAM*”. Casi la totalidad de los sonidos presentes en el mismo fueron contruidos o sintetizados a partir de técnicas relacionadas al *morph* espectral. Así, sintetizando grabaciones de campo de entornos naturales para transformarlos en sonidos irreconocibles de su fuente. Tobin es capaz de implementar dichas técnicas de *morphing* a través de la *workstation* digital “*Kyma*”. Este dispositivo es esencialmente una computadora en formato *rack*, especializada para el diseño sonoro y síntesis (Figura 5). A través de este dispositivo Tobin pudo aplicar técnicas de síntesis concatenativa y de convolución, que le permitieron manipular grabaciones de campo o instrumentos para transformarlos y combinarlos en sonidos totalmente nuevos y distintos a los originales.

³ <https://ar.pinterest.com/pin/534239574524316428/>



Figura 5: El hardware de Kyma. Future Music, 2009, Music Radar ⁴.



Figura 6: El software de Kyma, por Equipboard ⁵

Hoy en día tenemos nuevas técnicas que están comenzando a surgir en el vasto mundo del Machine Learning y la Inteligencia Artificial. Si bien los principales desarrollos en este campo tienden a estar orientados a el campo visual, poco a poco comienzan a surgir técnicas experimentales en el campo del sonido. Las que son de especial pertinencia para este trabajo son aquellas que obtienen resultados similares a aquellos de la técnica de interpolación espectral. Esto es, ambas técnicas operan sobre el espectro de los sonidos para realizar fundidos o simplemente síntesis para generar nuevos sonidos. Pero ambas difieren en el método o la manera a la que llegan a dichos resultados. Holly Herndon describe esta técnica específicamente en el campo del Machine Learning como “*Spawning*” “(...) a diferencia del *sampling*, que es una reproducción de los sonidos muestreados de una grabación, *Spawning* es la capacidad de crear obras a semejanza de otras interactuando con un modelo entrenado en ellas. *Sampling* del siglo XXI, con grandes implicaciones”.

Los primeros desarrollos en el campo de Machine Learning y sonido comienzan a darse a comienzos de los 2000 aproximadamente. Estas primeras investigaciones

⁴ <https://www.musicradar.com/reviews/tech/symbolic-sound-kyma-pacarana-213089>

⁵ <https://equipboard.com/items/symbolic-sound-kyma>

estaban orientadas a la clasificación, agrupamiento (*clustering*) y reducción de dimensionalidad de sonidos o música. AudioStellar nace a partir de estos primeros desarrollos en torno a el agrupamiento y reducción de dimensionalidad de sonidos y redirecciona estas investigaciones del ámbito científico hacia un ámbito artístico. Es decir, se toma esta idea del análisis exploratorio para desarrollar un instrumento, una herramienta para artistsxs.

Ya para el 2016 se crea el equipo de investigación Magenta, un proyecto que nace desde el equipo de Google Brain. En su presentación del proyecto, el director del equipo Douglas Eck describió el proyecto como:

(...) un proyecto de investigación para avanzar el estado del arte en inteligencia de máquinas para música y generación de arte. El machine learning ya fue usado extensivamente para entender contenido, como en el reconocimiento del habla o la traducción. Con Magenta, queremos explorar la otra parte, desarrollar algoritmos que puedan aprender como generar arte y música, potencialmente creando contenido convincente y artístico por su cuenta. (2016)

Poco después de un año de su creación, el equipo de Magenta presenta el “*NSynth*”, un sintetizador neural. El equipo de investigación lo define claramente:

A diferencia de un sintetizador tradicional que genera audio desde componentes hechos a mano como osciladores o tablas de ondas, *NSynth* utiliza redes neuronales profundas para generar sonidos a un nivel de samples individuales. Aprendiendo directamente de datos, *NSynth* provee a los artistas con un control intuitivo sobre el timbre y las dinámicas y la habilidad de explorar nuevos sonidos que hubieran sido difíciles o imposibles de producir con un sintetizador afinado a mano. (Magenta, 2017)

Una de las posibilidades del *NSynth* es poder interpolar entre los timbres de distintos sonidos generados por estas redes neuronales. Lo que destaca a *NSynth* de otros desarrollos similares en el área de Machine Learning es que este desarrollo no se restringe únicamente al mundo del software, sino que construyeron una interfaz para el mismo. La interfaz tiene semejanzas a la del *KaossPad* de *Korg*, una placa cuadrada táctil destinada al control x-y de distintos parámetros del sintetizador. De esta manera el *NSynth* se ofrece más como un instrumento musical abierto a la expresividad y al intercambio humano-máquina.

En síntesis, el estado del arte de todos estos desarrollos buscan combinar o mezclar sonidos para la generación de otros nuevos.

Materiales y Métodos

Materiales (Dataset)

Mi trabajo no comenzó con el desarrollo de una nueva funcionalidad para AudioStellar, primero fue necesario construir un *dataset* con el cual experimentar y testear el desarrollo de esta nueva herramienta. El significado de dataset viene del mundo de la Inteligencia Artificial y refiere a la colección de datos con la que se entrenan los distintos algoritmos de *Machine Learning*. En el caso de AudioStellar, el dataset se comprende de una colección de sonidos que se importan en el programa para que el algoritmo los analice y construya el mapa bidimensional a partir de los mismos. Para la construcción del dataset se optó por uno que comprenda un amplio espectro de sonidos. Entre las categorías de sonidos que se encuentran encontramos instrumentos de orquesta, instrumentos electrónicos, digitales, ruidos, grabaciones de campo, voces humanas, entre otros. Esta gran variedad de sonidos nos permitió más adelante experimentar ampliamente los resultados sonoros que provee la herramienta desarrollada.

Cabe destacar que para todos aquellos sonidos tonales dentro de la escala temperada se los restringió a la escala de Re menor pentatónica. Esto ayudó a que la construcción del dataset sea más ágil.

Para la construcción del dataset fue necesario recolectar y/o construir todos los sonidos seleccionados. Muchos de ellos provienen de librerías o bancos de sonidos como [Free Sound](#) u otras librerías similares. Otros sonidos, como las guitarras, fueron grabadas en mi estudio. Mientras que los sintetizadores analógicos fueron grabados en el Estudio/Laboratorio La Siesta del Fauno fundado por Ernesto Romeo y Pablo Gil. Allí tuve la oportunidad de grabar sintetizadores analógicos de la década de los 70, 80 y 90 que, sin la predisposición y amabilidad de Ernesto, no podría haberlo hecho. A su vez, tuvimos otro encuentro muy fructífero con Ernesto en su estudio en donde conversamos sobre distintas técnicas de *morphing/crossfade*, sobre todo en el mundo de los sintetizadores e instrumentos electrónicos. Esa charla sirvió de mucha inspiración y referencia para mi desarrollo.

Una vez reunidos todos los sonidos del dataset, siguió la edición y normalización de los sonidos. Esto fue necesario, primordialmente para nivelar todos los sonidos a un mismo volumen y poder apreciarlos por igual, pero sobre todo para que las técnicas de crossfade se aprovechen lo mejor posible. Para ello todos los sonidos fueron importados a *Reaper*, software de edición de audio en el cual se llevó a cabo dicho proceso. Una vez terminado el proceso de edición y normalización de sonidos, el dataset acumuló un total de 936 sonidos.

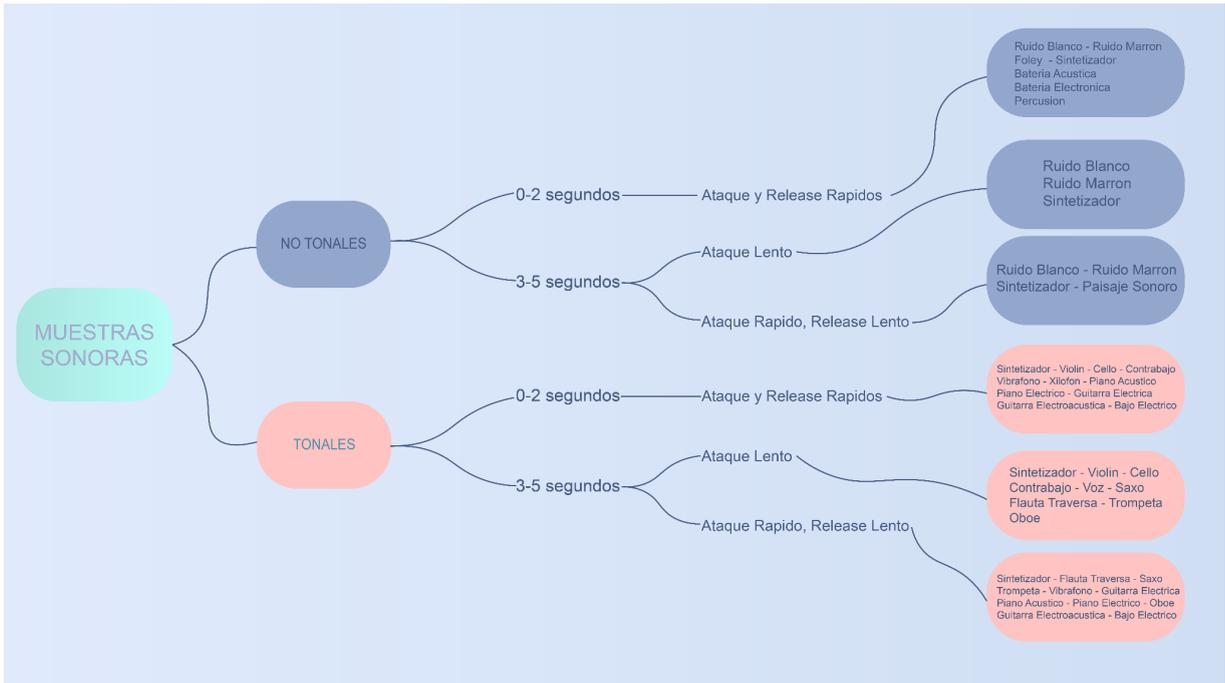


Figura 7: Esquema del dataset construido para el trabajo



Figura 8: Categorías finales de los distintos sonidos que componen al dataset

Métodos

Existen múltiples técnicas que proponen posibles respuestas a la incógnita de cómo relacionar, combinar o mezclar sonidos. Las técnicas de crossfade de volumen, interpolación espectral, síntesis concatenativa, convolución y algunas nuevas vinculadas al machine learning son algunas de ellas. Estas permitirían explorar distintas conexiones entre los sonidos de AudioStellar, pudiendo sintetizar un sonido en el vacío que separa a dichos sonidos. Cada una de estas presenta distintos desafíos a la hora de ser implementada en un software. Pero, a su vez, nos devuelven distintos resultados sonoros.

De las distintas técnicas y herramientas mencionadas más arriba se optó por implementar en el software la de crossfade de volumen y explorar de manera experimental las nuevas herramientas y técnicas de [Machine Learning aplicadas al sonido](#). Las demás técnicas fueron descartadas, por la disponibilidad de tiempo para realizar el trabajo. Implementar técnicas como las de interpolación espectral o Machine Learning implican un tiempo de trabajo que excede al disponible.

Morph Unit

El nombre de mi desarrollo, de esta nueva Unit, nace como inspiración de las técnicas de Morphing de audio, pero como se verá más adelante, su funcionalidad no se restringe solo a ello.

Implementar un sistema de fundido cruzado de volumen entre audios resulta sencillo y práctico comparado con las otras técnicas. No solo para las posibilidades temporales de este trabajo, sino también para las posibilidades técnicas que presenta el entorno de programación en donde AudioStellar es desarrollado. Gran parte del mismo está desarrollado en *openFrameworks*, un entorno de programación open-source basado en *c++* y destinado a artistxs y programadorxs creativxs. Cabe destacar que fue inicialmente pensado para personas abocadas al campo de las artes visuales. Esto quiere decir que sus posibilidades sonoras no son su fuerte. Sin embargo, el hecho de que sea un proyecto código abierto permitió que, a través de la contribución de la comunidad, se agregaran algunas posibilidades sonoras al entorno.

Antes de comenzar con el desarrollo y la aplicación de técnicas de crossfade de volumen, desarrollé una aplicación por fuera de AudioStellar para poder testear dichas técnicas dentro del mismo entorno de programación. Su funcionalidad y diseño fueron mínimos con el objetivo de poder darme una idea de cuáles eran los límites y alcances de *openFrameworks* a la hora de implementar dichas técnicas. El

diseño, sobre todo, fue encarado con el objetivo de simular una versión reducida de AudioStellar. La aplicación es capaz de reproducir dos sonidos simultáneamente, y en *loop*, para así poder realizar un fundido cruzado de volumen entre ambos. Ambos sonidos son representados con un círculo de color a cada lado de la pantalla (izquierda y derecha respectivamente). El fundido cruzado se da al mover el mouse de un lado hacia otro y esto, a su vez, se ve reflejado en la opacidad del color de cada círculo. Es decir, no solo se da el efecto de *crossfade* de volumen sino que también a medida que un sonido crece en volumen, también lo hace su opacidad, y viceversa.



Figura 9: Aplicación desarrollada previo al trabajo en AudioStellar para experimentar técnicas de *crossfade*.

Una funcionalidad clave en esta aplicación es la posibilidad de elegir la curva de este fundido, desde lineal hasta exponencial o logarítmica. Esto permite experimentar las distintas variaciones de esta técnica, desde fundidos más bruscos hasta otros más graduales y suaves.

Habiendo comprobado que era posible implementar técnicas de *crossfade* de volumen en openFrameworks, ya fue posible comenzar el desarrollo directamente en AudioStellar.

Modos de Interacción

La nueva Unit desarrollada, llamada "Morph Unit", propone entonces explorar sonoramente los espacios vacíos del mapa a través de la vinculación de los sonidos. Para hacer uso de esta funcionalidad, los usuarios son capaces de determinar una región circular en el mapa la cual determina que sonidos vincular

entre sí, dependiendo si están fuera o dentro de la misma. La ubicación de esta región en el mapa se determina de dos formas posibles. Una de ellas es a través del *clic* del mouse. La otra, en cambio, a través del arrastre del mouse mientras está siendo cliqueado.

Clic

Al hacer *clic* en cualquier región del mapa, se posiciona la circunferencia. Si dentro de ella se encuentra uno o más sonidos, serán reproducidos. Pero reproducir múltiples sonidos dentro de un radio no produce resultados sonoros que exploren ese vacío en el mapa. Son simplemente activaciones de sonidos individuales a lo largo del mapa sin vinculación o relación alguna. Lo mismo puede lograrse con cualquier software de edición de audio. Por eso aparece la noción de distancia, cada sonido con respecto al centro del radio tiene su respectiva distancia. Esta misma se traduce en un valor de volumen para cada sonido. Por ejemplo, mientras más lejos esté el sonido con respecto al centro de la circunferencia, menor será su volumen y viceversa. Así es que se da una mezcla entre sonidos cuyos volúmenes dependen de la distancia entre la zona cliqueada y la ubicación de los sonidos, produciendo así una masa sonora estática. Estática porque no hay cambio ni variación de estos valores de volumen.

Arrastre

Hasta aquí no hay técnicamente una aplicación de las técnicas de crossfade que he mencionado previamente. Recordemos que para realizar una técnica de este estilo se necesitan por lo menos dos fuentes sonoras y la variación en el tiempo de sus volúmenes. Para ello, en vez de utilizar clics del mouse separados en tiempo y espacio, se añade la posibilidad en esta Unit de arrastrar el mouse al ser cliqueado. Ahora hay una conexión, un camino o barrido que une las distintas partes del mapa. De esta manera, la región circular que determina qué sonidos se combinan va deslizándose y genera un barrido continuo de sonidos que entran y salen. A medida que el cursor se desliza por el mapa, los sonidos dentro de la región comienzan a ser reproducidos. A su vez, como sus volúmenes dependen de la distancia con respecto al centro de la circunferencia, estos varían constantemente.

El funcionamiento es el mismo, sigue existiendo la misma noción de distancia y su correlación con los volúmenes de los sonidos. Lo que cambia acá es la gestualidad. Involucra un desarrollo en el tiempo y un traslado en el espacio. Un barrido continuo del mouse a lo largo de una región del mapa no es lo mismo que un clic discreto en una región del mapa. Son dos posibilidades con resultados sonoros distintos, una propone una combinación de sonidos estática, mientras que la otra una dinámica, más relacionada con las ideas del *morphing*.

Interfaz Gráfica

Por otro lado, esta Unit cuenta con una interfaz gráfica capaz de manipular distintos parámetros de la misma. Entre ellos tenemos, por un lado, la posibilidad de controlar el tamaño del radio de la región que incluirá los sonidos a combinar. Esto permite elegir un radio tan grande que abarque la totalidad de los sonidos en el mapa, como también uno tan chico que sea capaz de incluir unos pocos sonidos. Y por otro lado, la posibilidad de cambiar la curva de *crossfade* desde una curva exponencial, a una lineal y hasta una logarítmica y entre medio. Ambos parámetros son controlables a través de un *slider* o perilla.

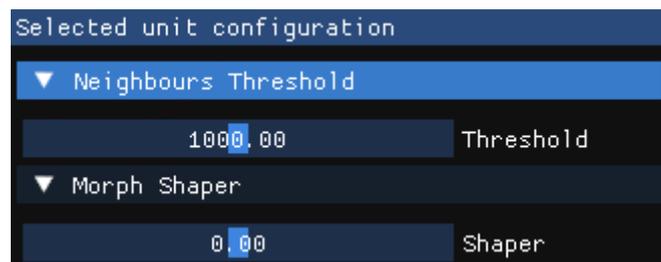


Figura 10: Interfaz gráfica de la Morph Unit

Por último, mi Unit incorpora una serie de funcionalidades que están disponibles en AudioStellar para ser implementadas en cualquier Unit, con el fin de expandir su funcionalidad. Una de ellas es la posibilidad de recibir mensajes a través del protocolo OSC (Open Sound Control). Estos mensajes están destinados a controlar la posición x-y del cursor en la Morph Unit, librando al usuario del uso del mismo para recorrer el mapa. De esta manera, los usuarios pueden enviar mensajes OSC a través de cualquier aplicación o lenguaje de programación que soporte dicha funcionalidad para controlar la Morph Unit.

Otra funcionalidad que incorpora esta Unit es la de las “trajectories” o trayectorias. En AudioStellar las trayectorias permiten grabar el movimiento del cursor y repetir este movimiento en bucle. De esta manera permite grabar una gestualidad del usuario y repetirla indefinidamente. La Morph Unit incorpora esta funcionalidad, permitiendo grabar un recorrido o barrido del mapa y perpetuarlo en el tiempo⁶.

⁶ Ver video: <https://www.youtube.com/watch?v=WeueV8P9GZc>

Experimentación y Resultados

Experimentar, escuchar, interactuar fue un proceso muy importante para poder familiarizarme con las posibilidades sonoras y tímbricas de mi trabajo. Sobre todo, para poder encontrar sus límites, alcances y posibles ajustes a realizar. Esta fue una idea impulsada principalmente por mi tutor, quien señaló “*conozco mucha gente, colegas, gente del laboratorio de inteligencia artificial en el DC muchas veces tienen un Corpus de audio de 10000 audios y no lo escuchan*” (Garber, 2021) .

En esta etapa de mi trabajo jugó un papel muy importante el dataset construido. Con este no solo pude probar mi desarrollo sino también las Units que ya existían previamente a mi trabajo para interactuar con el mapa bidimensional como también las distintas formas de agrupación de sonidos o visualización del mapa.

El dataset construido fue visualizado de tres formas distintas para experimentar con los sonidos. La primera forma o agrupación de sonidos en el mapa elegida es la que el programa nos sugiere por defecto, la de STFT o Short Time Fourier Transform. En este modo de visualización se agrupa a los sonidos según su similitud espectral, como se puede apreciar en la [Figura 11](#).

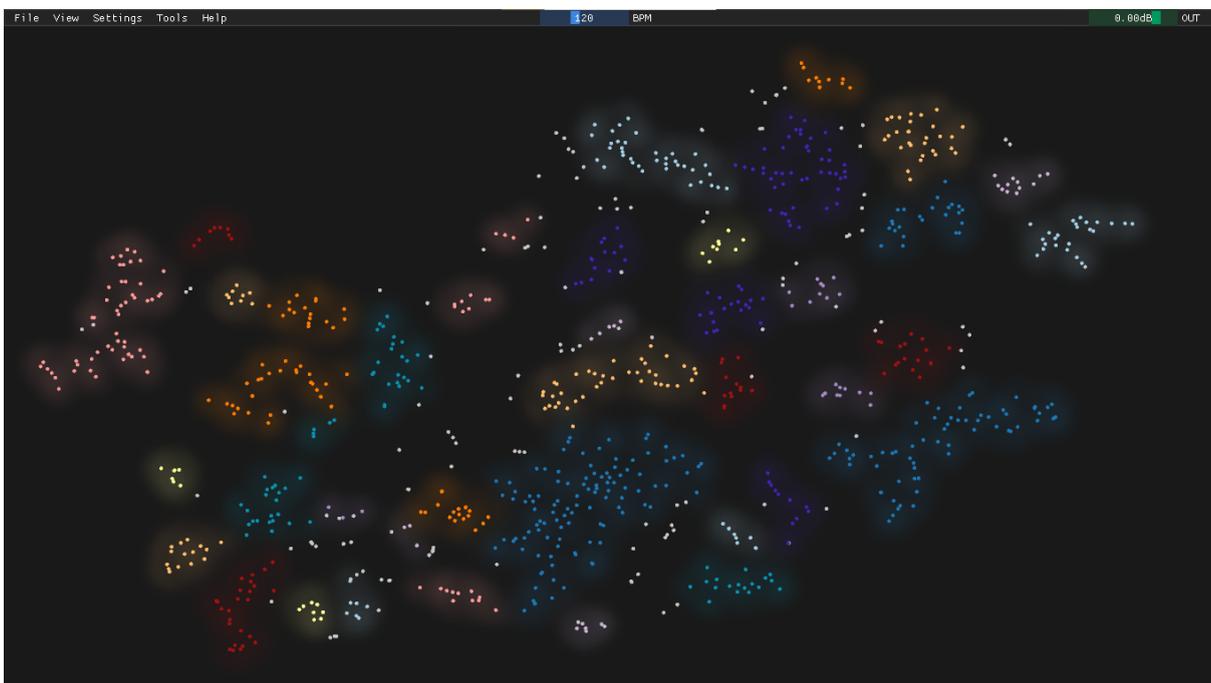


Figura 11: Visualización del dataset en el modo STFT

La segunda forma de visualización utilizada fue la de RMS o *Root Mean Squared* la cual agrupa sonidos según sus envolventes de volumen, es decir, que tiene en cuenta el desarrollo en amplitud a la hora de agrupar sonidos. Esta disposición del mapa se puede observar en la [Figura 12](#).

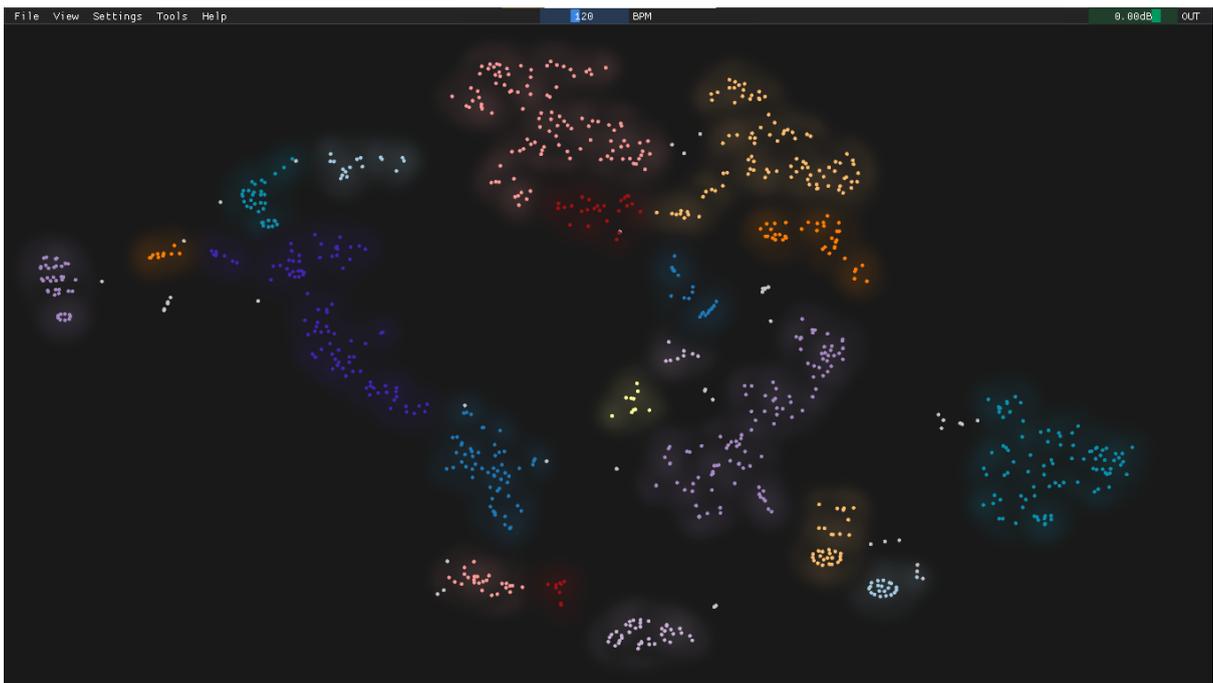


Figura 12: Visualización del dataset en el modo RMS

La tercera y última forma de visualización experimentada fue la de *Chromagram*. Este tipo de visualización agrupa a los sonidos según las notas musicales de la escala temperada como se puede apreciar en la [Figura 13](#).

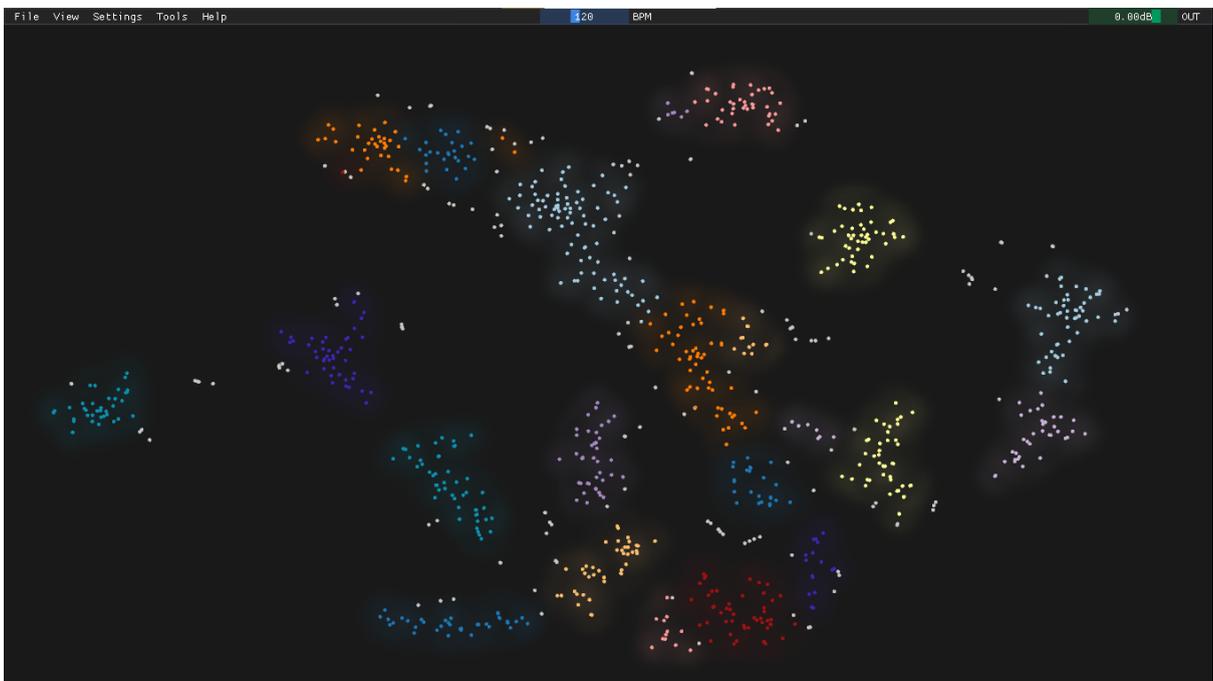


Figura 13: Visualización del dataset en el modo Chromagram

Visualizar mi dataset de todas estas formas me permitió descubrir nuevas relaciones y sonoridades entre los sonidos de mi dataset. Como mi desarrollo depende tanto de la relación entre los sonidos y de los vacíos entre ellos, probar estas formas de agrupación enriqueció la paleta de resultados.

Haber implementado una interfaz gráfica que me permita en tiempo real modificar parámetros como el radio de sonidos a vincular o la curva con la cual aplicar un *crossfade* de volumen fue muy provechoso a la hora de experimentar. Pudiendo probar distintas combinaciones de parámetros para escuchar distintos resultados. Algunos de estos resultados fueron grabados con el objetivo de documentar el funcionamiento del desarrollo. Para ello se creó un pequeño blog con el fin de reunir distintos videos documentando la experimentación. El blog puede ser visitado a través del siguiente link: <https://mmm.page/maximosigniorini>

Luego de experimentar con la Morph Unit, se llegaron a varias apreciaciones personales de la misma. A modo de detallar lo mejor posible cada una, serán organizadas por modo de interacción (clic o arrastre).

Clic

Al hacer clic, como se mencionó previamente, se posiciona la circunferencia y se reproducen aquellos sonidos dentro de la misma, con sus respectivos volúmenes en relación al centro de la circunferencia. Para sonidos con ataques muy cortos y tiempos de decaimiento largos esta forma de interacción no es tan enriquecedora. Sucede que cada sonido tiene un tiempo de ataque distinto, esto provoca que al ser reproducidos no todos empiecen al mismo tiempo. Esto hace que el oído pueda identificar con mayor facilidad cada sonido por separado, lo cual no es algo bueno si el objetivo del desarrollo apunta a conectar o mezclar sonidos. Por ello es que sonidos de tipo percusivos o con ataque y decaimiento rápido no funcionan de la mejor manera con este tipo de interacción.

En cambio, sonidos con tiempos de ataque y decaimiento prolongados se ven más beneficiados en este tipo de ejecución. Por ejemplo, si utilizáramos el modo de visualización *Chromagram* y tuviéramos agrupados a los sonidos por sus características tonales, uno podría fácilmente generar *clusters* o grupos de notas bastante complejos. De igual forma, si nos centramos en una región en la cual abundan sonidos más ruidosos, podremos generar complejas texturas que varían según en qué región del grupo se haga click.

Arrastre

Con el arrastre del mouse a través del mapa de AudioStellar, se producen los resultados más complejos y dinámicos. Aquí ya no importa demasiado si los sonidos tienen ataque y decaimiento rápido. Este tipo de gestualidad provoca una constante entrada y salida de sonidos. Como se ha mencionado previamente, la Morph Unit funciona principalmente a través de técnicas de crossfade de volumen. Pero en este caso se escucha un resultado un tanto más complejo que el de sonidos bajando y subiendo de volumen. Gracias a la posibilidad de agrupar sonidos según sus características tímbricas, esto hace que una simple técnica de crossfade de volumen se acerque más hacia técnicas de interpolación espectral. Una operación muy similar a la que realiza Wishart en *Red Bird*, pero con herramientas como AudioStellar y la Morph Unit la cual facilitan y potencian este proceso.

Procesadores de Efecto

La posibilidad de añadir procesadores de audio, enriquece aún más las posibilidades sonoras de la Morph Unit. Entre estos podemos utilizar Filtros, Reverberación, Delay, Chorus y un Saturador. Si bien todos estos expanden la paleta de posibilidades tímbricas, sobre todo la reverberación y el delay, es el chorus el que más destaca.

El chorus es un procesador de audio de tipo temporal, está compuesto por líneas de delay o retardo las cuales toman una copia de la señal de audio original y la desafinan o cambian su altura en el tiempo. Al sumar el audio original con la copia retardada y desafinada, se produce un efecto de *ensemble*, como si la señal original se hubiera multiplicado. Esto produce que el crossfade de volumen entre sonidos se amalgame y tenga un mayor campo estéreo. En otras palabras, este cambio en altura de los sonidos, sumado a su cambio en volumen fortalecen aún más esta idea de morph, de que un sonido se convierte en otro.

Por último, el desarrollo fue compartido con distintos colegas con el fin de enriquecer la experimentación. De esta manera el desarrollo podrá ser experimentado con distintos datasets, distintos sonidos y desde perspectivas distintas. Por ejemplo, un caso de uso interesante fue el de Nicolas Gulluni, colega compositor quien decidió utilizar mi desarrollo con instrumentos de orquesta con el fin de explorar nuevas sonoridades. Otros casos como el de Kevin Herzog fueron más simples, utilizando la Morph Unit con los sonidos que nos provee por defecto AudioStellar. En estos casos los resultados no fueron tan interesantes. Esto se debe a que los sonidos que trae el programa por defecto son de percusión o batería. Como se mencionó previamente este tipo de sonidos con tiempos de ataque cortos no funcionan de la mejor manera con las técnicas de interpolación. Otro aporte

interesante fue apuntado hacia la interfaz gráfica del módulo desarrollado. Stevan, un usuario de Perú del Discord de AudioStellar sugirió visualizar la circunferencia de la Morph Unit como los pinceles de *Photoshop* a los que se les puede ajustar su intensidad.

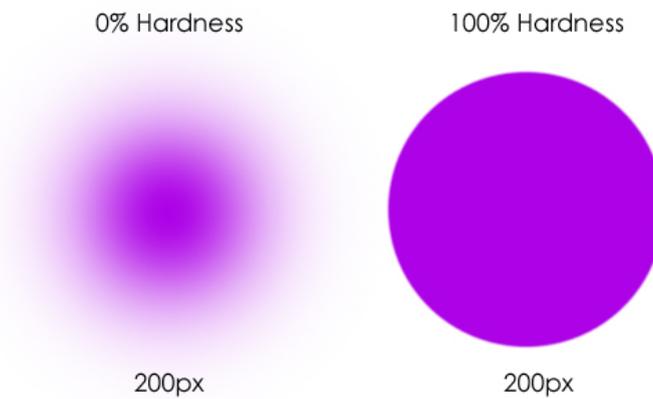


Figura 14: Dos herramientas de dibujo en Photoshop con distintos valores de intensidad.

Este cambio a la interfaz gráfica permitiría visualizar los valores de volumen de cada sonido dentro de la circunferencia. De esta manera pudiendo ver claramente aquellos con mayor volumen de manera más definida, mientras que aquellos que suenan más bajo apenas podrán ser visibles.

Machine Learning

En cuanto a las técnicas de Machine Learning, elegimos junto a mi tutor un enfoque exploratorio por las mismas razones que dejamos afuera las técnicas de síntesis concatenativa, mencionadas anteriormente. Ir por un enfoque experimental nos permitió evitar tener que implementar dichas técnicas en el programa y poder explorarlas por fuera. De esta manera podemos fácilmente probar algoritmos de Machine Learning, nuevos y complejos, para compararlos con las técnicas implementadas en AudioStellar.

Actualmente existen unas pocas técnicas experimentales de Machine Learning para realizar una operación de Morphing o mezcla entre sonidos, como se mencionó previamente. Optamos por dos desarrollos distintos para explorar. Por un lado, “*MelSpecVAE*” y por el otro la “*Differentiable Digital Signal Processing (DDSP)*”.

La *MelSpecVAE* es una arquitectura implementada por Moisés Horta en 2021. El significado del nombre de este desarrollo proviene de la arquitectura de redes neuronales utilizadas, las VAE o *Variational Autoencoders* son redes neuronales capaces de sintetizar espectrogramas Mel. A grandes rasgos permite combinar y mezclar timbres de distintos sonidos. Esta combinación y mezcla se logra a través del entrenamiento de las redes neuronales que aprenden a partir de sonidos a reconstruirlos, combinarlos y mezclarlos. El algoritmo nos permite interpolar entre dos puntos distintos en el espacio latente y sintetizar los sonidos intermedios, como también generar audio cortos a forma de *'one shot'*.

Estas redes neuronales fueron entrenadas con el dataset construido para el trabajo, con el objetivo de comparar los resultados sonoros con aquellos producidos por mi desarrollo. En este caso, los resultados de la *MelSpecVAE* dejaron mucho que desear. Tanto los sonidos cortos tipo “*one shot*” como los sonidos intermedios que generó la red neuronal resultaron con altos niveles de ruido, lo cual hace imposible determinar qué sonidos está combinando la red y mucho menos apreciar dicha combinación. Es posible que este resultado se deba a la poca cantidad de tiempo con la cual se entrenó la red neuronal. Sin embargo, la documentación del código provista por el autor es bastante reducida. No hay una explicación o señalamiento de la cantidad recomendable de horas a entrenar la red, ni tampoco una sugerencia de el formato de los sonidos con los cuales entrenar la red.

Por otra parte, se utilizó el desarrollo hecho por el equipo de investigación de *Magenta* en los laboratorios de Inteligencia Artificial de Google. El equipo desarrolló la librería conocida como DDSP o *Differentiable Digital Signal Processing* “(...) permite la integración directa de elementos clásicos de procesamiento de señales con métodos de deep learning. Centrándose en la síntesis de audio” (Engel et al, 2020) De dicha librería se utilizó la función de “Transferencia de Timbre”. Dicha función permite transferir el timbre de un sonido hacia el modelo de otro sonido entrenado por la red. Esto quiere decir que unx puede entrenar dicha red neuronal

con una colección de sonidos de algún instrumento o fuente sonora, para que la red sea capaz de sintetizar dicho sonido y de transferir su timbre a cualquier otro sonido. Se experimentó con modelos de flauta travesa, violín, trompeta, saxo y ruido de sintetizador. Dichos modelos fueron utilizados para transferir su timbre a otros sonidos como pianos, guitarras, voces, ruidos, etc. Los resultados sonoros fueron muy ricos y complejos tímbricamente. Estos fueron importados en el software *Sonic Visualizer* para poder apreciar visualmente los cambios en el espectro al combinar distintos sonidos. A continuación, un ejemplo de este análisis y los espectrogramas resultantes.

Para el presente ejemplo se entrenó a la red neuronal a partir de sonidos de flauta travesa y se utilizó el sonido de una voz hablada para transferir su timbre al de la flauta. El primer espectrograma con el que nos encontramos es el de la voz. Si nos detenemos en él, vemos que las frecuencias más prominentes aparecen entre los 172 Hz y los 258 Hz como también entre los 387 Hz y 473 Hz entre otras. Pero también vemos cierta periodicidad en la forma de onda, cierto ritmo que proviene de la cadencia del habla y los silencios.

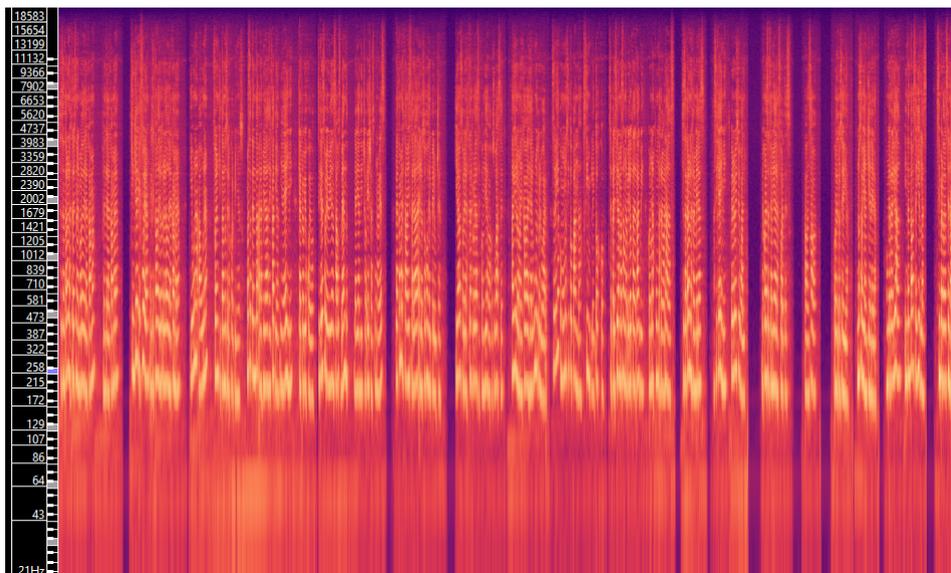


Figura 15: Visualización del espectrograma de la voz hablada

El segundo espectrograma es el de la flauta travesa ejecutando un La3 (Nota La, tercer octava). Si bien la red neuronal fue entrenada con muchísimos más sonidos de flauta, se muestra el espectrograma para comprender de manera general las características tímbricas del instrumento y como estas se combinan con el sonido de la voz. En cuanto a la flauta podemos observar frecuencias más prominentes entre los 328 Hz y los 515 Hz. En este caso, no notamos periodicidad ni ritmo alguno al ser una nota sostenida en el tiempo.

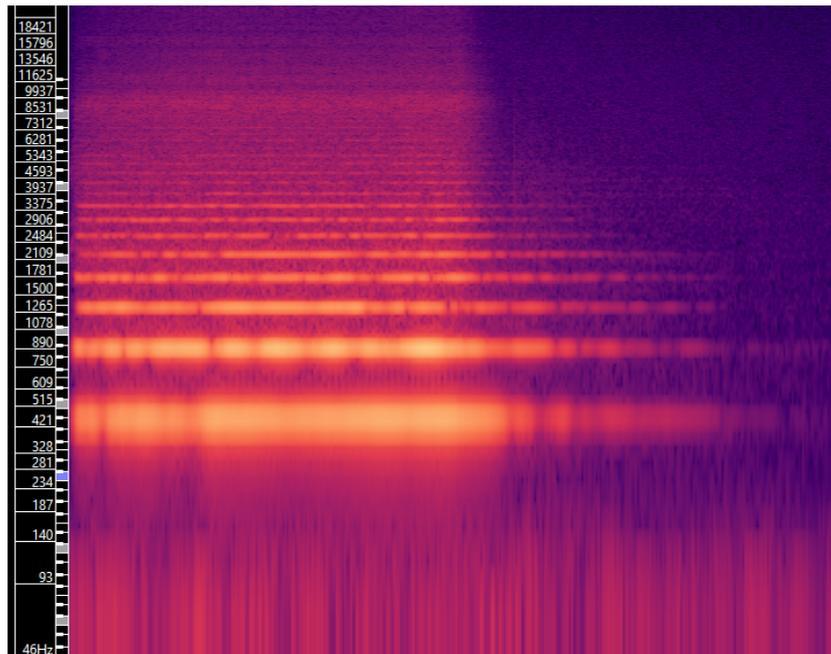


Figura 16: Visualización del espectrograma de la flauta traversa

Por último, tenemos el espectrograma de la combinación entre ambos sonidos. Aquí podemos notar algo interesante, las frecuencias más predominantes son casi idénticas a las de la flauta, con la diferencia que ahora hay cierto ritmo en el sonido, cierta modulación. Esto se debe a que la voz transfirió esta característica presente en el sonido al modelo de la flauta.

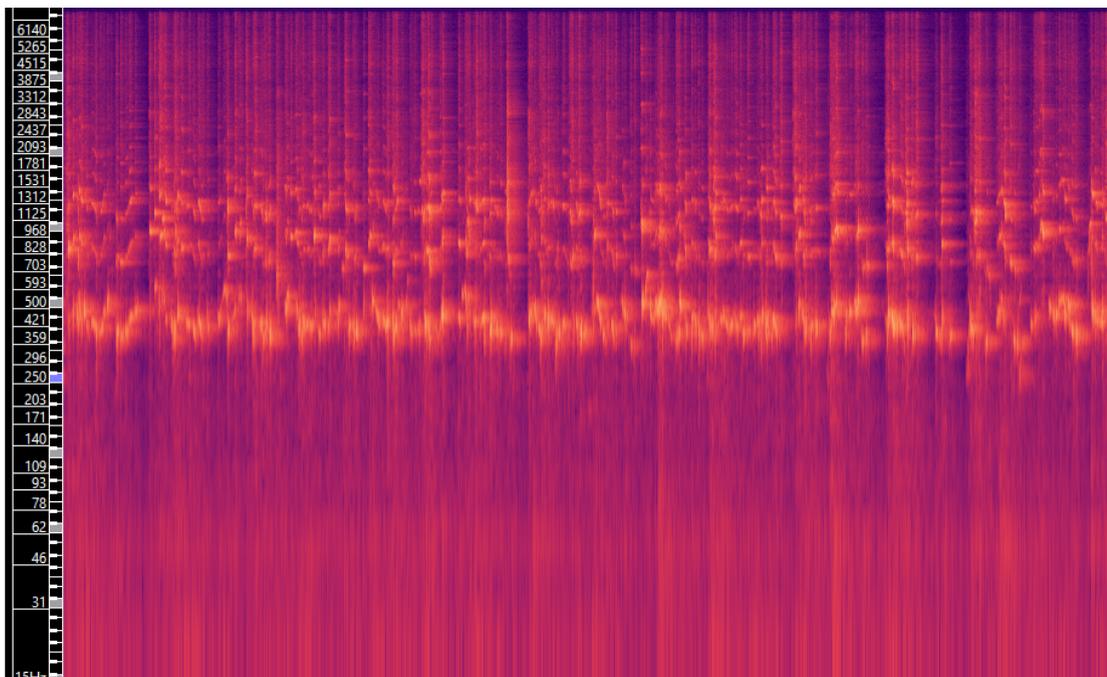


Figura 17: Visualización del espectrograma de la voz hablada combinada con la flauta traversa

Conclusiones

En resumen, fue exitosa la implementación y el desarrollo de una nueva funcionalidad que permita explorar los espacios vacíos en el mapa de AudioStellar.

Se logró el objetivo principal de este trabajo, hallar una posible respuesta a la incógnita de cómo relacionar los sonidos y cómo explorar estos espacios vacíos a partir de esta relación.

Como se mencionó previamente, si bien la técnica de crossfade de volumen es bastante simple y ampliamente utilizada, resultó muy poderosa y efectiva. Gracias a mi tutor, y sobre todo a lxs distintxs artistxs y desarrolladores de audio entendí el gran potencial que había en la técnica. Obras como *"Red Bird"* de Trevor Wishart fueron de gran inspiración para este trabajo. Entender que una obra tan compleja fue realizada solo con dicha técnica y con herramientas difíciles de operar como lo es la cinta analógica fue un incentivo para mi.

Durante el proceso del trabajo, y sobre todo hacia el final, entendí que era necesario comenzar con algo simple. Así lo define John Gall: *"Invariablemente se descubre que un sistema complejo que funciona ha evolucionado a partir de un sistema simple que funcionó"* (Gall, 1975). Algo tan simple como subir y bajar el volumen a distintos sonidos, se transformó en algo tan complejo como lo es la Morph Unit.

Es importante destacar que este trabajo no es un desarrollo que queda en mi computadora y nada más. Es un aporte a un software de código abierto, y más precisamente a un instrumento. Esto tiene grandes implicaciones. En primer lugar, porque va a haber gente que se descargue el software y lo utilice para crear música, lo cual para mi es algo tremendamente valioso. Y, en segundo lugar, porque al estar aportando a un proyecto de código abierto implica que cualquier persona puede contribuir a mi desarrollo⁷. Algo muy valioso del proyecto de AudioStellar es que se intenta construir una comunidad. Espacios virtuales como Discord o Facebook son muy valiosos no solo para compartir creaciones y performances sino también para aportar ideas nuevas al proyecto. Por ende mi desarrollo podrá evolucionar en un futuro, a partir del feedback de lxs usuarixs y de ver usos de casos distintos.

Esto hace que mi trabajo sea abierto en todo sentido. Por un lado abierto al uso de artistxs sonorxs que deseen experimentar y llevar al límite las posibilidades de la Morph Unit. Y por otro lado, abierto a desarrolladores que deseen modificar, expandir o rehusar mi desarrollo para uno nuevo.

Por otro lado, haber tenido la oportunidad de presentar mi trabajo en una conferencia como lo fue la de la AES (Audio Engineering Society) fue un gran logro

⁷ Código del desarrollo: <https://gitlab.com/ayrsd/audiostellar>

para mi⁸. Poder compartir mis ideas y desarrollos con colegas ingenierxs y programadores fue muy enriquecedor en todo sentido.

Pero otro de los objetivos primordiales de este trabajo era aportar un desarrollo significativo al software. Es por ello que mi trabajo será incluido en la próxima versión oficial de AudioStellar, la 1.2. Esto implica que cualquier persona que decida descargar el software podrá utilizar la Morph Unit.

Futuro

Este trabajo abre un potencial enorme para seguir investigando formas de explorar el mapa de AudioStellar. En principio restan explorar las dos técnicas que no fueron implementadas, las de interpolación espectral y las más novedosas que involucran técnicas de *machine learning*.

Día a día surgen nuevas técnicas y herramientas de audio en el campo de la inteligencia artificial. Estar al tanto de ellas y tomar provecho de las mismas para incorporarlas a AudioStellar será siempre un objetivo a futuro. Implementar alguna técnica como la que se utiliza en el sintetizador *NSynth* puede abrir un gran abanico de nuevas sonoridades (mucho más complejas, por cierto). Al tratarse de una interpolación tímbrica, naturalmente el resultado de esta técnica será distinto al de una interpolación de volumen como la que se implementó.

En este sentido, tener la oportunidad de formar parte del proyecto Estéticas Latentes permite continuar con esta línea de investigación. Junto con Laurence Bender y el resto del equipo, el enfoque del proyecto está dirigido a desarrollar y explorar estas nuevas técnicas de machine learning que permiten combinar y relacionar sonidos. El software "*pghipy*" desarrollado por Laurence y Leonardo Pepino es un ejemplo de ello. Entre otras posibles aplicaciones, el software permite reconstruir la fase de distintos sonidos para así poder interpolarlos tímbricamente, por ejemplo.

A su vez, no son solo nuevas técnicas que se pueden implementar en la Morph Unit. Sino también es posible implementar mejoras o variantes en la interfaz gráfica y en sus controles, como también nuevas formas de interacción con la unidad. Es decir, la idea central de la Morph Unit reside en definir un radio el cual engloba y conecta sonidos entre sí. Pero esta es solo una manera de explorar estos espacios vacíos, pueden existir muchas variantes que incluyan distintos tipos de interacción con el mapa por parte del usuario. Mi trabajo es solo una primera aproximación a esta exploración de espacios vacíos en el mapa de AudioStellar. El campo de investigación y desarrollo definitivamente está abierto a partir de este trabajo.

⁸ Link a la información de la conferencia: <http://argentina.aes.org/reservate-el-30-de-noviembre/>

Referencias

- Eck, Douglas. (1 de junio de 2016). Welcome to Magenta!
<https://magenta.tensorflow.org/blog/2016/06/01/welcome-to-magenta/>
- Engel, Jesse & Hantrakul, Lamtharn & Gu, Chenjie & Adam Roberts. (2020). DDSF, Differentiable Digital Signal Processing.
- Erickson, R. (1975). Sound Structure in Music: Berkeley: University of California Press.
- Gall, John. (1975). General systemantics, an essay on how systems work, and especially how they fail.
- Garber, Leandro & Ciccola, Tomás & Amusatogui, Juan Cruz. (2020). AudioStellar, an open source corpus-based musical instrument for latent sound structure discovery and sonic experimentation.
- Herndon, H. [@hollyherndon]. (4 de Noviembre de 2021). This is an example of what we call Spawning. Unlike sampling, which is a reproduction of sounds sampled from a recording, Spawning is the ability to create works in the likeness of others by interacting with a model trained on them. 21st century sampling, with big implications. [Tweet]. Twitter.
<https://twitter.com/hollyherndon/status/1456380312619995143>
- Jenkins, Mark. (2007). Analog Synthesizers, understanding, performing, buying.
- Magenta. (6 de abril de 2017). NSynth: Neural Audio Synthesis.
<https://magenta.tensorflow.org/nsynth>
- Miller, Darren. (2015). Compositional Outcomes of Audio Morphing Research.
- Wishart, Trevor. (1994). Audible Design.
- Wishart, Trevor & Emmerson, Simon. (1996). On Sonic Art.

Librerías de Sonidos

- **Baterías Electrónicas:**
 - Reverb Drum Machines. The Complete Collection.
<https://reverb.com/software/samples-and-loops/reverb/3514-reverb-drum-machines-the-complete-collection>
 - Sophie, Sophie Samples.
<https://splice.com/sounds/splice/sophie-samples>
- **Voces de hombres y mujeres** (exceptuando a los colaboradores mencionados en los agradecimientos): Freesound, <https://freesound.org/>
- **Paisajes Sonoros y Grabaciones de Campo:** Sound Ideas, Series 2000 Sound Effects Library.
<https://www.sound-ideas.com/Product/194/Series-2000-Sound-Effects-Library>

Anexo

- Código del desarrollo: <https://gitlab.com/ayrsd/audiostellar>
- Aplicacion compilada:
<https://drive.google.com/drive/folders/1vy2uMfRPeOuv6nxd4r2vOVITUGArrXS?usp=sharing>

Entrevista a Leandro Garber

(La siguiente entrevista se realizó en el marco de la materia Artes Electrónicas 3 junto a Kevin Herzog, Leandro Garcia y Mauro Santopietro)

Max: Lean, gracias por aceptar la entrevista, y si quieres podemos empezar por una presentación tuya. Si quieres contamos un poco de tu background artístico, un poco de tu background técnico si quieres también.

Garber: Okey, bueno, yo soy Leandro Garber, soy Licenciado de la Licenciatura de Artes Electrónicas. Antes estudié un par de años en Ciencias de la computación en la UBA. Bueno, no, fueron un par de años después dejé y quise como... Osea, me di cuenta que me interesaba la tecnología, me interesaba la programación. Yo programo desde muy muy chico, antes de entrar a la secundaria ya sabía programar. Mis primeros trabajos empecé ahí. Medio que aprendí a leer con la compu, siempre fui muy compu. Entonces era obvio que iba a seguir Ciencias de la computación, pero cuando estaba ahí y me gustaban las materias de matemáticas, álgebra. Pero cuando empecé la parte de computación, no, no me no me terminó de cerrar. Y quería poder usar lo que sabía de programación desde un lugar, si, como más creativo, más artístico, más poético. Así que nos pareció que quizás en artes electrónicas iba a poder encontrar ese lugar. Así que bueno, me cambié, hice la licenciatura. La terminé ya hace bastante, algo así como 6 u 8 años, espero no estar mintiendo, es algo así. Y unos años después empecé una maestría, que es la maestría en, yo le digo maestría en Machine Learning. En realidad se llama maestría en explotación de datos. Se llama maestría en explotación de datos y descubrimiento del conocimiento en la UBA, que es una maestría que se da en el DC, en el Departamento de Computación. De esa manera, como que me interesaba volver al DC, que es donde había estudiado antes. Pero bueno, ahora haciendo la maestría que dura 2 años. Y bueno, ya tenía todo este bagaje de artes electrónicas y bueno, tenía ganas de aprender algoritmos de inteligencia artificial y me anoté ahí. Obviamente la maestría no tiene nada de artístico. Diría que todo lo más contrario que podría llegar a ser onda bancos y medicina, esos eran los ejemplos. Entonces yo todo el tiempo de la maestría, bueno, me preguntaba cómo podría usar esto de manera creativa? Y fue difícil. Pero bueno, hice, empecé y en un momento empecé a usar ciertos materiales que uso en lo artístico. Por ejemplo, el sonido, samples de sonido en particular. Y empecé a probar algunos de estos algoritmos en sonido. Empecé pensando, haciendo efectos de sonido. Con ciertos algoritmos, componentes principales, bueno, algunos algoritmos conocidos. Bueno, daban algunos resultados. Y finalmente probé algunas técnicas que me llevaron años después a estar con el proyecto de AudioStellar

Max: Osea que AudioStellar, digamos, surgió posterior, digamos a la maestría. ¿O durante?

Garber: Bueno, yo ya termine, tengo el título de especialista, hay 2 títulos, uno a primer año que es especialista y en el segundo año me dan la maestría, Magister sería. El especialista, lo que entregue es un trabajo que sería como una previa a AudioStellar. Donde más o menos es, che, esto con sonido funciona. Incluso en mi trabajo lleve un parlante. Todos estaban como "Oh un parlante!". Incluso a los que me evaluaban estaban como "Que onda, que está pasando?" Y nada, les mostré, había hecho un ejemplo muy chiquito en web de que estaba agrupando las cosas bien. Bueno, nada con eso probé mi especialización,

después me di cuenta de que obviamente ya existían este tipo de experimentos, ya había toda una rama muy importante, una disciplina que se llama MIR (Music Information Retrieval). Que trata este tipo de cuestiones. Pero bueno, sí, es un poquito el trayecto que hice. Y después, bueno durante la carrera y después también presente algunas obras en distintos lugares.

Max: Claro, desarrollar mas tu lado artístico en la carrera, digamos.

Lean: Lean, yo quería preguntarte, porque dijiste que desde chico ya estabas como relacionado con la computación, y como que aprendiste leyendo a través de la computadora. ¿Y qué es lo que te conecta con el lado más artístico? Digamos como para decir, che, esto así, computación estructurada no es lo mío, sino algo más relacionado a ese costado. Qué fue eso que quizás, no se, te llamó la atención o te hizo como enfocarte para ese lado.

Garber: Creo que no, no es que haya visto una obra y ha dicho “Wow esto existe!”. No fue así. Si me daba cuenta que me gustaba muchísimo la música porque tocaba. En ese momento tenía banda que duró varios años, que nos presentamos un montón, etcétera. No era una banda bueno, había sintetizadores, pero yo tocaba la guitarra y cantaba. Y me di cuenta que me gustaba mucho la música, que era una pasión, digamos como que era algo que no iba a dejar de hacer nunca. Y, de repente, sentía que me estaba metiendo en algo que no... Que no tenía la suficiente creatividad. Y bueno, fui para el lado de artes electrónicas. Pensando que iba a encontrar alguna respuesta. Y bueno, algunas respuestas aparecieron, algunas posibilidades. Pero si no sé si hubo así como un corte. Se que en un momento me fui de vacaciones y ahí fue como “Che, no, o sea, tenes que seguir buscando un poco más”. Tenemos la suerte de vivir en un país que uno se puede cambiar de carrera. Digamos no, no, no todos tienen ese privilegio. Y bueno, yo lo tomé.

Lean: De ahí, entonces es que es que sale tu especialización en esa rama del arte, digamos en los algoritmos y en el sonido, en este caso. Porque bueno, una de las preguntas que charlamos con los chicos es, bueno, porque, o sea, te especializaste en esas ramas y no, no sé, en otras como puede ser algo plástico, esculturas u otro tipo de arte.

Garber: Sí, bueno, mira la realidad, si ustedes ven las distintas obras que presenté antes no son sonoras, son visuales. Yo en realidad nunca me sedujo mucho el arte visual en cuanto a yo hacerlo. A mí obviamente me encanta. Pero nunca tuve una formación, ni tuve familiares ni nada digamos que me acerquen mucho a lo visual. Sin embargo, también un poco la historia que se contaba, por lo menos en ese momento en artes electrónicas, era mucho en la historia del arte visual. Eso pasa mucho en las artes, que la historia de las artes parecería ser la historia de las artes visuales. Cuando hay muchísima, obviamente, muchísima historia de la música, pero a veces la música se le separa un poco. Yo empecé haciendo mucho arte generativo de distintas maneras, pero visual. Porque es un poco lo que más se ve, porque es un poco lo que conocía. También, como yo tengo una formación de música tradicional, yo fui al conservatorio de muy chico, mis familiares están muy vinculados a la música. Yo igual después toque más, no sé, rock, canción, etc. Pero siempre en la música más tradicional, por decirlo de una manera, más popular, si se quiere también. Y me costó mucho y me sigue costando al día de hoy hacer otras músicas. Justo recientemente con AudioStellar y eso es que empecé a involucrarme, a más en hacer música algorítmica o

estocástica, o bueno, arte sonoro, instalaciones sonoras. Pero es una linda pregunta por esto de que es verdad que al principio empecé como algo, hice todas cosas visuales. Sigo haciendo alguna cosita visual, pero la verdad es que lo que más me apasiona es la música.

Max: Y volviendo un poco a AudioStellar, quieres explicarnos así, muy brevemente, digamos que es AudioStellar.

Garber: Sí, claro. Bueno, AudioStellar es un proyecto que empezamos hace ya 3 años. Lo empezamos juntos con Tomas Ciccola. Yo solamente tenía hecho esto que les digo que era como una previa, que era mi, como una tesina, ni siquiera, un trabajo final de de especialización con el que me dieron el título. Y bueno, ya cuando hice eso y funcionaba en browser en un navegador dije “Esto podría ser más que esto, no?”. También ahí ya se planteó una pregunta, o sea, ahí se plantea una pregunta, les cuento un poquito igual de qué se trataba. Tenemos una carpeta de archivos de audio, podrían ser 100, 500, 1000 o 10000 audios, 10000 archivos de audio. Imagínense una carpeta, sin carpetas adentro, con 10000 archivos de audio, y yo te pregunto “Che, que tiene esta carpeta?”. Y bueno, escuchas uno por uno y bueno, este tiene un bombo, este tiene un pájaro, este tiene un bongo, esto una persona hablando, bueno el universo de lo musical es infinito, va, de lo sonoro en realidad. Entonces el software lo que hace y la prueba está que hice yo antes es cargar todos estos audios y aplicar ciertos algoritmos de Machine Learning para organizarlos en un mapa 2D. Ósea, permite visualizarlos básicamente. Bueno, ahí ya vemos el cruce entre lo visual y la música no? Están ahí. Pero básicamente tenes un mapita de 2 dimensiones con puntos. Cada punto es un audio, un archivo de audio. Y lo interesante es que los sonidos semejantes, o sea, que suenan parecido, que tímbricamente son parecidos. O, en realidad, espectralmente son parecidos, los agrupa juntos. Entonces vamos a tener pequeños comulitos de puntos, de sonidos similares. A la vez, un cúmulo de puntos y otro, que les llamamos clusters a estos cúmulos, cuando están cerca, también en general, se parecen entre ellos, entre los grupos. Entonces hace esta técnica que se llama reducción de dimensionalidad. Se llama así porque se piensa a cada archivo de audio como un punto en una dimensión, en una alta dimensión. Esto es matemáticamente, porque tenemos la dimensión 2D, 3D, 4D, 5D, bueno, imagínate 44000D. Ahí tenemos un punto en esta dimensión. Yo te lo hago así, medio en 3D, pero en realidad no existe en la cabeza. Y hay ciertas técnicas que permiten reducir esta dimensionalidad hasta 2 y bueno, te queda este mapa. Hasta ahí el interrogante que apareció y es ese interrogante que aún nos sigue, digamos, es: Bueno, tenemos este mapa, tenemos estos puntos, ok, está bueno para explorar esta colección de sonidos. ¿Pero qué más, no? Porque acá hay una interfaz, acá hay una metáfora, acá hay como ciertas reglas de juego que se podrían explorar desde lo sonoro, desde lo musical porque de repente aparece una idea de distancia, aparece una idea de espacio. Bueno, ¿cómo se puede aprovechar eso? O si eso puede generar nuevos sonidos, nuevas lógicas de trabajo. Esa sería medio la gran pregunta de AudioStellar y más o menos la gran pregunta de investigación, una de las principales que tenemos.

Max: Claro si no solo ya desde lo que dijiste, digamos de los algoritmos de Machine Learning desde lo sonoro, ya con el hecho de visualizar todo eso abris una dimensión nueva combinada con el sonido que te lleva a todas estas interrogantes que estás diciendo. Que quizás laburando solo desde el sonido no se te hubieran ni ocurrido o no salía a la vista quizás.

Garber: Sí, trabajar los samples, nosotros lo pensamos como un sampler experimental. Trabajar los samples como grupos. Cómo a partir de este mapa, ya no tenemos unos botoncitos tipo el Akai, donde está muy la idea del sample como individuo. Tienes individuos en botoncitos que se le aplican efectos. Acá ahora tienes grupos. Bueno, qué podemos hacer con estos grupos? Y bueno, no sé, podemos investigarlo. Va un poco por ahí. Esa es una de las grandes preguntas. Y que está relacionada con otra pregunta que sería, ¿cómo se puede intervenir ese mapa para tocar esos sonidos? ¿Cómo tiene que ser esa interfaz? ¿Cómo tiene que ser esa relación humano-máquina? o ¿Cómo puede ser para que sirva para algo sonoro o musical?

Max: Claro, porque una de las preguntas que también nos surgió es ¿Desde que surgió el proyecto pensaron a AudioStellar como un instrumento ya desde el principio? O quizás no tanto y era algo quizás más utilitario como para organizar sonidos y ya.

Garber: No, la verdad que desde el comienzo lo pensamos como instrumento. O sea, de hecho, sacamos un paper este año, lo presentamos en la conferencia CMC Computer Music Conference. Y ahí pueden ver que el abstract, o sea, el paper que hicimos, dice “Esta representación no sólo es buena para explorar librerías, sino que cuando lo usas con una interfaz para un humano puede ser usado como instrumento musical. Esa fue la premisa desde siempre. Algo que me gustaría agregar es que con Tomas Ciccola también nos surgió la posibilidad de en Muntref Arte y ciencia empezar un proyecto de investigación. Nos ofrecieron formar parte de un proyecto, empezar un proyecto nuevo. Esto nos ofreció Mariano Sardón en su momento. Y bueno, y ahí empezamos a tener, a dedicar horas a bueno a programar este tipo de cosas y ver que surge.

Lean: Lean y con esto estás diciendo, ¿Cómo concebís vos AudioStellar? O sea, si tuvieras que definirlo en pocas palabras, ¿Qué es, más una herramienta más un instrumento o qué significa para vos? Digamos, si tuvieras que darle una definición así, simple digamos, ¿Qué representa para vos esto?

Garber: No, yo, hoy por hoy, antes estábamos llamando instrumento musical tipo sampler, hoy lo llamaría a un sampler experimental. Pero bueno, es una definición para alguien que sabe que es un sampler, ¿no? Me cuesta pensar que es un instrumento. Me parece que sampler es más genérico y que se puede usar en distintas situaciones. Sobre todo porque un instrumento parecería ser que es para tocar en vivo o tocar para hacer música. Ahora estamos haciendo instalaciones, también puede salir para performance. También puede ser simplemente una herramienta para hacer Foley, no sé.

Max: Claro, sí, es como que está en los bordes de lo que se considera normalmente un instrumento, no?

Garber: Sí, sí, prefiero pensarlo directamente como un sampler experimental y en el caso de tener que explicar sampler se explicará y ya. En general, como estaba orientado a músicos en general se conoce el sampler.

Max: Y volviendo un poco quizá a los inicios de AudioStellar, ¿qué fue lo que te motivó a empezar AudioStellar? Quizás, no sé, viste algún desarrollo similar ¿o simplemente tuviste la idea de cómo se podría hacer esto?

Garber: No, la verdad que estaba muy obsesionado con poder relacionar todo lo que estaba estudiando con arte y la verdad que no podía, no podía relacionarlo. O sea, sentía que no estaba pudiendo conectar los puntos, valga la redundancia. Y nada, empecé a probar con audio. Pero no, no vi ninguna obra. Después fue que empecé a ver artistas, también no sólo artistas simplemente demos. Hay una demo de Google que es web que hace algo parecido, o sea, es como un ejemplito. Lo vi después y dije, "Uh, alguien ya lo hizo". Pero después me di cuenta que en realidad nuestro proyecto tiene un vuelo bastante distinto. Hay otras aplicaciones también que lo hacen, pero todo eso lo vimos medio después, cuando empezamos a hacerlo. Empezamos a ver qué cosas eran similares. Nos enteramos medio después, es algo que sucede mucho eso. Sobre todo porque siempre son cosas muy de nicho y que bueno, te metes en el nicho cuando ya estás haciendo algo capaz. Pero no, no hubo nada así en particular que nos haya motivado. Solamente esta motivación de juntar algún elemento artístico con estos temas que venía viendo. O sea, justo en esta materia que tenía que entregar, uno de los temas es la reducción de dimensionalidad.

Lean: A la vez entonces, si tenes que hablar de referentes, no, no tenés ejemplos claros. O sea, que fue algo que surgió a vos. No es que viste como te despertó esa esa llama la obra de un artista o el trabajo de un desarrollador. Todo esto lo viste después de haber empezado este proyecto.

Garber: Sí, y también el proyecto empezó desde un lugar mucho más técnico y mucho más relacionado con la ciencia, relacionado con la técnica, relacionado con la programación. No es que dijimos, "Bueno, vamos a ver artistas que hicieron algo parecido". Obviamente conocíamos, no sé, por ejemplo Ryoji Ikeda como para nombrar a alguien súper referente. Pero nunca jamás pensamos algo así como "Che, hagámoslo porque quizás se parece a algo de lo que hace él". No, para nada. Si conocíamos artistas que laburen, sonido algorítmico, que habíamos visto en la facultad. Incluso yo no hacía esa música, nunca hice esa música. Por esto les digo que para mí también es un desafío poder desvincularme de las escalas, del estribillo, del verso. Desvincularme de ese tipo de cosas a mí me cuesta mucho. Pero bueno, en los últimos tiempos realmente estoy avanzando en ese sentido. Pero la verdad que mucho tiempo fue algo técnico que iba a servir en algún momento para artistas. Pero fijate que el artista era alguien ajeno en este caso. Pero bueno, al final se dio que la empezamos obviamente a usar.

Lean: Lo pensabas que podría hacer algo que le ayude a otros artistas, o sea, también estaba dentro de su lógica como esa opción importante.

Garber: De hecho era la única opción en su momento. Este era un desarrollo para artistas sonoros. Empezó siendo así. De hecho dijimos, bueno para artistas sonoros, capaz les sirve también a gente que labore en cine, que haga sonido. No teníamos muy claro el target, la verdad. Hoy lo entendemos un poco más. Pero como les digo, igual es variado. Obviamente está principalmente orientado a artistas sonoros. Pero bueno, también la verdad que, por ejemplo, si uno labura con sonido. Por ejemplo, hace Machine Learning y tiene un corpus de audio. Parece ridículo lo que voy a decir, pero yo laburo en ciencia, con sonido y conozco mucha gente, colegas, gente del laboratorio de inteligencia artificial en el DC muchas veces tienen un Corpus de audio de 10000 audios y no lo escuchan. Como que sí, escuchan 1, 2, 5, 10, random, 20, pero después es todo visualizarlo, ver lo que dan los numeritos. Pero

muchas veces esto puede servir simplemente para hacer un proceso súper complejo, lo que uno quiera en Python y después cargarlo en AudioStellar y escucharlo, simplemente escucharlo.

Max: Como que trabajan con la materia prima, así, medio ajenamente.

Garber: Sí pasa, re pasa con lo que estás haciendo. Pero es como “Ah, no, no te sentaste a escucharlo media hora, 20 minutos”. Pero además no porque sean algo sino porque no hay una herramienta fácil para hacerlo, no es fácil. Igual, la realidad es que tampoco nos estamos acercando tanto a esos lugares. Pero es una de las posibilidades y yo sé que hay gente que lo está usando. Les digo cuales son los targets de nuestro proyecto.

Max: De hecho, una de las preguntas que teníamos era si tenías pensado armar una especie de comunidad a partir de AudioStellar o una especie de intercambio entre la gente que lo usa.

Garber: Y, ese es mi sueño, poder generar una comunidad. Es un proyecto Open Source, no lo dijimos, pero es de código libre. Mi sueño siempre fue tener un proyecto de código libre, tuve otros que obviamente fracasaron o no los termine o supe hacer. Este es el primer proyecto de código libre que más o menos se está usando, que más o menos está teniendo algo de difusión. Y que me encantaría poder formar una comunidad. Es muy difícil, pero bueno, lo estamos intentando. Es un proyecto muy joven, los proyectos de este tipo para que tengan una comunidad pueden tardar hasta 10 años. Así que nada, vamos probando. Si, por ejemplo, en un momento pensamos que iba a ser más fácil, que se iban a acercar programadores, que íbamos a poder contar con colaboraciones. Y un poco sí, pero es mucho más difícil de lo que parece. Incluso una de las primeras veces mostré el proyecto en la NIME, que es una de las conferencias más importantes relacionadas con el proyecto, New Interfaces for Musical Expression o Nuevas Interfaces para la expresión musical. Es una conferencia que congrega a científicos, artistas, músicos, musicólogos, teoría de la música, bueno, todo eso y cada uno habla de su cosa medio interdisciplinar. Yo mostré una demo de AudioStellar. Fue medio como un formato póster, pero también con el programa y se podía escuchar. Y se acercó en un momento el organizador de la conferencia, no de esa, sino de la que iba a ser el año que viene, que se iba a hacer en Inglaterra y al final no fui pero además fue pandemia, así que nada, imposible. Se me acercó y me preguntó “Bueno, qué vas a hacer, vas a comercializarlo?”. Lo normal. Y fue como “No, estoy haciendo una comunidad”. Y el chabon lo que me dijo fue “Ah, bueno, mucha suerte con eso”. A lo que yo le pregunté “Ah, es difícil?”, “¡Es muy difícil!” me dijo. Pero bueno, uno ve proyectos, por ejemplo, un proyecto como el Bela, no sé si lo conocen. Bela es un una plaquita tipo Raspberry Pi o Arduino para hacer instrumentos musicales y tiene una super comunidad, una muy buena comunidad. Entonces yo veo ese tipo de proyectos Open Source, Open hardware en ese caso, también veo otro tipo de proyectos, como Godot, por ejemplo, acá en Argentina, que es para hacer videojuegos y que también estuve muy cercano en su momento. Y nada, veo que pueden generar comunidad. Pero bueno, son otro tipo de proyectos y tardaron años y bueno, estrategia. Nuestro objetivo es poder hacer una comunidad. O sea, ese es mucho más nuestro objetivo que comercializarlo. Realmente nos gustaría poder nunca comercializarlo. Seguro siempre lo queramos hacer gratis. Quizás en algún momento cambiemos el modelo, si es que no da para más. Pero bueno, sí, es nuestro objetivo principal ese.

Lean: Justo una de las preguntas que también habíamos charlado es ¿Porque Open Source? Es decir, bueno, ¿Porque justamente es uno de sus objetivos que sea abierto para el libre uso?

Garber: Es una linda pregunta. La verdad que el Open Source siempre fue una de mis pasiones: la música, el Open Source y la programación. Yo uso Linux desde muy chico, desde los 15 años. De hecho, recuerdo haberle dicho a mi viejo cuando tenía 15 años, yo viví casi toda mi vida hasta los 20 años en Martín coronado, 3 de Febrero cerca de Caseros. Y siempre recuerdo que una vez me dijo “Che, y qué vas a querer hacer?”. Y yo le decía que quería hacer como una escuela de Linux en Martín Coronado, entendés? Es un proyecto que no tiene sentido si lo miras ahora. Pero yo siempre como que quería enseñar Linux porque como les digo, siempre estuve cerca de la compu y siempre me apasiono mucho aprender cómo funciona hasta el último detalle. Ahora igual, por suerte, estamos entrando en procesos un poco más generales. Pero de chico hasta el fondo quería saber y Linux es lo único que te permite eso. O sea, si vos querés saber los sistemas operativos, tenes que usar Linux, sino no sabes sistemas operativos. Sabes que hay un menú en Windows que no sé qué... Pero en Linux entendés cada partecita y eso a mí me entusiasma. Porque la idea de romper con la caja negra siento que te libera de alguna manera, que te permite inventar lo que quieras. No sé, estoy como muy relacionado con lo que se llama la ética hacker del libre conocimiento, de que todo el mundo tiene el derecho de acceder a toda la información que existe. No sé, son como ciertas cuestiones morales, filosóficas y políticas que al día de hoy nunca me dejaron. Y lo que nunca te dejó es porque no vas a dejarte nunca. Entonces, medio que por eso sacamos AudioStellar como Open Source. Además es una plataforma para inventar cosas. La idea de que AudioStellar no es una respuesta a algo, es una pregunta. Bueno, qué hacemos con esto?. Y yo les digo, bueno, pongamos unas partículas que se mueven en el espacio y suenan. Maxi dice, bueno, que suene muchas a la vez a distintos volúmenes y que se vayan transformando. También secuencias, hagamos secuencias en el espacio que tengan que ver con la distancia. Pero hay maneras de hacer esto, podría ser con el cuerpo, podría ser conectándose el cerebro y con las ondas cerebrales modificar el mapa. Entonces siempre es un lugar para inventar. Usa el protocolo OSC que sirve para mandarle data, sirve para que cada uno arme lo suyo, para que cada uno se lo apropie. Entonces no puede no ser Open Source, si bien hay proyectos que no son así, el Max por no es Open Source. Pero bueno, de alguna manera también le debo un montón al Open Source y es como una manera de dejar mi aporte.

Max: Y un poco también, siguiendo más que nada con la parte técnica del Open Source, me imagino que también por ese lado es que decidieron comenzar a desarrollar AudioStellar en lo que es Openframeworks. Para quien no lo conoce es un entorno de código creativo que también es Open Source. Está orientado para artistas en general y permite crear desde obras de arte visual hasta AudioStellar.

Garber: Si, elegimos Openframeworks porque es rápido y por lo que vos decís. O sea, acordate que queríamos hacer una comunidad, entonces queríamos que se acerquen programadores sedientos de colaborar con el proyecto. Y estos programadores era muy posible que sepan Openframeworks. En realidad lo habíamos empezado a hacer en

Processing en algún momento. Pero rápidamente nos pareció que íbamos a necesitar mucha performance y Openframeworks nos daba eso así que fuimos por ahí.

Max: Hoy por hoy, ¿en qué estado está el proyecto?

Garber: Bueno, está en un muy buen estado porque hace pocos meses sacamos la versión 1.0 después de 3 años. Digo 3 y quizás hasta me estoy quedando corto. Siempre tengo que revisar, yo soy pésimo con las fechas, siempre estoy diciendo 3 años, tengo miedo de estar diciendo 3 años hace un montón. Pero es hace más o menos 3 años, 3 años y medio, con sus altibajos. Sacamos la versión 1.0 que es muy estable, finalmente. Es muy difícil hacer un desarrollo tan complejo, multiplataforma y estable. Es muy difícil. Porque pasan muchas cosas y uno no tiene 1000 máquinas. Entonces, la verdad es que estoy como súper orgulloso de mí, de todo el equipo. Haber podido sacar esta versión estable significa que lo abris, lo usas y no se rompe, anda, puede estar horas, días, semanas prendido y no le pasa nada. Todo funciona como debería, no hay errores extraños, estafalarios. Pero si está en un estado que todavía se puede mejorar hasta el infinito. Aparte, es también un poco lo que quiero de la comunidad, que agreguen maneras de de interfacear con AudioStellar. Hay muchas ideas, pero bueno, siempre hay que sentarse a desarrollarlas.

Max: Bueno, esa era otra pregunta, cómo te imaginabas que seguía AudioStellar. Por un lado estaría esta idea decís de que un poco la comunidad contribuya o experimente con AudioStellar pero no sé si hay alguna otra idea, algún otro deseo, algún otro desarrollo por venir.

Lean: Claro, quizás más de ustedes. O sea, no, no en base a lo que puede hacer la comunidad, sino bueno, ustedes como creadores que les gustaría o que están pensando en agregarle a la versión que está ahora.

Garber: Bueno, en principio, la idea siempre es sacar por lo menos 2 versiones por año. Ahora sacamos la 1.0, queremos sacar la 1.1 en breve. Nos gusta este modelo de hacer releases, o sea, nuevas versiones rápido. No con 1000 millones de cosas nuevas, sino con un par de cosas nuevas, pero sacar todo el tiempo, no dormir con eso. El año pasado sacamos 3 versiones, por ejemplo. Ahora por lo menos dos, si da el tiempo para tres, tres. Nuestro equipo creció mucho este año, el equipo de investigación ahí en Muntref Arte y Ciencia. El equipo se llama "Análisis y representación sensible de datos". Comprende otras cosas que AudioStellar, aunque es el proyecto principal. Ahora estoy yo como director, está Tomás como codirector y hay cuatro becarios ahora, hay varios tesis. Entonces está creciendo mucho y bueno, qué hacemos con eso? En principio queríamos mejorar la interfaz, que sea más fácil de usar. Queremos agregar un montón de tutoriales porque al día de hoy si bien tiene un pequeño tutorial al principio queremos hacer muchos videos tutoriales. Que se entienda más, abrirlo más, seguir presentándonos, dar workshops, etc. También entender algún modelo de negocio que nos sirva para mantener el proyecto. O sea, por ahora está todo bien, pero tengo el miedo o la idea de que en algún momento vamos a necesitar algún tipo de modelo de negocios que no necesariamente sea vender el software. Que nos sirva para mantenerlo, para mantener los gastos que tengamos, para poder seguir creciendo. Desde donaciones hasta lo que sea, eso también, vamos a ver cómo funciona. Y nada, viajar, viajar con el software, hacer workshops por todo lo que podamos, seguir presentándonos en conferencias, relacionarnos con otros grupos de

investigación, ver cómo podemos generar colaboraciones desde ahí y, más que nada, y es algo que se está dando cada vez más, hacer música, hacer mucha música con el software. Porque al hacer música, o arte sonoro, como quieran llamarlo, yo lo llamo música porque yo quiero hacer música. Al hacer música ahí te das cuenta más lo que haces. Igual es muy difícil el perfil de ser programador y ser a la vez el que lo usa y es músico. Pero bueno, me estoy amigando un poco con esa idea.

Max: Y volviendo un poco a la comunidad, a la gente que usa AudioStellar, y hablando también de sacarle un poco el jugo y de que viaje el software ¿alguna creación que alguien haya compartido o que hayas visto te llamó la atención? ¿Que cosas que la gente hizo con el software te llamaron la atención que quizás no estaban dentro de tu imaginario de lo que se podía hacer con AudioStellar?

Garber: Hay un par que sé que que hay que no me llaman tanto la atención en cuanto a que es una de las posibilidades que más o menos podía ver, básicamente, músicos, haciendo música experimental, capaz con interfaces raras, con un anillo que mueve AudioStellar. Lo que sí me sorprende es un chabon en Israel haciendo eso, Agustín Spinetto en Japón haciendo una maestría con el tema AudioStellar, ver todo un título en japonés y AudioStellar en latín, con las letras en latín. Eso sí me sorprende. Un proyecto que me sorprendió es uno de Tatiana Cuoco, que dio unos cuantos workshop en Berlín, República Checa. Hizo un viaje por ahí e hizo varios workshops. Eran workshops que tenían que ver con teoría feminista, cuestiones de xenofeminismo o sea, feminismo y tecnología. Y, en un momento del workshop usaban AudioStellar y lo que usaban eran sonidos de distintas manifestaciones feministas o incluso charlas, etc. Y los ponían todos en el mapa de AudioStellar y lo escuchaban y armaban como una especie de jam usando AudioStellar. Pero también lo lo veían desde un lugar como de des-territorialización, decían, bueno estas manifestaciones y estas charlas y todo esto está territorializado, está en un lugar físico y el software me ayuda a hacer esta esta operación y analizarlo desde este lugar abstracto donde capaz manifestaciones totalmente alejadas de repente estaban juntas porque sonaba parecido. No sé, había un bombo, o ciertos gritos, etc. Era como que de repente hizo un cruce desde la filosofía, de la teoría a un software que yo no sé, nunca hubiese pensado por ahí la cosa. Entonces, nada, sí, sorprende la gente muchas veces lo que se les puede ocurrir.

Max: Ahí se re nota que juega un poco más un papel importante la idea del mapa, que quizás la mayoría de les musiques, quizás les atrae esta idea del mapa, pero no con tanta fuerza como en este proyecto, que es crucial me parece.

Garber: Por la idea del territorio, ¿no?

Max: Claro, tal cual tal cual.

Garber: Sí, igual con respecto al mapa nos dimos cuenta, por ejemplo, que, volviendo un poco a la idea de que si es un instrumento o no como que es algo audiovisual, la verdad. O sea, si ok haces música, pero está bueno ver como la música suena con los puntitos y como hay una secuencia que aunque no entiendas exactamente lo que está pasando, de repente te das cuenta de la secuencia, te das cuenta de las partículas. Y hay algo de lo que miras y lo que suena que cuando haces estas cosas se juntan, la experiencia se expande. Eso es algo que no imaginamos y que es así.

Lean: Justo estaba pensando en todo esto e iba a preguntarte, ¿Qué tan importante crees el desarrollo técnico o el conocimiento técnico de la herramienta para su uso? Para al fin y al cabo crear un evento artístico. Si es importante saberlo, si hay que estudiar o tenes que prepararte o la idea es poder no saber nada y utilizar igual el entorno y crear ¿Qué importancia le das más, al al costado técnico o al artístico? ¿Cómo juegan esas dos ramas? Como decías cuando vos arrancaste, en un punto están bien separadas. Una cosa es estudiar computación en la UBA y otra cosa es hacer arte con esa computación. No siempre se busca relacionarlo.

Garber: Sí, está buena esa pregunta. Desde el comienzo la idea era acercar el Machine Learning a los músicos y al público en general. Acercar ciertas nociones, lo usamos muchísimo en la educación nosotros. Por ejemplo, en clases, en Artes Electrónicas soy docente y puedo dar clases donde uso AudioStellar como para explicar cuestiones que si no, tengo que explicar en un pizarrón y es una clase de matemáticas de repente. Bueno no, de repente lo puedo mostrar desde un lugar mucho más práctico. Entonces vos puedes acercar algo de ese conocimiento a quien le interese. Hemos dado charlas para público general, por ejemplo, en Uruguay dimos una charla, en el MALBA, donde acercamos un par de ideas, de mini ideitas de las técnicas que usamos y de lo que es Machine Learning usando el software. La educación siempre fue lo principal, y además, la idea de que no necesitas saber nada para usarlo. O sea, si no querés saber nada es un programa que abrí y suena. Hacer un mapa nuevo y hacerlo pasar por todas estas cadenas de procesamiento es apretar file New, elegir la carpeta y darle go. O sea, no, no necesitas absolutamente nada para usarlo. De hecho, lo que más me motiva a mí es cuando lo usa gente sin nada de conocimiento técnico porque de repente te hace preguntas impensadas que para vos era obvio, y de repente no es tan obvio y gente con otras cabezas lo ve distinto y yo me tengo que replantear.

Lean: Si, ese feedback que se va generando.

Garber: Sí, uno de los objetivos principales del proyecto ahora es esto, tener más feedback de músicos que no sean técnicos. Pero bueno, obviamente súper bienvenidos también aquellos músicos computacionales y técnicos que también quieran inventar algo a partir del software. Digamos, los dos son bienvenidos y los dos pueden usar y realmente no hace falta saber nada muy en particular. O sea, si no sos músico ni nada, bueno, si, te puedes divertir un rato, pero le vas a sacar más jugo sabiendo lo que es una reverb, lo que es un paneo, no sé, ese tipo de cosas. Pero no, está realmente apuntado para distintos niveles. Siempre fue el objetivo y eso no va a cambiar.

Lean: Bueno, sí, evidentemente hay como una línea que rodea a AudioStellar, Open Source, o sea, libre uso, que no esté apuntado a gente que quizás ni sepan ni de música ni de desarrollo técnico computacional. Me parece que también engloba un poco el concepto como todas esas cosas a las que apunta.

Garber: Sí, que sea muy abierto, ¿no? Bueno, y la página tiene un botón gigante "download", cliqueas, dice Linux Mac Windows descargar, abrir. Más sencillo que esto no se, casi lo más fácil que podría llegar a ser porque está pensado que sea como algo que lo puedas usar cerrado lo puede usar abierto, como vos quieras. Hay un poco más de data de

esto en el paper. Lo comparamos con otros proyectos en particular el Wekinator, por ejemplo, de Rebecca Fiebrink. Ella tiene un paper que es "Cómo enseñarle Machine learning a artistas" ¿Que hay que enseñarles? Y hace un programa de qué cosas son importantes enseñar para que los artistas puedan entender e imaginar cosas de Machine Learning. De alguna manera, con el paper queríamos sumarnos a esa discusión.

Max: Se re nota por lo que estamos hablando de que juega un rol re importante la idea de la de transdisciplina o la combinación de de ramas. Y sobre todo habiendonos contado tu background, hablando un poco de que empezaste con ciencia de la computación, despues te fuiste al arte y como se ve eso impregnado en el proyecto. Por un lado lo puedo usar a alguien que está súper sumergido en el mundo técnico, pero por el otro lado, lo puede usar alguien que sabe usar más o menos la compu y los programas pero no sabe programar. Se re nota eso plasmado en el proyecto.

Garber: Sí, y cuanto más mejor. Y también por ejemplo, que te conté de Tatiana Cuoco, ella es artista de vídeo. Pero bueno, ella también es docente y da cosas que se relacionan con teoría, con filosofía que yo puedo llegar a leer, pero no manejo. Y cuanto más disciplina se suma, mejor. Cosas que todavía no se han sumado que siento que van a aparecer. No sé, cuestiones de musicología, cuestiones incluso de medio ambiente. Por ejemplo, lo que tiene que ver con bioacústica me súper interesa. Que tienen que ver con agrupar, por ejemplo, el sonido de aves o de chanchos, no sé, lo que sea. Y quizás distintos gemidos o distintos sonidos que generan tienen utilidad. Así que nada, la verdad que se puede cruzar desde muchos lados, ojalá suceda.

Lean: No sé si vos Lean querías decir algo más relacionado a AudioStellar que quizás no hayamos preguntado o charlado.

Garber: No, quizás invitarlos el mes que viene, este mes de Octubre, a fin de este mes voy a presentar junto a Julio Nusdeo una instalación en el Centro de Arte Sonoro, en la casa del bicentenario. Estoy en una residencia ahora este mes. Y vamos a presentar una instalación de AudioStellar si se quieren acercar ahí podemos ir charlando y ver algun experimentito. Esto va a ser más como una intervención en el patio, en 8 canales. Estamos haciendo sistemas sonoros que evolucionan usando AudioStellar. Si se quieren sumar vengan. Hay que sacar entrada lo único.

Max: La última pregunta se va un poco del tema de AudioStellar, pero tiene que ver con nuestro tema de investigación, que es el arte algorítmico. Te queremos preguntar, ¿Qué te imaginas que se avecina en este mundo del arte algorítmico? Hoy por hoy estamos muy en los inicios, si se quiere o en el desarrollo de todo lo que tiene que ver con redes neuronales y el Machine Learning, que está como muy en boga. Pero nos preguntamos si vos te imaginas algo a continuación de eso, o no sé, si se te imaginas algo.

Garber: No me hago tantas preguntas sobre el futuro, soy como más de interesarme en el presente. Sí están habiendo muchísimas aplicaciones y algoritmos de Machine Learning aplicados al arte. Es muy difícil, o sea, yo cuando pienso en arte algorítmico o pienso en arte visual algorítmico, siempre pienso en arte generativo, en patrones que uno programa. Y con Machine Learning, yo hice varias experimentaciones y veo también lo que hacen otros artistas. Y cambia mucho. Hasta ponerlo en el mismo nivel es casi como que, si, los dos

arte hecho con la compu. Pero bueno, en uno pones ciertas reglas y en el otro, las reglas son inducidas a partir de datos. Lo que sí está pasando es que, si bien se está avanzando mucho en este sentido, todavía no hay muchas herramientas para artistas. Hay cada vez más. No hay muchas y las que hay muchas veces son “Bueno, clicar este botón, crear” Y puedes elegir dos cositas o tienes unas perillitas. En nuestro proyecto de investigación un poquito lo trabajamos y en AudioStellar lo trabajamos bastante también. ¿Cómo eso puede llegar a ser algo expresivo? En el sentido de ¿cómo se va poder interfacear con eso de manera expresiva? Me parece que eso está bloqueando un poco. O lo que veo en el futuro quizás es que si van a aparecer más cosas, que permitan un manejo más plástico de las herramientas. Hoy es muy un notebook. Control enter, control enter, apretar el botoncito de play. “Uh, no anduvo. Tardó 8 horas. A ver qué me hizo después de 8 horas. Ah, esta imagen” A mí me parece un embole. Yo lo estoy haciendo igual, pero no sé, necesitamos más cuerpo. Esa es una de mis cruzadas un toque. Por eso también la música, o sea, cuando uno toca, se mueve y hay como algo muy analógico sucediendo. Hay algunos experimentos, todavía no hay tantos, pero bueno, eso es capaz mi respuesta. Espero que vengan interfaces humano-máquina más expresivas para interactuar con estos algoritmos. Bueno, que se puedan generar cosas nuevas.

Max: También pienso que más allá de lo del cuerpo, que está buenísimo también, un poco de vuelta con la idea esta del nicho, siento que como decís, quizás estas nuevas tecnologías todavía están como muy cerradas en el nicho y que sólo quizás alguien que es más o menos se da maña con con la programación o con estas herramientas puede llegar a hacer algo maleable. Pero que aún así también es un garrón porque es como decís, es con el mouse y el teclado, darle enter a líneas de código y ya. No hay posibilidades de abrir eso y de que sea permeable y más abierto a todo el mundo.

Garber: Me gustaría verlo, sí. Yo creo que con esto de deliberar herramientas fáciles de usar, va a terminar cayendo en manos de gente con otras mentalidades que haga cosas muy originales. Eso es lo que creo que puede pasar. Que se use en lugares insospechados gracias a estos cerebros distintos. Pero bueno, vamos a ver qué pasa. Que se yo, tampoco está mal programar y hacer cosas.

Lean: Estaremos entonces en la muestra. ¿Cuándo es? ¿Dijiste en octubre?

Garber: Sí, es el 28, el jueves 28 de Octubre. Yo tengo un Instagram, no lo estaba usando, pero ahora lo empecé a usar “@leandrogarber”. Ahí voy a hacer todo lo posible por subir la data. Es el 28 de octubre, sino también por Radio Caso que es quien lo organiza. También en este momento está en Tecnópolis dentro del Museo Muntref Arte y Ciencia hay otra instalación que hicimos en conjunto con Intercambios Transorganicos. También es de AudioStellar pero vinculado a la voz. En este caso también es una instalación sonora capaz un poquito más didáctica porque estaba en tecnópolis y está apuntado a un público, quizás distinto, con otro bagaje. Pero bueno, me parece que las dos cosas vale la pena ver, así que si quieren sumar alguna de esas están invitados.