

## ESTRATÉGIAS PARA O CONTROLE GESTUAL DE SONS ELETROACÚSTICOS EM TEMPO REAL<sup>1</sup>

DANILO SILVA AGUIAR<sup>2</sup>  
DANIEL LUÍS BARREIRO<sup>3</sup>

### Resumo

O artigo apresenta resultados obtidos na pesquisa de Iniciação Científica realizada através do programa PIBIC/CNPq/UFU. São abordados dois aplicativos desenvolvidos no ambiente de programação *Pure Data* (Pd), elaborados para obter formas de controle gestual de sons eletroacústicos em tempo-real utilizando o controlador de videogame *Wii mote* (da Nintendo). O primeiro aplicativo teve como objetivo criar um Theremin virtual para controlar a frequência e a amplitude dos sons através do movimento de Led's infravermelhos captados pelo *Wii mote*. A segunda implementação trata-se de uma ferramenta de controle de reprodução de amostras sonoras ativada pelo acelerômetro embutido no controlador da Nintendo. Aplicativos como estes aliados ao controlador *Wii mote* vem se tornando excelentes formas de criar instrumentos musicais digitais e explorar aspectos criativos propiciados pelas novas tecnologias no contexto musical, instigando os músicos a pesquisarem mais sobre este assunto.

**Palavras-chave:** música eletroacústica; *Wii mote*; interface de controle gestual; síntese e processamento sonoro em tempo-real; mapeamento de parâmetros.

### Abstract

The article presents the results obtained in the Scientific Initiation Research carried out within the PIBIC/CNPq/UFU program. It discusses two applications that were programmed in *Pure Data* (Pd) to gain gestural control of electroacoustic sounds in real time using the *Wii mote* videogame controller (by Nintendo). The first application aimed at the creation of a virtual Theremin to control the frequency and the amplitude of sounds by the movement of infrared LED's captured by the *Wii mote*. The second implementation is a tool to control the playback of sound samples activated by the accelerometer embedded in the Nintendo controller. Applications such as these combined with the *Wii mote* controller have become excellent ways to create digital musical instruments and explore the creative aspects enabled by new technologies in the musical context, encouraging musicians to carry out further research on this subject.

**Keywords:** electroacoustic music; *Wii mote*; gestural control interface; real-time synthesis and sound processing techniques; mapping of sonic parameters.

---

<sup>1</sup> Agradecemos ao apoio do Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) pela bolsa de Iniciação Científica concedida ao primeiro autor através do PIBIC/CNPq/UFU e à Fundação de Amparo à Pesquisa do Estado de Minas Gerais (FAPEMIG) pelo auxílio à pesquisa concedido ao segundo autor através do Edital 021/2008.

<sup>2</sup> Aluno do Bacharelado em Música (Violão) da Universidade Federal de Uberlândia (UFU), Instituto de Artes (IARTE), Curso de Música, Av. João Naves de Ávila, 2121 – Campus Santa Mônica – Bloco 1V – CEP: 38400-902 – Uberlândia (MG). E-mail: daniloaguiaarmusic@gmail.com

<sup>3</sup> Orientador. Universidade Federal de Uberlândia (UFU), Instituto de Artes (IARTE), Curso de Música, Av. João Naves de Ávila, 2121 – Campus Santa Mônica – Bloco 1V – CEP: 38400-902 – Uberlândia (MG). E-mail: dlbarreiro@demac.ufu.br

## 1. INTRODUÇÃO

A música eletroacústica é uma das vertentes musicais que, desde suas origens em finais da década de 1940, mergulhou profundamente nas potencialidades trazidas pelas novas tecnologias. Ao escutar obras de compositores como Pierre Schaeffer, Karlheinz Stockhausen, Denis Smalley, Bernard Parmegiani, Francis Dhomont, Phillippe Manoury, entre outros, verifica-se ser fundamental estudar as questões motrizes da música do século XX e suas relações com as novas tecnologias. Isso fica evidente no âmbito da música eletroacústica, seja ela da vertente acusmática (com sons fixados em suporte), mista (com instrumentos musicais e sons fixados em suporte) ou derivada da eletrônica em tempo real (*live electronics*) – a qual inclui instrumentos musicais e “sons criados ou manipulados eletronicamente durante a performance” (ROCHA, 2010, p.1233).

Aprofundando a citação de Rocha, pode-se dizer que a música eletroacústica em tempo real (*Live Electronic Music ou live electronics*) é aquela na qual os sons provenientes de instrumentos musicais ou de síntese e processamento sonoro realizados no computador são criados, modificados e operados por meios eletrônicos no instante em que a música está sendo executada.

A ideia de sons manipulados em tempo real já era algo explorado pelo compositor John Cage que, em 1939, no estúdio radiofônico da Cornish School, compunha a obra *Imaginary Landscape N° 1* (1939), feita para piano preparado, pratos e dois toca-discos que eram manipulados pelos intérpretes em diferentes velocidades (MENEZES, 1996, p.253). Com o passar do tempo, a aceleração dos avanços tecnológicos no século XX possibilitou retomar ideias como a de Cage e desenvolvê-las de maneiras ainda mais profundas com o uso cada vez mais presente dos computadores. Isso inclui a possibilidade de captar os sons de instrumentos musicais no instante da performance, depreender dados da análise do sinal de áudio com o uso de computadores, processar os sons instrumentais em tempo-real ou ainda desencadear processos de síntese sonora em correlação com parâmetros sonoros identificados na performance. Com a emergência dos computadores pessoais de baixo custo e com alta qualidade de processamento, tornou-se cada vez mais frequente a utilização de *laptops* para trabalhar com sons eletroacústicos em tempo real. A partir desta utilização, surgem também novos instrumentos musicais com sons gerados por computadores, que realizam uma gama variada de métodos de síntese sonora por meio de *software* (WANDERLEY, 2006, p.1), utilizando diferentes tipos de interfaces e sensores como dispositivos de controle dos

processos computacionais. Assim, novos instrumentos, novas ferramentas e interfaces começaram a surgir e a se desenvolver.

Uma das fontes para a concepção de novos instrumentos musicais digitais é a crescente indústria de videogames, que, nos últimos anos, vem desenvolvendo interfaces de controle gestual em ambientes digitais. Um exemplo é o controlador *Wiimote* (da Nintendo), que por meio de conexão *Bluetooth*, transmite protocolo de comunicação de porta serial sem-fio. Esse controle possui um acelerômetro que proporciona a identificação de deslocamentos e rotações nos três eixos espaciais, além de diversos botões e uma câmera que capta Led's emissores de sinal luminoso infravermelho. Um segundo acessório pode ser acoplado ao *Wiimote* – o *Nunchuk* – que contém um controle bidimensional (*joystick*), mais um acelerômetro e dois botões (FORNARI; MANZOLLI, 2010, p. 792).

Pesquisadores no Brasil e em outros países vêm investigando possibilidades de controle sonoro com o *Wiimote*. A proposta da pesquisa de Iniciação Científica realizada surgiu do contato com os trabalhos que vêm sendo desenvolvidos por pesquisadores tais como Jônatas Manzolli e José Fornari, do Núcleo Interdisciplinar de Comunicação Sonora (NICS), da Unicamp.

O objetivo desta pesquisa foi buscar soluções criativas para o uso do *Wiimote* como controlador de sons eletroacústicos em tempo real, explorando principalmente suas potencialidades de controle gestual. Os aplicativos desenvolvidos para o controle de sons eletracústicos em tempo real com o *Wiimote* são abordados nos tópicos a seguir.

## 2. MATERIAL E MÉTODOS

Com base nos fundamentos de experiências desenvolvidas com o *Wiimote* principalmente por pesquisadores do NICS, foram criados dois aplicativos (*patches*) em ambiente de programação *Pure Data* (Pd). Este *software* de programação gráfica para áudio e vídeo permite desenvolver análise, síntese e processamento de áudio em tempo real (ver <http://puredata.info>). O primeiro *patch* teve como objetivo desenvolver um Theremin virtual, inicialmente no sistema operacional *Windows*. A segunda experiência abordou uma ferramenta de controle de reprodução de amostras, desenvolvida no sistema *Mac OS*.

## 2.1. Wii Theremin

Criado em 1920 por Leon Theremin, o instrumento eletrônico Theremin buscava formas de acionamento instrumental diferentes das tradicionais. O instrumento, executado com o movimento das mãos próximas a duas antenas, introduziu o primeiro controle gestual dos sons no ar, sem a necessidade do contato físico com a fonte sonora ou com um dispositivo de acionamento do mecanismo de geração do som. O movimento de uma das mãos próxima a uma das antenas controla a altura (frequência) do som, enquanto que a outra controla a intensidade (amplitude) do som.

Partindo desta concepção e como preâmbulo à implementação de um Theremin virtual utilizando o *Wiiote*, elaboramos um primeiro experimento com um *patch* em Pure Data no qual a altura e a intensidade dos sons do Theremin virtual eram controladas apenas com as teclas do computador. Num segundo experimento foi elaborado um *patch* para usar o controlador *Wiiote*. Nesse caso, a câmera infravermelha do *Wiiote* foi utilizada para identificar dois *Led's* infravermelhos cujas posições projetadas no plano cartesiano foram mapeadas para controlar a frequência e a amplitude dos sons produzidos pelo instrumento virtual.

Como dito anteriormente, o *Wiiote* possui uma câmera infravermelha que capta feixes de sinal luminoso infravermelho, nesse caso produzido pelos *Led's*. Ao elaborar o Theremin virtual, foi preciso confeccionar um dispositivo de fácil manejo para acoplar os *Led's*. Assim, foi feita uma espécie de caneta infravermelha (Caneta IR) (Figura 1). Esta caneta – desenvolvida e divulgada pelo pesquisador e engenheiro da computação Johnny Chung Lee (ver <http://johnnylee.net/>) – possibilita que o *Wiiote* (conectado ao computador via Bluetooth) capte os movimentos dos *Led's* através de sua câmera infravermelha, passando os dados para programas que traduzem esses movimentos para algum parâmetro de controle no computador.

Nesta implementação, utilizamos um *script* de programação no *software GlovePie* para detectar os dados do *Wiiote* via *Bluetooth* e transmiti-los para o *Pure Data (Pd)*. No *Pd*, os dados são recebidos através do objeto *DumpOSC* e enviados para um *patch* de calibragem para que o aplicativo principal consiga captar os movimentos dos dois feixes infravermelhos com maior precisão. A calibragem permite delimitar a área de ação dos *Led's* em relação ao plano cartesiano.



Figura 1. Estrutura da Caneta Infravermelha (Caneta IR)

Posteriormente, com os dados devidamente calibrados, estes são transmitidos para o aplicativo principal (Figura 2), onde são recebidos pelos objetos “*r led1\_x*” e “*r led1\_y*” e encaminhados para controlarem a frequência e a amplitude dos sons produzidos a partir das coordenadas espaciais “X” e “Y”, respectivamente.

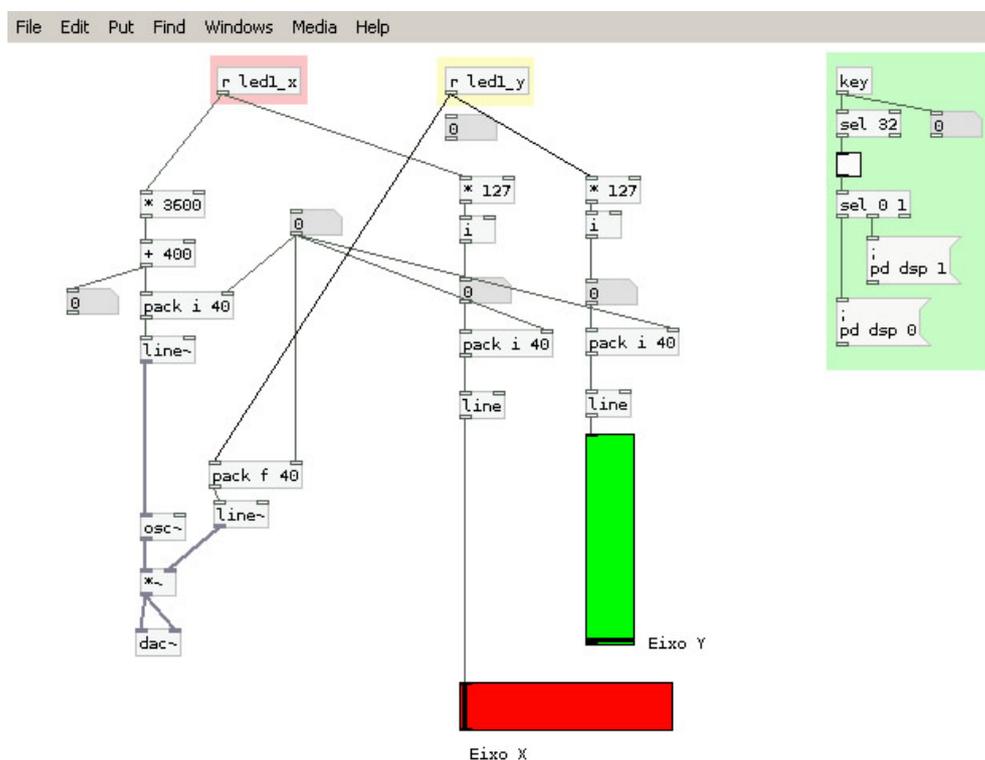


Figura 2. Patch Theremin Virtual

Com a questão da detecção dos movimentos dos Led's resolvida, o trabalho concentrou-se em expandir alguns aspectos do instrumento virtual. Desenvolvemos, por exemplo, um Theremin virtual que produzisse apenas sons distanciados por um intervalo musical de um tom, ou seja, sons enquadrados na escala de tons inteiros. Os valores que chegam para o eixo "X" são mandados para o *subpatch* "pd intervalos". Nele, os dados são filtrados pelo objeto "moses" e endereçados para os números MIDI correspondentes às notas que se enquadram na escala desejada. Nesse caso específico, geram-se valores MIDI relacionados à escala de tons inteiros partindo da nota ré (número 50), a qual é seguida por mi (52), fá# (54), sol# (56), lá# (58), e assim por diante. Logo depois, o número MIDI é transformado em frequência de áudio pelo objeto "mtof" e enviado para um oscilador que gera ondas senoidais (Figura 3).

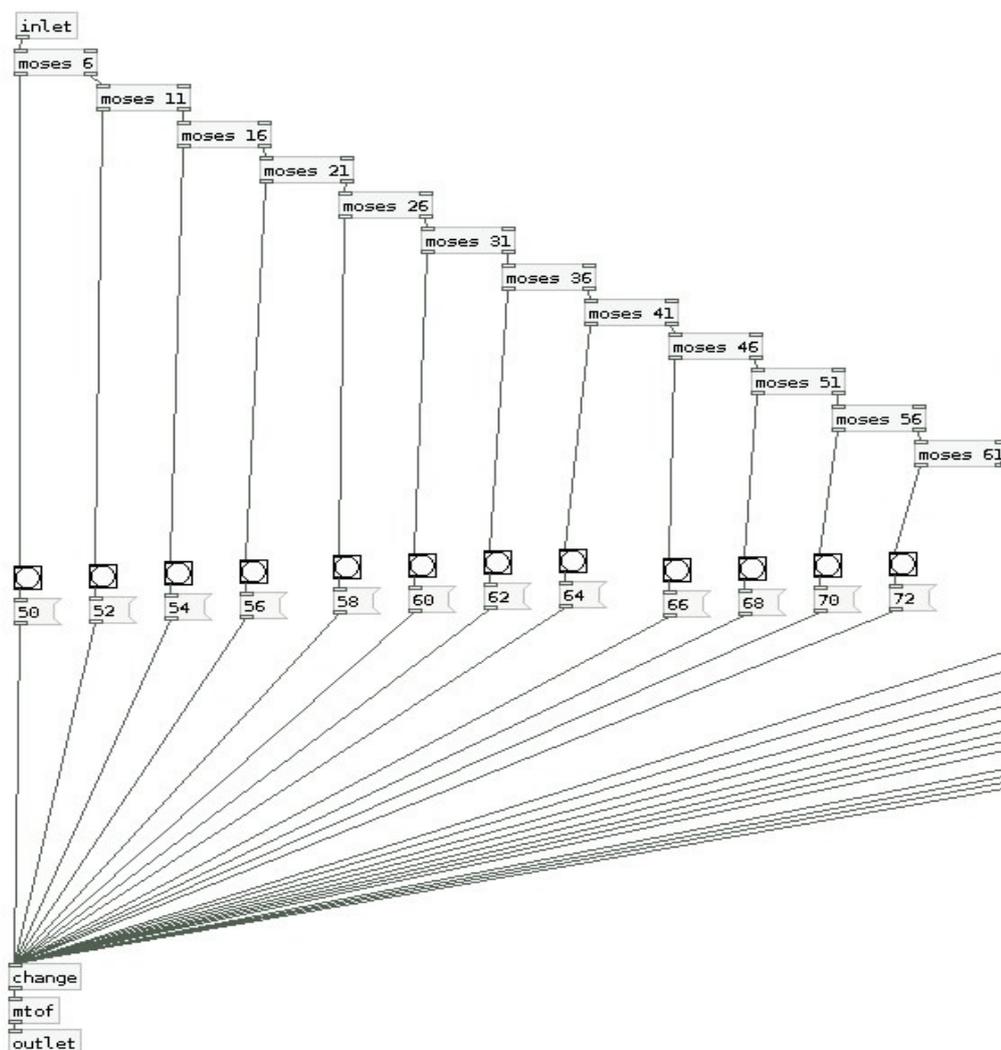


Figura 3. *Subpatch* Theremin Tons – Inteiros

## 2.2. *Wii*mote como controle de reprodução de amostras

Fornari e Manzolli (2010) desenvolveram *patches* no ambiente de programação Pd que captam gestos de impulso das mãos realizados com o *Wii*mote para disparar e controlar sons gerados por síntese, como, por exemplo, o som de um berimbau gerado com o modelo Karplus-Strong. Desenvolveram um *patch* utilizando o acelerômetro do *Wii*mote para identificar gestos similares aos de golpe de uma baqueta sobre a pele de um tambor. Com base na Terceira Lei de Newton (o Princípio da Ação e Reação), a força do movimento percussivo da baqueta do intérprete sobre a pele do tambor cria uma força contrária, a força de reação. Diante desta resistência da pele do tambor, o movimento do braço do músico regressa logo após atingir um ponto máximo da ação realizada.

Nos *patches* desenvolvidos por Fornari e Manzolli (2010)<sup>4</sup> a questão foi justamente encontrar uma forma de traduzir este exemplo da baqueta na pele do tambor para os parâmetros do controlador *Wii*mote. Os pesquisadores pensaram numa analogia entre o movimento do controlador e o movimento feito por uma baqueta em um instrumento percussivo para identificar o ponto de ataque. Portanto, em qualquer momento que o músico, ou mesmo um usuário leigo, realizar um ataque percussivo com o *Wii*mote, o *patch* irá captar os dados do acelerômetro e detectará a mudança na direção do movimento em um dos três eixos espaciais disparando um som sintetizado. No aplicativo dos pesquisadores Fornari e Manzolli, esta mudança de movimento, que é a identificação do ponto máximo da trajetória numa mesma direção, é reconhecida através da comparação entre os três últimos valores recebidos do *Wii*mote em um dos três eixos espaciais. Considerando  $t_0$  como a posição atual em um determinado eixo espacial e  $t-1$  e  $t-2$ , respectivamente, como os dois valores anteriores, o disparo do som sintetizado irá ocorrer quando  $t_0 < t-1 > t-2$  (Figura 4). Nesse caso, o momento  $t_0$  indica a mudança na direção do movimento, o que aciona o som sintetizado.

---

<sup>4</sup> Ver *Wii*mos em <<http://sites.google.com/site/tutifornari/academico/pd-patches>>

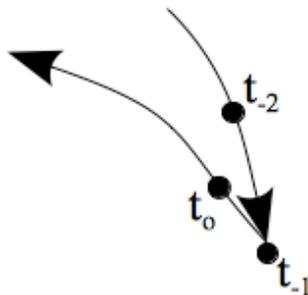


Figura 4. Representação do movimento do *Wiimote* num dos eixos espaciais

Com base nestas implementações de Fornari e Manzolli, foi desenvolvido o segundo aplicativo da pesquisa de Iniciação Científica utilizando o *Wiimote* como disparador de amostras sonoras. Verifica-se, assim, que ao invés de utilizar os dados do *Wiimote* para disparar um processo de síntese (como em Fornari e Manzolli, 2010), na nossa implementação os dados geram a reprodução de uma amostra de áudio pré-gravada. Para utilizar esse *patch*, os dados do *Wiimote* são endereçados ao Pd pelo software *Osculator*<sup>5</sup>, que é semelhante ao *GlovePie*, porém com a diferença de não exigir a elaboração de nenhum tipo de *script* de programação, já que o próprio programa identifica automaticamente os parâmetros do controlador e envia os dados segundo o endereçamento especificado pelo usuário, dados estes que são captados por outro *software* (no caso, o Pd).

Para conseguir melhores resultados musicais, ampliando a paleta sonora gerada pelo *patch*, foram desenvolvidos alguns recursos interessantes como a possibilidade de carregar seis amostras sonoras (*samples*) diferentes, cada uma das quais pode ser selecionada através dos botões de esquerda e direita do *Wiimote*. A velocidade do movimento do *Wiimote* no momento do disparo das amostras determina a amplitude do som reproduzido. Assim, quanto maior a força do ataque, maior será sua amplitude (resultando em um som com maior intensidade), e quanto menor a força de ataque, menor a amplitude da amostra (resultando em um som com menor intensidade).

Outro recurso interessante é a implementação da transposição das amostras (gerando sons mais graves ou mais agudos), que é feita através de diferentes velocidades de leitura das amostras. Se uma amostra sonora dura cinco segundos, por exemplo, e é lida pelo computador em dez segundos, isso significa que a sua leitura transcorre com metade da velocidade original, o que faz com que a frequência resultante seja também a metade da original,

<sup>5</sup> *Osculator*: <<http://www.osculator.net/>>

gerando, portanto, um som mais grave. O controle de transposição é executado por rotação do pulso (rotação no eixo “X” do *Wiimote*) associado ao uso do botão “B”, da seguinte forma: girando o *Wiimote* para a esquerda, diminuem os valores que controlarão a velocidade de leitura das amostras (resultando em transposições mais graves); girando para direita, aumentam os valores da velocidade de leitura das amostras (resultando em transposições mais agudas); acionando o botão “B” o valor da transposição a ser utilizado é registrado e, em qualquer momento, através do movimento de ataque, o som é disparado com base nesse valor (Figura 5). Também é possível realizar este processo de transposição com os botões “Minus” e “Plus” do *Wiimote*, porém sem a necessidade de registrar os valores da transposição com o botão “B”, permanecendo essa opção mais como uma função para ajustes mínimos no âmbito da transposição.

Existe também o recurso de aplicar modulação em anel (*Ring Modulation*) ao som disparado. A modulação em anel foi uma das primeiras técnicas usadas na manipulação de timbres na era analógica da música eletrônica. Aqui a modulação é acionada apertando o botão “A” no *Wiimote*. O botão “1”, por sua vez, aumenta a frequência de modulação e o botão “2” a diminui.

Como forma de obter maiores recursos musicais, utilizamos o acessório *Nunchuk*, que contém dois botões e um *joystick* bidimensional. Acoplado ao *Wiimote*, o *Nunchuk* traz maiores possibilidades de manipulação no âmbito sonoro, como foi o caso nessa pesquisa, através da implementação de um dispositivo de espacialização sonora. Esse dispositivo permite ao usuário distribuir o som panoramicamente entre os altofalantes da seguinte maneira: acionando o botão “C” do *Nunchuk*, a amostra de áudio é gradativamente direcionada para o altofalante do lado esquerdo e, acionando o botão “Z”, o som é gradativamente direcionado para o altofalante do lado direito (Figura 5).

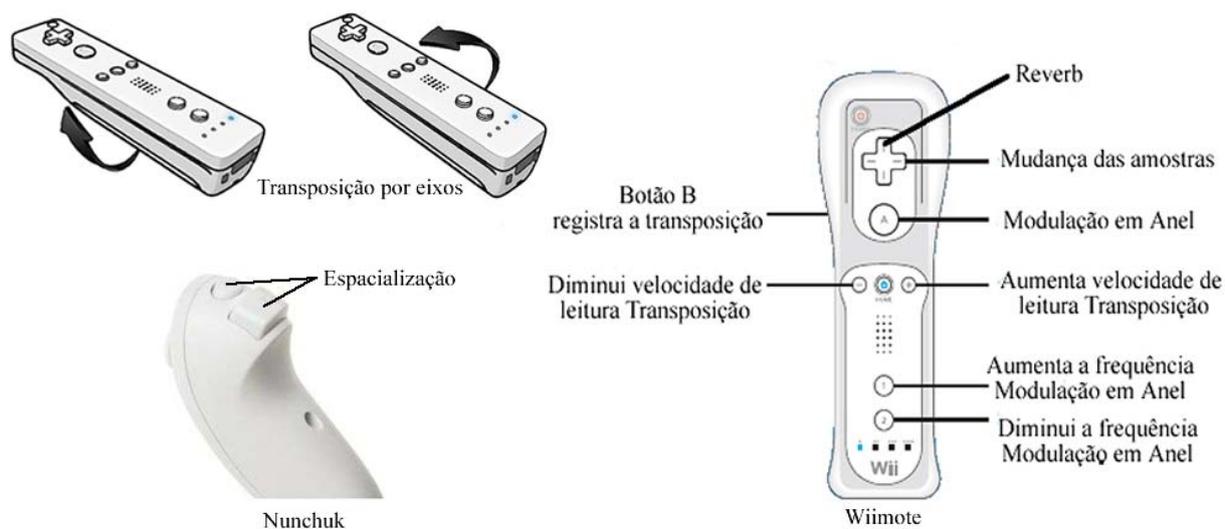


Figura 5. Representação das implementações no *Wiimote* e o no *Nunchuk*

### 3. RESULTADOS e DISCUSSÃO:

O controlador *Wiimote*, associado ao ambiente de programação *Pure Data* tem se mostrado uma boa opção para a implementação de instrumentos digitais. Com a ampla possibilidade de captação e controle gestual desse dispositivo e a crescente utilização desse tipo de tecnologia, devido ao seu baixo custo, o *Wiimote* vem sendo uma excelente forma de explorar aspectos criativos propiciados pelas novas tecnologias no contexto musical.

#### 3.1. Resultados alcançados com *Wii Theremin*

A experiência com o *Wii Theremin* rendeu resultados interessantes. A utilização de *Led's* infravermelhos acoplados às mãos através das canetas infravermelhas gerou um instrumento digital bastante similar ao verdadeiro instrumento de Leon Theremin. Além disto, o desenvolvimento norteou a liberdade de movimentação das mãos, resultando em um maior âmbito de manipulação dos parâmetros do instrumento, o qual o verdadeiro Theremin não viabiliza, devido à necessidade de proximidade das mãos às duas antenas para melhor precisão do controle de amplitude e frequência.

No entanto, algumas dificuldades rondaram este primeiro experimento. Os primeiros imprevistos apresentados no estudo do *software* *Pure Data* (Pd) foram certas ausências de

informações técnicas na documentação do programa. Enquanto o estudo esteve focado em tutoriais e aplicativos elaborados por outros pesquisadores, certas dúvidas exigiam que se recorresse à documentação e exemplos encontrados na biblioteca do programa para entendimento específico do que estava sendo feito. Porém, nem sempre constavam no *software* informações que fossem úteis para a realização e a compreensão do *patch*, o que exigia a busca de informações em outras fontes, como manuais (às vezes em outro idioma). Dessa forma, foram usados tutoriais de Johannes Kreidler<sup>6</sup> e Alexandre Porres<sup>7</sup> para maior entendimento do *software* Pd.

Após suprimir os obstáculos da documentação do ambiente de programação Pure Data (Pd), foi observado que os pesquisadores Fornari e Manzolli (2010) usavam um programa intermediário, chamado *Osculator* (para Macintosh), para estabelecer a comunicação entre o *Wiiote* e o Pd. O *Osculator* suporta o protocolo OSC (ver Wright et al., 2003), o que o torna capaz de ser utilizado com uma ampla variedade de *software*, incluindo o *Pure Data (Pd)* e o *Max/MSP*, podendo ainda estabelecer comunicação com diversos dispositivos MIDI. Durante os seis primeiros meses da pesquisa utilizamos o sistema Windows 7, o que nos obrigou a comunicar os dados do *Wiiote* com o Pd através de outro software, o *Glove Pie*. Nos meses de fevereiro a abril de 2011, revimos o uso do *software* *Glove Pie* e testamos outros programas que permitissem uma maior eficiência, como o *Osculator* (Macintosh) e um objeto do programa Pd para o sistema *Linux* chamado “*wiiote*”. Esse objeto, que é instalado no diretório de arquivos do *Pure Data*, realiza a conexão via *Bluetooth* sem o intermédio de outro *software*, como os mencionados acima. Como resultado dos testes foi escolhido o software *Osculator*, por ser mais estável e por usar o sistema Mac OS, o qual é o mais utilizado nos exemplos de *patches* que foram estudados.

Em relação à confecção da caneta infravermelha, houve também certos contratemplos. O aprendizado para confeccionar a caneta demandou tempo para encontrar os componentes certos, como os Led's infravermelhos, e também para o aprendizado de noções básicas de eletrônica para melhor construção do dispositivo. Foi observado que a caneta do pesquisador e engenheiro de computação Johnny Chung Lee foi implementada com uma pilha média de 1,5 V, que alimenta um Led de 1,5 V. Como esse tipo de Led é muito difícil de encontrar no Brasil, foi usada uma bateria de 3 V na parte traseira da caneta para alimentar um Led de 3 V. De qualquer forma, a caneta se mostrou como uma ótima ferramenta para produzir os feixes

---

<sup>6</sup> Johannes Kreidler: <<http://www.pd-tutorial.com/>>

<sup>7</sup> Alexandre Porres: <<http://sites.google.com/site/porres/pd>>

de luz infravermelha, pois ela possui um botão de acionamento (Liga/Desliga), o que facilita a usabilidade da ferramenta em contextos musicais.

Outra dificuldade encontrada diz respeito à interferência entre os dois Led's no *patch* "Theremin". Nesse experimento, os Led's mapeados aos eixos X (frequência) e Y (amplitude) eram confundidos pelo *patch* caso se aproximassem demais ou caso saíssem temporariamente do campo de captação da câmera infravermelha do *Wii mote*. Ora os Led's do eixo "X" se manifestavam como "Y" e vice-versa, causando certa confusão na performance. Os experimentos apontam para a possibilidade de minimizar o problema escolhendo um ambiente de pouca claridade, pois a luminosidade natural (que possui raios infravermelhos) interfere na captação do sinal luminoso dos Led's. Tomando conhecimento deste detalhe, o *patch* permite que a câmera do *Wii mote* capte somente os Led's, fazendo com que os parâmetros funcionem corretamente.

Entende-se que a experiência utilizando o controlador *Wii mote* como parte do instrumento digital Theremin foi compensada pelos resultados sonoros, principalmente pela implementação do *patch* "Theremin tons-inteiros" citado anteriormente, pois a aproximação com instrumento real, acrescida da liberdade em poder estender certos parâmetros, como mudança da frequência e amplitude através do software Pd, fazem com que o músico fique mais familiarizado com novos instrumentos e novas ideias musicais.

### **3.2. Resultados alcançados com o *Wii mote* como controle de reprodução de amostras**

O aplicativo para o controle de reprodução de amostras de áudio é o que conseguiu estabelecer melhores resultados musicais. O *Wii mote* como disparador de amostras se mostrou um instrumento de fácil utilização e com maiores recursos que o Theremin virtual, tal como o controle de intensidade (amplitude) que é acionado através da força de ataque (velocidade do movimento com o *Wii mote*). Os recursos de transposição das amostras, modulação em anel, reverberação, espacialização e variabilidade de escolha entre as amostras também são dignos de nota.

A grande quantidade de recursos implementados nessa experiência trouxe maiores dificuldades e exigiu uma maior quantidade de tempo dedicado ao estudo de tutorias e aplicativos de outros pesquisadores que pudessem ajudar na criação e desenvolvimento do *patch*.

A primeira dificuldade foi conseguir determinar uma forma para que o *patch* captasse os ataques que posteriormente iriam disparar as amostras. Como dito anteriormente, usamos a

estratégia de Fornari e Manzolli (2010), que é estabelecida através da comparação entre os três últimos valores recebidos do *Wiimote* em relação ao um dos três eixos espaciais. Com o aplicativo conseguindo captar os ataques, surgiu um segundo problema em relação ao excesso de sensibilidade do controle, resultando numa grande oscilação dos valores recebidos pelo Pd através do *Osculator*. Assim, qualquer movimento gerado pelo músico, por menor que fosse, fazia com que o *patch* o reconhecesse como um ataque e disparasse os sons sem a intenção do intérprete. Nas experiências realizadas, foi verificado que o uso do parâmetro “*pry/accel*” (que diz respeito à velocidade de movimentação do *Wiimote*, conforme a designação usada pelo software *Osculator*), gerou menos “falsos ataques” do que o uso dos parâmetros usados por Fornari e Manzolli (2010). Portanto, o que se identifica no *patch* desenvolvido nesta pesquisa não são posições do *Wiimote* num determinado eixo das coordenadas espaciais (como mostra a Figura 4), e sim uma súbita diminuição na velocidade de deslocamento do *Wiimote*.

Verificou-se nessa experiência a importância dos gestos, conforme atesta Traldi (2007, p.25) ao afirmar que “o gesto físico do intérprete passa a ser de enorme importância e é preciso que ele (o intérprete) tenha consciência das possibilidades gestuais e o que cada gesto irá desencadear em sua performance”.

Com a calibragem devidamente feita, foi preciso enfrentar outro problema, que seria disparar mais de uma amostra sem a interrupção da reprodução da amostra anterior. Foi criado o *subpatch* “*pd players*”, contendo 20 disparadores (*players*) de áudio independentes, que são acionados através do objeto *counter*, o qual permite o endereçamento de cada ataque para um determinado disparador. Assim, ao acionar amostras sucessivas, uma não interfere na outra, pois cada amostra sonora é reproduzida por um disparador diferente (Figura 6).

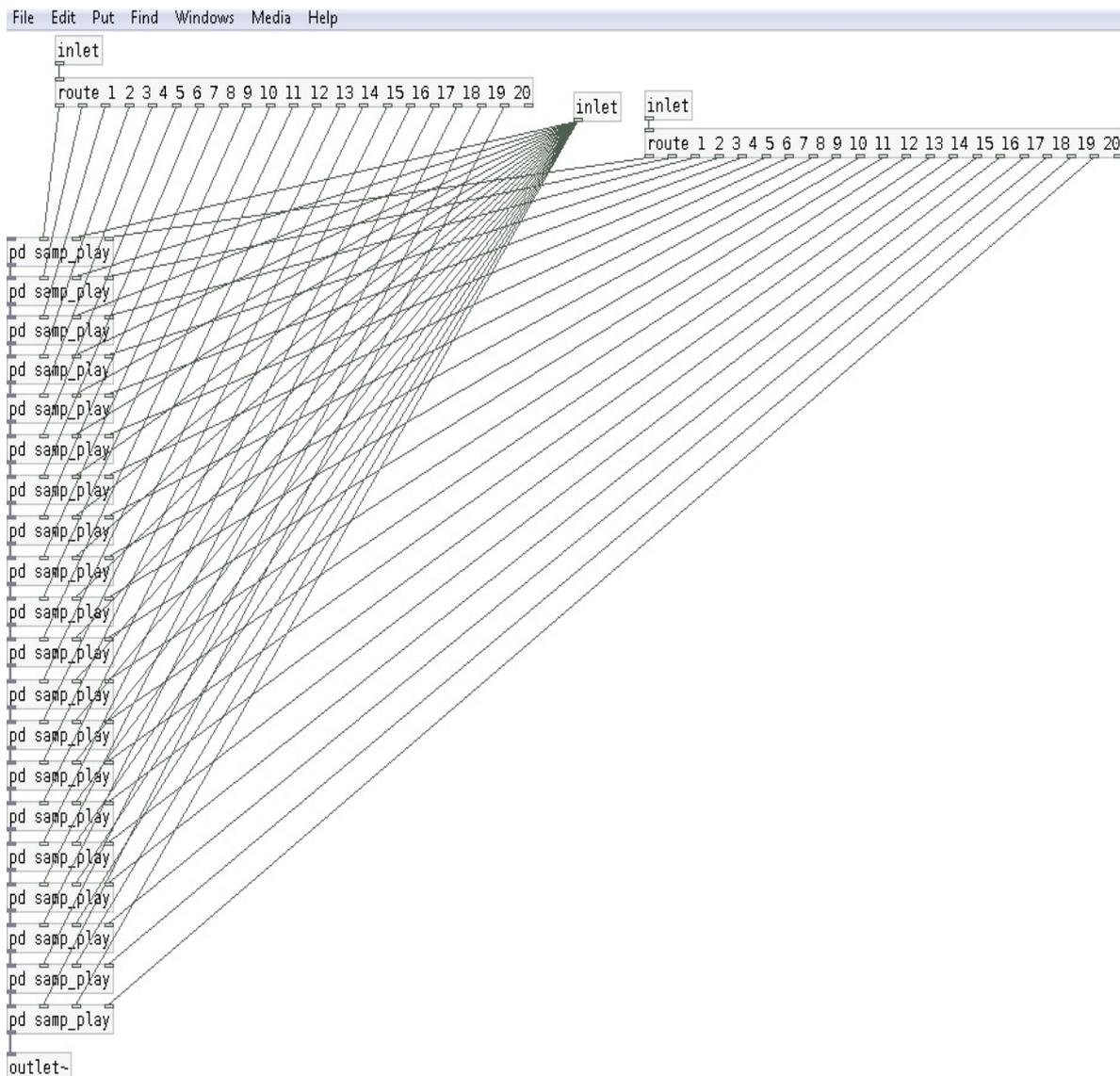


Figura 6. Exemplo dos 20 disparadores.

Conforme dito anteriormente, a amplitude para a reprodução das amostras é determinada pela velocidade de deslocamento do *Wiimote* (parâmetro “*pry/accel*”, segundo a nomenclatura usada pelo *Osculator*) no momento de disparo de uma amostra. Os dados de velocidade são enviados para os 20 *subpatches* disparadores de amostras contidos no *subpatch* “*pd players*”. Dentro de cada um dos 20 disparadores, esse parâmetro é escalonado para valores entre 0.5 e 0.749 e mapeado para controlar a fase de um oscilador. Soma-se 1 (um) aos valores gerado pelo oscilador (que inicialmente apresentam-se num âmbito entre -1 e 0), gerando, conseqüentemente, valores entre 0 (zero) e 1 (um) – que correspondem ao mínimo e máximo de amplitude possível para cada amostra disparada (Figura 7).

Para implementar o dispositivo de transposição foi usado o objeto *tabread4~* do Pd que permite variar a velocidade de leitura das amostras de áudio (Figura 7). Através do eixo

“X” de rotação do *Wii* é possível determinar a velocidade em que se pretende ler as amostras, conforme explicado anteriormente.

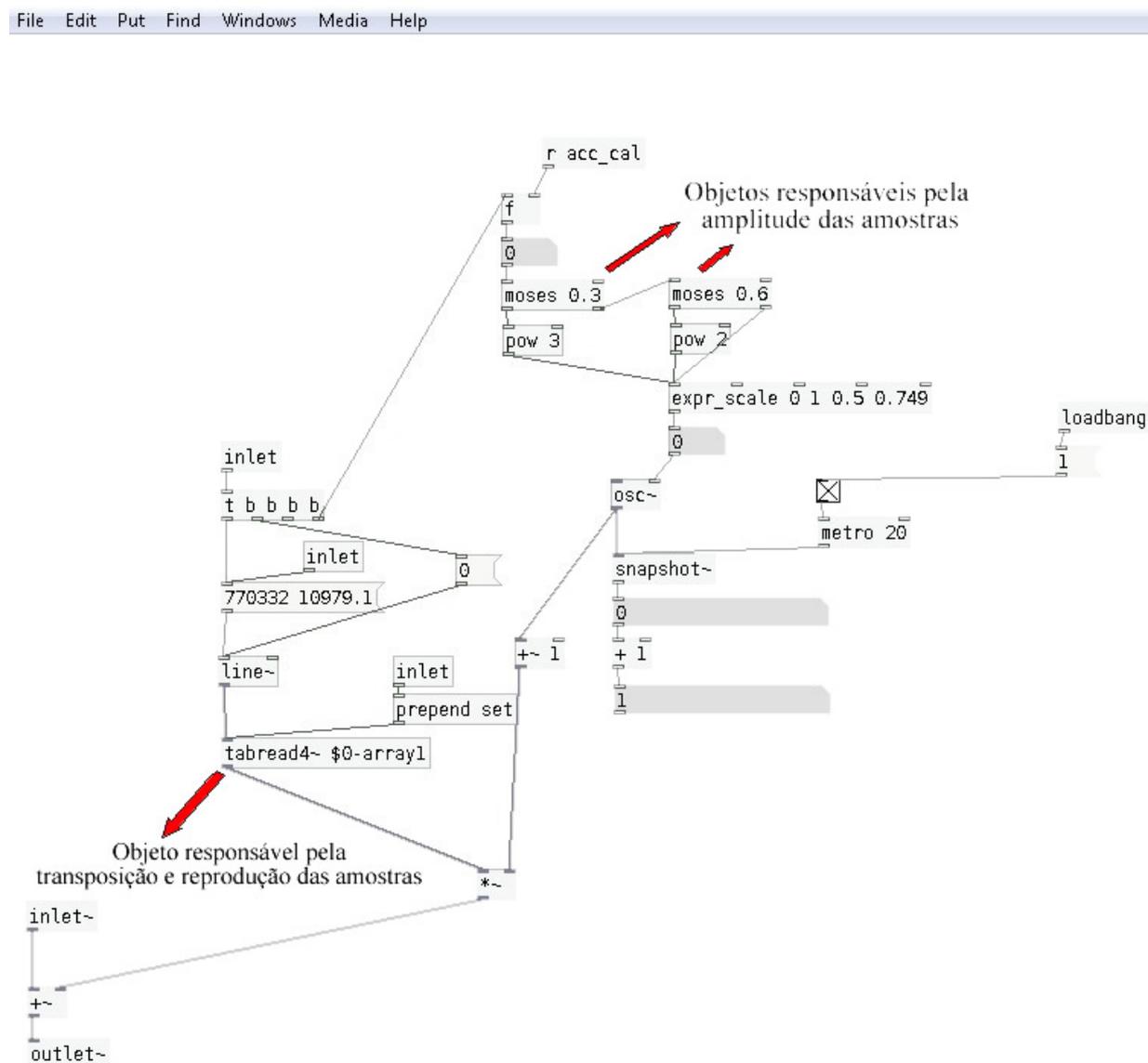


Figura 7. Implementação do Controle de Transposição e Amplitude das amostras

Para criar uma melhor ambiência nos sons, foi desenvolvido um dispositivo de reverberação e um sistema de espacialização. O dispositivo de reverberação provém de um *patch* disponível na biblioteca de distribuição do Pure Data (Pd). Portanto, esta implementação foi simplesmente copiada e colada no *patch*. O sistema de espacialização, por outro lado, foi fruto de algumas implementações adicionais feitas a partir de aplicativos do pesquisador Georg Holzmann<sup>8</sup>. O *patch* de Holzmann incluía possibilidades de espacialização

<sup>8</sup> Georg Holzmann: <<http://grh.mur.at/publications/sound-spatialization-pd>>

com trajetórias curvas. Em nossa implementação, no entanto, optou-se por um dispositivo mais simples com trajetória linear. Esse recurso, que é utilizado através do acionamento dos botões “C” e “Z” do controlador *Nunchuk*, permite dar a ilusão de movimentação do som no espaço entre dois altofalantes.

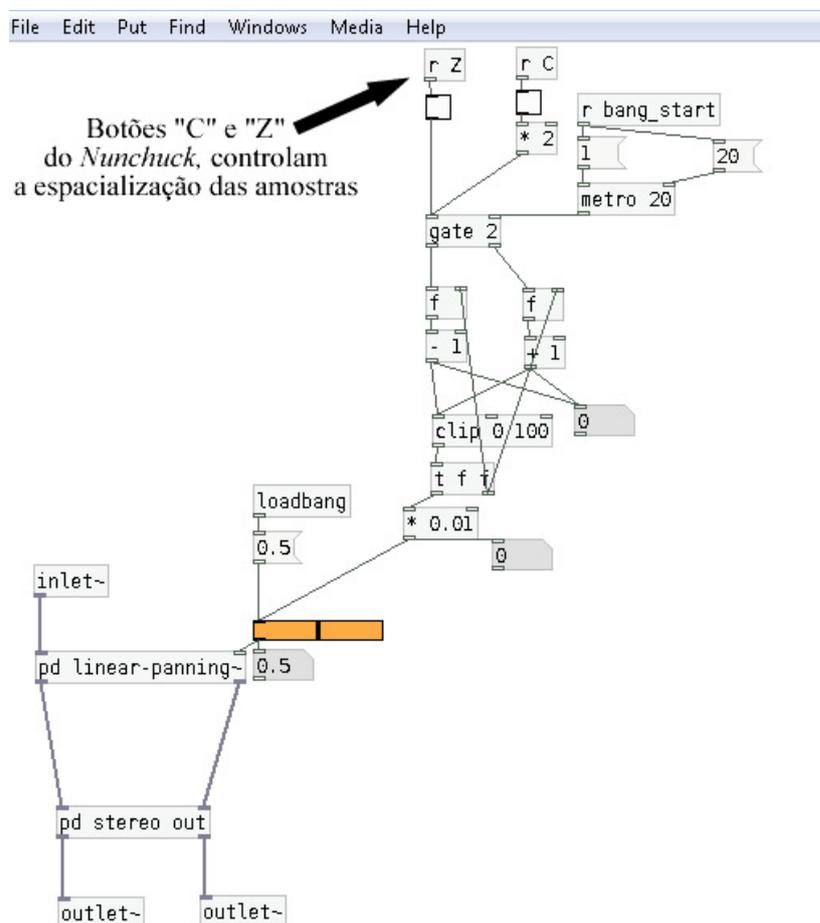


Figura 8. Aplicativo de espacialização das amostras

O aplicativo (*patch*) de reprodução de amostras pré-gravadas aqui relatado foi uma das implementações utilizadas em *[Wii]improviso*, uma improvisação livre com três *Wimotes* *abordada* em Traldi, Aguiar e Barreiro (2011). O *patch* possibilita que o intérprete escolha até seis amostras de áudio para poder manipulá-las através de transposição, controle de amplitude, modulação em anel e reverberação. Esse aplicativo viabilizou resultados interessantes e satisfatórios durante a improvisação. Uma das questões com a qual mais se tomou cuidado durante as sessões de improvisação foi a escolha das seis amostras a serem disparadas com o *Wimote*. Observou-se que essas amostras devem ser selecionadas cuidadosamente e testadas com todos os recursos disponíveis no *patch* antes da performance, pois a maior dificuldade na improvisação é ter controle sobre todos os parâmetros. Qualquer

som escolhido acidentalmente pode resultar em resultados indesejados – sons que soam como “clichês” num dado contexto, por exemplo. Para auxiliar o músico a identificar a escolha das amostras, foi criado um pequeno painel onde aparece o número correspondente à amostra selecionada como um recurso de *feedback* visual (Figura 9), o que contribuiu nas improvisações para o controle das sonoridades geradas.

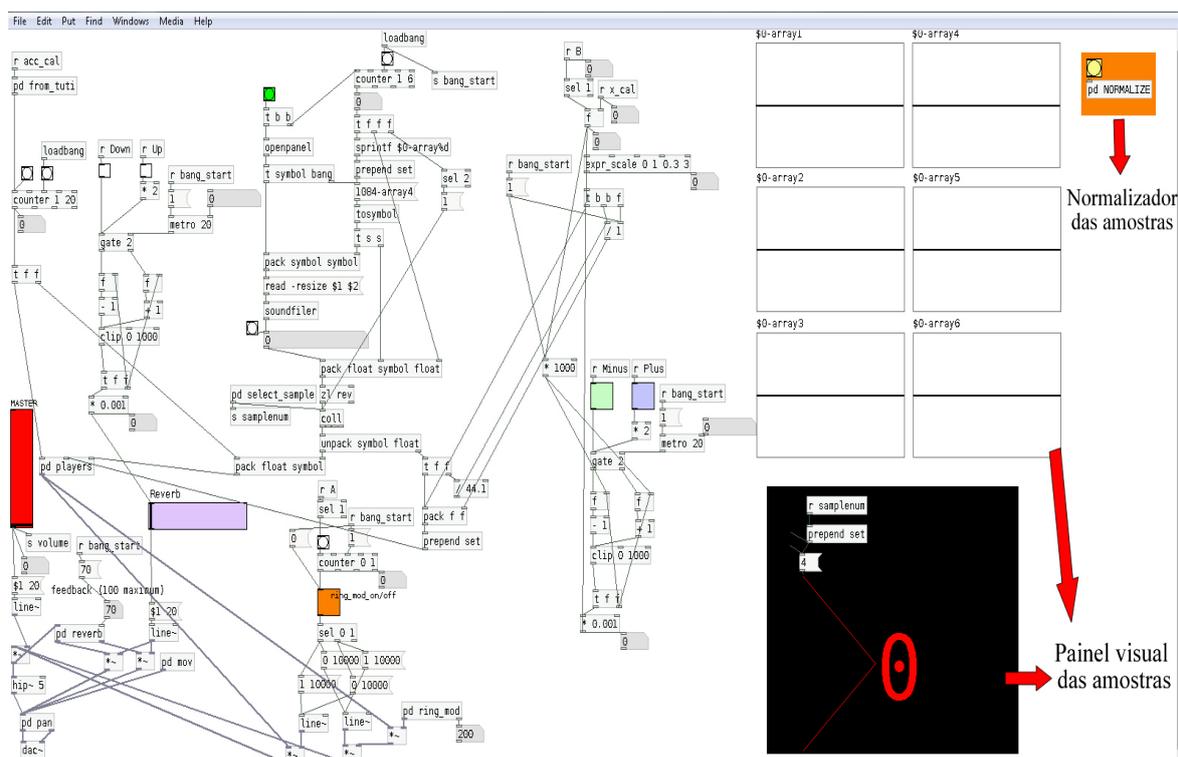


Figura 9. Painel Principal do aplicativo

#### 4. CONCLUSÃO

Os experimentos realizados com o *Wii mote* – tanto no caso do Theremin virtual quanto no aplicativo de reprodução de amostras – viabilizaram novas formas de lidar com dispositivos tecnológicos utilizando-os criativamente em um contexto musical.

Vimos na experiência do Theremin virtual que a utilização de Led's captados pela câmera infravermelha do controlador *Wii mote* possibilitou a manipulação dos parâmetros de frequência e amplitude de forma análoga a do Theremin real. Foi possível também expandir as potencialidades do instrumento através da utilização de escalas específicas, como a escala de tons inteiros, por exemplo. O leque de sonoridades que o Theremin virtual pode gerar é

também mais amplo que o do Theremin real, pois o *software* Pure Data (Pd) permite a programação de muitas alternativas de geração sonora além das ondas senoidais. O controle dos parâmetros através da movimentação de Led's infravermelhos abre inúmeras possibilidades para trabalhos colaborativos com dançarinos(as), por exemplo, permitindo que a música em uma performance possa ser gerada em tempo real com base nos movimentos coreográficos. Além disso, os Led's podem ser utilizados por músicos enquanto tocam seus instrumentos, permitindo que os gestos de execução instrumental sejam utilizados para controlar o processamento em tempo real dos sons dos instrumentos, ampliando, conseqüentemente, suas potencialidades sonoras e os recursos de controle disponíveis aos músicos.

O *Wiiote* como controlador de amostras se apresentou como uma ótima ferramenta musical, oferecendo diversos recursos como transposição, controle de dinâmica (amplitude), modulação em anel, reverberação e espacialização das amostras. Apesar das dificuldades enfrentadas durante as etapas de implementação, verificou-se através das sessões de improvisação livre que os desenvolvimentos proporcionaram resultados musicais satisfatórios. Porém, pretende-se realizar uma série de aprimoramentos nos aplicativos, tais como:

- a) Criar uma luva que se permita o acoplamento de Led's para substituir as canetas infravermelhas, o que garantirá maior versatilidade e liberdade de movimento em uma performance;
- b) Ampliar o número de Led's a serem captados (até o máximo de quatro, que é o limite de rastreamento do *Wiiote*), buscando formas que evitem a interferência entre os feixes de luz dos Led's individuais;
- c) Desenvolver um sistema de espacialização em que o *Wiiote* controle as amostras distribuindo-as em 4 ou 8 canais;
- d) Criar a possibilidade de que um músico grave os sons gerados pelos co-participantes de uma performance para que possa submetê-los às transformações disponibilizadas pelos *patches* (como a transposição, modulação em anel e reverberação), criando, assim, uma forma de interação ainda mais integrada;
- e) Incluir no *patch* de reprodução de amostras a possibilidade de utilizar mais do que seis amostras, garantindo, assim, uma maior gama de sonoridades;

- f) Desenvolver um painel de controle geral nos *patches* para uma melhor visualização das ações, com o objetivo de estabelecer um controle mais eficiente das mudanças de amostras e processamentos realizados.

O aspecto mais interessante dessa pesquisa é que os trabalhos realizados desembocaram em questões sobre como trabalhar novas perspectivas e novas ideias na música eletroacústica. Como diz Iazzetta:

O desafio da pesquisa em música eletroacústica reside hoje no desenvolvimento de sistemas que permitam o controle em tempo real de toda a riqueza sonora trazida pelos computadores, sintetizadores e outros aparelhos. Os sons eletrônicos são plásticos, maleáveis, moldáveis, justamente por estarem despregados da materialidade dos instrumentos mecânicos. Sua materialidade bruta dá lugar ao estado de virtualidade pelo qual eles se formam dentro dos *chips* dos computadores. E para que se possa explorar toda riqueza que eles potencializam, é preciso que se criem sistemas de controle interativo eficientes, baseados em interfaces funcionais. O desafio imposto para a criação desses sistemas musicais interativos está na reconciliação entre a corporalidade que sempre esteve ligada à música e a imaterialidade das estruturas sonoras que as novas tecnologias musicais vêm tornando disponíveis (IAZZETTA, 1997, p. 18).

Nesse contexto, vale destacar que o trabalho aqui exposto não é apenas fruto de uma questão meramente tecnológica, mas muito mais a prospecção de uma musicalidade que se beneficie das alternativas propostas pela tecnologia atual. O que procuramos foi um aguçamento da sensibilidade e da criatividade musical através do aprimoramento dos modos de controlar o desenvolvimento sonoro e colocá-lo no espaço acústico sob forma de música. Portanto, verifica-se a grande importância em estudar e vivenciar esta nova vertente que investiga a interação entre ser humano e tecnologia, instigando o músico a buscar novas soluções criativas e motivando-o a lidar com novos instrumentos e ferramentas para o controle dos eventos sonoros num contexto musical.

## REFERÊNCIAS BIBLIOGRÁFICAS

FORNARI, J.; MANZOLLI, J. Modelos de Síntese Expandidos por Interfaces de Jogos. In: XX CONGRESSO DA ANPPOM, 2010, Florianópolis. Anais..., Goiânia: ANPPOM, 2010. 1 CD-ROM.

HOLZMANN, G. Sound in Spatialization in Pd. Disponível: <<http://grh.mur.at/publications/sound-spatialization-pd>> Acesso em: 20/07/2011.

IAZZETTA, F. A Música, o Corpo e as Máquinas. Revista Opus: Revista da Associação Nacional de Pesquisa e Pós-Graduação em Música - ANPPOM, Goiânia, v.4, p 27-44, 1997. Disponível: <<http://www.eca.usp.br/prof/iazzetta/papers/opus.pdf>> Acesso em: 03/07/2011.

KREIDLER, J. Programming Electronic Music in Pd. 2009. Disponível: <<http://www.pdtutorial.com>> Acesso em: 20/07/2011.

LEE, J. C. Wiimote Projects. Disponível: <<http://johnnylee.net/>> Acesso em: 20/07/2011.

MENEZES, F. Música Eletroacústica – história e estéticas. São Paulo: Edusp, 1996. 288 p.

PORRES, A. Tutorial de Pure Data em Português 2.0. Disponível: <<http://sites.google.com/site/porres/pd>> Acesso em: 20/07/2011.

ROCHA, F. Questões de Performance em Obras Eletrônicas Mistas. In: XX CONGRESSO DA ANPPOM, 2010, Florianópolis. Anais..., Goiânia: ANPPOM, 2010, 1 CD-ROM.

TRALDI, C. Interpretação Mediada & Interfaces Tecnológicas para Percussão. Campinas. Dissertação (Mestrado em Música) - Universidade Estadual de Campinas, 2007. 121 p.

TRALDI, C. A.; AGUIAR, D. S.; BARREIRO, D. L. *[Wii]improviso*: controle gestual numa improvisação com sons eletroacústicos em tempo real. In: Anais do 13o. Simpósio Brasileiro de Computação Musical – SBCM 2011, 2011, Vitória. Anais..., Vitória/Porto Alegre: FAESA/UFRGS, 2011. 1 CD-ROM.

WANDERLEY, M. M. Instrumentos Musicais Digitais: gestos, sensors e interfaces. In: ILARI, B. (ed.) Em Busca da Mente Musical. Curitiba: Editora da Universidade Federal do Paraná, 2006. Disponível: <[http://idmil.org/\\_media/wiki/instrumentos\\_digitais\\_final.pdf](http://idmil.org/_media/wiki/instrumentos_digitais_final.pdf)> Acesso em: 20/07/2011.

WRIGHT, M.; FREED, A.; MOMENI, A. OpenSound Control: State of the Art 2003. In: Conference on New Interfaces for Musical Expression (NIME-03), 2003, Montreal, Canada. Proceedings... 2003, p.153-159. Disponível: <<http://opensoundcontrol.org/files/Open+Sound+Control-state+of+the+art.pdf>> Acesso em: 20/07/2011.