

Síntese de Imagens Controladas por Áudio

Mariana Zapparoli Martins, Marcelo Queiroz

Dep. de Ciência da Computação – Universidade de São Paulo (USP)
R. do Matão, 1010 – Cid. Universitária – 05508-090 – São Paulo - SP - Brasil

mzm@ime.usp.br, mqz@ime.usp.br

Abstract. *In this paper a Pd object library is described which combines audio analysis and image synthesis tools for generating visual accompaniments to musical input data. The user establishes connections that determine how musical parameters affect the synthesis of graphical objects, and controls these connections in real-time during performance. The library combines a straightforward communication protocol for exchanging musical and visual parameters with an easy-to-use interface which makes it accessible for users with no computer programming experience. Its potential applications areas include children's musical education and entertainment industry.*

Resumo. *Neste artigo é apresentada uma biblioteca de objetos Pd que combina ferramentas de análise de áudio e síntese de imagens para a geração de acompanhamentos visuais a uma entrada musical. O usuário estabelece conexões que determinam como os parâmetros musicais afetam a síntese de objetos gráficos, e controla estas conexões em tempo-real durante a performance. A biblioteca combina um protocolo de comunicação para intercâmbio de parâmetros musicais e visuais com uma interface fácil de usar, tornando-a acessível a usuários sem experiência em programação de computadores. Suas áreas possíveis de aplicação incluem a musicalização infantil e a indústria de entretenimento.*

1. Introdução

Este trabalho tem como objetivo descrever um conjunto de objetos para a ferramenta Pd (Puredata, ver [Puckette 1998]) que possibilite de forma simples e interativa a criação e modificação de associações entre informações de áudio e de objetos gráficos, a fim de permitir a criação de acompanhamentos visuais para uma entrada musical. Entende-se por acompanhamento visual uma animação (em vídeo) que corresponde a uma síntese de imagens onde a definição dos parâmetros que controlam os objetos gráficos utiliza informação musical extraída automaticamente da entrada de áudio e atualizada em tempo-real.

Recentemente, o aumento do poder de processamento dos computadores pessoais e a facilidade de circulação de informação multimídia em banda-larga fizeram com que muitos sistemas computacionais para associação de áudio e vídeo fossem desenvolvidos [Rowe 2001]. Alguns exemplos destes sistemas são: *Isadora*, *PixelToy*, *AVS*, *G-Force*, *Milkdrop Soundscape*, *Egosound* e o projeto *Conducting a Virtual Orchestra*.

Em particular, alguns sistemas (também chamados *instrumentos visuais* [Collopy 1999]) estão voltados à geração de imagens e permitem o controle interativo da síntese em tempo-real. Um destes instrumentos visuais é o *GEM* [Danks 1997], que utiliza os conceitos do *OpenGL* e fornece objetos Pd para síntese de imagens a partir da descrição de uma cena 3D.

Na próxima seção são discutidos alguns conceitos envolvidos na especificação dos objetos, com ênfase no protocolo de comunicação entre as camadas de processamento de áudio e síntese de imagens. Em seguida são apresentados brevemente os objetos implementados e por fim são feitas algumas considerações sobre aplicações e trabalhos futuros.

2. Entre som e imagem

No projeto de objetos gráficos que respondem a uma entrada musical, deve-se considerar em primeiro lugar que tipo de informação será extraída da entrada, levando em conta as restrições impostas pelo requisito de operação em tempo-real. A análise espectral utilizando janelas pequenas (*Short-Time Fourier Transform*) permite a obtenção de diversas características sonoras, tais como frequência predominante (no caso de sinais quasi-periódicos) e padrões de espalhamento ou concentração de energia em diversas bandas de frequência. A partir da envoltória dinâmica é possível identificar ataques. Da observação destas análises ao longo do tempo é possível detectar *glissandos*, *crescendos* e *decrecendos*, padrões rítmicos e mudanças de andamento.

A síntese de imagens por computador pode ser feita de inúmeras maneiras, sendo uma destas a construção da imagem a partir da descrição dos elementos que compõem uma cena tridimensional (*renderização*), tais como objetos, fontes de luz e câmeras. Os parâmetros que controlam estes elementos podem ser de posição, dimensões, cor, iluminação, posição da câmera e atributos da superfície dos objetos, como rugosidade, transparência, espelhamento, entre outros.

Independentemente da escolha feita em relação ao tipo de objeto que será sintetizado, um problema evidente ao se controlar imagens a partir de informações retiradas do áudio é a tradução dos parâmetros musicais em parâmetros visuais. Não se trata aqui de determinar associações entre estes parâmetros, o que evidentemente é um problema estético cuja solução (particular e subjetiva) será dada pelo usuário do sistema, mas em representar tais parâmetros de modo a permitir ao usuário o máximo de liberdade nestas associações. Neste contexto específico, parece natural propor que a tradução entre som e imagem seja intermediada por um universo de representação abstrato, que não seja nem musical nem visual a priori, mas que permita o estabelecimento de conexões diversas entre estes dois universos.

A partir da observação de características elementares do áudio, tais como envelope dinâmico, altura e ataques, parece natural representar tais características como sinais discretos (funções do tempo) com valores variando num intervalo arbitrário $[0...1]$, ou alternativamente como disparos (*bangs*). Essas duas modalidades de representação permitem um mapeamento convencional de parâmetros sonoros que admitem uma ordem linear (tais como altura e intensidade) através de sinais $[0...1]$ bem como a representação de eventos pontuais (tais como ataques ou detecção de notas específicas) através de disparos. Por outro lado, tamanho e posição de objetos gráficos ou cores no sistema RGB também podem ser representadas por valores entre $[0...1]$; disparos podem ser úteis para sinalizar o início de uma seqüência qualquer de movimentos de um objeto.

Desta maneira, a comunicação entre objetos de análise de áudio e objetos de síntese de imagens é feita através de sinais discretos de controle que variam no intervalo $[0...1]$ ou através de disparos. Cada valor produzido por um objeto de análise de áudio, bem como cada parâmetro de controle de síntese de imagem, pertencem claramente a uma destas duas categorias, e qualquer conexão entre saídas $[0...1]$ e entradas $[0...1]$ ou

entre saídas *bang* e entradas *bang* é possível, e estabelece uma relação causal entre a entrada de áudio e a síntese de imagens.

3. A implementação do sistema AIM

A escolha da ferramenta Pd como plataforma-base para processamento em tempo-real se deve a vários fatores, sendo os mais importantes o fato de se tratar de um software livre, multi-plataforma, que conta com objetos nativos para gerenciamento de áudio e MIDI além de inúmeras bibliotecas que estendem sua funcionalidade, como a biblioteca GEM para síntese de imagens. Desta maneira, foi natural planejar o desenvolvimento do sistema como uma coleção de objetos Pd que oferecesse facilidades para a obtenção de resultados visuais de forma simples e rápida a partir de uma entrada musical, incluídos aí objetos para geração de interfaces que pudessem ser facilmente controladas em tempo-real durante a performance.

Esta coleção de objetos foi batizada de AIM (*Audible Images*) e os objetos foram divididos em quatro categorias: objetos para análise do áudio de entrada, objetos para síntese de imagens, objetos para síntese e processamento de sinais, e interfaces. Um objeto central (*aim.control*) é responsável pela geração da janela GEM e pela entrada de áudio, possuindo controles para a câmera e o fundo de cena, bem como recursos para gerenciamento de microfones e *playlists*.

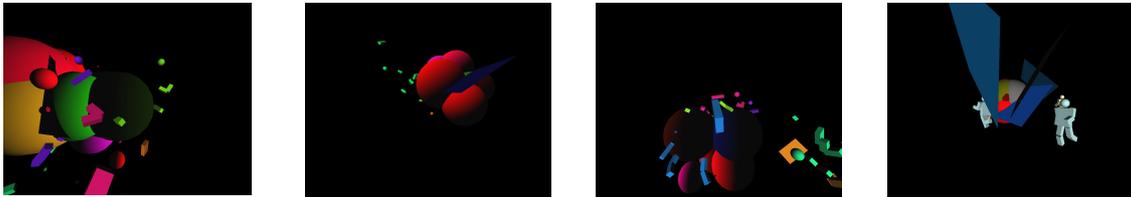
Pertencem à categoria de análise de áudio objetos tais como *aim.pitch* (para altura musical) e *aim.amplitude*, que tanto servem para extrair informações pontuais do sinal como permitem a obtenção de médias ao longo de um período maior de tempo, *aim.attack*, que detecta inícios de notas ou eventos percussivos, *aim.timedensity*, que obtém a densidade temporal de eventos através da contagem do número de ataques por segundo e *aim.spectraldensity*, que faz a análise da densidade espectral através da contagem da proporção de picos espectrais proeminentes.

Atualmente o sistema AIM possui três objetos primitivos e três objetos complexos para a síntese de imagens, todos a partir de figuras geométricas. A título de exemplo serão descritos os três objetos compostos: *aim.balls*, *aim.squares* e *aim.dolls*. O objeto *aim.balls* produz 28 esferas controladas por 7 parâmetros abstratos, que determinam propriedades distintas (raio, cores, posição) em cada bola. O objeto *aim.squares* controla o lançamento de quadrados que se movimentam em direção ao fundo da cena, sendo seus movimentos e cores controlados independentemente. O objeto *aim.dolls* gerencia o movimento de 6 bonecos formados por paralelepípedos e esferas, oferecendo controles de cor, raio das trajetórias e coesão das partes que compõem os bonecos, permitindo a aproximação visual com os outros objetos de síntese de imagem.

Os objetos para processamento e síntese de sinais permitem modificar um sinal de controle AIM, ou mesmo gerar um sinal de controle a partir de uma descrição formal, e fornecem uma flexibilidade maior nas conexões feitas entre a camada de análise de áudio e a camada de síntese de imagens. Dentro da categoria de objetos para o processamento de sinais estão os objetos *aim.invert*, *aim.chebyshev*, *aim.compress*, *aim.expand* e *aim.gate*. Na categoria de síntese de sinais a biblioteca possui os objetos *aim.sine* e *aim.cosine*, *aim.phasor* e *aim.random*.

As interfaces são objetos da biblioteca AIM que permitem que as conexões entre a camada de áudio e a camada gráfica possam ser alteradas em tempo-real de forma rápida e simples, através de chaves liga-desliga. Uma destas interfaces é o *aim.interface8*, que fornece uma matriz 8x8 de chaves liga-desliga que controlam o fluxo

de 8 sinais [0...1] provenientes de análise de áudio (ou síntese e processamento de sinais), correspondentes às linhas da matriz, para 8 controles de síntese de imagens, correspondentes às colunas. Além do controle direto pelo usuário, a interface também permite a leitura de arquivos contendo configurações das chaves liga-desliga, bem como um modo piloto-automático de geração destas conexões de forma aleatória.



Exemplos de saída utilizando diversos objetos de síntese de imagem

4. Conclusões e Trabalhos Futuros

Neste trabalho foi apresentada a biblioteca *AIM (Audible Images)*, cujo objetivo é servir de ferramenta para a exploração de associações entre parâmetros sonoros e parâmetros visuais, permitindo a geração de acompanhamentos visuais a uma entrada de áudio através da construção de patches simples e diretos, podendo ser usado por pessoas sem experiência de programação. Aplicações possíveis dessa biblioteca incluem projeções em festas e casas noturnas bem como o uso pessoal, desenvolvimento de jogos onde os objetos ou personagens relacionam-se com a música de fundo, e como recurso didático na musicalização infantil. O trabalho futuro inclui o desenvolvimento de novos objetos de análise de áudio, síntese de imagens e interfaces, bem como uma avaliação junto a usuários através de questionário, a fim de aprimorar a biblioteca.

5. Referências

- Collopy, F., Fuhrer, R. e Jamenson, D. (1999) “*Visual Music in a Visual Programming Language*”, In: IEEE 1999 Symposium on Visual Languages, pps. 111-118.
- Danks, M. (1997) “*Real-time Image and Video Processing in GEM*”. In: Proceedings of the International Computer Music Conference (ICMC) 1997, pps. 220-223.
- Iazzetta, F. (2003) “*A Performance Interativa em Pele*”, In: IX Simpósio Brasileiro de Computação Musical, Campinas, São Paulo.
- Nagashima, Y., (1998) “*Real-time Interactive Performance with Computer Graphics and Computer Music*”, In: Proc. of the 7th IFAC/IFIP/IFORS/IEA Symposium on Analysis, Design, and Evaluation of Man-Machine Systems.
- Puckette, M., Apel, T. e Zicarelli, D. (1998) “*Real-time audio analysis tools for PD and MSP*”, In: Proc. of the Intl. Computer Music Conference (ICMC) 1998.
- Rowe, R. (2001) *Machine Musicianship*, The MIT Press, Cambridge, Massachusetts.
- Tarabella, L., Magrini M. e Scapellato G. (1997) “*Devices for Interactive Computer Music and Computer Graphics Performances*”, In: IEEE First Workshop on Multimedia Signal Processing, pps. 65-70.