UNIVERSIDADE FEDERAL DE PERNAMBUCO
GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO
CENTRO DE INFORMÁTICA

EXTRAÇÃO DE INFORMAÇÃO SIMBÓLICA DE ÁUDIO: ALGORITMOS DE RECONHECIMENTO DE ONSETS

TRABALHO DE GRADUAÇÃO

Aluno: Roberto Cássio S. N. Júnior (rcsdnj@cin.ufpe.br)
Orientador: Geber L. Ramalho (glr@cin.ufpe.br)

Março de 2007

AGRADECIMENTOS

Agradeço a todos que de alguma maneira, direta ou indiretamente, intencionalmente ou não, tiveram influência na realização neste trabalho.

Ao pessoal da D'Accord Music Software, em especial Hugo e Américo, com todo o apoio, orientação, código fonte e incentivo que recebi para realização do projeto.

Ao meu orientador, Geber Ramalho, pela relação amistosa que mantém com o aluno e a excelente capacidade de cumprir sua função de orientador.

À minha família, e meus amigos, por todo o apoio e infra-estrutura social e emocional que me sustenta.

A Patrícia, minha namorada, que deu apoio crucial em momentos bem difíceis neste percurso.

A Ricardo Scholz, pela contribuição essencial na parte experimental do trabalho, bem como o fornecimento de algumas sugestões de suma importância.

À Existência, que, independente da pretensão de determinar se é cônscia ou não de si, permitiu pelo fluxo de execução das coisas de tal forma este trabalho fosse realizado.

RESUMO

O presente trabalho trata de fazer um estudo geral sobre como encontrar de forma automática a entrada de notas musicais em arquivos de áudio, e realiza experimentos com a intenção de fazer uma validação da qualidade das melhores técnicas existentes.

“E nunca considerem seu estudo uma obrigação, mas sim como uma oportunidade invejável de aprender sobre a influência libertadora da beleza no domínio do espírito, para seu prazer pessoal e para o proveito da comunidade à qual pertencerá o seu trabalho futuro”

Albert Einstein

ÍNDICE

1. Introdução

Onset é uma palavra que poderia ser traduzida literalmente em algo como "no momento do ajuste", ou ainda, "no momento em que é definido o estado", tendo em vista que a sua formação vem da junção das palavras "on" + "set" na língua inglesa. No caso tratado aqui, onset diz respeito aos momentos em um arquivo de áudio onde se percebe a entrada de uma nota musical ou instrumento percussivo. Em alguns momentos poderemos fazer referência tanto ao "problema da detecção de ataques" quanto à "detecção de onsets", nos referindo ao mesmo objeto de estudo, mas a rigor há uma diferença sutil que perceberemos mais adiante.

Será feito um estudo geral sobre o assunto, compreendendo o problema da detecção dos onsets numa música, e alguma das técnicas mais importantes de como solucioná-lo. Ademais, estudaremos como se comportam o um algoritmo tido atualmente como um dos melhores em um caso particular, que é o de bossa nova executada no violão.

1.1. Motivação

A área de interesse principal, a ser escolhida para realizar este trabalho de graduação, foi a de computação musical. Esta tem a capacidade de reunir conhecimentos diversos de forma bastante singular, incluindo áreas como física, matemática, e inteligência artificial. Diversos eram os temas disponíveis – vamos entender então por que detecção de ataques foi o escolhido.

1.1.1. Onset no contexto geral

Música é uma das formas de artes mais universalmente envolventes. Muito raro é encontrar alguma pessoa que não a aprecie, ainda que possua gosto bastante restrito. A julgar por esta capacidade de envolver o ser humano, até mesmo em seus mais profundos sentimentos, alguns têm até dificuldade de ver a música com definições mais pragmáticas, como por exemplo "uma complexa e bem organizada sucessão de momentos de sons e silêncio".

É no intuito de conciliar esta bela manifestação artística com o mundo aparentemente "frio" das máquinas que se sustenta a motivação para a escolha deste tema para o Trabalho de Graduação.

O interesse principal passa pela extração de informação simbólica em de música em geral. Todavia, para delimitar um foco compatível com um escopo para este tipo de trabalho, a opção escolhida foi um estudo sobre o problema da detecção de localização de ataques, ou "*onset detection*".

A detecção automática de eventos de onset dá margem a uma ampla gama de aplicações, tanto em música quanto no contexto geral de tratamento de áudio. Podemos destacar algumas diretas, tais como transcrição automática de melodia para formatos simbólicos (ex. MIDI), e técnicas de reconhecimento de voz, em que é diretamente notável a importância de se conhecer os momentos de em que os sons se iniciam.

Com a informação básica do momento de entrada de sons, percussivos ou tonais, é possível inferir outras características relevantes da música, como por exemplo o andamento, ou mesmo realizar o acompanhamento automático (*beat tracking*), assim como os seres humanos fazem ao bater com o pé ao som de uma melodia.

Detecção de onsets abre também novas possibilidades em aplicações como compressão, indexação e recuperação de informação musical. Encontrar uma música numa base de dados, por exemplo, pode ser uma operação bastante otimizada se levarmos em conta simplesmente a similaridade nos tempos em que ocorrem os elementos musicais relevantes.

Em compressão de dados, minimizar redundância é sempre desejável. Os tempos de execução dos eventos mais significativos de mudanças na música são uma informação que, logicamente, já está na codificação a ser tocada do áudio. A identificação destes padrões, com grau razoável de certeza, e de forma isolada e separada, pode servir para otimizar bastante a maneira como o fluxo de dados de áudio é armazenado.

1.1.2. O problema escolhido

Além das aplicações práticas que já foram exemplificadas, também podemos considerar a necessidade de análise da estrutura musical em si. Classificar uma música em um determinado gênero musical, deduzir quais os diferenciais que um determinado estilo tem. Ou mesmo, adentrar-se com mais profundidade neste assunto e dar ainda mais granularidade à classificação a ponto de tornar possível a distinção entre dois instrumentistas, levando em conta, entre outros fatores, pequenas variações de tempo que são geralmente encontradas na interpretação de cada um, em comparação com o que é especificado na partitura.

O que há nestas pequenas variações que gera riqueza (ou não) à performance artística? Quais os elementos humanamente perceptíveis (ainda que de forma subconsciente), e quais são os irrelevantes?

A resposta a estas perguntas não será obtida de maneira direta dentro do escopo deste trabalho. No entanto, este estudo tem potencial para auxiliar a respondê-las. Qual a precisão que conseguimos obter com estas técnicas? Que grau de confiabilidade é possível obter com estes resultados?

Tendo estas indagações em mente, e levando em conta os estudos que vêm sendo feitos sobre música brasileira – especificamente, o projeto "Um País, Um Violão", escopo da tese de mestrado de Ernesto Lima, veio à tona a grande pergunta em que se sustenta a motivação para este texto e os esforços de pesquisa: **que grau de confiança os algoritmos de detecção de onsets oferecem para este tipo de análise, ou seja, quando trabalham restritos a um conjunto de arquivos de áudio de violão, na execução deste gênero musical**?

Ainda sobre esta tese, existe disponível uma rica base de arquivos musicais captados de trechos de música brasileira, utilizando um violão que fornece saída acústica e MIDI. Um arquivo MIDI possui codificados, diretamente, os tempos de entrada das notas, ou seja, os onsets, dos quais tratamos. **Mas, qual a precisão do hardware de captação que está no violão, e o quanto esta informação correspondente à realidade em relação à maneira como percebemos os ataques na música?**

1.2 Objetivo

Tendo em mente os motivos que levaram a este tema, e as indagações apresentadas, este estudo iniciará apresentando uma análise das técnicas de detecção de onset existentes, o princípio por trás delas e quais as principais abordagens utilizadas na busca por aumentar o grau de certeza dos resultados. Uma explicação didática para entendimento da construção de um algoritmo mais simples também estará inclusa.

Uma vez que se tenha esta visão geral sobre o problema, conforme já esperado pelo que foi citado como motivação, será escolhido um algoritmo que tenha destaque na qualidade dos resultados (mesmo que com outro tipo de instrumento musical), e averiguado como este se comportará para o caso específico que foi levantado, ou seja, em uma performance de bossa nova ao violão.

Estes resultados serão comparados com os da captação MIDI do violão, sobre a base que foi mencionada, e analisaremos o que poderemos inferir dos mesmos.

1.3 Abordagem

Para alcançar estes resultados, o primeiro passo foi a leitura de diversos artigos a respeito do assunto, classificação em nível de relevância ao tema e seleção para estudo e abordagem.

Já com um conhecimento mais geral a respeito das técnicas existentes, a estratégia utilizada foi fazer uma implementação própria de um algoritmo de detecção. Uma implementação simples, que não tem intuito de trazer inovação e resultados comparáveis ao estado da arte, mas sim servir para o aprendizado prático e interativo sobre a construção de detectores do tipo e os problemas genéricos envolvidos.

Uma pesquisa sobre as metodologias para realizar a avaliação de resultados dos algoritmos também foi realizada. Desta, selecionamos uma para realizar as observações sobre as questões levantadas a respeito da execução no violão do estilo musical que foi escolhido como referência. Criamos nosso *framework* de testes e analisamos os dados obtidos.

2. Problema

Este capítulo definirá os principais elementos de estudo deste trabalho, trazendo alguns conceitos importantes e mesmo informações sobre seus contextos de utilização.

2.1. O som e sua representação

O som é uma compressão mecânica variando no tempo que se propaga de forma circuncêntrica, em meios que tenham massa e elasticidade como os sólido, líquido ou gasoso. Em particular, o que chega aos nossos ouvidos é uma sucessão de oscilações periódicas na pressão do ar, que faz vibrar nossos tímpanos no mesmo ritmo. A velocidade desta oscilação é chamada freqüência, e usualmente medida em unidades de oscilação por segundo (Hertz ou Hz).

Quando esta oscilação é transmitida ou armazenada em algum lugar, a entidade que varia com o tempo deixa de ser a pressão mecânica de algum material e é substituída por outra grandeza.

Esta grandeza pode oscilar de forma análoga, ou seja, de uma maneira que possa representar a mesma oscilação, com mudança apenas da escala de oscilação e da grandeza física oscilatória em si. Exemplos de transmissão ou armazenamento analógico podem ser vistos em fitas cassetes comuns (um campo magnético aumenta e diminui de acordo com o a variação do som original), discos de vinil (a profundidade de ranhuras na superfície é a entidade variável), ou um fio de um microfone conduzindo eletricidade para a transmissão do áudio.

A figura 1 tem a representação de uma oscilação desta natureza no tempo, particularmente da pressão atmosférica – que é o que faz vibrar nossos ouvidos para que percebamos o som.

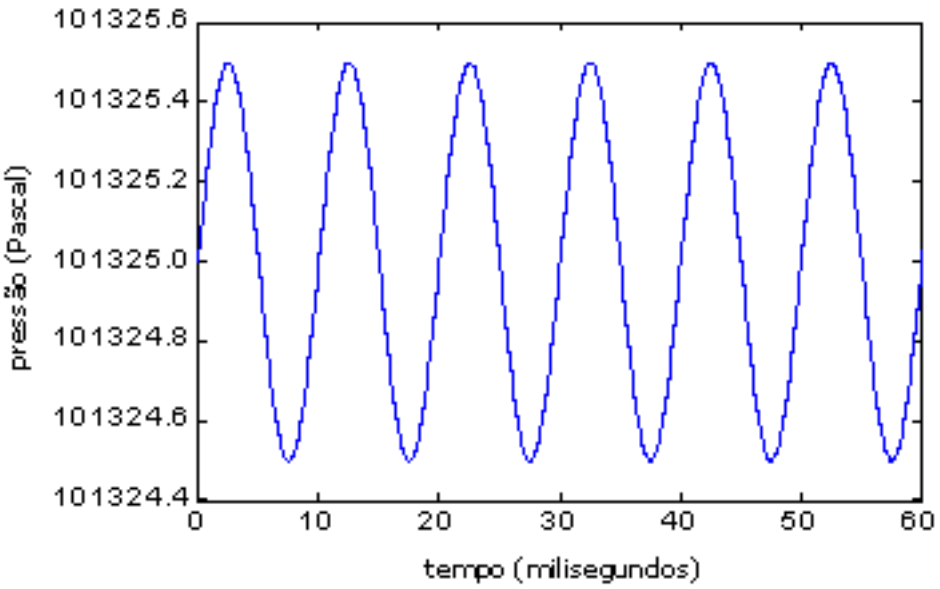

Fig. 1 - Oscilação da pressão atmosférica no tempo que nos faz perceber um som

A transmissão, processamento e armazenamento em sistemas são um tanto diferentes. Em linhas gerais, as medidas da grandeza física oscilatória são mapeadas para um conjunto discreto de valores (que pode ser representados por símbolos, tais como 0 ou 1, ou um caracter *ASCII*). Estas medidas são feitas a uma certa velocidade, como fotografias tiradas do som. O tamanho do conjunto discretos de valores para o qual o mapeamento está sendo feito é o tamanho da amostra, e é normalmente definido em bits por amostra (ou *bits/sample*). A quantidade destes valores que são mapeados em cada unidade de tempo é chamada taxa de amostragem (usualmente contabilizada em amostras/segundo).

Na prática, o resultado é uma representação de algo bem próximo do sinal analógico original, mas ligeiramente truncado, o que dependendo da qualidade utilizada não é perceptível ao ouvido humano. Uma visualização propositadamente exagerada deste fenômeno está na figura 2.

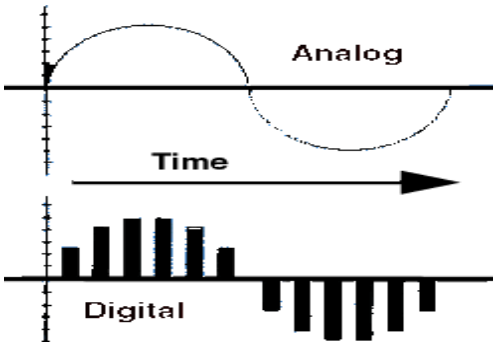

Fig. 2:Som armazenado digitalmente, em baixa resolução, para demonstrar como os dados digitais são truncados

Quase nunca a oscilação sonora é simples e bem comportada tal como a onda usada no exemplo anterior. Em geral, o som é uma composição de sinais em freqüências diversas, com parâmetros também variados, mudando com o tempo. A figura 3 exibe o comportamento do sinal auditivo ao longo do tempo para uma corda de violão sendo tocada.

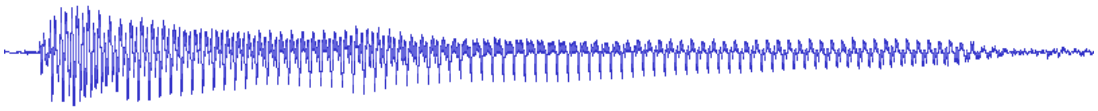

Fig. 3: O comportamento oscilatório no tempo, observado ao tocar uma corda de violão, é bem mais rico e complexo, pois contém sons de diversas freqüências misturadas.

2.2. Algumas definições: Ataque, decaimento, onset e transiente

Em geral falamos sobre "encontrar os ataques", mas, quando usamos a palavra em inglês "onset", temos uma definição do significado ligeiramente mais sutil e precisa. Vejamos esclarecimentos para estes e outros termos relevantes ao tema analisado.

2.2.1 Ataque

Ataque é o nome dado a todo o intervalo de tempo em que a intensidade do sinal está ascendendo, de forma brusca, o que no caso de uma música deve-se, geralmente, à entrada de uma nota musical ou surgimento de som percussivo. Em outras palavras, para um exemplo em que a intensidade do sinal estava zero e uma nota musical foi tocada (ou uma pancada foi dada em um instrumento percussivo), o ataque é o nome dado ao intervalo de tempo em que o sinal está saindo do nível de intensidade zero, até atingir um determinado máximo.

2.2.2 Decaimento

Logo após ocorrer do ataque, é comum (dependendo do tipo de instrumento tocado e da maneira como o som é executado) esperar que haja o chamado decaimento. É apenas o processo inverso: após atingir seu máximo, o sinal se sustenta por um determinado período de tempo (este chamado de sustentação), e começa a diminuir gradativamente (e rapidamente, dependendo do caso) de intensidade.

2.2.3 Transiente

Transiente diz respeito a um caso mais geral. Como o próprio nome sugere, trata-se daquilo que é transitório; em específico, chamamos de transiente os intervalos de tempo onde o a intensidade do sinal sofre algum tipo de variação rápida, crescente ou decrescente. Os intervalos onde há ataque e o decaimento, observados no tocar de numa nota de piano, por exemplo, são transientes.

2.2.4 Onset

Onset, diz respeito ao instante. Traduzindo de forma aproximada, seria "no momento do ajuste, da definição". Neste caso, trata-se do tempo exato da definição do ataque, o seu início – o momento preciso em que o volume do som começa a aumentar por causa de um novo elemento sonoro entrante.

2.2.5 Visualizando estes momentos

Fica muito mais fácil e palpável perceber o que são estas definições ao vê-les na prática. A figura 4, a seguir, dá uma mostra precisa e detalhada do que são e onde estão cada um dos elementos. Trata-se de um gráfico que mostra o volume sonoro variando no tempo, no momento em que um uma nota entra.

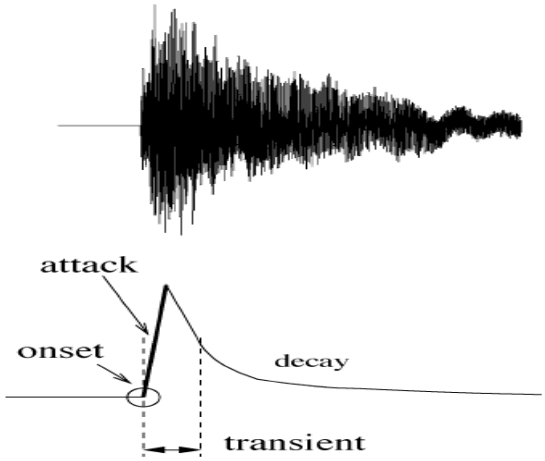

Fig. 4: O som original e o acompanhamento da evolução de sua amplitude média no tempo, onde podemos identificar o momento do onset, o ataque (também considerado um transiente), e o período de decaimento

2.3. Detalhes envolvidos na detecção

A detecção de onsets possui alguns detalhes, inclusive de natureza subjetiva, impondo que alguns critérios precisem ser bem definidos.

2.3.1. Onset verdadeiro e o percebido

O ataque, e por conseqüência o onset, são entidades bem definidas, conforme já foi colocado. No entanto, existe uma certa distância entre o que podemos identificar, visualmente, em um gráfico, marcando o início do ataque (ou num cronômetro, indicando por exemplo o instante em que uma corda foi tocada), e o que um ser humano percebe.

Existe uma área específica para estudar os sons e os efeitos que são capazes de provocar no cérebro: a psicoacústica. Enquadrando-se nesta área de pesquisa, alguns trabalhos indicam que a percepção do onset ocorre quando o volume chega a um nível entre 6 e 15 decibéis abaixo do máximo.

2.3.2. Múltiplos instrumentos (polifonia)

O momento dos onsets é algo bastante nítido e observável em uma música em que haja apenas um instrumento tocando. Todavia, quando se trata de polifonia, há detalhes a serem considerados e algumas decisões a serem tomadas.

Notas com intervalo de tempo muito curto entre si podem ser confundidas, pelo próprio ouvido humano, como uma única nota soando. Para estes casos, a complexidade é maior e já não são válidos os estudos de psicoacústica indicando a diferença entre a percepção humana e o tempo real do onset.

Mesmo sem contar com as variações na percepção humana, existem alguns fatores sobre os quais é preciso, muitas vezes, arbitrar decisões subjetivas. Um acorde, por exemplo – onde várias notas são tocadas praticamente ao mesmo tempo – pode ser visto como um ataque único ou vários juntos, e a maneira como a audição processa isto também é discutível. Este tópico será melhor abordado no capítulo a respeito de metodologias de avaliação de algoritmos.

2.4. Ataques em variados instrumentos musicais

Instrumentos musicais distintos apresentam curvas de ataque distintas, como é de se esperar. Uma forte pancada numa tecla de piano faz com que o crescimento no volume do som se dê de forma muito mais súbita que um sopro numa flauta ou o deslizar do arco em uma corda de violino. A forma de tocar também é determinante: basta pensar num baixo acústico sendo tocado com arco ou tendo as cordas puxadas com os dedos.

Além disso, também a depender do tipo de instrumento, a resposta é usualmente diferente nas diversas faixas de freqüências. Da mesma maneira que as intensidades de emissão são perceptivelmente diferentes para cada freqüência (há timbres com mais acentuação nos graves, outros nos médios, e outros nos agudos), as respostas de ataques podem se apresentar de forma não-uniforme.

Este conhecimento pode servir para melhor definir a estratégia e os parâmetros a serem utilizados na técnica de detecção de onsets.

2.4.1 Peculiaridades na execução do violão

Sendo o foco deste trabalho, é válido salientar que o violão também tem seus detalhes em relação aos outros instrumentos. Além de, naturalmente, possuir sua curva característica de ataque, esta é suscetível a mudanças em função de fatores como a maneira como o instrumento está sendo tocado (dedilhado, ou batido), e mesmo a posição e dedos utilizados na mão direita (no caso de destros), que determina timbres mais macios ou ásperos.

Outra característica a ser levado em conta é a presença de polifonia – ou, em palavras mais simples, o fato de estarem sendo tocadas várias cordas ao mesmo tempo.

Como esta análise está sendo delimitada ao estilo musical bossa nova, estes fatores são em parte uniformizados, controlados. Mas ainda há uma certa variação em função da própria música em si e do estilo do instrumentista que está tocando.

3. Estado da Arte

Neste capítulo temos uma visão geral das principais técnicas existentes. Todavia, para que sejam melhores absorvidas, passamos antes por uma introdução explicando, de maneira simples, o processo mais básico (intuitivo) para a detecção dos onsets.

3.1. Compreensão geral dos algoritmos

A grande maioria dos algoritmos de localização de ataques passa por um conjunto comum de etapas para obter os valores de tempo dos onsets, a partir do arquivo de áudio. Compreendê-las dá uma importante visão geral das soluções para o problema, e abre o entendimento, inclusive, para ter noção dos métodos que não sigam os caminhos mais ortodoxos.

O artigo *A tutorial on Onset Detection in Music Signals* traz, de forma bastante didática, alguns pontos comuns nestes detectores de ataques, e é com base nele que vamos fornecer esta noção geral, antes da análise do estado da arte propriamente dita.

Vejamos, pois, que etapas são estas e as possibilidades mais atuais do que é realizado em cada uma.

3.1.1. Pré-processamento

Como o próprio nome sugere, o pré-processamento dará ao sinal puro de áudio algum tratamento que vise atenuar ou enaltecer determinadas características. É uma espécie de "polimento" inicial do sinal para que dele possa se extrair os eventos procurados com maior clareza.

Esta etapa é opcional, por isso nem sempre é implementada nos algoritmos – embora possa, em muitos casos, aumentar a qualidade dos resultados.

Duas operações freqüentemente realizadas nesta etapa são: divisão em bandas e separação de transientes.

3.1.1.1 Divisão em bandas

Vários são os algoritmos que tiram proveito da análise da informação de maneira separada por cada faixa de freqüência do sinal. Em alguns casos, este processamento é usado como informação extra para complementar as estimativas globais de localização de ataques; em outros, esta abordagem é uma forma de aumentar a robustez do método de detecção.

Um exemplo bastante interessante de aplicação desta técnica está no método proposto por Duxbury, que utiliza um tipo especial de banco de filtros para separar o sinal em 5 bandas. O passo diferencial dado é a proposta de um esquema híbrido que considera mudanças na energia do sinal nas regiões de alta freqüência e mudanças no espectro nas freqüências inferiores. Implementações deste tipo evitam as restrições impostas quando tratamos o sinal como um todo indivisível, pois podemos ter estratégias diferentes para faixas de freqüências diferentes.

3.1.1.2 Separação entre transientes e partes estáveis

Já sabemos que os transientes são oscilações na intensidade do sinal que ocorrem de forma súbita. Ataques são tipos particulares de transientes – os correspondentes sons que estão surgindo na música, ao longo do tempo.

Existem, todavia, diversos momentos onde o som é razoavelmente estável, não apresentando característica transitiva e, portanto, sem chances de conter um onset. Na atividade de busca dos onsets, conseguir esta diferenciação de forma preliminar permite que se manipule somente a informação mais relevante, e sendo esta etapa bem realizada, as chances de erro ficam reduzidas.

3.1.2. Redução

A redução pode ser considerada a etapa chave do processo de encontrar onsets. Tem este nome em função da própria operação que provoca: o sinal de áudio é transformado em uma função mais simples e com taxa de amostragem reduzida, isto é, se o som possuía 44100 amostras de intensidade de áudio por segundo, após esta etapa teremos uma função que possuirá – por exemplo – apenas 100 amostras por segundo.

Este sinal de baixa resolução obtido, no entanto, possui expressividade muito mais elevada, de maneira que mesmo um olhar rápido sobre esta função já permite ter uma idéia aproximada de que pontos correspondem a onsets. Por este motivo, é também chamado de **função de detecção**.

Uma vez obtida a saída desta função, um pós-processamento para encontrar os picos deve já permitir que sejam conhecidos os instantes que estão sendo procurados.

3.1.3. Localização de picos

Se a função de detecção estiver bem projetada, os onsets, bem como outros transientes que surjam de forma abrupta, aparecerão como detalhes bem distintos, e facilmente localizáveis. É comum que os eventos procurados correspondam aos valores máximos locais observados, ou seja, os picos.

O processo de localização de picos, bem realizado, pode ser dividido em 3 etapas relevantes:

1. Pós-processamento – tratamento adicional dado à função de detecção que foi obtida com o processo de redução, que visa facilitar a localização dos onsets fazendo com que a função de redução possua maior uniformidade e consistência com relação à maneira que os eventos de surgimento de ataques nela aparecem. Pode ser um

processo de filtragem de ruídos (suavização), ou manipulações que tornem mais fácil determinar o limiar a partir do qual um instante é considerado de pico (correspondente a um onset), tal como normalizar a função, ou centralizá-la em torno de um certo valor (ajuste DC).

2. Determinação do limiar – é de se esperar que apareçam, no arquivo, picos que não estejam diretamente relacionados ao surgimento de ataques, devido a outros tipos de transientes mais sutis, que aparecem com alguma freqüência. Levando este fato em conta, é de extrema importância que o limiar de intensidade que distingue um pico causado por uma nota musical ou som percussivo entrando de outros eventos espúrios seja confiável.

3. Determinação efetiva dos picos – finalmente, quando o sinal tiver sido tratado de forma adequada e o limiar que permite distinguir um onset de quaisquer outros ruídos esteja definido da melhor maneira possível, basta encontrar os pontos de máximo que seguem o critério estabelecido, e a lista de instantes de tempo correspondentes aos eventos de onset estará pronta.

3.1.4. Fluxo do processamento de áudio

A fim de melhor esclarecer, vejamos uma na figura 5 uma ilustração mostrando a formacomo o áudio vai sendo transformado ao longo destas etapas.

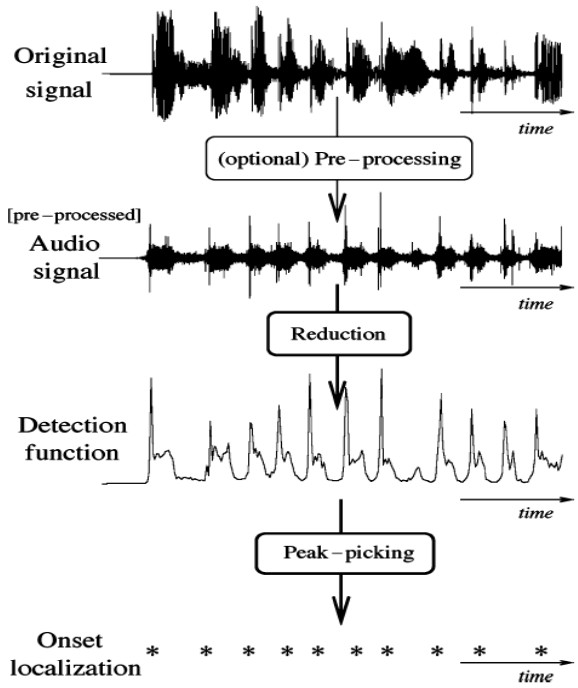

Fig. 5: O áudio sendo transformado ao longo das etapas do processo de detecção de onsets

3.2. Principais metodologias

Tendo em mente as etapas mais comuns de um algoritmo de detecção de onsets, já podemos fazer uma observação geral sobre as técnicas utilizadas nos algoritmos de ponta.

3.2.1. Funções de detecção

Vimos que, após um pré-processamento (opcional) do arquivo de áudio, o sinal vai passar por uma etapa chamada de redução, onde o conjunto de valores que antes representava ponto a ponto a evolução da oscilação sonora no tempo será reduzido a um conjunto menor, com resolução inferior, que expressa a probabilidade de um ataque estar iniciando em certa fatia de tempo.

Seguimos, então, algumas funções de detecção dentre as utilizadas com maior sucesso.

3.2.1.1. Fluxo espectral

Fluxo espectral é uma medida do quão rápido o espectro do sinal está mudando para um dado instante de tempo.

O espectro do sinal é o conjunto de valores das intensidades de energia para uma certa faixa de freqüências (bandas), dentre as múltiplas oscilações que estão embutidas no áudio. A obtenção deste espectro se dá por meio de uma operação chamada Transformada de Fourier, cujo cálculo não será explicado aqui.

O fluxo espectral pode ser obtido calculando-se a diferença na distribuição de energias ao longo das freqüências, de uma unidade de tempo para outra. Para este cálculo, o espectro é normalizado, e assim o resultado obtido acaba por ser independente da energia total.

Para este caso particular (servir de função de detecção de onsets), o fluxo espectral é restrito às mudanças positivas, e somado ao longo de todas as subdivisões consideradas do espectro. A fórmula utilizada é a seguinte:

$$SF(n) = \sum_{k=-\frac{N}{2}}^{\frac{N}{2}-1} H(|X(n,k)| - |X(n-1,k)|)$$

Onde $H(x) = \frac{x+|x|}{2}$ (função de retificação de meia-onda). *N* é o tamanho da janela de áudio que está sendo utilizada; *n* indica o índice atual desta janela. E *k* corresponde ao índice da banda de freqüência do espectro que está sendo contabilizada.

3.2.1.2. Desvio de fase

A fase é uma outra componente do sinal que podemos obter com a transformação de Fourier. Ela indica o quanto uma vibração está deslocada no tempo (ou espaço), em comparação com uma oscilação senoidal padrão.

A velocidade de mudança da fase para uma subdivisão do espectro de freqüências do sinal pode servir como estimativa da freqüência instantânea nesta subdivisão.

Um sinal $X(n,k)$ pode ser escrito em função de sua fase $\psi(n,k)$, com a utilização de notação de números complexos, ou seja:

$$X(n,k) = |X(n,k)|\, e^{j\psi(n,k)}$$

A fase $\psi(n,k)$ está no intervalo $-\pi < \psi(n,k) \leq \pi$. Desta forma, a freqüência instantânea é dada pela primeira diferença de fase $\psi'(n,k)$, dada por:

$$\psi'(n,k) = \psi(n,k) - \psi(n-1,k)$$

A medida de mudança na freqüência instantânea, indicadora de um possível onset da qual falamos, é dada pela segunda diferença de fase:

$$\psi''(n,k) = \psi'(n,k) - \psi'(n-1,k)$$

Seus valores também caem dentro do mesmo intervalo. Baseado então nesta medida, a função de detecção é construída a partir da média dos valores absolutos de variação de freqüência instantânea, para todas as divisões de espectro obtidas por meio Transformada de Fourier, o que minimiza as chances de perder uma detecção:

$$PD(n) = \frac{1}{N} \sum_{k=-\frac{N}{2}}^{\frac{N}{2}-1} |\psi''(n,k)|$$

3.2.1.3. Domínio complexo

Amplitude e fase do sinal podem ser analisados de forma conjugada, na busca de momentos em que o sinal se desvia da estabilidade. Isto é obtido calculando-se a amplitude e fase esperadas para a corrente região do espectro $X(n,k)$, baseado nos valores anteriores $X(n-1,k)$ e $X(n-2,k)$.

O valor alvo desejado $X_T(n,k)$ é estimado assumindo que a amplitude e a taxa de mudança de fase são constantes:

$$X_T(n,k) = |X(n-1,k)|\, e^{\psi(n-1,k)+\psi'(n-1,k)}$$

Então, definimos a função de detecção baseada em domínio complexo CD como sendo a soma de desvios absolutos em relação aos valores alvos:

$$CD(n) = \sum_{k=-\frac{N}{2}}^{\frac{N}{2}-1} |X(n,k) - X_T(n,k)|$$

3.2.1.4 Desvio de fase com pesos

Uma versão otimizada da função de detecção baseada em desvio de fase leva em conta a suscetibilidade da mesma a ruídos que existam no sinal cuja energia não é representativa.

Em outras palavras, o problema com a técnica que calcula o desvio de fase é que todas as k-ésimas fatias do espectro de freqüências são levadas em conta igualmente, em contraste com o fato de que a energia do sinal está distribuída com maior densidade entre as regiões de freqüência que contém partes dos tons soando no áudio, para um dado momento.

Diante disto, uma nova proposta de função de detecção de onsets foi feita, visando corrigir esta falha atribuindo pesos à a energia de cada região do espectro, com base em sua magnitude:

$$WPD(n) = \frac{1}{N} \sum_{k=-\frac{N}{2}}^{\frac{N}{2}-1} |X(n,k)\ \psi''(n,k)|$$

Este cálculo assemelha-se ao da função vista de domínio complexo, na qual a magnitude e a fase são consideradas conjuntamente. Todavia, a forma de combiná-las é diferente. Uma opção também é definir uma versão normalizada esta função, onde obtemos:

$$NWPD(n) = \frac{\sum_{k=-\frac{N}{2}}^{\frac{N}{2}-1} |X(n,k)\ \psi''(n,k)|}{\sum_{k=-\frac{N}{2}}^{\frac{N}{2}-1} |X(n,k)|}$$

3.2.1.5. Domínio complexo com retificação

A abordagem da função de detecção baseada em domínio complexo possui um problema: ela não faz distinção sobre quando a amplitude do sinal está variando de maneira crescente ou decrescente. Isto dificulta a distinção entre *onsets* e outros inícios de transientes na música. Uma versão melhorada desta função de detecção corrige este problema, fazendo isto da seguinte maneira:

$$RCD(n) = \sum_{k=-\frac{N}{2}}^{\frac{N}{2}-1} RCD(n,k)$$

Onde:

$$RCD(n,k) = \begin{cases} |X(n,k) - X_T(n,k)|, & \text{if} |X(n,k)| \geq \\ & |X(n-1,k)| \\ 0, & \text{otherwise} \end{cases}$$

Ou seja, a função de detecção tem seu valor diferente de zero apenas nos momentos onde o sinal teve o seu valor crescente.

3.2.2. Localização de picos

Sabemos que a função de detecção, recebendo como entrada o sinal de áudio, produz como saída um conjunto de valores de menor resolução que representa uma entidade a qual pode ser interpretada como a probabilidade de haver um ataque num dado instante.

Algumas funções de detecção dentre as de melhor qualidade foram vistas, contudo, para que sejam úteis é necessário que seus resultados sejam transformados nos eventos de onset de forma apropriada.

Para isto, o limiar da função de detecção necessita ser bem calibrado. E deve ser auto-ajustável ao longo da música, para que seja sensível, de forma coerente, às flutuações de média local da função.

Determinar este valor de corte, então, é uma tarefa de experimentação inerentemente experimental e dependente da parametrização dos experimentos. Isto tende a oscilar inclusive dependendo do tipo de música que está sendo trabalhada. Técnicas de aprendizagem podem ser muito úteis para fazer um ajuste fino dos parâmetros. Os melhores reconhecedores têm tudo isso embutido em seus cálculos para gerar as marcações de onsets.

4. Metodologias para Avaliação de Algoritmos

Existe uma vasta gama de técnicas e algoritmos para a detecção automática dos onsets em arquivos de áudio, e nós já explanamos algumas de grande importância, fornecendo uma visão geral a respeito do tema. Diante disto, surge naturalmente a pergunta: como fazer uma avaliação? Quanto uma técnica é melhor que a outra, e sob que condições estas diferenças se evidenciam? Como obter um "gabarito" com as marcações corretas das posições iniciais dos ataques? E como traduzir em números o desempenho de um algoritmo?

Tais questionamentos constituem o contexto do corrente capítulo.

4.1. Caracterizando onsets

Um dos primeiros passos quando se tem intenção de marcar um arquivo de áudio com seus instantes de onsets é fechar a definição do que será considerado como onset em si. Isto porque, conforme mencionamos, existe uma diferença entre o onset real – o momento onde a nota realmente começou a tocar – e o momento a partir do qual a nota se torna audível, e o ouvido humano percebe sua entrada.

Tudo fica mais simples quando se trabalha com sons monofônicos, isto é, apenas uma nota sendo tocada a cada vez no decorrer do tempo. No entanto, isto não é a realidade, na maioria dos casos.

Mesmo com apenas uma nota por vez, existem pertubações que podem tornar a tarefa mais árdua – como uma possível interferência de eco ou reverberação na gravação, ou ainda outros ruídos como a respiração do músico que ou sons espúrios emitidos pelo próprio instrumento.

Em múltiplas notas simultâneas, vem a questão da classificação de um acorde, por exemplo, como tendo vários ataques (respectivos a cada uma de suas notas), ou se será tomado como apenas um (se as notas forem suficientemente próximas). Ainda há também a chance de uma nota permanecer soando enquanto outras estão sendo atacadas, e isto também pode interferir no resultado.

Fatores como estes sugerem que a detecção dos onsets pode ser considerada, até um certo ponto, uma tarefa relativamente subjetiva, e por isto a precisão na especificação do que estamos procurando, em algum experimento, é item crucial.

4.2. Como marcar os onsets?

Como proceder para efetuar a marcação dos onsets em arquivos de áudio que vão servir como referência para testar a eficiência de algum algoritmo? Existem algumas técnicas padrão que aqui descreveremos.

4.2.1. Marcação usando instrumentos monitorados

Uma idéia bastante interessante para aqueles que disponham de recursos é o uso de instrumentos musicais que, por meio de algum dispositivo eletromecânico captam o instante exato em que uma nota musical foi executada. Em alguns instrumentos, como uma flauta ou um violino, isto pode ser tecnicamente inviável: é mais fácil construir o aparato adequado em algo como um piano ou órgão. Pianos monitorados por computador são, de fato, utilizados em alguns trabalhos.

Quando esta técnica é utilizada para marcar arquivos de áudio, não se pode esquecer da diferença entre os onsets verdadeiros e os percebidos. Dependendo do tipo de instrumento e da margem de erro considerada, esta diferença pode ser descartada.

De qualquer forma, ter os recursos para gerar marcação desta maneira pode ser dispendioso, além de restritivo quanto ao tipo de acervo de áudio com o qual se pode trabalhar. Portanto, em diversas situações o trabalho humano acaba sendo o meio mais adequado.

4.2.2. Marcação realizada com trabalho humano

Marcar manualmente os onsets num arquivo de áudio é uma tarefa que requer extremo esforço, demandando bastante tempo e concentração.

Para fazer uma marcação manualmente, destacamos aqui 3 técnicas:

- Plotagem do sinal – é observar o valor do sinal de áudio no decorrer do tempo, ou seja, a forma de onda original. Os ataques podem ser percebidos pelas regiões onde o sinal cresce de amplitude rapidamente, como é de se esperar. É uma técnica bastante eficiente, em particular para marcar a entrada de sons percussivos.

- Visualização do espectrograma – exibe como a intensidade sonora está distribuída, a cada instante, em diferentes faixas de freqüência. Ajuda a localizar os onsets globalmente. Uma característica bastante comum no aparecimento dos ataques é haver um sobressalto nas intensidades em toda a faixa de freqüências.

- Ouvir pedaços do sinal – Combinado com técnicas de visualização, permite ao indivíduo marcando o áudio atingir excelente eficiência e precisão.

Se o indivíduo que irá etiquetar os ataques tiver estas opções em mãos, mais alguma habilidade e nível de cuidado, poderá obter resultados bastante aceitáveis. Para tanto, é preciso contar com alguma ferramenta que disponibilize estas maneiras de “enxergar” o áudio.

4.2.3. Uma ferramenta de auxílio: Sound Onset Labelizer

Existe uma ferramenta que foi desenvolvida para facilitar a marcação de onsets em arquivos de áudio, utilizando as técnicas que foram citadas. É o Sound Onset Labelizer (etiquetador de onsets de som), uma ferramenta *open source* para ser usada no Matlab, desenvolvida referenciada pelo artigo *Methodology and tools for the evaluation of automatic onset detection algorithms in music*.

Este programa aumenta em muito a precisão e a eficiência de quem realiza a trabalhosa tarefa de marcar os onsets manualmente.

A tela do programa é dividida em 3 partes: na divisão superior, temos a representação do espectrograma do sinal; no meio, aparece a forma de onda em si no decorrer do tempo; embaixo, controles para uso do programa, como tocar um trecho de áudio ou marcar um onset.

O espectrograma e o gráfico da onda compartilham o mesmo eixo de tempo, e as operações de zoom atuam igualmente em ambas as partes. À direita do espectrograma temos um cursor que serve para regular o contraste de sua imagem, e à direita do gráfico do sinal podemos regular o tamanho com que a amplitude média aparecerá na tela.

Em testes feitos com o software, todos os 3 anotadores (os autores, considerados como experientes em ouvir) adotaram espontaneamente técnicas similares para marcar os onsets, seguindo estes passos:

1. aplicar zoom à janela, de maneira que o gráfico contenha algumas poucas notas (tipicamente, 1 a 2 segundos de som);
2. efetuar a marcação com baixa precisão, com o uso do espectrograma;
3. fazer o ajuste preciso, com o uso da opção "*autoplay*". Ela permite definir a posição de uma marcação imediatamente antes de um evento ocorrer.

Deve-se notar que estes passos foram determinados em senso comum, sem que instruções prévias sobre a anotação fossem dadas, exceto sobre a utilização da ferramenta em si. Esta sinergia sugere que o software realiza bem o propósito de facilitar a tarefa de maneira razoavelmente intuitiva.

4.2.4. Erro quando realizada marcação manual

Nos experimentos realizados pelos desenvolvedores do *Onset Audio Labelizer*, uma preocupação extremamente relevante foi mensurar qual o desvio que, em média, existe quando indivíduos – mesmo experientes – estão encarregados de marcar manualmente os onsets.

Como é de se esperar, há pequenas variações na marcação feita por um ser humano ou outro, pequenos erros que devem ser ponderados quando analisamos a qualidade de um algoritmo.

A tabela 1 mostra os resultados observados, para uma bateria de 17 arquivos, com 3 indivíduos etiquetando o início dos ataques, com o auxílio do Onset Audio Labelizer.

File #	Number of labelled onsets			Number of consistent onsets	Average timing difference
	1	2	3		
1	60	61	60	60	3.9 ms
2	38	38	46	33	13.6 ms
3	10	9	13	6	11.9 ms
4	25	25	26	25	2.5 ms
5	65	65	65	58	14.4 ms
6	79	79	79	78	7.2 ms
7	20	22	21	20	8.9 ms
8	58	58	58	58	7.7 ms
9	41	39	41	37	9.9 ms
10	20	20	20	19	7.0 ms
11	56	56	56	56	4.7 ms
12	62	62	66	59	9.9 ms
13	56	52	56	47	11.7 ms
14	61	54	52	53	9.0 ms
15	49	49	53	38	15.8 ms
16	12	12	12	4	28.4 ms
17	32	40	41	27	11.7 ms
Total	744	741	765	678	10.5 ms

Tab. 1: 3 anotadores realizando marcações, seus acertos e desvios do tempo exato que deveria ser marcado

Os arquivos de áudio são de gama bastante variada, e podemos notar, nos resultados, o quanto a qualidade da marcação dos anotadores é sensível ao estilo musical escolhido. Isto pode ser um tanto frustrante, dependendo do grau de precisão almejado. De qualquer forma, erro zero em medidas físicas é algo inexistente no mundo real, e o que devemos buscar sempre é um equilíbrio, envolvendo precisão e eficiência.

4.2.5. Sintetizando o próprio áudio

Outra alternativa para se obter arquivos com onsets etiquetados de forma segura é sintetizando os arquivos de áudio (usando um sintetizador MIDI), gerando os arquivos de forma que já se saiba onde estarão os ataques.

A desvantagem é que o áudio sintetizado tem, em geral, grandes diferenças em relação ao som natural de um instrumento. Sutis variações no timbre – que potencialmente modificariam os resultados – não costumam ocorrer com o som criado artificialmente. Além disso, há uma maior uniformidade no som artificial, e algumas diferenças e truncagens em relação ao real que, de certo, podem fazer com que uma análise de um algoritmo limitada a sons criados dessa maneira tenha resultados menos representativos.

4.3. Métricas de qualidade de algoritmos

Determinar um erro ou acerto de um reconhecimento de onsets não é uma tarefa difícil, se partirmos do princípio que já tenhamos conseguido o áudio com as marcações verdadeiras (confiáveis). Três são as situações possíveis:

1. Verdadeiro positivo (ou acerto) - existe um onset real na música e o software o encontra corretamente;

2. Falso negativo – existe um onset na música mas o programa não o encontrou;

3. Falso positivo – o programa encontra um onset em um instante que não corresponde a um onset real na música.

Estas medidas básicas são um bom ponto de partida, mas conjugá-las usando fórmulas apropriadas pode fornecer valores com maior expressividade. Uma ótima referência sobre métricas de avaliação é a competição de algoritmos de tratamento de áudio promovida pelo Laboratório Internacional de Extração de Informação Musical (*International Music Information Retrieval Systems Evaluation Laboratory* - IMIRSEL), o MIREX (*Music Information Retrieval Evaluation eXchange*).

O MIREX já está atualmente em sua terceira edição (2007), e dentre as competições promovidas, está incluída a comparação de algoritmos de localização de onsets.

Chamamos aqui VP para acertos (verdadeiro positivo), FP para falso positivo e FN para falso negativo. Para comparação dos participantes, o MIREX adota as seguintes medidas, que podem ser todas calculadas a partir da taxa do acertos e falsos positivos e negativos.

- Precisão: $P = VP / (VP + FP)$
- *Recall*: $R = VP / (VP + FN)$
- Medida-F: $F = 2*P*R/(P + R)$

5. Métodos / Experimentos

Em acordo com o que já foi exposto na introdução e definição do problema e escopo deste trabalho de graduação, vamos agora discorrer sobre a parte experimental do trabalho.

5.1. Metodologia utilizada

Antes de partir para a análise comparativa de um algoritmo de onset com os resultados obtidos no violão MIDI, julgamos importante ter a percepção e o sentimento relativo à tarefa de implementar um algoritmo para detecção de onsets. Com isto em mente, optamos por realizar a implementação de um método de reconhecimento bastante simples.

Construir um algoritmo com qualidade comparável ao que há de boa qualidade já existente nesta área seria inviável para um trabalho de graduação. As melhores propostas existentes demandam anos de pesquisa, entre elaborações de idéias, criação do código, testes, comparações e ajustes finos, e seria demasiadamente pretensioso buscar resultados assim para aqui apresentar.

Assim, um dos passos realizados neste trabalho, após ter estudado material bibliográfico suficiente, foi programar este software simplificado de reconhecimento dos onsets, para daí estar familiarizado com a essência da solução do problema e realizar alguns comentários que serão feitos aqui.

Uma vez que o conhecimento necessário já havia sido adquirido, partimos para a análise comparativa que mencionamos.

Apresentemos agora o curso destas experimentações e seus resultados.

5.1.1. Uma implementação simples de onset detection

O algoritmo criado segue as etapas básicas mencionadas anteriormente (na visão geral dos métodos para detectar onset).

A linguagem escolhida para implementação foi C++. As principais razões para esta escolha são a experiência já acumulada com a mesma, e o aproveitamento de bibliotecas escritas em C++ que foram gentilmente cedidas pela empresa D'Accord Music Software, para facilitarem operações como abertura de arquivo de áudio, carregamento em *buffers*, e demais detalhes técnicos inerentes à manipulação de deste tipo de arquivos.

5.1.1.1. Função de redução

A função escolhida para processar o sinal do áudio e gerar a correspondente saída (da qual pode-se extrair os onsets) foi bastante simples e intuitiva. O resultado que ela produz é obtido da forma que se segue.

Realiza-se primeiro a segmentação do arquivo em pequenos *buffers* (janelas), de tamanho parametrizável. Cada pequena janela destas conterá informação acumulada durante o período de tempo correspondente ao seu tamanho. O início de uma próxima janela pode ser ainda antes do final da janela anterior, havendo sobreposição (*overlapping*), de forma também parametrizável. O *overlapping* permite que as transições de resultados entre uma janela e outra sejam mais suaves.

O cálculo realizado com cada janela destas é a contabilização da energia acumulada do sinal, no intervalo de tempo correspondente ao tamanho de tal janela. A energia é calculada da seguinte forma:

$$E(n) = \frac{1}{N} \sum_{m=-\frac{N}{2}}^{\frac{N}{2}-1} [x(n+m)]^2 \, w(m).$$

Sendo N o número de amostras em cada janela, n o índice com a posição inicial da mesma e m o índice da amostra de áudio dentro da janela, a função w(m) serve para dar pesos diferentes a cada um dos pontos do sinal de áudio dentro do intervalo do somatório. No entanto, optamos por simplicidade, e utilizamos w(m) constante.

Com a medida de energia encontrada em cada janela, estamos já próximos ao final da etapa de redução.

De posse da energia ao longo do tempo, em cada janela do arquivo de áudio, a função de detecção de onsets segue um raciocínio trivial: o início dos ataques é marcado por um aumento súbito da energia; assim, devemos procurar pelos pontos onde ocorre uma abrupta variação de energia, ou seja, os máximos da derivada da mesma ao longo do tempo. Vejamos, em seguida, como tais pontos foram localizados.

5.1.1.2 Detecção de picos

A detecção de picos foi realizada utilizando um limiar adaptativo. Uma música geralmente possui variações em sua amplitude média ao longo de seu curso, seja por causa de picos locais muito intensos, variações na gravação, ou mesmo por característica do estilo tocado. Em resposta a este fato, uma opção razoável é definir o ponto de corte a partir de um conjunto de valores locais na função de detecção. Dentre os exemplos que vimos, adotamos a mediana, e então o limiar adaptativo $\tilde{\delta}(n)$ ficou assim calculado:

$$\tilde{\delta}(n) = \delta + \lambda \,\mathrm{median}\,\{|d(n-M)|, \ldots, |d(n+M)|\}$$

δ é o parâmetro de limiar referencial e é arbitrado com base em testes experimentais. M indica a faixa de valores locais que influenciarão no cálculo do de $\tilde{\delta}(n)$.

5.1.2 Avaliando a solução de algoritmos de onsets aplicada ao problema do violão

5.1.2.1 Grau de qualidade do estado da arte em reconhecimento de onsets

Conforme introduzimos no início deste relatório, uma das questões relevantes era mensurar o grau de confiabilidade que um algoritmo de boa qualidade de reconhecimento de onsets poderia atingir, para o caso do violão brasileiro.

O ponto de partida para responder esta pergunta, que foi consensuado com o orientador, foi valer-se de algoritmos que tivesse obtido bom desempenho na competição já citada do MIREX.

Entretanto, o que conseguimos encontrar disponível foram artigos acadêmicos especificando, de uma forma mais ou menos genérica, as técnicas que foram utilizadas na implementação. Decorrentes deste fato, 2 problemas: o grau de complexidade de determinadas técnicas era significativamente acima do desejado, e ainda que fôssemos programá-las, certamente teríamos resultados diferentes dos que eles obtiveram na competição, devido a especificidades de pormenores como parâmetros utilizados, etc.

Felizmente, contudo, encontramos uma boa alternativa. Um dos algoritmos que participava na competição – não de *onset detection,* mas sim de *beat tracking* – encontrava-se disponível, inclusive com partes do código fonte liberadas sob a licença GPL.

Em geral algoritmos que reconhecem o andamento das batidas das músicas utilizam internamente um reconhecedor de onsets, fato este já foi citado neste trabalho. E era o caso deste – ganhador do MIREX 2006, o BeatRoot versão 0.5.3, escrito por Simon Dixon.

A parte do código que realiza a localização dos ataques na música não possuía o código fonte acessível. Todavia, era fácil identificar no restante do programa (que tinha o código liberado) a parte onde era feita a chamada que criava os eventos de onset. Isto isolado, a alteração foi realizada, para que quando o programa encontrasse o andamento da música também gerasse uma saída com os onsets correspondentes da mesma.

Os resultados serão vistos e discutidos mais adiante.

5.1.2.2 Confiabilidade da captação do violão MIDI

Outro ponto importante levantado foi avaliar o grau de certeza nos dados de ataque que podem ser obtidos com o violão MIDI.

Apesar dos dados MIDI adquiridos serem gerados diretamente do hardware do captador, já é sabido que estes são suscetíveis a erros, como introduzir notas não desejadas. Mas aqui o nosso objeto principal de estudo é o tempo, sendo importante buscar uma medida ao menos aproximada do grau de confiança deste dispositivo no que diz respeito a esta variável.

Ainda que a precisão do hardware seja muito boa, existem outros itens de potencial influência no resultado, como a diferença entre o onset verdadeiro e a curva de ataque gerada pelo sintetizador MIDI, para o instrumento cujo som tiver sido escolhido como saída.

Um programinha simples foi construído, então, com o intuito de receber como entrada um arquivo MIDI e gerar uma saída indicando onde estão os onsets. No MIDI esta informação está codificada de maneira praticamente direta, bastando um cálculo em função da velocidade padrão programada para a execução do arquivo.

Mais à frente, veremos que foi obtido a título de comparação, com relação aos dados que tínhamos captados do violão MIDI.

5.1.2.3 Detalhes da marcação dos onsets

O software utilizado para realizar a etiquetagem dos onsets nas músicas foi o Sound Onset Labelizer já descrito no capítulo 4, seguindo o processo que foi mencionado – com a ressalva de que não pudemos contar com 3 pessoas para marcação.

Quanto aos onsets obtidos do MIDI, basta observar o instante de entrada de cada nota. No entanto, como a análise de múltiplos onsets mesclados em apenas um (*merge*) não foi implementada, arbitramos que havendo um acorde sendo executado (ou seja, múltiplas notas com instantes de tempo praticamente iguais), o tempo considerado seria o da primeira nota.

5.1.2.4 Considerações os dados e as comparações realizadas

Um problema que existe quando trabalhamos com o MIDI e o áudio é o alinhamento dos mesmo. Eles são criados a partir da mesma performance, mas mesmo assim a forma de gravação faz com que o MIDI não comece exatamente no mesmo instante que o arquivo de som (*wave).*

A solução escolhida para este problema foi alinhar os arquivos assumindo que o primeiro onset está correto, e daí partir para encontrar os desvios (se existentes) nos próximos. Esta técnica funciona razoavelmente bem se o a primeira detecção do MIDI ou do algoritmo não foi um falso positivo. No entanto, caso isto aconteça, é fácil descartar o experimento, pois percebe-se que os resultados se diferenciam com total disparidade da taxa de acerto média observada nos testes em que isto não ocorre.

As músicas selecionadas eram executadas por 2 músicos distintos.

Como gostaríamos de saber se a precisão do MIDI ou dos algoritmos de *onset* seria o bastante para fazer análise de *microtiming* (tempo com precisão de milisegundos), adotamos um limiar de erro de 15ms para mais ou para menos. Onsets fora deste limite de tempo serão considerados como falsos

positivos ou falsos negativos.

Também em função de nossa preocupação com precisão de tempo, em adição às medidas que são apresentadas no site do MIREX, calculamos, dentre os verdadeiros positivos, a média de erro do onset em relação à posição real dada pelo gabarito, bem como o erro máximo encontrado.

5.1.2.5 Resultados das comparações

Abaixo, os dados que obtivemos com os experimentos, nas tabelas 2 e 3.

Experimento	Duração	Positivos verdadeiros	Falsos positivos	Falsos negativos	Distância Máxima (ms)	Distância média (ms)	Precisão	*Recall*	Medida-F
1.		18	5	5	13,38	5,01	0,7826	0,7826	0.7826
2.		16	5	4	6,96	4,06	0,7619	0.8000	0.7804
3.		43	9	3	13,60	5,74	0,8269	0,9347	0.8775
4.		27	2	1	7,37	3,52	0,9310	0,9642	0.9473
5.		26	6	3	14,82	6,76	0,8125	0,8965	0.8524

Tab. 2: Resultados do Midi vs. Gabarito

Experimento	Duração	Positivos verdadeiros	Falsos positivos	Falsos negativos	Distância Máxima (ms)	Distância média (ms)	Precisão	*Recall*	Medida-F
1.		19	11	4	14,02	7,39	0,6333	0.8260	0.7169
2.		19	7	1	10,11	4,40	0,7307	0.9500	0.8260
3.		38	6	8	14,60	4,89	0,8636	0.8260	0.8444
4.		24	4	3	11,90	4,48	0,8620	0.8928	0.8771
5.		27	10	2	14,85	3,98	0,7297	0.9310	0.8181

Tab. 3: Resultados do Algoritmo BeatRoot vs. Gabarito

Analisando o erro no tempo quando há acerto, nota-se que o Midi se saiu melhor, de forma clara, nos experimentos 1, 2 e 4. Nos experimentos 3 e 5 houve vantagem do BeatRoot, entretanto sem nenhuma disparidade significativa.

Comentários a respeito destes resultados serão feitos no próximo capítulo.

6. Resultados e conclusões

Dentre os itens expostos como objetivos neste trabalho, um deles foi adquirir conhecimento, inclusive prático (com a implementação do algoritmo simples de testes, e da ferramenta de comparação) a respeito da detecção de onsets e diversos conceitos e temas relacionados, e o autor sente que este objetivo foi atingido com sucesso.

No que diz respeito à análise dos dados coletados, já esperávamos que os arquivos MIDI tivessem uma certa distância da perfeição. E levando em consideração o que já se conhece sobre gravações feitas com este tipo de captação, os falsos positivos verificados têm grande probabilidade de serem gerados devido a ruídos na captação, que faz o conversor registrar sons que não existem devido a algumas flutuações mais sutis.

Os falsos negativos, por sua vez, podem ter sido influenciados pela metodologia de avaliação: conforme dissemos, o comparador que contabiliza os acertos, falsos positivos e falsos negativos não trata os casos de múltiplos onsets que se fundem em uma única região de tempo, de modo que algumas notas ao serem tocadas simultaneamente podem ter acidentalmente caído fora do intervalo de tolerância, uma vez que é o onset da primeira o escolhido como representante do grupo.

Quanto ao desempenho do algoritmo que trata o áudio diretamente para inferir as posições dos ataques (BeatRoot), este nos surpreendeu. Apesar de perder para a qualidade do MIDI no resultado geral, chegou a superá-lo em alguns dos testes, e mostrou ter desempenho bastante aceitável.

A pesquisa e experimentação realizada foi de grande valia no âmbito didático, e ao mesmo tempo serviu como um bom ponto de partida para dar indicativos de respostas às questões levantadas.

7. Trabalhos Futuros

Alguns potenciais melhoramentos e aprofundamentos para oportunidades posteriores foram identificados, com base numa reflexão crítica sobre os resultados deste trabalho.

Seria uma boa melhoria definir uma metodologia mais robusta para fazer o alinhamento dos onsets, contornando de forma mais otimizada o problema das músicas não iniciarem exatamente ao mesmo tempo. Alinhar pelo primeiro onset, que foi a estratégia adotada, demonstrou bom comportamento quando o primeiro ataque encontrado é válido, no entanto forçava o descarte do experimento quando esta condição não era observada.

Esta experiência também pode alcançar graus de precisão e relevância estatística maiores se pudermos contar com mais pessoas fazendo a marcação dos onsets, de maneira a ter redundância e reduzir a chance erros. Num trabalho mais estendido, mais algoritmos também poderiam ser testados, bem como variações com diferentes limiares tolerados de imprecisão.

BIBLIOGRAFIA

Página do MIREX - http://www.music-ir.org/mirex2006/index.php/Audio_Onset_Detection

Pierre Laveau & Gäel Richard

Methodology and Tools for Evaluation of Automatic Onset Detection Algorithms in Music

Paul Brössier – The Audio Library at MIREX 2006

Dixon, Simon – Onset Detection Revisited

Alexander Lerch & Ingmar Klich - On the Evaluation of Automatic Onset Tracking Systems

Axel Röbel – Onset Detection in Poliphonic Signals by Means of Transient Peak Classification

Yunfeng Du & Ming Li & Jian Liu – Spectral flux based onset detection

Emir Kapanci and Avi Pfeffer – A hierarquical approach to Onset Detection

Juan Pablo Bello, Laurent Daudet, Samer Abdallah, Chris Duxbury, Mike Davies, and Mark B. Sandler – A tutorial on Onset Detection in Music Signals

Chris Duxbury, Mark Sandler, Mike Davies – A Hybrid Approach to Musical Note Onset Detection

Alexandre Lacoste & Douglas Eck – A Supervisied Classification Algorithm for Note Onset Detection