Detecção de Onsets em Notas de Músicas Instrumentais de Piano utilizando Representação Pitch e Aprendizado de Máquina

Luciana R. Costa[1], Frederico S. Pinagé[1], Gabriel M. Araujo[2], Eulanda M. Santos[1], Waldir S. S. Júnior[1]

[1]Universidade Federal do Amazonas (UFAM), AM-Brasil

[2]Centro Federal de Educação Tecnológica Celso Suckow da Fonseca (Cefet/RJ), RJ-Brasil

Emails: lucianarolim08@gmail.com, gabriel.araujo@cefet-rj.br, eulanda@icomp.ufam.edu.br, {fredericopinage, waldirjr}@ufam.edu.br

***Resumo*— *Onsets* são os instantes de tempo de início dos eventos musicais em sinal de música. A sua detecção automática serve de base para diversas aplicações tais como transcrição automática de música de um ou mais instrumentos musicais, alinhamento de áudio com *score* e estimação do tempo da música. Este artigo descreve um sistema de detecção automática de *onsets* em sinais musicais de piano usando a representação tempo-frequência *pitch* juntamente com classificadores SVM (*Support vector machine*), *Gradient boosting* ou CNN 1D (*One-dimensional convolutional neural network*). Os dois primeiros possuem desempenho similares no conjunto de dados BS1. O *Gradient boosting* apresenta maior sensibilidade, mas com mais detecções espurias.**

***Palavras-Chave*— Recuperação de informação, SVM, *boosting*, CNN 1D.**

***Abstract*— Onsets are the start time instants of musical events in a music signal. Its automatic detection serves as the basis for several applications such as automatic music transcription from one or more musical instruments, audio alignment with score, and music time estimation. This paper describes an automatic onsets detection system in piano music signals using the time-frequency representation pitch together with SVM (Support vector machine), Gradient boosting, or CNN 1D (One-dimensional convolutional neural network) classifiers. The first two have similar performance on the BS1 data set. Gradient boosting has higher sensitivity but with more spurious detections.**

***Keywords*— Information retrieval, SVM, *boosting*, CNN 1D.**

I. Introdução

A análise de sinais de música e a extração de informações musicalmente relevantes para construir aplicações musicais fazem parte do campo de pesquisa de recuperação de informação de música (MIR, *music information retrieval*). O MIR surgiu da necessidade de criar tecnologias que facilitassem o acesso, a recuperação e exploração de conteúdo digital musical [1], por conta da diversidade e grande quantidade de arquivos digitais de músicas disponíveis.

Alguns exemplos de sistemas desenvolvidos nessa área de pesquisa são o *Musixmatch*[1] e o *SoundHound*[2], que utilizam um ou mais critérios de similaridade para recuperar informações de músicas. A plataforma *Spotify*[3] que, dentre outras funções, sugere músicas baseado nas preferências do usuário e, por úÍltimo, sistemas de transcrição automática de música, como o *ScoreCloud*[4], que transcreve um sinal de áudio monofônico, com melodia cantada ou tocada por um único instrumento musical, em uma partitura digital.

Técnicas de aprendizado de máquina tem sido muito empregadas para resolver problemas de processamento digital de sinais de áudio e música [2] e as aplicações desses tipos de técnicas se estendem por vários subcampos de pesquisa explorados no MIR. Dentre elas, podemos citar a utilização de uma rede conjunta com *Residual Convolutional Block Attention Module* (Res-CBAM) para extrair a melodia contida em sinais de música polifônica [3]. O uso de um arquitetura de rede neural profunda *U-net* para transcrever a melodia de sinais de música do instrumento musical baixo [4]. E o emprego dos algoritmos de aprendizado de máquina kNN, SVM e a rede neural Perpectron Multicamadas [5], [6], para detectar gêneros musicais de forma a construir um jogo de geração de ritmos [7].

A detecção automática de *onsets* é uma tarefa MIR que consiste em detectar, de forma automática, o tempo de início de cada evento musical em um sinal de música [8], [9]. Geralmente é a etapa inicial de diversas aplicações como transcrição automática de música, alinhamento de música com *score*, estimação de tempo musical [10], dentre outros.

Diferentes algoritmos foram propostos ao longo dos anos para tratar do problema de detecção automática de *onsets* e as abordagens propostas podem ser divididas em três categorias [11]. A primeira é composta por abordagens que usam somente técnicas de processamento digital de sinais. Geralmente exploram características como a amplitude, fase e/ou frequência do sinal [12], [13]. A segunda é composta por abordagens que usam modelos ou dados estatísticos. Normalmente usam HMM (*hidden markov model*) ou PCA (*principal component analysis*) [14]–[17]. A última categoria utiliza aprendizado de máquina [18].

Quando se trata da representação do sinal de áudio, o espectrograma é a representação tempo-frequência comumente utilizada na construção de sistemas de detecção de *onset* [19]–[21]. O espectrograma mostra como as intensidades das componentes de frequências do sinal de música variam ao longo do tempo [22], logo, um coeficiente é associado a uma componente de frequência do sinal.

Uma representação mais compacta, no entanto, na qual um coeficiente é associado a uma banda de frequências do sinal, é uma opção alternativa que pode reduzir o custo computacional da tarefa de detecção automática de *onsets*. Por isso, propomos investigar a representação tempo-frequência *pitch*, que mede como a energia média quadrática em tempo curto em cada banda de frequência varia ao longo do tempo [23]. A representação *pitch* é composta por um número fixo de bandas de frequências, especificamente oitenta e oito.

Adicionalmente, outro fator observado na etapa de classificação dos sistemas de detecção automática de *onsets* do estado-da-arte, é o uso frequente da CNN como classificador, seja sozinha ou em conjunto com outros classificadores [19], [24], [25]. Porém, a estrutura de dados de entrada utilizada com mais frequência tem sido a 2D ou 3D. Com base nisso, propomos explorar a utilização da CNN 1D na tarefa de

[1]https://www.musixmatch.com/pt-br
[2]https://www.soundhound.com/soundhound
[3]https://www.spotify.com/br/
[4]https://scorecloud.com/portugues/

detecção automática de *onsets*.

Tendo em vista os fatores mencionados, neste artigo, apresentamos um sistema de detecção automática de *onsets* em sinais de música de piano usando aprendizado de máquina. No *framework* proposto, duas abordagens são consideradas. Na primeira, um processo de rotulagem manual de características é empregado para selecionar submatrizes na representação tempo-frequência *pitch*. Os classificadores investigados nessa abordagem são SVM e *Gradient boosting*. Na segunda abordagem, a combinação representação *pitch* e CNN 1D é explorada para detectar automaticamente os *onsets*, sem o processo de rotulagem manual de características.

O sistema possui três etapas (representação e extração de características, classificação e pós-processamento). O conjunto de dados BS1 foi utilizado assim como subconjuntos das bases de dados UMA [26] e MAESTRO [27]. Os resultados obtidos mostram que a CNN 1D e SVM obtiveram desempenhos similares quando avaliadas no conjunto BS1. No conjunto MAESTRO, a CNN 1D apresentou desempenho superior. O *Gradient boosting*, por sua vez, obteve os maiores valores de sensibilidade em ambas os conjuntos ao custo do incremento na quantidade de detecções espúrias.

As contribuições do artigo são:

- Investigar a representação tempo-frequência *pitch*, uma representação alternativa ao espectrograma, no contexto de detecção automática de *onsets*.
- Propor um framework para detecção automática de *onsets* no qual duas abordagens são investigadas:
  - Utilizando rotulagem manual de submatrizes *pitch* e os classificadores de aprendizado de máquina clássico SVM e GB.
  - Utilizando a CNN 1D, sem rotulagem manual das submatrizes *pitch*.

O artigo está organizado da seguinte forma: A seção II contém a descrição dos trabalhos relacionados ao sistema proposto, a seção III introduz o sistema proposto e as etapas nas quais foi dividido, o conjunto de dados utilizado, os procedimentos experimentais realizados e os resultados obtidos. Por último, a seção IV sumariza as conclusões e trabalhos futuros.

II. Trabalhos Relacionados

O artigo [24] investiga o uso de CNN 2D e 3D na transcrição automática de sons de bateria, incluindo a detecção de *onsets*. Os autores propuseram duas abordagens. Na primeira, uma CNN detecta *onsets* em sons contento chimbal, bumbo e tarola. Os instantes obtidos são utilizados na etapa de transcrição. A segunda emprega uma CNN para cada instrumento. Na primeira abordagem, o detector de *onsets* atingiu um *F-measure* de 93,50%, enquanto na segunda foi alcançado um valor 93,40%.

Em [25], a transcrição é feita pelo modelo *transition-aware*. O modelo é composto pelos ramos de transição de ataque, de transição completa e pela etapa de seleção de picos. Cada um dos ramos aplica a transformada Q constante no sinal de entrada, extrai as características do espectrograma CQT usando CNN 3D, faz uma modelagem dependente do tempo usando uma BiLSTM (*bidirectional long short-term memory*) e, por último, a detecção dos tempos dos *onsets* e a estimação de *frames* é feita por uma camada totalmente conectada. Os melhores desempenhos foram obeitos nos conjuntos MAPS e OMAPS, com *F-measures* de 87,52% e 88,21%, respectivamente, ultrapassando o desempenho do modelo *baseline high-resolution* por 4,35% e 10,31%. Nos conjuntos MAESTRO-v1 e MAESTRO-v2 [27], as *F-measures* obtidas foram 1,93% e 1,30% abaixo das obtidas pelo *high-resolution*.

A detecção automática de *onsets* usando redes de estado de eco (ESNs, *echo state networks*) foi proposta em [28]. A ESN é um tipo de rede neural recorrente (RNN). Nessa abordagem, um sinal mono é dividido em segmentos sobrepostos, denominados *frames*, usando uma janela *Hann*. Em seguida, a extração de características é realizada em cada *frame* a partir de uma STFT (*short-time fourier transform*), seguida por um banco de filtros triangular com frequência logarítmica e uma versão adaptada do algoritmo *spectral flux*. O vetor de características resultante alimenta a ESN, que retorna a função de detecção de *onset* (ODF, *onset detection function*). Por último, um algoritmo de seleção de picos baseado em limiar é aplicado sobre a ODF, retornando um resultado binário. Os valores de precisão e *F-measure* obtidos na abordagem proposta foram de $92,00\%$ e $88,60\%$, respectivamente.

O trabalho em [19] urtiliza uma arquitetura de rede neural unificada que prediz múltiplos estados de notas para fazer a transcrição automática de sinais polifônicos de piano. Dentre os estados de notas investigados estão *onset*, *off*, sustentação, *re-onset* e *offset*, dos quais cinco representações são definidas através da combinação desses estados. A arquitetura apresentada integra um modelo acústico com um modelo de linguagem musical (MLM, *musical language model*). Após a estimação dos estados, a sua inferência pode ser executada a partir de duas abordagens. A primeira utiliza um algoritmo guloso e a segunda um algoritmo de pequisa por feixe. Por último, uma regra simples é aplicada para determinar as notas. O sistema foi avaliado no conjunto MAESTRO v1.0.0 [27], que contém áudios polifônicos de piano. Além disso, um limiar de 50 ms foi usado para detectar os tempos dos *onsets*. O melhor resultado na detecção de *onsets* alcançou um *F-measure* de $94,67\%$.

Um transformador codificador-decodificador com decodificação padrão é empregado em [20] para fazer a transcrição automática de sons polifônicos de piano. O áudio de entrada é dividido em segmentos sobre os quais é calculado o espectrograma mel na escala logarítmica. O *frame* do espectrograma juntamente com a descrição simbólica correspondente é fornecido como entrada de uma rede. Em seguida, o codificador, que tem o papel de extrair as características de áudio musicalmente significativas, processa as entradas usando uma pilha de camadas de autoatenção e fornece, como resultado, uma sequência de associações. Essa sequência então é passada para a pilha de camadas do decodificador autoregressivo que aplica dois tipos de autoatenção, uma sobre a saída do codificador e outra sobre a saída do decodificador. A decodificação da saída do modelo é feita por um algoritmo guloso autoregressivo e a saída do modelo é uma distribuição *softmax* com um vocabulário de eventos discretos similares ao MIDI, composto pelos eventos *onset*, *offset*, *pitch* e velocidade. O modelo proposto obteve um *F-measure* de $95,95\%$ no conjunto MAESTRO [27].

III. Sistema Proposto

A. Construção de modelo para classificação

O projeto do modelo de classificação de características *pitch* está ilustrado na Figura 1. O primeiro bloco representa uma transformação $N(x(t))$. A transformação consiste em um conjunto de filtros que mapeia um sinal de áudio de entrada $x(t)$ em uma representação matricial de *pitch* $\mathcal{P}$. A matriz $\mathcal{P}$ é composta por coeficientes denominados de *short-time mean square power* (STMSP) e possui dimensões $L \times C$, onde L é um valor fixo que representa as oitenta e oito bandas de frequência contidas na representação *pitch* e C varia de acordo com a duração do sinal de áudio de entrada. A representação *pitch* delimita cada uma das bandas de frequência com base na frequência central das notas musicais contidas na escala igualmente-temperada de doze tons. Dessa forma, um acúmulo maior de energia em uma determinada banda de $\mathcal{P}$ pode indicar a presença da nota musical correspondente a esta banda.

O segundo bloco da Figura 1 representa a rotulação das coordenadas de referência. Esta etapa está detalhada na Figura 2, na qual seleciona-se coordenadas de coeficientes STMSP

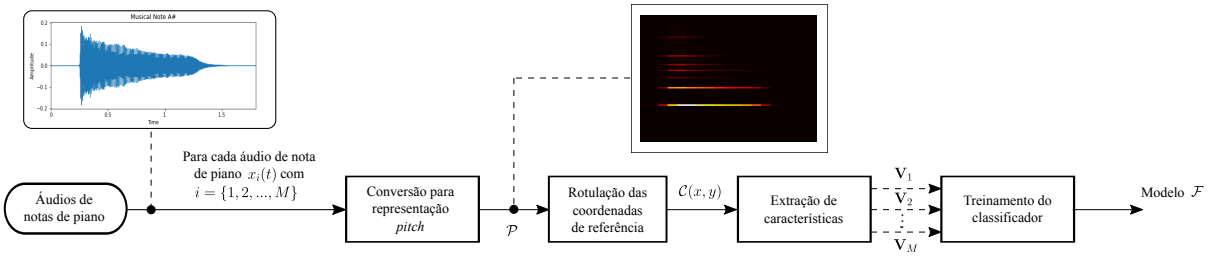

Fig. 1. Diagrama da etapa de construção de modelo para classificação do sistema de detecção de *onsets*.

contidos na representação *pitch* $\mathcal{P}$. Dois tipos de rotulações são realizadas. Na primeira, rotula-se coordenadas *onset* $\mathcal{O}$ que estão localizadas em regiões que indicam a presença de *onset*. Na segunda, rotula-se coordenadas não-*onset* $\mathcal{NO}$, situadas em regiões que indicam a ausência de *onsets*. A presença ou ausência de *onsets* é identificada de forma visual com base nas cores contidas no gráfico da representação *pitch*. Como resultado do processo de rotulação, os conjuntos de coordenadas de referência *onset* $\{\mathcal{O}_1, \mathcal{O}_2, ..., \mathcal{O}_M\}$ e não-*onset* $\{\mathcal{NO}_1, \mathcal{NO}_2, ..., \mathcal{NO}_M\}$ são gerados e formam o conjunto de coordenadas de referência final $\mathcal{C}(x, y)$.

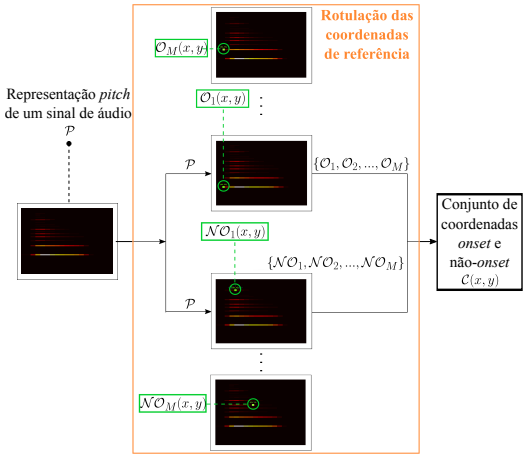

Fig. 2. Processo de rotulação das coordenadas de referência da etapa de construção de modelo para classificação.

O terceiro bloco da Figura 1 representa o processo de extração de características. Esse processo está detalhado na Figura 3. Primeiramente, define-se as dimensões das submatrizes $\mathcal{A}_{l\times c}$ que serão extraídas e tem como referência o conjunto de coordenadas $\mathcal{C}(x, y)$. Depois, as submatrizes são extraídas e, posteriormente, vetorizadas formando o conjunto de características de treinamento $\mathbf{V} = [\mathcal{V}_1, \mathcal{V}_2, \cdots, \mathcal{V}_i, \cdots, \mathcal{V}_M]$, onde o índice i representa o número de submatrizes extraídas e está dentro do intervalo $[1, 2, ..., M]$.

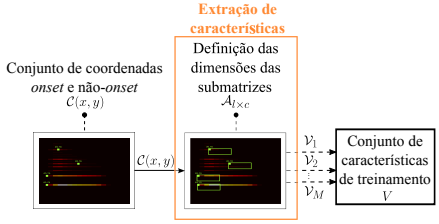

Fig. 3. Diagrama da extração de características da etapa de construção de modelo para classificação.

A última etapa consiste no treinamento do classificador, que recebe os vetores de características *pitch* $\mathbf{V}$ e gera um modelo $\mathcal{F}$. O modelo é capaz de mapear um ou mais vetores *pitch* em uma das classes: *onset* ou não-*onset*. O modelo $\mathcal{F}$ é obtido usando a SVM, *Gradient boosting* ou CNN 1D.

Na abordagem utilizada com a CNN 1D, a subetapa de rotulação de características, descrita anteriormente, não ocorre pois, após a conversão para a representação *pitch*, vetoriza-se o conjunto de matrizes $\mathcal{P}$ resultantes do bloco 1 e esses vetores de características *pitch* são fornecidos como entrada para treinar o classificador, no último bloco. Adicionalmente, os vetores de características são rotulados como *onset* ou não-*onset* verificando se dentro do intervalo de tempo do segmento de áudio há alguma marcação referência de *onset*.

B. Classificação dos vetores de características pitch

Similar a etapa de construção do modelo, na etapa de classificação, recebe-se janelas de áudio $w_i(t)$ de uma música instrumental de piano e gera-se na saída um conjunto de predições $\hat{y}$ feitas pelo classificador. O número de janelas i está dentro do intervalo $[1, 2, ..., K]$ e o tamanho do passo h, que define o intervalo de distância entre uma janela e a janela seguinte, é um parâmetro definido manualmente.

Na extração de características, recebe-se a representação *pitch* P da janela de áudio $w_i(t)$ para que as submatrizes $\mathcal{A}_{l\times c}$ sejam extraídas. As dimensões $l \times c$ das submatrizes são definidas previamente à varredura pela representação *pitch* P e as coordenadas de referência $\mathcal{C}(x, y)$ são identificadas após as delimitações das submatrizes. Cada submatriz extraída possui seus coeficientes STMSP com suas respectivas coordenadas de referência $c_{ij}(x, y)$. Associados aos elementos de referência, têm-se os tempos t_{ij}, que serão armazenadas para o pós-processamento. As submatrizes extraídas da representação *pitch* P de uma janela de áudio $w_i(t)$ são vetorizadas e formam o conjunto de características *pitch* $\mathcal{V}_i$ e, ao fim da extração de características de todas as janelas de áudio, obtém-se o conjunto de características de teste $[\mathcal{V}_1, \mathcal{V}_2, ..., \mathcal{V}_M]$.

Na abordagem utilizada com a CNN 1D, o sinal de áudio de entrada é dividido em segmentos menores, esses segmentos são convertidos para representação *pitch* e, em seguida, vetorizados. Por último, os vetores de características são classificados como *onset* ou não-*onset*.

C. Pós-processamento

O pós-processamento, a última etapa do sistema proposto, consiste em verificar quais tempos de *onsets* preditos estão dentro dos intervalos de tempo *ground truth*. Durante a etapa de classificação dos vetores de características *pitch*, cada submatriz extraída é convertida para um vetor de características *pitch*. Cada vetor é associado a uma coordenada de referência $c_{ij}(x, y)$ e um tempo t_{ij}, que corresponde, aproximadamente, ao instante de tempo do *onset*. Portanto, realiza-se uma busca para verificar se cada instante de tempo dos vetores de características preditos como *onset* está dentro de algum dos intervalos *ground truth* dos *onsets* $[t_{ref} - l; t_{ref} + l]$. O

t_{ref} representa o instante de tempo de *onset* referência, e a constante l, delimita a distância do intervalo de tempo, em milissegundos. A predição do classificador será considerada correta se o instante de tempo do vetor de características predito como *onset* estiver dentro de algum dos intervalos definidos. Caso contrário, a predição é considerada incorreta. Para definir o tamanho da janela de tolerância utilizada em cada sinal de áudio, vários tamanhos de janelas foram testados, considerando um intervalo máximo de 150 milissegundos. O pós-processamento encerra quando a busca pelos instantes de tempo de todos vetores de características preditos como *onset* tiver sido concluída.

Como a rotulação manual dos vetores de características não é realizada com a CNN 1D, os vetores de características não possuem instantes de tempo t_{ij} associados aos mesmos, dessa forma, a etapa de pós-processamento não é aplicada na abordagem com a CNN 1D.

D. Conjuntos de dados

Foram utilizadas 3 conjuntos: UMA, BS1 e MAESTRO.

O BS1 contém 49 arquivos de áudio de piano formado por dois subconjuntos. O primeiro (BS1-A) contém 30 arquivos de áudio obtidos de [29] e [30]. O segundo (BS1-B) é composto por 19 arquivos retirados do conjunto descrito em [11] (conhecido como *Bock*).

Foi utilizado um subconjunto do UMA composto por cem arquivos de áudio. Esses arquivos contém dez arquivos de cada um dos dez graus de polifonia existentes no conjunto.

O conjunto MAESTRO contém 129 arquivos de áudio. Neste trabalho foi utilizado a versão $V3.0.0$.

E. Procedimentos experimentais

Nos experimentos realizados com os classificadores SVM e *Gradient boosting* (parte A), o conjunto UMA foi utilizado para treinamento enquanto os conjuntos BS1 e MAESTRO foram utilizados para teste, separadamente. Em outras palavras, os experimentos da parte A executaram uma *cross-dataset validation*. Nos experimentos realizados com a CNN 1D (parte B), Os conjuntos BS1 e MAESTRO foram, separadamente divididos em treinamento como para validação e teste. A métrica utilizada foi o *Recall*, cuja fórmula está descrita em [31].

1) Procedimentos da parte A: Na parte A dos experimentos, para formar o conjunto de dados de treinamento, o subconjunto de arquivos de áudio da base UMA, descrito em III-D, foi utilizado. Na etapa de extração de características, duas rotulações manuais de regiões que indicam a presença de *onsets* e duas rotulações manuais de regiões que indicam a ausência de *onset* foram feitas por áudio, resultando em 200 vetores de características *pitch onset* e 200 vetores de características *pitch* não *onset*, totalizando 400 vetores ou observações no conjunto de treinamento.

Três configurações foram utilizadas para extrair as características. Na primeira configuração, as dimensões da matriz foram 3×3, com tamanho da janela de $0,5$ segundos e tamanho do passo de $0,5$ segundos. Na segunda configuração, as dimensões da matriz foram 4×4 e os valores dos outros dois parâmetros não se alteraram. Na terceira e última configuração, as dimensões da matriz foram 5×5 e os valores dos outros dois parâmetros permaneceram os mesmos.

Para avaliar os classificadores SVM e *Gradient boosting*, os conjuntos BS1 e MAESTRO foram utilizados separadamente. Em outras palavras, todos os arquivos do BS1 foram utilizados para formar o primeiro conjunto de teste assim como todos os arquivos do conjunto MAESTRO foram utilizados para compor o segundo. Na extração de características, as três configurações utilizadas nos dados de treinamento foram também utilizadas nos conjuntos de teste.

Na etapa de pós-processamento, o tamanho mais adequado da janela de tolerância, referente a cada arquivo de áudio, foi encontrado automaticamente testando vários tamanhos de janelas em um intervalo máximo de 150 milissegundos. O tamanho da janela com o menor número de FN (*false negatives*) foi o tamanho escolhido e utilizado no respectivo arquivo de áudio. Adicionalmente, a média de todos os tempos de *onsets* contidos dentro de cada janela de tolerância foi calculada e o tempo resultante desse cálculo foi o tempo de *onset* final considerado.

Ao todo, doze experimentos foram realizados na parte A. Os hiperparâmetros de cada um dos dois classificadores foram definidos utilizando a técnica de otimização *grid search* em cada um deles.

2) Procedimentos da parte B: Dois experimentos foram realizados com a CNN 1D. No primeiro, o conjunto BS1 foi utilizada para formar o conjunto de dados de treinamento, validação e teste. No segundo, o conjunto MAESTRO foi utilizada para formar os três subconjuntos.

Cada arquivo de áudio foi dividido em segmentos de $0,2$ segundos. Em seguida, as características *pitch* foram extraídas de cada segmento de áudio. Os vetores de características *pitch* resultantes foram dividido em dois subconjuntos, um para treinamento e outro para teste. As amostras utilizadas para formar o conjunto de treinamento não foram usadas para formar o conjunto de teste. Com relação ao BS1, 2000 vetores de características foram utilizados para treinamento e 580 vetores foram utilizados para avaliar a CNN 1D. Com relação ao MAESTRO, 2000 vetores de características foram utilizados para treinamento e 1345 vetores foram utilizados para testar o classificador. Do subconjunto de dados de treinamento, 20% foi utilizado para validação em ambos os casos.

Com relação aos parâmetros de treinamento, o número de épocas definido foi 100 e o *batch size* foi 32. O otimizador e função de perda utilizados foram *Adam* e *binary cross entropy*, respectivamente.

F. Resultados

Os resultados obtidos estão descritos na Tabela I, que mostra os valores de *recall* alcançados nos experimentos. Os resultados mostram, primeiramente, que o tamanho das matrizes utilizadas influenciaram no desempenho dos algoritmos GB e SVM, tanto no conjunto BS1 quanto no MAESTRO. O *recall* foi maior em todos os casos nos quais as dimensões da matriz utilizada foram 3×3 (configuração 1). A sensibilidade diminuiu na medida em que o tamanho da matriz aumentou para 4×4 e depois para 5×5 (configurações 2 e 3). A maior queda de sensibilidade ocorreu na transição da configuração 2 para a 3 no conjunto BS1.

Apesar do GB ter apresentado os melhores resultados de *recall*, uma investigação mais aprofundada nas predições feitas pelo classificador revelou que, principalmente no MAESTRO, o GB repetiu um mesmo padrão de detecção de *onsets* na maioria dos áudios. Em outras palavras, o algoritmo detectou *onsets* mesmo nas regiões sem a presença dos mesmos. Pelo fato de, na maoria dos casos, haver pelo menos um tempo de *onset* de referência próximo ao instante de tempo predito como *onset*, a janela de tolerância utilizada englobava o tempo de *onset* predito e, por consequência, uma alta quantidade de TPs foi computada. No conjunto BS1, houve alguns intervalos em que esse padrão de detecção se repetiu. Porém, detecções em locais distantes das regiões com *onsets* continuaram. De posse dessa informação, concluímos que o GB não aprendeu os padrões que caracterizam um *onset* e permitem identificá-lo.

Os resultados obtidos com CNN 1D foram parecidos com os resultados obtidos com SVM, utilizando a configuração 1 no conjunto BS1, com menos de 1% a mais no valor de *recall* obtido pela CNN 1D. Com o MAESTRO, ainda considerando

a configuração 1, a CNN 1D teve um desempenho consideravelmente melhor do que o SVM, apresentando $10,71\%$ a mais no valor de *recall*.

TABELA I
RESULTADOS OBTIDOS NA DETECÇÃO DE *onsets*.

Algoritmo	Configuração	Base de dados	*Recall* (%)
GB	1	BS1	96,24
GB	2	BS1	95,03
GB	3	BS1	76,07
SVM	1	BS1	85,97
SVM	2	BS1	83,50
SVM	3	BS1	65,60
GB	1	MAESTRO	96,10
GB	2	MAESTRO	93,52
GB	3	MAESTRO	91,62
SVM	1	MAESTRO	69,79
SVM	2	MAESTRO	69,43
SVM	3	MAESTRO	65,24
CNN 1D	—	BS1	86,42
CNN 1D	—	MAESTRO	80,50

Gradient boosting (GB).
Support vector machine (SVM).
Rede neural convolucional 1D (CNN 1D).

IV. CONCLUSÕES

Neste artigo, é proposto um *framework* para detecção automática de *onsets* em sinais de música de piano usando aprendizado de máquina. O *framework* foi utilizado de duas formas diferentes, na primeira abordagem, um processo de rotulação manual das características *pitch* foi aplicado e, para classificação, a SVM e GB foram investigados. Na segunda abordagem, a combinação representação *pitch* e CNN 1D foi explorada.

Os principais diferenciais entre as duas abordagens propostas e as abordagens dos trabalhos relacionados [19], [20], [24], [25], [28] estão na utilização da representação tempo-frequência *pitch*, uma representação mais compacta, invés do espectrograma, na utilização da estrutura de dados de entrada 1D na CNN invés da estrutura de dados 2D ou 3D e, por último, na exploração de um método de rotulação manual de características, que pode ser utilizado, principalmente, em casos nos quais o conjunto de dados disponível é pequeno.

Os melhores resultados foram obtidos com a configuração 1, com vetores de características de dimensão 9, o que acarreta em um menor custo computacional. O melhor resultado no conjunto BS1 foi obtido utilizando o classificador CNN 1D, com um *recall* de $86,42\%$. O SVM obteve um resultado levemente inferior. O GB também apresentou um alto recall, mas as custas de um elevado número de detecções errôneas. O melhor resultado no conjunto MAESTRO também foi obtido pela CNN 1D, com um *recall* de $80,50\%$. O GB também apresentou elevado número de detecções espúrias no MAESTRO. Uma das possíveis razões de os três classificadores terem apresentado melhor desempenho no conjunto BS1 é o fato da maioria dos arquivos de áudio contidos na base MAESTRO terem um volume relativamente mais baixo (quando comparados aos áudios do BS1). É possível que isso tenha impactado no número de detecções, principalmente utilizando o SVM.

Como trabalhos futuros pode-se explorar cenários com um maior volume de sinais de áudio, outras representações do sinal de som como, por exemplo, representações por subespaço ou representações *chroma*, podem ser investigadas. Além disso, outros modelos de CNN 1D podem ser explorados, pois os resultados foram promissores.

AGRADECIMENTOS

Parte dos resultados desta pesquisa foram subsidiados por ENVISION Indústria de Produtos Eletrônicos LTDA nos termos da Lei Brasileira Federal No. 8.387/91 (SUFRAMA).

REFERÊNCIAS

[1] M. Müller. *Information Retrieval for Music and Motion*. Springer, 2007.
[2] A. Lerch and P. Knees. Machine learning applied to music/audio signal processing. *Electronics*, 2021.
[3] Y. Chen and Y. Feng. Singing melody extraction based on joint network with res-cbam. In *Proc. Int. Conf. on Intelligent Computing and Signal Processing (ICSP)*, 2022.
[4] J. AbeBer and M. Müller. Jazz bass transcription using a u-net architecture. *Electronics*, 2021.
[5] W.S. Silva Jr, E.A.B da Silva, and S. Goldenstein. *Reconhecimento de Padrões utilizando Filtros de Correlação com Análise de Componentes Principais*. Tese de doutorado, UFRJ, 06 2010.
[6] J. Anderson, S.G.J. Luiz Carlos, et al. Reconhecimento de placas veiculares em cenários complexos utilizando o método do subespaço mútuo e redes neurais convolucionais. In *Proc. Simp. Brasileiro de Telecomunicações e Processamento de Sinais (SBrT)*, pages 1–5, 2022.
[7] E.A.L. Estolas, A.F.V. Malimban, J.T. Nicasio, et al. Automatic beatmap generating rhythm game using music information retrieval with machine learning for genre detection. In *Proc. Int. Conf. on HNICEM*, 2020.
[8] S. Dixon. Onset detection revisited. In *Proc. Int. Conf. on Digital Audio Effects (DAFx-06)*, 2006.
[9] J.P. Bello, L. Daudet, S. Abdallah, C. Duxbury, M. Davies, and M. B. Sandler. A tutorial on onset detection in music signals. In *IEEE Transactions on Speech and Audio Processing*, 2005.
[10] G. Morais, M. E. P. Davies, M. Queiroz, and M. Fuentes. Tempo vs. pitch: Understanding self-supervised tempo estimation. In *Proceedings of the 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023.
[11] S. Bock, F. Krebs, and M. Schedl. Evaluating the online capabilities of onset detection methods. In *Int. Society for Music Information Retrieval Conference (ISMIR)*, 2012.
[12] N. Silva, P. C. Weeraddana, and C. Fischione. On musical onset detection via the s-transform. In *Proceedings of the 52nd Asilomar Conference on Signals, Systems, and Computers*, 2018.
[13] K. Subramani, S. Sridhar, Rohit M. A., and P. Rao. Energy-weighted multi-band novelty functions for onset detection in piano music. In *Proceedings of the 2018 Twenty Fourth National Conference on Communications (NCC)*, 2018.
[14] E. Nakamura, E. Benetos, K. Yoshii, and S. Dixon. Towards complete polyphonic music transcription: Integrating multi-pitch detection and rhythm quantization. In *Proceedings of the 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018.
[15] W.S. Silva Jr, G. Araujo, E.A.B. da Silva, and S. Goldenstein. Facial fiducial points detection using discriminative filtering on principal components. In *Proc. IEEE Int. Conf. on Image Processing (ICIP)*, pages 2681 – 2684, 10 2010.
[16] G. Bernardo, S. Lincon, et al. A semi-supervised convolutional neural network based on subspace representation for image classification. *EURASIP Journal on Image and Video Processing*, 2020, 06 2020.
[17] G. Bernardo, F. Kazuhiro, et al. Advances in subspace learning and its applications. In *Pro. Conf. on Graphics, Patterns And Images (SIBGRAPI-EST)*, pages 35–41, 10 2021.
[18] C. Chuan and E. Chew. The effect of key and tempo on audio onset detection using machine learning techniques: A sensitivity analysis. In *Proc. Int. Symp. on Multimedia (ISM)*, 2006.
[19] T. Kwon, D. Jeong, and J. Nam. Polyphonic piano transcription using autoregressive multi-state note model. In *Int. Society for Music Information Retrieval Conference (ISMIR)*, 2020.
[20] C. Hawthorne, I. Simon, R. Swavely, E. Manilow, and J. Engel. Sequence-to-sequence piano transcription with transformers. In *Int. Society for Music Information Retrieval Conference (ISMIR)*, 2021.
[21] T. N. Magalhães and M. A. Loureiro. Training a convolutional neural network for note onset detection on the clarinet. In *18th Brazilian Symposium on Computer Music (SBCM)*, 2021.
[22] J. O. Smith III. *Mathematics of the Discrete Fourier Transform (DFT), with Audio Applications*. W3K, 2007.
[23] M. Müller and S. Ewert. Chroma toolbox: Matlab implementations for extracting variants of chroma-based audio features. In *12th International Society for Music Information Retrieval Conference (ISMIR)*, 2011.
[24] C. Jacques and A. Roebel. Automatic drum transcription with convolutional neural networks. In *Proc. Int. Conf. on Digital Audio Effects (DAFx)*, 2018.
[25] X. Wang, W. Xu, J. Liu, W. Yang, and W. Cheng. Transition-aware: A more robust approach for piano transcription. In *Proc. Int. Conf. on Digital Audio Effects (DAFx2021)*, 2021.
[26] A.M. Barbancho, I. Barbancho, L.J. Tardón, and E. Molina. *Database of Piano Chords: An Engineering View of Harmony*. Springer, 2013.
[27] C. Hawthorne, A. Stasyuk, A. Roberts, I. Simon, C. A. Huang, S. Dieleman, E. Elsen, J. Engel, and D. Eck. Enabling factorized piano music modeling and generation with the MAESTRO dataset. In *International Conference on Learning Representations*, 2019.
[28] P. Steiner, A. Jalalvand, S. Stone, and P. Birkholz. Feature engineering and stacked echo state networks for musical onset detection. In *Proc. Int. Conf. on Pattern Recognition (ICPR)*, 2021.
[29] Freesond team. Freesound. `https://freesound.org/`. 2022-09-01.
[30] Canton Becker. Sampleswap. `https://sampleswap.org/`. 2022-09-01.
[31] A. Mesaros, T. Heittola, and T. Virtanen. Metrics for polyphonic sound event detection. In *Applied Sciences*, 2016.