

Pengembangan Aplikasi Klasifikasi Suara Alat Musik Kalimba

Ferry Chai

Informatika, Fakultas Ilmu Komputer Dan Desain, Universitas Kalbis
Jalan Pulomas Selatan, Kav. 22 Jakarta 13210
¹Email: 2018103791@student.kalbisl.co.id

Abstract: This study aims to develop an application that implement deep learning with the Convolutional Neural Network (CNN) for classifying the sound of the kalimba and not kalimba. The application development in this research used the incremental method. In increment 1, the dataset will be cut into ten seconds and then converted into a mel-spectrogram image with the help of librosa. The test evaluation results from the experiments carried out were 98.33% accuracy, 0.0394 loss and 98% F1 score with 150 epochs of training. In increment 2, the model is implemented as a GUI with the help of TKinter. This study shows that CNN can be used to classify the sound of the kalimba.

Keywords: Kalimba Musical Instrument, Deep Learning, CNN, Mel-spectrogram, Classification, Incremental

Abstrak: Penelitian ini bertujuan mengembangkan aplikasi yang mengimplementasikan deep learning dengan metode Convolutional Neural Network (CNN) untuk klasifikasi suara alat musik kalimba dan bukan suara alat musik kalimba. Pembangunan aplikasi pada penelitian ini menggunakan metode inkremental. Pada inkremental satu, Dataset akan dipotong menjadi sepuluh detik lalu dikonversi menjadi bentuk gambar mel-spektogram dengan bantuan librosa. Hasil evaluasi pengujian dari percobaan yang dilakukan adalah accuracy sebesar 98,33%, loss sebesar 0,0394 dan F1 score sebesar 98% dengan pelatihan sebanyak 150 epoch. Pada inkremental dua, model diimplementasikan dalam bentuk GUI dengan bantuan TKinter. Dari hasil percobaan ini menunjukkan bahwa metode CNN dapat melakukan klasifikasi suara alat musik kalimba.

Kata kunci: Alat Musik Kalimba, Deep Learning, CNN, Mel-spektogram, Klasifikasi, Inkremental

I. PENDAHULUAN

A. Latar Belakang

Kalimba merupakan sebuah alat musik petik yang tergolong ke dalam jenis *lamellophone* (alat musik berlidah atau berpelat tipis) dimana cara memainkan alat musik ini adalah dengan menekan *tuts* sesuai dengan irama nada kunci lagu yang dimainkan [1]. Beberapa lagu atau musik yang dapat dimainkan dengan alat musik piano, dapat dimainkan dengan alat musik kalimba, sehingga sebutan lain untuk

alat musik kalimba adalah *thumbs piano* [2]. Bahkan ada yang mengira dengan mahir memainkan alat musik piano, maka dapat memainkan alat musik kalimba, padahal jelas berbeda cara memainkannya [3].

Di internet banyak aplikasi atau *software* yang menyediakan fitur bermain musik secara *online* seperti gitar, drum, piano dan lain sebagainya. Namun tidak banyak aplikasi atau *software* yang memperkenalkan suara alat musik kalimba. Hal ini terjadi karena mungkin belum ada parameter yang dapat membedakan alat musik kalimba dengan

alat musik lainnya. Untuk membuat parameter tersebut maka dibutuhkan sistem program *machine learning* yang dapat mengklasifikasikan suara alat musik, yaitu *Deep Learning*.

Sebelumnya pada tahun 2019, terdapat penelitian yang dilakukan oleh Danny Lionel dkk, untuk mengklasifikasi genre musik dengan *deep learning*. Penelitian tersebut berjudul “Klasifikasi Genre Musik Menggunakan Metode Deep Learning Convolutional Neural Network dan Mel-Spektogram” [4]. Sesuai dengan judulnya, penelitian tersebut menggunakan metode Mel-Spektogram dengan melakukan ekstraksi fitur untuk mengubah *audio file* menjadi suatu gambar, lalu melakukan klasifikasi dengan *neural network*, yaitu *Convolutional Neural Network* atau CNN. Oleh karena itu, topik penelitian ini adalah membangun aplikasi yang dapat mengklasifikasi suara alat musik kalimba dengan metode *deep learning*, yaitu CNN.

B. Rumusan Masalah

Bagaimana mengembangkan aplikasi yang dapat mengklasifikasi suara alat musik kalimba dengan model yang melakukan pembelajaran ekstraksi fitur dalam mengenali suara alat musik kalimba dengan algoritma CNN sehingga menghasilkan hasil uji yang terbaik.

C. Tujuan Penelitian

Mengembangkan aplikasi yang dapat mengklasifikasi suara alat musik kalimba dengan algoritma CNN sehingga dapat menentukan apakah input yang diberikan merupakan alat musik kalimba atau bukan.

II. METODE PENELITIAN

A. Metode Inkremental

Metode inkremental adalah salah satu metode pengembangan perangkat lunak yang bertujuan untuk meminimalisir kesalahan dalam proses pengembangan perangkat lunak [5]. Berikut adalah penjelasan tahapan-tahapan dalam metode inkremental [6]:

1. Analisis

Analisis sistem dilakukan untuk mengidentifikasi dan mengevaluasi permasalahan atau hambatan yang ingin diteliti, sehingga dapat menentukan keperluan dan data yang akan digunakan dalam pengembangan.

2. Desain

Desain berkonsentrasi pada bagaimana sistem dibangun untuk memenuhi kebutuhan pada fase analisis, sehingga dapat digunakan pada fase *coding*.

3. Coding

Tahap implementasi desain dengan bahasa pemrograman sehingga dapat dikenali oleh komputer. Hasil dari proses *coding* adalah berupa kode program.

4. Testing

Testing dilakukan untuk mengetahui apakah kode program yang dibuat bebas dari kesalahan sintaks ataupun logika dan juga untuk mengetahui apakah kode program sudah sesuai harapan dan mampu menyelesaikan masalah yang diidentifikasi sebelumnya di tahap analisa.

B. Convolutional Neural Network (CNN)

CNN merupakan salah satu jenis algoritma *deep learning* yang dapat menerima input berupa gambar, menentukan aspek atau objek apa saja dalam sebuah gambar yang bisa digunakan mesin untuk “belajar” mengenali gambar, dan membedakan

antara satu gambar dengan yang lainnya [7].

Sesuai dengan namanya *neural network*, arsitektur CNN menyerupai pola koneksi neuron atau sel saraf dalam otak manusia. CNN terinspirasi dari *Visual Cortex*, yaitu bagian pada otak yang bertugas untuk memproses informasi dalam bentuk visual. Arsitektur CNN terdiri dari 2 bagian, yaitu *Feature Extraction Layer* dan *Fully-Connected Layer* [8].

C. Mel-Frequency Cepstral Coefficient (MFCC)

Mel Frequency Cepstrum Coefficients (MFCC) merupakan suatu koefisien yang memiliki beberapa variabel di dalamnya, yaitu berisi frekuensi, amplitudo, *power*, dan intensitas pada suatu data *audio* yang diprediksi. MFCC menggunakan beberapa parameter yang akan berperan penting dalam menentukan tingkat keberhasilan pengenalan *signal* suara [9].

D. Librosa

Librosa adalah *package* yang berasal dari *Python* yang digunakan untuk analisis musik dan *audio*. *Librosa* biasa digunakan saat ingin melakukan analisis *audio* seperti *generate* musik ataupun *speech recognition*. *Librosa* juga membantu memvisualisasikan sinyal *audio* dan juga melakukan *feature extraction* yang di dalamnya menggunakan teknik *signal processing* [10].

III. HASIL DAN PEMBAHASAN

Proses penelitian ini berisi proses pengembangan perangkat lunak. Pada tahap ini, metode yang digunakan adalah metode inkremental. Metode inkremental yang dilakukan dibagi atas dua tahap, yaitu tahap pertama untuk membuat

pelatihan model JST dan tahap kedua untuk membuat perancangan *Graphical User Interface-nya* (GUI).

A. Pengumpulan Dataset

Dataset yang akan digunakan pada penelitian ini adalah *dataset* suara alat musik kalimba, piano, marimba, dan *recorder*, dengan durasi tiap data audio adalah 10 detik. Karena belum ada yang menyediakan *dataset* audio tersebut, maka data audionya akan bersumber dari video yang terdapat di internet, yaitu *Youtube*. Setelah video tersebut diunduh, lalu dikonversi menjadi format *.wav* dengan bantuan *converter audio online*. Selanjutnya data yang sudah berbentuk format *.wav*, data tersebut akan dipotong atau *trim* menjadi sepuluh detik. *Dataset* yang akan digunakan adalah video yang berjudul “*Relaxing Kalimba Music* (Tantri Silvian)” oleh Tantri Silvian [11].

B. Inkremental Satu

1. Analisis

Metode yang akan digunakan merubah suara audio menjadi bentuk gambar *spectral* adalah MFCC. Sehingga dengan metode MFCC ini, *dataset* yang mempunyai format awal *.wav* akan berubah menjadi bentuk gambar *spectral* atau bisa juga disebut dengan bentuk mel-spektogram. Jika sudah berbentuk mel-spektogram yang berupa citra, maka bisa dilanjutkan dengan pembuatan model dengan algoritma CNN, sehingga model yang dibuat akan mengklasifikasi suara alat musik berdasarkan representasi mel-spektogram yang dihasilkan pada masing-masing suara alat musik.

2. Desain

Desain untuk pembuatan model ini adalah menggunakan arsitektur yang

dimiliki oleh CNN. Dimana arsitektur CNN mempunyai banyak lapisan konvolusi, namun secara garis besar dibagi menjadi dua bagian, yaitu *feature extraction layer* dan *fully-connected layer*.

Karena penelitian ini melakukan klasifikasi berdasarkan citra mel-spektogram, maka algoritma CNN merupakan metode yang tepat untuk menjadi arsitektur pada model penelitian ini. Percobaan *training* yang dilakukan untuk mencari hasil model yang terbaik agar dapat diimplementasikan terhadap aplikasi yang dikembangkan adalah sebanyak empat kali percobaan, yaitu *training* dengan *epoch* 50, 100, 150 dan 200.

3. Coding

Dataset pada penelitian ini terbagi menjadi dua *class*, yaitu suara alat musik kalimba dan suara yang bukan alat musik kalimba, dimana suara yang bukan alat musik kalimba adalah suara alat musik piano, marimba, dan *recorder*. Jumlah *dataset* yang akan diuji untuk masing-masing *class* adalah 100 data, sehingga banyak data yang diuji adalah 200 data. Setelah mengumpulkan *dataset*, hal pertama yang perlu dilakukan adalah *dataset*-nya diolah terlebih dahulu dengan cara memotong durasi data menjadi sepuluh detik lalu dikonversi dalam bentuk gambar mel-spektogram, dan menghasilkan tiga data baru. Sehingga jumlah *dataset* nya pun jadi bertambah, dari yang awalnya berjumlah 200 data bertambah menjadi 600 data. Lalu *dataset* dibagi menjadi dua bagian, yaitu data *train* dan data *test* dengan perbandingan 80 : 20. Sehingga jumlah data *train* adalah 480 data, dan jumlah data *test* adalah 120 data. Setelah *split data*, maka selanjutnya adalah melakukan proses *training* dengan melakukan empat kali percobaan dengan

percobaan yang telah disebutkan sebelumnya.

4. Testing

Setelah melakukan *training*, Hasil *accuracy*, *loss*, dan *F1 score* tiap *training* akan divisualisasikan dalam bentuk grafik dengan bantuan *library* dari *Tensorboard*. Di dalam terdapat dua garis, yaitu garis kuning yang menunjukkan data *training* dan garis biru yang menunjukkan data *testing*. Berikut adalah hasil evaluasi dari empat percobaan yang telah dilakukan:

Tabel 1 Hasil Evaluasi Model

No	Jumlah Epoch	Accuracy	Loss	F1 Score
1.	50	50%	0,83	0,49
2.	100	86,6%	0,281	0,86
3.	150	98,3%	0,0394	0,98
4.	200	99,16%	0,0205	0,99

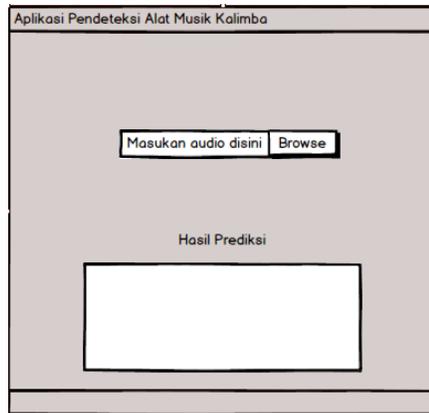
Percobaan yang akan digunakan untuk menghasilkan model yang terbaik dan efektif adalah percobaan ketiga, karena dengan komputasi *training* yang lebih sedikit dari percobaan keempat, dapat menghasilkan evaluasi yang tidak terlalu jauh dengan percobaan keempat.

C. Inkremental Dua

1. Analisis

Setelah melakukan pembuatan model klasifikasi alat musik kalimba, Pada tahap pembuatan GUI, *platform* yang digunakan berbeda dengan yang digunakan pada saat inkremental satu, yaitu *Visual Studio Code*. *Library* yang dibutuhkan untuk membuat GUI adalah *library* TKinter.

2. Desain



Gambar 1. *Mock Up* Aplikasi

Gambar di atas merupakan *mock up* desain aplikasi yang telah direncanakan, Dengan *window* sederhana, lalu ditambahkan komponen untuk memasukan audio yang dimasukan, tombol *button* untuk memproses klasifikasi audio yang dimasukan, dan kolom yang menampilkan hasil dari data audio yang telah diprediksi.

3. Coding

Data *audio* yang dimasukan harus mempunyai ekstensi *.wav*. Setelah dimasukan kedalam GUI, data *audio* tersebut akan diproses seperti yang dilakukan di incremental satu sebelumnya, lalu data tersebut akan diprediksi dengan model yang telah dibuat. Setelah diprediksi, hasilnya pun akan ditampilkan ke GUI.

Untuk ukuran *window* nya adalah 600 x 600 piksel dan ukurannya tidak dapat berubah-ubah atau *resizeable*. Selanjutnya *load* model yang telah kita buat pada inkremen satu agar dapat digunakan untuk memprediksi data *audio* yang dimasukan. Fungsi *LabelFrame* pada *TKinter* adalah untuk menampilkan judul GUI yang dibuat. Beberapa fungsi *Label* dari *library*

TKinter yang berguna untuk menampilkan teks dalam GUI, Contohnya adalah mengarahkan *user* memasukan data *audio* atau menampilkan hasil prediksinya.

4. Testing

Pengujian yang dilakukan untuk menguji GUI adalah *white box* dan *black box*. Pengujian tersebut digunakan untuk menguji apakah data *audio* yang dimasukan oleh *user* dapat diproses dan diprediksi, serta menguji apakah fungsi komponen GUInya dapat berjalan sesuai keinginan atau tidak.

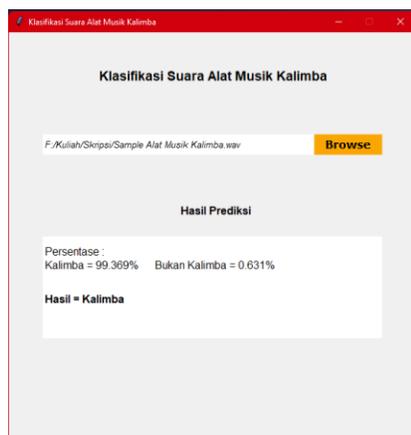
Tabel 2 Hasil Pengujian *White Box*

No	Pengujian	Skenario Pengujian	Hasil
1	Input file audio	Memasukan satu file audio yang ingin diprediksi	Berhasil
2	Slicing	Memotong durasi audio menjadi sepuluh detik	Berhasil
4	Ekstraksi Data	Data diekstraksi menjadi bentuk mel-spektoqram	Berhasil
5	Proses Prediksi	Prediksi input yang telah dimasukan dengan model CNN yang telah dibuat	Berhasil
6	Hasil Prediksi	Menampilkan hasil prediksi model	Berhasil

Tabel 3 Hasil Pengujian *Black Box*

No	Pengujian	Skenario Pengujian	Hasil
1	Fungsi input file audio	Memasukan satu file audio yang ingin diprediksi	Berhasil
2	Kolom menampilkan hasil prediksi	Menampilkan hasil prediksi dari model klasifikasi yang telah dibuat	Berhasil

Berikut tabel diatas merupakan hasil pengujian *white box* dan *black box* yang dilakukan. Aplikasi akan diuji dengan memasukan data audio alat musik kalimba dan bukan alat musik kalimba. Hasil yang didapatkan dari pengujian aplikasi tersebut adalah 99,369% untuk data audio alat musik kalimba dan 93,489% untuk data audio bukan alat musik kalimba.



Gambar 2. Hasil Pengujian Aplikasi

IV. SIMPULAN

Pengembangan aplikasi klasifikasi suara alat musik kalimba dengan metode CNN dapat melakukan klasifikasi suara alat musik kalimba dengan mengkonversi data *audio* tersebut menjadi gambar mel-spektrogram atau MFCC, Sehingga dapat membedakan antara suara alat musik kalimba dengan alat musik melodis yang lain. Percobaan yang digunakan untuk memperoleh model terbaik adalah 150 *epoch* dengan nilai *accuracy* sebesar 98,3%, nilai *loss* sebesar 0,0394 dan nilai *F1 Score* sebesar 0,98.

DAFTAR RUJUKAN

- [1] S. Rahman, "Pemanfaatan Batok Kelapa Sebagai Media Pembuatan Bio-Instrumen Musik," *Besaung J. Seni Desain dan Budaya*, vol. 5, no. 2, pp. 135–140, 2020, doi: 10.36982/jsdb.v5i4.1178.
- [2] "Are Kalimba Notes The Same As Piano? – Kalimba HQ." <https://kalimbahq.com/are-kalimba-notes-the-same-as-piano/> (accessed

- Feb. 15, 2022).
- [3] "How does the Kalimba Relate to the Piano? - Blog, Item, News and Announcements - Kalimba Magic." <https://www.kalimbamagic.com/blog/item/how-does-the-kalimba-relate-to-the-piano> (accessed Aug. 03, 2022).
- [4] D. Lionel, R. Adipranata, and E. Setyati, "Klasifikasi Genre Musik Menggunakan Metode Deep Learning Convolutional Neural Network dan Mel-Spektrogram," *J. Infra Petra*, vol. 7, no. 1, pp. 51–55, 2019, [Online]. Available: <http://publication.petra.ac.id/index.php/teknik-informatika/article/view/8044>.
- [5] Roger S. Pressman, *Software Engineering: A Practitioner's Approach*. 2010.
- [6] Abdul Khadir, "Sistem Pendukung Keputusan Feasibility Study Untuk Menilai Kelayakan Sebuah Bisnis," *Sist. Pendukung Keputusan*, vol. 8, no. 1, pp. 1–7, 2017.
- [7] "Memahami Convolutional Neural Networks dengan TensorFlow." <https://algoritma.blog/convolutional-neural-networks-tensorflow/> (accessed Mar. 06, 2022).
- [8] "Apa itu Convolutional Neural Network? | by QOLBIYATUL LINA | Medium." <https://medium.com/@16611110/apa-itu-convolutional-neural-network-836f70b193a4> (accessed Mar. 06, 2022).
- [9] A. D. Adriana, "Perangkat Lunak Untuk Membuka Aplikasi Pada Komputer Dengan Perintah Suara Menggunakan Metode Mel Frequency Cepstrum Coefficients," *Komputa J. Ilm. Komput. dan Inform.*, vol. 2, no. 1, pp. 21–26, 2013, doi: 10.34010/komputa.v2i1.76.
- [10] "How to install Librosa Library in Python? - GeeksforGeeks." <https://www.geeksforgeeks.org/how-to-install-librosa-library-in-python/> (accessed Jun. 14, 2022).
- [11] "(33) Relaxing Kalimba Music (Tantri Silvian) - YouTube." <https://www.youtube.com/watch?v=qtBb9FBVGbo&t=17s> (accessed Mar. 14, 2022).