TRIBHUVAN UNIVERSITY
INSTITUTE OF ENGINEERING

Kathmandu Engineering College
Department of Computer Engineering

Major Project Proposal
On

Composition of music from music sheets using CNN

[Code No: CT 707]

By:
Rojan Manandhar - KAT074BCT050
Sajish Shrestha - KAT074BCT057
Salil Rajkarnikar - KAT074BCT058
Sandesh Raj Shakya - KAT074BCT062

Kathmandu, Nepal
2078, Jestha

ACKNOWLEDGEMENT

Before all else, we are grateful to our teachers for providing us the opportunity to develop our major project. We would like to thank the **Department of Computer Engineering** for helping us in this report. We would also like to thank the faculty members **Er. Sudeep Shakya, Head of Department of Computer Engineering and Er. Kunjan Amatya, Deputy Head of Department of Computer Engineering** for providing us the required guidance that we need for this project. We are extremely indebted to **Er. Sapana Thakulla** for the necessary templates and documents which has helped us a lot in the completion of our project.

We would also like to thank all our friends and family for the assistance they provided during this report. We hope all their help and support can result in a project that all of us can be proud of and can inherit many new skills in the field of programming. Finally, we would like to give our sincere gratitude towards anyone who helped us directly or indirectly to finalize this report.

ABSTRACT

Music is one of the most beautiful creations of humanity. Everyone loves melodious music and our program helps the individuals to get comfortable with the music tones. The individuals can use the real world object(music sheets) which is converted into suitable readable format for the computer to manipulate it and play music tones based on the content in the sheet which can be played on different virtual instruments. Musical sheets from the past that don't have a machine-readable format can be converted to a format that can be read by other programs as well. This can be done using a deep learning approach which includes a Convolutional Neural Network(CNN). The aim is to then compose the music contained in the music sheet.

Keywords: Music Sheet, deep learning, machine-readable format, CNN.

TABLE CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

AI	Artificial Intelligence
CNN	Convolutional Neural Network
ConvNet	Convolutional Neural Network
MEI	Music Encoding Initiative
MIDI	Musical Instrument Digital Interface
MusicXML	Music Extensible Markup Language
ReLU	Rectified Linear Unit
RNN	Recurrent Neural Network

CHAPTER ONE: INTRODUCTION

1.1 BACKGROUND THEORY

Music, art of sound, is something everyone adores. Musical signs, music scores such as staff, brace etc. are fascinating and really fun to watch even if most of us can't understand them. This program that we will be developing will help everyone to scan those musical sheets and get them to know what those sound like.

"Composition of music from music sheets using CNN" is able to scan musical sheets and analyze the signs using AI/Convolution Neural Network. This program will be implemented using cameras in our phone. It will scan musical sheets, analyze it and produce a machine-readable format, then virtually play it in sound of the instrument chosen by the user. This program can help musical composers and be fun to use for music lovers to try and understand musical sheets.

"Composition of music from music sheets using CNN" scans musical sheets and is also able to simulate it in different musical instrument's sounds. Users use their camera to scan the musical notes which will then be analyzed by program and get it ready to be converted into sound. It then allows the user to select the instrument in which the sound is to be played. This program then plays it virtually in the chosen instrument's version. This can be helpful to music composers as it allows them to save time and analyze the music they've created.

This app will help music composers to analyze the music they've created which will save their time and money. This app can also be fun to play with as we all love music, scanning musical sheets and listening to what they sound like will be really fun to listen to and a great way of learning musical instruments.

1.2 PROBLEM STATEMENT

We all love music and have tried to play music in one form or another, strumming guitar strings even though we know nothing about guitars, banging on desks/tables, humming etc. Music composers have a hard time checking out the music they have composed. Trying out the composed music in every instrument to find out if the music has been composed properly and sounds good in every instrument required is a difficult and time consuming task. There is a requirement and demand of programs to simulate musical instruments and those available in the market are incomplete and too expensive to be used by all. AI will help analyze and help simulate the composed music in different musical instrument's sounds.

1.3 OBJECTIVES

- To develop a program that helps to detect the music note in the music sheet.
- To make the audible sound to be played on the virtual instrument tone.
- To produce a format of the music that can be reused later on.

1.4 SCOPE OF PROJECT

This program can be used by the musicians who are experimenting with creating a new music album or people who are new to music and want to learn about the types of music notes, music tones, etc. According to research, this application will help the young people mostly as they are the ones who go through the learning phase. Composing music is a complex part without knowing how to manipulate or combine the notes, so this program will help to overcome the problem.

1.5 APPLICATIONS

- It helps to stimulate music.
- It allows us to analyze music sheets/scores.
- It helps to learn instruments with ease.

CHAPTER TWO: LITERATURE REVIEW

There are a handful of music composition apps available but there are very few that can scan the music sheet to play music sheet or support music with grand staff (staff including both treble clef and bass clef).

2.1 Playscore Pro

PlayScore Pro is a sheet music scanner that plays all kinds of music directly from photos, images and PDF. The PlayScore Pro sheet music reader will play it right back to you, scrolling through the song, following measure by measure.[1]

Following are the key features of Playscore Pro:

- It can scan and play music directly from photos, images and PDF

Although Playscore Pro is a well-developed music scanner, it has some limitations:

- It requires a high-end camera and/or high quality images
- It can only scan one staff of music at a time
- It cannot manually compose music
- It charges upto $5.99 a month

2.2 Guitar Pro

Guitar Pro is a multitrack editor of guitar and bass tablature and musical scores , possessing a built-in MIDI-editor, a plotter of chords, a player, a metronome and other tools for musicians. It has versions for Windows and MAC-OS (Intel processors only) and is written by the French company Arobas Music.Guitar Pro was initially designed as a tablature editor, but has since evolved into a full-fledged score writer including support for many musical instruments other than guitar.[2]

Following are the key features of Guitar Pro:

- Read music score and tabs.
- It allows easy music score edition.
- Provides tools for composing music.
- Allows you to print and share your files.

Following are the limitations of Guitar Pro:

- It cannot scan and read music directly from photos, images and PDF .
- It does not support grand staff as it was primarily designed for Guitar.
- Its pricing starts at $69.95 per user, as a one-time payment. They do not have a free version.

2.3 Scan Score

ScanScore is an application that we had not heard of at all until I connected with them earlier this year. ScanScore is a Windows-only application that lets you scan in sheet music (either with a smartphone or tablet or a conventional scanner) to easily listen to or edit your scores.

With ScanScore you can not only quickly get your paper sheet music into digital format, but you can also clean up older, hard-to-read scores, transpose your part automatically, fix misprints, and even have your music played back to you. Additionally, ScanScore can export your music files into almost any music program for further editing and layout adjustments. If you already have your music digitized, you can also import PDF files![3]

Following are the key features of scan score:

- Scan printed and digital sheet music
- Play back your score
- Make edits and corrections to the notes
- Transpose notes, staves or the whole score
- Export your score for further processing in another program

Following are the limitations of Scan score:

- The full version isn't cheap, but most musicians can get by with the Ensemble version
- Limited editing options mean you really should have another program if you want to get the parts "just right".
- Windows-only (no Mac or Linux option)

CHAPTER THREE: METHODOLOGY

3.1 PROCESS MODEL

"Composition of music from music sheets using CNN" will be developed using the incremental model. Incremental model is a process of software development where requirements are broken down into multiple standalone modules of the software development cycle. Incremental development is done in steps from analysis, design, implementation, testing/verification, maintenance. This model will help us to achieve the goal as we will need to keep analyzing the program and bring changes to the problem and we also need to add the features which might be necessary to make the program appropriate and working functionally.

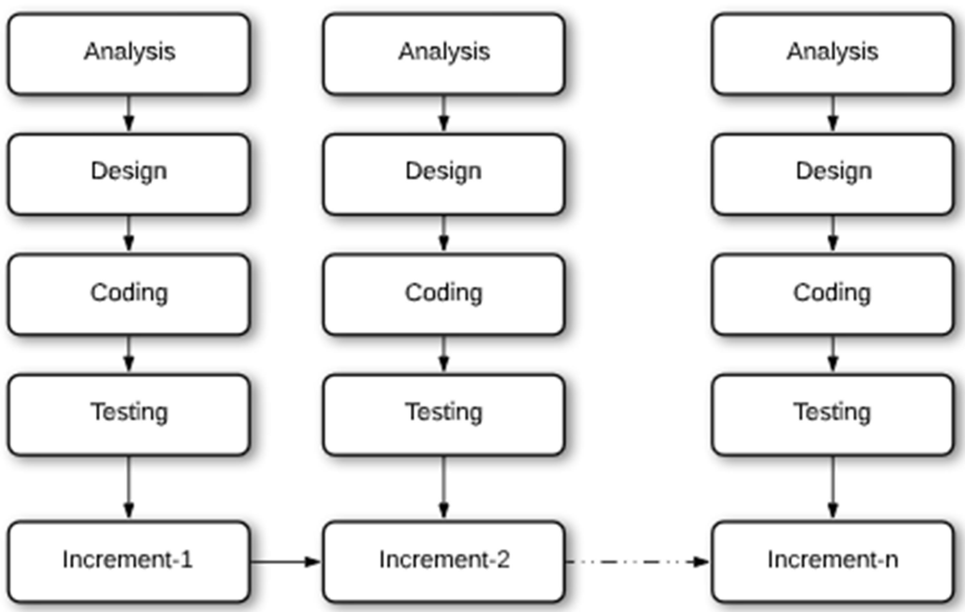

Fig. 3.1. Incremental Model

3.2 BLOCK DIAGRAM

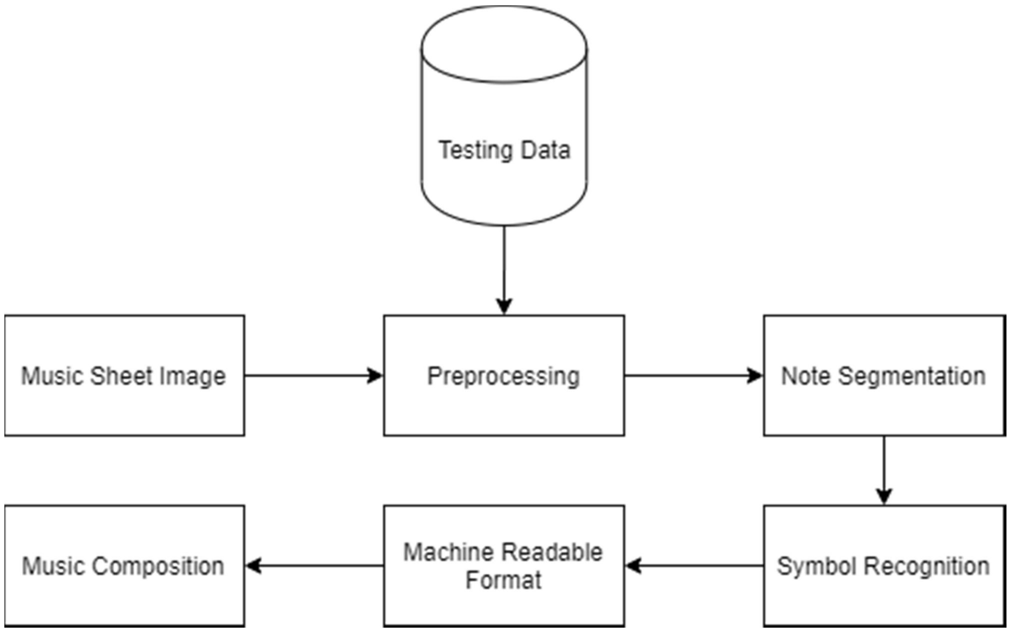

Fig. 3.2. System Block Diagram

This program takes an image as an input which contains the music sheet. The task of the program is to convert an image of a music sheet into a machine-readable format, such as MIDI, MEI or MusicXML which can then be used to compose the music contained. After preprocessing and note segmentation, we use a CNN to classify each music symbol. The steps of the process are described below.

3.2.1 Preprocessing

The first step is to preprocess the image in order to make the subsequent analysis easier. Since we are dealing with photos taken with a camera, it often comes with some types of distortions and noises. The steps for the preprocessing procedure are as follows:

3.2.1.1 Binarization

Image binarization is the process of taking a grayscale image and converting it to black-and-white, essentially reducing the information contained within the image from 256 shades of gray to 2: black and white, a binary image. This is sometimes known as image thresholding, although thresholding may produce images with more than 2 levels of gray.[4]

3.2.1.2 Perspective Correction

It is the process of changing the perspective of a given image for getting better insights about the required information.[5] A good option for a reliable set of points which can be used to compute the transformation matrix is finding the corners of the stabes, as they usually form a quadrilateral and will always be present on any image. To do this, we filter the lines to select only those which are parallel and equally distant. Each group of five equally distant lines is assumed to be part of a staff. Finally, we select one of the groups given an heuristic function, determine the segments which form part of the upper and lower staff lines, and consider their starting and end points as the corners of the staff.[6]

3.2.1.3 Staff location detection

Determining where each staff is in the image is important. To accomplish this, we perform connected component analysis and get the bounding box of each component. For a component to be considered a staff candidate, it needs to have a width of at least half of the total image width, and needs to have an aspect ratio of 5:1 or more.[6]

3.2.2 Note Segmentation

After the preprocessing procedure, we have most of the distortion caused by perspective corrected, all the staves binarized and located. The next step is to segment all the notes and symbols to be able to feed them to the neural network. The steps for note segmentation are as follows:

3.2.2.1 Staff removal

To ease pitch detection and the segmentation of musical symbols, the staff lines are detected and removed taking into account that the lines might not be completely horizontal.

3.2.2.2 Detection of musical symbols

First, we perform connected component analysis, and find the bounding box of each component. Then, we recursively join the boxes which are very close to or inside another. This is made because some symbols could become disconnected in some of the previous steps.

3.2.2.3 Segmentation of note heads

Our approach is to feed individual notes and symbols to the neural network to recognize them separately, and then reconstruct the music using that information. Because of that, notes connected by a beam need to be separated. To accomplish this, all black note heads are detected and determine which of them are note heads given some properties (size, aspect ratio, area and being close to a stem).[6]

3.2.3 Symbol Recognition

Once every symbol has been segmented, the recognition stage begins. First of all, the bars are detected given their very distinct properties (very small width and very small aspect ratio), the end bar is also detected separately. As for the rest of the symbols, they are sent to the neural network for recognition.[6]

3.2.3.1 Neural network architecture

Here, a CNN is used for the classification task. It might change in the future after more testing has been done. For now, we will only consider notes and rests from whole to a sixteenth beat. More information about the neural network architecture is given in the next section.

3.3 ALGORITHM

The algorithm to be used is Convolution Neural Network. We have chosen this algorithm as it has produced satisfactory results in many graphics recognition tasks. It is also a simple and generic architecture which is expected to produce expected results because the classification task isn't very demanding.

3.3.1 Convolutional Neural Network

A Convolutional Neural Network is a Deep Learning algorithm which can take in an input image, assign importance (learnable weights and biases) to various aspects/objects in the image and be able to differentiate one from the other. The pre-processing required in a ConvNet is much lower as compared to other classification algorithms. The architecture performs a better fitting to the image dataset due to the reduction in the number of parameters involved and reusability of weights. In other words, the network can be trained to understand the sophistication of the image better. The role of ConvNet is to reduce the images into a form which is easier to process, without losing features which are critical for getting a good prediction. [7]

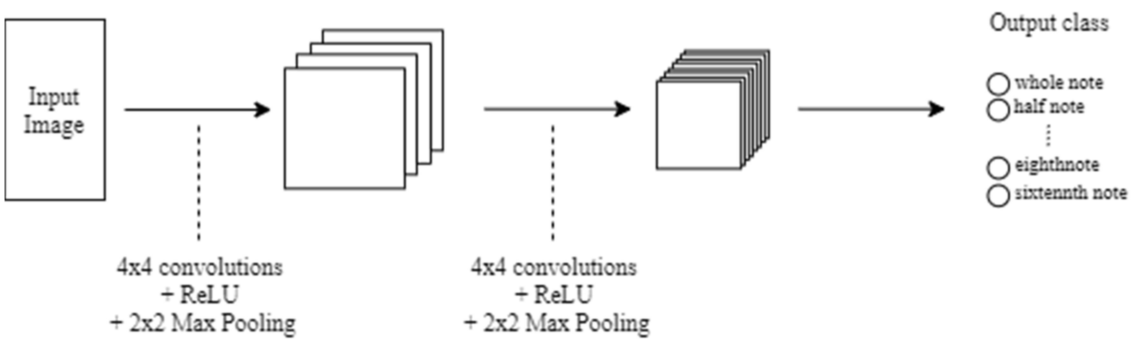

Fig. 3.3.1. Convolutional Neural Network

3.3.1.1 Convolutional Layer

A convolution layer is a fundamental component of the CNN architecture that performs feature extraction, which typically consists of a combination of linear and nonlinear operations, i.e. convolution operation and activation function.Convolution is a specialized type of linear operation used for feature extraction, where a small array of numbers, called a kernel, is applied across the input, which is an array of numbers, called a tensor. An element-wise product between each element of the kernel and the input tensor is calculated at each location of the tensor and summed to obtain the output value in the corresponding position of the output tensor, called a feature map. This procedure is repeated applying multiple kernels to form an arbitrary number of feature maps, which represent different characteristics of the input tensors.[8]

3.3.1.2 Pooling Layer

Similar to the Convolutional Layer, the Pooling layer is responsible for reducing the spatial size of the Convolved Feature. This is to decrease the computational power required to process the data through dimensionality reduction. Furthermore, it is useful for extracting dominant features which are rotational and positional invariant, thus maintaining the process of effectively training the model.[7]

The most popular form of pooling operation is max pooling, which extracts patches from the input feature maps, outputs the maximum value in each patch, and discards all the other values. Max Pooling also performs as a Noise Suppressant. It discards the noisy activations altogether and also performs de-noising along with dimensionality reduction.[8]

3.3.1.3 Fully Connected Layer

The output feature maps of the final convolution or pooling layer is typically flattened, i.e., transformed into a one-dimensional (1D) array of numbers (or vector), and connected to one or more fully connected layers, also known as dense layers, in which every input is connected to every output by a learnable weight. Once the features extracted by the convolution layers and downsampled by the pooling layers are created, they are mapped by a subset of fully connected layers to the final outputs of the network, such as the probabilities for each class in classification tasks. The final fully connected layer typically has the same number of output nodes as the number of classes. Each fully connected layer is followed by a nonlinear function, such as ReLU.[8]

3.4 TOOLS TO BE USED

The tools to be used in the development of this project are as follows:

3.4.1 Python

Python is an interpreted high-level general-purpose programming language. Python's design philosophy emphasizes code readability with its notable use of significant indentation. Its language constructs as well as its object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects.

3.4.2 NumPy

NumPy is a library for the Python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays.

3.4.3 Pandas

Pandas is a software library written for the Python programming language for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and time series.

3.4.4 Keras

Keras is an open-source software library that provides a Python interface for artificial neural networks. Keras acts as an interface for the TensorFlow library.

3.4.5 OpenCV

OpenCV (Open Source Computer Vision Library) is a library of programming functions mainly aimed at real-time computer vision. Starting in 2011, OpenCV features GPU acceleration for real-time operations.

CHAPTER FOUR: EPILOGUE

4.1 EXPECTED RESULT

By developing this programme, we aim to provide better facilities for people working/studying music. This programme helps to read/scan and play music sheets. The programme focuses on reading music sheets with grand staff. It also focuses on giving the music sheet in such a format that can be used by other softwares as well to edit, create and compose the music contained in it.

4.2 GANTT CHART

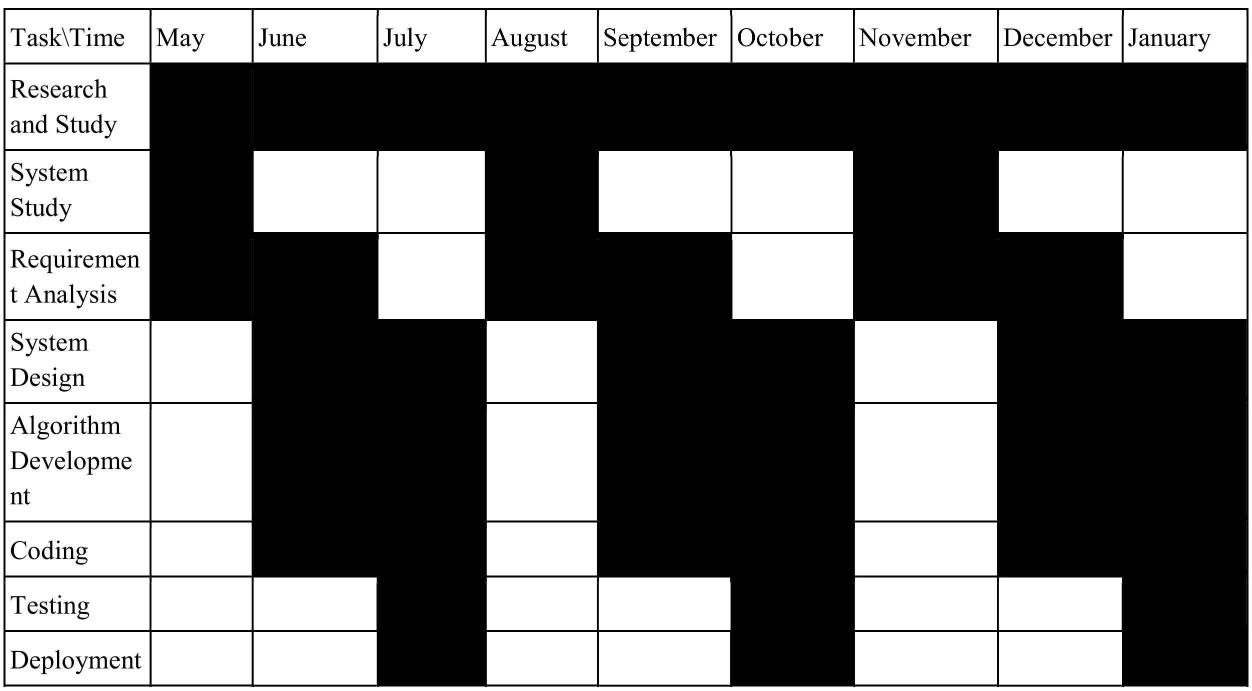

Fig. 4.2 . Gantt Chart

REFERENCES

[1] “Playscore Pro” , Playscore Pro [Online].

Available:https://download.cnet.com/PlayScore-Pro/3000-2141_4-77467071.html

[2] “Guitar Pro” , Guitar Pro [Online].

Available:https://support.guitar-pro.com/hc/en-us

[3] “Scan Score” Scan Score [Online].

Available:https://scan-score.com/en/about-us/

[4] “Image Binarization (1) : Introduction”, Craft of Coding, February 13, 2017 [Online]. Available:https://craftofcoding.wordpress.com/2017/02/13/image-binarization-1-introduction/

[5] “Introduction to Image Pre-processing”, Great Learning Team, August 17, 2020. [Online]. Available:https://www.mygreatlearning.com/blog/introduction-to-image-pre-processing/

[6] “Camera-Based Optical Music Recognition Using a Convolutional Neural Network”, Adrià Rico and Alicia Fornés, January 29, 2018. [Online]. Available:https://ieeexplore.ieee.org/document/8270203/

[7] “A Comprehensive Guide to Convolutional Neural Networks”, Sumit Saha, December 15, 2018. [Online]. Available:https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53

[8] “Convolutional neural networks: an overview and application in radiology”, Rikiya Yamashita, Mizuho Nishio, Richard Kinh Gian Do & Kaori Togashi, June 22 2018. [Online]. Available:https://insightsimaging.springeropen.com/articles/10.1007/s13244-018-0639-9