

**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT  
KHOA CƠ KHÍ CHẾ TẠO MÁY**

\*\*\*📖\*\*\*



**HCMUTE**

**TRÍ TUỆ NHÂN TẠO  
(MMH: 221ARIN337629)**

**ĐỀ TÀI: PHÁT HIỆN CÁC LOẠI NHẠC CỤ CỔ TRUYỀN VIỆT NAM**

**GVHD: PGS.TS Nguyễn Trường Thịnh**

**SVTH: Nguyễn Anh Hào - 20104006**

**TP. Thủ Đức, tháng 12 năm 2022**

# MỤC LỤC

<b>CHƯƠNG 1: TỔNG QUAN</b>	<b>1</b>
1.1. Giới thiệu:	1
1.2. Ý nghĩa thực tiễn:	2
1.3. Mục tiêu đề tài:	2
1.4. Giới hạn đề tài:	2
<b>CHƯƠNG 2: CƠ SỞ LÝ THUYẾT</b>	<b>3</b>
2.1. Trí tuệ nhân tạo:	3
2.1.1. Định nghĩa:	3
2.1.2. Quá trình hình thành và phát triển:	3
2.2. Sơ lược Machine Learning:	4
2.3. Sơ lược về Deep Learning:	5
2.4. CNN (Convolutional Neural Network):	6
2.4.1. Khái niệm:	6
2.4.2. Cấu trúc của mạng CNN:	6
<b>CHƯƠNG 3: DỮ LIỆU VÀ CHƯƠNG TRÌNH</b>	<b>8</b>
3.1. Dữ liệu:	8
3.2. Chương trình:	8
3.2.1. Tách dữ liệu:	8
3.2.2. Xử lý dữ liệu:	9
3.2.3. Thiết kế CNN:	9
3.2.4. Tiến hành train mô hình	10
3.3. Kết quả:	10
<b>CHƯƠNG 4: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN</b>	<b>14</b>
4.1. Kết luận:	14
4.2. Hướng phát triển	14
<b>TÀI LIỆU THAM KHẢO:</b>	<b>14</b>

## MỤC LỤC HÌNH ẢNH

Hình 1. Các loại nhạc cụ cổ truyền Việt Nam.....	2
Hình 2. Sự phát triển của Trí tuệ nhân tạo (AI) .....	4
Hình 3. Cách thức hoạt động của Machine Learning.....	5
Hình 4. Cách thức hoạt động của Deep Learning.....	6
Hình 5. Cấu trúc của mạng CNN.....	7
Hình 6. Các thư mục chứa dữ liệu của các loại nhạc cụ.....	8
Hình 7. Khai báo các thư viện.....	9
Hình 8. Khai báo hình ảnh.....	9
Hình 9. Mô hình CNN được thiết lập.....	10
Hình 10. Kết quả của mô hình CNN.....	10
Hình 11. Mô hình được train trên Google Colab.....	10
Hình 12. File lưu lại mô hình đã train.....	11
Hình 13. Kiểm tra độ chính xác mô hình.....	11
Hình 14. Biểu đồ trực quan về độ chính xác qua các lần học.....	11
Hình 15. Test mô hình sau khi train.....	12
Hình 16. Kết quả test mô hình.....	12
Hình 17. Code giao diện trên Pycharm.....	12
Hình 18. Giao diện ứng dụng.....	13
Hình 19. Kết quả của mô hình.....	13

# CHƯƠNG 1: TỔNG QUAN

## 1.1. Giới thiệu:

Trí tuệ nhân tạo hay trí thông minh nhân tạo (Artificial intelligence - viết tắt là AI) là một ngành thuộc lĩnh vực khoa học máy tính (Computer science). Công nghệ công nghệ AI là công nghệ mô phỏng các quá trình suy nghĩ và học tập của con người cho máy móc, đặc biệt là các hệ thống máy tính. Các quá trình này bao gồm việc học tập (thu thập thông tin và các quy tắc sử dụng thông tin), lập luận (sử dụng các quy tắc để đạt được kết luận gần đúng hoặc xác định) và tự sửa lỗi.

Một số lợi ích mà AI mang lại có thể kể đến như:

- *Phát hiện và hạn chế rủi ro:* AI có thể giúp con người dự báo trước những rủi ro của toàn nhân loại như dịch bệnh, thảm họa thiên nhiên, cho đến những rủi ro mang tính cá nhân như rủi ro trong kinh doanh,...
- *Tiết kiệm sức lao động của con người:* Nhờ có AI mà con người có thể tối ưu hóa hoạt động sản xuất, giảm bớt nhân công trong việc vận hành dây chuyền.
- *Giải phóng sức sáng tạo:* Công nghệ trí tuệ nhân tạo có thể thay con người đảm nhiệm nhiều công việc như đánh giá dữ liệu, giao tiếp với khách hàng... qua đó tạo điều kiện và cho phép con người có thể tập trung khai thác sâu hơn khả năng sáng tạo của bản thân, phát triển chuyên môn một cách bài bản, sâu sắc hơn.
- *Cầu nối ngôn ngữ:* Với trí tuệ nhân tạo, những rào cản về ngôn ngữ đang dần được gỡ bỏ để con người có thể thoải mái tiếp xúc với mọi nền văn hóa, mọi ngôn ngữ, mọi quốc gia, qua đó mở rộng thêm nhiều cơ hội học tập, làm việc khác.
- *Cá nhân hóa:* AI giúp đánh giá và cá nhân hóa dữ liệu giúp con người có thể thấy được những thứ mà họ muốn thấy thông qua hành vi của người dùng.

Trong bài viết này, việc phát hiện các loại nhạc cụ cổ truyền Việt Nam là sử dụng các hình ảnh chụp lại của loại nhạc cụ cần nhận diện, hình ảnh được đưa vào máy tính để phát hiện và xác định tên loại nhạc cụ để người dùng có thể phân biệt chính xác được các loại nhạc cụ cổ truyền của Việt Nam, phục vụ được cho cả lĩnh vực giáo dục và lĩnh vực sản xuất. <sup>[1]</sup>



*Hình 1. Các loại nhạc cụ cổ truyền Việt Nam*

## **1.2. Ý nghĩa thực tiễn:**

Các loại nhạc cụ dân tộc Việt Nam không những thể hiện nét đẹp văn hóa Việt mà còn tạo nên những âm thanh độc đáo khác nhau. Các âm thanh muôn màu mang đến những cảm xúc khác nhau cho người nghe. Từ độ chính xác và hiệu quả thì AI đã được ứng dụng vào việc phát hiện các loại nhạc cụ cổ truyền Việt Nam. Việc phát hiện các loại nhạc cụ cổ truyền Việt Nam có thể được ứng dụng để tạo ra một số phần mềm giáo dục, nhằm phổ cập kiến thức về các loại nhạc cụ dân tộc. Việc phát hiện các loại nhạc cụ cổ truyền Việt Nam cũng có thể được ứng dụng trong lĩnh vực sản xuất như phân phối các loại nhạc cụ vào kho lưu trữ đúng khu vực của từng loại.

## **1.3. Mục tiêu đề tài:**

Mục tiêu của đề tài là giải quyết bài toán phát hiện vật dựa trên thuật toán CNN (Convolutional Neural Network), tối ưu hóa mô hình để phát hiện nhạc cụ trong một khung hình. Cải tiến việc phát hiện các loại nhạc cụ trong thời gian thực và phát triển mô hình để phát hiện nhiều loại nhạc cụ trong cùng một khung hình.

## **1.4. Giới hạn đề tài:**

Ở đề tài “Phát hiện các loại nhạc cụ cổ truyền Việt Nam” này chỉ dừng lại ở việc hoàn thành mô hình thuật toán ở trên Google Colab, tạo ra giao diện tương tác đơn giản trên phần

mềm Pycharm và kiểm tra độ chính xác của mô hình bằng cách đưa ảnh chụp lại vào laptop cá nhân, máy tính.

## **CHƯƠNG 2: CƠ SỞ LÝ THUYẾT**

### **2.1. Trí tuệ nhân tạo:**

#### **2.1.1. Định nghĩa:**

Trí tuệ nhân tạo hay trí thông minh nhân tạo (Artificial intelligence – viết tắt là AI) là một ngành thuộc lĩnh vực khoa học máy tính (Computer science). Là trí tuệ do con người lập trình tạo nên với mục tiêu giúp máy tính có thể tự động hóa các hành vi thông minh như con người.

Trí tuệ nhân tạo khác với việc lập trình logic trong các ngôn ngữ lập trình là ở việc ứng dụng các hệ thống học máy (machine learning) để mô phỏng trí tuệ của con người trong các xử lý mà con người làm tốt hơn máy tính.

Cụ thể, trí tuệ nhân tạo giúp máy tính có được những trí tuệ của con người như: biết suy nghĩ và lập luận để giải quyết vấn đề, biết giao tiếp do hiểu ngôn ngữ, tiếng nói, biết học và tự thích nghi,...

Tuy rằng trí thông minh nhân tạo có nghĩa rộng như là trí thông minh trong các tác phẩm khoa học viễn tưởng, nó là một trong những ngành trọng yếu của tin học. Trí thông minh nhân tạo liên quan đến cách cư xử, sự học hỏi và khả năng thích ứng thông minh của máy móc.

#### **2.1.2. Quá trình hình thành và phát triển:**

Alan Turing (1912–1954) là một nhà toán học, logic học và mật mã học người Anh, người được cho là cha đẻ của ngành khoa học máy tính và trí tuệ nhân tạo. Ông được biết đến rộng rãi với vai trò là người giải được mật mã Enigma. Ý tưởng xây dựng một chương trình AI xuất hiện lần đầu vào tháng 10/1950, khi mà ông xem xét vấn đề “liệu máy tính có khả năng suy nghĩ hay không?”.

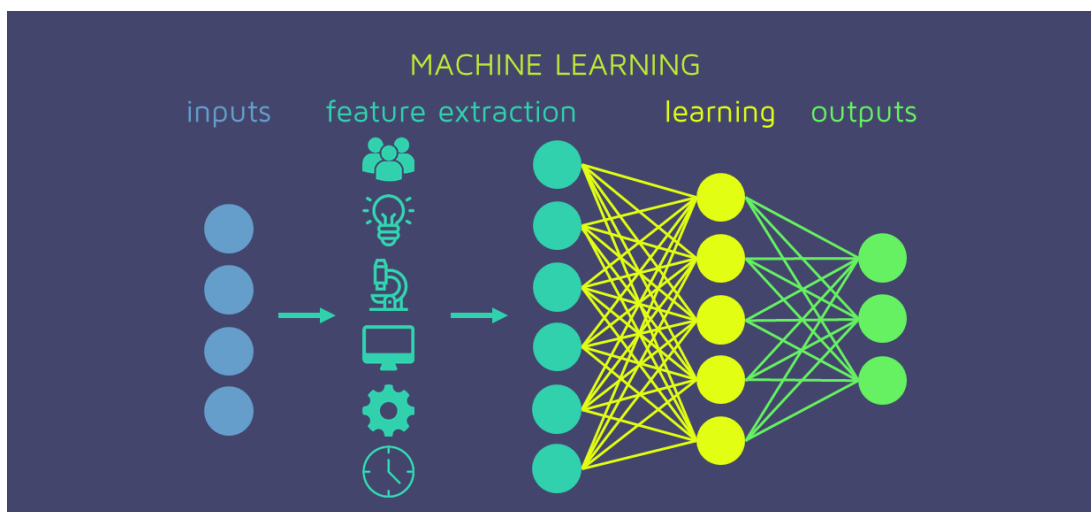


Hình 2. Sự phát triển của Trí tuệ nhân tạo (AI)

## 2.2. Sơ lược Machine Learning:

Machine learning theo định nghĩa cơ bản là ứng dụng các thuật toán để phân tích cú pháp dữ liệu, học hỏi từ nó, và sau đó thực hiện một quyết định hoặc dự đoán về các vấn đề có liên quan. Vì vậy, thay vì code phần mềm bằng cách thức thủ công với một bộ hướng dẫn cụ thể để hoàn thành một nhiệm vụ cụ thể, máy được “đào tạo” bằng cách sử dụng một lượng lớn dữ liệu và các thuật toán cho phép nó học cách thực hiện các tác vụ.

Một trong những lĩnh vực ứng dụng tốt nhất cho Machine learning trong nhiều năm qua là computer vision, mặc dù nó vẫn đòi hỏi rất nhiều kỹ năng code thủ công để có thể hoàn thành công việc. Mọi người vẫn sẽ viết các lớp phân loại bằng tay như các bộ lọc để chương trình có thể xác định nơi mà một đối tượng bắt đầu và kết thúc. Phát hiện hình dạng để xác định nếu nó có tám mắt. Một phân loại để nhận dạng các chữ cái “S-T-O-P”. Từ tất cả những ứng dụng phân loại, họ sẽ phát triển các thuật toán để làm cho hình ảnh và “học” khả năng nhận diện dấu hiệu liệu nó có phải là kí hiệu stop hay không? . Tốt, nhưng không hoàn toàn hoàn hảo. Đặc biệt vào một ngày sương mù khi tầm nhìn không thấy rõ, hoặc cây che khuất một phần. Đó là lý do khiến computer vision và image detection không gây được thiện cảm, vì nó quá nhạy cảm và dễ phát sinh lỗi.



*Hình 3. Cách thức hoạt động của Machine Learning*

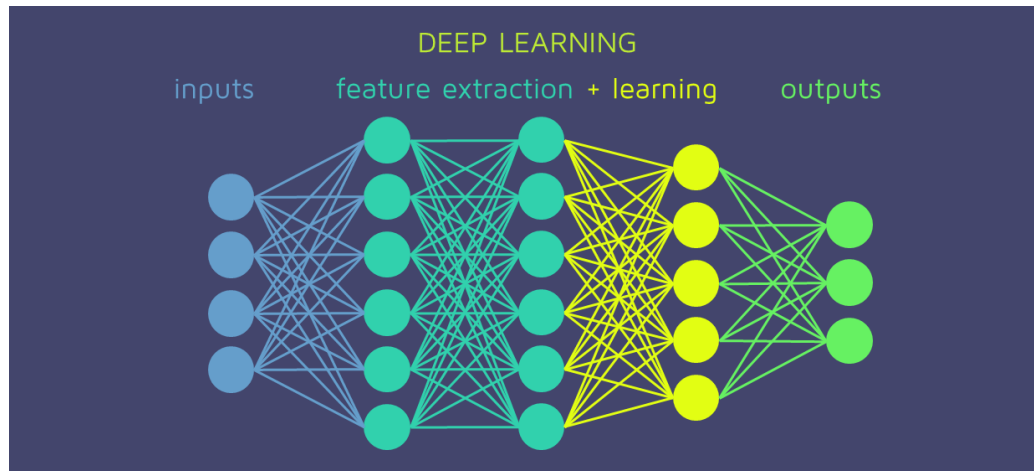
### **2.3. Sơ lược về Deep Learning:**

Deep Learning là tập hợp con của Machine Learning và nó có tác dụng hỗ trợ cho máy tính tự huấn luyện chính nó để có thể thực hiện mọi tác vụ tương tự như con người. Điều này chính là giúp máy tính bắt chước con người cách học hỏi và suy nghĩ. Các hệ thống của Deep Learning có khả năng cải thiện được những hiệu suất của chúng với quyền truy cập vào dữ liệu sẽ được nhiều hơn. Thông thường, phiên bản máy sẽ có nhiều kinh nghiệm hơn; những máy móc đã có đủ kinh nghiệm thì sẽ được mang đi phục vụ cho những công việc như: lái xe, phát hiện cỏ dại,... Deep Learning có hỗ trợ cho việc dịch ngôn ngữ, phân loại các hình ảnh, nhận dạng giọng nói. Chính vì thế, nó có thể được ứng dụng để giải quyết mọi nhu cầu cần nhận dạng mẫu mà không cần đến sự can thiệp của con người.

Deep Learning hoạt động dựa vào mạng lưới thần kinh nhân tạo và nó bao gồm nhiều lớp chứa các dữ liệu mô phỏng cách thức vận hành của não người. Bộ phận mạnh thần kinh nhân tạo này giống với não người bao gồm: Các node (nó là đơn vị thần kinh trong mạng thần kinh nhân tạo) là những neuron thần kinh. Hầu hết các node bản thân thường chỉ có khả năng trả lời cho những câu hỏi đơn giản và cơ bản nhất; với những tác vụ khó thì chúng sẽ tiến hành liên kết với nhau để trả lời.



Ngày nay, khả năng nhận dạng hình ảnh của máy móc được luyện tập thông qua deep learning trong một số tình huống tốt hơn so với con người. AlphaGo của Google đã được huấn luyện chơi cờ vây – nó điều chỉnh mạng lưới thần kinh của mình bằng cách tự mình chống lại chính mình. [2]



*Hình 4. Cách thức hoạt động của Deep Learning*

## **2.4. CNN (Convolutional Neural Network):**

### **2.4.1. Khái niệm:**

CNN là tên viết tắt của từ Convolutional Neural Network (hay còn gọi là CNN - mạng nơ ron tích chập). Đây là một trong những mô hình Deep Learning vô cùng tiên tiến. CNN sẽ cho phép bạn xây dựng các hệ thống thông minh với độ chính xác vô cùng cao. Hiện nay, CNN được ứng dụng rất nhiều trong những bài toán nhận dạng object trong ảnh.

### **2.4.2. Cấu trúc của mạng CNN:**

Mạng CNN là một tập hợp các lớp Convolution chồng lên nhau và sử dụng các hàm nonlinear activation như ReLU và tanh để kích hoạt các trọng số trong các node. Mỗi một lớp sau khi thông qua các hàm kích hoạt sẽ tạo ra các thông tin trừu tượng hơn cho các lớp tiếp theo.

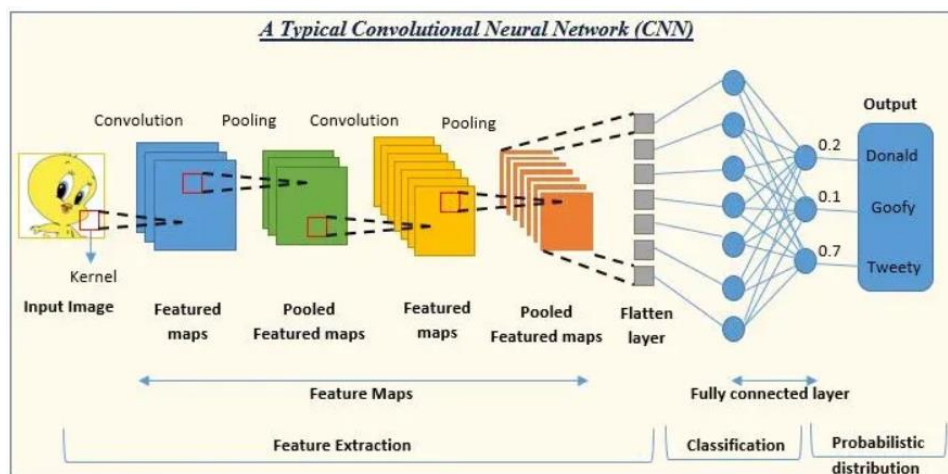
Mỗi một lớp sau khi thông qua các hàm kích hoạt sẽ tạo ra các thông tin trừu tượng hơn cho các lớp tiếp theo. Trong mô hình mạng truyền ngược (feedforward neural network) thì mỗi neural đầu vào (input node) cho mỗi neural đầu ra trong các lớp tiếp theo.

Mô hình này gọi là mạng kết nối đầy đủ (fully connected layer) hay mạng toàn vẹn (affine layer). Còn trong mô hình CNNs thì ngược lại. Các layer liên kết được với nhau thông qua cơ chế convolution.

Layer tiếp theo là kết quả convolution từ layer trước đó, nhờ vậy mà ta có được các kết nối cục bộ. Như vậy mỗi neuron ở lớp kế tiếp sinh ra từ kết quả của filter áp đặt lên một vùng ảnh cục bộ của neuron trước đó.

Mỗi một lớp được sử dụng các filter khác nhau thông thường có hàng trăm hàng nghìn filter như vậy và kết hợp kết quả của chúng lại. Ngoài ra có một số layer khác như pooling/subsampling layer dùng để chắt lọc lại các thông tin hữu ích hơn (loại bỏ các thông tin nhiễu).

Trong quá trình huấn luyện mạng (training) CNN tự động học các giá trị qua các lớp filter dựa vào cách thức mà bạn thực hiện. Ví dụ trong tác vụ phân lớp ảnh, CNNs sẽ cố gắng tìm ra thông số tối ưu cho các filter tương ứng theo thứ tự raw pixel > edges > shapes > facial > high-level features. Layer cuối cùng được dùng để phân lớp ảnh.



Hình 5. Cấu trúc của mạng CNN

Trong mô hình CNN có 2 khía cạnh cần quan tâm là tính bất biến (Location Invariance) và tính kết hợp (Compositionality). Với cùng một đối tượng, nếu đối tượng này được chiếu

theo các gốc độ khác nhau (translation, rotation, scaling) thì độ chính xác của thuật toán sẽ bị ảnh hưởng đáng kể.

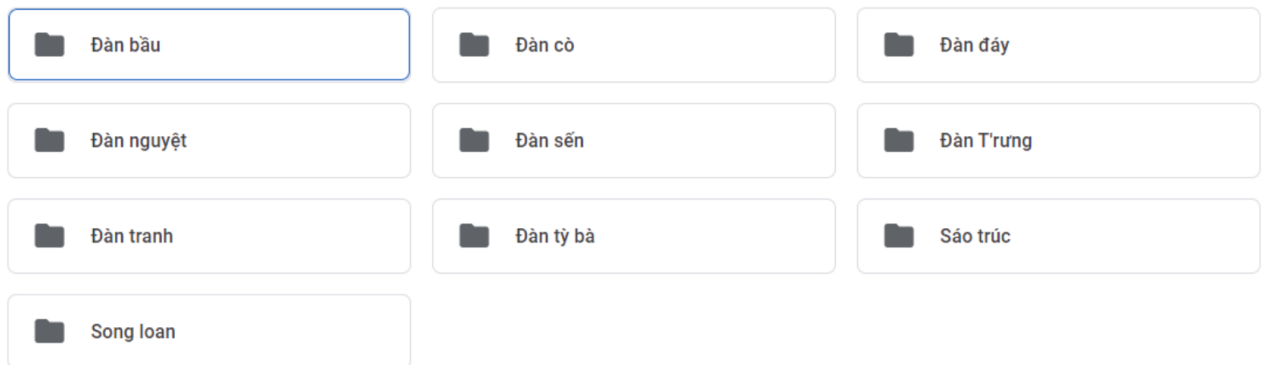
Pooling layer sẽ cho bạn tính bất biến đối với phép dịch chuyển (translation), phép quay (rotation) và phép co giãn (scaling). Tính kết hợp cục bộ cho ta các cấp độ biểu diễn thông tin từ mức độ thấp đến mức độ cao và trừu tượng hơn thông qua convolution từ các filter.

Đó là lý do tại sao CNN cho ra mô hình với độ chính xác rất cao. Cũng giống như cách con người nhận biết các vật thể trong tự nhiên. <sup>[3]</sup>

## CHƯƠNG 3: DỮ LIỆU VÀ CHƯƠNG TRÌNH

### 3.1. Dữ liệu:

Trong nghiên cứu này, bộ dữ liệu được thu thập từ 10 loại nhạc cụ cổ truyền bao gồm: Đàn bầu, Đàn cò, Đàn đáy, Đàn nguyệt, Đàn sến, Đàn T'rung, Đàn tranh, Đàn tỳ bà, Sáo trúc và Song loan. Mỗi thư mục của mỗi loại nhạc cụ dưới đây gồm 50 hình ảnh được thu thập từ nguồn Internet.



Hình 6. Các thư mục chứa dữ liệu của các loại nhạc cụ

### 3.2. Chương trình:

#### 3.2.1. Tách dữ liệu:

Trong giai đoạn này, toàn bộ tệp dữ liệu được chia thành 3 thư mục: Train, Validation, Test. Trong đó 100% dữ liệu được sử dụng cho Train phục vụ mục đích học của mô hình và 60% dữ liệu được sử dụng cho Validation phục vụ cho mục đích kiểm tra kết quả học. Với

Test ta sẽ lấy ra các hình từ thư viện ảnh (không phải ảnh trong tệp dữ liệu đã train) để test sau khi train. Sau khi phân tách Validation, Train thì tệp dữ liệu Train chứa 500 hình ảnh, trong khi tệp dữ liệu Validation chứa 300 hình ảnh.

### 3.2.2. Xử lý dữ liệu:

Trong giai đoạn này, tất cả các hình ảnh được thay đổi kích thước thành kích thước cố định là 150 x 150 (rộng x cao).

### 3.2.3. Thiết kế CNN:

```
from sklearn.model_selection import train_test_split
import matplotlib.pyplot as plt
import os
import numpy as np
import tensorflow as tf
from keras.utils import np_utils
from keras.models import Sequential
from keras.layers import Dense, Activation, Dropout, Conv2D, MaxPooling2D, Flatten
from tensorflow.keras.optimizers import SGD, RMSprop
from keras.callbacks import EarlyStopping, ModelCheckpoint
from sklearn.utils import validation
from sklearn import preprocessing
from tensorflow.keras.utils import to_categorical
from tensorflow.keras.models import load_model
from tensorflow.keras.utils import load_img, img_to_array
from tensorflow.keras.preprocessing.image import ImageDataGenerator
```

*Hình 7. Khai báo các thư viện*

```
[ ] from google.colab import drive
drive.mount('/content/drive')

Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remount=True).

[ ] %cd /content/drive/MyDrive/Colab Notebooks/Nhac_cu
/content/drive/MyDrive/Colab Notebooks/Nhac_cu

[ ] train_datagen=ImageDataGenerator(rescale=1./255, shear_range=0.2, zoom_range=0.2, horizontal_flip=True)

[ ] x_train=train_datagen.flow_from_directory('/content/drive/MyDrive/Colab Notebooks/Nhac_cu/Train',target_size=(150,150), batch_size=32, class_mode='categorical')
Found 500 images belonging to 10 classes.

[ ] validation=train_datagen.flow_from_directory('/content/drive/MyDrive/Colab Notebooks/Nhac_cu/Train',target_size=(150,150), batch_size=32, class_mode='categorical')
Found 500 images belonging to 10 classes.
```

*Hình 8. Khai báo hình ảnh*

```

model=Sequential()
model.add(Conv2D(32,(3,3), activation='relu', kernel_initializer='he_uniform', padding='same',input_shape=(150,150,3)))
model.add(Conv2D(32,(3,3), activation='relu', kernel_initializer='he_uniform', padding='same'))
model.add(MaxPooling2D(2,2))
model.add(Conv2D(64,(3,3), activation='relu', kernel_initializer='he_uniform', padding='same'))
model.add(Conv2D(64,(3,3), activation='relu', kernel_initializer='he_uniform', padding='same'))
model.add(MaxPooling2D(2,2))
model.add(Conv2D(128,(3,3), activation='relu', kernel_initializer='he_uniform', padding='same'))
model.add(Conv2D(128,(3,3), activation='relu', kernel_initializer='he_uniform', padding='same'))
model.add(MaxPooling2D(2,2))
model.add(Flatten())
model.add(Dense(128,activation='relu',kernel_initializer='he_uniform'))
model.add(Dropout(0.2))
model.add(Dense(10,activation='softmax'))
model.summary()

```

*Hình 9. Mô hình CNN được thiết lập*

Model: "sequential"

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 150, 150, 32)	896
conv2d_1 (Conv2D)	(None, 150, 150, 32)	9248
max_pooling2d (MaxPooling2D)	(None, 75, 75, 32)	0
conv2d_2 (Conv2D)	(None, 75, 75, 64)	18496
conv2d_3 (Conv2D)	(None, 75, 75, 64)	36928
max_pooling2d_1 (MaxPooling2D)	(None, 37, 37, 64)	0
conv2d_4 (Conv2D)	(None, 37, 37, 128)	73856
conv2d_5 (Conv2D)	(None, 37, 37, 128)	147584
max_pooling2d_2 (MaxPooling2D)	(None, 18, 18, 128)	0
flatten (Flatten)	(None, 41472)	0
dense (Dense)	(None, 128)	5308544
dropout (Dropout)	(None, 128)	0
dense_1 (Dense)	(None, 10)	1290
Total params: 5,596,842		
Trainable params: 5,596,842		
Non-trainable params: 0		

*Hình 10. Kết quả của mô hình CNN*

### 3.2.4. Tiến hành train mô hình

```

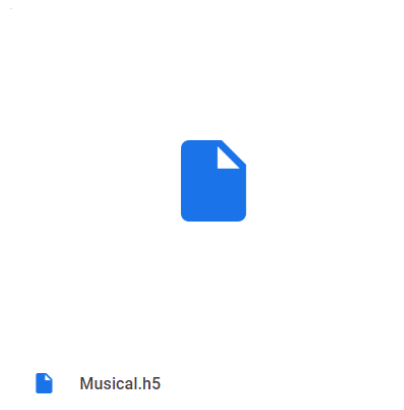
model.compile(optimizer='adam',loss='categorical_crossentropy',metrics=['accuracy'])
history=model.fit(x_train,epochs=100,batch_size=128,verbose=1, validation_data=validation, callbacks=[EarlyStopping(monitor='val_loss', patience=20)])

```

*Hình 11. Mô hình được train trên Google Colab*

## 3.3. Kết quả:

Sau khi train xong trên Google Colab và tiến hành lưu mô hình, tại Google Drive sẽ có file Musical.h5 như hình



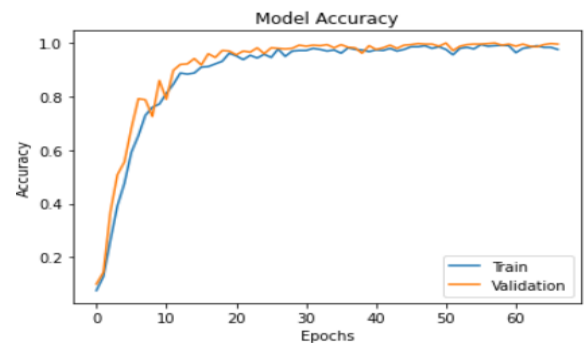
Hình 12. File lưu lại mô hình đã train

Tiến hành đánh giá độ chính xác của mô hình và xem biểu đồ trực quan về độ chính xác qua các lần học vừa train, độ chính xác chung đạt 99,6%.

```
Score=model.evaluate(x_train,verbose=0)
print('Train Loss', Score[0])
print('Train Accuracy', Score[1])
```

Train Loss 0.012072446756064892  
Train Accuracy 0.9959999918937683

Hình 13. Kiểm tra độ chính xác mô hình



Hình 14. Biểu đồ trực quan về độ chính xác qua các lần học

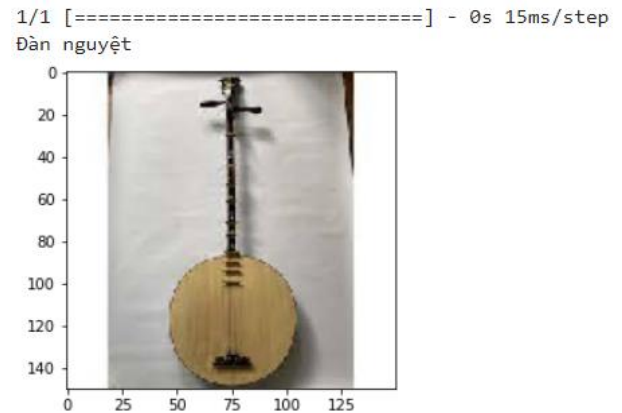
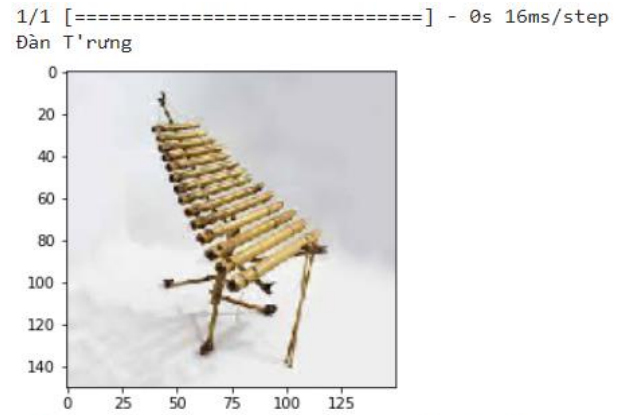
Tiến hành kiểm tra khả năng nhận diện ảnh của mô hình, kết quả nhận dạng khá chính xác những loại nhạc cụ đã train cho mô hình.

```

for i in os.listdir(test):
    img=load_img(test+'/'+i,target_size=(150,150))
    plt.imshow(img)
    img=img_to_array(img)
    img=img.astype('float32')
    img=img/255
    img=np.expand_dims(img,axis=0)
    result=model_CNN.predict(img)
    if round(result[0][0])==1:
        prediction='Sáo trúc'
    if round(result[0][1])==1:
        prediction='Song loan'
    if round(result[0][2])==1:
        prediction="Đàn T'rưng"
    if round(result[0][3])==1:
        prediction='Đàn bầu'
    if round(result[0][4])==1:
        prediction='Đàn cò'
    if round(result[0][5])==1:
        prediction='Đàn nguyệt'
    if round(result[0][6])==1:
        prediction='Đàn sến'
    if round(result[0][7])==1:
        prediction='Đàn tranh'
    if round(result[0][8])==1:
        prediction='Đàn tỳ bà'
    if round(result[0][9])==1:
        prediction='Đàn đáy'
    print(prediction)
    plt.show()

```

Hình 15. Test mô hình sau khi train



Hình 16. Kết quả test mô hình

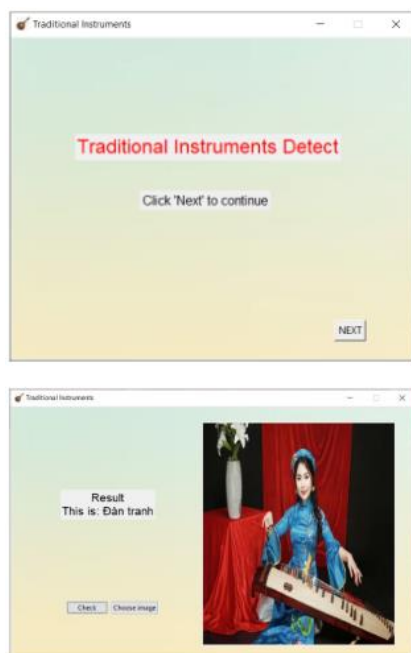
Tiến hành tạo giao diện ứng dụng trên Pycharm

```

#thưviện
import ...
wd1=Tk()
wd1.geometry('500x400+540+260')
wd1.title('Traditional Instruments')
wd1.iconbitmap('icon.ico')#biểu tượng
wd1.resizable(False,False)
canvas_wd1 = Canvas(wd1, width=500, height=400)
bg_wd1 = ImageTk.PhotoImage(Image.open('bg1.jpg'))#ảnh nền
a=Label(wd1,font=('Arial',18), text='Traditional Instruments Detect',fg='red')
a.place(x=80,y=120)
b=Label(wd1,font=('Arial',12), text="Click 'Next' to continue",fg='black')
b.place(x=160,y=190)
canvas_wd1.create_image(0, 0, anchor=NW, image=bg_wd1)
canvas_wd1.pack()
def open_wd2():
    wd1.withdraw()
    button_next=Button(wd1,text='NEXT',command=open_wd2)
    button_next.place(x=400, y=350)
#window2
wd2 = Toplevel(wd1)
wd2.title('Traditional Instruments')
wd2.iconbitmap('icon.ico')
wd2.geometry('728x455')
wd2.resizable(False, False)
wd2.withdraw()
canvas_wd2 = Canvas(wd2, width=728, height=455)
bg_wd2 = ImageTk.PhotoImage(Image.open('bg1.jpg'))
canvas_wd2.create_image(0, 0, anchor=NW, image=bg_wd2)
canvas_wd2.pack()

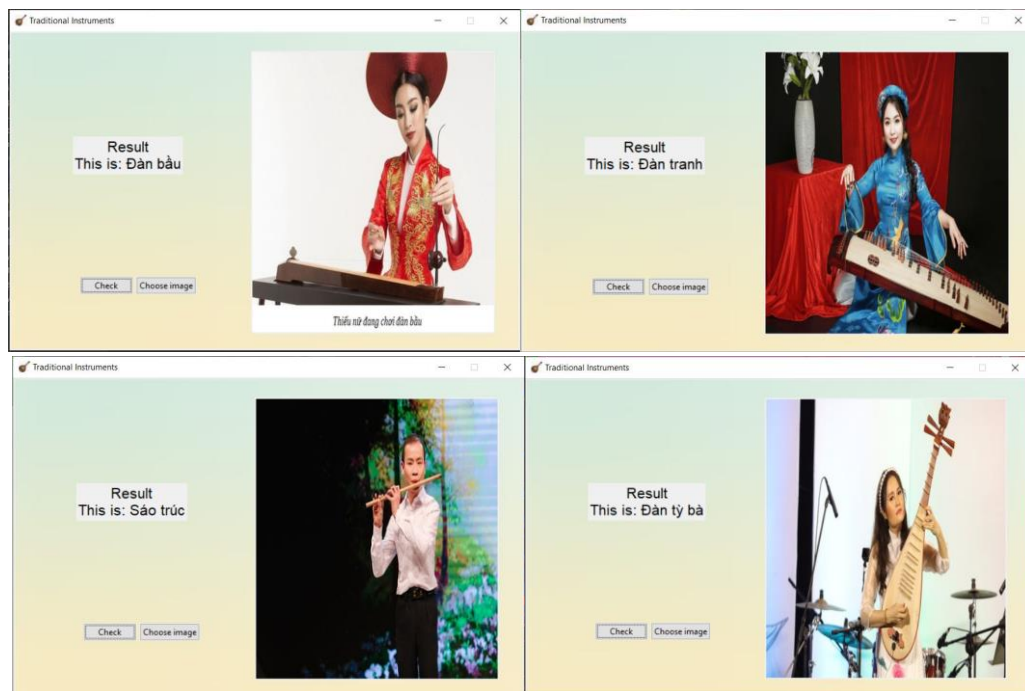
```

Hình 17. Code giao diện trên Pycharm



Hình 18. Giao diện ứng dụng

Một số kết quả của mô hình với độ chính xác 99,6%:



Hình 19. Kết quả của mô hình



## CHƯƠNG 4: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

### 4.1. Kết luận:

Đề tài "Phát hiện các loại nhạc cụ cổ truyền Việt Nam" đã áp dụng mô hình mạng CNN. Cơ sở dữ liệu để train và test cho mô hình được thu thập từ nguồn Internet. Sau khi train, mô hình có độ chính xác đạt 99,6%. Các loại nhạc cụ được phát hiện và nhận diện chính xác. Qua đề tài này, có thể thấy sơ bộ về cách để thực hiện một mô hình ứng dụng công nghệ AI đơn giản từ việc thu thập dữ liệu, xử lý dữ liệu, chọn mô hình phù hợp để có kết quả nhanh và chính xác nhất (độ chính xác cao). Từ đó, có thể triển khai, ứng dụng công nghệ AI vào những đề tài khác trong các lĩnh vực dựa trên nền tảng đã có ở bài viết này.

### 4.2. Hướng phát triển

Để ứng dụng mô hình này vào thực tiễn cần cải thiện một số điều như sau:

- Tăng số lượng loại nhạc cụ, số lượng hình ảnh của mỗi loại, chất lượng hình ảnh của cơ sở dữ liệu đầu để mô hình CNN phát hiện được nhiều loại nhạc cụ với độ chính xác cao.
- Số lần train mô hình cần tăng cao hơn để mô hình CNN có độ chính xác cao.
- Có thể sử dụng YOLO thay thế cho CNN để có thể thực hiện test realtime và có thể nhận diện nhiều loại nhạc cụ khác nhau trong thời gian thực.

### TÀI LIỆU THAM KHẢO:

[1] Trần Thị Oanh, AI là gì? Ứng dụng của trí tuệ nhân tạo trong cuộc sống. Được lấy từ: <https://meta.vn/huong-dan/tu-van/ai-la-gi-7067>

[2] Sự khác nhau giữa AI, Machine Learning và Deep Learning. Được lấy từ: <https://topdev.vn/blog/su-khac-nhau-giua-ai-machine-learning-va-deep-learning/>

[3] Thuật toán CNN là gì? Cấu trúc mạng Convolutional Neural Network. Được lấy từ: <https://topdev.vn/blog/thuat-toan-cnn-convolutional-neural-network/>

[4] Machine Learning và Deep Learning: Sự khác biệt là gì? Được lấy từ: <https://funix.edu.vn/hoi-dap-cntt/machine-learning-va-deep-learning-su-khac-biet-la-gi/>

