At first, I made the model as I saw fit. Then I ran the program it showed great accuracy but when I tried to predict, it couldn’t. so I just increased neurons and layers and found very little success. The code before the help of rocky is below.

import tensorflow as tf

from tensorflow.keras.models import Sequential

from tensorflow.keras.layers import Dense, Dropout, Activation, Flatten, Conv2D, MaxPooling2D

X = X / 255.0 #normalizing data to train models faster

model = Sequential()

model.add( Conv2D(128, (4, 4), input\_shape = X.shape[1:]) )

model.add(Activation("relu"))

model.add(MaxPooling2D(pool\_size=(2, 2)))

model.add( Conv2D(128, (4, 4)) )

model.add(Activation("relu"))

model.add(MaxPooling2D(pool\_size=(2, 2)))

#later added

model.add( Conv2D(128, (3, 3)) )

model.add(Activation("relu"))

model.add(MaxPooling2D(pool\_size=(2, 2)))

model.add( Conv2D(64, (3, 3)) )

model.add(Activation("relu"))

model.add(MaxPooling2D(pool\_size=(2, 2)))

model.add( Conv2D(64, (3, 3)) )

model.add(Activation("relu"))

#model.add(MaxPooling2D(pool\_size=(2, 2)))

#

model.add(Flatten())

model.add(Dense(256))

model.add(Activation("relu"))

model.add(Dropout(0.2))#later added

model.add(Dense(24)) #?????

model.add(Activation('sigmoid'))

model.compile(loss = "sparse\_categorical\_crossentropy",

optimizer = "adam",

metrics = ['accuracy'])

model.fit(X, y, batch\_size = 32, epochs = 20, validation\_split = 0.2)# validation split is out of sample data

After rocky helped I changed my code as instructed by him . that code is written below.

import tensorflow as tf

from tensorflow.keras.models import Sequential

from tensorflow.keras.layers import Dense, Dropout, Activation, Flatten, Conv2D, MaxPooling2D

X = X / 255.0 #normalizing data to train models faster

model = Sequential()

model.add( Conv2D(64, (4, 4), input\_shape = X.shape[1:]) ) #64 - 128 has to check

model.add(Activation("relu"))

model.add(MaxPooling2D(pool\_size=(2, 2))) #padding = same .suggested jodi problem hoy.

model.add( Conv2D(64, (4, 4)) )

model.add(Activation("relu"))

model.add(MaxPooling2D(pool\_size=(2, 2)))

#later added

model.add( Conv2D(64, (3, 3)) )

model.add(Activation("relu"))

model.add(MaxPooling2D(pool\_size=(2, 2)))

model.add( Conv2D(16, (3, 3)) )

model.add(Activation("relu"))

model.add(MaxPooling2D(pool\_size=(2, 2)))

# model.add( Conv2D(16, (3, 3)) )#16/64

# model.add(Activation("relu"))

# #model.add(MaxPooling2D(pool\_size=(2, 2)))

#

model.add(Flatten())

model.add(Dense(64)) #64 -256

model.add(Activation("relu"))

# model.add(Dropout(0.2))#later added hasnt checked copied from a youtube video

model.add(Dense(24, activation = 'softmax')) #?????

#model.add(Activation('sigmoid')) #softmax use korte hobe....in future

model.compile(loss = "sparse\_categorical\_crossentropy",

optimizer = "adam",

metrics = ['accuracy'])

model.fit(X, y, batch\_size = 32, epochs = 4, validation\_split = 0.3)# validation split is out of sample data age .1 chhilo..chang korlam

when I checked after this edit…the accuracy was great…