model3=Sequential()

model3.add(Conv2D(32,(4,4),activation='relu',padding='same',input\_shape=(48,48,1)))  #moving ahead with grayscale inputs as RGB didnt show much big of a difference

model3.add(Conv2D(32,(4,4),activation='relu',padding='same'))

model3.add(Conv2D(64,(4,4),padding='same',activation='relu'))                        #removed maxpooling layers

model3.add(Conv2D(64,(4,4),padding='same',activation='relu'))                        #2 layers of conv2D with 64 filters and kernel size increased to (4,4) to capture bigger details

model3.add(Conv2D(64,(4,4),padding='same',activation='relu'))                        #removed maxpooling layers

model3.add(Conv2D(64,(4,4),padding='same',activation='relu'))                        #2 layers of conv2D with 64 filters and kernel size increased to (4,4) to capture bigger details

model3.add(MaxPooling2D(2,2))

model3.add(Conv2D(128,(4,4),padding='same',activation='relu'))                       #2 layers of conv2D with 128 filters

model3.add(Conv2D(128,(4,4),padding='same',activation='relu'))

model3.add(Conv2D(128,(4,4),padding='same',activation='relu'))                       #2 layers of conv2D with 128 filters

model3.add(Conv2D(128,(4,4),padding='same',activation='relu'))

model3.add(MaxPooling2D(2,2))

model3.add(Conv2D(256,(4,4),padding='same',activation='relu'))                       #2 layers of conv2D with 256 filters

model3.add(Conv2D(256,(4,4),padding='same',activation='relu'))

model3.add(Conv2D(256,(4,4),padding='same',activation='relu'))                       #2 layers of conv2D with 256 filters

model3.add(Conv2D(256,(4,4),padding='same',activation='relu'))

model3.add(MaxPooling2D(2,2))

model3.add(Flatten())

model3.add(Dropout(0.2))                                              #to correct the overfitting that may be caused

model3.add(BatchNormalization())

model3.add(Dense(32,activation='relu'))

model3.add(Dropout(0.2))                                              #to correct the overfitting that may be caused

model3.add(BatchNormalization())

model3.add(Dense(32,activation='relu'))

model3.add(Dense(4,activation="Softmax"))

model3.summary()