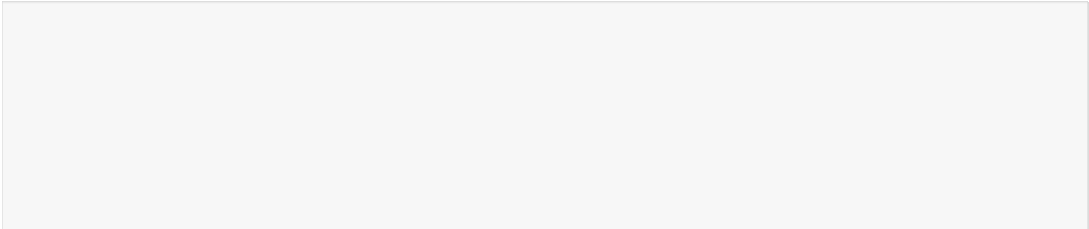
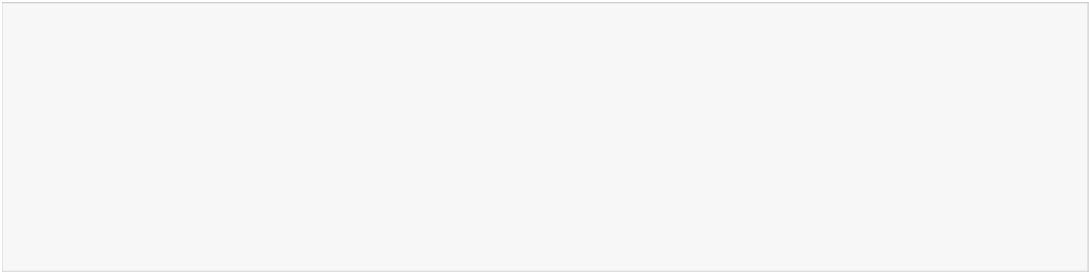
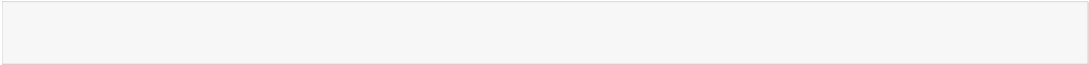
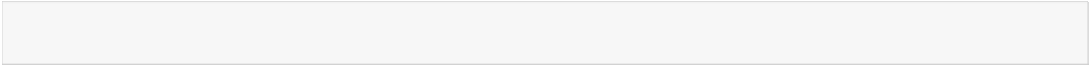
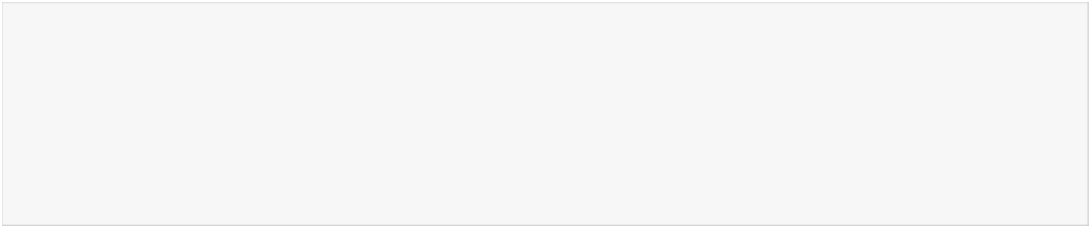
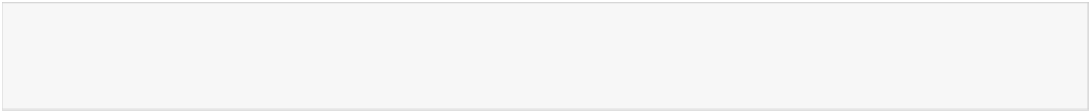
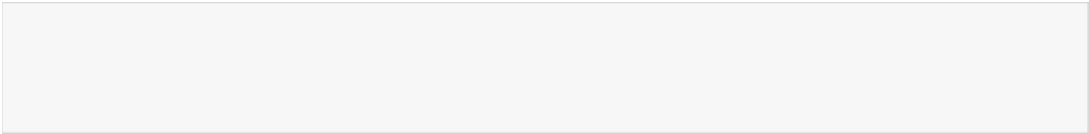
TEAM ID:PNT2022TMID30495



TITLE:classification of arrhythmia by using deep learning with 2-D ECG spectral image representation

Date:19November 2022

In [1]:

# #Image Augmentation

from tensorflow.keras.preprocessing.image import ImageDataGenerator rain\_datagen = ImageDataGenerator(rescale=1./255,

zoom\_range=0.2, horizontal\_flip=True)

In [2]:

# #Create Model #Importing req library

from tensorflow.keras.models import Sequential

from tensorflow.keras.layers import Convolution2D, MaxPooling2D, Flatten, Dense

In [3]:

# #Add Layers

model = Sequential() # Initializing sequential model model.add(Convolution2D(32,(3,3),activation='relu',input\_shape=(64,64,3))) # convolution layer

model.add(MaxPooling2D(pool\_size=(2, 2))) # Max pooling layer model.add(Flatten()) # Flatten layer model.add(Dense(300,activation='relu')) # Hidden layer 1 model.add(Dense(150,activation='relu')) # Hidden layer 2 model.add(Dense(4,activation='softmax')) # Output layer

In [4]:

# #Compile the model

model.compile(optimizer='adam',loss='categorical\_crossentropy',metrics=['accuracy'])

In [5]:

# #Save the model

model.save('ECG.h5')

In [7]:

# #Test the model #Testing 1

from tensorflow.keras.preprocessing import image import numpy as np

img = image.load\_img('/content/fig\_2113.png',target\_size=(64,64)) # Reading image

x = image.img\_to\_array(img) x = np.expand\_dims(x,axis=0)

pred = np.argmax(model.predict(x))

op = ['Normal','Left Bundle Branch Block','Right Bundle Branch Block','Premature Atrial Contraction','Premature Ventricular Contraction','Ventricular Fibrillation']

op[pred]

1/1 [==============================] - 0s 25ms/step Out[7]:

'Normal'

In [9]:

# #Test the model #Testing 2

from tensorflow.keras.preprocessing import image import numpy as np

img = image.load\_img('/content/fig\_5898.png',target\_size=(64,64)) # Reading image

x = image.img\_to\_array(img) x = np.expand\_dims(x,axis=0)

pred = np.argmax(model.predict(x))

op = ['Left Bundle Branch Block','Normal','Right Bundle Branch Block','Premature Atrial



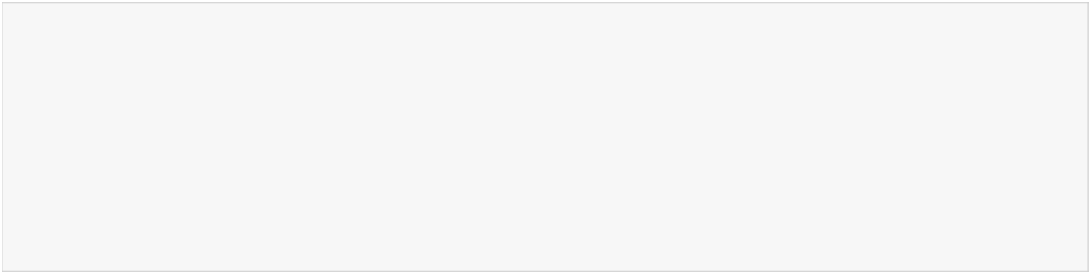


Contraction','Premature Ventricular Contraction','Ventricular Fibrillation'] op[pred]

1/1 [==============================] - 0s 27ms/step Out[9]:

'Left Bundle Branch Block'

In [10]:



#Test the model #Testing 3

from tensorflow.keras.preprocessing import image import numpy as np

img = image.load\_img('/content/fig\_12.png',target\_size=(64,64)) # Reading image

x = image.img\_to\_array(img) x = np.expand\_dims(x,axis=0)

pred = np.argmax(model.predict(x))

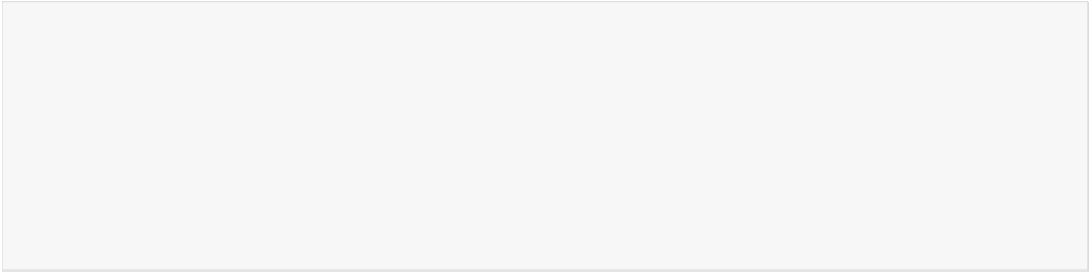
op = ['Right Bundle Branch Block','Normal','Left Bundle Branch Block','Premature Atrial Contraction','Premature Ventricular Contraction','Ventricular Fibrillation']

op[pred]

1/1 [==============================] - 0s 25ms/step Out[10]:

'Right Bundle Branch Block'

In [12]:



#Test the model #Testing 4

from tensorflow.keras.preprocessing import image import numpy as np

img = image.load\_img('/content/fig\_24.png',target\_size=(64,64)) # Reading image

x = image.img\_to\_array(img) x = np.expand\_dims(x,axis=0)

pred = np.argmax(model.predict(x))

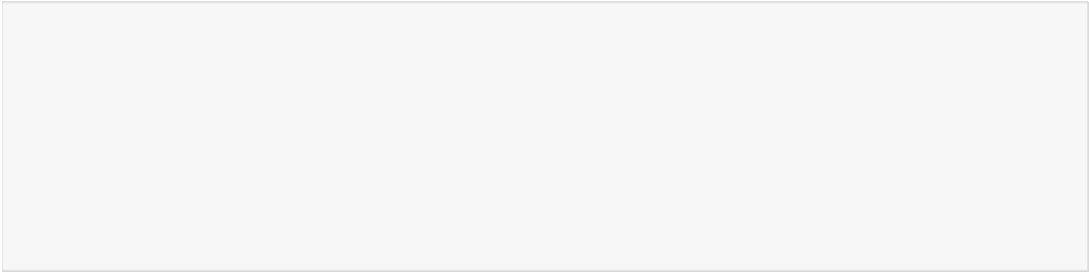
op = ['Premature Atrial Contraction','Normal','Left Bundle Branch Block','Right Bundle B ranch Block','Premature Ventricular Contraction','Ventricular Fibrillation']

op[pred]

1/1 [==============================] - 0s 24ms/step Out[12]:

'Premature Atrial Contraction'

In [13]:



#Test the model #Testing 5

from tensorflow.keras.preprocessing import image import numpy as np

img = image.load\_img('/content/fig\_5656.png',target\_size=(64,64)) # Reading image

x = image.img\_to\_array(img) x = np.expand\_dims(x,axis=0)

pred = np.argmax(model.predict(x))

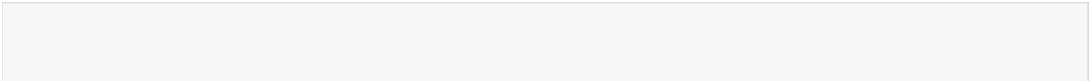
op = ['Premature Ventricular Contraction','Normal','Left Bundle Branch Block','Right Bun dle Branch Block','Premature Atrial Contraction','Ventricular Fibrillation']

op[pred]

1/1 [==============================] - 0s 25ms/step Out[13]:

'Premature Ventricular Contraction'

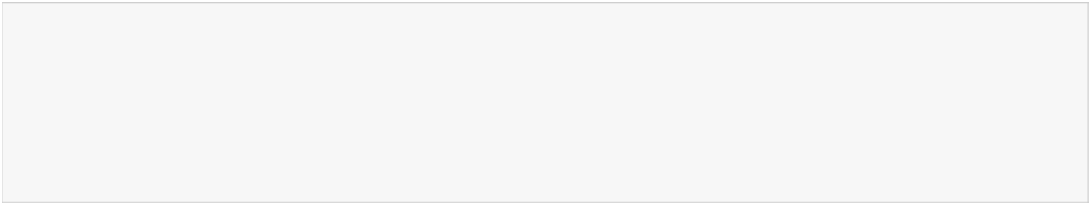
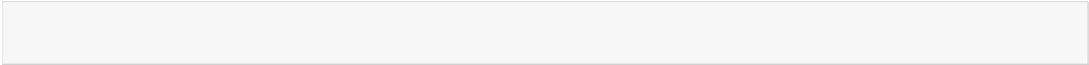
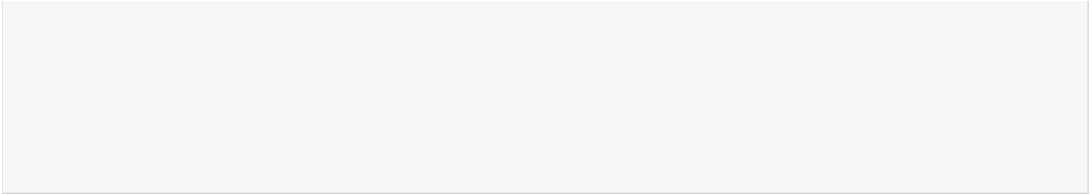
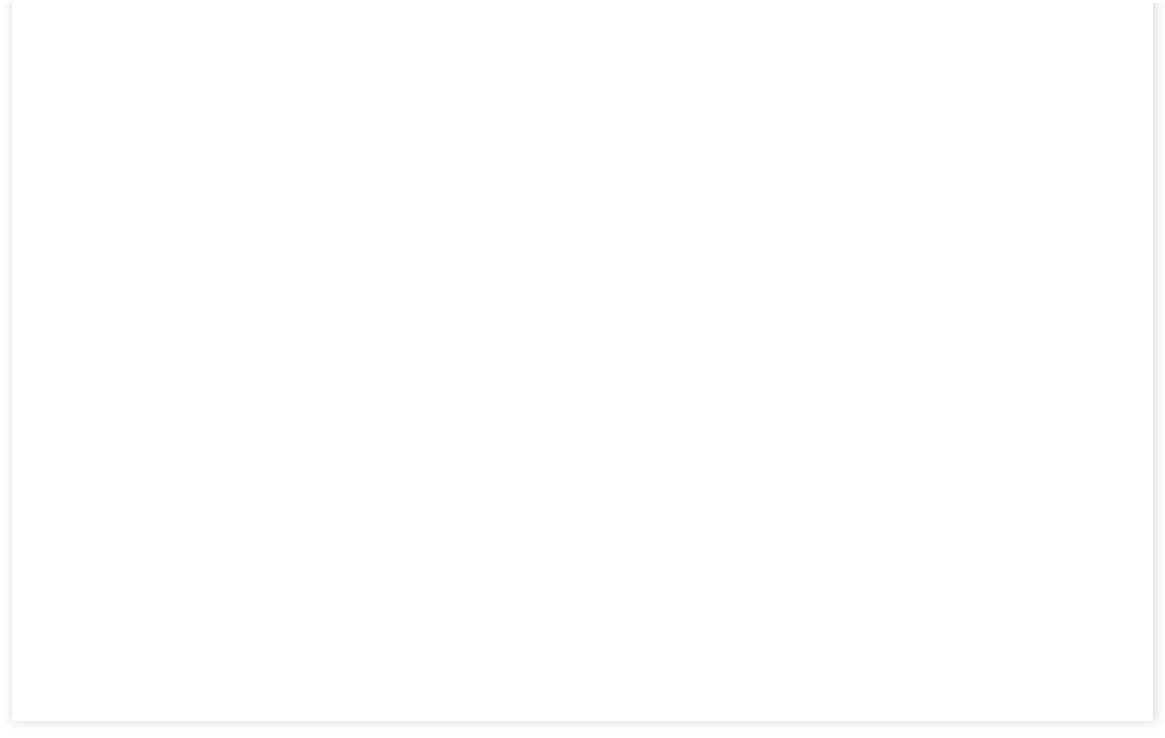
In [14]:



#Test the model #Testing 6

from tensorflow.keras.preprocessing import image

import numpy as np



img = image.load\_img('/content/VFEfig\_122.png',target\_size=(64,64)) # Reading image

x = image.img\_to\_array(img) x = np.expand\_dims(x,axis=0)

pred = np.argmax(model.predict(x))

op = ['Ventricular Fibrillation','Normal','Left Bundle Branch Block','Right Bundle Branc h Block','Premature Atrial Contraction','Premature Ventricular Contraction']

op[pred]

1/1 [==============================] - 0s 31ms/step Out[14]:

'Ventricular Fibrillation'

In [15]:

# #Train the model

from tensorflow.keras.callbacks import EarlyStopping, ReduceLROnPlateau

In [16]:

early\_stop = EarlyStopping(monitor='val\_accuracy',

patience=5)

lr = ReduceLROnPlateau(monitor='val\_accuaracy',

factor=0.5, min\_lr=0.00001)

callback = [early\_stop,lr]