Classifying sign language digits using modified mobile nets model and plotting the results on confusion matrix

In [1]:

```python
import numpy as np
import tensorflow as tf
from tensorflow import keras
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Activation, Dense, Flatten, BatchNormalization, Conv2D, MaxPool2D
from tensorflow.keras.optimizers import Adam
from tensorflow.keras.metrics import categorical_crossentropy
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from sklearn.metrics import confusion_matrix
from keras.preprocessing import image
import itertools
import os
from keras.applications import imagenet_utils
import shutil
from keras.applications.vgg16 import preprocess_input, decode_predictions
import random
import glob
import matplotlib.pyplot as plt
import warnings
#from IPython.display import Image
#from PIL import Image
warnings.simplefilter(action='ignore', category=FutureWarning)
%matplotlib inline
```

In [17]:

```python
#Organize data into train , valid and test
os.chdir('C:\\Users\\User\\Desktop\\Projects\\Signlanguage')
if os.path.isdir('train/0') is False:
    os.mkdir('train')
    os.mkdir('valid')
    os.mkdir('test')

    for i in range (0,10):
        shutil.move(f'{i}','train')
        os.mkdir(f'valid/{i}')
        os.mkdir(f'test/{i}')

        valid_samples=random.sample(os.listdir(f'train/{i}'),30)
        for j in valid_samples:
            shutil.move(f'train/{i}/{j}', f'valid/{i}')

        test_samples=random.sample(os.listdir(f'train/{i}'),5)
        for k in test_samples:
            shutil.move(f'train/{i}/{k}', f'test/{i}')

os.chdir('../..')
```

In [18]:

```python
 train_path='C:\\Users\\User\\Desktop\\Projects\\Signlanguage\\train'
valid_path='C:\\Users\\User\\Desktop\\Projects\\Signlanguage\\valid'
test_path='C:\\Users\\User\\Desktop\\Projects\\Signlanguage\\test'
```

In [19]:

```python
train_batches = ImageDataGenerator(preprocessing_function=tf.keras.applications.mobilenet.preprocess_input) \
    .flow_from_directory(directory=train_path, target_size=(224,224),  batch_size=10)
valid_batches = ImageDataGenerator(preprocessing_function=tf.keras.applications.mobilenet.preprocess_input) \
    .flow_from_directory(directory=valid_path, target_size=(224,224), batch_size=10)
test_batches = ImageDataGenerator(preprocessing_function=tf.keras.applications.mobilenet.preprocess_input) \
    .flow_from_directory(directory=test_path, target_size=(224,224),  batch_size=10,shuffle=False)
```

```
Found 1712 images belonging to 10 classes.
Found 300 images belonging to 10 classes.
Found 50 images belonging to 10 classes.
```

In [20]:

```python
assert train_batches.n==1712
assert valid_batches.n==300
assert test_batches.n==50
assert train_batches.num_classes==valid_batches.num_classes==test_batches.num_classes==10
```

In [21]:

```python
mobile=tf.keras.applications.mobilenet.MobileNet()
```

In [22]:

```python
mobile.summary()
```

```
Model: "mobilenet_1.00_224"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
input_2 (InputLayer)         [(None, 224, 224, 3)]     0
conv1 (Conv2D)               (None, 112, 112, 32)      864
conv1_bn (BatchNormalization (None, 112, 112, 32)      128
conv1_relu (ReLU)            (None, 112, 112, 32)      0
conv_dw_1 (DepthwiseConv2D)  (None, 112, 112, 32)      288
conv_dw_1_bn (BatchNormaliza (None, 112, 112, 32)      128
conv_dw_1_relu (ReLU)        (None, 112, 112, 32)      0
conv_pw_1 (Conv2D)           (None, 112, 112, 64)      2048
conv_pw_1_bn (BatchNormaliza (None, 112, 112, 64)      256
conv_pw_1_relu (ReLU)        (None, 112, 112, 64)      0
conv_pad_2 (ZeroPadding2D)   (None, 113, 113, 64)      0
conv_dw_2 (DepthwiseConv2D)  (None, 56, 56, 64)        576
conv_dw_2_bn (BatchNormaliza (None, 56, 56, 64)        256
conv_dw_2_relu (ReLU)        (None, 56, 56, 64)        0
conv_pw_2 (Conv2D)           (None, 56, 56, 128)       8192
conv_pw_2_bn (BatchNormaliza (None, 56, 56, 128)       512
conv_pw_2_relu (ReLU)        (None, 56, 56, 128)       0
conv_dw_3 (DepthwiseConv2D)  (None, 56, 56, 128)       1152
conv_dw_3_bn (BatchNormaliza (None, 56, 56, 128)       512
conv_dw_3_relu (ReLU)        (None, 56, 56, 128)       0
conv_pw_3 (Conv2D)           (None, 56, 56, 128)       16384
conv_pw_3_bn (BatchNormaliza (None, 56, 56, 128)       512
conv_pw_3_relu (ReLU)        (None, 56, 56, 128)       0
conv_pad_4 (ZeroPadding2D)   (None, 57, 57, 128)       0
conv_dw_4 (DepthwiseConv2D)  (None, 28, 28, 128)       1152
conv_dw_4_bn (BatchNormaliza (None, 28, 28, 128)       512
conv_dw_4_relu (ReLU)        (None, 28, 28, 128)       0
conv_pw_4 (Conv2D)           (None, 28, 28, 256)       32768
conv_pw_4_bn (BatchNormaliza (None, 28, 28, 256)       1024
conv_pw_4_relu (ReLU)        (None, 28, 28, 256)       0
conv_dw_5 (DepthwiseConv2D)  (None, 28, 28, 256)       2304
conv_dw_5_bn (BatchNormaliza (None, 28, 28, 256)       1024
conv_dw_5_relu (ReLU)        (None, 28, 28, 256)       0
conv_pw_5 (Conv2D)           (None, 28, 28, 256)       65536
conv_pw_5_bn (BatchNormaliza (None, 28, 28, 256)       1024
conv_pw_5_relu (ReLU)        (None, 28, 28, 256)       0
conv_pad_6 (ZeroPadding2D)   (None, 29, 29, 256)       0
conv_dw_6 (DepthwiseConv2D)  (None, 14, 14, 256)       2304
conv_dw_6_bn (BatchNormaliza (None, 14, 14, 256)       1024
conv_dw_6_relu (ReLU)        (None, 14, 14, 256)       0
conv_pw_6 (Conv2D)           (None, 14, 14, 512)       131072
conv_pw_6_bn (BatchNormaliza (None, 14, 14, 512)       2048
conv_pw_6_relu (ReLU)        (None, 14, 14, 512)       0
conv_dw_7 (DepthwiseConv2D)  (None, 14, 14, 512)       4608
conv_dw_7_bn (BatchNormaliza (None, 14, 14, 512)       2048
conv_dw_7_relu (ReLU)        (None, 14, 14, 512)       0
conv_pw_7 (Conv2D)           (None, 14, 14, 512)       262144
conv_pw_7_bn (BatchNormaliza (None, 14, 14, 512)       2048
conv_pw_7_relu (ReLU)        (None, 14, 14, 512)       0
conv_dw_8 (DepthwiseConv2D)  (None, 14, 14, 512)       4608
conv_dw_8_bn (BatchNormaliza (None, 14, 14, 512)       2048
conv_dw_8_relu (ReLU)        (None, 14, 14, 512)       0
conv_pw_8 (Conv2D)           (None, 14, 14, 512)       262144
conv_pw_8_bn (BatchNormaliza (None, 14, 14, 512)       2048
conv_pw_8_relu (ReLU)        (None, 14, 14, 512)       0
conv_dw_9 (DepthwiseConv2D)  (None, 14, 14, 512)       4608
conv_dw_9_bn (BatchNormaliza (None, 14, 14, 512)       2048
conv_dw_9_relu (ReLU)        (None, 14, 14, 512)       0
conv_pw_9 (Conv2D)           (None, 14, 14, 512)       262144
conv_pw_9_bn (BatchNormaliza (None, 14, 14, 512)       2048
conv_pw_9_relu (ReLU)        (None, 14, 14, 512)       0
conv_dw_10 (DepthwiseConv2D) (None, 14, 14, 512)       4608
conv_dw_10_bn (BatchNormaliz (None, 14, 14, 512)       2048
conv_dw_10_relu (ReLU)       (None, 14, 14, 512)       0
conv_pw_10 (Conv2D)          (None, 14, 14, 512)       262144
conv_pw_10_bn (BatchNormaliz (None, 14, 14, 512)       2048
conv_pw_10_relu (ReLU)       (None, 14, 14, 512)       0
conv_dw_11 (DepthwiseConv2D) (None, 14, 14, 512)       4608
conv_dw_11_bn (BatchNormaliz (None, 14, 14, 512)       2048
conv_dw_11_relu (ReLU)       (None, 14, 14, 512)       0
conv_pw_11 (Conv2D)          (None, 14, 14, 512)       262144
conv_pw_11_bn (BatchNormaliz (None, 14, 14, 512)       2048
conv_pw_11_relu (ReLU)       (None, 14, 14, 512)       0
conv_pad_12 (ZeroPadding2D)  (None, 15, 15, 512)       0
conv_dw_12 (DepthwiseConv2D) (None, 7, 7, 512)         4608
conv_dw_12_bn (BatchNormaliz (None, 7, 7, 512)         2048
conv_dw_12_relu (ReLU)       (None, 7, 7, 512)         0
conv_pw_12 (Conv2D)          (None, 7, 7, 1024)        524288
conv_pw_12_bn (BatchNormaliz (None, 7, 7, 1024)        4096
conv_pw_12_relu (ReLU)       (None, 7, 7, 1024)        0
conv_dw_13 (DepthwiseConv2D) (None, 7, 7, 1024)        9216
conv_dw_13_bn (BatchNormaliz (None, 7, 7, 1024)        4096
conv_dw_13_relu (ReLU)       (None, 7, 7, 1024)        0
conv_pw_13 (Conv2D)          (None, 7, 7, 1024)        1048576
conv_pw_13_bn (BatchNormaliz (None, 7, 7, 1024)        4096
conv_pw_13_relu (ReLU)       (None, 7, 7, 1024)        0
global_average_pooling2d_1 ( (None, 1024)              0
reshape_1 (Reshape)          (None, 1, 1, 1024)        0
dropout (Dropout)            (None, 1, 1, 1024)        0
conv_preds (Conv2D)          (None, 1, 1, 1000)        1025000
reshape_2 (Reshape)          (None, 1000)              0
predictions (Activation)     (None, 1000)              0
=================================================================
Total params: 4,253,864
Trainable params: 4,231,976
Non-trainable params: 21,888
_________________________________________________________________
```

Modifing Model

In [23]:

```python
x=mobile.layers[-6].output
output=Dense(units=10, activation='softmax')(x)
```

In [26]:

```python
model=keras.Model(inputs=mobile.input, outputs=output)
```

In [29]:

```python
for layer in model.layers[:-23]:
    layer.trainable=False
```

In [30]:

```python
model.summary()
```

```
Model: "model"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
input_2 (InputLayer)         [(None, 224, 224, 3)]     0
conv1 (Conv2D)               (None, 112, 112, 32)      864
conv1_bn (BatchNormalization (None, 112, 112, 32)      128
conv1_relu (ReLU)            (None, 112, 112, 32)      0
conv_dw_1 (DepthwiseConv2D)  (None, 112, 112, 32)      288
conv_dw_1_bn (BatchNormaliza (None, 112, 112, 32)      128
conv_dw_1_relu (ReLU)        (None, 112, 112, 32)      0
conv_pw_1 (Conv2D)           (None, 112, 112, 64)      2048
conv_pw_1_bn (BatchNormaliza (None, 112, 112, 64)      256
conv_pw_1_relu (ReLU)        (None, 112, 112, 64)      0
conv_pad_2 (ZeroPadding2D)   (None, 113, 113, 64)      0
conv_dw_2 (DepthwiseConv2D)  (None, 56, 56, 64)        576
conv_dw_2_bn (BatchNormaliza (None, 56, 56, 64)        256
conv_dw_2_relu (ReLU)        (None, 56, 56, 64)        0
conv_pw_2 (Conv2D)           (None, 56, 56, 128)       8192
conv_pw_2_bn (BatchNormaliza (None, 56, 56, 128)       512
conv_pw_2_relu (ReLU)        (None, 56, 56, 128)       0
conv_dw_3 (DepthwiseConv2D)  (None, 56, 56, 128)       1152
conv_dw_3_bn (BatchNormaliza (None, 56, 56, 128)       512
conv_dw_3_relu (ReLU)        (None, 56, 56, 128)       0
conv_pw_3 (Conv2D)           (None, 56, 56, 128)       16384
conv_pw_3_bn (BatchNormaliza (None, 56, 56, 128)       512
conv_pw_3_relu (ReLU)        (None, 56, 56, 128)       0
conv_pad_4 (ZeroPadding2D)   (None, 57, 57, 128)       0
conv_dw_4 (DepthwiseConv2D)  (None, 28, 28, 128)       1152
conv_dw_4_bn (BatchNormaliza (None, 28, 28, 128)       512
conv_dw_4_relu (ReLU)        (None, 28, 28, 128)       0
conv_pw_4 (Conv2D)           (None, 28, 28, 256)       32768
conv_pw_4_bn (BatchNormaliza (None, 28, 28, 256)       1024
conv_pw_4_relu (ReLU)        (None, 28, 28, 256)       0
conv_dw_5 (DepthwiseConv2D)  (None, 28, 28, 256)       2304
conv_dw_5_bn (BatchNormaliza (None, 28, 28, 256)       1024
conv_dw_5_relu (ReLU)        (None, 28, 28, 256)       0
conv_pw_5 (Conv2D)           (None, 28, 28, 256)       65536
conv_pw_5_bn (BatchNormaliza (None, 28, 28, 256)       1024
conv_pw_5_relu (ReLU)        (None, 28, 28, 256)       0
conv_pad_6 (ZeroPadding2D)   (None, 29, 29, 256)       0
conv_dw_6 (DepthwiseConv2D)  (None, 14, 14, 256)       2304
conv_dw_6_bn (BatchNormaliza (None, 14, 14, 256)       1024
conv_dw_6_relu (ReLU)        (None, 14, 14, 256)       0
conv_pw_6 (Conv2D)           (None, 14, 14, 512)       131072
conv_pw_6_bn (BatchNormaliza (None, 14, 14, 512)       2048
conv_pw_6_relu (ReLU)        (None, 14, 14, 512)       0
conv_dw_7 (DepthwiseConv2D)  (None, 14, 14, 512)       4608
conv_dw_7_bn (BatchNormaliza (None, 14, 14, 512)       2048
conv_dw_7_relu (ReLU)        (None, 14, 14, 512)       0
conv_pw_7 (Conv2D)           (None, 14, 14, 512)       262144
conv_pw_7_bn (BatchNormaliza (None, 14, 14, 512)       2048
conv_pw_7_relu (ReLU)        (None, 14, 14, 512)       0
conv_dw_8 (DepthwiseConv2D)  (None, 14, 14, 512)       4608
conv_dw_8_bn (BatchNormaliza (None, 14, 14, 512)       2048
conv_dw_8_relu (ReLU)        (None, 14, 14, 512)       0
conv_pw_8 (Conv2D)           (None, 14, 14, 512)       262144
conv_pw_8_bn (BatchNormaliza (None, 14, 14, 512)       2048
conv_pw_8_relu (ReLU)        (None, 14, 14, 512)       0
conv_dw_9 (DepthwiseConv2D)  (None, 14, 14, 512)       4608
conv_dw_9_bn (BatchNormaliza (None, 14, 14, 512)       2048
conv_dw_9_relu (ReLU)        (None, 14, 14, 512)       0
conv_pw_9 (Conv2D)           (None, 14, 14, 512)       262144
conv_pw_9_bn (BatchNormaliza (None, 14, 14, 512)       2048
conv_pw_9_relu (ReLU)        (None, 14, 14, 512)       0
conv_dw_10 (DepthwiseConv2D) (None, 14, 14, 512)       4608
conv_dw_10_bn (BatchNormaliz (None, 14, 14, 512)       2048
conv_dw_10_relu (ReLU)       (None, 14, 14, 512)       0
conv_pw_10 (Conv2D)          (None, 14, 14, 512)       262144
conv_pw_10_bn (BatchNormaliz (None, 14, 14, 512)       2048
conv_pw_10_relu (ReLU)       (None, 14, 14, 512)       0
conv_dw_11 (DepthwiseConv2D) (None, 14, 14, 512)       4608
conv_dw_11_bn (BatchNormaliz (None, 14, 14, 512)       2048
conv_dw_11_relu (ReLU)       (None, 14, 14, 512)       0
conv_pw_11 (Conv2D)          (None, 14, 14, 512)       262144
conv_pw_11_bn (BatchNormaliz (None, 14, 14, 512)       2048
conv_pw_11_relu (ReLU)       (None, 14, 14, 512)       0
conv_pad_12 (ZeroPadding2D)  (None, 15, 15, 512)       0
conv_dw_12 (DepthwiseConv2D) (None, 7, 7, 512)         4608
conv_dw_12_bn (BatchNormaliz (None, 7, 7, 512)         2048
conv_dw_12_relu (ReLU)       (None, 7, 7, 512)         0
conv_pw_12 (Conv2D)          (None, 7, 7, 1024)        524288
conv_pw_12_bn (BatchNormaliz (None, 7, 7, 1024)        4096
conv_pw_12_relu (ReLU)       (None, 7, 7, 1024)        0
conv_dw_13 (DepthwiseConv2D) (None, 7, 7, 1024)        9216
conv_dw_13_bn (BatchNormaliz (None, 7, 7, 1024)        4096
conv_dw_13_relu (ReLU)       (None, 7, 7, 1024)        0
conv_pw_13 (Conv2D)          (None, 7, 7, 1024)        1048576
conv_pw_13_bn (BatchNormaliz (None, 7, 7, 1024)        4096
conv_pw_13_relu (ReLU)       (None, 7, 7, 1024)        0
global_average_pooling2d_1 ( (None, 1024)              0
dense (Dense)                (None, 10)                10250
=================================================================
Total params: 3,239,114
Trainable params: 1,873,930
Non-trainable params: 1,365,184
_________________________________________________________________
```

In [31]:

```python
model.compile(optimizer=Adam(learning_rate=0.0001), loss='categorical_crossentropy', metrics=['accuracy'])
```

In [32]:

```python
model.fit(x=train_batches, validation_data=valid_batches,epochs=10,verbose=2)
```

```
Epoch 1/10
172/172 - 245s - loss: 0.6597 - accuracy: 0.8072 - val_loss: 0.2458 - val_accuracy: 0.9367
Epoch 2/10
172/172 - 220s - loss: 0.1166 - accuracy: 0.9836 - val_loss: 0.1046 - val_accuracy: 0.9733
Epoch 3/10
172/172 - 220s - loss: 0.0711 - accuracy: 0.9912 - val_loss: 0.0692 - val_accuracy: 0.9767
Epoch 4/10
172/172 - 220s - loss: 0.0366 - accuracy: 0.9982 - val_loss: 0.0533 - val_accuracy: 0.9767
Epoch 5/10
172/172 - 217s - loss: 0.0239 - accuracy: 0.9988 - val_loss: 0.0525 - val_accuracy: 0.9833
Epoch 6/10
172/172 - 225s - loss: 0.0174 - accuracy: 0.9988 - val_loss: 0.0423 - val_accuracy: 0.9833
Epoch 7/10
172/172 - 221s - loss: 0.0216 - accuracy: 0.9977 - val_loss: 0.0505 - val_accuracy: 0.9833
Epoch 8/10
172/172 - 221s - loss: 0.0160 - accuracy: 0.9988 - val_loss: 0.0528 - val_accuracy: 0.9800
Epoch 9/10
172/172 - 215s - loss: 0.0109 - accuracy: 0.9988 - val_loss: 0.0342 - val_accuracy: 0.9833
Epoch 10/10
172/172 - 214s - loss: 0.0069 - accuracy: 1.0000 - val_loss: 0.0315 - val_accuracy: 0.9867
```

Out[32]: <tensorflow.python.keras.callbacks.History at 0x25502b12d00>

Predict sign language digits

In [33]:

```python
test_labels=test_batches.classes
```

In [34]:

```python
predictions=model.predict(x=test_batches, verbose=0)
```

In [37]:

```python
cm=confusion_matrix(y_true=test_labels,y_pred=predictions.argmax(axis=1))
```

In [42]:

```python
def plot_confusion_matrix(cm, classes,
                          normalize=False,
                          title='Confusion matrix',
                          cmap=plt.cm.Blues):
    """
    This function prints and plots the confusion matrix.
    Normalization can be applied by setting `normalize=True`.
    """
    plt.imshow(cm, interpolation='nearest', cmap=cmap)
    plt.title(title)
    plt.colorbar()
    tick_marks = np.arange(len(classes))
    plt.xticks(tick_marks, classes, rotation=45)
    plt.yticks(tick_marks, classes)

    if normalize:
        cm = cm.astype('float') / cm.sum(axis=1)[:, np.newaxis]
        print("Normalized confusion matrix")
    else:
        print('Confusion matrix, without normalization')

    print(cm)

    thresh = cm.max() / 2.
    for i, j in itertools.product(range(cm.shape[0]), range(cm.shape[1])):
        plt.text(j, i, cm[i, j],
                 horizontalalignment="center",
                 color="white" if cm[i, j] > thresh else "black")

    plt.tight_layout()
    plt.ylabel('True label')
    plt.xlabel('Predicted label')
```

In [43]:

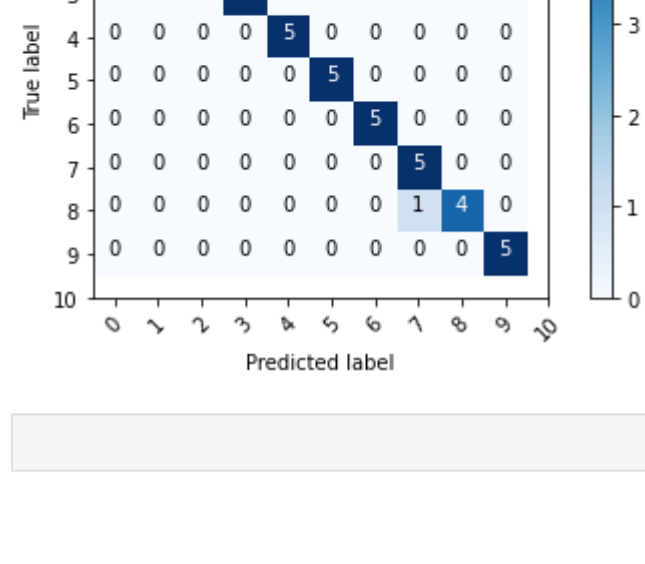
```python
test_batches.class_indices
```

```
Out[43]: {'0': 0,
 '1': 1,
 '2': 2,
 '3': 3,
 '4': 4,
 '5': 5,
 '6': 6,
 '7': 7,
 '8': 8,
 '9': 9}
```

In [45]:

```python
cm_plot_labels=['0','1','2','3','4','5','6','7','8','9','10']
plot_confusion_matrix(cm=cm, classes=cm_plot_labels,title='confusion Matrix')
```

```
Confusion matrix, without normalization
[[5 0 0 0 0 0 0 0 0 0]
 [0 5 0 0 0 0 0 0 0 0]
 [0 0 5 0 0 0 0 0 0 0]
 [0 0 0 5 0 0 0 0 0 0]
 [0 0 0 0 5 0 0 0 0 0]
 [0 0 0 0 0 5 0 0 0 0]
 [0 0 0 0 0 0 5 0 0 0]
 [0 0 0 0 0 0 0 5 0 0]
 [0 0 0 0 0 0 0 1 4 0]
 [0 0 0 0 0 0 0 0 0 5]]
```



In [ ]: