In [1]:
```
import torch
import cv2
from ultralytics import YOLO
import matplotlib.pyplot as plt
import glob
from datetime import datetime

```

```
C:\Users\s.ezati\Anaconda3\envs\CharSeg\lib\site-packages\numpy\_distributor_
init.py:30: UserWarning: loaded more than 1 DLL from .libs:
C:\Users\s.ezati\Anaconda3\envs\CharSeg\lib\site-packages\numpy\.libs\libopen
blas64__v0.3.21-gcc_10_3_0.dll
C:\Users\s.ezati\Anaconda3\envs\CharSeg\lib\site-packages\numpy\.libs\libopen
blas64__v0.3.23-gcc_10_3_0.dll
  warnings.warn("loaded more than 1 DLL from .libs:"
```

In [2]:
```
image_path = 'output/test/*/*.jpg'
#  image_path[12:-6]
# int('4')
```

In [3]:
```
model = YOLO('farsi-alphabet-classification-best.pt')#YOLO('runs/classify/
```

In [7]:

```
start = datetime.now()
count = 0
label = []
predict = []
name = 30
for img in glob.glob(image_path):
    label_name = img.split("\\")
    image = cv2.imread(img)
#     print(image.shape)
    im_name = label_name[1]

    if im_name == 'PWD':
        name = 10
    elif im_name == 'alef':
        name = 11
    elif im_name == 'be':
        name = 12
    elif im_name == 'dal':
        name = 13
    elif im_name == 'ein':
        name = 14
    elif im_name == 'h':
        name = 15
    elif im_name == 'jim':
        name = 16
    elif im_name == 'lam':
        name = 17
    elif im_name == 'mim':
        name = 18
    elif im_name == 'nun':
        name = 19
    elif im_name == 'pe':
        name = 20
    elif im_name == 'qaf':
        name = 21
    elif im_name == 'sin':
        name = 22
    elif im_name == 'ta':
        name = 23
    elif im_name == 'taxi':
        name = 24
    elif im_name == 'vav':
        name = 25
    elif im_name == 'ye':
        name = 26
    else:
        name = int(im_name)

    label.append(name)
#     print(name)
    res = model.predict(image )
#     print(res[0].probs.top1)
    pred_y = res[0].probs.top1

    if pred_y == 'PWD':
        pred = 10
    elif pred_y == 'alef':
```

```
        pred = 11
    elif pred_y == 'be':
        pred = 12
    elif pred_y == 'dal':
        pred = 13
    elif pred_y == 'ein':
        pred = 14
    elif pred_y == 'h':
        pred = 15
    elif pred_y == 'jim':
        pred = 16
    elif pred_y == 'lam':
        pred = 17
    elif pred_y == 'mim':
        pred = 18
    elif pred_y == 'nun':
        pred = 19
    elif pred_y == 'pe':
        pred = 20
    elif pred_y == 'qaf':
        pred = 21
    elif pred_y == 'sin':
        pred = 22
    elif pred_y == 'ta':
        pred = 23
    elif pred_y == 'taxi':
        pred = 24
    elif pred_y == 'vav':
        pred = 25
    elif pred_y == 'ye':
        pred = 26
    else:
        pred = int(pred_y)

    predict.append(pred)
    count = count + 1
#     print( "_____________________________________")



end = datetime.now()
print(end- start)
```

```
0: 64x64 0 1.00, dal 0.00, sin 0.00, mim 0.00, ein 0.00, 7.0ms
Speed: 0.0ms preprocess, 7.0ms inference, 0.0ms postprocess per image at sh
ape (1, 3, 64, 64)

0: 64x64 0 1.00, ein 0.00, mim 0.00, 7 0.00, vav 0.00, 7.0ms
Speed: 1.0ms preprocess, 7.0ms inference, 0.0ms postprocess per image at sh
ape (1, 3, 64, 64)

0: 64x64 0 1.00, dal 0.00, vav 0.00, h 0.00, mim 0.00, 6.0ms
Speed: 1.0ms preprocess, 6.0ms inference, 1.0ms postprocess per image at sh
ape (1, 3, 64, 64)

0: 64x64 0 1.00, 9 0.00, dal 0.00, vav 0.00, ein 0.00, 7.0ms
Speed: 1.0ms preprocess, 7.0ms inference, 0.0ms postprocess per image at sh
ape (1, 3, 64, 64)

0: 64x64 0 1.00, dal 0.00, vav 0.00, mim 0.00, 7 0.00, 7.2ms
Speed: 1.0ms preprocess, 7.2ms inference, 0.0ms postprocess per image at sh
```

In [8]:

```
print(end- start)
print (count)
```

```
0:01:00.694096
7168
```

# time:

test time : 0:01:00.694096

test dataset size = 7168

detect class for per image time = ۰/۵۰۲۲۳۲۱۴۲۸۵۷۱۴۲۹

# confussion matrix

In [22]:

```
import matplotlib.pyplot as plt
import numpy
from sklearn import metrics
%matplotlib inline

```

In [23]:

```
actual = numpy.array(label)
predicted = numpy.array(predict)
```

In [24]:
```
print(metrics.classification_report(actual, predicted))

```

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 1.00 | 1.00 | 1.00 | 261 |
| 1 | 1.00 | 1.00 | 1.00 | 266 |
| 2 | 0.97 | 0.98 | 0.98 | 263 |
| 3 | 0.99 | 0.98 | 0.98 | 269 |
| 4 | 0.99 | 0.97 | 0.98 | 274 |
| 5 | 0.99 | 1.00 | 0.99 | 255 |
| 6 | 1.00 | 1.00 | 1.00 | 290 |
| 7 | 1.00 | 1.00 | 1.00 | 287 |
| 8 | 1.00 | 0.99 | 0.99 | 268 |
| 9 | 1.00 | 1.00 | 1.00 | 267 |
| 10 | 1.00 | 1.00 | 1.00 | 251 |
| 11 | 1.00 | 1.00 | 1.00 | 263 |
| 12 | 1.00 | 1.00 | 1.00 | 262 |
| 13 | 1.00 | 1.00 | 1.00 | 264 |
| 14 | 1.00 | 1.00 | 1.00 | 265 |
| 15 | 1.00 | 1.00 | 1.00 | 267 |
| 16 | 1.00 | 1.00 | 1.00 | 261 |
| 17 | 1.00 | 1.00 | 1.00 | 264 |
| 18 | 1.00 | 1.00 | 1.00 | 264 |
| 19 | 1.00 | 1.00 | 1.00 | 267 |
| 20 | 1.00 | 1.00 | 1.00 | 263 |
| 21 | 1.00 | 1.00 | 1.00 | 259 |
| 22 | 1.00 | 1.00 | 1.00 | 259 |
| 23 | 1.00 | 1.00 | 1.00 | 266 |
| 24 | 1.00 | 1.00 | 1.00 | 266 |
| 25 | 1.00 | 1.00 | 1.00 | 264 |
| 26 | 1.00 | 1.00 | 1.00 | 263 |
| accuracy | | | 1.00 | 7168 |
| macro avg | 1.00 | 1.00 | 1.00 | 7168 |
| weighted avg | 1.00 | 1.00 | 1.00 | 7168 |

In [29]:
```
1
2 confusion_matrix = metrics.confusion_matrix(actual, predicted)
3
4 cm_display = metrics.ConfusionMatrixDisplay(confusion_matrix = confusion_m
5                                              ['0', '1', '2','3','4','5','6'
6                                               'alef', 'be', 'dal', 'ein', '
7                                               'nun', 'pe', 'qaf', 'sin', 't
8 fig, ax = plt.subplots(figsize=(15,15))
9 cm_display.plot(ax=ax, cmap='plasma')
```

Out[29]: <sklearn.metrics._plot.confusion_matrix.ConfusionMatrixDisplay at 0x2b360ba0160>