

atividade-5

November 11, 2017

```
In [528]: import os

import cv2
import numpy as np

import pylab as plt
%matplotlib inline

In [529]: path = os.getcwd() + os.sep
png = path + '../db_images/png/captcha.png'
jpeg = path + '../db_images/jpeg/captcha.jpeg'
path_images = [png, jpeg]
```

0.1 Imagens Originais e Tons de Cinzas

```
In [530]: images = []
for path_img in path_images:
    img = cv2.imread(path_img)
    img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
    it = {'title': 'Original',
          'X': img}
    images.append(it)

grays = []
for obj in images:
    gray = cv2.cvtColor(obj['X'], cv2.COLOR_BGR2GRAY)
    it = {'cmap': 'gray',
          'title': 'Original Gray',
          'X': gray}
    grays.append(it)

In [531]: imgs_and_grays = images + grays
```

1 Segmentação baseada em limiarização

```
In [532]: all_modulos = list(dir(cv2))

In [533]: [item for item in all_modulos if 'THRESH' in item]
```

```
Out [533]: ['ADAPTIVE_THRESH_GAUSSIAN_C',
            'ADAPTIVE_THRESH_MEAN_C',
            'AGAST_FEATURE_DETECTOR_THRESHOLD',
            'AgastFeatureDetector_THRESHOLD',
            'CALIB_CB_ADAPTIVE_THRESH',
            'CAP_PROP_INTELPERC_DEPTH_CONFIDENCE_THRESHOLD',
            'FAST_FEATURE_DETECTOR_THRESHOLD',
            'FastFeatureDetector_THRESHOLD',
            'THRESH_BINARY',
            'THRESH_BINARY_INV',
            'THRESH_MASK',
            'THRESH_OTSU',
            'THRESH_TOZERO',
            'THRESH_TOZERO_INV',
            'THRESH_TRIANGLE',
            'THRESH_TRUNC']
```

```
In [534]: types_thresh = [
            {'channels': True, 'name': 'cv2.ADAPTIVE_THRESH_GAUSSIAN_C', 'type': cv2.ADAPTIVE_TH},
            {'channels': True, 'name': 'cv2.ADAPTIVE_THRESH_MEAN_C', 'type': cv2.ADAPTIVE_TH},
            {'channels': True, 'name': 'cv2.CALIB_CB_ADAPTIVE_THRESH', 'type': cv2.CALIB_CB_A},
            {'channels': True, 'name': 'cv2.THRESH_BINARY', 'type': cv2.THRESH_BINARY},
            {'channels': True, 'name': 'cv2.THRESH_BINARY_INV', 'type': cv2.THRESH_BINARY_INV},
            {'channels': True, 'name': 'cv2.THRESH_MASK', 'type': cv2.THRESH_MASK},
            {'channels': False, 'name': 'cv2.THRESH_OTSU', 'type': cv2.THRESH_OTSU},
            {'channels': True, 'name': 'cv2.THRESH_TOZERO', 'type': cv2.THRESH_TOZERO},
            {'channels': True, 'name': 'cv2.THRESH_TOZERO_INV', 'type': cv2.THRESH_TOZERO_INV},
            {'channels': False, 'name': 'cv2.THRESH_TRIANGLE', 'type': cv2.THRESH_TRIANGLE},
            {'channels': True, 'name': 'cv2.THRESH_TRUNC', 'type': cv2.THRESH_TRUNC},
        ]
```

```
In [535]: threshs = []
for obj in imgs_and_grays:
    for ths in types_thresh:
        it = obj.copy()
        img = it.get('X')

        kwargs = {
            'src': img,
            'thresh': 50,
            'maxval': 255,
            'type': ths.get('type')
        }

        if not ths.get('channels'):
            len_shape = len(img.shape)
            if len_shape == 3:
                for i in range(len_shape):
```

```

        it = obj.copy()
        img = it.get('X')
        kwargs.update({'src': img[:, :, i]})
        retval, dst = cv2.threshold(**kwargs)
        it.update({'title': it.get('title') + '\n' + 'Channel ' + str(i)
                  'X': dst,
                  'cmap': 'gray'})
        threshs.append(it)
    else:
        retval, dst = cv2.threshold(**kwargs)

        it.update({'title': it.get('title') + '\n' + 'Thresh ' + ths.get('name')
                  'X': dst})
        threshs.append(it)

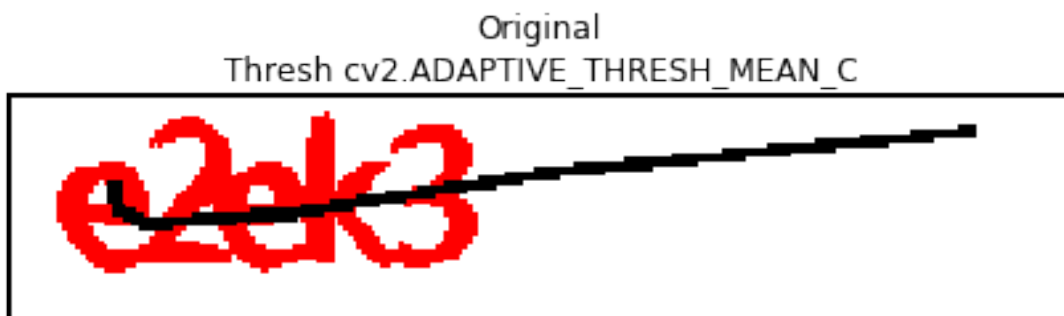
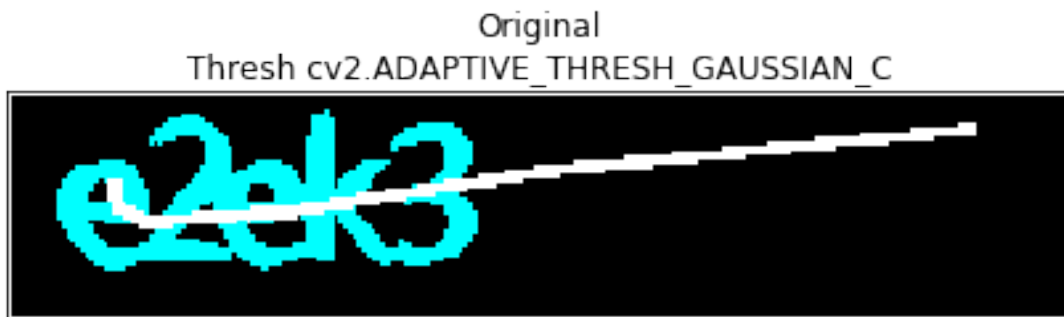
```

```
In [536]: exhibir_imagens(threshs)
```

```

/home/iury/.pyenv/versions/3.6.2/envs/lab/lib/python3.6/site-packages/matplotlib/pyplot.py:524:
  max_open_warning, RuntimeWarning)

```



Original
Thresh cv2.CALIB_CB_ADAPTIVE_THRESH



Original
Thresh cv2.THRESH_BINARY



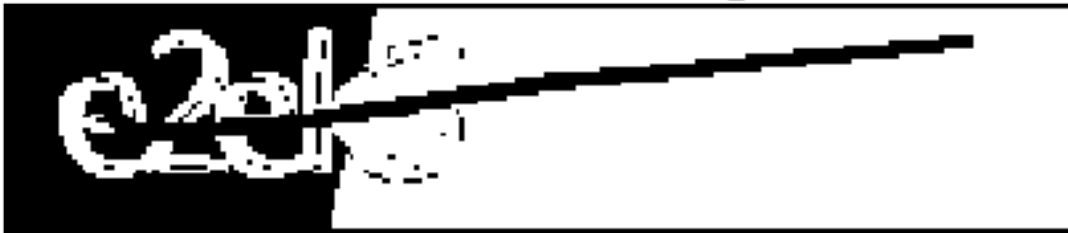
Original
Thresh cv2.THRESH_BINARY_INV



Original
Thresh cv2.THRESH_MASK



Original
Channel 0 Thresh cv2.THRESH_OTSU



Original
Channel 1 Thresh cv2.THRESH_OTSU



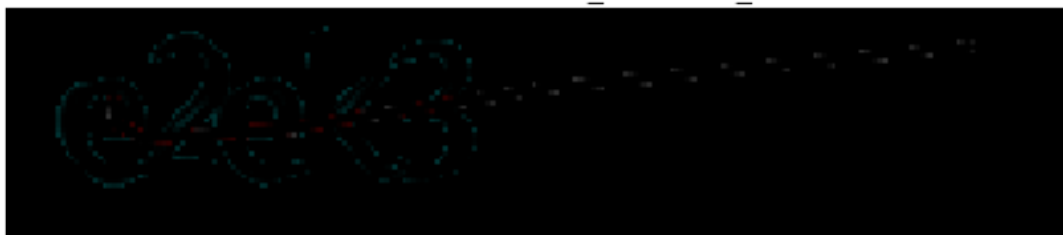
Original
Channel 2 Thresh cv2.THRESH_OTSU



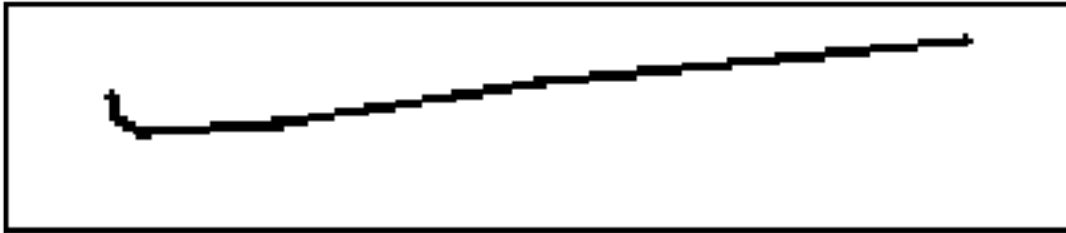
Original
Thresh cv2.THRESH_TOZERO



Original
Thresh cv2.THRESH_TOZERO_INV



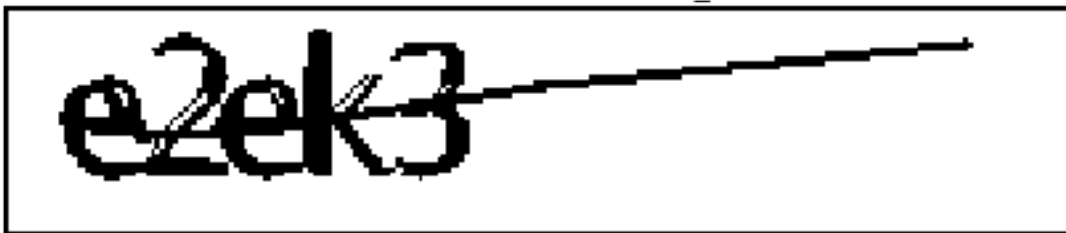
Original
Channel 0 Thresh cv2.THRESH_TRIANGLE



Original
Channel 1 Thresh cv2.THRESH_TRIANGLE



Original
Channel 2 Thresh cv2.THRESH_TRIANGLE



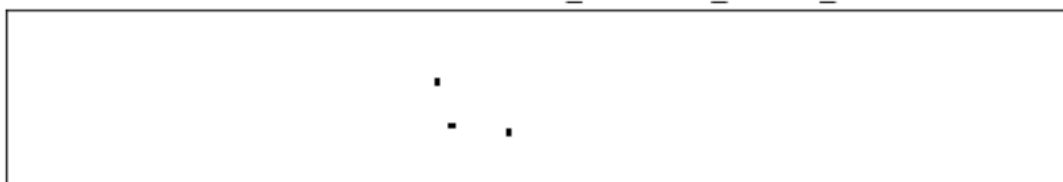
Original
Thresh cv2.THRESH_TRUNC



Original
Thresh cv2.ADAPTIVE_THRESH_GAUSSIAN_C



Original
Thresh cv2.ADAPTIVE_THRESH_MEAN_C



Original
Thresh cv2.CALIB_CB_ADAPTIVE_THRESH



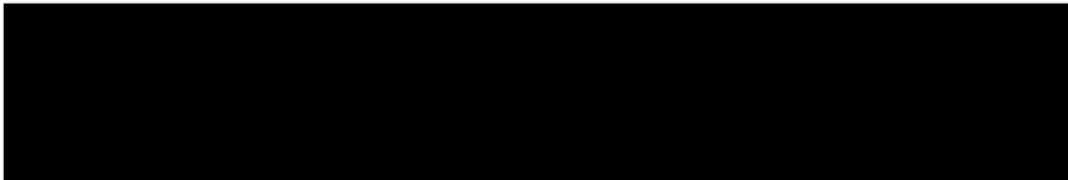
Original
Thresh cv2.THRESH_BINARY



Original
Thresh cv2.THRESH_BINARY_INV



Original
Thresh cv2.THRESH_MASK



Original
Channel 0 Thresh cv2.THRESH_OTSU



Original
Channel 1 Thresh cv2.THRESH_OTSU



Original
Channel 2 Thresh cv2.THRESH_OTSU



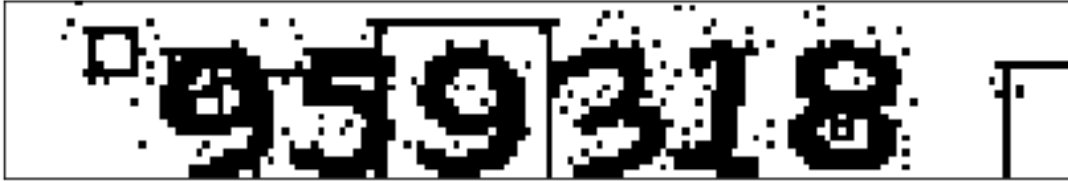
Original
Thresh cv2.THRESH_TOZERO



Original
Thresh cv2.THRESH_TOZERO_INV



Original
Channel 0 Thresh cv2.THRESH_TRIANGLE



Original
Channel 1 Thresh cv2.THRESH_TRIANGLE



Original
Channel 2 Thresh cv2.THRESH_TRIANGLE



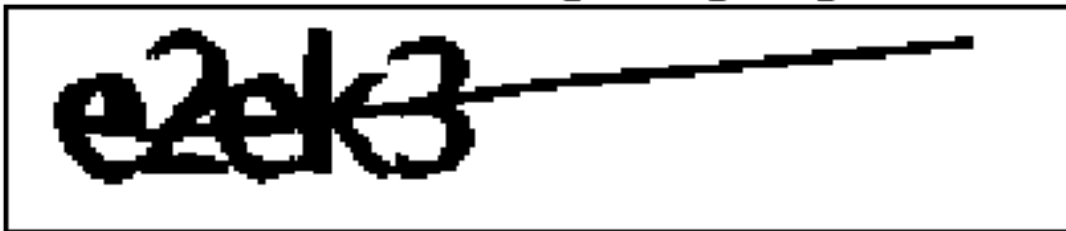
Original
Thresh cv2.THRESH_TRUNC



Original Gray
Thresh cv2.ADAPTIVE_THRESH_GAUSSIAN_C



Original Gray
Thresh cv2.ADAPTIVE_THRESH_MEAN_C



Original Gray
Thresh cv2.CALIB_CB_ADAPTIVE_THRESH



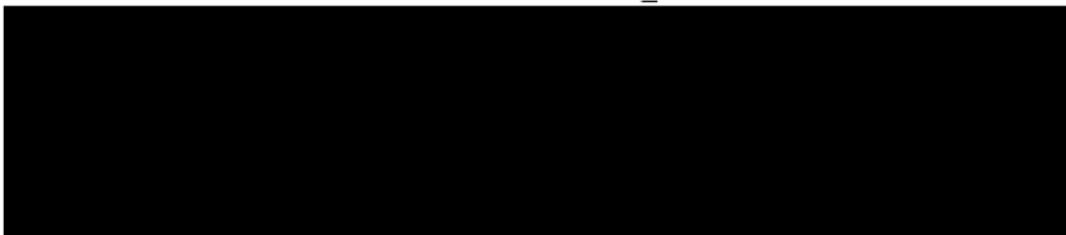
Original Gray
Thresh cv2.THRESH_BINARY



Original Gray
Thresh cv2.THRESH_BINARY_INV



Original Gray
Thresh cv2.THRESH_MASK



Original Gray
Thresh cv2.THRESH_TOZERO



Original Gray
Thresh cv2.THRESH_TOZERO_INV



Original Gray
Thresh cv2.THRESH_TRUNC



Original Gray
Thresh cv2.ADAPTIVE_THRESH_GAUSSIAN_C



Original Gray
Thresh cv2.ADAPTIVE_THRESH_MEAN_C



Original Gray
Thresh cv2.CALIB_CB_ADAPTIVE_THRESH



Original Gray
Thresh cv2.THRESH_BINARY



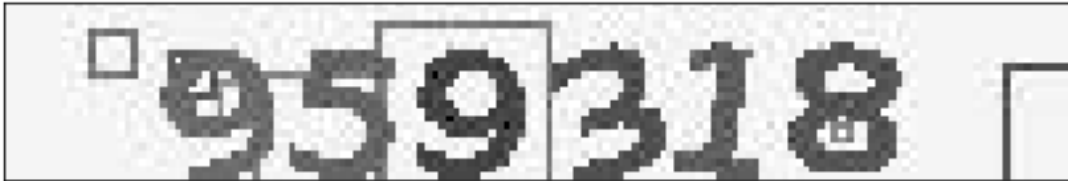
Original Gray
Thresh cv2.THRESH_BINARY_INV



Original Gray
Thresh cv2.THRESH_MASK



Original Gray
Thresh cv2.THRESH_TOZERO



Original Gray
Thresh cv2.THRESH_TOZERO_INV



Original Gray
Thresh cv2.THRESH_TRUNC



2 Segmentação baseada em agrupamento

2.1 K-means

In [537]: kmeans = []

```
for obj in imgs_and_grays:
    it = obj.copy()
    img = it.get('X')

    kwargs = {
        'K': 2,
        'bestLabels': None,
        'criteria': (cv2.TERM_CRITERIA_EPS + cv2.TERM_CRITERIA_MAX_ITER, 10, 0.5),
        'attempts': 10,
        'flags': cv2.KMEANS_RANDOM_CENTERS
    }

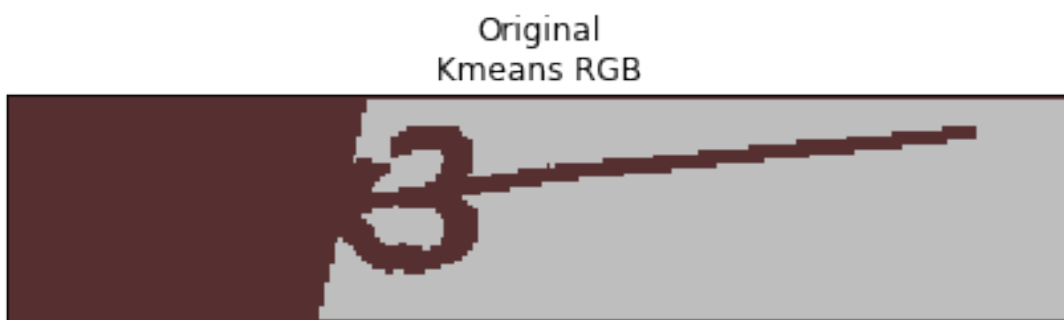
    if len(img.shape) == 3:
        kwargs.update({'data': np.float32(img.reshape((-1, img.shape[2])))})
        it.update({'title': it.get('title') + '\n' + 'Kmeans RGB'})
    else:
        kwargs.update({'data': np.float32(img.reshape((-1, 1)))})
        it.update({'title': it.get('title') + '\n' + 'Kmeans Gray'})

    retval, bestLabels, centers = cv2.kmeans(**kwargs)

    center = np.uint8(centers)
    res = center[bestLabels.flatten()]
    res2 = res.reshape((img.shape))

    it.update({'X': res2})
    kmeans.append(it)
```

In [538]: exhibir_imagens(kmeans)



Original
Kmeans RGB



Original Gray
Kmeans Gray



Original Gray
Kmeans Gray



2.2 mean shift

2.3 agrupamento hierárquico

2.4 crescimento de regiões

3 Segmentação interativa

3.1 graph cut

3.2 grow cut

3.3 grab cut

3.4 snakes

3.5 Método de exibição das imagens

```
In [539]: def exibir_imagens(list_process=[]):  
    for obj in list_process:  
        plt.figure()  
        plt.title(s=obj.get('title', ''))  
        plt.xticks(obj.get('xticks', []))  
        plt.yticks(obj.get('yticks', []))  
        plt.imshow(X=obj.get('X'), cmap=obj.get('cmap', None))  
        plt.tight_layout()
```

```
In [540]: all_images = []
```

```
In [541]: all_images.extend(images)  
all_images.extend(grays)  
all_images.extend(threshs)  
all_images.extend(kmeans)
```

```
In [542]: exibir_imagens(all_images)
```

```
/home/iury/.pyenv/versions/3.6.2/envs/lab/lib/python3.6/site-packages/matplotlib/pyplot.py:524  
max_open_warning, RuntimeWarning)
```



Original



Original Gray



Original Gray



Original

Thresh cv2.ADAPTIVE_THRESH_GAUSSIAN_C



Original
Thresh cv2.ADAPTIVE_THRESH_MEAN_C



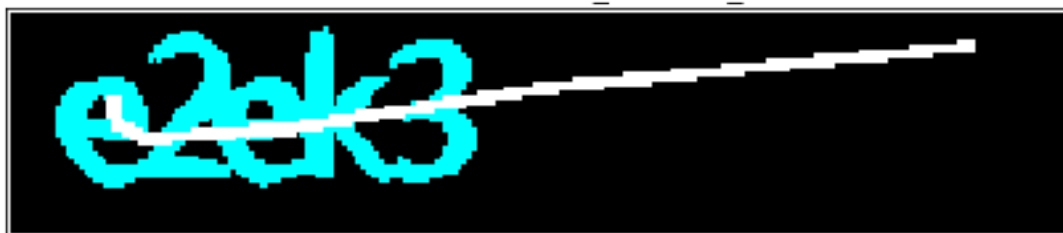
Original
Thresh cv2.CALIB_CB_ADAPTIVE_THRESH



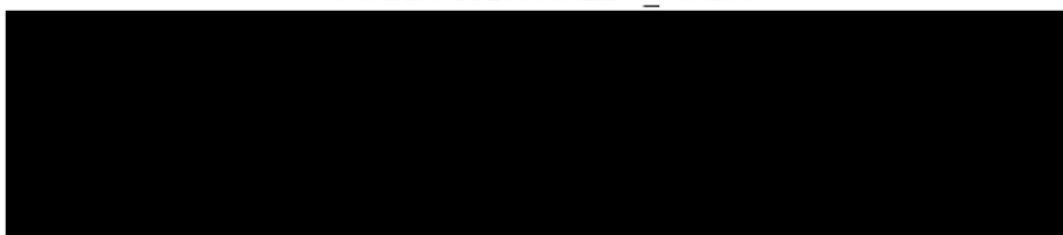
Original
Thresh cv2.THRESH_BINARY



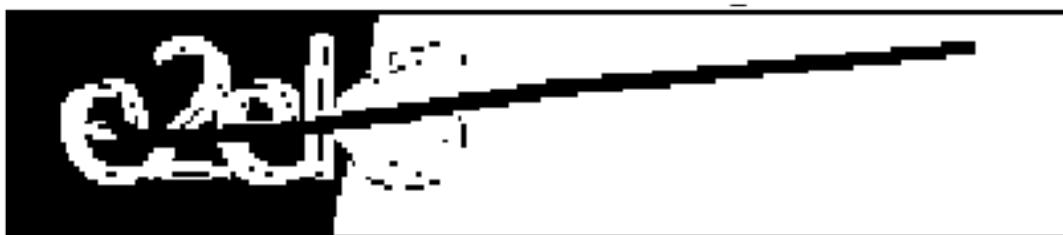
Original
Thresh cv2.THRESH_BINARY_INV



Original
Thresh cv2.THRESH_MASK



Original
Channel 0 Thresh cv2.THRESH_OTSU



Original
Channel 1 Thresh cv2.THRESH_OTSU



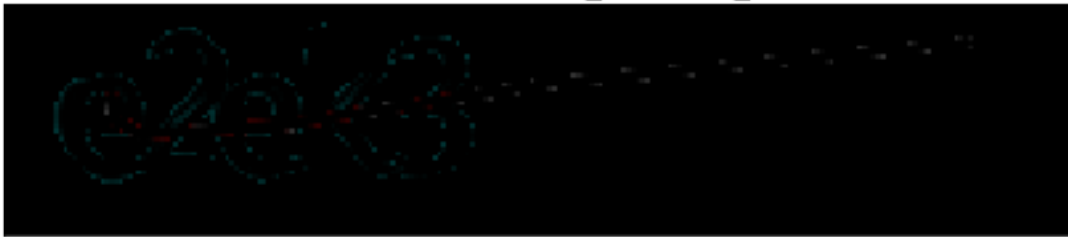
Original
Channel 2 Thresh cv2.THRESH_OTSU



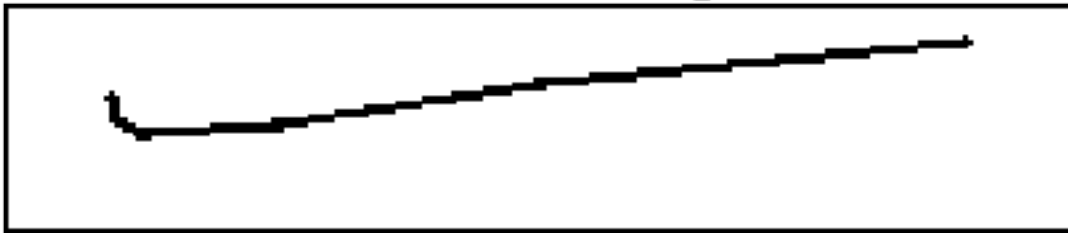
Original
Thresh cv2.THRESH_TOZERO



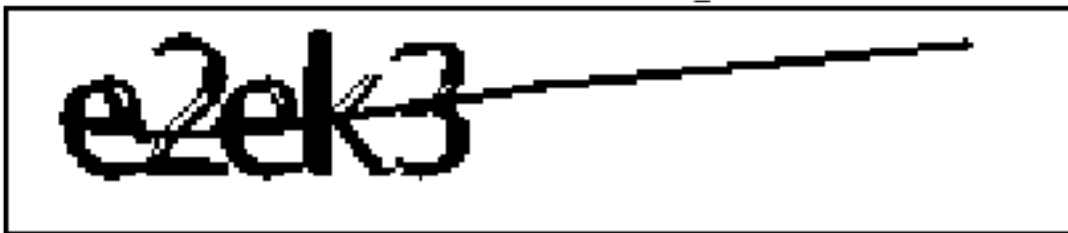
Original
Thresh cv2.THRESH_TOZERO_INV



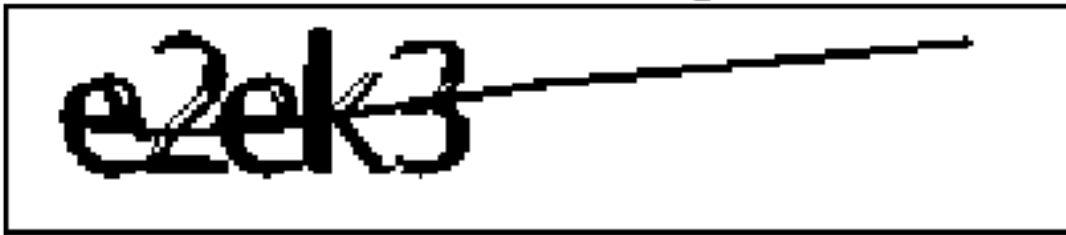
Original
Channel 0 Thresh cv2.THRESH_TRIANGLE



Original
Channel 1 Thresh cv2.THRESH_TRIANGLE



Original
Channel 2 Thresh cv2.THRESH_TRIANGLE



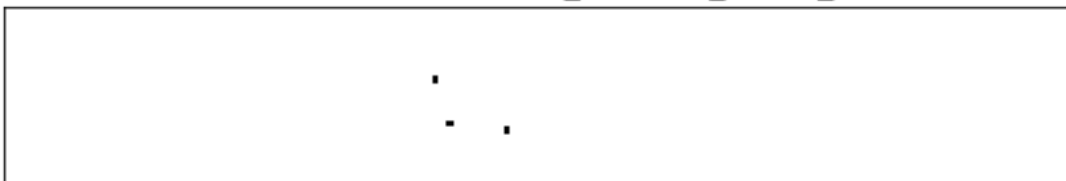
Original
Thresh cv2.THRESH_TRUNC



Original
Thresh cv2.ADAPTIVE_THRESH_GAUSSIAN_C



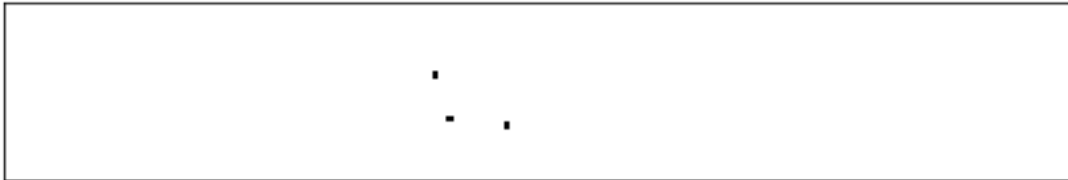
Original
Thresh cv2.ADAPTIVE_THRESH_MEAN_C



Original
Thresh cv2.CALIB_CB_ADAPTIVE_THRESH



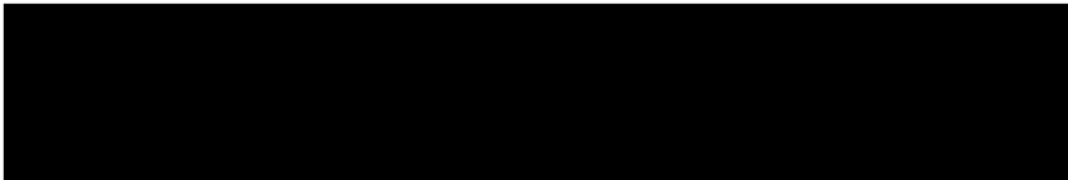
Original
Thresh cv2.THRESH_BINARY



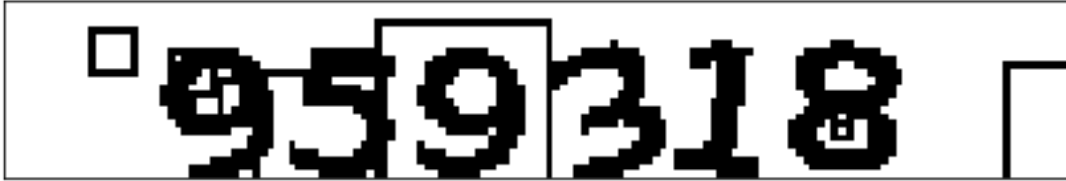
Original
Thresh cv2.THRESH_BINARY_INV



Original
Thresh cv2.THRESH_MASK



Original
Channel 0 Thresh cv2.THRESH_OTSU



Original
Channel 1 Thresh cv2.THRESH_OTSU



Original
Channel 2 Thresh cv2.THRESH_OTSU



Original
Thresh cv2.THRESH_TOZERO



Original
Thresh cv2.THRESH_TOZERO_INV



Original
Channel 0 Thresh cv2.THRESH_TRIANGLE



Original
Channel 1 Thresh cv2.THRESH_TRIANGLE



Original
Channel 2 Thresh cv2.THRESH_TRIANGLE



Original
Thresh cv2.THRESH_TRUNC



Original Gray
Thresh cv2.ADAPTIVE_THRESH_GAUSSIAN_C



Original Gray
Thresh cv2.ADAPTIVE_THRESH_MEAN_C



Original Gray
Thresh cv2.CALIB_CB_ADAPTIVE_THRESH



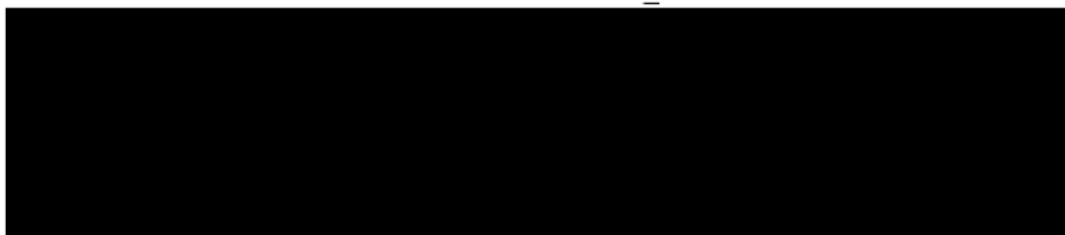
Original Gray
Thresh cv2.THRESH_BINARY



Original Gray
Thresh cv2.THRESH_BINARY_INV



Original Gray
Thresh cv2.THRESH_MASK



Original Gray
Thresh cv2.THRESH_TOZERO



Original Gray
Thresh cv2.THRESH_TOZERO_INV



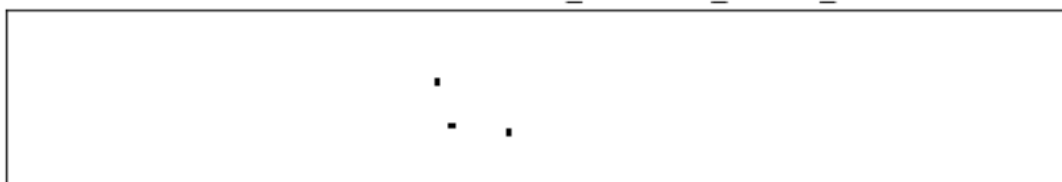
Original Gray
Thresh cv2.THRESH_TRUNC



Original Gray
Thresh cv2.ADAPTIVE_THRESH_GAUSSIAN_C



Original Gray
Thresh cv2.ADAPTIVE_THRESH_MEAN_C



Original Gray
Thresh cv2.CALIB_CB_ADAPTIVE_THRESH



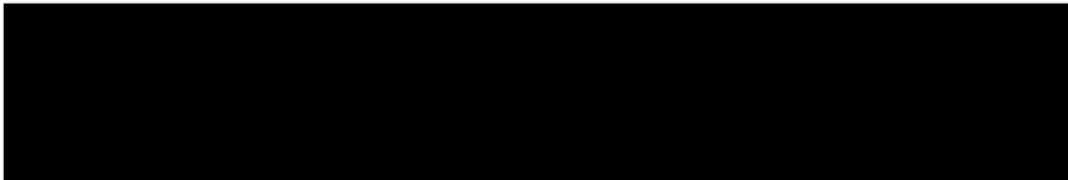
Original Gray
Thresh cv2.THRESH_BINARY



Original Gray
Thresh cv2.THRESH_BINARY_INV



Original Gray
Thresh cv2.THRESH_MASK



Original Gray
Thresh cv2.THRESH_TOZERO



Original Gray
Thresh cv2.THRESH_TOZERO_INV



Original Gray
Thresh cv2.THRESH_TRUNC



Original
Kmeans RGB



Original
Kmeans RGB



Original Gray
Kmeans Gray



Original Gray
Kmeans Gray

