Requirements – (experimental)

Pip install pytesseract

Pip install cv2

For terminal –

Brew install tesseract

OCR Modification can be done – (source stackoverflow)

For detecting the rectangle

from PIL import Image

from tesserocr import PyTessBaseAPI, RIL

image = Image.open('/usr/src/tesseract/testing/phototest.tif')

with PyTessBaseAPI() as api:

api.SetImage(image)

boxes = api.GetComponentImages(RIL.TEXTLINE, True)

print 'Found {} textline image components.'.format(len(boxes))

for i, (im, box, \_, \_) in enumerate(boxes):

# im is a PIL image object

# box is a dict with x, y, w and h keys

api.SetRectangle(box['x'], box['y'], box['w'], box['h'])

ocrResult = api.GetUTF8Text()

conf = api.MeanTextConf()

print (u"Box[{0}]: x={x}, y={y}, w={w}, h={h}, "

"confidence: {1}, text: {2}").format(i, conf, ocrResult, \*\*box)

**Iterator over the classifier choices for a single symbol:**

from tesserocr import PyTessBaseAPI, RIL, iterate\_level

with PyTessBaseAPI() as api:

api.SetImageFile('/usr/src/tesseract/testing/phototest.tif')

api.SetVariable("save\_blob\_choices", "T")

api.SetRectangle(37, 228, 548, 31)

api.Recognize()

ri = api.GetIterator()

level = RIL.SYMBOL

for r in iterate\_level(ri, level):

symbol = r.GetUTF8Text(level) # r == ri

conf = r.Confidence(level)

if symbol:

print u'symbol {}, conf: {}'.format(symbol, conf),

indent = False

ci = r.GetChoiceIterator()

for c in ci:

if indent:

print '\t\t ',

print '\t- ',

choice = c.GetUTF8Text() # c == ci

print u'{} conf: {}'.format(choice, c.Confidence())

indent = True

print '---------------------------------------------'