Kindleの各ページを順番にスキャンして画像ファイルを作成し、OCRにかけてテキストを得る、自動的に。

処理の流れ

前半

Kindleの場合

1. Kindle本のページの左上と右下の座標を取得する。
    - MPP Utility
2. Kindleから、各ページのヘッダー、ページ番号、横見出しなどを含まない矩形画像を連続的に切り出して、PNGファイルとしてフォルダーに保存する。
    - progs\scan_kindle.py
3. フォルダー内の画像ファイルを順番に読み込んで、一つのPDFファイルにする。
    - progs\images2pdf.py

PDFの場合

1. PDFをJPGにする。
    - 4Videosoft Free PDF to JPEG Converter.exe

後半

1. 画像ファイルを読み込んで、必要があれば、各ページのヘッダー、ページ番号、横見出しなどを含まない矩形画像を連続的に切り出して、中のテキストを抽出する。
    - progs\pyocr_tesseract2_main.py
2. フォルダー中のtxtファイルを読んで、WAVファイルに書き出す。(Lifebookで)
    - progs\texts2wavs.py

相互参照リンク

- 自分
    - GitHub\RPA\PyAutoGui\memo.md
- プログラム
    - GitHub\RPA\PyAutoGui\progs (Lifebook)
- Tesseract
    - Home · UB-Mannheim/tesseract Wiki
- Issues

    - kindle 自動スクショ 連続 · Issue #751 · t-magic/keep2git
    - pyocr+tesseract+anaconda+win · Issue #753 · t-magic/keep2git
- GoogleDrive
    - ディープラーニングG検定公式テキスト - Google ドライブ
        - スキャンしてiPadへ - Google ドライブ
    - book1 - Google ドキュメント
        - PDFをGoogle Drive Documentで開いた場合、非常に精度が高い。
- 関連Webページ
    - Windowsアプリとして(こちらは、しなくてよかった)
        - Tesseract OCR をWindowsにインストールする方法 | ガンマソフト株式会社
    - Pythonで(こちらでよかった)
        - Anacondaで日本語OCR環境を作る（tesseract+pyocr） - Qiita
    - PythonでOCRを実行する方法 | ガンマソフト株式会社
        - Python3.7のため、PyOCRは、condaでは入れられなかったので、この参考ページの通り、pipで入れた。
        - 「原稿画像加工（黒っぽい色以外は白=255,255,255にする）」ということもしている。
- 学習済みデータ(LSTM)
    - ベスト
        - tessdata_best/jpn.traineddata at master · tesseract-ocr/tessdata_best
    - 普通
        - tesseract-ocr/tessdata: Trained models with support for legacy and LSTM OCR engine

---

1.
    -

PyAutoGui

1. conda install -c conda-forge pyautogui

```
conda install -c conda-forge pyautogui
conda install -c conda-forge opencv
```

PyCharm

Tesseractのインストール

- Tesseract OCR をWindowsにインストールする方法 | ガンマソフト株式会社
    - これはインストールしなくてよかった。
    - プログラム

        - C:\Program Files\Tesseract-OCR\tesseract.exe
    - テストファイル
        - C:\Repository\GitHub\RPA\PyAutoGui\data\ocr-test.png
    - コマンド
        - > cd C:\Repository\GitHub\RPA\PyAutoGui\data
          >
          > "C:\Program Files\Tesseract-OCR\tesseract.exe" ocr-test.png ocr-test-out -l jpn

以上はWindows アプリの場合。以下はPythonでアクセスする場合

- PythonでOCRを実行する方法 | ガンマソフト株式会社
    - conda install -c brianjmcguirk pyocr
        - NGだった。Python3.6まで
    - pip install pyocr

```
(py37_PyAutoGui) C:\Repository\GitHub\RPA\PyAutoGui\data>pip install
pyocr
Collecting pyocr
  Downloading pyocr-0.7.2.tar.gz (65 kB)
     |████████████████████████████████| 65 kB 299 kB/s
Requirement already satisfied: Pillow in
c:\programdata\anaconda3\envs\py37_pyautogui\lib\site-packages (from
pyocr) (7.1.2)
Building wheels for collected packages: pyocr
  Building wheel for pyocr (setup.py) ... done
  Created wheel for pyocr: filename=pyocr-0.7.2-py3-none-any.whl
size=36503
sha256=034bd77e53a7f398a96417926a63944eee63879aa50b4f9f247b2da97874d11b
  Stored in directory:
c:\users\tateno\appdata\local\pip\cache\wheels\0c\21\8e\552839aab8152847
fb44ffff9e8c84bd10ff345562aff0bd88
Successfully built pyocr
Installing collected packages: pyocr
Successfully installed pyocr-0.7.2

(py37_PyAutoGui) C:\Repository\GitHub\RPA\PyAutoGui\data>
```

tesseractが見えていないので、

アプリとは別にインストールすることにした。(これが正解)

- Anacondaで日本語OCR環境を作る（tesseract+pyocr） - Qiita
- conda install -c conda-forge tesseract

```
(py37_PyAutoGui) C:\Repository\GitHub\RPA\PyAutoGui\data>conda install -c
conda-forge tesseract
Collecting package metadata (current_repodata.json): done
Solving environment: done

## Package Plan ##
```

```
  environment location: C:\ProgramData\Anaconda3\envs\py37_PyAutoGui

  added / updated specs:
    - tesseract


The following packages will be downloaded:

    package                    |            build
    ---------------------------|-----------------
    bzip2-1.0.8                |       hfa6e2cd_2         148 KB  conda-
forge
    leptonica-1.78.0           |       h919f142_2         1.7 MB  conda-
forge
    libarchive-3.3.3           |    h0c0e0cf_1008         1.4 MB  conda-
forge
    libiconv-1.15              |    hfa6e2cd_1006         672 KB  conda-
forge
    libwebp-1.0.2              |       hfa6e2cd_5         356 KB  conda-
forge
    libxml2-2.9.10             |       h5d81f1c_1         3.4 MB  conda-
forge
    openjpeg-2.3.1             |       h57dd2e7_3         225 KB  conda-
forge
    tesseract-4.1.1            |       h328755b_1        15.5 MB  conda-
forge
    ------------------------------------------------------------
                                           Total:        23.4 MB

The following NEW packages will be INSTALLED:

  bzip2              conda-forge/win-64::bzip2-1.0.8-hfa6e2cd_2
  leptonica          conda-forge/win-64::leptonica-1.78.0-h919f142_2
  libarchive         conda-forge/win-64::libarchive-3.3.3-h0c0e0cf_1008
  libiconv           conda-forge/win-64::libiconv-1.15-hfa6e2cd_1006
  libwebp            conda-forge/win-64::libwebp-1.0.2-hfa6e2cd_5
  libxml2            conda-forge/win-64::libxml2-2.9.10-h5d81f1c_1
  lzo                conda-forge/win-64::lzo-2.10-hfa6e2cd_1000
  openjpeg           conda-forge/win-64::openjpeg-2.3.1-h57dd2e7_3
  tesseract          conda-forge/win-64::tesseract-4.1.1-h328755b_1


Proceed ([y]/n)?
```

その結果

```
C:\ProgramData\Anaconda3\envs\py37_PyAutoGui\python.exe
C:/Repository/GitHub/RPA/PyAutoGui/progs/pyocr_tesseract.py
Will use tool 'Tesseract (sh)'
Available languages: eng, osd
Will use lang 'eng'

Process finished with exit code 0
```

データのダウンロード

- tesseract-ocr/tessdata_best: Best (most accurate) trained LSTM models.

```
C:\ProgramData\Anaconda3\envs\py37_PyAutoGui\python.exe
C:/Repository/GitHub/RPA/PyAutoGui/progs/pyocr_tesseract.py
Will use tool 'Tesseract (sh)'
Available languages: eng, jpn, osd
Will use lang 'eng'

Process finished with exit code 0
```

```
C:\ProgramData\Anaconda3\envs\py37_PyAutoGui\python.exe
C:/Repository/GitHub/RPA/PyAutoGui/progs/pyocr_tesseract2.py
て す と テス ト

Process finished with exit code 0
```

```
C:\ProgramData\Anaconda3\envs\py37_PyAutoGui\python.exe
C:/Repository/GitHub/RPA/PyAutoGui/progs/pyocr_tesseract2.py
番号 AB12345678CD

これ は OCR テス ト 用 の テキ スト で す 。

吾 輩 は 猫 で ある 。 名 前 は まだ な い 。

どこ で 生れ た か 頓 (と ん ) と 見 当 が つか ぬ 。 何 で も 薄暗い し めじ めし た所 で ニャ ー ニ ャ
ー ー 泣い て いた 事 だ け は 記憶 し て いる 。 吾 圧 は ここ で 始め て 人 間 と いう も の を 見 た 。 し か も
あと で 聞く と それ は 書生 と いう 人 間 中 で 一 番 獲 悪 (どう あく ) な 種族 で あっ た そう だ 。 こ
の 書生 と いう の は 時 々 我々 を 捕 (つか ま ) えて 者 て 食う と いう 話 で ある 。 し か し その 当時
は 何と いう 考 (か ん が え ) も な か っ た か ら 別 段 恐 し いと も 思わ な か っ た 。 た だ 彼 の 掌 ( て
の ひら ) に 載せ られ て スー と 持ち 上 げ ら れ た 時 何だ か フワ フワ し た 感じ が なあ っ た ば か り で
ある 。 掌 の 上 で 少し 落ち 付い て 書生 の 顔 を 見 た の が いわ ゆる 人 間 と いう も の の 見 始 ( み は
じ め ) で あろ う 。 こ の 時 妙 な も ゃ の だ と 思っ た 感 し が 今 で も 残っ て いる 。 第 一 毛 を 以 て 装飾
され べき は ず の 顔 が つる つる し て まる で 薬 缶 (や か ん ) だ 。 そ の 後 猫 に も 大 分 逢 ( あ ) っ
た が こん な 浅 輪 に は 一 度 も 出 会 (で く ) わし た 事 が な い 。 の みな ら ず 顔 の 真中 が 余り に 突
起 し て いる 。 そう し て その 瘠 の 中 か ら 時 々 お うお うと 畑 (けむ り ) を 吹く 。 どう も 咽 (お )
せ ぽ く て 実に 弱っ た 。 こ れ が 人 間 の 飲む 畑 草 (タバ コ ) と いう も の で ある 事 は 新 (よう や )
く こ の 上 頃 (ご ろ ) 知っ た 。
```

```
Process finished with exit code 0
```

Kindleから来たデータ

- C:\Repository\GitHub\RPA\PyAutoGui\data\book1
  - picture_0001.png - picture_0005.png

Tesseractのドキュメンテーション

- Home · tesseract-ocr/tesseract Wiki
- tesseract/tesseract.1.asc at master · tesseract-ocr/tesseract

パラメーター

- tesseract_layout
  - PythonでOCR

```
0 = Orientation and script detection (OSD) only.
1 = Automatic page segmentation with OSD.
2 = Automatic page segmentation, but no OSD, or OCR
3 = Fully automatic page segmentation, but no OSD. (Default)
4 = Assume a single column of text of variable sizes.
5 = Assume a single uniform block of vertically aligned text.
6 = Assume a single uniform block of text.
7 = Treat the image as a single text line.
8 = Treat the image as a single word.
9 = Treat the image as a single word in a circle.
10 = Treat the image as a single character.
```