(https://github.com/tesseract-ocr/tesseract/wiki/Training-Tesseract-%E2%80%93-Make-Box-Files)

Training Tesseract – Make Box Files

Shreeshrii edited this page on 26 Feb 2018 · 4 revisions

NOTE: The instructions below are for **older** 3.0x versions of Tesseract.
注：以下の手順は、古い3.0xバージョンのTesseract用です。

Make Box Files　ボックスファイルを作る

For the Run Tesseract for Training step, Tesseract needs a 'box' file to go with each training image. The box file is a text file that lists the characters in the training image, in order, one per line, with the coordinates of the bounding box around the image. Tesseract 3.0 has a mode in which it will output a text file of the required format, but if the character set is different to its current training, it will naturally have the text incorrect. So the key process here is to manually edit the file to put the correct characters in it.
Tesseract for Trainingステップでは、Tesseractは各トレーニング画像に対応するための「ボックス」ファイルを必要とします。ボックスファイルは、トレーニング画像内の文字を1行に1つずつ順番に、画像の周囲の境界ボックスの座標とともにリストしたテキストファイルです。 Tesseract 3.0には、必要な形式のテキストファイルを出力するモードがありますが、文字セットが現在のトレーニングと異なる場合、当然テキストが正しくありません。そのため、ここで重要なプロセスは、ファイルを手動で編集して正しい文字を入れることです。

Run Tesseract on each of your training images using this command line:
このコマンドラインを使用して、各トレーニング画像に対してTesseractを実行します。

```
tesseract [lang].[fontname].exp[num].tif [lang].[fontname].exp[num]
batch.nochop makebox
```

e.g. 例えば

```
tesseract eng.timesitalic.exp0.tif eng.timesitalic.exp0 batch.nochop
makebox
```

Now the hard part. You have to edit the file [lang].[fontname].exp[num].box and put the UTF-8 codes for each character in the file at the start of each line, in place of the incorrect character put there by Tesseract. Example: The distribution includes an image eurotext.tif. Running the above command produces a text file that includes the following lines (lines 141-154):
今難しい部分です。ファイル[lang]。[fontname] .exp [num] .boxを編集し、ファイルの各文字のUTF-8コードを各行の先頭に配置します。代わりに、Tesseractによって誤った文字が配置されます。
例：配布には画像eurotext.tifが含まれています。上記のコマンドを実行すると、次の行を含むテキストファイルが生成されます（行141-154）。

```
s 734 494 751 519 0
p 753 486 776 518 0
r 779 494 796 518 0
i 799 494 810 527 0
n 814 494 837 518 0
g 839 485 862 518 0
t 865 492 878 521 0
u 101 453 122 484 0
b 126 453 146 486 0
e 149 452 168 477 0
r 172 453 187 476 0
d 211 451 232 484 0
e 236 451 255 475 0
n 259 452 281 475 0
```

Since Tesseract was run in English mode, it does not correctly recognize the umlaut. This character needs to be corrected using an editor that supports UTF-8. In this case the u needs to be changed to ü.
Tesseractは英語モードで実行されたので、ウムラウトを正しく認識しません。 この文字は、UTF-8をサポートするエディタを使用して修正する必要があります。 この場合、uをüに変更する必要があります。

Recommended editors that support UTF-8: Notepad++, gedit, KWrite, Geany, Vim, Emacs, Atom, TextMate, Sublime Text. Choose one! Linux and Windows both have a character map that can be used for copying characters that cannot be typed.
UTF-8をサポートする推奨エディタ：Notepad ++、gedit、KWrite、Geany、Vim、Emacs、Atom、TextMate、Sublime Text。 選択してください LinuxとWindowsの両方に、入力できない文字をコピーするために使用できる文字マップがあります。

In theory, each line in the box file should represent one of the characters from your training file, but if you have a horizontally broken character, such as the lower double quote „ it will probably have 2 boxes that need to be merged!
理論的には、ボックスファイルの各行はトレーニングファイルの文字の1つを表す必要がありますが、下二重引用符のように横に折れた文字がある場合は、結合する必要がある2つのボックスがあるでしょう。

Example: lines 116-129:
例:116-129行目:

```
D 101 504 131 535 0
e 135 502 154 528 0
r 158 503 173 526 0
, 197 498 206 510 0
, 206 497 214 509 0
s 220 501 236 526 0
c 239 501 258 525 0
h 262 502 284 534 0
n 288 501 310 525 0
e 313 500 332 524 0
l 336 501 347 534 0
l 352 500 363 532 0
e 367 499 386 524 0
” 389 520 407 532 0
```

As you can see, the low double quote character has been expressed as two single commas. The bounding boxes must be merged as follows:
ご覧のとおり、低二重引用符は2つのシングルコンマとして表現されています。 境界ボックスは次のようにマージする必要があります。

- First number (left) take the minimum of the two lines (197)
  最初の数字(左)は2行のうち最小のものを取る(197)

- Second number (bottom) take the minimum of the two lines (497)
  2番目の数字(下)は2行のうち最小のものを取る(497)

- Third number (right) take the maximum of the two lines (214)
  3番目の数字(右)は2行のうち最大のものをとる(214)

- Fourth number (top) take the maximum of the two lines (510)
  4番目の数字(一番上)は、最大2行(510)を取ります。

This gives: これは与える:

```
D 101 504 131 535 0
e 135 502 154 528 0
r 158 503 173 526 0
„ 197 497 214 510 0
s 220 501 236 526 0
c 239 501 258 525 0
h 262 502 284 534 0
n 288 501 310 525 0
e 313 500 332 524 0
l 336 501 347 534 0
l 352 500 363 532 0
e 367 499 386 524 0
” 389 520 407 532 0
```

If you didn't successfully space out the characters on the training image, some may have been joined into a single box. In this case, you can either remake the images with better spacing and start again, or if the pair is common, put both characters at the start of the line, leaving the bounding box to represent them both. (As of 3.00, there is a limit of 24 bytes for the description of a "character". This will allow you between 6 and 24 unicodes to describe the character, depending on where your codes sit in the unicode set. If anyone hits this limit, please file an issue describing your situation.)
トレーニング画像上の文字の間隔をうまく調整できなかった場合は、いくつかが1つのボックスにまとめられている可能性があります。この場合は、イメージをより良い間隔で作成し直してやり直すか、ペアが共通している場合は、両方の文字を表すために境界ボックスを残して両方の文字を行頭に配置します。（3.00以降、 "文字"の記述には24バイトの制限があります。これにより、コードがUnicodeセット内のどこにあるかに応じて、6から24のUnicodeで文字を記述できます。限度額、あなたの状況を説明する問題を提出してください。

Note that the coordinate system used in the box file has (0,0) at the **bottom-left.**
ボックスファイルで使用されている座標系の左下に（0,0）があることに注意してください。

The last number on each line is the page number (0-based) of that character in the multi-page tiff file.
各行の最後の番号は、複数ページのTIFFファイル内のその文字のページ番号（0から始まる）です。

There are several visual tools for editing box file - please check AddOns wiki.
ボックスファイルを編集するための視覚的なツールがいくつかあります - AddOns wikiをチェックしてください。

Bootstrapping a new character set
新しい文字セットのブートストラップ

If you are trying to train a new character set, it is a good idea to put in the effort on a single font to get one good box file, run the rest of the training process, and then use Tesseract in your new language to make the rest of the box files as follows:
もしあなたが新しい文字セットを訓練しようとしているのなら、一つのフォントに努力して一つの良いボックスファイルを作り、残りのトレーニングプロセスを実行し、そしてあなたの新しい言語でTesseractを使うのが良い考えです。その他のボックスファイルは次のとおりです。

```
tesseract [lang].[fontname].exp[num].tif [lang].[fontname].exp[num] -l
yournewlanguage batch.nochop makebox
```

This should make the 2nd box file easier to make, as there is a good chance that Tesseract will recognize most of the text correctly. You can always iterate this sequence adding more fonts to he training set (i.e. to the command line of mftraining and cntraining below) as you make them, but note that there is no incremental training mode that allows you to add new training data to existing sets. This means that each time you run mftraining and cntraining you are making new data files from scratch from the tr files you give on the command line, and these programs cannot take an existing intproto / pffmtable / normproto and add to them directly.
Tesseractがほとんどのテキストを正しく認識する可能性が高いので、これは2nd boxファイルを作りやすくするでしょう。このシーケンスを繰り返して、トレーニングセットにフォントを追加しながら（つまり、以下のmftrainingおよびcntrainingのコマンドラインに）追加できますが、既存のセットに新しいトレーニングデータを追加することができるインクリメンタルトレーニングモードはありません。 。つまり、mftrainingやcntrainingを実行するたびに、コマンドラインで指定したtrファイルから最初から新しいデータファイルを作成することになり、これらのプログラムは既存のintproto / pffmtable / normprotoを取得して直接追加することはできません。

Tif/Box pairs provided! TIF /ボックスのペアが用意されています！

Tif/Box file pairs are available in the Downloads Archive on SourceForge for these languages:
Tif / Boxファイルのペアは、SourceForgeのダウンロードアーカイブにあります。

Dutch English French German German-fraktur Italian Spanish
オランダ語 英語 フランス語 ドイツ語 ドイツ語-fraktur イタリア語 スペイン語

Note the tiff files are G4 compressed to save space, so you will have to have libtiff or uncompress them first).
TIFFファイルはスペースを節約するためにG4圧縮されているので、libtiffを使うか、最初に解凍する必要があります。

Expect to add LCD font library。! LCDフォントライブラリを追加する予定です。