ディープラーニングを用いた歴史的手書き文献の自動翻刻

――コーパス開発の効率化に向けて

宮川　創

Handwritten Text Recognition for Historical Documents through Deep Learning

― Towards More Efficient Corpus Development Process

MIYAGAWA So

This paper discusses the differences and suitable uses of three handwritten text recognition (HTR) programs developed in Europe: Transkribus, eScriptorium/Kraken, and OCR4all. It commences with an overview of deep learning, HTR, and OCR (optical character recognition) before progressing to review the three programs of interest from the perspectives of history, developer, accuracy rate, layout recognition (including writing orientation), user experience, and cost. All three programs use deep-learning machine-learning technologies. They have also all been proven to reach accuracy rates of close to one hundred percent when appropriately trained depending on the quality of the images of handwritten text, training data, and validation data. Second, the user experience is very important; Transkribus has the simplest installation procedure and graphical user interface, while OCR4all and eScriptorium require users to have expert computer skills. Third, in terms of cost, users of Transkribus are required to purchase credits to access the system and use HTR models to recognize a new text, while eScriptorium and OCR4all do not rely on credit purchase. Finally, we conclude this paper with an overview of suitable cases for each program.

キーワード：HTR（手書きテキスト認識）、OCR（光学文字認識）、自動翻刻、
コーパス開発、ディープラーニング（深層学習）

はじめに[1)]

現在、ヨーロッパでは、ディープラーニング（deep learning：深層学習）を用いた、OCR（optical

1）本稿は日本語におけるデジタルヒューマニティーズに関する最大のメールマガジンである『人文情報学月報』（人文情報学研究所発行、編集長永崎研宣）において、著者が受け持っている連載である「欧州・中東デジタル・ヒューマニティーズ動向」に投稿した「手書きテキスト認識・自動翻刻ソフトウェア・

character recognition：光学文字認識)、あるいは、HTR（handwritten text recognition：手書きテキスト認識）をなす複数のプログラムの開発が進展している。その中でも、特に、文字列を打ちこんで操作する CUI（character user interface）ではなく、カーソルを動かしボタンを押すなどして視覚的に操作する GUI（graphical user interface）を持つ OCR/HTR プログラムが増えてきている。

OCR は、紙に印刷された、もしくは書かれた文字をスキャンして得られた文字の画像を、プログラムを使って、読み取り（認識し)、符号化されたデジタルな文字、通常であれば Unicode を出力する技術を指す。OCR にはコンピュータやワードプロセッサでプリントされた文書や、活版印刷で印刷された文書など、文字が均質な（タイプセットされた）文書の文字をデジタル化していくものもあれば、古代や中世の写本など、人が手で書いた文書の文字をデジタル化するものも射程に入る。前者の印刷された文字の場合、欧州の諸言語なら ABBYY FineReader、[2] 日本語なら「読取革命」[3] や「e.Typist」、[4] 様々な言語に対応した Acrobat Pro[5] の OCR 機能など、市販されている OCR ソフトがある。しかしながら、これらの市販の OCR ソフトウェアは、現在流通しているフォントには対応しているが、文献学者や歴史学者が扱う古いフォントや古い字体の文書および手書きには対応していないことが通常である。市販の OCR ソフトのなかでも、最も多数の言語をカバーしている ABBYY は、ラテン語などいくつかの歴史的な言語にも対応しているものの、コプト語や古ジャワ語など歴史的な文献言語の多くには対応していない。

このような観点から、歴史的な文献を扱うデジタル人文学者の場合、まず、その文献をデジタル化する必要があるが、それをすべて手入力で行うには相当な時間と労力がいる。[6] そのため、OCR を用いて、まず機械に認識させたあと、人間が機械のエラーを修正していくというプロ

Transkribus の基本知識と最新動向」（第 121 号【前編】、2021 年 8 月 31 日)、「歴史文書の手書きテクスト認識（HTR）に関して」（第 98 号、2019 年 9 月 30 日)、「人文学のための深層学習・多層人工ニューラルネットワークを用いた光学文字認識(OCR)：kraken を中心に」（第 115 号【後編】、2021 年 2 月 28 日)、「深層学習を用いた kraken による OCR と、kraken を用いた HTR を通してデジタル学術編集版作成を目指す eScripta」（第 116 号、2021 年 3 月 31 日)、「ウェブブラウザ上で使用可能な、歴史的文献資料の自動デジタル翻刻アプリケーション：Transkribus Lite と OCR4all」（第 118 号、2021 年 5 月 31 日)、「ヨーロッパで進む人文学のためのデジタル・ツールへの一部課金モデルの導入：Transkribus と Trismegistos」（第 112 号、2020 年 11 月 30 日）を編集・統合し、さらに大幅に加筆・修正を加えたものである。

2）様々な言語に対応でき、精度も定評がある市販の OCR ソフトである。Win 版では、PDF 編集ソフトと一緒になった ABBYY FineReader PDF、Mac 版では、OCR およびレイアウト編集機能だけの ABBYY FineReader Pro が販売されている。 “ABBYY FineReader Pro,” ABBYY, accessed on May 19, 2021, https://pdf.abbyy.com/.

3）「読取革命 16」ソースネクスト、https://www.sourcenext.com/product/pc/use/pc_use_003021/（最終閲覧日 2021 年 9 月 29 日)。

4）「e.Typist v. 15.0」メディアドライブ、https://mediadrive.jp/products/et/（最終閲覧日 2021 年 9 月 29 日)。

5）“Adobe Acrobat,” Adobe, accessed on May 19, 2021, https://acrobat.adobe.com/us/en/acrobat.html.

6）もちろん、この手入力作業が、写経のような効果や学問上の新しい気づきなどをもたらすことはあると思われる。

セスが最も時間と労力が少なく高品質なデジタル翻刻が行えると思われる。この利点は、手書きテキストの場合も同様である。手書きテキストの場合は、印刷されたテキストよりも、行認識やレイアウト認識、文字の方向認識などが複雑になるため、それらの認識が高い精度で行えるプログラムが求められる。ヨーロッパでは、手書きテキストのためのOCRだけではなく、行・レイアウト・方向など手書きテキストをより効率的に高い精度で認識しデジタル化できるようにする技術のことを特別にHandwritten Text Recognition、略してHTRと呼んでいる。特殊な歴史文書のフォントや手書きにも深層学習によるトレーニングによって対応できるOCRエンジンとしてはOCRopusやCalamari、[7] Tesseract[8]など様々なオープンソースのものが存在する。だが、これらは手書きテキスト認識に特有の複雑なレイアウトへの対応が十分ではない。今回論じるHTRプログラムのなかには、これらの既存のOCRエンジンにレイアウト機能を補足したものも存在する（OCR4all、eScriptorium/Kraken）。

本稿は、現在ヨーロッパで開発中の3つの歴史的文書のためのHTRアプリケーションである、Transkribus、eScriptorium/kraken、OCR4allについて詳述し、最後に比較して今後の展望を述べる。

1. Transkribus[9]

ヨーロッパでは、OCRだけでは手書きのテキストの認識に困難が伴うため、近年、HTRの開発が盛んであり、その代表例がTranskribusである。Transkribusは、READ（Recognition and Enrichment of Archival Documents）プロジェクト[10]のもと、オーストリアのインスブルック大学が中心となり、スペインのバレンシア工科大学、ドイツのグライフスバルト大学、フィンランドのフィンランド国立公文書館など12の研究機関との共同で開発したものである。このソフトウェア上で文献の画像を読み込むテキスト、手稿本や写本など、手書き文献のテキストの行を認識させ、その行ごとに画像の下にあるエディタに翻刻を入力していく（手動の翻刻）。文字体系の文字数にもよるが30–70ページほど手動の翻刻ができたら、データをグラウンドトゥルース、すなわち機械に学習させるためのトレーニングデータとしてHTRエンジンにパターンマッチングのトレーニングをさせ、文献の残りのページを高い精度で機械に認識させる。その結果はTEI XMLで出力できる。

7）現在は専らOCRopusのPython版であるOCRopyが用いられている。“ocropus/ocropy: Python-based tools for document analysis and OCR,” GitHub, accessed September 30, 2021, https://github.com/ocropus/ocropy. Calamariは、OCRopyの構造をTensorFlowによって再構築したものである。“Calamari-OCR/calamari: Line based ATR Engine based on OCRopy,” GitHub, accessed September 30, 2021, https://github.com/Calamari-OCR/calamari.

8）“Tesseract: Open Source OCR Engine（main repository),” GitHub, accessed September 30, 2021, https://github.com/tesseract-ocr/tesseract.

9）“Transkribus,” READ COOP, accessed September 29, 2021, https://read.transkribus.eu/.

10）“Friends & Partners,” READ COOP, accessed September 29, 2021, https://read.transkribus.eu/network/.

Transkribus は GUI を備えており、コンピュータに詳しくない者でも GUI を通して容易に用いることができる。Transkribus には Windows 版、Mac 版、Linux 版がある。これは、基本的に Linux でしか動かない OCRopy や、基本は Linux 上で動くが、Docker を介すれば Mac でも動く OCRocis（OCRopy のプロセスを簡略化・汎用化したもの）と比べて、より広いローカル環境で用いることができることを意味する。また、これら OCRopy などの OCR ソフトウェアとは異なり、Transkribus はバイナリ化（白黒化）や画像の補正などを ScanTailor などの別のソフトウェアで行う必要はない。Transkribus では、全ての画像やドキュメントは Transkribus のサーバにアップロードされるため、ローカルで動かすことが基本の OCRopy などと比べてセキュリティの面で多少の課題がある。アップロードされたドキュメントは公開か非公開かを選べる。

Transkribus で画像を読み込んだ後は、まず、画像のうちテキストがある場所のセグメンテーションをする必要がある。セグメンテーションでは、リージョンとベースラインおよびポリゴンの認識が自動で行われる。ベースラインとは、線で表される、行を区切るためのラインであり、行の文字の基部を示す。リージョンとは、長方形エリアで表される、複数行のカラムや文章のかたまりである。ポリゴンは、ベースライン上にある文字が書かれてある場所をポリゴン状に細かく指定したものである。Transkribus は、左から右の横書き、右から左の横書きの行の認識に対応している。縦書きも、元の画像を 90 度回転させて擬似的に横書きにすれば、対応することができる。リージョン、ベースライン、ポリゴンの認識は、自動で行われる。なお、これらのレイアウト認識は、すでに Transkribus に搭載されているモデルしか使用することができない。図 1 のようにベースラインの下に線がよく伸びるような複雑な行セグメンテーションや、90 度回転させて書かれた注釈なども対応できるが、線が下の行を突き抜けたり、印章などのようにベースラインが円を描いていたりした場合、レイアウト認識がうまくいかないことがある。レイアウト認識もユーザが自由に機械学習させることができるようにすれば、これ

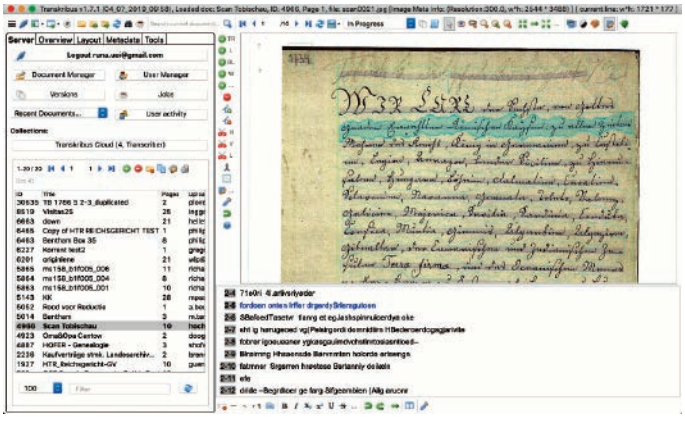

図 1　Transkribus の翻刻編集画面。

らの問題はある程度は解決できるであろう。また、特にグレースケールの文献で、下地と文字の濃淡の差が明瞭ではない文書やインクの裏写りが顕著な文書は、レイアウト認識が失敗することが多い。これは、半自動で文書の画像をスキャンに適した白黒のものにし、文字以外のノイズもある程度除去できるScanTailorというソフトを用いて文書の画像を前処理することで、ある程度は対処できる。

図1のように、青で表示されているポリゴンの行に書かれている文字の翻刻を右下のプレーンに入力することが可能である。もちろん、このように手動で入力することもできるが、Transkribusには、自動で文字を認識する機能も備わっている。ただし、その場合は、複数のページを手動で入力することが必要である。Transkribus Wikiでは100ページをまず手動で翻刻することが推奨されている。[11] ただし、そのページ数は、文字の種類の数、行数、行内の文字数にも大きく左右される。ただし、精度が落ちる可能性は高いものの、30–70ページ程度でも可能である。この手動で入力されたデータをグラウンドトゥルースとして用いて、モデルのトレーニングが行われる。この時に、既存のモデルをベースに新しいモデルをトレーニングさせることも可能である。もし、対象の文書に近い言語や書体のモデルがあれば、それをベースモデルにして、新しい翻刻データでトレーニングし、新しいモデルを作成することが可能である。ただし、コプト文字のように、Transkribusのライブラリに既存の公開モデルがない文字の場合、一からトレーニングする必要がある。

既存の公開モデルがその文字・言語でない場合、Transkribusを新たな文字で用いるにはグラウンドトゥルースを非常に多く用意しなければならない。推奨される100ページ分のグラウンドトゥルースを一から作成してHTRを動かすのは、効率的ではないように思われる。ただし、ある書き手が何冊にもなるテキストを手書きした場合、Transkribusは、大量の翻刻を自動で作成するための非常に有効なツールとなる。というのは、推奨される100ページ分のグラウンドトゥルースの作成という、骨の折れる作業はあるものの、一旦トレーニングすると、同一の書き手の複数の手書きテキスト（例えば、写本や手紙）の翻刻を自動で生成できるからである。

作成した翻刻は、Text Encoding Initiativeが制定したXML形式であるTEI XMLやPDFで出力することができる。特にTEI XMLは、デジタルヒューマニティーズにおいてテキストのマークアップ形式のデファクトスタンダードになっている今、大変重要であると思われる。

Transkribusは、2020年10月19日から一部課金化された。新規ユーザには500クレジットが付与されるが、無料分を使い切ると課金しないと使用できなくなる。[12] この日以降、Transkribusを起動させるとポップアップウィンドウが表示される。Transkribusの目玉機能

11）“[W]e recommend starting with around 20,000 words（100 pages）of training data.”（“Handwritten Text Recognition Workflow,” Transkribus Wiki, accessed September 29, 2021, https://transkribus.eu/wiki/index.php/Handwritten_Text_Recognition_Workflow）.

12）“Transkribus Credits,” READ COOP SCE, accessed November 19, 2020, https://readcoop.eu/transkribus/credits/.

である HTR を使わなければ、ほぼ Transkribus で有用なことができないため、この一部課金システムを利用するしかなくなる。この一部課金化は、Transkribus クレジットを購入し、クレジット分だけページを機械認識させるという仕組みである。まず、支払うシステムとしては、HTR するページの分だけ買うオンデマンド、そして、毎年定期的に支払いその分安くなるサブスクリプションの2つがある。また、使う HTR エンジンに応じて値段が異なる。オープンソースの PyLaia Handwritten エンジンを使えば 1 クレジット = 1 ページとなるが、ロストック大学が開発している HTR +エンジンを使えば、1 クレジットは 1 ページ以下と、1 ページあたりの値段が高くなる。

Transkribus 自体は、実はオープンソースで、はじめは GitHub で公開されており、後に GitLab に移った。ライセンスは、Linux と同じように GNU General Public License（GPL）ライセンスを使用しているため、Transkribus 自体は常にオープンソースとして維持せざるを得ない。しかし、今回は、Transkribus が作った PyLaia Handwritten や CITlab HTR+ といったディープラーニングモデルの使用に課金しているという仕組みである。

もちろん、開発を維持していくには費用がかかる。しかし、このような一部課金化は、発展途上国の個人や学生などにしてみれば高額であることが多く、このような一部課金化は非常に深刻であり、学問の発展に寄与するのか微妙である。Transkribus は、博士論文のプロジェクトで当ツールを用いる人に限って、無料での使用を許可しているようだが、プロジェクトの計画などを提出して、開発チームに認められる必要があり、許可なしに手軽に無料で使用することはできない。

必須のツールであったものが、このように一部課金化されたのは残念だとしか言いようがない。特に、学生や在野研究者などにとってはかなり厳しい措置なのではないだろうか。もちろん、サービスの発展や維持にコストがかかるのは事実で、すべてをボランティアに維持してもらってもサービスの低下が考えられる。Linux や Open Office などボランティアの無償の労働で支えられている高品質のソフトウェアのように維持されて欲しいが、そのようなボランティアコミュニティがあまり育っていない。人文学のうちのある一分野のためのデジタル・ツールは、金銭をユーザから徴収して維持していく以外方法はないのであろうか。一つのアイデアとしてはより多くのボランティアを誘致することであり、ツールの重要性をその人文学の分野の専門家以外にも知らしめていかなければならない。特にエンジニアの協力は不可欠であり、人文学の魅力を IT の専門家に伝えていく必要があると思われる。

課金化に加えて、Transkribus Lite[13] というブラウザ上で動く Transkribus が開発されており、近頃はそれが十分に使えるレベルになっている。図 2 は、後期中世日本語で書かれたローマ字キリシタン資料である『コンテムツスムンヂ』の、ヘルツォーク・アウグスト図書館（ヴォル

13）“Transkribus Lite,” Transkribus, accessed August 16, 2021, https://transkribus.eu/lite/de.

フェンビュッテル）に所蔵されている刊本[14] を Transkribus Lite で表示させた画面である。筆者が、日本語文献学者でボーフム大学日本語・日本文学科教授の Sven Osterkamp 氏と共同で Transkribus をトレーニングさせ、作成したモデルで OCR 処理を行った。2021 年 8 月 16 日時点の機能としては、ローカル版からクラウド上に保存したファイルの文字認識のエラーの修正が可能である。HTR を独自にトレーニングしたり、新しく文書を HTR したりするのは、ローカル版でしかできないようである。しかし、今後、ローカル版でできることが、ブラウザ版でも順次できるようになっていくようである。

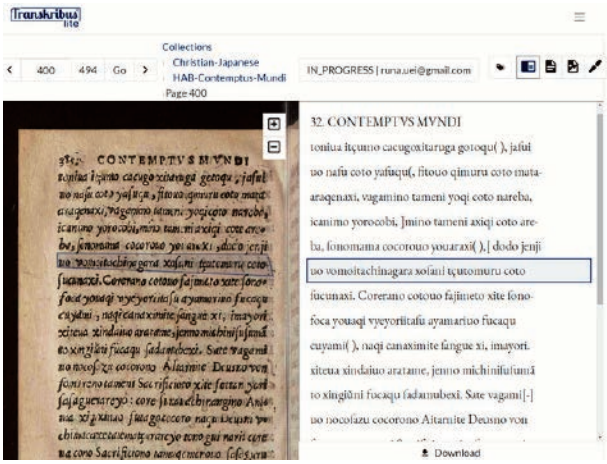

図 2　ローマ字版日本語『コンテムツスムンヂ』をローカル版 Transkribus によって自動で翻刻したものを Transkribus Lite で表示させたもの

2. eScriptorium/Kraken

フランスでは eScriptorium[15] という、Transkribus の代替となるような HTR ソフトが開発

14）“Contemptus mundi jenbu: Core Yovoitoi, Iesu Christono gocŏxeqiuo manabi tatematçuru michiuo voxiyuru qiŏ/von Kempen, Thomas *1379–1471*.– Online-Ausgabe.– ［Amakusa］: [Druckerei der jesuitischen Mission in Japan], Toqini goxuxxeno nenqi. 1596,” accessed September 29, 2021, HAB-Herzog August Bibliothek, http://diglib.hab.de/drucke/57-13-eth/start.htm.

15）“eScriptorium-scripta,” GitLab Inria, accessed September 29, 2021, https://gitlab.inria.fr/scripta/scriptorium. Hypertheses のブログによれば、昨年 10 月に一部課金モデルになった Transkribus から eScritproium（Kraken などを用いた eScripta の HTR プログラム）に切り替えるプロジェクトが出てきたとのことである。Peter A. Stokes, “Moving from Transkribus to eScriptorium,” Hypotheses, accessed March 19, 2021, https://escripta.hypotheses.org/449. Transkribus は現在 HTR として DH で最もよく用いられているプログラムである。Peter Anthony Stokes, Daniel Stökl Ben Ezra, Benjamin Kiessling, and Robin Tissot, “EScripta: A New Digital Platform for the Study of Historical Texts and Writing,” In *DH2019 Book of Abstracts*, 2019, accessed March 19, 2021, https://dev.clariah.nl/files/dh2019/boa/0322.html.

されている。このソフトの HTR エンジンとなっているものが Kraken[16] である。Kraken はライプチヒ大学のコンピュータ科学者である Benjamin Kiessling 氏が、アラビア文字のために OCRopus を改良したのが始まりである。OCRopus は西洋諸言語のように左から右に書く文字順にしか対応できていなかったが、Kraken は、右から左、現在は上から下への書き順に対応している。さらに、HTR として用いられる技術的な水準に到達している高性能な行分割機能（line segmentation）を実装している。Kraken は様々な言語・文字で精度の高い OCR 結果を誇っている。それは、Kiessling 氏の DH2019 ユトレヒト大会での予稿[17] で見ることができる。そこでは、活版印刷物では、アラビア語の文字認識の精度が平均値 99.5%・最大 99.6%（標準偏差 0.05）、ペルシア語で平均値 98.3%・最大 98.7%（標準偏差 0.33）、古典シリア語で平均値 98.7%・最大 99.2%（標準偏差 0.38）、歴史的アクセント表記のギリシア語で平均値 99.2%・最大 99.6%（標準偏差 0.26）、ラテン語で平均値 98.8%・最大 99.3%（標準偏差 0.09）、ラテン語インキュナブラ[18] で平均値 99.0%・最大 99.2%（標準偏差 0.11）、フラクトゥーア[19] で平均値 99.0%・最大 99.3%（標準偏差 0.31）、キリル文字文献で平均値 99.3%・最大 99.6%（標準偏差 0.15）であったと報告されている。また、手書き写本では、ヘブライ語で文字認識の精度の平均値が 96.9%、中世ラテン語で平均値 98.2%という非常に高い文字認識の精度を記録したことが発表されている。

もちろんこれは、その言語の出版物の特定のフォントを機械学習させた結果であり、トレーニングなしでやった結果ではない。アラビア語では OpenITI[20] や KITAB[21] プロジェクトで作成された様々なフォントのためのモデルがあり、[22] もしフォントが同じものがあれば、それらを使えるが、そうでなければ、新しく Kraken をトレーニングさせる必要がある。Kraken は Linux で動かすことが推奨されていたため、Linux ディストロの 1 つである Debian 上で Kraken を動かした。しかし、今回、Anaconda を用いて Mac 上でも動かす方法があることがわかった。今回筆者が使ったコンピュータは M1 チップ（Apple Silicon）搭載の MacBook Pro（late 2020 モデル）である。まず、過去に Homebrew を使ってインストールした wget

16）Kraken, accessed September 29, 2021, http://kraken.re/.

17）Benjamin Kiessling, “Kraken-an Universal Text Recognizer for the Humanities,” DH2019, 2019, accessed September 29, 2021, https://dev.clariah.nl/files/dh2019/boa/0673.html.

18）ドイツで 1455 年に Johannes Gutenberg がグーテンベルク聖書を出版した後、15 世紀のうちに出版された西洋での初期活版印刷物を指す。

19）中世ヨーロッパのカリグラフィーの書体を基にした活字体の 1 つ。特にドイツで第二次世界大戦頃まで使われた。ドイツ文字や亀の子文字、ひげ文字などとも呼ばれる。

20）The Open Islamicate Texts Initiative の略であり、アーガー・ハーン大学、ウィーン大学、メリーランド大学カレッジパーク校が参加している、前近代イスラーム文献のコーパス開発を中心とする DH プロジェクトであり、アラビア文字の OCR の精度の向上もプロジェクトの目標の一つである。“About,” Open Islamicate Text Initiative（OpenITI）, accessed 19 February, 2021, openiti.org/about.

21）KITAB, accessed February 19, 2021, http://kitab-project.org/.

22）“mittagessen/kraken-models,” GitHub, accessed February, 19, 2021, https://github.com/mittagessen/kraken-models.

をターミナル上で用いて、Kraken をインストールした。起動時は Anaconda を使用して、ターミナル上で「conda activate kraken」のコマンドを用いて、Kraken を起動させた。

Mac 上で様々な文献でグラウンドトゥルース入力画面を生成して、OCRopus よりも強化されたとされる Kraken の行認識機能を試した。ここで、OCR モデルを作成する最初の一歩は、まず、グラウンドトゥルースを作ることであるが、OCRopus や Kraken では、html 形式の専門のグラウンドトゥルース入力ページが作成される。そこでは、画像内のテキストのそれぞれの行ごとにボックスが用意され、そのボックスに翻刻を Unicode で書いていくことが求められる。まず、アラビア文字のタイプセットで印刷された書籍のページの画像では、行は正しく認識されていた（図 3）。

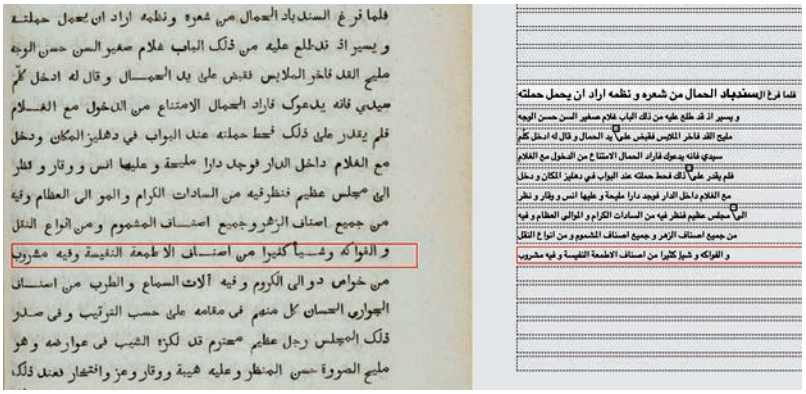

図 3 『千夜一夜物語』の書籍のページの ground truth の作成画面。行は正しく認識されている。画像および翻刻は、国立民族学博物館の“Arabian Nights”Database Search より。[23)]

次に、コプト語の手書き写本である Papyrus Bodmer 6 の 2 ページ見開きでも行認識を行ってみたが、一行目が 2 ページに渡って誤って認識されているが、それ以外は、ほぼ完璧な行認識で、ページの区別も認識されていた。

そのほか、縦書きの文字であるモンゴル文字文献も試してみた。縦書きで、行が左から右に流れる場合、特別なコマンド[24)] を使って、グラウンドトゥルース入力画面を作成しなければならない。結果は芳しくなく、正しい行認識ができていなかった（図 4）。

23）“Vol. 03, Page 006,” “Arabian Nights” Database Search, accessed February 19, 2021, http://www.dhii.jp/ANs/ans.php?m=show&n=300621&key=sndbAd#.

24）ketos transcribe-d vertical-lr-o output.html［画像ファイル名］

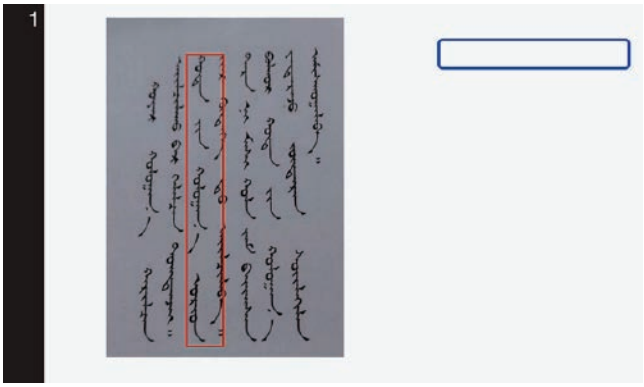

図 4　モンゴル文字のテキストが入った画像[25] を Kraken で認識した

今回、手書き写本のレイアウト認識が十分果たせなかったため、手書き写本ではなく活版印刷されたコプト語印刷物を、Kraken を用いて文字認識させた。文献は活版印刷で印刷された 20 世紀前半のものであり、行間隔は十分であった。ただ元の画像がグレースケールで解像度の粗いものしかないとのことだったので、そのあまり OCR に向かない画像を用いた。1 ページ分の文字量が少なかったので、30 ページ分を切り出して、Kraken 内蔵のプログラムを用いて、白黒化及び行認識及びグラウンドトゥルース入力ファイルの生成を行った。一部だけ 2 行あるところが、1 行として認識されたこと以外は、問題はなかった。コプト文字は左から右に書かれ、行は上から下に流れるため、デフォルトのコマンドで処理できる。その後、ブラウザでグラウンドトゥルース入力用の HTML ファイルを開いて、グラウンドトゥルースを入力、つまり、写真のテキストの翻刻を行った。Kraken では、1 行ごとに入力ボックスに入力するのだが、対応する行が、左に並べられた文献の画像上に赤枠で示される。

Kraken は多層人工ニューラルネットワークを用いている。より正確に言えば、その中でも CLSTM neural network library を用いている。多層人工ニューラルネットワークモデルによる深層学習は、Transkribus の OCR エンジンである PyLaia もしくは HTR＋（PyTorch を基盤とする）でも、OCR4all のエンジンである Calamari（TensorFlow を基盤とする）でも用いられている。深層学習によるトレーニングを経て 8 分ほどで、最適なモデルファイルを Kraken は出力した。その最適なモデルファイルのグラウンドトゥルースに対する精度は 97.6％であった。そして、同じフォントを用いている低画質の資料の全てのページの画像をそのモデルで文字認識させ、結果をテキストファイルに出力した。Kraken はテキストファイルだけでなく、デジタルヒューマニティーズでテキストマークアップに使われる世界標準形式である TEI XML、OCR 専用のファイル形式である hOCR、市販の OCR ソフトである ABBYY

25）戴慈良（改写）、陈宗耀（绘）、包金山（译）『城里老鼠和乡下老鼠』社会主义核心价值观幼儿绘本・5 ～ 6 岁（内蒙古少年儿童出版社）より。モンゴル語学者の外賀葵氏（京都大学）の協力・情報提供を得た。

FineReader で使われる abbyyXML フォーマットでの出力にも対応している。

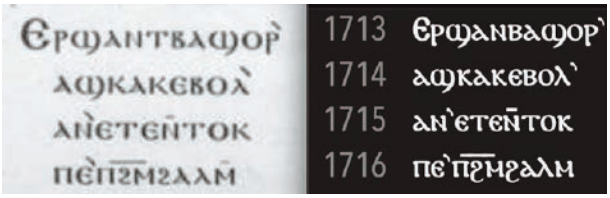

図 5　Kraken に認識させた出版物の画像[26]（左）と Kraken の文字認識の結果（右）。

Kraken の文字認識の欠点を補うツールが、PSL 研究大学を中心となっている eScripta プロジェクトが開発している eScriptorium である。PSL 研究大学とはパリのエコール・ノルマル・シュペリウールなどのグランゼコール（高等専門大学校）などが 2010 年に連合して設立した新しい大学である。eScriptorium は、文献の写真の HTR からデジタル学術編集版のウェブ公開までの一連のプロセスのスタンダードを提示する。ツールは CNN（convolutional neural network）を用いた新しい行認識プログラム、Kraken、Archetype、Pyrrha、TEI Publisher である。CNN のものは、Kraken 内蔵の行認識プログラムよりも優れた行認識能力を持っているようであり、これによって HTR が可能になる。これで行認識した後、学習済の Kraken で文字を認識させ、Archetype[27] で文献情報や言語学的情報をタグ付けした後、Pyrrha[28] で誤認識などのエラーを修正し、最後は TEI Publisher[29] で、TEI XML のデータを変形させてウェブ・デジタル学術編集版としてその文献をウェブ公開する。将来的には、eScriptorium は、行認識、文字認識、エラー修正、タグ付け、ウェブでのデジタル学術編集版の公開という一連のプロセスのモデルを提示することが期待されている。

3. OCR4all[30]

ヴュルツブルク大学の OCR4all プロジェクトは、TensorFlow を OCRopus のモデルに適用させた Calamari を使用し、しかもそれをウェブで誰でも用いることができるようにすることを目標にしている。Uwe Springmann 氏らが開発した PoCoTo[31] と呼ばれるポストコレクショ

26）この写真では、Émile Chassinat, *Le Quatrième Livre des Entretiens et Épîtres de Shenouti*（le Caire: Imprimerie de l'institut français d'archéologie orientale, 1911）の p. 38, l. 35–45 である。
27）Archetype, accessed March 19, 2021, https://archetype.ink/.
28）Thibault Clérice, Julien Pilla, and Jean-Baptiste-Camps, "hipster-philology/pyrrha: 2.1.0," Zenodo, accessed September 30, 2021, https://zenodo.org/record/3524771#.YVUZIppByHt.
29）TEI Publisher, accessed March 19, 2021, https://teipublisher.com/index.html.
30）OCR4all, accessed on May 19, 2021, http://www.ocr4all.org/.
31）"CIS-LMU Post Correction Tool（PoCoTo）," IMPACT Centre of Competence, accessed September 30, 2021, https://www.digitisation.eu/tools-resources/tools-for-text-digitisation/cis-lmu-post-correction-tool-

ンツールや、Christian Reul 氏らが開発したブラウザ上で使用可能であるプリプロセッシングツール LAREX[32] が OCR4all に搭載されている。今まで OCRopus は、コマンドラインで、しかも Docker[33] や VirtualBox[34] を使わないかぎり、基本的には Linux 上でしか動かせなかった。しかしながら、それでは歴史文書の文字をコンピュータに読み取らせたい一般ユーザにとっては扱いにくい。この OCR4all の GUI があるウェブベースのアプリケーションだと、誰でも自分の手持ちのコプト語テキストを手軽に OCR にかけることが可能である。OCR4all は最近、パッケージ化やユーザガイド作成などがなされ、徐々にユーザにとって使いやすいものになってきている。OCR4all は、将来は、すべてウェブベースとなり、ブラウザのみで使用可能になるそうだが、現状は、VirtualBox か Docker を通してしか利用できない。VirtualBox を用いた場合、仮想マシン上で OCR4all が内蔵された Ubuntu を起動させたあと、ローカルマシン上でブラウザを用いて、VirtualBox を通して稼働している OCR4all にアクセスし、作業をしていく流れである。この時、ローカルマシン上の使用フォルダをうまく設定しておかないと、OCR4all に認識させたいファイルが読み込めないことになってしまうので、注意が必要である。

筆者らはテストデータとして付いてきた近世のドイツ語の書籍の写真の前処理、ノイズの除去、行・レイアウト・書字方向の認識などのセグメンテーションまで行った。前処理では、カラーで撮られた写真を白黒にバイナライズしたり、歪みを補正したりを自動で行い、ノイズ除去では、ごみやほこりやインクのこぼれなどのノイズを自動で除去した。そのあとのセグメンテーションは HTR の要であるが、LAREX によってなされている。段落や行を認識するのだが、斜めや縦方向に行が書かれている場合も認識可能である。セグメンテーションが終われば、文字が書かれているブロックの範囲、行のベースライン、挿絵の範囲などが様々な色で表示される（図 6）。誤っている箇所があれば、それぞれの範囲の枠線を構成する小さな点をドラッグすることで、範囲を変更できる。

---

pocoto/.

32）Christian Reul, Uwe Springmann, and Frank Puppe, “LAREX: A semi-automatic open-source Tool for Layout Analysis and Region Extraction on Early Printed Books,” in *DATeCH2017: Proceedings of the 2nd International Conference on Digital Access to Textual Cultural Heritage*, 137–42, 2017. https://doi.org/10.1145/3078081.3078097. なお、LAREX および OCR4all の GitHubレポジトリは、“OCR4all/LAREX,” GitHub, accessed September 30, 2021, https://github.com/OCR4all/LAREX.

33）“Docker: Empowering App Development for Developers,” Docker, accessed September 30, 2021, https://www.docker.com/.

34）Oracle VM VirtualBox, accessed September 30, 2021, https://www.virtualbox.org/.

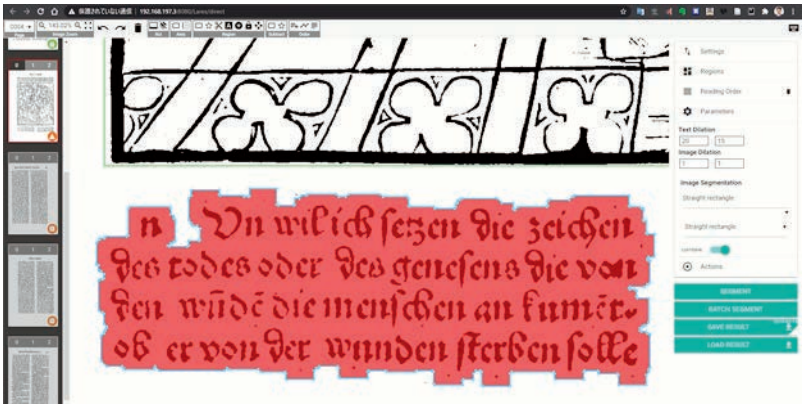

図 6　OCR4all に搭載されている LAREX 上で自動認識されたテキスト範囲を表示している画面。

終わりに

以上、深層学習の基礎を述べた後、それらを用いた HTR アプリケーションである、Transkribus、eScriptorium/Kraken、OCR4all について述べた。使いやすさに関しては Transkribus に軍配が上がるが、文字認識の精度の高い Kraken を用いた eScriptorium や、レイアウト認識が優れた LAREX を用いている OCR4all も今後の開発が期待できる。比較のため、これまでの議論を表 1 にまとめる。

表 1　2021 年 8 月 16 日現在の、Transkribus、eScriptorium、および OCR4all の比較

	Transkribus	eScriptorium	OCR4all
深層学習を用いた文字認識エンジン	PyLaia/CITlab HTR+	Kraken	Calamari
レイアウト認識エンジン	CITlab Advanced	CNN を用いたものを開発中	LAREX
導入のしやすさ	Java 環境の整備が必要だが比較的容易	開発中のため導入には CUI 操作など非常に高い技術が必要	VirtualBox 経由であるため、高い技術が必要
GUI	GUI 完備	GUI は開発中	GUI 完備
料金	500 クレジット使用後はクレジット購入の必要	無料	無料
対応 OS	Windows, Mac, Linux	Linux のみだが、VirtualBox 経由で Win/Mac で起動可	Linux のみだが、VirtualBox 経由で Win/Mac で起動可
ブラウザ上で動くオンラインクラウド版	Transkribus Lite（オンラインクラウド版）	現在のところなし	仮想マシンを通じてブラウザ上で動く（オフライン）

この表を見てわかるとおり、人文学者にとっての使いやすさは、GUI があり、インストール方法も、仮想マシンなどを経由しない Transkribus が群を抜いている。また、ネット経由で、ブラウザ上で動かすことが可能な点も Transkribus が一歩進んでいる。しかしながら、Transkribus は 500 クレジット分を使ってしまえば、後はクレジットを購入しなければならなくなり、他の二者が無料なのに比べて、費用がかかる。ただし、eScriptorium および OCR4all も開発が完了した際には、資金などの状況によっては、課金化がなされるかもしれない。

また、縦書きへの対応は、画像を 90 度回転させるという荒技を行えば Transkribus でも縦書きに対応させることができるものの、Transkribus は、OCR4all や eScriptorium のように、デフォルトで縦書きに対応してはいない。

現在のところ、コンピュータスキルをある程度もつユーザがコストを無料に抑えたい場合は OCR4all、縦書きにも対応させたい場合は eScriptorium、様々な技量のユーザ同士がチームで共同作業をする場合は Transkribus という使い分けができよう。