# 画像ベースとテキストベースのモデルを用いた表の構造解析の性能検証

四條光 [1]　進藤裕之 [1]　渡辺太郎 [1]
[1] 奈良先端科学技術大学院大学大学
{shijo.hikaru.sf3, shindo, taro}@is.naist.jp

## 概要

表は科学論文，web サイト， 新聞など様々な媒体に現れるため，表を解析することは膨大な文書を管理するために重要である．表の構造解析を解くために， 深層学習ベースの画像エンコーダとテキストデコーダから構成される画像ベースのモデルが考案され，非常に高い精度を達成している．一方，エンコーダにマルチモーダルのモデルを使用する研究が登場している．こうした背景から実際にエンコーダには画像ベースのモデルとマルチモーダルのモデルのどちらが優れているか比較することは重要である．本研究では， 表の構造解析のエンコーダ・デコーダモデルを構築し，エンコーダに画像ベース，テキストベース, マルチモーダルの 3 つの異なるモデルを使用し，表の構造解析のスコアを比較することで，どのモデルが優れているのかの比較を行なった．実験の結果，画像ベースのアプローチが良いと示唆された．

## 1　はじめに

表の構造解析とは， 表画像から表の構造的要素 (行， 列， ヘッダー) を抽出し， 対応する HTML や LaTeX に変換するタスクのことである．本タスクにより，世の中に存在する多くの表を機械的に処理しまとめることで，文書管理や情報抽出 [1] などの多くの技術の応用につながる．表の構造解析の初期の研究 [2, 3] ではルールベースにより表の解析を行っていたが 近年では深層学習の発展とともに様々な解析モデルが発展してきた．数ある手法の中でも特に主流になっているのが image-to-text のモデルである [4, 5, 6, 7]． これらは, 画像エンコーダ・テキストデコーダから構成され, 画像エンコーダで特徴量を抽出し, テキストデコーダにより HTML タグを生成していく. 一方で画像のみではなく，画像とテキストの両方を入力とするマルチモーダルのエンコーダを用いたモデル [8] が登場しているが，実際にどちらの特徴量を扱うモデルが表の構造解析に適しているのかというのは検討の余地がある．図 1 の (a) は画像ベースの Tableformer であり，HTML タグとそのセルの座標を別々に出力するのに対し，(b) はマルチモーダルの TableVLM であり，HTML のタグのみを出力する．

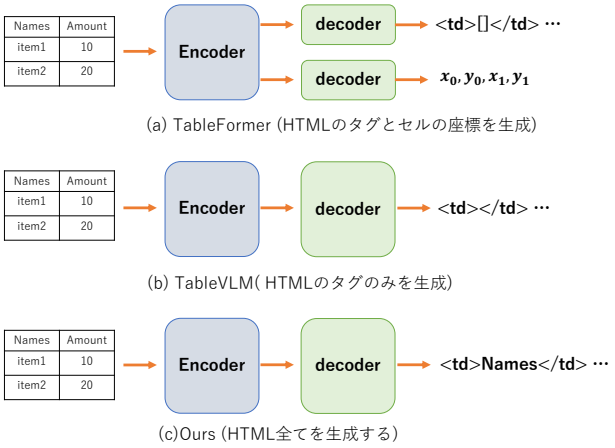



**図 1**　先行研究との比較

本研究では (c) のような HTML 全てを 1 つのデコーダで出力するという枠組みのなかで， Encoder をテキストベース，画像ベース，またはマルチモーダルの 3 つのモデルで表の構造解析のベンチマークである PubTabNet と FinTabNet の精度を比較することで優れているモデルを分析した．その結果として以下の知見が得られた．

- 完全な HTML(タグとセルの内容) を生成する方が, 他の生成の仕方よりも優れている.
- テキストベースやマルチモーダルモデルは，データ効率が良く少量のデータでも精度が得られる
- 大量のデータが得られる状況下では，画像ベースが一番優れている

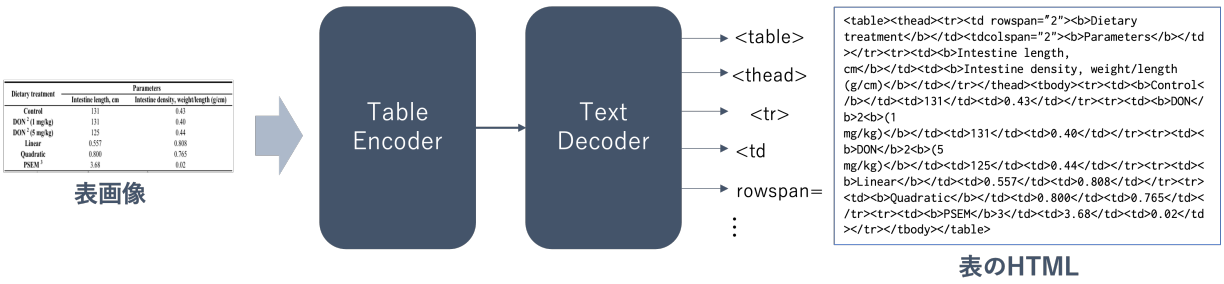



**図 2**　モデルの概要: 表画像から対応する表の HTML を生成するエンコーダ・デコーダモデル．エンコーダで表の潜在表現を得て，デコーダで自己回帰的に HTML のトークンを生成する

## 2　実験方法

画像ベース, テキストベース, または画像とテキストを組み合わせたマルチモーダルのモデルの 3 つを表解析のデータセットで評価し，比較する．

### 2.1　モデル構造

図 2 に示すように，単純なエンコーダ・デコーダモデルを使用する．テキストデコーダには BART-decoder をエンコーダにはテキストベースのモデルとして LayoutLMv3-L と画像ベースのモデルとして Swin Transformer、そしてマルチモーダルモデルとして LayoutLMv3 を使用する．

#### 2.1.1　Swin Transformer エンコーダ

Swin エンコーダ [9] には，表画像 $x \in \mathbb{R}^{(3\times W_0\times H_0)}$ を固定の長方形 $(3, H, W)$ に変形する．この時アスペク比は保持され，長さが足りない場合はパッディングされる．変形された画像はパッチに分割され入力される．入力されたパッチはマージを繰り返して最終的には，潜在表現 $z \in \mathbb{R}^{(N,d)}$ に変換される．($N$ は最終的なパッチの数，$d$ は潜在表現の次元である．)

#### 2.1.2　LayoutLMv3 エンコーダ

LayoutLMv3 エンコーダ [10] には，表画像から OCR で得られたテキストを WordPiece[11] により分割するしたトークン $t_i (0 \le i < L)$ とそのバウンディングボックス $b_i \in (x_0, y_0, x_1, y_1)(0 \le i < L)$，また，表画像 $x \in \mathbb{R}^{(3\times W_0\times H_0)}$ を固定の正方形 $(3, H, W)$ に変形する．以上の 3 つを，マルチモーダルトランスフォーマーに入力することで，レイアウトの関係性を捉えることができ，最終的には，それぞれの単語と画像の潜在表現 $z \in \mathbb{R}^{(L+N,d)}$ に変換される．($N$ は画像パッチの数である．)

#### 2.1.3　LayoutLM3-L エンコーダ

LayoutLMv3-L エンコーダは，LayoutLMv3 エンコーダとは入力に画像を使わないことが異なる．つまり入力には，表画像から OCR で得られたテキストを WordPiece[11] により分割するしたトークン $t_i (0 \le i < L)$ とそのバウンディングボックス $b_i \in (x_0, y_0, x_1, y_1)(0 \le i < L)$ のみである．よって最終的な出力は,$z \in \mathbb{R}^{(L,d)}$ に変換される．

#### 2.1.4　デコーダ

エンコーダから得られた潜在表現 $z$ を transformer デコーダを使って HTML にデコードする．デコーダの Self-Attention と Cross-Attention により，自己回帰的に HTML のトークンを生成する．

### 2.2　実験設定

Swin Transformer には，$(H, W) = (448, 896)$ の画像を入力とし，window size=7，層が [2, 2, 14, 2] のパラメータ数 $77M$ のモデルを使用した．LayoutLMv3 エンコーダには，6 層の $d = 768$，最大系列長 $L = 512$ のパラメータ数 $81M$ のモデルを使用している．また LayoutLMv3-L エンコーダも LayoutLMv3 エンコーダと同様の設定とする．ここで，3 つのモデルの比較を行うためにパラメータ数を近づけた設定にしていることに注意されたい．デコーダには，4 層の BART デコーダを使用し，$d = 1024$，最大系列長 $L = 1024$ に設定した．それぞれのモデルは事前学習済み重みで初期化を行った．モデルの学習は AdamW[12] 最適化手法を利用して，学習率を 0.0001，重み減衰を 0.02，$(\beta_1, \beta_2) = (0.9, 0.99)$ として，バッチサイズが 192 で 20 エポックが学習する．また，初期の学習の全体の 5%を wamp up させ，線形的に学習率を 0.0001 まで増加させる．また，生成する HTML の系列長や入力のテキストの系列長がモデルの最大系列長 $L$ を超える場合は，

**表 1** TEDS での評価

| | | | FinTabNet | | PubTabNet | |
|---|---|---|---|---|---|---|
| モデル | モダリティ | OCR | TEDS-Struc(%) | TEDS(%) | TEDS-Struc(%) | TEDS(%) |
| TableFormer[4] | V | ✓[1] | 96.8 | - | **96.75** | 93.60 |
| SLANet[7] | V | ✓[2] | - | - | 96.62 | 82.39 |
| Swin Transformer-BART | V | | 95.60 | 88.93 | 96.29 | **95.12** |
| PaddleOCR + LayoutLMv3-L-BART | L | ✓ | 97.21 | 94.77 | 95.06 | 90.80 |
| TesseractOCR + LayoutLMv3-L-BART | L | ✓ | 95.97 | 91.79 | 93.50 | 83.62 |
| PaddleOCR + LayoutLMv3-BART | VL | ✓ | **97.56** | **95.23** | 96.25 | 93.69 |
| TesseractOCR + LayoutLMv3-BART | VL | ✓ | 95.72 | 91.59 | 95.59 | 91.32 |

モデルの最大系列長 $L$ の長さで切り詰めることにする．また，LayoutLMv3 エンコーダ, LayoutLMv3-L エンコーダの入力には比較のために PaddleOCR[3]，TesseractOCR[4]，またアノテーションから得られた完全なテキスト[5] の 3 つを使用する．

## 2.3 データセット

**PubTabNet[6]**: PubTabNet は，509K の科学論文の表画像と対応する HTML でアノテーションされている大規模なデータセットである．このデータセットを，97%と 3%の割合で学習，評価データに利用する．

**FinTabNet[13]**: FinTabNet は,112K の S&P 500 企業の年次報告書からの複雑な表と対応する HTML 情報がアノテーションされている．また PubTabNet の表と比べてグラフィカルな線が少なく，色の変化がより多い傾向がある．このデータセットを 81%，9.5%，9.5%で学習，検証，評価データに利用する．

## 2.4 評価方法

表の構造解析の評価指標として用いられる Tree-Edit-Distance-Simmilarty(TEDS)[6] により生成した HTML の評価を行う．TEDS は以下の式で与えられる．

$$\mathrm{TEDS}(T_a, T_b) = 1 - \frac{\mathrm{EditDist}(T_a, T_b)}{\max(|T_a|, |T_b|)} \quad (1)$$

$T_a$， $T_b$ は HTML の木構造を表し，$EditDist()$ は二つの木構造編集距離を求める．また $|T|$ は $T$ のノードの数を表す．評価として HTML のタグのみの木構造を $T$ として，編集距離を求める TEDS-Struc と，HTML のすべて (タグとセルの中身) の木構造を $T$ とする TEDS の 2 つを用いて評価を行う．

**表 2** FinTabNet での Swin Transformer-BART のスコア

| 学習データ | TEDS-Struc(%) | TEDS(%) |
|---|---|---|
| fintabnet | 95.60 | 88.93 |
| pubtabnet+fintabnet | **97.06** | **95.95** |

# 3 実験結果

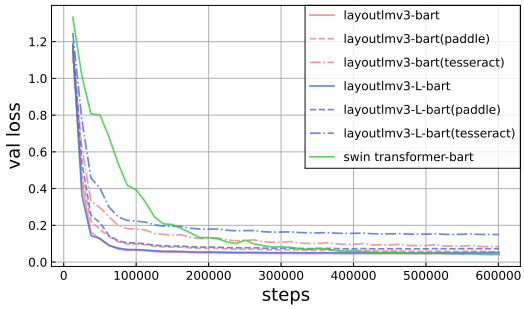



**図 3** PubTabNet においての学習曲線

表 1 に TEDS で評価した結果を示した．TableFormer と SLANet はベースラインとして追加したものである．TableFormer と SLANet どちらも画像エンコーダ・Dual デコーダの構造をしており，得られたセル座標から PDF のマッチングや OCR を用いて，セルのテキストを取得し，HTML を生成する．

**ベースラインとの比較**: 画像ベースのモデルを比較すると，PubTabNet において TableFormer や SLANet に比べて Swin Transfomrer-BART の方が TEDS が高く，このことから HTML 全てを生成する有効性を示すことができた．

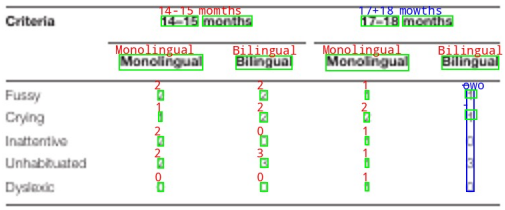
**モデルの比較:** PubTabNet では，Swin Transformer-BART が TEDS の一番スコアが高い．一方, FinTabNet

1) 得られたセルの座標より PDF とのマッチングを行う
2) 得られたセルの座標より PaddleOCR を使用する
3) https://github.com/PaddlePaddle/PaddleOCR
4) https://github.com/tesseract-ocr/tesseract
5) 正解の HTML から得られたセルのテキストとその bbox を入力とする

**表 3**　入力を完全なテキストと bbox にした際の，TEDS での評価[5)]

| | | FinTabNet | | PubTabNet | |
|---|---|---|---|---|---|
| モデル | モダリティ | TEDS-Struc(%) | TEDS(%) | TEDS-Struc(%) | TEDS(%) |
| TableVLM[8] | VL | - | - | 96.92 | - |
| LayoutLMv3-L-BART | L | 98.34 | 97.31 | 96.82 | 95.12 |
| LayoutLMv3-BART | VL | **98.6** | **97.65** | **97.11** | **95.73** |

(a)




(b)

| | 14-15 months | | 17-18 months | |
|---|---|---|---|---|
| | Monolingual | Bilingual | Monolingual | Bilingual |
| Positive | 2 | 2 | 1 | 1 |
| Positive | 1 | 2 | 2 | 1 |
| Positive | 2 | 0 | 1 | 0 |
| Positive | 2 | 1 | 1 | 0 |
| Positive | 0 | 1 | 1 | 0 |

**図 4**　OCR の出力が誤りを含むときの予測: (a) は tesseract の出力である．緑の長方形が検知したバウンディングボックスであり，青の長方形は誤っているバウンディングボックス．赤の文字が読み取ったテキストであり，青は誤って認識した文字である．(b) は (a) の OCR の出力を layoutlmv3-BART に入力したときの生成された HTML である．

では傾向が異なり，PaddleOCR+LayoutLMv3-BART が一番スコアが高く，Swin Transformer-BART が一番スコアが低い結果となっている．これは FinTabNet と PubTabNet のデータ数の違いが原因である．図 3 には，validation loss のステップごとの変化を表しているが， Swin Transformer-BART だけが極端に loss が下がるのが遅いことがわかる．このことから画像ベースのモデルには多くのデータセットが必要であり，少ないデータしかない FinTabNet で精度が悪化したと考えられる (逆に，テキストベースやマルチモーダルはデータ効率が良い)．そこで表 2 では Swin Transformer-BART を PubTabNet で学習した後に FinTabNet で finetune した結果を示した．やはりデータ数を増やすと精度が上昇することが確認できた．以上から表 1 と 2 の結果を合わせて比較すると，以下のことが言える

- 少量のデータしか得られない場合は，テキストベースやマルチモーダルが適している．
- 大量のデータが得られる状況下では，画像ベースが一番優れている

しかし，近年は大規模なデータを入手が可能なため画像ベースの方が適していると考える．

**理想的な入力の際の LayoutLM 系の TEDS**: 表 3 には, 入力に完全なテキストを使用した時の TEDS の結果を示している.TableVLM は，LayoutLMv3-BART と同じ構造であり，出力は HTML のタグのみを出力するモデルである．TEDS-struc に関しては，図 1 の (b) の tableVLM のスコアを超えており，このことから HTML のタグのみを生成するよりも HTML の全てを生成した方が結果として表の構造の予測の精度を上げられると考えられる．また, 表 1，3 の結果を比較すると LayoutLMv3-BART, LayoutLMv3-L-BART も OCR を使用した時よりも精度が向上しており, LayoutLMv3-BART が Swin Transfomrer-BART の結果を上回り非常に精度が高くなることがわかる. よって非常に精度が高い OCR を使用することができる環境ならマルチモーダルやテキストベースの方が良いが, 現状そこまで精度の高い OCR を得ることは難しいため, やはり画像ベースの方が良いと言える.

**OCR の影響**: 図 4 には， TesseractOCR から得られた文字と座標を (a) に，それらを LayoutLMv3-BART に入力した際の出力結果を (b) に示している．(a) に示すように，Tesseract から得られるテキストは誤りがある上に， 検知されない文字や，誤った bbox を出力してしまう．しかし，それらを入力してもある程度正しい表が生成されている．これはモデルが内部的に誤り訂正をしていることや，表の構造のルールを保持していることが考えられる．このことから，図 1 の (a) のように，後から OCR でセルのテキストを得るよりも，HTML 全てを生成する方法の方が優れていることがわかる．

## 4　おわりに

本研究では，HTML を生成するエンコーダ・デコーダモデルを構築し，表の構造解析に適しているエンコーダを調査した．結果として画像ベースのアプローチが良いと示唆された．

## 謝辞

本研究は，JST ムーンショット型研究開発事業 グラント番号 JPMJMS2236 の支援を受けたものです．

## 参考文献

[1] Hiroyuki Shindo Yuji Matsumoto Masashi Ishii Hiroyuki Oka, Atsushi Yoshizawa. Machine extraction of polymer data from tables using xml versions of scientific articles, science and technology of advanced materials: Methods. Vol. 1, , 2021.

[2] T. Hassan and R. Baumgartner. Table recognition and understanding from pdf files. In **Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)**, Vol. 2, pp. 1143–1147, 2007.

[3] Ermelinda Oro and Massimo Ruffolo. Pdf-trex: An approach for recognizing and extracting tables from pdf documents. In **2009 10th International Conference on Document Analysis and Recognition**, pp. 906–910, 2009.

[4] Ahmed Nassar, Nikolaos Livathinos, Maksym Lysak, and Peter Staar. Tableformer: Table structure understanding with transformers. In **Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)**, pp. 4614–4623, June 2022.

[5] Jiaquan Ye, Xianbiao Qi, Yelin He, Yihao Chen, Dengyi Gu, Peng Gao, and Rong Xiao. Pingan-vcgroup's solution for icdar 2021 competition on scientific literature parsing task b: Table recognition to html, 2021.

[6] Xu Zhong, Elaheh ShafieiBavani, and Antonio Jimeno Yepes. Image-based table recognition: data, model, and evaluation, 2020.

[7] Chenxia Li, Ruoyu Guo, Jun Zhou, Mengtao An, Yuning Du, Lingfeng Zhu, Yi Liu, Xiaoguang Hu, and Dianhai Yu. Pp-structurev2: A stronger document analysis system, 2022.

[8] Leiyuan Chen, Chengsong Huang, Xiaoqing Zheng, Jinshu Lin, and Xuanjing Huang. TableVLM: Multi-modal pre-training for table structure recognition. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, **Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)**, pp. 2437–2449, Toronto, Canada, July 2023. Association for Computational Linguistics.

[9] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In **Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)**, pp. 10012–10022, October 2021.

[10] Yupan Huang, Tengchao Lv, Lei Cui, Yutong Lu, and Furu Wei. Layoutlmv3: Pre-training for document ai with unified text and image masking, 2022.

[11] Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Lukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. Google's neural machine translation system: Bridging the gap between human and machine translation, 2016.

[12] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization, 2019.

[13] Xinyi Zheng, Doug Burdick, Lucian Popa, Xu Zhong, and Nancy Xin Ru Wang. Global table extractor (gte): A framework for joint table identification and cell structure recognition using visual context, 2020.

# A　付録

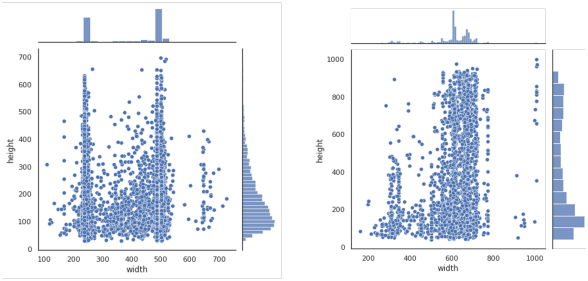
## A.1　Ablation Study

完全な HTML か HTML のタグのみを予測するかでの影響を調べた．表 2 では，Swin Transformer-BART で完全な HTML を学習したものと、HTML のタグのみを学習したものでのスコアを比較している．その結果 html のタグのみを予測するよりも TEDS-Struc の精度が上昇している．よって，HTML のタグを予測するよりも全ての HTML を出力した方が良いことがわかる．

**表 4**　TEDS での評価

| 予測対象 | TEDS-Struc(%) | TEDS(%) |
|---|---|---|
| html | **95.60** | 88.93 |
| html のタグのみ | 94.47 | - |

## A.2　データセット

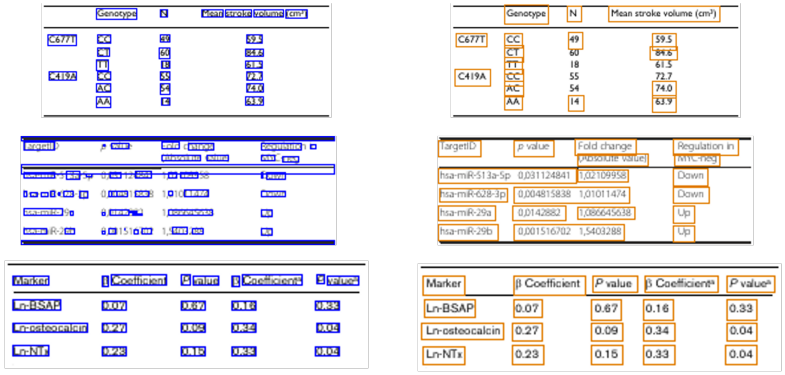



(a)PubTabNetの解像度　(b)FinTabNetの解像度

**図 5**　データセットの表画像の解像度

図 5 に実験に使用したデータセットの表画像の解像度の散布図を示した．このようにどちらも同じ分布をしている．またこの解像度の分布より Swin Transformer の解像度を (448, 896) と決定した．

## A.3　OCR




(a) **Tesseract OCR**から得られるbbox　(b) **Paddle OCR**から得られるbbox

**図 6**　OCR の例: (a) が TesseractOCR の例，(b) が PaddleOCR の例

図 6 は，実験に使用した OCR の出力の例を示している．(a) は TesseractOCR の例， (b) は PaddleOCR の例である．TesseractOCR は単語ベースで認識するのに対して，PaddleOCR ではセグメント単位で認識するという違いがある．また OCR の精度としては PaddleOCR の方が TesseractOCR よりも精度が高い．