

# Table Recognition for Optical Character Recognition (OCR)

Mahesa Lizardy /13520116  
Program Studi Teknik Informatika  
Sekolah Teknik Elektro dan Informatika  
Institut Teknologi Bandung, Jalan Ganesha 10 Bandung  
E-mail (gmail): 13520116@std.stei.itb.ac.id

**Abstract**—*Table Recognition* merupakan salah satu aspek penting dalam *Optical Character Recognition (OCR)* yang memungkinkan untuk pengenalan karakter dengan tetap mempertahankan struktur tabel dari gambar. Seringkali, mempertahankan struktur teks sangatlah penting untuk mendapatkan informasi yang komprehensif. Penelitian ini menyoroti tantangan-tantangan utama yang terkait dengan variasi format tabel dan struktur dokumen yang beragam. Berbagai pendekatan seperti *deep learning* dan *image processing* sudah banyak digunakan pada penelitian ini. Pada makalah ini akan digunakan pendekatan *image processing* untuk melakukan *table recognition*

**Keywords**—*component; table recognition, table extraction, optical character recognition, OCR*

## I. PENDAHULUAN

*Table recognition* adalah proses mengekstraksi struktur dan konten tabel dari gambar atau dokumen. Proses ini melibatkan tugas-tugas seperti *table detection*, *table structure recognition*, dan *table content recognition*. *Table detection* akan menemukan tabel pada gambar. *Table structure recognition* akan mengidentifikasi baris, kolom, dan posisi dari sel. *Table content recognition* mengidentifikasi dan mengekstrak konten teks dari tabel (2). Pengenalan tabel biasanya menggunakan model *deep learning* atau *image processing* sederhana untuk mendeteksi tabel (*table detection*). *Table recognition* telah digunakan di beberapa bidang seperti *document analysis*, *data extraction*, dan *information organization*.

*Table detection* dapat dilakukan dengan beberapa pendekatan, seperti metode *deep learning*, maupun metode *image processing*. Meskipun pendekatan *deep learning* telah mendapatkan perhatian yang signifikan dalam beberapa tahun terakhir, metode *image processing* masih relevan dan memiliki keunggulan tersendiri. Teknik *image processing* seperti dilasi, erosi, dan deteksi kontur dapat digunakan untuk mengidentifikasi tabel dan mengekstrak informasi tabel.

Data tabel seringkali tersebar dalam berbagai gambar dan dokumen yang dapat ditemukan secara luas disektor bisnis maupun kita temukan di internet. Namun, pada saat melakukan proses pengenalan atau pengambilan informasi dari tabel tersebut, penting untuk memiliki pemahaman yang kuat

terhadap struktur tabel itu sendiri. Hal ini memungkinkan kita untuk memperoleh informasi secara komprehensif dan akurat. Hal tersebut dapat sangat berguna dalam berbagai aplikasi seperti analisis data hingga pengolahan informasi lebih lanjut. Oleh karena itu, *table recognition* diperlukan untuk mengekstrak informasi dari tabel pada gambar atau dokumen.

## II. LANDASAN TEORI

### A. Image Processing

*Image processing* adalah suatu cabang ilmu computer vision yang berkaitan dengan manipulasi gambar atau citra digital menggunakan teknik-teknik komputasi. Tujuannya adalah untuk meningkatkan kualitas gambar, mengubah karakteristiknya, atau menghasilkan informasi baru dari gambar-gambar tersebut. *Image processing* mencakup berbagai teknik untuk mengolah gambar, termasuk analisis, pemrosesan, pengenalan pola, dan pemodelan. Pada kebanyakan *computer vision application*, *Image processing* digunakan untuk pra-proses gambar serta mengubahnya menjadi bentuk yang sesuai untuk analisis lebih lanjut (4).

Dalam *image processing*, gambar digital diwakili sebagai kumpulan piksel (titik-titik kecil), di mana setiap piksel memiliki nilai yang mewakili intensitas cahaya atau warna pada posisi tertentu. Teknik-teknik *image processing* dapat digunakan untuk memanipulasi piksel-piksel ini, seperti mengubah kecerahan, kontras, atau warna gambar. Selain itu, teknik pengolahan citra juga dapat digunakan untuk menghilangkan noise (gangguan), mengidentifikasi objek, mendeteksi tepi, dan merancang filter untuk menyoroti atau menghapus elemen-elemen tertentu dalam gambar.

### B. Citra Biner

Citra biner adalah citra yang hanya memiliki dua nilai graylevel, 0 dan 1 atau representasi lainnya. 0 merupakan representasi untuk putih, sedangkan 1 merupakan representasi untuk hitam (1). Citra biner digunakan untuk beberapa keperluan seperti kebutuhan memori untuk setiap pixel sedikit (hanya 1 bit), digunakan untuk pendeteksian tepi, digunakan untuk memisahkan (segmentasi) objek dari gambar latar belakangnya, dan lain-lain.

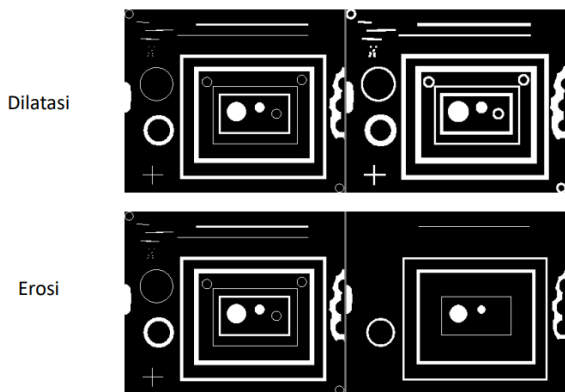


Gambar. 1. Contoh beberapa Citra biner (1)

Terdapat beberapa operasi morfologi pada citra biner. Morfologi adalah sekumpulan operasi pengolahan citra yang memproses citra berdasarkan bentuknya. Operasi morfologi menyederhanakan citra, mengukur dan mempertahankan bentuk utama objek. Selain itu, operasi morfologi dapat juga berupa mengisi *pixel-pixel* yang hilang. Beberapa operasi morfologi diantaranya Dilatasi (*dilation*), Pengikisan atau erosi (*erotion*), Pembukaan (*opening*), dan Penutupan (*closing*).

#### C. Dilatasi (Dilasi) & Erotion (Erosi)

Operasi dilasi (*dilation*) bertujuan untuk memperlebar bentuk dari suatu objek. Sedangkan erosi (*erotion*) bertujuan untuk mengikis bentuk objek. Dilatasi dan erosi adalah operasi dasar, dapat digabungkan menjadi rangkaian yang lebih kompleks (1). Erosi dan dilatasi bukan operasi yang dapat dibalik (*not invertible*). *Not invertible* adalah jika sesuatu citra dikikis atau melebar, citra aslinya tidak dapat diperoleh kembali (1).



Gambar. 2. Contoh operasi Dilatasi dan Erosi (1)

Penggunaan operasi ini sangat penting dalam beberapa aplikasi seperti segmentasi citra, deteksi tepi, dan pengolahan citra lainnya. Operasi ini digunakan ketika perubahan bentuk atau struktur objek merupakan bagian penting dari analisis dan pemrosesan data gambar.

#### D. Pengambangan (*Thresholding*)

Pengambangan (*Thresholding*) dapat digunakan untuk memisahkan objek dengan latar belakangnya. *Thresholding*

dilakukan berdasarkan pada nilai intensitas *pixel-pixel* dan sebuah nilai ambang *T*. Berdasarkan nilai ambang tersebut, setiap *pixel* citra yang lebih dari nilai ambang tersebut akan dianggap titik objek, sedangkan yang kurang dari nilai ambang akan dianggap latar belakang. Hasil segmentasi adalah segmen *foreground* dan segmen *background* (1).

$$f_B(i, j) = \begin{cases} A, & f_g(i, j) \leq T \\ B, & \text{lainnya} \end{cases}$$

Gambar. 3. Operasi *thresholding* (1)

Terdapat beberapa cara menentukan nilai ambang batas *T*. Salah satunya yaitu berdasarkan analisis histogram dengan mengidentifikasi puncak dan lembah. Lembah terdalam diantara dua bukit menyatakan nilai *T*. Teknik pengambangan dapat dibagi menjadi *Global Thresholding* (nilai ambang bergantung pada keseluruhan nilai-nilai *pixel*), *Local Thresholding* (Nilai ambang bergantung pada *pixel-pixel* bertetangga, hanya untuk sekelompok *pixel* saja), dan *Adaptive Thersholding* (Nilai ambang berubah secara dinamis bergantung pada perubahan pencahayaan di dalam citra).

#### E. Contour Detection

Kontur merupakan rangkaian *pixel-pixel* tepi membentuk batas daerah (*region boundary*). Kontur dapat terbuka atau tertutup. Kontur tertutup berkoresponden dengan batas yang mengelilingi suatu daerah. *Pixel-pixel* di dalam daerah tersebut dapat ditemukan dengan algoritma pengisian (*filling algorithm*). Batas daerah berguna untuk mendiskripsikan bentuk objek dalam tahap analisis citra yang digunakan untuk mengenali objek tersebut (1).

*Contour detection* melibatkan identifikasi batas-batas objek dalam suatu gambar, yang berguna untuk berbagai aplikasi seperti *object recognition*, *shape analysis*, dan *image segmentation*. Algoritma pendeteksian kontur dapat bekerja dengan baik apabila terdapat kontras yang tinggi antara objek dengan latar belakangnya.

#### F. Table Detection

*Table Detection* merupakan proses mengidentifikasi tabel dalam dokumen, gambar, atau sumber data lainnya. *Table Detection* merupakan tugas penting dalam analisis gambar karena memungkinkan pengambilan informasi penting dari tabel. Terdapat beberapa model *deep learning* untuk deteksi tabel. Selain menggunakan *deep learning*, metode *image processing* seperti operasi morfologi pada citra biner dapat digunakan untuk mendeteksi tabel pada gambar dan dokumen.

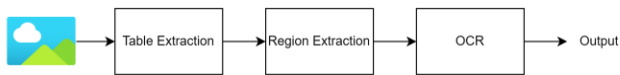
#### G. Tesseract

Tesseract merupakan salah satu *Optical Character Recognition* (OCR) *open source* yang dikembangkan oleh Google (3). Tesseract digunakan untuk mengenali teks pada gambar dan mengubahnya ke dalam teks digital. Arsitektur tesseract meliputi beberapa langkah seperti segmentasi karakter, pengenalan karakter, dan perakitan teks.

Tesseract telah dikenal luas dan digunakan di beberapa aplikasi pengenalan karakter. Namun, kinerja tesseract bergantung pada kualitas citra input dan kompleksitas struktur halaman serta apakah teks pada gambar merupakan teks hasil cetakan mesin atau tulisan tangan manusia (3).

### III. PEMBAHASAN

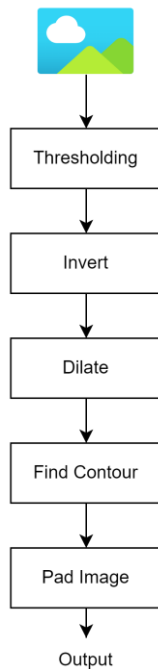
Untuk melakukan *Table Recognition* pada gambar dan dokumen, gambar atau dokumen akan melalui tiga *pipeline* yaitu *Table Extraction*, *Region Extraction*, dan *OCR*. *Pipeline* tersebut dapat diilustrasikan pada gambar berikut



Gambar. 4. *Pipeline table recognition*.

#### A. *Table Extraction*

*Pipeline table extraction* bertujuan untuk mengekstrasi tabel dari gambar/dokumen. Pada *pipeline table extraction* akan dilakukan beberapa proses seperti pengambangan (*thresholding*), *invert*, *dilate* (dilatasi), *find contour*, dan *pad image* yang dapat dilihat pada gambar berikut.



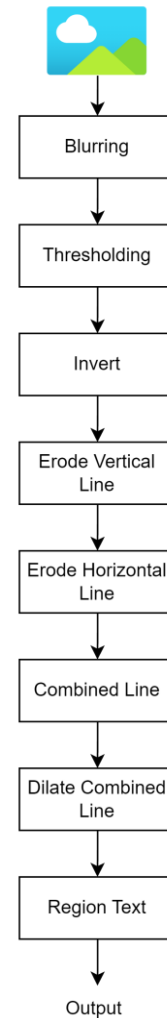
Gambar. 5. Alur proses pada *pipeline table extraction*

*Thresholding* akan menghasilkan citra biner dari masukan gambar/dokumen. Hasil *thresholding* tersebut akan dilakukan *Invert* untuk mengubah nilai *pixel* pada gambar menjadi *pixel* yang berlawanan. *Dilatasi* akan menebalkan atau melebarkan objek pada citra yang nantinya akan dicari kontur berbentuk persegi yang merupakan representasi dari tabel. Setelah tabel

ditemukan, maka gambar akan di-*crop* pada bagian tabel tersebut dan akan dilakukan *padding* sebesar 5% dari tinggi gambar masukan.

#### B. *Region Extraction*

*Pipeline* kedua yaitu *region extraction*. *pipeline* ini bertujuan untuk mengekstrasi *region-region* dari teks. Pada *pipeline* tersebut akan dilakukan beberapa proses seperti *blurring*, *thresholding*, *invert*, *erode vertical line*, *erode horizontal line*, *combined line*, *dilate combined line*, dan *region text*.



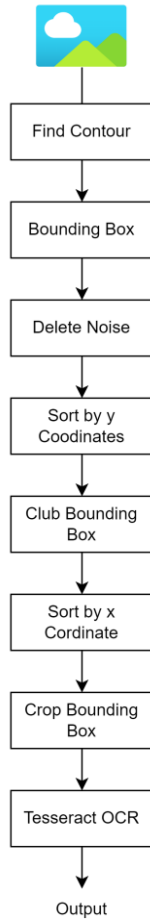
Gambar. 6. Alur proses pada *pipeline Region Extraction*

*Thresholding* akan menghasilkan citra biner dari masukan gambar/dokumen hasil dari *pipeline sebelumnya*. Hasil *thresholding* tersebut akan dilakukan *Invert* untuk mengubah nilai *pixel* pada gambar menjadi *pixel* yang berlawanan. Berdasarkan hasil *invert* tersebut akan dilakukan erosi untuk mendapatkan *horizontal line* dan *vertical line*. *Horizontal line* dan *vertical line* akan digabungkan dan akan dilakukan *dilate* untuk menebalkan garis garis tersebut. Berdasarkan hasil

tersebut akan di-invert kembali dan menghilangkan semua elemen diluar tabel untuk mendapatkan *region text*.

### C. OCR

*Pipeline* terakhir yaitu OCR. *pipeline* ini bertujuan untuk melakukan pengenalan karakter pada tabel setiap sel-nya dan menyusun hasil OCR menjadi informasi terstruktur. Pada *pipeline* akan dilakukan beberapa proses seperti *find-contour*, *bounding box*, *delete Noise*, *sort by y coordinate*, *club boundingbBox*, *sort by x coordinate*, *crop bounding box*, dan *tesseract OCR*.



Gambar. 7. Alur proses pada *pipeline* OCR

*Find Contour* bertujuan mencari kontur berdasarkan hasil dari *pipeline Region Extraction* yang sudah dilakukan sebelumnya. Berdasarkan hasil kontur-kontur yang ditemukan akan dibentuk *bounding box* disetiap kontur nya. Terkadang terdapat beberapa noise dari pembentukan *Bounding Box* sehingga perlu dihilangkan dari list *bounding box*. List *bounding box* akan diurutkan berdasarkan nilai kordinat y dan dikelompokan. Setelah dikelompokan akan diurutkan berdasarkan kordinat x. Setiap *bounding box* pada *image* atau citra akan di-*crop* berdasarkan kordinat dari *bounding box* tersebut. Setelah di-*crop* akan dilakukan pengenalan menggunakan *tesseractOCR* secara satu-persatu.

### D. Hasil Table Recognition

Berikut akan digunakan beberapa gambar untuk menguji hasil *table recognition* yang sudah dikembangkan. Citra yang akan diuji adalah sebagai berikut.

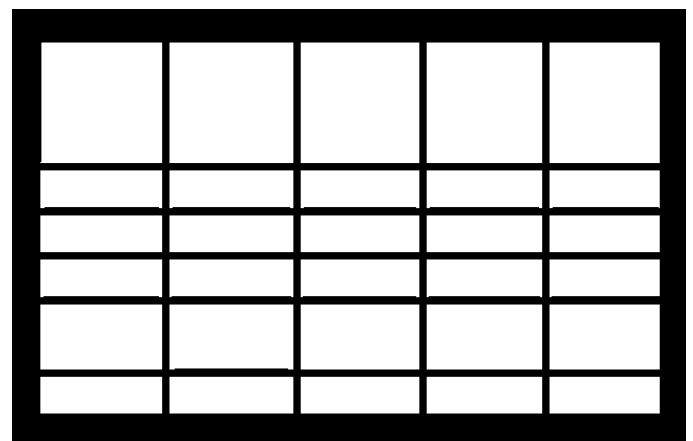
Language	No. of chars (millions)	No. of words (millions)	Char error rate (%)	Word error rate (%)
English	39	4	0.5	3.72
EFIGSD	213	26	0.75	5.78
Russian	38	5	1.35	5.48
Simplified Chinese	0.25	NA	3.77	NA
Hindi	1.4	0.33	15.41	69.44

### 7. Conclusions & Future Work

Gambar. 8. Citra masukan pertama (source: <https://d3i71xaburhd42.cloudfront.net/0a482d33dc0c7ca08cc333158c5c0e82d4c04560/8-Table1-1.png>)

Language	No. of chars (millions)	No. of words (millions)	Char error rate (%)	Word error rate (%)
English	39	4	0.5	3.72
EFIGSD	213	26	0.75	5.78
Russian	38	5	1.35	5.48
Simplified Chinese	0.25	NA	3.77	NA
Hindi	1.4	0.33	15.41	69.44

Gambar. 9. Output *pipelane table extraction* citra pertama



Gambar. 10. Output *pipelane Region Extraction* citra pertama

```
[
  [
    "Language",
    "No. of chars (millions)",
    "No. of words (millions)",
    "Char error rate (*)",
    "Word error rate (%)"
  ],
  ["English", "39", "4", "0s", "3.72"],
  ["EFIGSD", "213", "26", "0.75", "5.78"],
  ["Russian.", "38", "5", "135", "5.48"],
  ["Simplified Chinese", "0.25", "NA", "3.77",
"NA"],
  ["Hindi", "14", "0.33", "15.41", "69.44"]
]
```

Gambar. 11. Citra output *pipeline* OCR citra pertama

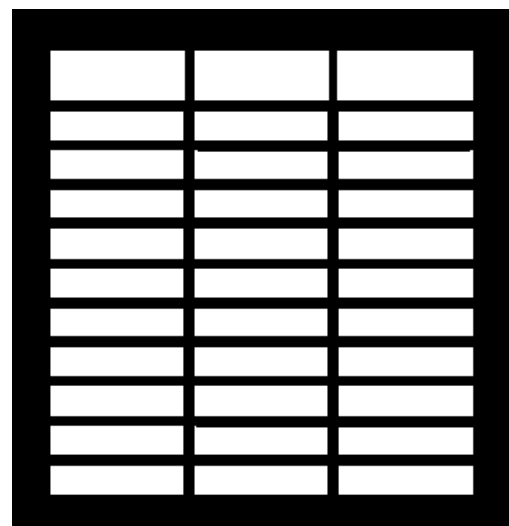
Student	Study Time (hours)	Grade
Bob	2	84
Carlos	4	91
Cindy	5	92
Florence	3	89
Kim	4	88
Lori	4	93
Marisa	1	78
Pat	2	89
Thomas	5	94
Wendy	2.5	87

Gambar. 13. Output *pipelane* table extraction citra kedua

### Study Time vs. Grades

Student	Study Time (hours)	Grade
Bob	2	84
Carlos	4	91
Cindy	5	92
Florence	3	89
Kim	4	88
Lori	4	93
Marisa	1	78
Pat	2	89
Thomas	5	94
Wendy	2.5	87

Gambar. 12. citra masukan kedua (source: [https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcSrrpHWRIQdbO-SfMg6puUFew44paOOX78C\\_D-yeDiM502vryJ5](https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcSrrpHWRIQdbO-SfMg6puUFew44paOOX78C_D-yeDiM502vryJ5))



Gambar. 14. Citra output *pipeline* OCR citra pertama

```
[
  ["Student", "Study Time (hours)", "Grade"],
  ["Bob", "2", "End"],
  ["Carles.", "4", "a1"],
  ["Cindy", "5", "92"],
  ["Florence", "3", "89"],
  ["Kim", "4", "88"],
  ["Lori", "4", "93"],
  ["Marisa", "1", "78"],
  ["Pat", "2", "89"],
  ["Thomas", "5", "94"],
  ["Wendy", "25", "87"]
]
```

Gambar. 15. Output *pipelane* OCR citra kedua

#### IV. KESIMPULAN & SARAN

Metode *image processing* dapat digunakan untuk melakukan *table recognition*. Dengan menggunakan metode tersebut didapatkan hasil yang cukup baik. Terdapat beberapa hal yang bisa ditingkatkan seperti mengubah parameter yang terdapat pada kode seperti konfigurasi OCR yang digunakan, serta parameter-parameter pada *image processing*.

#### VIDEO LINK AT YOUTUBE (*Heading 5*)

Untuk memberikan gambaran tentang persoalan yang diangkat dan solusinya, penjelasan dalam bentuk video dapat dilihat dengan mengakses video pada link berikut [https://youtu.be/\\_MmiaqgzUNE](https://youtu.be/_MmiaqgzUNE).

#### UCAPAN TERIMA KASIH

Penulis mengucapkan rasa syukur kepada Tuhan Yang Maha Esa yang telah mengizinkan penulis untuk menyelesaikan makalah sederhana untuk mengeksplorasi teknik-teknik yang dapat digunakan untuk melakukan color grading terhadap suatu citra tujuan tertentu. Ucapan terima kasih juga disampaikan kepada Bapak Dr. Ir. Rinaldi Munir, M.T., yang telah memberikan bimbingan dan ilmu dalam mata kuliah interpretasi dan pengolahan citra. Penulis juga mengucapkan terima kasih kepada asisten. Penulis juga mengucapkan terima kasih kepada peneliti-peneliti terdahulu yang dirujuk melalui referensi.

#### REFERENCES

- [1] R. Munir, Pohon, Slide Kuliah Interpretasi dan Pengolahan Citra IF4073 2023/2024
- [2] Rashid, S. F., Akmal, A., Adnan, M., Aslam, A. A., & Dengel, A. (2017). Table Recognition in Heterogeneous Documents Using Machine Learning. In 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR). 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR). IEEE. <https://doi.org/10.1109/icdar.2017.132>
- [3] Smith, R. (2007). An Overview of the Tesseract OCR Engine. In Ninth International Conference on Document Analysis and Recognition (ICDAR 2007) Vol 2. Ninth International Conference on Document Analysis and Recognition (ICDAR 2007) Vol 2. IEEE. <https://doi.org/10.1109/icdar.2007.4376991>
- [4] Szeliski, R. (2010) Computer vision: Algorithms and applications. Springer.

#### PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 19 November 2023



Mahesa Lizardy  
13520116