



Department of Digital Business

Journal of Artificial Intelligence and Digital Business (RIGGS)

Homepage: <https://journal.ilmudata.co.id/index.php/RIGGS>

Vol. 4 No. 2 (2025) pp: 6586-6594

P-ISSN: 2963-9298, e-ISSN: 2963-914X

Ekstraksi Informasi Struk Belanja Melalui Pemanfaatan Tesseract dan Regular Expressions

Amelia Ananda Setiawan, Rangga Gelar Guntara, Btari Mariska Purwaamijaya

^{1,2,3} Bisnis Digital, Kampus Daerah Tasikmalaya, Universitas Pendidikan Indonesia

¹amelananda@upi.edu, ²ranggagelar@upi.edu, ³btarimariska@upi.edu

Abstrak

Pencatatan keuangan merupakan hal yang penting agar seseorang dapat menyadari posisi keuangan dan memantaunya dengan baik. Salah satu bukti yang digunakan dalam pencatatan keuangan yaitu berasal dari struk belanja. Namun, proses pencatatan keuangan masih dianggap menyulitkan karena masih mengandalkan input data secara manual. Penelitian ini bertujuan untuk membangun sistem ekstraksi struk belanja untuk membantu proses pencatatan keuangan. Sistem dirancang dengan menerapkan model YOLO untuk mendeteksi area, Tesseract untuk mengekstrak teks dari gambar, dan Regular Expressions untuk mengolah data. Hasil pengujian alpha menggunakan black box menunjukkan seluruh fungsionalitas berjalan dengan baik. Selain itu, hasil pada pengujian beta menggunakan user acceptance testing juga menunjukkan bahwa sistem dinilai membantu pengguna dalam proses pencatatan keuangan dengan rata-rata nilai 4,47 dari skala 5.

Kata kunci: Pencatatan Keuangan, Struk Belanja, Ekstraksi, YOLO, Tesseract, Regular Expressions

1. Latar Belakang

Belanja merupakan salah satu aktivitas pengeluaran sehari-hari sebagai upaya pemenuhan kebutuhan baik primer, sekunder, maupun tersier [1]. Dalam aktivitas belanja, masyarakat memiliki preferensi tempat yang beragam. Berdasarkan data dari Populix, sebanyak 77% masyarakat Indonesia lebih memilih berbelanja di minimarket, diikuti 58% di supermarket, dan 26% di hypermarket [2]. Selain itu, belanja juga dapat terjadi di berbagai tempat lain seperti restoran, kafe, atau pusat perbelanjaan. Dari data tersebut mencerminkan bahwa belanja merupakan aktivitas rutin yang menghasilkan suatu transaksi.

Dalam setiap transaksi belanja, terdapat catatan yang dihasilkan berupa struk sebagai bukti pembayaran. Terdapat informasi penting dalam struk belanja seperti produk atau jasa yang dibeli, kuantitas, harga satuan, serta total pembelian. Informasi ini tidak hanya berfungsi sebagai bukti transaksi, tetapi memiliki manfaat untuk membantu individu dalam memantau pengeluaran melalui pencatatan keuangan [3]. Pencatatan keuangan merupakan hal yang penting karena dapat membantu seseorang dalam mencapai tujuan hidup [4]. Pencatatan keuangan memudahkan pemiliknya untuk mengatur keuangan dengan baik. Lebih lanjut dalam penelitiannya menyatakan bahwa dengan pencatatan keuangan, seseorang dapat menyadari posisi keuangan mereka karena pemasukan maupun pengeluaran dapat terpantau dengan baik [4].

Namun, dalam praktiknya, pencatatan keuangan secara manual dianggap menyulitkan karena dokumen rentan hilang atau terlupakan [4]. Selain itu, pencatatan menggunakan pulpen dan kertas juga kurang efisien terutama jika jumlah kertas yang digunakan semakin banyak. Survei yang dilakukan oleh Jenius terhadap 2.619 responden juga menunjukkan bahwa 6 dari 10 orang tidak terbiasa mencatat pengeluaran pribadinya. Dari jumlah tersebut, 40,5% beralasan malas, 31% lupa, dan 16,1% belum menemukan cara yang efektif [5]. Hal tersebut menunjukkan bahwa masih banyak orang yang kesulitan untuk mencatat keuangan.

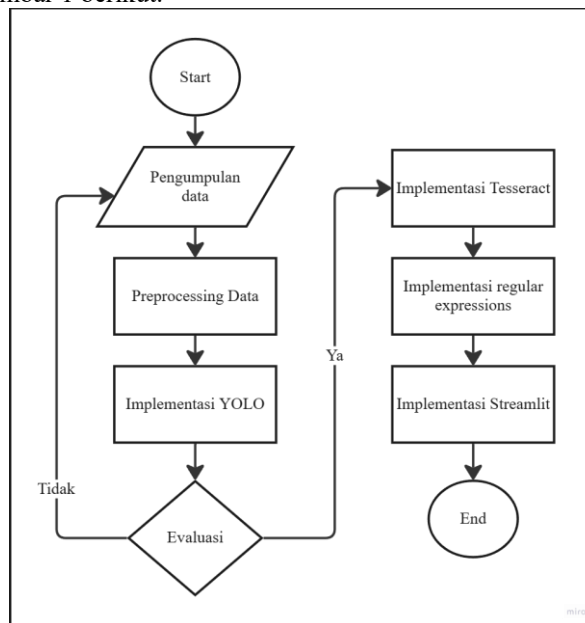
Berdasarkan data tersebut, salah satu solusi untuk mengatasi permasalahan sulitnya mencatat keuangan khususnya manual karena malas yakni melalui aplikasi pencatatan digital. Namun, meskipun berbagai aplikasi pencatatan keuangan telah tersedia, fitur pencatatan otomatis masih belum banyak dilakukan. Terdapat studi yang menyatakan bahwa 10 aplikasi pencatatan keuangan tidak memiliki fitur *Optical Character Recognition* (OCR) [6]. Hal ini menunjukkan bahwa pengguna masih harus memasukkan data transaksi secara manual, yang dapat menjadi hambatan utama dalam pencatatan keuangan yang lebih efisien. Proses ini tidak hanya memakan waktu tetapi juga berpotensi meningkatkan risiko kesalahan input. Oleh karena itu, dalam penelitian ini dilakukan pengembangan

sistem berbasis OCR yang mampu mengekstrak struk belanja menjadi data terstruktur bertujuan untuk membantu proses pencatatan keuangan.

2. Metode Penelitian

Dalam penelitian ini, metode yang digunakan yaitu *System Development Life Cycle (SDLC)* dengan pendekatan model *waterfall*. Pendekatan ini dilakukan secara linear dan dilakukan secara bertahap (*sequential*) [7]. Namun, penelitian ini hanya dilakukan sampai pada tahap *testing* atau pengujian. Tahap *deployment* dan *maintenance* tidak dilakukan karena sistem yang dikembangkan bersifat prototipe.

- 2.1 Analisis: tahap ini berfokus pada analisis yang berkaitan dengan pembangunan sistem ekstraksi struk belanja. Tahapan dilakukan mulai dari analisis sistem yang sedang berjalan, sistem usulan, dan teknologi yang digunakan.
- 2.2 Perancangan: tahap ini berfokus pada menyusun rancangan dari YOLO, Tesseract, dan *regular expressions*.
- 2.3 Implementasi: tahap ini dilakukan implementasi dari analisis dan perancangan yang telah dibuat dengan beberapa tahap seperti Gambar 1 berikut.



Gambar 1. Alur Implementasi

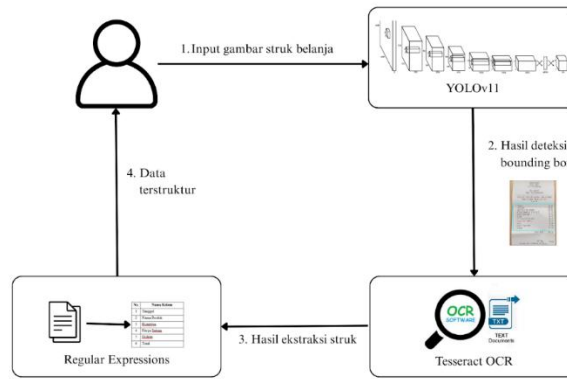
- 2.4 Pengujian: pada tahap ini pengujian dilakukan dengan 2 cara yaitu pengujian *alpha* dan *beta* [8]. Pengujian *alpha* dilakukan menggunakan *blackbox testing* untuk memastikan sistem berjalan sesuai dengan spesifikasi yang telah ditentukan [9], sedangkan pengujian *beta* dilakukan dengan kuesioner untuk memperoleh masukan langsung mengenai pengalaman pengguna dan sejauh mana sistem memenuhi kebutuhan.

3. Hasil dan Diskusi

3.1. Analisis

Saat ini, sistem yang sedang berjalan dilakukan dengan dua metode. Setelah *user* berbelanja dan menerima struk, metode pertama yang dilakukan yakni membuka buku pencatatan yang selanjutnya setiap informasi akan ditulis secara manual. Adapun metode yang kedua yaitu pengguna beralih ke pencatatan digital melalui aplikasi, namun proses *input* tetap dilakukan secara manual oleh pengguna.

Sistem yang diusulkan menawarkan pendekatan otomatis, yaitu setelah menerima struk, struk tersebut kemudian dimasukkan ke dalam sistem dengan memfotonya. Selanjutnya, data dari struk belanja akan otomatis terinput. Dalam pembangunannya, komponen dan metode yang digunakan yaitu dengan penerapan YOLO, Tesseract, dan *regular expressions* dengan arsitektur dan alur seperti Gambar 2.



Gambar 2. Arsitektur Sistem

3.2. Perancangan

Pada tahap ini, dilakukan perancangan dari analisis yang sudah dibuat. Proses perancangan dilakukan mulai dari YOLO sebagai model untuk mendeteksi objek struk belanja. YOLO dirancang untuk mendeteksi area struk belanja yang berisi informasi nama produk, kuantitas, harga satuan, harga total, dan diskon seperti Gambar 3. Oleh karena itu model YOLO akan dilatih menggunakan data yang sudah dianotasi sehingga menghasilkan *bouding box* yang sudah ditentukan. Hasil dari deteksi ini yang kemudian akan menjadi dasar dalam pemotongan area gambar untuk diproses lebih lanjut menggunakan Tesseract.

| Struk Belanja | | | |
|---------------------|-----|----------------|-------|
| Bandung, Jawa Barat | | | |
| nama produk | qty | harga per unit | total |
| air botol | 1 | 100 | 100 |
| mie | 1 | 200 | 200 |
| beras | 1 | 300 | 300 |
| diskon | | | (10) |
| | | TOTAL | 550 |

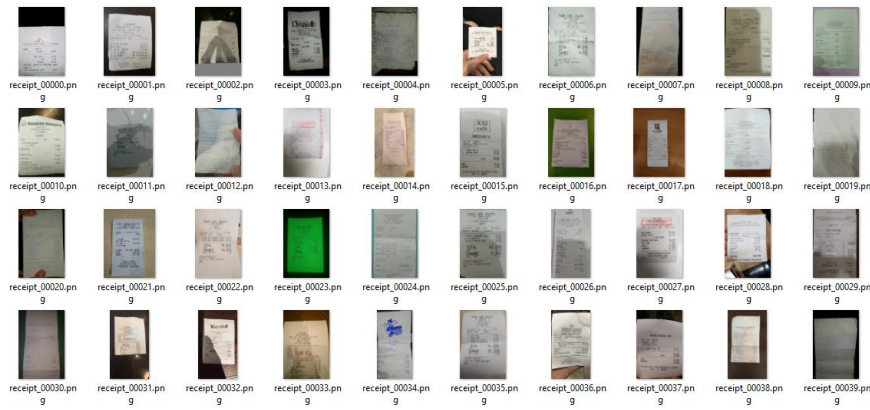
Gambar 3. Perancangan Area YOLO

Setelah perancangan YOLO selesai, proses selanjutnya yaitu ekstraksi teks. Tesseract diterapkan dengan parameter `-oem 3` dan `psm default`. Kemudian untuk *regular expressions*, proses ini dirancang untuk membaca 2 pola, yaitu diskon dan normal. Pola diskon ditandai dengan dalam kurung diikuti digit. Untuk pola normal, terdapat pola yang disesuaikan berdasarkan variasi struk untuk membaca item dengan struktur berupa nama produk kuantitas, harga satuan, dan total harga.

3.3. Implementasi

3.3.1. Pengumpulan Data

Pada tahap ini, data struk belanja dikumpulkan melalui dua sumber, yaitu data primer dan sekunder. Data primer terdiri atas 40 struk yang dikumpulkan secara langsung, sedangkan data sekunder diperoleh dari dua sumber, yaitu 200 data dari website ExpressExpense [10] serta dataset CORD (*Consolidated Receipt for Post-OCR Parsing*) [11] dengan contoh data dapat dilihat pada Gambar 4.



Gambar 4. Contoh Hasil Dataset

3.3.2. Preprocessing Data

Pada tahap ini terdapat 3 tahap utama yakni anotasi, pembagian, dan augmentasi data. Anotasi data dilakukan dengan memberi label pada area yang berisi nama produk, harga, dan kuantitas seperti Gambar 5. Tujuan dari tahap ini yaitu agar nantinya model dapat mengenali dan belajar dari objek yang sudah diberi label.



Gambar 5. Contoh Hasil Anotasi Data

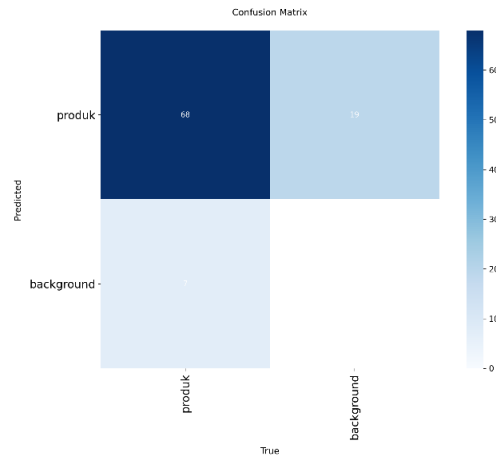
Setelah dianotasi, terkumpul 450 data yang selanjutnya akan dibagi. Proses pembagian dilakukan dengan proporsi 80% *train*, 10% *valid*, dan 10% *test*. Namun karena saat augmentasi data *train* akan bertambah 2 kali lipat, maka dengan persentase yang sama pada tahap ini pembagian menjadi 300 data untuk *train*, 75 untuk *valid*, dan 75 untuk *test*. Setelah dibagi, data *train* kemudian di augmentasi agar data menjadi lebih bervariasi serta meminimalisir *overfitting*. Proses augmentasi dilakukan menggunakan Roboflow dengan pengaturan seperti Tabel 1. Adapun hasil dari augmentasi data *train* menjadi 600.

Tabel 1. Pengaturan Augmentasi

| No | Jenis Augmentasi | Keterangan |
|----|------------------|-------------|
| 1 | Rotasi | -5% dan +5% |
| 2 | Grayscale | 10% |
| 3 | Noise | 0.3% |
| 4 | Blur | 0.6 pixel |

3.3.3. Implementasi dan Evaluasi YOLO

Pada tahap ini, model YOLO dilatih menggunakan data yang sudah di *preprocessing*. Model dilatih menggunakan Google Colab karena menyediakan berbagai layanan seperti CPU, GPU, dan TPU dengan performa yang cukup stabil [12]. Model dilatih dengan inisialisasi parameter berupa *image size* 640, *batch size* 16, iterasi 80 *epochs*, *patience* 20, dan *learning rate* 0.001. Setelah proses pelatihan selesai, model kemudian dievaluasi menggunakan data *test* dengan beberapa *metrics* yakni *confusion matrix*, *precision*, *recall*, *f1 score*, dan mAP yang hasilnya ditunjukkan pada Gambar 6 serta Tabel 2. Setelah di evaluasi, didapatkan model terbaik yang nantinya akan digunakan pada tahap implementasi Tesseract.



Gambar 6. Hasil *Confusion Matrix*

Tabel 2. Hasil Evaluasi Model

| Data | Precision | Recall | F1 Score | mAP 50 | mAP 50-95 |
|------|-----------|--------|----------|--------|-----------|
| test | 0.90 | 0.85 | 0.88 | 0.93 | 0.67 |

Hasil evaluasi tersebut menggambarkan bahwa model memiliki performa cukup baik dalam mendeteksi objek khususnya struk belanja dengan contoh hasil seperti Gambar 7. Temuan ini sejalan dengan penelitian yang membandingkan berbagai model deteksi objek dan menemukan bahwa YOLO menjadi model terbaik untuk mendeteksi struk belanja [13].



Gambar 7. Contoh Hasil Deteksi YOLO

3.3.4. Implementasi Tesseract

Pada tahapan ini, struk belanja yang diekstrak diunggah terlebih dahulu kemudian dilakukan *preprocessing* dengan mengubahnya menjadi *grayscale*. Model YOLO yang sudah dilatih akan mendeteksi area berupa *bounding box* pada struk yang diunggah. Area tersebutlah yang selanjutnya diproses menggunakan Tesseract untuk mengkonversi gambar menjadi teks seperti Gambar 8.

IDM KTG PLSTK 1W BSR 1 300 300
DAIA DET+SFT VLT 800 1 17900 17, 900
VC SAYAP MAS UTA : (1, 900)
SELECTION KAPAS 50GR 1 11600 11, 600
TOTOLE KLD JMUR 80G 1 13300 13, 300
DERMA PTCH PLS DAY6S 1 19200 19, 200
VITACIMIN STRIP 2'S 1 2300 2,300 —
US SKT. BAJU 20930 1 25000 25, 000

Gambar 8. Contoh Hasil Tesseract

Dari hasil tersebut, Tesseract sebagai OCR berjalan baik untuk mengekstrak teks dari gambar struk belanja. Namun, pada kondisi tertentu seperti struk yang terlipat, buram, atau mengandung banyak *noise*, hasil ekstraksi menjadi kurang akurat. Hal ini disebabkan oleh keterbatasan Tesseract dalam mengenali karakter ketika kualitas gambar kurang optimal.

3.3.5. Implementasi *Regular Expressions*

Pada tahapan ini, teks mentah yang sudah diekstrak menggunakan Tesseract kemudian diolah dengan mengidentifikasi dan memetakan pola tertentu. Adapun regex dalam penelitian ini berfokus untuk mengenali pola berupa nama produk, kuantitas, harga satuan, total harga, dan diskon sehingga menjadi data terstruktur seperti Gambar 9. Secara keseluruhan, Regex mampu memetakan data sesuai tujuan. Namun meskipun fleksibel untuk digunakan, Regex tetap memiliki limitasi karena bergantung pada aturan *parsing* yang sudah ditetapkan sehingga kurang adaptif terhadap variasi format struk yang berbeda [14]

| | Product | qty | Harga per Unit | Total Price | Discount |
|---|----------------------|------|----------------|-------------|----------|
| 0 | IDM KTG PLSTK 1W BSR | 1 | 300 | 300 | None |
| 1 | DAIA DET+SFT VLT 800 | 1 | 17,900 | 17,900 | None |
| 2 | VC SAYAP MAS UTA | None | None | None | 1900 |
| 3 | SELECTION KAPAS 50GR | 1 | 11,600 | 11,600 | None |
| 4 | TOTOLE KLD JMUR 80G | 1 | 13,300 | 13,300 | None |
| 5 | DERMA PTCH PLS DAY6S | 1 | 19,200 | 19,200 | None |
| 6 | VITACIMIN STRIP 2'S | 1 | 2,300 | 2,300 | None |
| 7 | US SKT BAJU 20930 | 1 | 25,000 | 25,000 | None |

Gambar 9. Contoh Hasil *Regular Expressions*

3.3.6. Integrasi Streamlit

Pada tahap ini, seluruh komponen dan teknik yang sudah dikembangkan diintegrasikan ke dalam *website* menggunakan Streamlit. Tujuan dari tahap ini adalah untuk memudahkan proses pengujian sistem, baik dalam tahap *alpha* maupun *beta*, sehingga pengguna dapat memahami alur dan fungsionalitas sistem secara intuitif dengan tampilan hasil seperti Gambar 10 berikut.

Deteksi & OCR Struk Belanja

Upload gambar

Drag and drop file here
Limit 200MB per file • JPG, PNG, JPEG

Browse files

Gambar 10. Tampilan Streamlit

Proses pengembangan Streamlit dikemas dalam dua format yang disesuaikan dengan tujuan pengujian. Untuk pengujian *alpha*, seluruh proses ditampilkan secara eksplisit dalam antarmuka mulai dari *upload* gambar, hasil deteksi YOLO, hasil ekstraksi Tesseract, serta hasil dari *regular expressions* berupa data terstruktur. Tampilan pengujian *alpha* dapat dilihat pada Gambar 11. Adapun untuk pengujian *beta*, format ini dirancang lebih sederhana dan berfokus pada pengalaman *end-user*. Dalam antarmuka, Streamlit hanya menampilkan proses *upload* gambar dan hasil akhir berupa data terstruktur seperti Gambar 12.

Hasil OCR dan Ekstraksi Data

IDM KTG PLSTK 1W BSR 1 300 300
 DAIA DET+SFT VLT 800 1 17900 17, 900
 VC SAYAP MAS UTA : (1, 900)
 SELECTION KAPAS 50GR 1 11600 11, 600
 TOTOLE KLD JMUR 80G 1 13300 13, 300
 DERMA PTCH PLS DAY6S 1 19200 19, 200
 VITACIMIN STRIP 2'S 1 2300 2,300 —
 US SKT. BAJU 20930 1 25000 25, 000

| | Product | qty | Harga per Unit | Total Price | Discount |
|---|----------------------|------|----------------|-------------|----------|
| 0 | IDM KTG PLSTK 1W BSR | 1 | 300 | 300 | None |
| 1 | DAIA DET+SFT VLT 800 | 1 | 17,900 | 17,900 | None |
| 2 | VC SAYAP MAS UTA | None | None | None | 1900 |
| 3 | SELECTION KAPAS 50GR | 1 | 11,600 | 11,600 | None |
| 4 | TOTOLE KLD JMUR 80G | 1 | 13,300 | 13,300 | None |
| 5 | DERMA PTCH PLS DAY6S | 1 | 19,200 | 19,200 | None |
| 6 | VITACIMIN STRIP 2'S | 1 | 2,300 | 2,300 | None |
| 7 | US SKT BAJU 20930 | 1 | 25,000 | 25,000 | None |

Gambar 11. Tampilan Streamlit untuk Pengujian *Alpha*

Sistem Ekstraksi Struk Belanja



Hasil Ekstraksi Data

| | Product | Quantity | Harga per Unit | Total Price | Discount |
|---|----------------------|----------|----------------|-------------|----------|
| 0 | IDM KTG PLSTK 1W BSR | 1 | 300 | 300 | None |
| 1 | DAIA DET+SFT VLT 800 | 1 | 17,900 | 17,900 | None |
| 2 | VC SAYAP MAS UTA | None | None | None | 1900 |
| 3 | SELECTION KAPAS 50GR | 1 | 11,600 | 11,600 | None |
| 4 | TOTOLE KLD JMUR 80G | 1 | 13,300 | 13,300 | None |
| 5 | DERMA PTCH PLS DAY6S | 1 | 19,200 | 19,200 | None |
| 6 | VITACIMIN STRIP 2'S | 1 | 2,300 | 2,300 | None |
| 7 | US SKT BAJU 20930 | 1 | 25,000 | 25,000 | None |

Gambar 12. Tampilan Streamlit untuk Pengujian *Beta*

3.4. Pengujian

Pada tahapan ini, dilakukan pengujian untuk mengevaluasi fungsionalitas sistem serta memastikan bahwa sistem bekerja sesuai dengan tujuan yang telah ditetapkan. Adapun tahap pengujian dibagi menjadi 2 tahap, yaitu *alpha testing* dan *beta testing*. Dalam *alpha testing*, pengujian dilakukan menggunakan *black box testing* yang berfokus pada fungsionalitas dengan hasil seperti Tabel 3 berikut.

Tabel 3. Hasil Pengujian *Black Box Testing*

| No. | Kegiatan Pengujian | Test Case | Output yang Diharapkan | Hasil |
|-----|--|---|--|--------|
| 1 | Menerima gambar struk belanja | Mengunggah gambar struk belanja | Muncul nama file gambar yang diunggah | Sesuai |
| 2 | Validasi format file input saat unggah | Klik tombol unggah | Hanya file gambar .png, .jpg, .jpeg yang dapat dipilih | Sesuai |
| 3 | Mendeteksi area pada gambar menggunakan YOLO | Mengunggah gambar struk belanja | Muncul gambar struk belanja dengan <i>bounding box</i> pada area yang akan diekstrak | Sesuai |
| | | Mengunggah gambar struk dengan format berbeda dan gambar selain struk belanja | Muncul gambar yang diunggah namun <i>bounding box</i> tidak muncul | Sesuai |
| 4 | Mengekstrak teks dari gambar | Mengunggah gambar struk belanja | Muncul teks yang berhasil diekstrak pada area <i>bounding box</i> | Sesuai |
| | | Mengunggah gambar buram, terlipat, atau terdapat <i>noise</i> | Muncul teks dalam sistem namun hanya sebagian yang sesuai dengan gambar | Sesuai |

DOI: <https://doi.org/10.31004/riggs.v4i2.1731>

Lisensi: Creative Commons Attribution 4.0 International (CC BY 4.0)

| | | | | |
|---|---------------------------------------|---------------------------------|--|--------|
| 5 | Parsing teks menjadi data terstruktur | Mengunggah gambar struk belanja | Muncul data seperti nama produk, harga satuan, total, dan diskon dalam bentuk tabel. | Sesuai |
|---|---------------------------------------|---------------------------------|--|--------|

Berdasarkan hasil diatas, sistem berhasil berjalan sesuai dengan fungsionalitasnya masing-masing dan mampu menjalankan proses ekstraksi secara *end-to-end* dari *input* gambar hingga *output* berupa data terstruktur. Adapun untuk pengujian *beta* dilakukan dengan metode *User Acceptance Testing* (UAT) kepada responden yang pernah melakukan pencatatan keuangan baik manual maupun digital dari struk belanja. Pengujian dilakukan dengan menggunakan kuesioner berbasis skala likert, yang diawali oleh demonstrasi sistem kepada 37 responden melalui Google Meet atau rekaman video. Berikut merupakan hasil dari kuesioner *user acceptance testing* dalam tabel 4.

Tabel 4. Hasil Kuesioner UAT

| No | Pernyataan | Tanggapan | | | | | Rata-rata |
|-----------------------|---|---------------------|--------------|--------|--------|---------------|-----------|
| | | Sangat Tidak Setuju | Tidak Setuju | Netral | Setuju | Sangat Setuju | |
| 1 | Sistem ini membantu saya dalam pencatatan keuangan | 0 | 0 | 0 | 15 | 22 | 4,59 |
| 2 | Saya merasa dengan sistem ini waktu yang dibutuhkan untuk mencatat keuangan dapat menjadi lebih singkat | 0 | 0 | 0 | 17 | 19 | 4,48 |
| 3 | Saya merasa dengan sistem ini dapat mengurangi beban pencatatan manual | 0 | 0 | 1 | 14 | 22 | 4,56 |
| 4 | Saya merasa dengan sistem ini dapat mengurangi kemalasan dalam mencatat keuangan | 0 | 2 | 5 | 10 | 20 | 4,29 |
| 5 | Sistem ini mudah digunakan | 0 | 1 | 2 | 11 | 23 | 4,51 |
| 6 | Pencatatan keuangan khususnya dari struk belanja menjadi lebih mudah dengan sistem ini | 0 | 0 | 1 | 17 | 19 | 4,48 |
| 7 | Saya merasa sistem ini bermanfaat dan positif | 0 | 0 | 0 | 13 | 24 | 4,64 |
| 8 | Saya mempertimbangkan untuk menggunakan sistem ini dalam pencatatan keuangan | 0 | 2 | 3 | 18 | 14 | 4,18 |
| Rata-rata keseluruhan | | | | | | | 4,47 |

Berdasarkan tabel tersebut, pengujian menunjukkan bahwa rata-rata nilai dari seluruh pernyataan mencapai 4,47 dari skala 5, yang menggambarkan bahwa sistem ekstraksi ini diterima dengan baik oleh pengguna dan dinilai mampu membantu dalam proses pencatatan keuangan. Sistem dinilai membantu karena waktu yang dibutuhkan menjadi lebih singkat, mengurangi beban input manual, mengurangi kemalasan, dan mudah digunakan. Temuan ini sejalan dengan penelitian sebelumnya yang menunjukkan bahwa sistem serupa yang dilengkapi fitur OCR memperoleh skor *User Acceptance Testing* sebesar 81,56%, di mana salah satu indikatornya adalah kemudahan dalam pencatatan transaksi melalui ekstraksi teks [6]. Selain itu, temuan ini juga didukung oleh penelitian lain yang mengatakan bahwa integrasi *Optical Character Recognition* (OCR) membantu dalam sistem pencatatan keuangan pribadi karena dapat mengurangi potensi *human error* dalam proses *input* dan menjadi lebih akurat [15].

4. Kesimpulan

Dari hasil temuan dan pembahasan diatas, maka dapat disimpulkan bahwa solusi berupa pembangunan sistem ekstraksi belanja yang dibangun dengan menerapkan YOLO, Tesseract, dan *Regular Expressions* mampu membantu pengguna dalam proses pencatatan keuangan. Evaluasi model YOLO menunjukkan performa yang cukup baik berdasarkan beberapa matrik evaluasi. Selain itu, proses ekstraksi dengan Tesseract dan pengolahan menggunakan *Regular Expressions* berjalan dengan baik dan sesuai tujuan. Hasil pengujian *alpha* menunjukkan bahwa seluruh fungsi sistem berjalan sesuai output yang diharapkan, sementara hasil pengujian *beta* membuktikan bahwa sistem dinilai bermanfaat dan mendukung kebutuhan pengguna dalam proses pencatatan keuangan.

Referensi

- [1] Y. Widiyawati, C. D. S. Ningsih, F. Lestari, and G. Pramita, "ANALISIS PENGARUH BELANJA ONLINE TERHADAP PERILAKU PERJALANAN BELANJA DIMASA PANDEMI COVID-19," *Journal of Infrastructural in Civil Engineering (JICE)*, vol. 03, p. 25, Jul. 2022.
- [2] U. Sugiarti, "77% Masyarakat Indonesia Lebih Suka Belanja di Minimarket.", 2024. [Online]. Tersedia: <https://goodstats.id/article/77-masyarakat-indonesia-menyukai-belanja-di-minimarket-CRkaQ>.
- [3] C. Sayallar, A. Sayar, and N. Babalik, "An OCR Engine for Printed Receipt Images using Deep Learning Techniques," *International Journal of Advanced Computer Science and Applications*, vol. 14, no. 2, 2023, doi: 10.14569/IJACSA.2023.0140295.
- [4] A. Rosidi and A. Afriyudi, "Aplikasi Pencatatan Keuangan Pribadi Berbasis Web Mobile," *Jurnal Teknologi Informatika dan Komputer*, vol. 9, no. 1, pp. 100–113, Mar. 2023, doi: 10.37012/jtik.v9i1.1447.
- [5] M. C. Dinisari, "Lebih dari 40 Persen Orang Tidak Mencatat Cash Flow Karena Malas," <https://entrepreneur.bisnis.com/read/20200625/52/1257724/lebih-dari-40-persen-orang-tidak-mencatat-cash-flow-karena-malas>.
- [6] R. Sari, I. M. Adi, and A. Hidayati, "Personal Track Your Cash: Prototipe Aplikasi Pembacaan Struk0020Belanja Menggunakan OCR dan Google Vision," *SNIV: SEMINAR NASIONAL INOVASI VOKASI*, vol. 2, pp. 565–573, 2023.

DOI: <https://doi.org/10.31004/riggs.v4i2.1731>

Lisensi: Creative Commons Attribution 4.0 International (CC BY 4.0)

- [7] A. ALazzawi, Q. M. Yas, and B. Rahmatullah, "A Comprehensive Review of Software Development Life Cycle methodologies: Pros, Cons, and Future Directions ," *Iraqi Journal for Computer Science and Mathematics*, vol. 4, no. 4, pp. 173–190, 2023.
- [8] H. Yakub, B. Daniawan, A. Wijaya, and L. Damayanti, "Sistem Informasi E-Commerce Berbasis Website Dengan Metode Pengujian User Acceptance Testing," *JSITIK: Jurnal Sistem Informasi dan Teknologi Informasi Komputer*, vol. 2, no. 2, pp. 113–127, Apr. 2024, doi: 10.53624/jsitik.v2i2.362.
- [9] I. Wahyudi, Fahrullah, F. Alameka, and Haerullah, "ANALISIS BLACKBOX TESTING DAN USER ACCEPTANCE TESTING TERHADAP SISTEM INFORMASI SOLUSIMEDSOSKU," *Jurnal Teknosains Kodepena*, vol. 4, no. 1, pp. 1–9, 2023.
- [10] ExpressExpense, "Free Receipt Images – OCR / Machine Learning Dataset", 2020. [Online]. Tersedia: <https://expressexpense.com/blog/free-receipt-images-ocr-machine-learning-dataset/>.
- [11] S. Park *et al.*, "CORD: A Consolidated Receipt Dataset for Post-OCR Parsing", 2019. [Online]. Tersedia: <https://github.com/clovaai/cord>.
- [12] R. G. Guntara, "Pemanfaatan Google Colab Untuk Aplikasi Pendeteksian Masker Wajah Menggunakan Algoritma Deep Learning YOLOv7," *Jurnal Teknologi Dan Sistem Informasi Bisnis*, vol. 5, no. 1, pp. 55–60, Feb. 2023, doi: 10.47233/jteksis.v5i1.750.
- [13] C. A-Sawaareekun and R. Lipikorn, "Menu item extraction from Thai receipt images using deep learning and template-based information extraction," in *Proceeding of the 2024 6th International Conference on Information Technology and Computer Communications*, New York, NY, USA: ACM, Oct. 2024, pp. 107–113. doi: 10.1145/3704391.3704407.
- [14] A. Kaderabek, "Exploring Optical Character Recognition (OCR) as a Method of Capturing Data from Food-Purchase Receipts," *Survey Methods: Insights from the Field*, vol. 1, no. 3, 2023.
- [15] M. Nazeem, R. Anitha, Navaneeth S, and R. R. Rajeev, "Open-Source OCR Libraries: A Comprehensive Study for Low Resource Language," *Proceedings of the 21st International Conference on Natural Language Processing (ICON)*, pp. 416–421, 2024.