

Development Of Optical Character Recognition Technology In Flutter For Text Detection In Images

[Pengembangan Teknologi Optical Character Recognition Di Flutter Berupa Deteksi Teks Pada Gambar]

Shierly Mayco Angela¹⁾, Ade Eviyanti^{*2)}

¹⁾ Program Studi Teknik Informatika, Universitas Muhammadiyah Sidoarjo, Indonesia

²⁾ Program Studi Teknik Informatika, Universitas Muhammadiyah Sidoarjo, Indonesia

*Email Penulis Korespondensi: adeeviyanti@umsida.ac.id

Abstract. *OCR functions to extract text from images to expedite data input. The method employed in this research utilizes the Rapid Application Development approach. The objective of this application is to create a mobile application capable of detecting text in images with high accuracy and swift responsiveness. The study involved 10 respondents who were presented with several questions, resulting in an accuracy rate of 88%. Consequently, the positive outcome from the Savert application indicates its reliability in converting written text into digital text.*

Keywords - OCR, Flutter, Detection, Text, Image

Abstrak. *Fungsi OCR bertujuan untuk mengekstrak teks dari gambar. Metode yang digunakan dalam penelitian ini menggunakan pendekatan Rapid Application Development. Tujuan dari aplikasi ini adalah menciptakan aplikasi android yang mampu mendeteksi teks dari gambar dengan akurasi tinggi dan respons yang cepat. Penelitian ini melibatkan 10 responden yang diberikan beberapa pertanyaan, yang menghasilkan tingkat akurasi sebesar 88%. Oleh karena itu, aplikasi Savert dapat dinyatakan berhasil untuk mengubah teks gambar menjadi teks digital.*

Kata Kunci - OCR, Flutter, Deteksi, Teks, Gambar

I. PENDAHULUAN

Pengembangan teknologi optical character recognition (OCR) telah menjadi salah satu tonggak penting dalam perkembangan dunia digital. OCR berfungsi mengekstraksi teks dari gambar guna mempercepat input data. Dengan menggunakan teknologi Optical Character Recognition, teks dalam gambar dapat diambil secara otomatis tanpa perlu adanya intervensi manual dari pengguna [1]. Pada masa sekarang, penggunaan OCR telah disederhanakan berkat adanya bantuan dari library Tesseract OCR, yang juga menunjukkan tingkat akurasi yang tinggi [2]. Oleh karena itu, pengembangan teknologi OCR yang dapat terintegrasi dengan framework mobile seperti Flutter semakin penting dalam aplikasi mobile karena memungkinkan pengguna untuk mengekstrak teks dari gambar yang diambil melalui kamera ponsel dan galeri ponsel, mengubahnya menjadi text digital yang dapat diolah.

Perkembangan tulisan telah mengalami kemajuan pesat. Tulisan merupakan gabungan karakter huruf dan angka dalam bahasa tertentu. Setiap karakter memiliki identitas uniknya sendiri, yang membedakan satu dengan yang lainnya, sehingga membentuk rangkaian tulisan yang memiliki kemampuan untuk menyampaikan informasi [3]. Seorang peneliti menciptakan perangkat canggih yang mampu membaca teks dari gambar dengan tingkat akurasi yang sangat tinggi [4]. Perangkat ini dapat digunakan untuk mengekstrak teks dari berbagai jenis font, baik yang berlisensi gratis maupun berbayar.

Deteksi teks pada gambar adalah salah satu kasus penggunaan yang penting dalam teknologi OCR. Dalam berbagai situasi, pengguna seringkali perlu mengambil data teks dari gambar untuk analisis lebih lanjut atau pengolahan data [5]. Contohnya adalah komunitas penggemar komik kesulitan dalam mencari referensi komik digital sehingga membutuhkan sebuah sistem untuk pencarian informasi ekstraksi teks [6]. Pengembangan teknologi OCR di Flutter yang mampu mendeteksi teks pada gambar untuk menciptakan peluang inovasi keperluan aplikasi memberikan fitur cerdas & efisien. Selain itu, dengan kemampuan Flutter yang fleksibel dan dapat diintegrasikan dengan berbagai plugin, pengembang dapat menggabungkan teknologi OCR dengan elemen-elemen lain dalam aplikasi, menciptakan pengalaman pengguna yang menyeluruh dan optimal [7].

Penelitian oleh Izzah yang berjudul Sistem Deteksi Kemiripan Teks Pada Berita Berbahasa Indonesia Menggunakan Algoritma Ratcliff/Obershelp. Plagiarisme pada teks berita jadi masalah serius dalam penelitian, akibat kurangnya proses stemming saat menggunakan algoritma Ratcliff/Obershelp, menyebabkan penurunan akurasi hasil [8]. Untuk tingkatkan akurasi, perangkat lunak terapkan algoritma Nazief dan Adriani pada tahap pra-pemrosesan. Penelitian berikut bertujuan mencegah plagiarisme dalam karya tulis yang sering ditemukan di berbagai media, termasuk internet [9].

Lalu penelitian oleh Mamuriyah yang berjudul Perancangan dan Pembuatan Alat untuk Mendeteksi Teks Hangul dan Inggris pada Menu Makanan Menggunakan metode OCR. Penelitian menyoroti kesulitan wisatawan Muslim dalam memahami komposisi makanan dalam bahasa Korea. Solusinya adalah menggunakan teknologi Pengolahan Citra dan OCR untuk membantu mereka memilih makanan halal dengan tepat [10]. Dalam penelitian berikut, pendekatan lokalisasi teks juga digunakan untuk mengidentifikasi teks pada gambar menu makanan guna diolah oleh sistem [4].

Ada juga penelitian oleh Filsa yang berjudul Kinerja Algoritma Canny Untuk Mendeteksi Tepi Dalam Mengidentifikasi Tulisan Pada Citra Digital Meme. Penelitian ini menyoroti signifikansi citra dalam konteks multimedia sebagai media visual yang penting. Pengolahan citra digital melibatkan penerapan teknik Analisis Objek, termasuk penggunaan algoritma deteksi tepi Canny yang terbukti handal dalam mengidentifikasi dan memetakan karakter [11]. Tujuan penelitian adalah meningkatkan akurasi deteksi teks pada OCR dengan memanfaatkan hasil algoritma deteksi tepi Canny sebagai wilayah potensial teks [12].

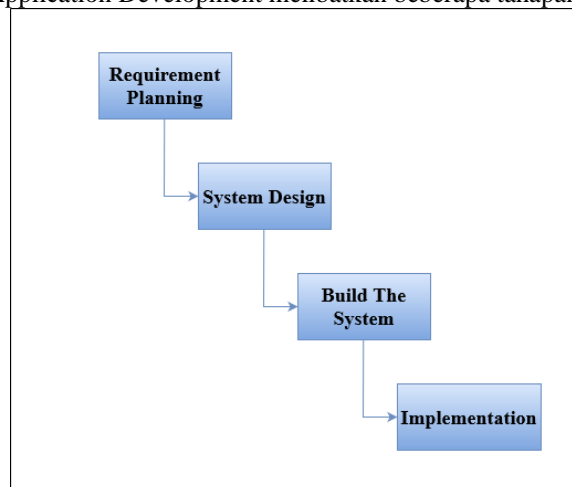
Sedangkan penelitian oleh Dermawan yang berjudul Peringkasan Teks Untuk Deteksi Kejadian Pada Dokumen Twitter Berbahasa Indonesia Dengan Metode Affinity Propagation. Penelitian berikut berfokus pada popularitas Twitter sebagai platform media sosial dengan lebih dari 313 juta pengguna dan 500 juta tweet harian, dengan tujuan memberikan ringkasan cepat tentang peristiwa penting di Twitter [13]. Melalui pengklasteran, didapatkan kelompok yang mewakili peristiwa pada waktu dan tempat tertentu [14].

Yang terakhir penelitian oleh Aprilino yang berjudul Implementasi Algoritma Yolo dan Tesseract OCR pada Sistem Deteksi Plat Nomor Otomatis. Penelitian berikut bertujuan membangun sistem untuk mengidentifikasi dan ekstraksi informasi dari plat nomor kendaraan melalui gambar [15]. Metodenya melibatkan pengembangan sistem deteksi berbasis YOLO guna meningkatkan efisiensi, akurasi, dan efektivitas dalam pengambilan data plat nomor kendaraan [2].

Berdasarkan penelitian sebelumnya, ada peluang untuk mengembangkan penelitian dalam deteksi teks dalam aplikasi berbasis gambar untuk perangkat seluler. Hal berikut disebabkan oleh kurangnya penggunaan kerangka kerja Flutter dalam penelitian sebelumnya. Dalam penelitian yang berjudul "Pengembangan Teknologi Optical Character Recognition dalam Flutter untuk Deteksi Teks dalam Gambar," teknologi OCR canggih digabungkan dengan kerangka kerja Flutter yang populer, menghasilkan aplikasi seluler yang dapat mendeteksi teks dalam gambar dengan akurasi tinggi dan respons cepat [16]. Flutter, yang dikembangkan oleh Google dan bersifat sumber terbuka, lebih mudah diakses untuk pengembangan multi-platform, termasuk Android, iOS, dan situs web. Bahasa pemrograman yang digunakan adalah Dart. Diharapkan bahwa temuan dari penelitian memberikan kontribusi positif bagi dunia teknologi dan memberikan solusi efisien untuk kebutuhan yang semakin beragam dan banyak dari pengguna aplikasi seluler [17].

II. METODE

Dalam penelitian ini, teknologi Optical Character Recognition (OCR) dikembangkan dalam platform Flutter dengan fokus pada deteksi teks pada gambar. Pendekatan penelitian yang diadopsi adalah Rapid Application Development (RAD). Strategi yang fokus pada pembangunan yang cepat dan terfokus [18]. Batasan utama dalam model pengembangan adalah keterbatasan waktu yang singkat. Agar tujuan tercapai, pengembangan harus melewati semua tahap dari awal hingga akhir, memastikan kelancaran sesuai rencana dan hasil optimal. Pengembangan sistem menggunakan metode Rapid Application Development melibatkan beberapa tahapan yang perlu dijalani [19];

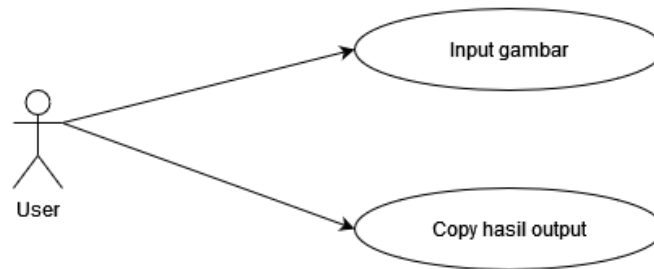


Gambar 1. Metode Rapid Application Development

Requirement Planning (Perencanaan Kebutuhan)

Tahapan Perencanaan Kebutuhan dalam pengembangan teknologi Optical Character Recognition (OCR) menggunakan Flutter untuk mendeteksi teks pada gambar, difokuskan pada mengidentifikasi masalah yang ada dan mengumpulkan data dari pengguna atau pemangku kepentingan (stakeholder) [20]. Dalam pengembangan aplikasi, perancangan sistem dilakukan dengan membuat flowchart yang menggambarkan alur dari aplikasi secara grafik. Flowchart merupakan representasi visual dari prosedur dalam program tertentu. Flowchart membantu analis dan pengembang dalam memecah masalah menjadi segmen-segmen yang lebih kecil, mengevaluasi pilihan operasional yang berbeda, serta merencanakan pengembangan aplikasi secara lebih terstruktur dan efektif.

2.1 Analisa Kebutuhan



Gambar 2. UML Diagram

Pada gambar 2 menggambarkan kebutuhan user yang dapat menginput gambar dan mengcopy hasil output yang dihasilkan dari aplikasi.

2.2 Pengumpulan Data

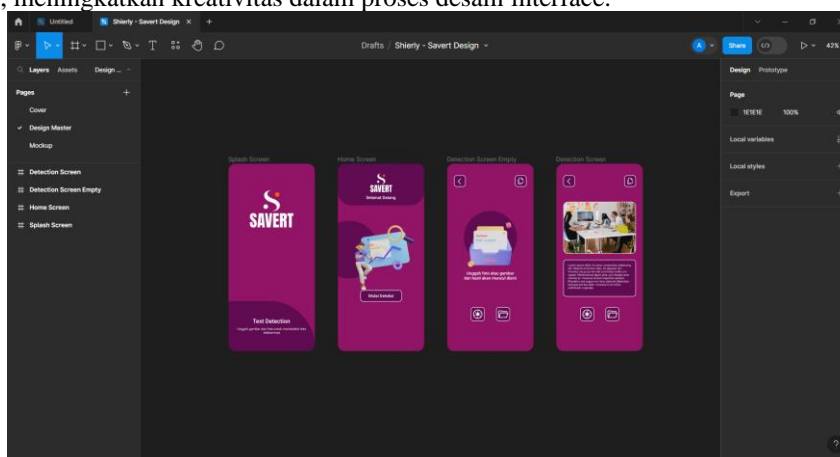
Pengumpulan data dilakukan agar memudahkan proses pembuatan sistem dari aplikasi.

1. Kuisisioner

Metode pengumpulan data dengan kuisisioner adalah suatu pendekatan yang menggunakan formulir atau daftar pertanyaan tertulis yang diberikan kepada responden untuk mengumpulkan informasi atau tanggapan terkait dengan topik penelitian atau survei.

System Design (Desain Sistem)

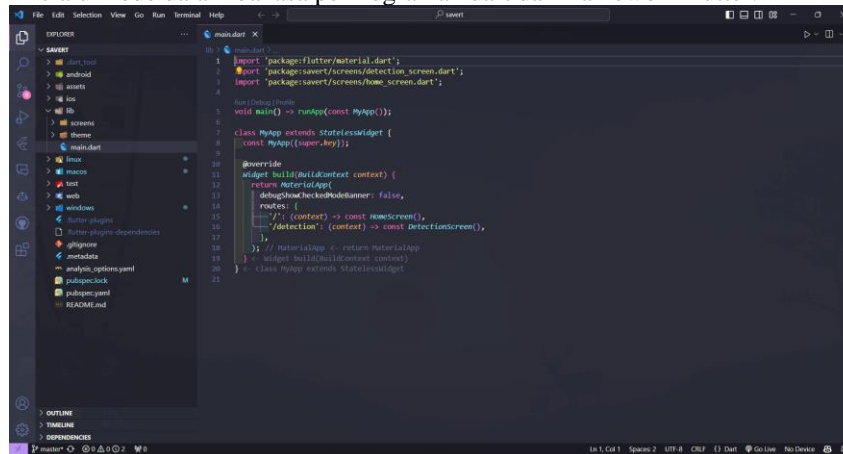
Pada tahapan system design, dilakukan proses desain dan iterasi berulang untuk memastikan kesesuaian desain dengan kebutuhan yang telah diidentifikasi pada tahapan sebelumnya. Luaran dari tahap desain berupa ER Diagram, UML, dan Prototype [21]. Dalam proses desain interface aplikasi berikut digunakan tool bernama Figma. Figma adalah perangkat lunak yang mempermudah desainer dalam membuat antarmuka menarik, efisien, dan menarik bagi pengguna aplikasi, meningkatkan kreativitas dalam proses desain interface.



Gambar 3. Desain Aplikasi Figma

Build The System (Membangun Sistem)

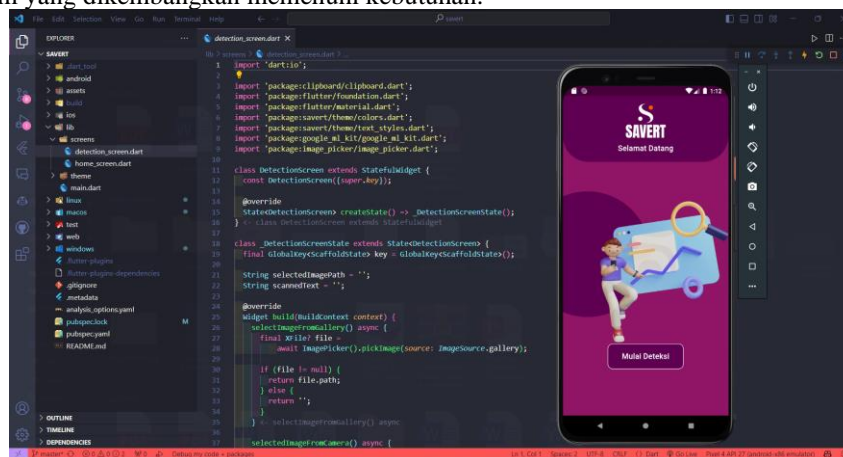
Tahap selanjutnya setelah melakukan design UI yaitu tahap build the system. Pada tahap merancang struktur dari ER Diagram, UML, dan mockup diimplementasikan ke sebuah aplikasi yang berjalan. Berhubung aplikasi deteksi teks pada gambar tidak menyimpan data, jadi tidak membutuhkan ER diagram atau basis data. UML dan mockup diimplementasikan melalui kode dalam bahasa pemrograman dart dan framework flutter.



Gambar 4. Proses Perancangan Aplikasi

Implementation (Implementasi)

Pada tahap implementasi dalam pengembangan teknologi Optical Character Recognition (OCR) menggunakan Flutter untuk mendeteksi teks pada gambar, dilakukan proses pengujian. Teknologi flutter sendiri merupakan framework yang di kembangkan oleh google untuk membuat aplikasi multi platform. Setiap unit program digabungkan menjadi sistem dan diuji dengan Black Box Testing. Teknik Black Box Testing bertujuan memverifikasi apakah sistem sesuai dengan spesifikasi yang telah ditetapkan atau menemukan kesalahan (bug). Melalui Black Box Testing, sistem diuji untuk memastikan komponen-komponennya berfungsi sesuai kebutuhan yang telah diidentifikasi sebelumnya. Kemudian pengujian untuk user menggunakan UAT (User Acceptance Testing). UAT merupakan tipe pengujian yang dilaksanakan oleh pihak pengguna akhir atau pemangku kepentingan utama untuk memastikan bahwa aplikasi atau sistem yang dikembangkan memenuhi kebutuhan.



Gambar 5. Uji Testing Aplikasi

III. HASIL DAN PEMBAHASAN

Hasil Aplikasi

Dalam langkah berikut terdapat penjelasan mengenai metode pelaksanaan pengujian sistem untuk aplikasi deteksi teks pada gambar. Pengujian tersebut mencakup pengujian blackbox testing untuk sistem dan pengujian UAT untuk pengguna.

Splash Screen

Halaman splash screen adalah tampilan pertama yang muncul ketika aplikasi dijalankan, setelah beberapa detik splash screen berpindah ke home screen.



Gambar 6. Tampilan Splash Screen

Home Screen

Halaman home screen berisi ucapan selamat datang dengan adanya button bertuliskan mulai deteksi, jika button ditekan maka menavigasikan ke menu image convert.



Gambar 7. Home Screen

Main Feature Page

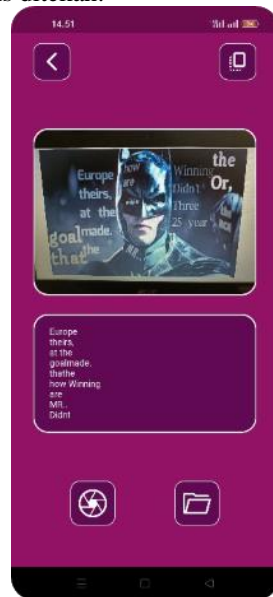
Pada halaman fitur utama berisi beberapa fitur seperti tombol button kembali digunakan untuk kembali ke home screen, ada button dengan icon camera digunakan untuk mengambil gambar secara langsung dari camera ponsel, lalu ada button icon folder yang digunakan mengambil gambar dari galeri ponsel, dan terakhir ada button dengan icon copy yang digunakan untuk meng-copy hasil dari scan teks.



Gambar 8. Main Feature Page

Process Output

Halaman output proses adalah halaman yang menampilkan gambar yang dipilih dan output scan teks. Lalu gambar dibawahnya adalah contoh jika fitur copy teks ditekan.



Gambar 9. Process Output



Gambar 10. Text Output Copy Feature

Implementasi

Tahap implementasi adalah tahap pengujian aplikasi, disini menggunakan dua pengujian yang pertama menggunakan blackbox testing dan yang kedua menggunakan pengujian UAT yang diambil dari polling kuisioner dengan total 10 orang.

Tabel 1. Pengujian Blackbox

Pengujian			
No	Nama Pengujian	Hasil Yang Diharapkan	Hasil Pengujian
1.	Splash Screen	Menampilkan Tampilan Awal (Splash Screen) yang beralih secara otomatis ke Layar Utama (Home Screen) setelah beberapa detik.	Berhasil
2.	Klik button yang tersedia pada home screen	Setelah button diklik, berpindah ke menu fitur utama.	Berhasil
3.	Tombol back button	Menavigasikan kembali ke home screen.	Berhasil
4.	Tombol icon camera	Dapat membuka camera ponsel dan mengambil gambar secara langsung.	Berhasil
5.	Tombol icon folder	Dapat membuka galeri ponsel dan memilih gambar.	Berhasil
6.	Proses scan gambar	Menampilkan hasil output teks dari gambar.	Berhasil
7.	Tombol icon copy	Meng-copy teks dari hasil output.	Berhasil

Tabel 2. Hasil Pengujian User Acceptance Testing

Pertanyaan	Skor				
	Sangat Tidak Setuju	Tidak Setuju	Netral	Setuju	Sangat Setuju
Desain UI menarik			2	1	7
Aplikasi mudah dipahami			1	6	3
Aplikasi tidak terlalu berat			3	5	2
Aplikasi dapat dijalankan secara offline					10

Semua fitur berjalan dengan baik	6	4
----------------------------------	---	---

Menghitung Presentase UAT

Tabel 3. Range Persetujuan

Kategori	Range
Sangat Tidak Setuju (STS)	0% - 20%
Tidak Setuju (TS)	21% - 40%
Netral (N)	41% - 60%
Setuju (S)	61% - 80%
Sangat Setuju (SS)	81% - 100%

$$\text{Presentase UAT} = \frac{\text{Total Skor}}{M} \times 100\%$$

$$\text{Presentase UAT} = \frac{220}{250} \times 100\% = 88\%$$

Range persetujuan dengan hasil pengujian termasuk Sangat Setuju (SS) karena berada pada range 81% - 100%.

VII. SIMPULAN

Setelah melalui serangkaian pengujian dapat disimpulkan aplikasi deteksi teks pada gambar bisa mempermudah untuk memindahkan ke teks digital. Hasil yang didapat dari pengujian UAT dapat memenuhi sasaran yaitu mendapatkan hasil dalam presentase 88%.

Pentingnya pengujian fitur aplikasi menciptakan landasan yang kuat kepada para pengguna. Dengan hasil yang baik dari aplikasi Savert telah dapat diandalkan dalam mengubah teks tulis menjadi teks digital.

REFERENSI

- [1] I. Wijaya and C. Lubis, "Pengimplementasian Ocr Menggunakan Cnn Untuk Ekstraksi Teks Pada Gambar," *J. Ilmu Komput. dan Sist. Inf.*, vol. 10, no. 1, 2022, doi: 10.24912/jiksi.v10i1.17836.
- [2] A. Aprilino and I. H. Al Amin, "Implementasi Algoritma Yolo dan Tesseract OCR pada Sistem Deteksi Plat Nomor Otomatis," *J. TEKNOINFO*, vol. 16, no. 1, pp. 54–59, 2022.
- [3] F. Maedjaja and Efraim, "Sistem deteksi teks pada cover buku dengan pendekatan karakter teks," *Infact Ukrim*, vol. 6, no. 2, 2021.
- [4] N. Mamuriyah and J. Jacky, "Perancangan dan Pembuatan Alat untuk Mendeteksi Teks Hangul dan Inggris pada Menu Makanan Menggunakan metode OCR (Optical Character Recognition)," *Telcomatics*, vol. 6, no. 1, pp. 1–10, 2021, doi: 10.37253/telcomatics.v6i1.5054.
- [5] M. Rizal Toha and A. Triayudi, "PENERAPAN MEMBACA TULISAN DI DALAM GAMBAR MENGGUNAKAN METODE OCR BERBASIS WEBSITE (STUDI KASUS: e-KTP)," *JST (Jurnal Sains dan Teknol.)*, vol. 11, no. 1, pp. 175–183, 2022, doi: 10.23887/jstundiksha.v11i1.42279.
- [6] A. Kumar Siliwangi and D. Prabowo, "Pencarian Informasi Berbasis Teks dalam Komik Digital Menggunakan OCR," *J. Sains, Bisnis dan Teknol.*, vol. 8, no. 2, pp. 1886–1894, 2022.
- [7] M. N. Pangesti and V. Frendiana, "Rancang Bangun Tampilan Aplikasi My Aquaponic Menggunakan Framework Flutter," vol. 1, no. 1, pp. 348–357, 2022.
- [8] Y. C. Sipayung, "Identifikasi Tingkat Kemiripan Dokumen Teks Menggunakan Fungsi Hash Pada Algoritma Winnowing," *Univ. Sumatera Utara*, vol. 1, no. 3, pp. 82–91, 2021.
- [9] Hajriansyah, "Identifikasi Jenis Rempah-Rempah Menggunakan Metode CNN Berbasis Android," *J. Ris. Sist. Inf. Dan Tek. Inform.*, vol. 8, no. 1, pp. 223–232, 2023.
- [10] W. Saputro and S. Amelia, "Implementasi Pendataan Warga RW 007 Penggilingan Jakarta Timur Menggunakan Metode OCR Tesseract," *J. Tek. Elektro dan Komputasi*, vol. 5, pp. 130–143, 2023.
- [11] A. Munandar, M. H. Santoso, and S. Sulistiyasni, "Jurnal Media Pratama Jurnal Media Pratama," vol. 15, no. 1, pp. 43–61, 2021.
- [12] N. Fitriani and U. Sholihah, "Sistem Rekomendasi Toko Servis Komputer di Kota Sampit Menggunakan Framework Flutter," vol. 2, no. 2, pp. 33–39, 2023.
- [13] T. P. Putri and Febriani, "Clustering Data Cuti Sakit Menggunakan Algoritma Affinity Propagation (Studi Kasus: Perusahaan Telekomunikasi Di Jakarta)," *J. Ilm. Teknol. dan Rekayasa*, vol. 27, no. 1, pp. 69–84, 2022, doi: 10.35760/tr.2022.v27i1.5823.
- [14] N. Izzah, N. Yusliani, and D. Roodiah, "Sistem Deteksi Kemiripan Teks Pada Berita Berbahasa Indonesia Menggunakan algoritma Ratcliff/Obershelp," *J. Linguist. Komputasional*, vol. 5, no. 1, p. 1, 2022, doi: 10.26418/jlk.v5i1.65.
- [15] ILLMAWATI REEZKY and HUSTINAWATI, "YOLO v5 untuk Deteksi Nomor Kendaraan di DKI Jakarta YOLO V5 for Vehicle Plate

- Detection in DKI Jakarta,” *J. Ilmu Komput. Agri-Informatika*, vol. 10, no. 1, pp. 32–43, 2022, [Online]. Available: www.kaggle.com
- [16] N. Filsa, Widodo, and B. Prasetya Adhi, “Kinerja Algoritma Canny untuk Mendeteksi Tepi dalam Mengidentifikasi Tulisan pada Citra Digital Meme,” *PINTER J. Pendidik. Tek. Inform. dan Komput.*, vol. 3, no. 1, pp. 45–53, 2019, doi: 10.21009/pinter.3.1.8.
- [17] R. Dermawan, F. A. Bachtiar, and P. P. Adikara, “Peringkasan Teks Untuk Deteksi Kejadian Pada Dokumen Twitter Berbahasa Indonesia Dengan Metode Affinity Propagation,” *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 3, no. 3, pp. 2208–2214, 2019.
- [18] P. S. A. P. Wulandari, K. T. Martono, and I. P. Windasari, “Pengembangan Sistem Pendeteksi Gesture Angka pada Tangan secara Realtime Berbasis Android,” *Edu Komputika J.*, vol. 7, no. 1, pp. 1–9, 2020, doi: 10.15294/edukomputika.v7i1.38655.
- [19] F. Putrawansyah, “Application Running Text Information Berbasis Android,” *JATISI (Jurnal Tek. Inform. dan Sist. Informasi)*, vol. 6, no. 1, pp. 116–125, 2019, doi: 10.35957/jatisi.v6i1.161.
- [20] F. N. Sabrina, “Aplikasi Steganografi Pada Media Gambar Menggunakan Algoritma Least Significan Bit,” *J. Tera*, vol. 1, no. 2, pp. 185–201, 2021, [Online]. Available: <http://jurnal.undira.ac.id/index.php/jurnaltera/article/view/55>
- [21] H. Faqih, A. B. Hikmah, and W. Azizah, “Implementasi Metode Rapid Application Development Pada Pengembangan Aplikasi e-Fin Mosque Z,” *Indones. J. Softw. Eng.*, vol. 8, no. 1, pp. 83–91, 2022, doi: 10.31294/ijse.v8i1.13007.

Conflict of Interest Statement:

The author declares that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.