

Koreksi Orientasi Teks pada Dokumen Digital Menggunakan Transformasi *Hough*

Lianny Wydiastuty Kusuma^{1)*}

¹⁾Fakultas Sains dan Teknologi, Universitas Buddhi Dharma
Jl. Imam Bonjol No. 41 Karawaci, Tangerang, Indonesia

¹⁾liannyw_k@yahoo.com

Article history:

Received 21 Nov 2024;
Revised 28 Nov 2024;
Accepted 28 Nov 2024;
Available online 27 Des 2024

Keywords:

Deteksi Tepi *Canny*
Digital Archiving
Koreksi Orientasi Dokumen
Rotasi Gambar
Transformasi *Hough*

Abstrak

Dokumen-dokuman yang diarsipkan sangat penting bagi kebutuhan kantor, rumah, industri maupun museum sejarah. Alat pendukung seperti *scanner* membantu mengubah dokumen fisik menjadi dokumen digital. Dokumen digital dapat digunakan sewaktu-waktu jika dokumen fisik yang asli mengalami kerusakan, bencana ataupun kehilangan. Terkadang teks dalam dokumen digital hasil scan tidak tegak lurus atau memiliki orientasi yang salah seperti miring lebih dari 1 derajat. Agar mudah dibaca oleh orang-orang, maka dalam penelitian ini diusulkan menggunakan metode transformasi *Hough* sebagai solusi pada masalah tersebut. Transformasi *Hough* umumnya digunakan dalam pendeteksian garis, lingkaran atau bentuk lainnya. Langkah yang dilakukan yaitu dimulai dari *grayscale*, *gaussian blur* untuk menghaluskan gambar dan menghilangkan *noise*, pendeteksian tepi dengan *Canny Edge Detection*, deteksi garis dengan transformasi *Hough* dan melakukan rotasi gambar untuk memperbaiki kemiringan teks. Dataset yang dipakai adalah gambar *scan* ketikan teks dengan *noise* dan memiliki variasi sudut kemiringan. Hasil dari 10 sampel gambar menunjukkan bahwa sekitar 70% gambar dapat dirotasi mendekati kondisi tegak lurus dengan nilai sudut rata-rata kurang dari 1 derajat. Kemudian 30% lainnya memiliki nilai sudut rata-rata di antara 1 dan 2 derajat, meskipun begitu hal ini tidak terlalu signifikan dan masih dapat ditolerir bagi indera penglihatan manusia. Waktu pemrosesan tiap gambar rata-rata membutuhkan 0,022 detik. Hasil penelitian dapat digunakan untuk memperbaiki kemiringan pada dokumen digital dan meningkatkan akurasi sistem OCR.

I. PENDAHULUAN

Dokumen-dokumen dalam bentuk digital yang merupakan hasil dari perangkat *scanner*, membantu banyak organisasi ataupun perusahaan menyimpan dokumen-dokumen tersebut dalam bentuk digital. Selama beberapa tahun terakhir, berbagai jenis *scanner* telah berevolusi untuk berbagai pekerjaan [1]. Di antaranya memberikan kualitas pemindaian yang lebih baik, beberapa menggunakan lebih sedikit ruang, dan beberapa dirancang khusus untuk satu jenis media asli. *Flatbed scanner* memiliki permukaan datar tempat dokumen diletakkan untuk dipindai dan umum digunakan di kantor dan rumah untuk berbagai jenis dokumen, termasuk foto dan buku [2]. *Sheet-fed scanner* bekerja dengan menarik dokumen melalui mesin untuk dipindai dan cocok untuk memindai dokumen dalam jumlah besar dengan cepat, seperti lingkungan kantor yang padat [3]. *Handheld scanner* memiliki ukuran kecil yang bisa digerakkan secara manual di atas dokumen, praktis untuk penggunaan portabel, tetapi kurang akurat dibandingkan *scanner* lainnya, dan biasanya digunakan untuk keperluan di bidang forensik [4]. *Portable scanner* biasanya lebih kecil dan mudah dibawa, ideal untuk orang yang sering bepergian, dan beberapa model bahkan memiliki baterai yang bisa terhubung ke perangkat *mobile* [5]. *Overhead scanner* menggunakan kamera yang dipasang di atas untuk memindai dokumen atau objek di bawahnya sehingga cocok untuk buku atau dokumen besar yang sulit dipindai dengan *flatbed scanner* [6]. *Drum scanner* digunakan dalam industri grafis untuk memindai gambar dengan resolusi sangat tinggi, maka dari itu memiliki harga lebih mahal dan jarang digunakan di luar aplikasi profesional [7].

Scanner mempunyai berbagai kegunaan di banyak bidang. Pertama sebagai digitalisasi dokumen dalam pengarsipan untuk mengubah dokumen fisik menjadi format digital sehingga ruang penyimpanan menjadi efisien dan pengurangan penggunaan kertas. Bisa juga sebagai backup untuk membuat salinan digital dari dokumen penting sebagai cadangan jika dokumen asli hilang atau rusak. Dari sisi aksesibilitas, dokumen digital mudah

* Corresponding author

dibagikan melalui surat elektronik atau penyimpanan *cloud*, dan pengaksesan dokumen bisa darimana saja tanpa harus membawa dokumen fisik. Pencarian dan menemukan dokumen digital lebih cepat dan dapat dikolaborasi lebih baik dalam sebuah tim pada dokumen digital yang sama.

Akan tetapi tidak semua file tersebut mudah dibaca jika hasil pindai tersebut tidak tegak lurus. Maka dari itu, diperlukan sebuah algoritma berbasis fitur geometris yang dapat membetulkan posisi orientasi teks dalam file tersebut. Transformasi *Hough* adalah algoritma yang digunakan dalam pemrosesan citra dan pengenalan pola untuk mendeteksi bentuk-bentuk tertentu dalam gambar, terutama garis yang dimodelkan secara parameter [8][9]. Dalam penelitian ini, diusulkan menggunakan metode transformasi *Hough* untuk mendeteksi garis dalam gambar sehingga dapat menemukan orientasi teks berdasarkan sudut garis horizontal atau vertikal yang dominan.

II. TINJAUAN PUSTAKA

A. Transformasi *Hough*

Transformasi *Hough* merupakan algoritma paling populer dan efektif untuk mendeteksi garis gambar, dimana parameter garis sudut digunakan pertama kali pada tahun 1972 [10]. Transformasi *Hough* digunakan dalam pemrosesan citra dan pengenalan pola untuk mendeteksi bentuk-bentuk tertentu dalam gambar, terutama garis, lingkaran, dan bentuk-bentuk lainnya. Penggunaannya bermanfaat dalam analisis gambar digital untuk mengidentifikasi fitur-fitur geometris.

Transformasi *Hough* dibahas dalam penelitian sebelumnya, penelitian deteksi garis dalam pelacakan target pada platform udara [11], transformasi *Hough* menghasilkan hasil yang lebih akurat dan stabil dalam data eksperimen dibandingkan RANSAC. Lalu penelitian mempelajari estimasi pergerakan agregasi sel darah merah [12], transformasi *Hough* lebih akurat dan lebih sensitif daripada *Fourier*. Penelitian mendeteksi garis pada kabel listrik [13], transformasi *Hough* dapat mengoreksi garis putus yang terpisah oleh Filter Gabor. Berdasarkan penelitian-penelitian tersebut, dapat disimpulkan transformasi *Hough* lebih unggul dalam hal keakuratan, kestabilan, dan kesensitivitas, serta dapat memperbaiki garis yang sudah diproses dengan Filter Gabor, maka dari itu pada penelitian ini diusulkan transformasi *Hough* sebagai acuan dalam mengoreksi orientasi teks pada dokumen digital. Selain itu, transformasi *Hough* juga tersedia dalam OpenCV.

Langkah pertama untuk mendeteksi garis adalah membuat representasi parameter garis, terutama untuk garis vertikal. Persamaan yang dipakai seperti pada rumus 1.

$$\rho = x \cos \theta + d \sin \theta \quad (1)$$

Dimana:

ρ = jarak terpendek dari asal (0,0) ke garis

θ = sudut garis terhadap sumbu x

Kemudian dilanjutkan dengan melakukan transformasi ke ruang parameter yang dimana untuk setiap titik tepi (*edge*) dalam gambar, algoritma *Hough* akan menghitung semua pasangan (ρ, θ) yang mungkin untuk garis yang melewati titik tersebut dan menghasilkan kurva sinusoidal dalam ruang parameter. Berikutnya langkah pemetaan dan akumulasi, dimana sebuah matriks akumulator digunakan untuk mencatat jumlah kejadian setiap pasangan (ρ, θ). Setiap kali garis melewati sebuah titik tepi, nilai di akumulator untuk pasangan (ρ, θ) tersebut ditingkatkan. Setelah semua titik tepi diproses, puncak dalam matriks akumulator menunjukkan parameter garis yang paling mungkin ada dalam gambar. Puncak ini sesuai dengan garis yang sering dilewati oleh titik-titik tepi. Terakhir menggunakan pasangan (ρ, θ) dari puncak akumulator untuk menggambar garis pada gambar asli.

Perkembangan transformasi *Hough* juga dapat dilakukan seperti oleh Yancong Lin [8] dengan menggabungkan transformasi *Hough* dan inversi transformasi *Hough* (HT-IHT) yang bertujuan untuk meningkatkan efisiensi dan mengurangi ukuran data pelatihan. Kai Zhao [9] juga menggabungkan teknik transformasi *Hough* dengan CNN serta merancang metrik evaluasi baru dengan mempertimbangkan jarak *Euclidean* dan jarak sudut antara garis, yang menghasilkan kecepatan deteksi lebih baik secara signifikan dibanding metode lainnya dalam hal kuantitatif dan kualitatif.

B. Canny Edge Detection

Canny Edge Detection atau bisa disebut dengan algoritma *Canny* merupakan algoritma dalam pemrosesan citra untuk pendeteksian tepi [14]. Algoritma ini dirancang untuk mendeteksi tepi secara optimal dengan meminimalkan kesalahan deteksi, memastikan presisi lokasi dan hanya mendeteksi satu tepi per fitur. Penelitian sebelumnya yang terkait penggunaan algoritma *Canny* yaitu dalam identifikasi retinopati diabetik [15], pemberian *watermark* pada gambar [16], dan pendeteksian kapal pada lalu lintas maritim [17]. Algoritma *Canny* mengambil inputan berupa gambar *grayscale* atau skala abu-abu dan menghasilkan output gambar yang menunjukkan lokasi diskontinuitas intensitas atau tepi. Algoritma *Canny* pun tersedia di OpenCV.

Langkah pertama adalah mendeteksi intensitas tepi dengan menggunakan operator Sobel untuk menghitung gradien dalam arah horizontal dan vertikal. Persamaan ditunjukkan pada rumus 2.

$$G_x = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix}, G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ +1 & +2 & +1 \end{bmatrix} \quad (2)$$

Lalu menghitung magnitudo gradien dengan persamaan yang ditampilkan pada rumus 3.

$$G = \sqrt{G_x^2 + G_y^2} \quad (3)$$

Dilanjutkan dengan menghitung arah gradien dengan persamaan pada rumus 4.

$$\theta = \arctan \left(\frac{G_y}{G_x} \right) \quad (4)$$

Langkah selanjutnya adalah penekanan non-maksimum dengan tujuan mempertajam tepi dengan menekan piksel yang bukan bagian dari garis tepi utama. Dengan cara membandingkan intensitas gradien piksel dengan tetangganya dalam arah gradien (θ), dan jika intensitas piksel tidak maksimum dibandingkan dengan tetangganya dalam arah gradien, nilai intensitas piksel diubah menjadi 0.

Terakhir adalah *Hysteresis Thresholding*, yang bertujuan untuk menentukan tepi yang “kuat” dan “lemah” berdasarkan intensitas. Cara yang dilakukan adalah menggunakan dua nilai ambang batas *High Threshold* (HT) dan *Low Threshold* (LT). Jika piksel dengan intensitas lebih tinggi dari HT dianggap tepi kuat, jika piksel dengan intensitas antara LT dan HT dianggap tepi lemah, dan jika piksel dengan intensitas lebih rendah dari LT diabaikan. Lalu menghubungkan tepi lemah ke tepi kuat jika mereka bertetangga. Jika tidak maka tepi lemah diabaikan.

C. Grayscale

Grayscale atau skala abu-abu adalah format representasi citra dimana setiap piksel diwakili oleh satu nilai intensitas, biasanya dari hitam (nilai terendah), abu-abu, hingga putih (nilai tertinggi) [18]. Nilai intensitas piksel berkisar antara 0 hingga 255 untuk gambar 8-bit. 0 untuk hitam atau intensitas terendah dan 255 untuk putih atau intensitas tertinggi. Gambar *grayscale* hanya memiliki 2 dimensi yaitu tinggi dan lebar. Karena *grayscale* hanya memiliki 2 dimensi, maka gambar *grayscale* memerlukan lebih sedikit penyimpanan dibandingkan gambar berwarna. Gambar *grayscale* menjadi dasar bagi banyak teknik dalam pemrosesan citra [19], visi komputer [20], dan grafik komputer [21] maupun citra medis [22] misalnya *X-Ray* dan *CT Scan*.

Proses konversi gambar berwarna (RGB) ke *grayscale* dilakukan dengan menggabungkan nilai intensitas dari saluran warna merah (R), hijau (G), dan biru (B) menjadi satu nilai. Koefisien $R = 0.2989$, $G = 0.5870$, dan $B = 0.1140$ [23] berasal dari sensitivitas mata manusia terhadap warna dimana mata manusia lebih sensitif terhadap warna hijau maka G memiliki bobot tertinggi. Persamaan umum yang digunakan pada rumus 5

$$I = 0.2989 \times R + 0.5870 \times G + 0.1140 \times B \quad (5)$$

Dimana:

I = nilai intensitas *grayscale*

D. Gaussian Blur

Gaussian blur atau *gaussian smoothing* bertujuan mengurangi detail dan *noise* atau kebisingan pada gambar karena noise dapat menyebabkan tepi palsu. Penemu konsep ini adalah Johann Carl Friedrich Gauss yang merupakan matematikawan dan ilmuwan yang berasal dari Jerman [24]. Penggunaan *gaussian blur* sebagai langkah *pre-processing* dalam proses pengolahan citra, misal penelitian dalam mendeteksi cacat isolator saluran listrik [25], mengenali penyakit pada tanaman singkong [26], dan mengklasifikasi tumor otak di bidang medis. Persamaannya ditampilkan pada rumus 6.

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (6)$$

Dimana:

x, y = koordinat piksel

σ = deviasi standar

E. Rotasi Gambar

Rotasi gambar adalah transformasi geometris di mana sebuah gambar diputar di sekitar titik tertentu (biasanya pusat gambar) pada sudut tertentu, searah jarum jam atau berlawanan arah jarum jam. Rotasi ini dapat dilakukan dalam ruang dua dimensi (2D) atau tiga dimensi (3D) [27]. Pengaplikasian rotasi gambar dibutuhkan pada pengolahan citra untuk menyesuaikan orientasi gambar atau memutar gambar ke arah yang diinginkan, lalu di bidang visi komputer dalam augmentasi data yang dimana memperbanyak data pelatihan model kecerdasan buatan, untuk animasi, desain grafis maupun fotografi.

Ciri rotasi gambar mempunyai titik pusat rotasi dimana titik tetap walaupun gambar diputar. Sudut rotasi adalah besarnya putaran yang diukur dalam derajat ($^{\circ}$) atau radian, jika nilainya adalah positif maka rotasi berlawanan dengan arah jarum jam, dan jika nilai adalah negatif maka rotasi ke arah dengan jarum jam. Rotasi gambar dilakukan dengan memutar koordinat piksel (x, y) berdasarkan titik pusat $(0, 0)$ dengan persamaan yang ditunjukkan pada rumus 7.

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \quad (7)$$

Dimana:

x', y' = koordinat baru setelah rotasi

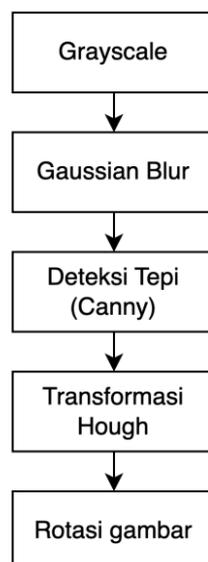
F. File Scan

File scan adalah dokumen digital yang dihasilkan dari proses pemindaian sebuah dokumen fisik menggunakan perangkat seperti *scanner* [28]. Biasanya berupa gambar atau dokumen elektronik lainnya seperti dalam bentuk PDF (*Portable Document Format*) yang merepresentasikan salinan dari dokumen asli. Hasil gambar disimpan dalam berbagai format seperti JPEG, PNG, TIFF dan BMP. Kegunaan dokumen digital untuk sebagai arsip, berbagi informasi atau memproses dokumen untuk analisa lebih lanjut seperti pengenalan teks melalui *Optical Character Recognition* (OCR).

III. METODE

Dataset dari Kaggle [29] dipakai dalam penelitian ini karena mempunyai *noise* dan tidak memiliki orientasi teks yang tegak lurus. Dataset sebanyak 600 file gambar dengan format PNG dan mempunyai ukuran yang sama yaitu 400 x 566 piksel. Gambar-gambar tersebut merupakan hasil pindai berupa ketikan dan diketik dalam huruf alfabet.

Langkah-langkah yang diusulkan dalam penelitian ini seperti yang dijabarkan pada gambar 1.



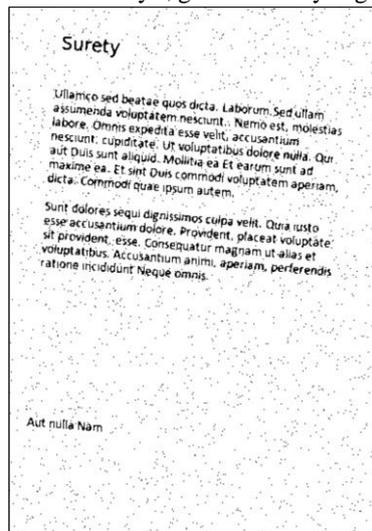
Gambar 1 Metodologi Penelitian

1. *Grayscale* merupakan tahap awal sebuah gambar yang akan dikonversikan ke skala abu-abu dengan koefisien $R = 0.2989$, $G = 0.5870$, dan $B = 0.1140$.
2. *Gaussian Blur* dilakukan untuk menghaluskan gambar dengan mengurangi *noise*.

3. Deteksi tepi atau *Canny Edge Detection* dimana tahap lanjutan yang diawali dengan pendeteksian intensitas tepi, magnitudo gradien, menghitung arah gradien, penekanan non-maksimum dan *Hysteresis Thresholding* untuk menentukan tepi kuat dan tepi lemah.
4. Transformasi *Hough* diterapkan untuk mendeteksi garis.
5. Rotasi gambar dilakukan dengan kondisi jika nilai sudut rata-rata dengan nilai positif lebih dari 1 derajat maka akan dirotasi berlawanan dari arah jarum jam. Dan jika nilai sudut rata-rata dengan nilai negatif kurang dari 1 derajat maka akan dirotasi searah jarum jam. Gambar-gambar tersebut diputar untuk dikoreksi orientasinya ke titik pusat (0, 0).

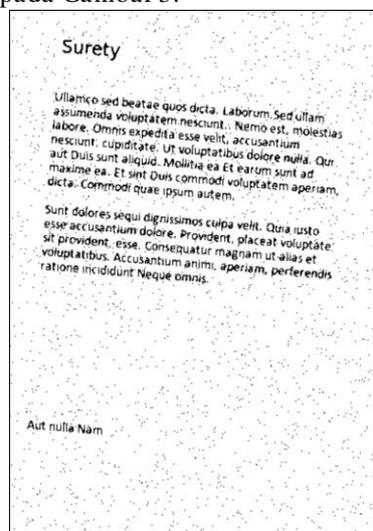
IV. HASIL

Dari metode yang dijabarkan di bab sebelumnya, gambar asli yang dipakai dapat dilihat di Gambar 2.



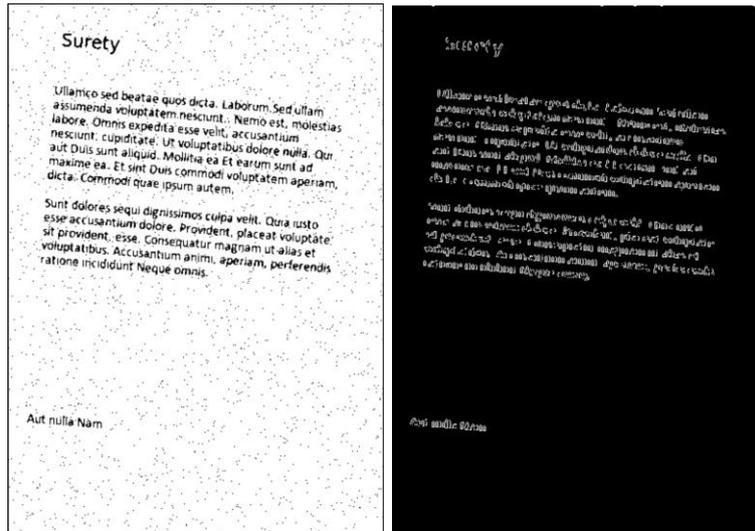
Gambar 2 Gambar asli

Dilakukan proses *grayscale* atau konversi ke skala abu-abu dengan koefisien $R = 0.2989$, $G = 0.5870$, dan $B = 0.1140$ sebagai langkah awal proses pada Gambar 3.



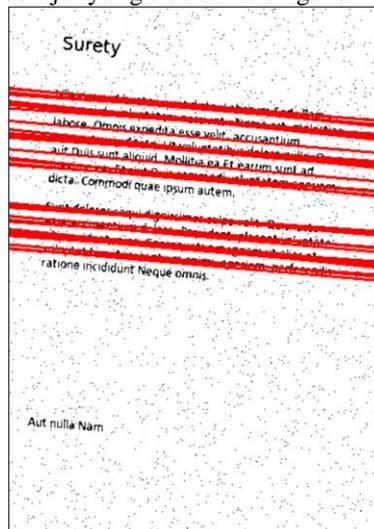
Gambar 3 Proses *grayscale*

Lalu tahap kedua bisa dilihat pada Gambar 4, *gaussian blur* dilakukan untuk menghilangkan *noise* (kiri) dan dilanjutkan proses *Canny Edge Detection* (kanan).



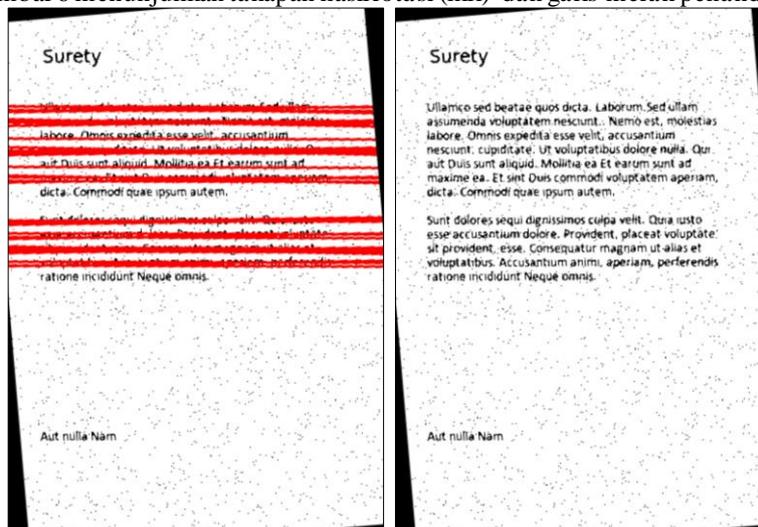
Gambar 4 Proses *Gaussian Blur* (kiri) dan hasil deteksi tepi (kanan)

Gambar 5 menunjukkan tahapan pendeteksian garis dengan transformasi *Hough*. Nilai sudut rata-rata yang didapat dari gambar ini adalah 4.848 derajat yang ditandai oleh garis merah.



Gambar 5 Hasil deteksi garis

Karena sudut rata-rata yang didapat adalah bernilai positif, maka rotasi pada gambar dilakukan berlawanan arah jarum jam. Gambar 6 menunjukkan tahapan hasil rotasi (kiri) dan garis merah penanda dihilangkan (kanan).

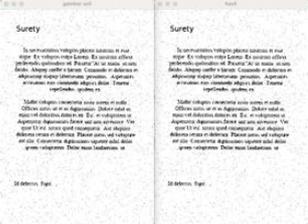
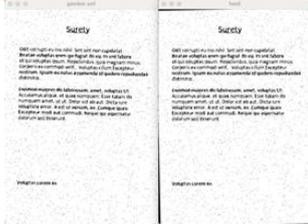
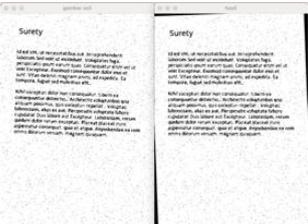
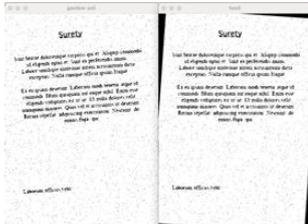
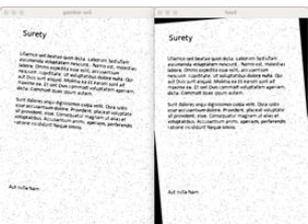
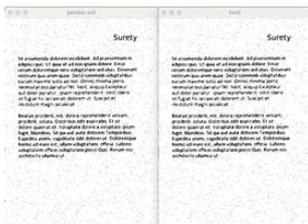
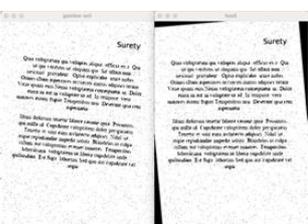
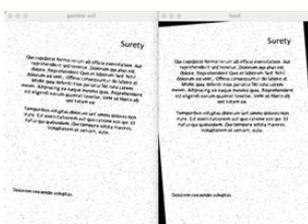
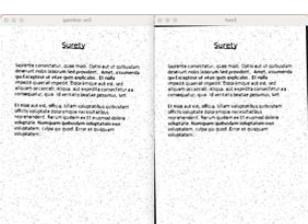
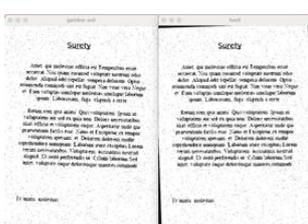


Gambar 6 Hasil rotasi (kiri) dan garis merah penanda dihilangkan (kanan)

V. PEMBAHASAN

Berdasarkan hasil pemrosesan di atas, maka dilakukan sejumlah 10 gambar sampel secara acak. Berikut adalah contoh hasil sejumlah 10 gambar setelah melalui proses yang sama diikuti dengan nilai sudut rata-rata pada Tabel 1.

TABEL 1
HASIL PADA 10 CONTOH GAMBAR YANG BERBEDA

No	Gambar dan Hasil	Nilai Sudut Rata-Rata	Waktu proses (detik)	No	Gambar dan Hasil	Nilai Sudut Rata-Rata	Waktu proses (detik)
1		0.226	0.011	6		1.103	0.018
2		2.060	0.021	7		-3.944	0.021
3		4.848	0.022	8		-0.137	0.021
4		4.903	0.022	9		4.649	0.018
5		1.028	0.014	10		1.892	0.056

Dari tabel 1 dapat disimpulkan sebanyak delapan dari sepuluh gambar yang mengalami perubahan rotasi Gambar 1 dan 8 tidak mengalami rotasi karena memiliki nilai sudut rata-rata sebesar 0.226 dan -0.137, dimana kedua nilai tersebut kurang dari 1 derajat baik nilai positif maupun negatif. Untuk nilai sudut rata-rata positif yang lebih dari 1 derajat, gambar akan dirotasi berlawanan searah jarum jam. Sedangkan untuk nilai sudut rata-rata negatif yang kurang dari 1 derajat, gambar dirotasi searah jarum jam seperti pada gambar ke 7. Untuk waktu pemrosesan memiliki rata-rata selama 0,022 detik tiap gambar. Hal ini menunjukkan keseluruhan proses cukup cepat karena kurang dari 1 detik.

Hasil dari delapan gambar yang telah mengalami rotasi pada tabel 1 diuji kembali untuk memeriksa apakah mendapatkan nilai sudut rata-rata sudah kurang dari 1 derajat seperti yang ditunjukkan pada tabel 2.

TABEL 2
NILAI SUDUT RATA-RATA DARI HASIL YANG SUDAH DIPROSES

No	Gambar ke-	Nilai Sudut Rata-Rata (sebelum)	Nilai Sudut Rata-Rata (setelah)
1	2	2.060	-0.118
2	3	4.848	-0.784
3	4	4.903	-1.094
4	5	1.028	-0.081
5	6	1.103	0
6	7	-3.944	1.235
7	9	4.649	-1.500
8	10	1.892	-0.458

Tabel 2 dapat disimpulkan dari delapan gambar tersebut, hanya 3 gambar yang masih lebih dari 1 derajat yaitu gambar 4, 7 dan 9 dengan nilai sudut rata masing-masing adalah 1.094, 1.235, dan -1.500. Akan tetapi nilai-nilai sudut tersebut mendekati 1 derajat, tidak melebihi 2 derajat. Hal tersebut tidak menjadi masalah untuk indera penglihatan manusia karena nilai sudut rata-rata di antara 1 dan 2 derajat tidaklah terlalu signifikan dan masih dapat ditolerir.

VI. KESIMPULAN

Hasil dari 10 sampel gambar yang diuji sekitar 70% gambar yang mendekati tegak lurus atau nilai sudut rata-rata kurang dari 1 derajat. Sedangkan 30% gambar lainnya memiliki nilai sudut rata-rata sebesar 1.094, 1.235, dan -1.500, yang dimana nilai-nilai tersebut masih di antara 1 dan 2 derajat setelah diuji kembali yang sebenarnya hampir mendekati orientasi tegak lurus. Hal ini sebenarnya tidak menjadi masalah karena nilai sudut rata-rata di antara 1 dan 2 derajat tidak terlalu signifikan dan masih dapat ditolerir oleh indera penglihatan manusia. Untuk waktu pemrosesan rata-rata tiap gambar adalah 0,022 detik dimana terbilang cukup cepat karena di bawah 1 detik. Hasil penelitian ini dapat digunakan untuk memperbaiki kemiringan teks pada dokumen hasil pemindaian dan meningkatkan akurasi sistem OCR.

REFERENCES

- [1] M. L. Chambers, *Scanners For Dummies*. in –For dummies. Wiley, 2004. [Online]. Available: <https://books.google.co.id/books?id=hivOZOYiSy8C>
- [2] S. L. Buchmann, “Moths on the Flatbed Scanner: The Art of Joseph Scheer,” *Insects*, vol. 2, no. 4, pp. 564–583, Dec. 2011, doi: 10.3390/insects2040564.
- [3] Federal Research Center " et al., “Towards a unified framework for identity documents analysis and recognition,” *Computer Optics*, vol. 46, no. 3, Jun. 2022, doi: 10.18287/2412-6179-CO-1024.
- [4] E. Callegari et al., “The Precision, Inter-Rater Reliability, and Accuracy of a Handheld Scanner Equipped with a Light Detection and Ranging Sensor in Measuring Parts of the Body—A Preliminary Validation Study,” *Sensors*, vol. 24, no. 2, p. 500, Jan. 2024, doi: 10.3390/s24020500.
- [5] J. F. Kelly, *The Ultimate iPad: Your Digital Life at Your Fingertips*. Pearson Education, 2014. [Online]. Available: <https://books.google.co.id/books?id=V2QHBAAAQBAJ>
- [6] J. D. Monson, *Getting Started with Digital Collections: Scaling to Fit Your Organization*. American Library Association, 2017. [Online]. Available: <https://books.google.co.id/books?id=MXs4DwAAQBAJ>
- [7] J. M. Perrin, *Digitizing Flat Media: Principles and Practices*. in LITA Guides. Rowman & Littlefield Publishers, 2015. [Online]. Available: <https://books.google.co.id/books?id=epbnCgAAQBAJ>
- [8] Y. Lin, S. L. Pinteá, and J. C. Van Gemert, “Deep Hough-Transform Line Priors,” in *Computer Vision – ECCV 2020*, vol. 12367, A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, Eds., in Lecture Notes in Computer Science, vol. 12367. , Cham: Springer International Publishing, 2020, pp. 323–340. doi: 10.1007/978-3-030-58542-6_20.
- [9] K. Zhao, Q. Han, C.-B. Zhang, J. Xu, and M.-M. Cheng, “Deep Hough Transform for Semantic Line Detection,” *IEEE Trans. Pattern Anal. Mach. Intell.*, pp. 1–1, 2021, doi: 10.1109/TPAMI.2021.3077129.
- [10] R. O. Duda and P. E. Hart, “Use of the Hough transformation to detect lines and curves in pictures,” *Commun. ACM*, vol. 15, no. 1, pp. 11–15, Jan. 1972, doi: 10.1145/361237.361242.
- [11] L. Jacobs, J. Weiss, and D. Dolan, “Object tracking in noisy radar data: Comparison of Hough transform and RANSAC,” in *IEEE International Conference on Electro-Information Technology, EIT 2013*, Rapid City, SD, USA: IEEE, May 2013, pp. 1–6. doi: 10.1109/EIT.2013.6632715.
- [12] A. Kempczyński, M. Bosek, and B. Grzegorzewski, “Comparison of Hough and Fourier transform approach in the study of kinetics of red blood cell aggregates,” presented at the 16th Polish-Slovak-Czech Optical Conference on Wave and Quantum Aspects of Contemporary Optics, Polanica Zdroj, Poland, Dec. 2008, p. 714118. doi: 10.1117/12.822389.

- [13] U. Marmol and N. Borowiec, "Detection of Line Objects by Means of Gabor Wavelets and Hough Transform," *Archives of Civil Engineering*, pp. 339–363, Jul. 2020, doi: 10.24425/ace.2020.134401.
- [14] J. Canny, "A Computational Approach to Edge Detection".
- [15] G. U. Nneji, J. Cai, J. Deng, H. N. Monday, M. A. Hossin, and S. Nahar, "Identification of Diabetic Retinopathy Using Weighted Fusion Deep Learning Based on Dual-Channel Fundus Scans," *Diagnostics*, vol. 12, no. 2, p. 540, Feb. 2022, doi: 10.3390/diagnostics12020540.
- [16] Z. B. Faheem *et al.*, "Image Watermarking Using Least Significant Bit and Canny Edge Detection," *Sensors*, vol. 23, no. 3, p. 1210, Jan. 2023, doi: 10.3390/s23031210.
- [17] X. Chen, J. Ling, S. Wang, Y. Yang, L. Luo, and Y. Yan, "Ship detection from coastal surveillance videos via an ensemble Canny-Gaussian-morphology framework," *J. Navigation*, vol. 74, no. 6, pp. 1252–1266, Nov. 2021, doi: 10.1017/S0373463321000540.
- [18] N. Cao and Y. Liu, "High-Noise Grayscale Image Denoising Using an Improved Median Filter for the Adaptive Selection of a Threshold," *Applied Sciences*, vol. 14, no. 2, p. 635, Jan. 2024, doi: 10.3390/app14020635.
- [19] J. Li and X. Gui, "Fully Automatic Grayscale Image Segmentation: Dynamic Thresholding for Background Adaptation, Improved Image Center Point Selection, and Noise-Resilient Start/End Point Determination," *Applied Sciences*, vol. 14, no. 20, p. 9303, Oct. 2024, doi: 10.3390/app14209303.
- [20] J. Wang and S. Lee, "Data Augmentation Methods Applying Grayscale Images for Convolutional Neural Networks in Machine Vision," *Applied Sciences*, vol. 11, no. 15, p. 6721, Jul. 2021, doi: 10.3390/app11156721.
- [21] University of Kufa / Faculty of Education / Department of Computer Science/Iraq, A. A. M. Qazzaz, H. A. Alsabbagh, and University of Kufa / Faculty of Education / Department of Computer Science/Iraq, "An Overview of the Most Important Methods for Coloring Grayscale Images," *FJIECE*, vol. 3, no. 1, pp. 45–67, Apr. 2024, doi: 10.46649/fjiece.v3.1.5a.14.4.2024.
- [22] T. Siriapisith, W. Kusakunniran, and P. Haddawy, "Pyramid graph cut: Integrating intensity and gradient information for grayscale medical image segmentation," *Computers in Biology and Medicine*, vol. 126, p. 103997, Nov. 2020, doi: 10.1016/j.compbiomed.2020.103997.
- [23] "RECOMMENDATION ITU-R BT.601-7 – Studio encoding parameters of digital television for standard 4:3 and wide-screen 16:9 aspect ratios".
- [24] "Carl Friedrich Gauss - Biography," Maths History. Accessed: Nov. 27, 2024. [Online]. Available: <https://mathshistory.st-andrews.ac.uk/Biographies/Gauss/>
- [25] X. Tao, D. Zhang, Z. Wang, X. Liu, H. Zhang, and D. Xu, "Detection of Power Line Insulator Defects Using Aerial Images Analyzed With Convolutional Neural Networks," *IEEE Trans. Syst. Man Cybern, Syst.*, vol. 50, no. 4, pp. 1486–1498, Apr. 2020, doi: 10.1109/TSMC.2018.2871750.
- [26] O. O. Abayomi-Alli, R. Damaševičius, S. Misra, and R. Maskeliūnas, "Cassava disease recognition from LOW-QUALITY images using enhanced data augmentation model and deep learning," *Expert Systems*, vol. 38, no. 7, p. e12746, Nov. 2021, doi: 10.1111/exsy.12746.
- [27] P.-E. Danielsson and M. Hammerin, "High-accuracy rotation of images," *CVGIP: Graphical Models and Image Processing*, vol. 54, no. 4, pp. 340–344, 1992, doi: [https://doi.org/10.1016/1049-9652\(92\)90080-H](https://doi.org/10.1016/1049-9652(92)90080-H).
- [28] P. Royster, "The Art of Scanning".
- [29] "Dataset Scanned documents OCR." Accessed: Jun. 28, 2024. [Online]. Available: <https://www.kaggle.com/datasets/omnamahshivai/dataset-scanned-documents-ocr>