

Analisa Titik Sudut Pada Pemindai Dokumen Menggunakan Python

Yoga Putra Pratama¹, Rachmah Nanda Kartika², Prayudi Arika³, Rachmad Eko Pranoto⁴,
Ramdani Harjo³

Politeknik Negeri Jakarta

yoga.putra.pratama@grafika.pnj.ac.id, rachmah.nandakartika@grafika.pnj.ac.id ,

prayudhi.arika@grafika.pnj.ac.id, eko.pranoto@grafika.pnj.ac.id ,

ramdhani.harjo@grafika.pnj.ac.id

Abstract

Artificial intelligence is an evolving technology, and one of its applications is in document scanners. In this project, Python is used to improve the quality of damaged documents. The goal is to create a highly precise document scanner that can be used by various users. The design incorporates edge detection and corner detection to accurately select documents. The findings indicate that while Python can run the program, it crashes when handling partially cropped images. Additionally, the program's adaptability makes it suitable for use on different platforms.

Keywords : Python, Document , Edge Detection, Corner Detection

1. PENDAHULUAN

Era digitalisasi telah menjadi kebutuhan penting dalam kehidupan, termasuk kebutuhan dokumen cetak. Meskipun begitu, dokumen cetak masih banyak digunakan. Kerusakan pada kertas yang disebabkan oleh kondisi lingkungan sekitar dan data penting dalam dokumen mengharuskan penyimpanan dokumen dalam bentuk digital. Kemajuan teknologi, terutama penggunaan *Artificial Intelligence (AI)*, telah merambah ke berbagai sektor untuk memberikan solusi yang efektif. Seiring bertambahnya dokumen cetak, diperlukan pemindai dokumen untuk mempermudah penyimpanan.

Python merupakan salah satu pemrograman yang memiliki fitur kecerdasan buatan. Python memiliki kemudahan untuk dipelajari dan dikembangkan berkat dukungan *library* dan komunitas yang besar. Kemudian *library OpenCV*, sebuah *library* untuk pengolahan gambar, mampu menyediakan program dengan kemampuan pengolahan visual dan deteksi objek yang akurat.

Pada penelitian ini, pendeteksi tepi digunakan untuk membatasi area garis tepian pada objek gambar, memberikan tanda pada bagian tertentu, dan memperbaiki bagian gambar yang akan dipisahkan. Deteksi tepi juga membantu dalam pemilihan bagian dokumen yang akan ditandai. Setelah deteksi tepi dilakukan, deteksi sudut digunakan untuk menentukan presisi objek, seperti yang ditunjukkan dalam penelitian sebelumnya dengan hasil akurasi dan tingkat keakuratan yang baik.

2. TINJAUAN PUSTAKA

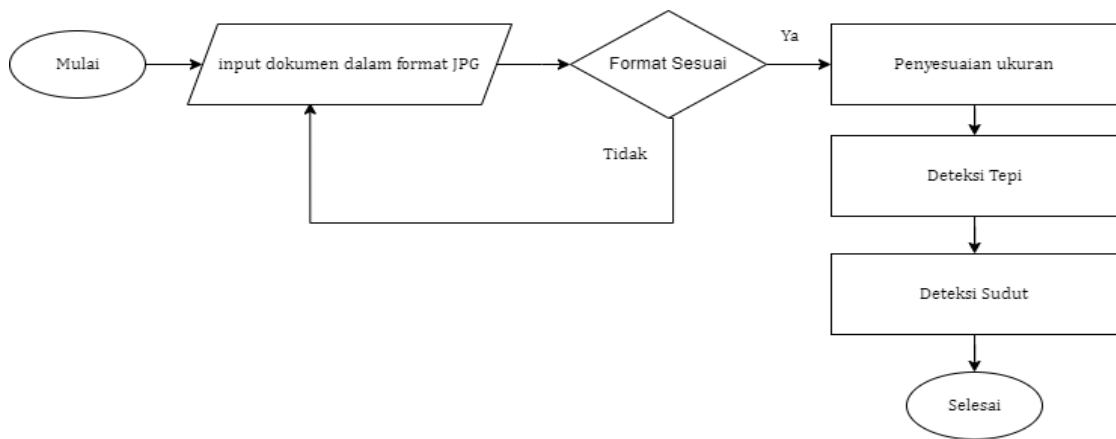
- a. Penelitian yang berjudul “Upaya Penyelamatan Informasi Melalui Proses Digitalisasi Arsip Iuran Pembangunan Daerah Pada Tahun 1984-1986 Di Desa Nagrak” (E Syahadatina dkk, 2021) pentingnya digitalisasi arsip untuk penyimpanan dikarenakan kertas memiliki kerentanan terhadap suhu sehingga menyebabkan kerusakan
- b. Penelitian yang berjudul “Studi kasus pengembangan dan penggunaan *Artificial Intelligence* sebagai penunjang kegiatan masyarakat Indonesia”, (T Wahyudi, 2023), pemanfaatan teknologi kecerdasan buatan untuk mengoptimalkan kualitas dengan menyesuaikan perangkat yang tersedia ;

- c. Penelitian yang memiliki judul “ Penggunaan Metode Canny Edge Detection Untuk Mendeteksi Tepi Uang Kertas Asli Berbasis Java” (H Alam dkk, 2023) penelitian ini menggunakan metode deteksi tepi untuk menganalisis tepian pada uang kertas ;

3. METODE PENELITIAN

Pada penelitian ini , program melakukan pengecekan yaitu fokus kepada objek benda yang mudah mengalami kerusakan. Objek yang akan dianalisa menggunakan format gambar berbentuk JPG yang secara umum digunakan pada perangkat seperti smartphone. dokumen yang digunakan untuk sampel ujicoba dengan berbagai kondisi. Kemudian apabila format dokumen yang dideteksi sudah sesuai maka akan dilakukan penyesuaian gambar. Melakukan dan kemudian program akan menganalisa sudut berjumlah 4 titik sesuai bentuk dokumen.

3.1 Prosedur Penelitian



Gambar 1. Flowchart Penelitian

Pada Perancangan ini , gambar diinput ke dalam bahasa pemrograman python dengan format jpg, kemudian dilakukan pengecekan apakah format yang dimasukan sudah sesuai, setelah itu gambar disesuaikan ukurannya agar objek terlihat jelas, setelah itu dilakukan deteksi tepi dan deteksi sudut untuk melihat apakah dokumen dapat terlihat dengan jelas.

4. HASIL PENELITIAN DAN PEMBAHASAN

Pengujian dari penelitian ini menggunakan sebanyak 5 sampel dokumen yang terdapat di berbagai kondisi, yaitu kondisi baik, kondisi pada tempat yang kekurangan cahaya, dan gambar yang terpotong pada salah satu bagian sudut. Ujicoba ini menggunakan pemrograman python untuk mengetahui hasil dari pemindai dokumen tersebut.

4.1 Hasil Rancangan Program

Pada penelitian ini, rancangan program dibuat dengan menggunakan bahasa Python dengan menggunakan library OpenCV untuk dapat memiliki dukungan terhadap penelitian ini dan menggunakan library skimage untuk mengolah gambar

```

test1.py - C:\Users\yppra\Downloads\DocumentScanner\test1.py (3.12.1)
File Edit Format Run Options Window Help
import cv2
import imutils
from skimage.filters import threshold_local
from pyimagesearch.transform import four_point_transform
import numpy as np
import aspose.words as aw

img_path = 'tes28.jpg'
big_img = cv2.imread(img_path)
cv2.imshow('org img',big_img)
cv2.waitKey(0)

ratio = big_img.shape[0] / 500.0
org = big_img.copy()
img = imutils.resize(big_img, height = 500)
cv2.imshow('resizing',img)
cv2.waitKey(0)

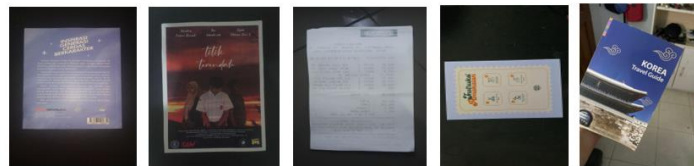
gray_img = cv2.cvtColor(img.copy(),cv2.COLOR_BGR2GRAY)
blur_img = cv2.GaussianBlur(gray_img,(5,5),0)
edged_img = cv2.Canny(blur_img,75,200)
cv2.imshow('edged',edged_img)
cv2.waitKey(0)

cnts,_ = cv2.findContours(edged_img.copy(),cv2.RETR_LIST,cv2.CHAIN_APPROX_SIMPLE)
cnts = sorted(cnts,key=cv2.contourArea,reverse=True)[:5]
for c in cnts:
    peri = cv2.arcLength(c,True)
    approx = cv2.approxPolyDP(c,0.02*peri,True)
    if len(approx)==4:
        doc = approx
        break
    
```

Gambar 2 Perancangan Program

4.2 Sampel Ujicoba

Sampel yang digunakan dalam penelitian ini yaitu 5 jenis seperti Sampul Buku, Poster Film, Struk Belanja, Instruksi Produk, dan Buku, penggunaan sampel ini dilakukan dikarenakan dokumen ini seringkali mengalami kerusakan pada seiring berjalannya waktu. Berikut adalah contoh sampel yang digunakan



Gambar 3. Sampel Uji coba

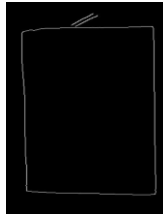
4.3 Hasil pengujian Sampel

Berdasarkan hasil pengujian sampel, tahapan yang dilakukan adalah melakukan deteksi tepi pada gambar untuk melihat bagian-bagian yang dapat dibaca oleh OpenCV, sehingga dapat mempermudah untuk melanjutkan pada deteksi sudut, kondisi pada saat pengambilan gambar memiliki pengaruh pada kualitas yang dihasilkan.

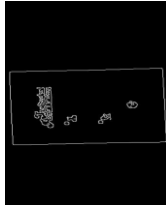
Tabel 1. Uji Coba Program

Nama Sampel	Hasil Deteksi Tepi	Hasil Deteksi Sudut
Sampul Buku		
Poster Film		

Struk Belanja



Instruksi Produk



Buku *Travel Guide*



Berdasarkan hasil uji coba maka didapatkan hasil sebagai berikut :

Hasil dari pengujian Deteksi Tepi menunjukkan beberapa temuan menarik. Pada Sampel Sampul Buku, garis tepi terdeteksi dengan baik karena kontras tinggi antara objek dan latar belakang, meski teks dengan kecerahan tinggi juga terdeteksi. Pada Poster Film memperlihatkan beberapa garis tepi akibat kontras tinggi di beberapa bagian, termasuk teks dan elemen dalam objek. Struk Belanja mendeteksi garis tepi pada objek serta bagian kecil di atasnya. Pada Instruksi Produk, deteksi tepi bekerja dengan baik meskipun pengambilan gambar dilakukan secara *landscape*. Sementara itu, Buku *Travel guide* menunjukkan deteksi tepi tetap efektif meski dokumen terpotong di ujungnya.

Untuk deteksi sudut, hasilnya sebagai berikut: Sampul Buku menunjukkan hasil yang jelas tanpa distorsi karena kontras baik dan latar belakang yang tidak ramai. Poster Film, meskipun pencahayaannya kurang, hasilnya cukup baik tetapi detail halus kurang tajam. Pada sampel Struk Belanja garis tepi pada objek dan bagian kecil di atasnya dapat terdeteksi dengan baik. Instruksi Produk menunjukkan hasil deteksi sudut yang baik dan terlihat jelas, dan Sampel 5 menunjukkan deteksi tepi yang terpotong pada bagian ujung menyebabkan program menjadi error dikarenakan tepian tidak terdeteksi secara penuh.

5. KESIMPULAN DAN SARAN

Dari penelitian yang sudah dilakukan, maka didapatkan kesimpulan bahwa Pemrograman Python dapat digunakan dan mendukung dalam penggunaan Kecerdasan Buatan termasuk didalamnya adalah pemindai dokumen. Namun dalam penelitian ini terdapat beberapa catatan yaitu diantaranya, dalam proses pengambilan gambar untuk dokumen, diperlukan pencahayaan yang baik, posisi pengambilan gambar yang tepat menghasilkan pindaian dokumen yang dapat terlihat jelas. Pada penelitian ini, Python dapat tetap mendeteksi hasil yang baik meskipun dalam kondisi kurang cahaya, posisi yang tidak tepat tetapi apabila gambar tidak penuh ataupun terpotong maka program akan secara otomatis menjadi error sehingga proses pindaian dokumen menjadi terhambat. Saran untuk penelitian selanjutnya yaitu digabungkan

dengan platform lain seperti menggunakan smartphone, website dan lain lain agar penggunaan menjadi lebih efektif dan efisien.

6. DAFTAR PUSTAKA

- A. A. Mekarisce, , 2020 “Teknik Pemeriksaan Keabsahan Data pada Penelitian Kualitatif di Bidang Kesehatan Masyarakat,” *J. Ilm. Kesehat. Masy. Media Komun. Komunitas Kesehat. Masy.*, vol. 12, no. 3, pp. 145–151
- A. Kurniawan, 2017 “Aplikasi Computer Vision Dengan Metode Fuzzy Image Processing Pada Text Reader,” *TELKA - Telekomun. Elektron. Komputasi dan Kontrol*, vol. 3, no. 2, pp. 94–102
- A. O. P. Dewi, , 2020 “Kecerdasan Buatan sebagai Konsep Baru pada Perpustakaan,” *Anuva J. Kaji. Budaya, Perpustakaan, dan Inf.*, vol. 4, no. 4, pp. 453–460.
- A. S. Anas and A. Rizal, 2019 “Deteksi Tepi Dalam Pengolahan Citra Digital,” *Semin. Nas. TIK dan Ilmu Sos.*, vol. 2, no. x, pp. 1–6.
- D. Susanti and M. S. Abdurrohman, 2020 “Rancang Bangun Alat Pemindai Dokumen Menggunakan Raspberry Pi,” *INFOTECH J.*, vol. 6, no. 1, pp. 44–48,
- E. Syahadatina, A. Badar, and W. Gamayanti, 2021 “Upaya Penyelamatan Informasi Melalui Proses Digitalisasi Arsip Iuran Pembangunan Daerah Pada Tahun 1984-1986 Di Desa Nagrak,” *Proceeding Uin Sunan Gunung Djati Bandung*, vol. 1, no. 29, pp. 59–69
- H. Alam, Y. Manurung, H. Widya, and W. H. Nanda, 2023 “Penggunaan Metode Canny Edge Detection Untuk Mendeteksi Tepi Uang Kertas Asli Berbasis Java,” *Fak. Tek.*, pp. 177–181
- M. D. N. Azis, S. A. Syakri, and Z. K. Simbolon, 2020 “Rancang Bangun Aplikasi Perbaikan Citra Hasil Scan Dokumen Lama Dengan Metode Filtering,” *J. Teknol. Rekayasa ...*, vol. 4, no. 1.
- S. Ghifari Aminudin Fad’li, Marsofiyati, 2023 “Implementasi Arsip Digital untuk Penyimpanan Dokumen Digital,” *J. Manuhara*, vol. 1, no. 4, pp. 1–10.
- T. Wahyudi, 2023 “Studi Kasus Pengembangan dan Penggunaan Artificial Intelligence (AI) Sebagai Penunjang Kegiatan Masyarakat Indonesia,”. vol. 9, no. 1 pp. 28–32