

PENGOLAHAN CITRA DIGITAL

“PERBAIKAN CITRA”



DISUSUN OLEH:

Nur Fadillah

F55121047

KELAS B

PROGRAM STUDI TEKNOLOGI INFORMASI

JURUSAN TEKNIK INFORMATIKA

FAKULTAS TEKNIK

UNIVERSITAS TADULAKO

2023

Kode program perbaikan citra tranformasi negative, penghalusan, dan peningkatan kecerahan.

```
1 import tkinter as tk
2 from tkinter import filedialog
3 from PIL import Image, ImageTk
4 import cv2
5 import numpy as np
6
7 # fungsi untuk memproses citra dengan metode Transformasi Negatif
8 def negative_transform(img):
9     negative_img = 255 - img
10    return negative_img
11
12 # fungsi untuk memperbaiki citra dengan metode smoothing
13 def smoothing_correction(img):
14     kernel = cv2.getStructuringElement(cv2.MORPH_RECT, (5,5))
15     smoothed_img = cv2.morphologyEx(img, cv2.MORPH_OPEN, kernel)
16     return smoothed_img
17
18 def sharpening(img):
19     kernel = np.array([[ -1,-1,-1], [-1,9,-1], [-1,-1,-1]])
20     sharpened_img = cv2.filter2D(img, -1, kernel)
21     return sharpened_img
22
23 def noise_reduction(img):
24     denoised_img = cv2.fastNlMeansDenoisingColored(img, None, 10, 10, 7, 21)
25     return denoised_img
26
27 # fungsi untuk memperbaiki citra dengan metode peningkatan kecerahan
28 def brightness_correction(img):
29     brightness = 50
30     corrected_img = cv2.add(img, brightness)
31     return corrected_img
32
33 # fungsi untuk menampilkan gambar dalam kotak
34 def show_image(img, x, y, title):
35     img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
36     img = Image.fromarray(img)
37     img = ImageTk.PhotoImage(img)
38     label = tk.Label(root, image=img)
39     label.image = img
40     label.place(x=x, y=y)
41     title_label = tk.Label(root, text=title)
42     title_label.place(x=x, y=y+20)
43
44 # fungsi untuk memproses citra dan menampilkan hasilnya
45 def process_image(method):
46     global original_img
47     if method == 'negative_transform':
48         corrected_img = negative_transform(original_img)
49         show_image(corrected_img, 300, 140, 'Hasil Transformasi Negatif')
50     elif method == 'smoothing':
51         corrected_img = smoothing_correction(original_img)
52         show_image(corrected_img, 500, 140, 'Hasil Metode Smoothing')
53     elif method == 'brightness':
54         corrected_img = brightness_correction(original_img)
55         show_image(corrected_img, 700, 140, 'Hasil Metode Kecerahan')
56
57 # fungsi untuk menampilkan informasi pembuat program
58 def show_creator():
59     creator_label = tk.Label(root, text='Nama : Nur Fadilah | NIM : F55121047 | Kelas : B | Jurusan : Teknologi Informasi | Prodi : Teknik Informa
60     creator_label.place(x=140, y=480)
61
62 # fungsi untuk membuka gambar
63 def open_image():
64     global original_img
65     file_path = filedialog.askopenfilename()
66     if file_path:
67         original_img = cv2.imread(file_path)
68         original_img = cv2.resize(original_img, (200, 250))
69         show_image(original_img, 70, 140, 'Gambar Original')
70         size_label.config(text='Dimensi: {} x {}'.format(original_img.shape[1], original_img.shape[0]))
71
72 # membuat jendela utama
73 root = tk.Tk()
74 root.geometry('1000x600')
75 root.title('GUI Aplikasi Pengolahan Citra')
76
77 # menambahkan judul gambar original
78 title_label = tk.Label(root, text='Gambar Original')
79 title_label.place(x=50, y=20)
80
81 # menambahkan tombol untuk membuka gambar
82 open_button = tk.Button(root, text='Buka Gambar', command=open_image)
83 open_button.place(x=50, y=50)
84
85 # menambahkan label untuk menampilkan dimensi gambar
86 size_label = tk.Label(root, text='Dimensi: -')
87 size_label.place(x=150, y=50)
88
89 # menambahkan kotak untuk metode perbaikan citra
90 correction_box = tk.LabelFrame(root, text='Metode Perbaikan Citra', padx=5, pady=5)
```

```

101 correction_box.place(x=350, y=20, width=400, height=70)
102
103 # tombol untuk metode Transformasi Negatif
104 negative_transform_button = tk.Button(correction_box, text='Transformasi Negatif', command=lambda: process_image('negative_transform'))
105 negative_transform_button.pack(side=tk.LEFT, padx=5)
106
107 # tombol untuk perbaikan metode smoothing
108 smoothing_button = tk.Button(correction_box, text='Penghalusan', command=lambda: process_image('smoothing'))
109 smoothing_button.pack(side=tk.LEFT, padx=5)
110
111 # tombol untuk perbaikan metode Peningkatan Kecerahan
112 brightness_button = tk.Button(correction_box, text='Peningkatan Kecerahan', command=lambda: process_image('brightness'))
113 brightness_button.pack(side=tk.LEFT, padx=5)
114
115 # menambahkan kotak untuk menampilkan hasil perbaikan citra
116 result_box = tk.LabelFrame(root, text='Hasil Perbaikan Citra', padx=5, pady=5)
117 result_box.place(x=50, y=100, width=900, height=330)
118
119 # menambahkan kotak untuk informasi pembuat program
120 creator_box = tk.LabelFrame(root, text='Disusun Oleh', padx=5, pady=5)
121 creator_box.place(x=130, y=450, width=700, height=70)
122
123 # menampilkan informasi pembuat program
124 show_creator()
125
126 # menjalankan aplikasi
127 root.mainloop()

```

Hasil perbaikan citra.

