

Muhammad Zidan Akbar

Mohamad Fawaz Tauhid Ridho

Dyas Tri Apriliansyah

Tugas Pengolahan citra (Morfologi citra)

Berikut adalah langkah-langkah untuk membuat aplikasi sederhana pengolahan morfologi citra menggunakan Python di Visual Studio Code (VSCode). Aplikasi ini akan memanfaatkan pustaka OpenCV untuk melakukan operasi morfologi dasar seperti dilasi dan erosi pada gambar.

Langkah 1: Persiapan Lingkungan

1. ***Install Python*** (jika belum): Unduh dan instal Python dari [python.org](https://www.python.org/downloads/).
2. ***Buka VSCode***: Pastikan Python telah terdeteksi di VSCode. Kalau belum, pilih interpreter Python yang benar.
3. ***Install Ekstensi Python***: Buka ****Extensions*** di sidebar VSCode dan cari "Python", lalu instal ekstensi tersebut.

Langkah 2: Buat Folder Proyek

1. Buat folder baru untuk proyek ini (misalnya MorphologyApp).
2. Di dalam VSCode, buka folder proyek tersebut dengan ***File > Open Folder***.

Langkah 3: Buat Virtual Environment (opsional, tapi disarankan)

1. Buka ***Terminal*** di VSCode (dengan menekan Ctrl + Shift + P, lalu ketik Terminal: Create New Integrated Terminal).
2. Ketik perintah berikut untuk membuat environment virtual:

```
bash
```

```
python -m venv venv
```

3. Aktifkan virtual environment:

- ***Windows***: `.\venv\Scripts\activate`

- ***Mac/Linux***: `source venv/bin/activate`

Langkah 4: Install OpenCV

1. Di terminal yang sama, ketik:

```
bash
```

```
pip install opencv-python
```

Langkah 5: Buat File Python

1. Di dalam folder proyek, buat file Python baru (misalnya morphology.py).

Langkah 6: Implementasi Program

Masukkan kode berikut ke dalam morphology.py untuk melakukan operasi morfologi dasar (dilasi dan erosi) pada gambar yang diunggah.

python

```
import cv2
```

```
import numpy as np
```

Fungsi untuk menerapkan operasi dilasi

```
def apply_dilation(image, kernel_size=(5,5), iterations=1):
```

```
    kernel = np.ones(kernel_size, np.uint8)
```

```
    dilated_image = cv2.dilate(image, kernel, iterations=iterations)
```

```
    return dilated_image
```

Fungsi untuk menerapkan operasi erosi

```
def apply_erosion(image, kernel_size=(5,5), iterations=1):
```

```
    kernel = np.ones(kernel_size, np.uint8)
```

```
    eroded_image = cv2.erode(image, kernel, iterations=iterations)
```

```
    return eroded_image
```

Membaca gambar

```
image_path = 'path/to/your/image.jpg' # Ganti dengan path gambar Anda
```

```
image = cv2.imread(image_path, 0) # Membaca dalam grayscale Terapkan operasi morfologi
```

```
dilated_image = apply_dilation(image)
```

```
eroded_image = apply_erosion(image)
```

Tampilkan hasil

```
cv2.imshow("Original Image", image)
```

```
cv2.imshow("Dilated Image", dilated_image)
```

```
cv2.imshow("Eroded Image", eroded_image)
```

```
cv2.waitKey(0)
```

```
cv2.destroyAllWindows()
```

Langkah 7: Menjalankan Program

1. Pastikan file gambar yang ingin diolah telah disiapkan dan sesuaikan path gambar pada kode di atas.
2. Simpan perubahan di morphology.py.
3. Jalankan kode dengan perintah berikut di terminal VSCode:

```
bash
```

```
python morphology.py
```

Langkah 8: Melihat Hasil

Jika berhasil, tiga jendela akan muncul:

- ***Original Image***: Menampilkan gambar asli.
- ***Dilated Image***: Menampilkan hasil dilasi pada gambar.
- ***Eroded Image***: Menampilkan hasil erosi pada gambar.

Dengan aplikasi sederhana ini, kamu dapat mencoba dan memahami operasi morfologi citra seperti dilasi dan erosi.

Code

```
import cv2
import numpy as np
from tkinter import *
from tkinter import filedialog
from tkinter import messagebox
from PIL import Image, ImageTk
import os

# Inisialisasi root window Tkinter
root = Tk()
root.title("Operasi Morfologi - Dilasi & Erosi")
root.geometry("800x600")

# Variabel global
img = None
processed_img = None
```

```

# Fungsi untuk membuka gambar
def open_image():
    global img
    filepath = filedialog.askopenfilename(filetypes=[("Image files", ".png;.jpg;.jpeg;.bmp")])
    if not filepath:
        return

    img = cv2.imread(filepath, cv2.IMREAD_GRAYSCALE)
    _, img = cv2.threshold(img, 127, 255, cv2.THRESH_BINARY) # Mengubah menjadi
    biner

    display_image(img, original=True)

# Fungsi untuk menampilkan gambar pada GUI
def display_image(image, original=False):
    # Konversi gambar untuk ditampilkan
    image = Image.fromarray(image)
    image = image.resize((400, 400))
    photo = ImageTk.PhotoImage(image=image)

    # Tampilkan gambar asli atau hasil
    if original:
        panel1.config(image=photo)
        panel1.image = photo
    else:
        panel2.config(image=photo)
        panel2.image = photo

# Fungsi untuk menyimpan gambar
def save_image():
    global processed_img

```

```

if processed_img is None:
    messagebox.showerror("Error", "Tidak ada gambar untuk disimpan!")
    return

filepath = filedialog.asksaveasfilename(defaultextension=".png", filetypes=[("PNG files",
"*.png")])
if filepath:
    cv2.imwrite(filepath, processed_img)
    messagebox.showinfo("Sukses", "Gambar berhasil disimpan!")

# Fungsi untuk melakukan dilasi
def apply_dilation():
    global img, processed_img
    if img is None:
        messagebox.showerror("Error", "Buka gambar terlebih dahulu!")
        return

    kernel = np.ones((5, 5), np.uint8)
    processed_img = cv2.dilate(img, kernel, iterations=1)
    display_image(processed_img)

# Fungsi untuk melakukan erosi
def apply_erosion():
    global img, processed_img
    if img is None:
        messagebox.showerror("Error", "Buka gambar terlebih dahulu!")
        return

    kernel = np.ones((5, 5), np.uint8)
    processed_img = cv2.erode(img, kernel, iterations=1)
    display_image(processed_img)

```

```
# GUI Layout

panel1 = Label(root)
panel1.pack(side="left", padx=10, pady=10)

panel2 = Label(root)
panel2.pack(side="right", padx=10, pady=10)


button_frame = Frame(root)
button_frame.pack(side="bottom", pady=20)


# Tombol untuk membuka, menyimpan, dan operasi morfologi

open_button = Button(button_frame, text="Buka Gambar", command=open_image)
open_button.grid(row=0, column=0, padx=10, pady=10)


dilate_button = Button(button_frame, text="Dilasi", command=apply_dilation)
dilate_button.grid(row=0, column=1, padx=10, pady=10)


erode_button = Button(button_frame, text="Erosi", command=apply_erosion)
erode_button.grid(row=0, column=2, padx=10, pady=10)


save_button = Button(button_frame, text="Simpan Gambar", command=save_image)
save_button.grid(row=0, column=3, padx=10, pady=10)


root.mainloop()
```