**PENGOLAHAN CITRA DIGITAL**

**UJIAN TENGAH SEMESTER**

**FACE DETECTION**

A green shield with a logo

Description automatically generated

**NAMA : BETA NURUL AWWALIN**

**NIM : 21104410091**

**ANGKATAN/ KELAS : 2021 / B**

**JURUSAN TEKNIK INFORMATIKA**

**FAKULTAS TEKNOLOGI INFORMASI**

**UNIVERSITAS ISLAM BALITAR**

**2024**

**BAB I**

**PENDAHULUAN**

**Deteksi Wajah dengan OpenCV**

Deteksi wajah dengan OpenCV adalah sebuah teknik untuk menemukan dan memisahkan wajah manusia dalam gambar atau video. Teknik ini menggunakan algoritma pembelajaran mesin yang dilatih pada kumpulan data gambar wajah dan non-wajah untuk mengenali ciri-ciri khas wajah manusia.

**Bagaimana Cara Kerjanya?**

Pemuatan Gambar : Pertama, gambar atau video dimuat ke dalam program.

Konversi Warna : Gambar diubah dari format warna asli (misalnya, RGB) ke format

grayscale.

Deteksi Fitur : Algoritma deteksi wajah mencari fitur-fitur penting pada gambar,

seperti mata, hidung, dan mulut.

Klasifikasi : Fitur-fitur tersebut dianalisis dan dibandingkan dengan pola yang

telah dipelajari dari data pelatihan.

Penandaan Wajah : Jika pola yang cocok ditemukan, algoritma akan menandai area

gambar yang berisi wajah.

**Manfaat Deteksi Wajah:**

1. Pengenalan

Wajah Digunakan untuk mengidentifikasi orang dalam gambar atau video.

1. Pelacakan Wajah

Memungkinkan pelacakan pergerakan wajah dalam video.

1. Otentikasi Pengguna

Digunakan untuk memverifikasi identitas pengguna dalam sistem keamanan.

1. Editing Gambar

Membantu dalam pengeditan gambar, seperti menambahkan filter atau efek khusus pada wajah.

1. Analisis Emosi

Digunakan untuk menganalisis ekspresi wajah dan mendeteksi emosi.

**OpenCV:**

OpenCV (Open Source Computer Vision Library) adalah pustaka perangkat lunak sumber terbuka yang populer untuk pemrosesan gambar dan penglihatan komputer. Pustaka ini menyediakan berbagai fungsi untuk deteksi wajah, termasuk:

* Cascade Classifiers: Algoritma deteksi wajah klasik yang menggunakan fitur Haar untuk mengenali wajah.
* HOG (Histogram of Oriented Gradients): Algoritma deteksi wajah yang lebih modern yang menggunakan deskriptor gradien untuk mengenali wajah.
* Deep Learning: Algoritma deteksi wajah terbaru yang menggunakan jaringan saraf tiruan untuk mencapai akurasi yang lebih tinggi.

**BAB II**

**LANGKAH PENGERJAAN DAN PENJELASAN**

Berikut merupakan langkah - langkah menggunakan face detection opencv dengan visual studio code :

1. Mendownload

**BAB III**

**HASIL PENGERJAAN**

**3.1 Sourcode Python**

|  |
| --- |
| #mengimpor library opencv  import cv2  #membuat objek pendeteksi bernama 'face\_ref', 'face\_ref.xml' berisi model deteksi wajah  face\_ref = cv2.CascadeClassifier("face\_ref.xml")  #membuka kamera default komputer/laptop  camera = cv2.VideoCapture(0)  #mendefinisikan fungsi bernama "face\_detection" yg menerima frame gambar sebagai input  def face\_detection(frame):      #mengkonversi gambar dari format warna (RGB) menjadi grayscale [deteksi wajah berfungsi baik dengan gambar grayscale]      optimized\_frame = cv2.cvtColor(frame, cv2.COLOR\_RGB2GRAY)      #menggunakan 'face\_ref' untuk mendeteksi wajah dari gambar yg dikonversi grayscale      faces = face\_ref.detectMultiScale(optimized\_frame, scaleFactor=1.1)      #mengembalikan hasil deteksi wajah      return faces  #mendefinisikan fungsi bernama "frame"  def drawer\_box(frame):      #untuk memulai loop untuk menjalankan deteksi wajah      for x, y, w, h in face\_detection(frame):          #untuk menggambar sebuah kotak persegi panjang disekitar wajah yang terdeteksi          cv2.rectangle(frame, (x, y), (x +w, y+h), (255, 255, 255), 4)          #untuk menuliskan teks "Beta NA" dibawah kotak wajah yang terdeteksi          cv2.putText(frame, "Beta NA", (x , y + h + 50), cv2.FONT\_HERSHEY\_COMPLEX, 1, (255, 255, 255), 2)  #mendefinisikan fungsi untuk menutup jendela tampilan kamera  def close\_window():      camera.release()      #menutup semua jendela kamera      cv2.destroyAllWindows()      #keluar dari program python      exit()  #berisi loop utama program  def main():      while True:          #membaca frame gambar dari kamera disimpan ke variabel 'frame'          \_, frame = camera.read()          #memanggil fungsi "drawer\_box" untuk menggambar kotak dan text          drawer\_box(frame)          #menampilkan frame gambar dengan kotak dan teks berjudul "Beta\_FaceDetection"          cv2.imshow("Beta\_FaceDetection", frame)          #untuk memeriksa inputan user pada tombol 'x' akan menutup jendela          if cv2.waitKey(1) & 0xFF == ord('x'):              #berfungsi menutup jendela              close\_window()  #untuk memastikan kode dijalakan ketika python dijalankan berlangsung  if \_\_name\_\_ == '\_\_main\_\_':      #memanggil fungsi main untuk memulai loop utama program      main() |

**3.2 Hasil Percobaan**

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated