**LAPORAN**

**Visi Komputer dan Pengolahan Citra**

Tugas Stereo Vision



Disusun oleh:

Nama : Ahmada Haiz Zakiyil Ilahi

NRP : 1123800010

Kelas : 1 S2 Teknik Elektro

Dosen : Dr Setiawardhana ST, MT

**PROGRAM PASCASARJANA TERAPAN**

**POLITEKNIK ELEKTRONIKA NEGERI SURABAYA**

**2023/2024**

**TUGAS STEREO VISION**

1. **Program Kamera Minoru (minoru.py)**

* **Kode Program:**

|  |
| --- |
| import numpy as np import cv2  left\_camera = cv2.VideoCapture(4)  right\_camera = cv2.VideoCapture(6)  # id cam l = 4  # id cam r = 6  # Check if the cameras opened successfully  if not left\_camera.isOpened() or not right\_camera.isOpened():      print("Error: Unable to open one or both cameras")      exit()  # Buat objek StereoSGBM  stereo = cv2.StereoSGBM\_create(numDisparities=3, blockSize=15)  while True:       # Ambil frame dari kedua kamera      ret1, left\_frame = left\_camera.read()      ret2, right\_frame = right\_camera.read()      if not (ret1 and ret2):          break      # Konversi frame menjadi citra abu-abu      left\_gray = cv2.cvtColor(left\_frame, cv2.COLOR\_BGR2GRAY)      right\_gray = cv2.cvtColor(right\_frame, cv2.COLOR\_BGR2GRAY)      try:       # Hitung disparitas          disparity = stereo.compute(left\_gray, right\_gray) + 0.0000001      except cv2.error as e:          print(f"Error computing disparity: {e}")          continue      # Hitung jarak ke objek      # Faktor konversi yang bergantung pada konfigurasi kamera      depth = 1.0 / disparity       # Tampilkan citra kedua kamera dan citra kedalaman      cv2.imshow('Camera Laptop', left\_frame)      cv2.imshow('Camera HP', right\_frame)      cv2.imshow('Depth Map', depth)      if cv2.waitKey(1) & 0xFF == ord('q'):          break  left\_camera.release()  right\_camera.release()  cv2.destroyAllWindows() |

* **Analisa:**

Program ini merupakan implementasi pemrosesan citra stereo menggunakan OpenCV dan bahasa pemrograman Python. Dua kamera, yang diidentifikasi dengan indeks 4 dan 6, diinisialisasi untuk melakukan tangkapan citra secara simultan. Setiap iterasi loop, program mengonversi citra warna dari kedua kamera menjadi citra abu-abu untuk menyederhanakan proses pemrosesan. Menggunakan objek StereoSGBM dari OpenCV, program menghitung disparitas antara citra kiri dan kanan, dan hasilnya diubah menjadi citra kedalaman. Ada penanganan khusus untuk mengatasi kesalahan yang mungkin terjadi selama perhitungan disparitas. Citra dari kedua kamera serta citra kedalaman ditampilkan dalam tiga jendela terpisah. Program dapat dihentikan dengan menekan tombol 'q'. Setelah loop selesai, sumber daya kamera dilepaskan, dan semua jendela ditutup. Kode ini memberikan implementasi praktis dari konsep pemrosesan citra stereo untuk mendapatkan informasi kedalaman menggunakan alat pemrograman OpenCV dalam lingkungan bahasa pemrograman Python.

1. **Program Stereo Camera Disparity Map (stereo.py)**

* **Kode Program:**

|  |
| --- |
| import cv2 import numpy as np  # Open the left and right cameras (modify indices if needed)  left\_cap = cv2.VideoCapture(4)  right\_cap = cv2.VideoCapture(6)  # Check if the cameras opened successfully  if not left\_cap.isOpened() or not right\_cap.isOpened():      print("Error: Could not open one or both cameras.")      exit()  # Set up stereo block matching parameters (adjust as needed)  stereo = cv2.StereoBM\_create(numDisparities=16, blockSize=5)  while True:      # Read frames from both cameras      ret1, left\_frame = left\_cap.read()      ret2, right\_frame = right\_cap.read()      # Check if both frames were read successfully      if not ret1 or not ret2:          print("Error: Could not read frames from one or both cameras.")          break      # Convert frames to grayscale      left\_gray = cv2.cvtColor(left\_frame, cv2.COLOR\_BGR2GRAY)      right\_gray = cv2.cvtColor(right\_frame, cv2.COLOR\_BGR2GRAY)      # Compute disparity map      disparity = stereo.compute(left\_gray, right\_gray)      # Normalize disparity values for better visualization      disparity = cv2.normalize(disparity, None, 0, 255, cv2.NORM\_MINMAX)      disparity = disparity.astype(np.uint8)      # Display the disparity map      cv2.imshow('Disparity Map', disparity)      cv2.imshow('left frame', left\_frame)      cv2.imshow('right frame', right\_frame)      # Exit the loop if the 'q' key is pressed      if cv2.waitKey(1) & 0xFF == ord('q'):          break  # Release both cameras and close the window  left\_cap.release()  right\_cap.release()  cv2.destroyAllWindows() |

* **Analisa:**

Program ini adalah implementasi pencocokan stereo menggunakan OpenCV dan Python, di mana dua kamera (dengan indeks 4 dan 6) digunakan untuk merekam citra sepanjang waktu. Setiap iterasi loop, citra dari kedua kamera diambil, dikonversi ke citra grayscale, dan disparitas antara keduanya dihitung menggunakan metode StereoBM. Disparitas hasil kemudian dinormalisasi untuk visualisasi yang lebih baik, dan citra kedalaman yang dihasilkan ditampilkan dalam jendela terpisah. Program memberikan kontrol untuk menghentikan eksekusi dengan menekan tombol 'q'. Setelah selesai, sumber daya kamera dilepaskan, dan semua jendela ditutup. Kode ini menyajikan implementasi praktis dari teknik pemrosesan citra stereo untuk memperoleh citra kedalaman menggunakan OpenCV dan mengilustrasikan parameter yang dapat disesuaikan untuk pencocokan stereo.