

[PCD2025_2C_D3_2023]_ Tubes Eksplorasi

PENGEMBANGAN SISTEM PENGENALAN WAJAH DAN
DETEKSI SUKU MENGGUNAKAN COMPUTER VISION

D3 TEKNIK INFORMATIKA
JURUSAN TEKNIK KOMPUTER DAN
INFORMATIKA POLITEKNIK NEGERI BANDUNG



Daftar Isi

Daftar Isi.....	2
PENDAHULUAN	3
1. Deskripsi Aplikasi	3
2. Fitur Utama	3
3. Fungsi dan Manfaat di Kehidupan Nyata	3
4. Algoritma yang Digunakan.....	4
5. Struktur Proyek dan Fungsi.....	6
6. Hasil Eksplorasi	6
7. Kendala yang Dihadapi dan Solusi	7
8. Rekomendasi Pengembangan	7
9. Aplikasi Dijalankan	7

PENDAHULUAN

1. Deskripsi Aplikasi

Aplikasi People Flow dirancang untuk mendeteksi dan menghitung jumlah orang yang masuk dan keluar dari suatu ruangan menggunakan webcam secara real-time. Aplikasi ini memungkinkan pengguna untuk menetapkan batas kapasitas ruangan atau menggunakannya tanpa batas, dengan notifikasi saat ruangan penuh dan laporan per menit tentang aktivitas masuk/keluar.

2. Fitur Utama

1. Deteksi orang menggunakan MediaPipe Pose untuk pelacakan pose tubuh.
2. Penghitungan orang yang masuk (bergerak ke kanan melintasi garis vertikal) dan keluar (bergerak ke kiri).
3. Antarmuka Streamlit untuk pengaturan kapasitas, tombol mulai/berhenti, dan tampilan video real-time.
4. Notifikasi "Ruangan Penuh" saat kapasitas maksimum tercapai.
5. Laporan per menit (format jj:mm) dalam format CSV yang mencatat jumlah masuk dan keluar hanya saat ada perubahan.
6. Visualisasi garis vertikal imajiner untuk deteksi perlintasan.

3. Fungsi dan Manfaat di Kehidupan Nyata

Fungsi

1. Monitoring Kapasitas Ruangan: Aplikasi ini memungkinkan pengelola fasilitas untuk memantau jumlah orang dalam ruangan secara real-time, dengan opsi batas kapasitas untuk mencegah overcrowding.
2. Pencatatan Aktivitas: Menyediakan laporan per menit untuk analisis pola masuk/keluar, berguna untuk pengelolaan waktu dan sumber daya.
3. Notifikasi Otomatis: Memberikan peringatan saat ruangan penuh, membantu menjaga kepatuhan terhadap batas kapasitas.

Manfaat

1. Keamanan dan Kepatuhan: Digunakan di tempat umum seperti mal, museum, atau ruang acara untuk memastikan jumlah pengunjung tidak melebihi batas aman, terutama dalam konteks kesehatan (misalnya, pembatasan pasca-pandemi).
2. Efisiensi Operasional: Membantu pengelola gedung atau acara mengoptimalkan alur pengunjung, misalnya dengan menyesuaikan jadwal staf berdasarkan laporan aktivitas.
3. Analisis Data: Laporan per menit dapat digunakan untuk studi perilaku pengunjung, seperti waktu puncak kunjungan, untuk perencanaan strategis.
4. Kemudahan Penggunaan: Antarmuka Streamlit yang sederhana memungkinkan pengguna non-teknis untuk mengoperasikan aplikasi dengan mudah.

4. Algoritma yang Digunakan

Aplikasi ini menggabungkan beberapa algoritma untuk deteksi, pelacakan, dan pengelolaan data:

1. Deteksi Pose (MediaPipe Pose):

Deskripsi: Menggunakan model machine learning dari MediaPipe untuk mendeteksi landmark tubuh (misalnya, bahu, pinggul) dalam frame video webcam.

a. Proses:

- Input: Frame RGB dari webcam.
- Output: Koordinat landmark tubuh (x, y) dalam skala normalisasi [0, 1].
- Parameter: min_detection_confidence=0.7, min_tracking_confidence=0.7, model_complexity=1.

b. Tujuan: Mengidentifikasi posisi seseorang dalam frame.

2. Pelacakan Centroid:

Deskripsi: Menghitung centroid (pusat massa) seseorang berdasarkan rata-rata koordinat bahu kiri dan pinggul kiri.

a. Proses:

- Konversi koordinat normalisasi ke piksel berdasarkan dimensi frame.
- Simpan centroid (cx, cy) untuk setiap orang yang terdeteksi.
- Gunakan jarak Euclidean untuk mencocokkan centroid baru dengan ID yang sudah ada (threshold: 100 piksel).

b. Tujuan: Melacak pergerakan individu untuk menghitung perlintasan garis.

3. Deteksi Perlintasan Garis Vertikal:

Deskripsi: Menentukan apakah seseorang masuk atau keluar berdasarkan pergerakan centroid melintasi garis vertikal imajiner (line_x=400).

a. Proses:

- Bandingkan posisi x centroid saat ini (cx) dengan posisi sebelumnya (prev_x). Jika prev_x <= line_x dan cx > line_x, tambah count_in (masuk).
- Jika prev_x >= line_x dan cx < line_x, tambah count_out (keluar).
- Periksa batas kapasitas sebelum menambah count_in.

b. Tujuan: Menghitung jumlah masuk dan keluar secara akurat.

4. Pengelolaan Kapasitas:

Deskripsi: Memeriksa jumlah orang saat ini (count_in - count_out) terhadap batas kapasitas (max_capacity).

a. Proses:

- Jika max_capacity ditentukan dan jumlah orang mencapai batas, set is_full=True dan tampilkan notifikasi.
- Update status: "Ruang Penuh", "Ruang Kosong", atau "Sisa Kapasitas".

b. Tujuan: Mengontrol overcrowding dan memberikan notifikasi.

5. Pencatatan Laporan per Menit:

Deskripsi: Mencatat perubahan masuk/keluar dalam format jj:mm hanya saat ada perubahan.

a. Proses:

- Simpan selisih count_in dan count_out saat berubah, dengan timestamp datetime.now().strftime("%H:%M").
- Agregasi data per menit saat menyimpan ke CSV.

b. Tujuan: Menyediakan laporan ringkas untuk analisis.

5. Struktur Proyek dan Fungsi

```
people_flow/
    ├── app.py
    ├── detection.py
    ├── data/
        └── reports/
    └── requirements.txt
└── README.md
```

Deskripsi File

1. app.py:

- Fungsi Utama: Menyediakan antarmuka pengguna dengan Streamlit, mengelola alur aplikasi, dan menyimpan laporan.
- Fungsi Penting:
 - save_report(data): Mengagregasi data per menit dan menyimpan ke CSV.
 - Logika utama: Menginisialisasi webcam, memanggil process_frame dari detection.py, memperbarui UI, dan mencatat perubahan masuk/keluar.
- Komponen UI:
 - Sidebar: Pilihan batas kapasitas, input kapasitas maksimum, tombol mulai/berhenti.
 - Main: Tampilan video, status, warning "Ruang Penuh", dan laporan.

2. detection.py:

- Fungsi Utama: Menangani deteksi pose, pelacakan centroid, dan penghitungan masuk/keluar.
- Fungsi Penting:
 - __init__(max_capacity): Inisialisasi MediaPipe Pose dan variabel pelacakan.
 - get_centroid(landmarks, frame_shape): Menghitung centroid dari landmark bahu dan pinggul.
 - process_frame(frame): Memproses frame untuk deteksi pose, pelacakan, perlintasan garis, dan status kapasitas. Mengembalikan frame, count_in, count_out, status, dan is_full.

3. requirements.txt:

- Berisi dependensi: streamlit==1.38.0, opencv-python==4.10.0.84, mediapipe==0.10.14, numpy==1.26.4, pandas==2.2.2, scipy==1.14.1.

4. data/reports/:

- Menyimpan laporan CSV dengan format: Waktu (jj:mm), Masuk, Keluar.

6. Hasil Eksplorasi

• Deteksi dan Pelacakan:

- MediaPipe Pose berhasil mendeteksi pose satu orang per frame dengan akurasi tinggi pada pencahayaan yang baik.
- Centroid berdasarkan bahu dan pinggul memberikan pelacakan yang stabil untuk perlintasan garis vertikal.
- Penghitungan masuk/keluar akurat untuk skenario satu orang melewati garis.

• Antarmuka Pengguna:

- Streamlit menyediakan UI yang intuitif, dengan sidebar untuk pengaturan dan tampilan real-time untuk video, status, dan laporan.
- Warning "Ruang Penuh" dengan st.error dan emoji ! cukup menonjol untuk menarik perhatian.

• Laporan:

- Laporan per menit hanya mencatat perubahan masuk/keluar, menghasilkan file CSV yang ringkas dan mudah dianalisis.
 - Format jj:mm memudahkan analisis pola waktu.
- **Kapasitas:**
- Logika batas kapasitas bekerja dengan baik, mencegah count_in melebihi max_capacity dan menampilkan status yang relevan.

7. Kendala yang Dihadapi dan Solusi

1. **Kendala:** Deteksi landmark berulang untuk orang besar dalam frame, menyebabkan penghitungan ganda.
 - **Solusi:** Menghitung centroid dari bahu dan pinggul untuk merepresentasikan satu orang, dan meningkatkan min_detection_confidence ke 0.7 untuk menyaring deteksi lemah.
2. **Kendala:** Laporan CSV mencatat data setiap frame, meskipun tidak ada perubahan.
 - **Solusi:** Mencatat data hanya saat count_in atau count_out berubah, dengan pelacakan prev_count_in dan prev_count_out. Data diagregasi per menit saat disimpan.
3. **Kendala:** MediaPipe Pose hanya mendeteksi satu orang per frame, sedangkan kebutuhan awal adalah hingga tiga orang.
 - **Solusi:** Menggunakan pelacakan centroid berbasis riwayat frame untuk mensimulasikan deteksi multi-orang, meskipun dibatasi oleh API Python MediaPipe. Pilihan akhir kembali ke deteksi satu orang untuk stabilitas.
4. **Kendala:** Parameter max_num_poses tidak didukung di MediaPipe Pose Python.
 - **Solusi:** Menghapus parameter tersebut dan menggunakan pendekatan berbasis clustering centroid untuk melacak hingga tiga orang, meskipun kembali ke deteksi satu orang untuk stabilitas.
5. **Kendala:** Pengaturan kapasitas tidak fleksibel, dengan input max_capacity selalu muncul.
 - **Solusi:** Mengganti checkbox dengan radio button untuk memilih "Ada Batas Maksimal" atau "Tidak Ada Batas Maksimal", menampilkan input hanya saat diperlukan.

8. Rekomendasi Pengembangan

- **Multi-Person Detection:**
 - Beralih ke model seperti YOLOv5 atau YOLOv8 dengan DeepSORT untuk deteksi dan pelacakan multi-orang (hingga 10 orang per frame).
 - Tambahkan bounding box untuk visualisasi setiap orang, meningkatkan kejelasan pelacakan.

9. Aplikasi Dijalankan

Pengaturan

Batas Kapasitas

- Tidak Ada Batas Maksimal
- Ada Batas Maksimal

Kapasitas Maksimum

2

- +

Mulai

Berhenti



Menentukan kapasitas ruangan

Orang yang terdeteksi akan diberikan ID



Masuk: 2 | Keluar: 0 | Status: Ruangan Penuh

⚠ RUANGAN PENUH! Harap tunggu hingga ada orang keluar.

Pesan warning saat ruangan penuh

RUNNING... Stop Deploy

People Flow - Sistem Penghitung Orang

	Waktu (jj:mm)	Masuk	Keluar
0	09:31	1	0
1	09:32	3	2

Laporan

Laporan orang masuk/keluar