

BAB 3 DESAIN DAN PEMILIHAN DATA

Pada bab ini menjelaskan mengenai desain sistem dan pemilihan data yang akan dijadikan objek.

3.1 DESAIN SISTEM

Bagian ini menjelaskan mengenai deteksi objek kepala, penjejakan objek kepala, dan penghitungan objek kepala.

Sistem diharapkan dapat mengenali pengunjung dari hasil rekaman kamera CCTV maupun secara *realtime*. Kamera akan ditempatkan pada bagian atas pintu masuk pusat keramaian. Data video dari kamera CCTV akan dijadikan input dari sistem, selanjutnya akan dideteksi apakah terdapat objek kepala, lalu dilakukan penjejakan, dan selanjutnya dilakukan penghitungan.

3.1.1 Deteksi Objek Kepala

Sebelum dilakukan penghitungan, perlu dilakukan penjejakan terhadap pengunjung. Penjejakan dapat dilakukan bila posisi pengunjung dapat diketahui dari tampilan kamera. Sedangkan posisi pengunjung dapat diketahui dengan melakukan pendeteksian objek kepala pengunjung.

Dalam melakukan pendeteksian objek kepala, digunakan basis data yang dilatih secara tersendiri. Proses pelatihan akan menghasilkan basis data berupa *file xml* yang berisi informasi ciri objek kepala.

3.1.2 Penjejakan Objek Kepala

Setelah posisi objek kepala diketahui, perlu dilakukan penjejakan pergerakan pengunjung untuk didapatkan kriteria apakah pengunjung memasuki pusat keramaian atau tidak dalam proses penghitungan pengunjung nantinya.

Penjejakan objek kepala dilakukan dengan mencatat posisi pengunjung ke dalam *array* seperti gambar berikut:

Tabel 3. 1 Ilustrasi Array yang Mencatat Objek Kepala Pengunjung (telah diolah kembali)[15]

	Frame 1	Frame 2	Frame 3
Pengunjung 1	(x,y) = (3,4) Radius = 22	(x,y) = (4,5) Radius = 25	?
Pengunjung 2	(x,y) = (100,100) Radius = 32	(x,y) = (104,107) Radius = 29	?
Pengunjung 3	(x,y) = (204,103) Radius = 44	(x,y) = (209,120) Radius = 51	?

Frame saat ini

(210,152)
Radius = 59

Pengukuran
Jarak Euclidian

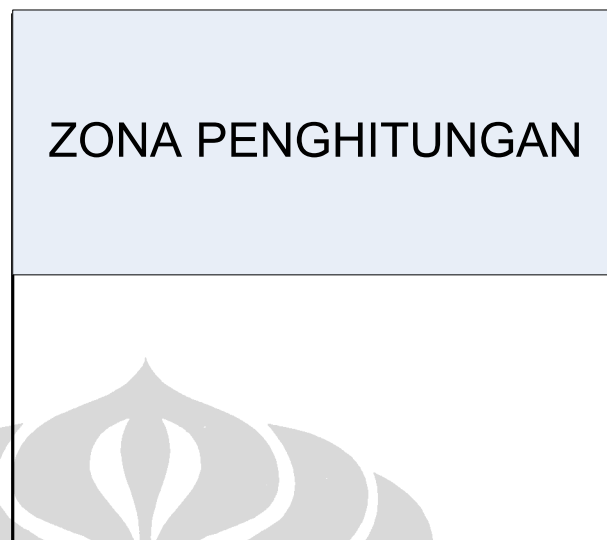
Pengunjung akan tercatat sebagai pengunjung 1, 2 atau 3 berdasarkan pada jarak paling minimum / metode jarak *Euclidian* antara posisi pengunjung pada tiap *frame*. Misalnya pengunjung 1 terdeteksi pada koordinat (3,4), maka pada *frame* selanjutnya pengunjung 1 akan didefinisikan sebagai objek kepala yang terdeteksi dengan jarak paling minimum dengan jarak *frame* sebelumnya, yaitu pada koordinat (4,5).

3.1.3 Penghitungan Pengunjung

Seorang pengunjung didefinisikan memasuki suatu pusat keramaian bila memenuhi kriteria berikut[15]:

1. Objek kepala terdeteksi pada 4 *frame* sebelumnya
Untuk memastikan bahwa objek yang terdeteksi bukanlah data *false* positif, atau yang bukan objek kepala, tetapi terdeteksi sebagai objek kepala.
2. Objek kepala telah menghilang keberadaannya dari *frame* saat ini
Untuk memastikan bahwa pengunjung telah memasuki pusat keramaian.
3. Objek kepala berada tepat pada satu *frame* sebelumnya di dalam zona penghitungan.

Untuk memastikan bahwa objek kepala benar-benar telah pernah melewati zona penghitungan. Posisi zona penghitungan dibatasi oleh sebuah garis lurus yang akan diposisikan pada pintu masuk suatu pusat keramaian.



Gambar 3. 1 Letak Zona Penghitungan

3.2 PEMILIHAN DATA

Pada bagian ini menjelaskan mengenai pemilihan objek kepala, pengumpulan data, dan pelatihan data.

3.2.1 Pemilihan Objek Kepala

Sebelumnya telah dijelaskan bahwa, untuk melakukan penghitungan pengunjung, perlu dilakukan penjejukan pengunjung. Dalam melakukan penjejukan pengunjung perlu diketahui posisi pengunjung setiap *frame*. Selanjutnya untuk mengetahui posisi pengunjung perlu dilakukan dengan mendeteksi pengunjung. Dalam mendeteksi pengunjung perlu dipilih objek yang lebih baik dari wajah. Selain itu posisi pendeteksian juga akan berpengaruh pada perbaikan kinerja penghitungan.

Karena pendeteksian wajah dari depan tidak mampu mentoleransi pengunjung yang masuk dalam satu baris lurus maupun pengunjung yang menoleh ke arah lain selain kamera. Maka dicarilah sebuah metode baru untuk memperbaikinya. Diusulkan sebuah metode penghitungan pengunjung dengan objek kepala sebagai parameter dikenalnya seorang pengunjung.

Objek kepala dipilih sebagai parameter karena berbagai kelebihanannya dibandingkan wajah sebagai berikut:

1. Objek kepala dikenali dari atas memiliki jangkauan yang lebih luas terhadap posisi pengunjung.
2. Dengan memilih objek kepala yang dikenali secara vertikal sebagai parameter pengenalan, maka kemungkinan pengunjung memutar wajahnya ke arah manapun secara horizontal atau dengan toleransi 5 derajat dapat ditoleransi oleh sistem.

3.2.2 Pengumpulan Data

Sebelum dilakukan pelatihan data, perlu dikumpulkan data terlebih dahulu. Setelah data terkumpul, maka akan dilakukan pelatihan menggunakan *haar-like features* dan *boosting*. Pelatihan ini dapat disebut dengan *haartraining*[8, 10, 12]. *Haartraining* dibentuk oleh dua komponen utama yaitu *haar-like features* dan *boosting*.

Dalam pelatihan dibutuhkan data positif dan data negatif. Dalam pelatihan menggunakan *haartraining* digunakan 1458 data positif yang diambil sesuai dengan batasan ruang lingkup, yaitu pria dengan rambut normal (1cm-5cm), diambil satu putaran penuh. Jenis rambut ikal atau biasa dan tidak terlalu keriting atau kribu. Sedangkan data negatif diambil dari data di internet[9]. Jumlah data negatif yang digunakan adalah 3000. Jumlah yang dibutuhkan untuk hasil yang bagus adalah 7000 data positif[9, 13]. Jumlah data positif dan data negatif dipilih penulis karena dianggap sudah dapat mewakili jenis rambut yang didefinisikan.

Proses pemotongan data positif dilakukan secara manual dengan bantuan Microsoft Office Picture Manager 2003. Berikut adalah proses pemotongan gambar yang dilakukan.



Gambar 3. 2 Contoh pemotongan gambar data positif

Gambar di sebelah kiri adalah data pelatihan asli, sedangkan yang disebelah kanan adalah data pelatihan yang telah dipotong. Contoh hasil proses pemotongan gambar dapat di lihat di lampiran 5.

3.2.3 Pelatihan Data

Seperti yang telah dijelaskan sebelumnya pelatihan data dilakukan dengan menggunakan *haar-like features* dan *boosting*. Proses pelatihan data dilakukan sebanyak tujuh kali hingga mencapai hasil yang diharapkan.

Berikut adalah tahapan-tahapan proses pelatihan data yang dilakukan [12, 8, 10, 13]:

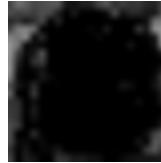
1. Mendata data nama-nama data negatif dan menyimpannya di dalam sebuah *file* dalam format *dat*. Perintah yang dilakukan dalam konsol cygwin adalah sebagai berikut: `$ find negatives/ -name '*.jpg' > negatives.dat`
2. Mendata data nama-nama data positif dan menyimpannya di dalam sebuah *file* dalam format *dat*. Perintah yang dilakukan dalam konsol cygwin adalah sebagai berikut: `$ find positives/ -name '*.png' > positives.dat`
3. Pembuatan data sampel pelatihan positif menggunakan *createsample.pl*. Pada step ini objek yang telah ditandai dalam bounding *rectangle* disimpan ke dalam *file* *.vec*. Perintah yang dilakukan dalam konsol cygwin adalah sebagai berikut:
`$ perl createtrainsamples.pl positives.dat negatives.dat samples`

```
$ find samples/ -name '*.vec' > samples.dat
```

```
$ mergevec samples.dat samples.vec
```

```
$ createsamples -vec samples.vec -show -w 20 -h 20
```

Berikut adalah contoh sampel data pelatihan positif untuk objek kepala:



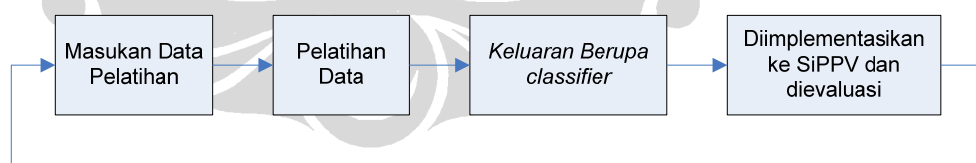
Gambar 3. 3 Contoh Hasil Pembuatan Sampel Positif

5. Pelatihan menggunakan *haartraining.cpp*. Pada konsol Microsoft Windows Command Prompt dilakukan perintah sebagai berikut:

```
haartraining -data trainout -vec samples.vec -bg negatives.dat -nstages 20 -
nsplits 2 -minhitrate 0.999 -maxfalsealarm 0.5 -npos 2009 -neg 3000 -w 20 -h
20 -mem 1024 -mode ALL
```

Seperti yang sudah dijelaskan sebelumnya proses pelatihan akan menghasilkan sebuah *cascade classifier* untuk objek kepala berupa *file xml* yang akan digunakan oleh SiPPV nantinya.

Proses pelatihan dilakukan berulang-ulang sesuai dengan kebutuhan hasil *classifier* yang diharapkan. Lama pelatihan untuk sekali pelatihan lebih kurang 2x24jam. Proses pelatihan data untuk SiPPV telah dilakukan lima kali. Berikut adalah contoh skema pelatihan data:



Gambar 3. 4 Siklus Pelatihan Data