

# PROGRAM PENDETEKSI DAN PENGHITUNG JUMLAH PENGUNJUNG DENGAN PENGOLAHAN CITRA DIGITAL MENGGUNAKAN *WEBCAM*

Wirawan Hidayat Pembimbing Ir. Muhammad Aswin, MT. dan Adharul Muttaqin, ST., MT.

**Abstrak.** Sampai saat ini untuk mendapatkan data jumlah pengunjung umumnya masih menggunakan cara manual. Cara manual ini kurang efektif karena kemungkinan terjadinya *error* cukup tinggi baik yang disebabkan oleh *human error* maupun oleh alat itu sendiri (*device error*).

Salah satu perkembangan teknologi sebagai solusi yang bisa digunakan untuk mengatasi masalah tersebut adalah memanfaatkan salah satu ilmu komputer yaitu pencitraan digital. Apabila pengambilan data dilakukan dengan kamera, lalu diproses dengan PC (Personal Computer), maka jalannya proses

Data jumlah pengunjung suatu tempat perbelanjaan sangat penting untuk mengetahui seberapa besar minat pengunjung untuk mengunjungi tempat tersebut. Data-data tersebut selalu update setiap hari untuk dijadikan trend grafik jumlah pengunjung. Selain itu, data tersebut dapat menjadi prediksi akan perkembangan suatu tempat atau pusat perbelanjaan mulai dari renovasi hingga reinovasi.

Masalah yang diangkat dalam skripsi ini adalah bagaimana mengambil data agar diperoleh jumlah pengunjung menggunakan pengolahan citra digital, kemudian bagaimana cara mendeteksi adanya suatu objek, serta bagaimana cara menghitung jumlah pengunjung. Beberapa hal yang menjadi batasan dalam pembuatan program ini adalah pengunjung adalah manusia dan yang dihitung hanya yang melewati *detection window* yang telah ditentukan sebelumnya, posisi webcam tetap (*static*) dan berada di atas pintu masuk tegak lurus 90° terhadap bidang latar dan yang dihitung adalah obyek tunggal.

## II. PEMBAHASAN

Blok diagram sistem menunjukkan garis besar kinerja program dalam mendeteksi dan menghitung jumlah pengunjung.

akan jadi lebih efisien, karena tidak memerlukan banyak tenaga untuk pengambilan data

Skripsi ini dibuat sebuah program pendeteksi dan penghitung jumlah pengunjung dengan pencitraan digital menggunakan webcam. Image dari webcam kemudian diambil selisih piksel antara background image dan frame image kemudian dikuadratkan dan dibagi luasan *detection window*.

**Kata kunci:** webcam, penghitungan pengunjung, metode penghitungan, *detection window*

## I. PENDAHULUAN



Gambar 2.1 Blok diagram sistem

( Sumber : Perancangan )

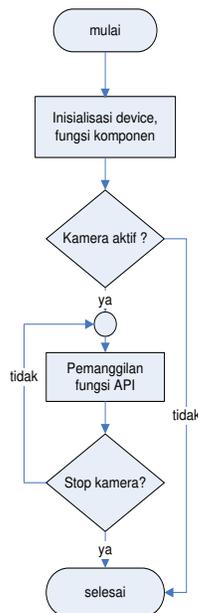
Cara kerja dari program pendeteksi dan penghitung jumlah pengunjung dengan teknik pengolahan citra digital menggunakan *webcam* yaitu dimulai dengan langkah pertama yang harus dilakukan adalah mengambil gambar *background* yang akan dipakai sebagai acuan pemisahan objek. Pengambilan *background* dilakukan dengan cara menangkap *image* dari *video* input sebelum dilakukan perekaman. Perlu diperhatikan bahwa posisi *webcam* berada di atas pintu masuk dan tegak lurus 90° terhadap bidang latar. Dengan demikian diharapkan *background* yang didapat mewakili *frame* kosong yang paling sesuai dengan kondisi terakhir sebelum dilakukan perekaman *webcam*. Setelah mendapatkan *background* yang diinginkan, langkah berikutnya adalah menentukan lingkup atau batasan pendeteksian pada *background* dan *frame* atau dapat disebut dengan istilah *detection window*. *Detection window* pada *background* dan *frame input* diletakkan pada posisi yang sama. Hal ini dilakukan dengan tujuan untuk membatasi daerah yang akan diproses. Proses pendeteksian dan perhitungan

dilakukan dengan cara mencari perbedaan nilai piksel yang terdapat pada *detection window* dan dibandingkan dengan nilai *threshold* .

### 2.1 Inisialisasi Webcam

Inisialisasi *webcam* dilakukan dengan memanfaatkan fitur komponen VFW ( *Video for Windows*) sebagai antar muka perangkat keras. VFW merupakan teknologi pertama kali yang dikeluarkan Windows untuk teknologi *video capture* dalam bentuk API (*Application Programming Interface*), yang tersimpan dalam library “*avicap32.dll*”. VFW menghubungkan rangkaian antara *camera device* hingga ke program aplikasi.

Pemanggilan Fungsi API tersebut melalui beberapa tahapan yaitu tahap pertama adalah membaca semua driver kamrea yang terpasang kemudian memberikan penomoran pada setiap driver yang telah dibaca.. Sebelum dilakukan pembacaan *image* hasil tangkapan dari *webcam* , terlebih dahulu disediakan *buffer* untuk penyimpanan data *image* sementara. Setiap *video capture* berhasil menangkap satu gambar (satu *frame*), maka data *frame* tersebut akan disimpan dalam suatu *buffer* yang terdapat pada API *video capture*. Setelah itu API akan mengirimkan sinyal *trigger* yang akan mengaktifkan suatu *event* selanjutnya.



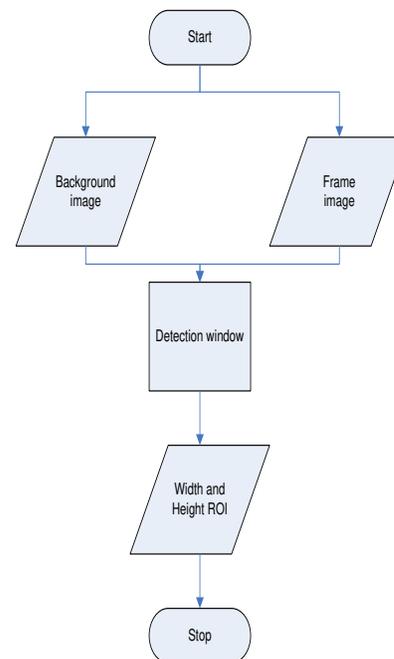
Gambar 2.3 Flowchart Inisiaalisasi *webcam*

### 2.3 Background Capture dan Frame Capture

Proses *background capture* dan *frame capture* merupakan proses awal sebuah pengolahan citra digital. Proses *background capture* dan *frame capture* sebenarnya menggunakan proses yang sama namun yang membedakan adalah jumlah *image* yang ditangkap oleh *webcam* . pada proses *background capture*, *image* yang ditangkap hanya satu saja karena digunakan untuk patokan atau referensi pada proses selanjutnya. Sedangkan pada proses *frame capture*, *image* yang ditangkap adalah *image* yang ”berjalan” seiring dengan *frame* yang berhasil ditangkap dengan skala waktu tertentu.

### 2.4 Detection Window

*Input* merupakan hasil dari proses *background capture* dan proses *frame capture* yang berupa *image*. *Image* tersebut kemudian ditentukan *detection window* pada daerah tertentu yang ditentukan sendiri oleh user. Penentuan letak atau posisi dari *detection window* pada *background image* dan *frame image* harus sama agar memudahkan dalam proses perhitungan piksel.



Gambar 2.5 Flowchart *detection window*  
(Sumber: Perancangan )

Setelah didapatkan letak *detection window* pada kedua *image* tersebut, didapatkan lebar dan tinggi

ROI ,nilai dari tiap koordinat dari *detection window* dan juga jumlah piksel.

Untuk mengubah citra berwarna yang mempunyai nilai matrik masing-masing r, g dan b menjadi citra *grayscale* dengan nilai s, maka konversi dapat dilakukan dengan mengambil rata-rata dari nilai r, g dan b sehingga dapat dituliskan menjadi:

$$I(i,j) = \frac{R(i,j) + G(i,j) + B(i,j)}{3}$$

## 2.5 Perhitungan Piksel

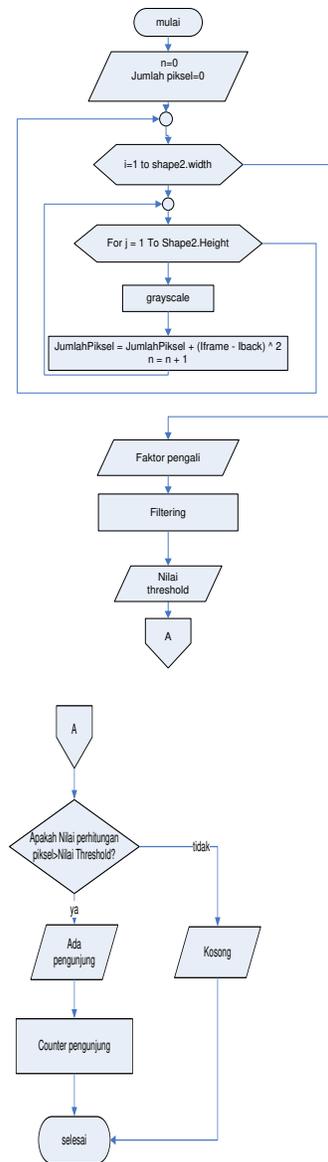
Hasil dari prose *detection window* dan *grayscale* kemudian dilakukan proses perhitungan piksel. Untuk mendapatkan hasil perhitungan, dilakukan menggunakan metode yang telah ditentukan sebelumnya. Proses perhitungan berdasarkan pada persamaan berikut:

$$Nilai = \frac{\sum_{i=a}^c \sum_{j=b}^d (F_{i,j} - B_{i,j})^2}{N}$$

Dimana:

Nilai = Hasil perhitungan  
 F (i,j) = nilai piksel pada *frame*  
 B (i,j) = nilai piksel pada *background*  
 N = jumlah piksel pada *detection window*

*window*

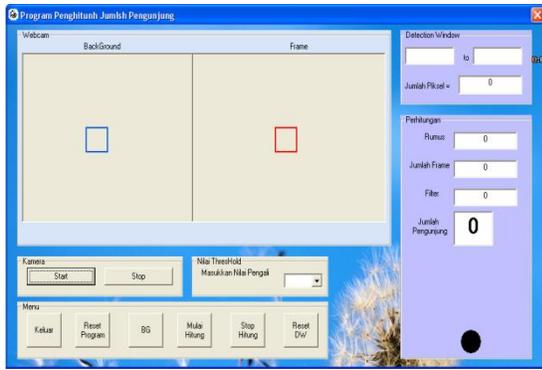


Gambar 2.7 Proses perhitungan piksel

(Sumber: Perancangan)

## 2.6 Implementasi

Program pendeteksi dan penghitung jumlah pengunjung dengan teknik pencitraan digital menggunakan *webcam* memiliki tampilan utama seperti pada gambar2.8



Gambar 2.8 Tampilan form utama

(Sumber : Perancangan )

Berikut akan dijelaskan mengenai komponen komponen yang digunakan dalam program pendeteksi dan penghitung jumlah pengunjung sesuai implementasi antarmuka yang ditunjukkan pada Gambar 4.14. Komponen yang digunakan meliputi picture box, command button, frame, label , shape, combo box, serta text box.

**III. PENGUJIAN**

**3.1 Pengujian Koneksi Webcam**

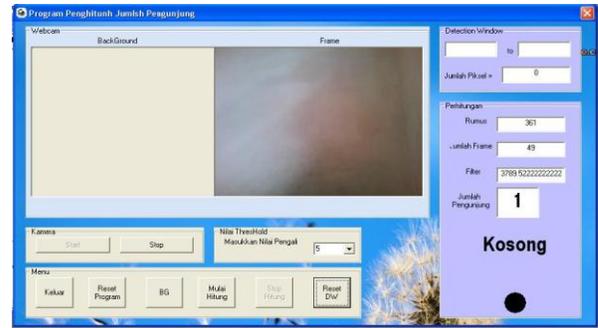
Pengujian ini bertujuan untuk mengetahui apakah webcam dapat terkoneksi dengan laptop yang menandakan bahwa komponen VFW(Video for Window) dapat berkerja dengan baik. Pengujian koneksi webcam dilakukan secara langsung ketika program dijalankan. Hasil pengujian ditunjukkan pada table 3.1

Tabel 3.1 Data Hasil Pengujian

No	Device	Hasil
1	Acer Crystal Eye	Berhasil Terhubung

(Sumber : Pengujian Aplikasi)

Table diatas adalah hasil pengujian koneksi webcam terhadap laptop. Dari hasil pengujian didapatkan hasil bahwa komponen VFW (Viedeo for Window) bekerja dengan baik. Sehingga webcam dapat terhubung.



Gambar 3.1 Koneksi webcam

(Sumber : Pengujian )

**3.2 Pengujian Program Pendeteksi dan Penghitung Jumlah Pengunjung Dengan Kondisi Cahaya Agak Gelap**

Pengujian program pendeteksi dan penghitung jumlah pengunjung dilakukan pada kondisi cahaya yang agak gelap dengan tujuan agar mengetahui respon program terhadap kondisi tersebut.

**3.3 Pengujian Program Pendeteksi dan Penghitung Jumlah Pengunjung Dengan Kondisi Cahaya Normal**

Pengujian program pendeteksi dan penghitung jumlah pengunjung ini dilakukan pada kondisi cahaya yang normal dengan tujuan adar mengetahui respon program terhadap kondisi tersebut. Hasil pengujian program seperti pada table 3.3

Table 3.3 Hasil pengujian dalam kondisi cahaya terang

N o	Jumlah sebenarnya	Jumla h terhitung	% keberhasi lan
1	20	20	100%
2	20	20	100%
3	20	20	100%
4	20	20	100%
Rata-rata			100%

(Sumber : Pengujian )

**3.4 Pengujian Program Pendeteksi dan Penghitung Jumlah Pengunjung Dengan Kondisi Cahaya Terang**

Pengujian program pendeteksi dan penghitung jumlah pengunjung dilakukan pada kondisi cahaya yang terang dengan tujuan agar mengetahui respon program terhadap kondisi tersebut. Hasil pengujian program seperti pada table 3.4

Table3.4 Hasil pengujian dalam kondisi cahaya terang

No	Jumlah sebenarnya	Jumlah terhitung	% keberhasilan
1	20	21	95%
2	20	20	100%
3	20	22	90%
4	20	20	100%
Rata-rata			96%

(Sumber : Pengujian )

### 3.5 Pengujian Program Pendeteksi dan Penghitung Jumlah Pengunjung Dengan Kecepatan Normal

Pengujian program pendeteksi dan penghitung jumlah pengunjung dilakukan dengan kecepatan normal bertujuan agar dapat mengetahui respon program terhadap kondisi tersebut. Hasil pengujian program seperti pada table 3.5

Tabel 3.5 Hasil pengujian dengan kecepatan normal

No	Jumlah sebenarnya	Jumlah terhitung	% keberhasilan
1	20	20	100%
2	20	19	95%
3	20	20	100%
4	20	20	100%
Rata-rata			98%

(Sumber : Pengujian )

### 3.6 Pengujian Program Pendeteksi dan Penghitung Jumlah Pengunjung Dengan Kecepatan Cepat

Pengujian program pendeteksi dan penghitung jumlah pengunjung dilakukan dengan kecepatan agak cepat bertujuan agar dapat mengetahui respon program terhadap kondisi tersebut. Hasil pengujian program seperti pada table 3.6

Tabel 3.6 Hasil pengujian dengan kecepatan cepat

No	Jumlah sebenarnya	Jumlah terhitung	% keberhasilan
1	20	19	95%
2	20	20	100%
3	20	19	95%
4	20	20	100%
Rata-rata			97%

(Sumber : Pengujian )

### 3.7 Pengujian keseluruhan

Secara keseluruhan pengujian program penghitung dan pendeteksi jumlah pengunjung dengan pencitraana digital menggunakan webcam memiliki tingkat keberhasilan sebesar 96% .seperti terlihat pada table 3.7

Tabel 3.6 Hasil pengujian keseluruhan

Pengujian	sebenarnya	terhitung	% keberhasilan
Cahaya agak gelap	80	77	96%
Cahaya Normal	80	80	100%
Cahaya Terang	80	83	90%
Kecepatan cepat	80	78	97%
Kecepatan Normal	80	79	98%
Rata-rata			96%

(Sumber : Pengujian )

## IV PENUTUP

### 4.1 Kesimpulan

Berdasarkan hasil perancangan, implementasi, pengujian ,dan analisis sistem maka dapat diambil kesimpulan sebagai berikut:

1. Program ini berhasil mendeteksi dan mendapatkan data jumlah

pengunjung yang melewati *detection window* dengan tingkat kesalahan (*error*) sebesar 96%

2. Tingkat keberhasilan penghitungan dan pendeteksian pada program ini dipengaruhi oleh kecepatan pengunjung melewati *detection window*. Semakin cepat kecepatan melewati *detection window* maka semakin besar kemungkinan tidak tertangkap oleh *webcam*.
3. Penentuan nilai pengali juga sangat berpengaruh pada tingkat keberhasilan program ini. Maka harus ditentukan nilai pengali yang tepat sebelum menjalankan program. Nilai pengali ini

menentukan besarnya nilai *threshold*.

#### 4.2 Saran

Saran yang dapat diberikan untuk pengembangan tugas akhir ini adalah:

1. Cahaya sangat berpengaruh dalam hasil proses perhitungan, maka perlu dikembangkan untuk membuat intensitas yang otomatis dalam perhitungan.
2. Karena hanya dimodel untuk pengunjung masuk satu per satu, maka perlu dikembangkan untuk model satu pintu dengan dua arah. Sehingga perhitungan orang masuk dan orang keluar dalam satu kali proses.

#### DAFTAR PUSTAKA

- [1] Pardosi, Mico, 2005, "Visual Basic 6.0", Selaras, Surabaya.
- [2] Sigit, Riyanto., Basuki, Achmad., Ramadijanti, Nana., Pramadihanto, Dadet., 2005. Step by Step Pengolahan Citra Digital. Yogyakarta: Andi
- [3] Kuncara, Chandra, 2004, "Perancangan dan Pembuatan Program Penghitung Jumlah Orang Menggunakan Webcam.", URL: [http://dewey.petra.ac.id/jiunkpe\\_dg\\_4414.html](http://dewey.petra.ac.id/jiunkpe_dg_4414.html)
- [4] Jae-Won, Kim, Choi, Kang-Sun, Choi, Byeong-Doo, Ko, Sung-Jea. 2010. *Real-time Vision-based People Counting System for the Security Door*. PDF.
- [5] Kurnia, Dede, 2011, "Pengukuran Kepadatan Arus Lalu Lintas Menggunakan Sensor Kamera", URL: <http://dhetiastapurwatna.wordpress.com/2009/05/19/sistem-lalu-lintas/>.
- [6] Suntoyo, Andi, 2011, "Pengambilan Gambar dari Web Camera untuk Membangun Sistem Informasi", URL : <http://www.andisun.com/jurnal/pengambilan-gambar-dari-web-camera-untuk-membangun-sistem-informasi-2>.
- [6] Anonim . Avicap32.dll . <http://www.bigresource.com/VB-Using-avicap32-dll--g1FiUXUdEX.html> ( diakses Desember 2011)