

## Penerapan dan Analisis Perhitungan Orang dengan Metode *Codebook* Studi kasus : Perhitungan orang dalam sebuah antrian

Farid Hidayat<sup>1</sup>, Bedy Purnama<sup>2</sup>, Febryanti Sthevanie<sup>3</sup>

<sup>1</sup>Mahasiswa Prodi S1 Teknik Informatika, Fakultas Informatika, Universitas Telkom

<sup>2</sup>Dosen Pembimbing 1 Prodi S1 Teknik Informatika, Fakultas Informatika, Universitas Telkom

<sup>3</sup>Dosen Pembimbing 2 Prodi S1 Teknik Informatika, Fakultas Informatika, Universitas Telkom

<sup>1</sup>[faird14@gmail.com](mailto:faird14@gmail.com), <sup>2</sup>[bedy.purnama@gmail.com](mailto:bedy.purnama@gmail.com), <sup>3</sup>[febrvantisthevanie@gmail.com](mailto:febrvantisthevanie@gmail.com)

---

### Abstrak

Pada era sekarang perekaman video menggunakan CCTV merupakan teknologi yang sangat baik untuk melakukan pengawasan dan pengenalan. Salah satu kegunaannya yakni melakukan perhitungan orang dalam video untuk melakukan estimasi keramaian pada tempat yang memiliki antrian lebih dari satu. Contohnya pada tempat rekreasi seperti Dufan, Trans Studio Bandung, dan Wisata Bahari Lomongan. Sehingga pengunjung dapat melihat tempat hiburan mana yang sedang sepi ataupun ramai.

Studi kasus yang digunakan yakni melakukan perhitungan pada kasus satu antrian sehingga nantinya dapat diimplementasikan pada kasus yang lebih kompleks seperti kasus yang telah dijelaskan diatas. Dalam penelitian Tugas Akhir ini diimplementasikan dengan menggunakan metode *codebook*. Metode *codebook* digunakan untuk melakukan proses pemisahan *background* dan *foreground* sehingga didapatkan *blob image*. *Blob image* sendiri nantinya akan digunakan untuk melakukan proses ekstraksi manusia menggunakan *aspect ratio*.

Dari hasil pengujian didapatkan bahwa perhitungan dengan metode *codebook* dilanjutkan ekstraksi manusia menggunakan *aspect ratio* mampu mengenali obyek manusia dengan akurasi 86% sesuai dengan kasus yang diujikan.

**Kata kunci** : Perhitungan orang, Antrian, *Blob image*, *Codebook*, *Region of Interest(ROI)*, *Aspect ratio*

---

### Abstract

In the current era of video recording with CCTV is an excellent technology for monitoring and recognition. One role is that is doing the calculation in the video to estimate the crowd at a place that has more than one queue. For example in recreation areas such as Dufan, Trans Studio Bandung, and Marine Tourism Lomongan. So visitors can see the place where the entertainment is being quiet or busy.

Case studies are used that perform calculations in the case of a single queue can later be implemented in more complex cases such as the case described above. In this final project research implemented with codebook method. Codebook method is used to perform the separation of foreground and background processes to obtain BLOB image. Blob image itself will be used to perform the extraction process using the human aspect ratio.

From the test results it was found that the calculation of the human extraction method followed codebook using the aspect ratio mampu humans recognize objects with an accuracy of 86% according to the cases tested.

**Keywords**: Calculation people, Queue, Blob image, codebook, Region of Interest (ROI), Aspect Ratio

---

1. Pendahuluan

Pada era sekarang perekaman video merupakan teknologi yang sangat baik untuk melakukan sistem pengawasan salah satunya dengan menggunakan kamera CCTV. Berkembangnya kamera CCTV sangat dipengaruhi oleh teknologi yang terkomputerisasi sehingga mampu secara otomatis melakukan pengawasan tanpa adanya bantuan operator. Salah satu fungsionalitasnya adalah melakukan perhitungan orang dalam video untuk melakukan estimasi keramaian pada tempat yang memiliki antrian lebih dari satu. Contohnya pada tempat rekreasi seperti Dufan, Trans Studio Bandung, dan Wisata Bahari Lomongan. Sehingga pengunjung dapat melihat tempat hiburan mana yang sedang sepi ataupun ramai. Pada Tugas Akhir ini studi kasus yang digunakan yakni melakukan perhitungan pada kasus satu antrian sehingga nantinya dapat diimplementasikan pada kasus yang lebih kompleks seperti kasus yang telah dijelaskan diatas.

Dalam penelitian-penelitian sebelumnya, dilakukan beberapa metode untuk melakukan perhitungan orang. Beberapa diantaranya adalah *Ellipse Detection and Forward/Backward Tracing* [5], *Multiple Lines* [2], *Curve Analysis Method* [13], *Head Detection* [4]. Metode yang digunakan dalam sistem ini adalah metode *background subtraction* menggunakan *codebook* karena memiliki kecepatan proses yang lebih cepat dan penggunaan memori yang lebih efisien dibandingkan dengan metode MoG maupun kernel [6].

Dalam Tugas Akhir ini peneliti mencoba untuk mengimplementasikan aplikasi perhitungan orang menggunakan metode *codebook* yang kemudian dilakukan proses deteksi obyek manusia untuk melakukan perhitungan jumlah orang. Metode *codebook* digunakan untuk memisahkan *background* dan *foreground* sehingga didapatkan *blob image*. Kemudian *blob image* ini digunakan untuk proses ekstraksi manusia menggunakan *aspect ratio* untuk mendeteksi obyek manusia. Obyek manusia akan dihitung ketika berada pada daerah ROI (*Region of Interest*). Kemudaaian dari hasil perhitungan akan dilakukan analisis performansi dengan beberapa kasus yang diujikan.

Adapun tahapan penelitian yang dipakai yaitu : Identifikasi masalah dan pekerjaan yang berkaitan, Studi literatur, Desain, dan Pembahasan

2. Tinjauan Pustaka

2.1 Codebook

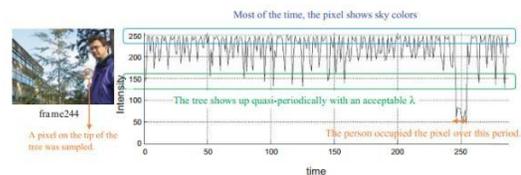
Algoritma *codebook* adalah model yang dibangun melalui *learning* yang dilakukan secara berurutan. Setiap piksel *codebook* dibangun oleh satu atau lebih *codeword*. Sampel dari tiap piksel bersatu dalam himpunan *codeword* berdasarkan distorsi warna bersama dengan batas kecerahannya.

Dalam proses deteksi menggunakan perbedaan antara warna dan kecerahan *background*. Sebuah piksel dikategorikan bukan merupakan *background* ketika kondisi *codeword* kurang dari *threshold* dan tingkat kecerahannya berada pada jangkauan *codeword* tidak dipenuhi.

Misal  $\phi$  dan  $\tau_i$  menotasikan *background model* (*Codebook* yang telah diperbaiki setelah filtering sementara) dan nilai *threshold* berturut-turut. Pada umumnya,  $\tau_i$  diatur nilainya sama dengan setengah dari jumlah frame *training*,  $N/2$

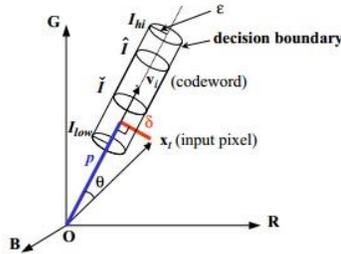
$$\phi = \{ \phi_i | \phi_i \in \ell \wedge \phi_i \leq \tau_i \}$$

*Codeword* yang memiliki  $\phi$  besar, akan dieliminasi dari *codebook* oleh persamaan Meskipun memiliki frekuensi  $f_j$  yang besar,  $\phi$  yang besar tersebut berarti bahwa itu adalah event kemunculan sebuah *foreground* yang stasioner/diam hanya pada periode  $\phi$ . Sedangkan apabila terjadi pada  $f_j$  yang kecil, dengan terjadinya sebuah kemungkinan dimana muncul *background event* yang langka dan terjadi secara *quasy-periodic*.  $\phi$  dapat digunakan untuk mendiskriminasi/memisahkan *codeword* pada *background* yang aktual dari *codeword* pada *foreground* yang bergerak.



Gambar 2. 1 Contoh penggunaan MNLR

Untuk memfasilitasi perubahan *brightness* dalam melakukan deteksi, maka statistik  $\hat{\lambda}$  dan  $\hat{\lambda}$  disimpan, dimana *brightness* minimum dan *brightness* maksimum dari semua piksel di masukkan ke sebuah *codeword* di 6-tuple  $(\hat{\lambda}, \hat{\lambda}, f, \lambda_i, \phi, \phi)$ . Perubahan variasi *brightness* difasilitasi dalam *range* tertentu yang membatasi level *shadow* dan *highlight*. Range tersebut adalah  $[\phi_w, \phi_h]$ , untuk setiap *codeword* didefinisikan sebagai berikut :



Gambar 2. 2 YUV Color model

2.2 Ekstraksi fitur

Ekstraksi fitur merupakan sebuah metode yang bertujuan untuk mendapatkan beberapa informasi tentang fitur yang akan diteliti pada sebuah obyek. Jika obyek yang akan diteliti adalah manusia yang sedang berjalan, maka ada beberapa fitur yang bisa diambil dari obyek manusia tersebut.

1. Aspek Rasio[13,14].
2. Gradien Vertikal ( $G_y$ ) dan Horizontal ( $G_x$ ) dari suatu obyek[13,14].

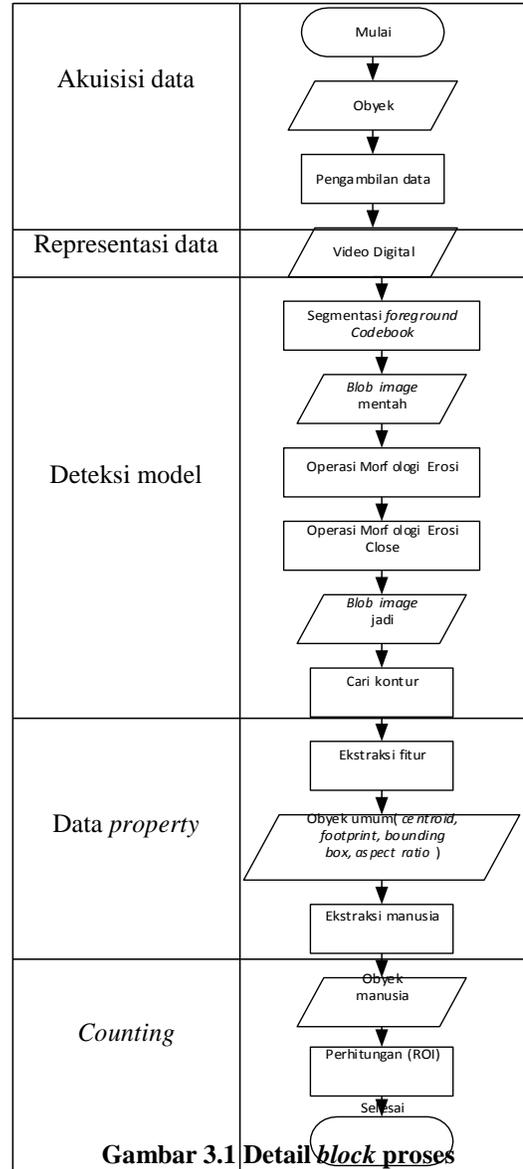
Ketika seseorang sedang berjalan, nilai gradien horizontalnya lebih kecil dibandingkan dengan nilai gradien vertikalnya ( $G_x < G_y$ ). Berbeda dengan kucing yang memiliki nilai gradien vertikalnya ( $G_x > G_y$ ).



Gambar 2. 3 Aspect ratio obyek

3. Perancangan

Sistem yang dibangun dalam Tugas Akhir ini dibuat dengan metode *codebook*. *Codebook* digunakan untuk memisahkan antara *foreground* dan *background* yang kemudian akan dapat diambil sebuah *blob image* yang akan digunakan sebagai deteksi obyek dengan ekstraksi manusia menggunakan *aspect ratio*. Setelah itu dilakukan tahapan perhitungan dengan menggunakan area yang dibentuk oleh ROI (*Region of Interest*). Berikut gambaran untuk *block proses* sistem secara keseluruhan :



Gambar 3.1 Detail block proses

Sistem *people counting* ini diimplementasikan dalam lingkungan perangkat keras dan perangkat lunak berikut ini.

Spesifikasi Perangkat Keras

Spesifikasi perangkat keras yang digunakan untuk membangun sistem ini adalah :

- Processor Intel®Core™ i3-350M CPU @ 2.27 GHz 2.27 GHz
- Memory 2.00 GB RAM
- Hardisk 250 GB
- Webcamera, Kamera Digital, atau IP Camera

**Spesifikasi Perangkat Lunak**

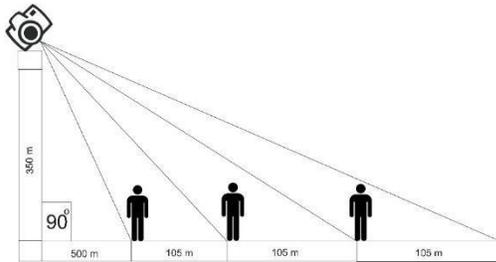
Perangkat lunak yang digunakan untuk membangun sistem ini adalah sebagai berikut :

- Sistem Operasi Windows 8 Pro 32-Bit
- OpenCV 2.4.2 library
- Microsoft Visual Studio 2010 Ultimate C++/C
- CMake

**4. Pembahasan**

Eksperimen telah dilakukan dengan beberapa data uji dan data latihan dengan spesifikasi perangkat lunak dan perangkat keras yang telah disebutkan.

Data yang digunakan dalam penelitian ini adalah data video yang diperoleh dari rekaman kamera di gedung LC dengan jarak dan sudut sesuai dengan **gambar 4.1**. Data video yang dihasilkan memiliki resolusi 800 x 600 piksel dengan format avi.



**Gambar 4.3 Pengambilan video**

Berikut data latihan yang digunakan dalam penelitian ini :

Data latihan I

No	Video	Jumlah
1	Satu orang berjalan dengan jarak 500 cm	10
2	Satu orang berjalan dengan jarak 605 cm	10
3	Satu orang berjalan dengan jarak 710 cm	10

Data latihan I digunakan untuk menganalisis parameter pada proses Ekstraksi manusia menggunakan *aspect ratio*

Data latihan II

No	Video	Jumlah
1	Satu orang berjalan dengan warna <i>background</i> sangat berbeda dengan warna bayangan	3

2	Satu orang berjalan dengan warna <i>background</i> mirip dengan warna bayangan	3
---	--	---

Data latihan yang digunakan untuk mengubah parameter *low* pada proses segmentasi *background codebook* untuk menghilangkan efek bayangan

Kemudian untuk melakukan pengujian performansi dari sistem yang dibangun digunakanlah data uji yang terdiri dari sebelas video yang memiliki *noise* dan tanpa adanya *noise*. Berikut data uji yang digunakan dalam penelitian ini :

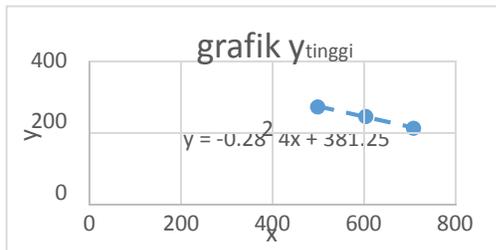
No.	Video	Jumlah	Data
1	Satu orang berjalan satu persatu untuk mengantri pada jarak 500 cm	1	Uji I
2	Satu orang berjalan satu persatu untuk mengantri pada jarak 605 cm	1	Uji I
3	Satu orang berjalan satu persatu untuk mengantri pada jarak 710 cm	1	Uji I
4	Enam orang berjalan satu persatu untuk mengantri tanpa dempetan pada jarak 500 cm	1	Uji II
5	Enam orang berjalan satu persatu untuk mengantri tanpa dempetan pada jarak 605 cm	1	Uji II
6	Enam orang berjalan satu persatu untuk mengantri tanpa dempetan pada jarak 710 cm	1	Uji II
7	Enam orang berjalan satu persatu untuk mengantri dengan dempetan pada jarak 500 cm	1	Uji III
8	Enam orang berjalan satu persatu untuk mengantri dengan dempetan pada jarak 605 cm	1	Uji III
9	Enam orang berjalan satu persatu untuk mengantri dengan dempetan pada jarak 710 cm	1	Uji III

10	Kucing dan satu orang berjalan pada area perhitungan	1	Uji IV
11	Enam orang mengantri dengan satu orang belum melewati area perhitungan	1	Uji V

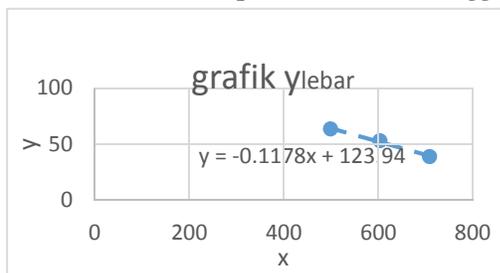
Dari data video diatas kemudian dilakukan dua buah skenario uji yang akan dilakukan terhadap sistem perhitungan orang yaitu analisis terhadap data latih dan pengujian perhitungan orang menggunakan data uji. Analisis terhadap data latih terbagi menjadi dua yakni data latih I dan II. Analisis data latih I yaitu untuk menentukan parameter lebar dan tinggi obyek manusia serta menentukan pola perubahan tinggi dan lebar pada obyek manusia. Analisis data latih II digunakan untuk melakukan analisis parameter *low codebook* untuk menghilangkan bayangan. Kemudian dilakukan pengujian perhitungan orang menggunakan data uji yang telah diambil. Dalam proses perhitungan ini dibantu menggunakan hasil dari analisis data latih I dan II untuk mengoptimalkan proses pengenalan dan perhitungan obyek.

Berikut adalah hasil eksperimen yang telah dilakukan:

**4.1 Hasil analisis data latih I**



**Gambar 4.1 Grafik persamaan linier tinggi**



**Gambar 4.2 Grafik persamaan linier lebar**

Dari data yang telah diolah didapatkan nilai-nilai yang nantinya akan digunakan sebagai batas penentuan obyek manusia pada proses ekstraksi manusia dengan *aspect ratio*. Dapat dilihat pada gambar grafik, bahwa

tinggi dan lebar obyek disangat dipengaruhi oleh jarak obyek dengan jarak kamera. Semakin dekat dengan kamera maka panjang dan lebar obyek akan semakin besar.

Kemudian didapatkan pola perubahan tinggi dan lebar yang dihitung menggunakan regresi linier sehingga dihasilkan persamaan sebagai berikut :

- $y_{lebar} = -0,1178x + 123,94$
- $y_{tinggi} = -0,2824x + 381,25$

**4.2 Hasil data latih II**

Pengujian ini dilakukan untuk mendapatkan nilai parameter *low* yang digunakan untuk menghilangkan bayangan. Skenarionya dialukan pada tiga orang yang masing-masing berjalan dengan background yang kontras dengan warna bayangan dan tiga orang yang masing-masing berjalan dengan background yang sama dengan warna bayangan. Kemudian tabel diisi dengan indek sebagai berikut :

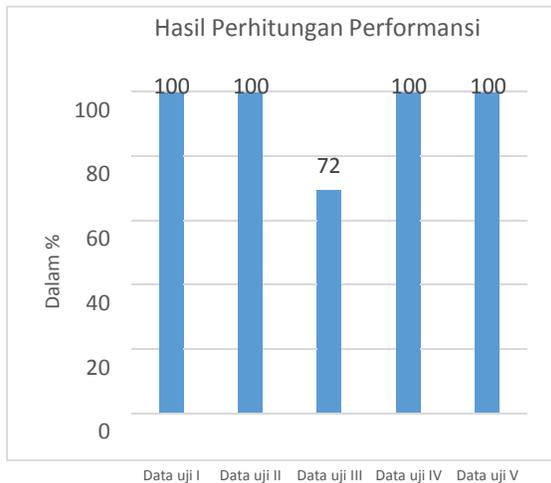
- 1 = tanda bahwa bayangan terdeteksi *background*
- 2 = tanda bahwa bayangan terdeteksi *foreground*
- 3 = tanda bahwa obyek terdeteksi *background*
- 4 = tanda bahwa obyek terdeteksi *foreground*

Hasil dari pengujian yang diharapkan adalah sistem mampu mendeteksi obyek sebagai *foreground*(4) dan bayangan sebagai *background*(1). Dari data yang didapat, menunjukkan bahwa pada penelitian ini parameter terbaik terletak antara 20 sampai dengan 30. Sehingga nilai parameter *low codebook* yang dipakai pada penelitian ini adalah rata-rata dari nilai 20 dengan 30 yakni sebesar 25.

video	tipe	nilai parameter <i>low</i>					
		0	10	20	30	40	50
1	kontras	2,4	2,4	1,4	1,4	1,4	1,4
2	kontras	2,4	2,4	1,4	1,4	1,4	1,4
3	kontras	2,4	2,4	1,4	1,4	1,4	1,3
4	sama	2,4	2,4	1,4	1,4	1,3	1,3
5	sama	2,4	2,4	1,4	1,4	1,3	1,3
6	sama	2,4	2,4	1,4	1,4	1,3	1,3

### 4.3 Hasil pengujian data uji

Dari keseluruhan hasil pengujian data uji didapatkan hasil keseluruhan akurasi dari tiap kasus yang direpresentasikan pada gambar berikut :



**Gambar 4.3 Akurasi keseluruhan**

Pada pengujian data uji I dilakukan dengan melakukan Ekstraksi manusia menggunakan *aspect ratio* yang sesuai dengan hasil analisis data latih yang telah dilakukan. Pengujian pada jarak 500 cm menggunakan batas lebar antara 48 sampai 82 dan tinggi 215 sampai 255. Untuk pengujian pada jarak 605 cm menggunakan batas lebar antara 37 sampai 74 dan tinggi 191 sampai 225. Pada jarak 710 cm lebar yang digunakan dari 31 sampai 79 dan tingginya dari 166 sampai 192. Selain itu pengujian kali ini diharapkan sistem dapat menentukan obyek manusia secara tepat menggunakan perbandingan tinggi dan lebar sesuai dengan data latih sesuai dengan jarak masing-masing. Dari hasil perhitungan orang yang didapatkan akurasi sistem untuk kasus 1 adalah 100 %, begitu juga dengan kasus 2 dan 3. Dari akurasi ini menunjukkan bahwa sistem dapat melakukan perhitungan obyek orang secara tepat. Data latih yang telah dianalisis juga sangat baik dalam menangani kasus ini.

Pada pengujian data uji II dilakukan dengan melakukan Ekstraksi manusia menggunakan *aspect ratio* yang sesuai dengan hasil analisis data latih yang telah dilakukan. Pengujian pada jarak 500 cm menggunakan batas lebar antara 48 sampai 82 dan tinggi 215 sampai 255. Untuk pengujian pada jarak 605 cm menggunakan batas lebar antara 37 sampai 74 dan tinggi 191 sampai 225. Pada jarak 710 cm lebar yang digunakan dari 31 sampai 79 dan tingginya dari 166 sampai 192. Tujuan pengujian pada skenario ini adalah menganalisis akurasi sistem perhitungan orang dengan keadaan terdapat sekumpulan orang yang

berjalan secara teratur dan tidak adanya dempetan dengan tujuan untuk mengantri. Selain itu pengujian ini dilakukan untuk menguji akurasi data latih I yang didapat pada setiap jarak. Didapatkan hasil pengolahan dari kasus 1, 2, dan 3 sebesar 100 %. Hal ini disebabkan karena jumlah orang yang masuk dan jumlah orang yang keluar sesuai dengan yang

diharapkan. Dari akurasi yang didapatkan menunjukkan bahwa sistem mampu menangani sekumpulan orang yang berjalan secara teratur dengan akurasi perhitungan orang 100 %. Selain itu hasil analisis pada data latih teruji baik dalam menangani perhitungan orang untuk setiap jarak.

Pada pengujian data uji III diharapkan dari pengujian ini adalah sistem dapat mendeteksi obyek manusia pada jarak tertentu pada pengambilan video serta mampu menangani perilaku orang yang secara alami terjadi pada antrian. Untuk itu data uji yang digunakan

terjadi dempetan, orang mengobrol, banyak gerak karena gelisah, dan orang membawa tas punggung. Selain itu data uji ini juga diambil mirip dengan data uji II yakni sekumpulan orang yang mengantri. Pada pengujian ini digunakan proses pembagian dengan lebar standard baku yang telah didapatkan dari data uji I. Dari pengujian yang dilakukan didapatkan akurasi sistem pada kasus 1 adalah sebesar 75 %. Ini disebabkan karena ada beberapa kesalahan perhitungan diantaranya sistem menghitung jumlah orang berlebih dari jumlah yang sesungguhnya. Tidak hanya itu sistem juga menghitung kurang dari jumlah orang yang masuk dalam antrian. Akan tetapi kesalahan perhitungan ini terjadi hanya pada beberapa *frame* yang tidak stabil. Untuk beberapa *frame* sistem sudah cukup baik untuk menghitung orang pada kasus 1. Kemudian pada kasus II akurasi sistem menurun lagi dengan akurasi sebesar 66 %. Hal ini terjadi karena pada beberapa *frame* sistem tidak akurat dalam menghitung orang pada antrian. Ini disebabkan karena banyaknya pergerakan yang membuat lebar obyek menjadi sangat besar sehingga ketika dilakukan proses perbandingan menjadi tidak stabil. Pada kasus 3 akurasi sistem naik kembali dengan nilai sebesar 75 %. Pada kasus ini hal yang sama terjadi yakni proses pergerakan orang pada antrian yang membuat lebarnya menjadi tidak stabil. Secara keseluruhan akurasi sistem pada kasus ini adalah 72 % dari rata-rata akurasi pada setiap kasus.

Pengujian data uji IV ini dilakukan untuk melakukan analisis terhadap kucing yang berjalan pada antrian. Karena kucing memiliki nilai lebar yang lebih besar dari tingginya membuat kucing memiliki perbedaan

yang cukup besar dengan obyek manusia. Dari hasil percobaan menunjukkan bahwa kucing tidak terdeteksi dalam antrian sehingga membuat akurasi pada pengujian ini adalah 100%.

Dari pengujian data uji V diharapkan sistem tidak melakukan perhitungan terhadap obyek yang dikenali

manusia dari proses *aspect ratio* ketika berada pada daerah antrian. Hasil pengolahan akurasi sistem menunjukkan bahwa orang tidak melewati daerah antrian sehingga orang tersebut tidak dihitung oleh sistem. Kemudian dari akurasi 100% yang dihasilkan menunjukkan bahwa sistem ini mampu melakukan perhitungan dengan baik pada kasus ini.

## 5. Kesimpulan

Perhitungan orang dengan menggunakan metode codebook dapat memisahkan foreground dan background dengan baik akan tetapi perlu dilakukan perubahan nilai parameter low untuk menghilangkan efek bayangan pada blob image yang didapatkan.

Ekstraksi manusia menggunakan aspect ratio kurang optimal untuk melakukan pengenalan obyek manusia pada kasus Dempetan oleh karena itu dilakukan pembagian ROI obyek manusia menggunakan lebar obyek manusia dari hasil data latih yang didapat. Akan tetapi pengenalan obyek manusia menggunakan cara ini juga kurang efisien untuk melakukan penanganan terhadap kasus Dempetan.

Akurasi perhitungan pada sistem yang dianalisis adalah 86 %. Nilai ini sesuai dengan harapan yang ingin dicapai. Alasan mengapa tidak 100 % karena terjadinya Dempetan yang mengakibatkan sistem tidak bisa menghitung obyek manusia dengan akurat.

## Daftar Pustaka

- [1] Ano Rangga. "Implementasi dan Analisis Metode Optical Flow untuk Penghitungan Orang". Bandung: IT Telkom, 2013.
- [2] Barandiaran, Javier. 2009. *Real-Time People Counting Using Multiple Lines*. IEEE
- [3] Bradsky, G., & Kaehler, A. 2008. *Learning OpenCV Computer Vision with the OpenCV Library*. O'Reilly Media.
- [4] Hu, Yaowu. 2011. *A New Fast and Robust Method Based on Head Detection for People- Flow Counting Sistem*. IEEE
- [5] Huang, Chung-Lin. 2011. *People Counting using Ellipse Detection and Forward/Backward Tracing*. IEEE
- [6] Kyungnam, Thanarat, David, dan Davis L. 2005. "Real-time foreground-background

*segmentation using codebook model*". London, Academic Press Ltd.

- [7] Merad, Aziz, dan Thome. 2010. "Fast People Counting Using Head Detection From Skeleton Graph". IEEE
- [8] Narayanan, Harishankar. 2011. Blob Detection in Kinect. [Online] Available at: <http://code42tiger.blogspot.com/2011/03/blob-detection-in-kinect.html> [Accessed 2 Desember 2014].
- [9] Oliv, Dam's. 2010. Ben-Hatchell-sequence. [Online] Available at: <http://skateboarders.le-site-du-skateboard.com/ben-hatchell/ben-hatchell-sequence/> [Accessed 2 Desember 2014].
- [10] Raja David Hasugian. *Tugas Akhir : Perhitungan Aliran Orang secara Waktu Nyata dengan Pendekatan Deteksi Kepala (People Counting)*. Bandung : IT Telkom, 2013
- [11] Vinay, Chittaranjan, dan Sural S. 2007. "Automatic detection human fall in video". Berlin, Heidelberg
- [12] Yusfia Hafid Aristyagama. *Tugas Akhir : Tracking Obyek menggunakan Kalman Filter Studi Kasus : Tracking Obyek Manusia yang Berjalan (Single Object) pada Overlapping View Multi-Camera*. Bandung : Universitas Telkom, 2013
- [13] Zainuddin, Z. dan Aik, Lim Eng. 2009. *Real-Time People Counting Sistem using Curve Analysis Method*. International Journal of Computer and Electrical Engineering

