

HUMAN DETECTION MENGGUNAKAN METODE HISTOGRAM OF ORIENTED GRADIENTS (HOG) BERBASIS OPEN_CV

Kusno Suryadi, Supriyanto Sikumbang

Teknik Elektro, Universitas Gajayana Malang
e_mail : cegukks@yahoo.com

Abstrak

Pengolahan citra untuk mendeteksi tubuh manusia menggunakan metode HOG berbasis Opencv. mampu membedakan objek manusia dan bukan manusia. Sistem ini juga akan melakukan proses record data berupa jumlah manusia yang terdeteksi dalam bentuk data dan video. Cara kerja dari aplikasi human *detection* menggunakan metode HOG berbasis OpenCV ini adalah, Input berupa video yang ditangkap oleh webcam dengan format file .AVI akan dicacah mejadi frame-frame gambar yang kemudian diproses oleh HOG *detection* sehingga diputuskan hasil pendeteksian apakah manusia atau bukan manusia. Hasil keputusan dari HOG *detection* akan ditandai dengan *bounding box* berbentuk persegi (*rectangle*). Video yang tersusun dari frame-frame yang terdeteksi adanya kehadiran manusia tersebut lalu disimpan dalam format file .AVI. Setelah itu hasil pendeteksian adanya kehadiran manusia tersebut disimpan pada sebuah tabel dalam format file microsoft excel.

Kata kunci: *image processing, human detection, HOG detection*

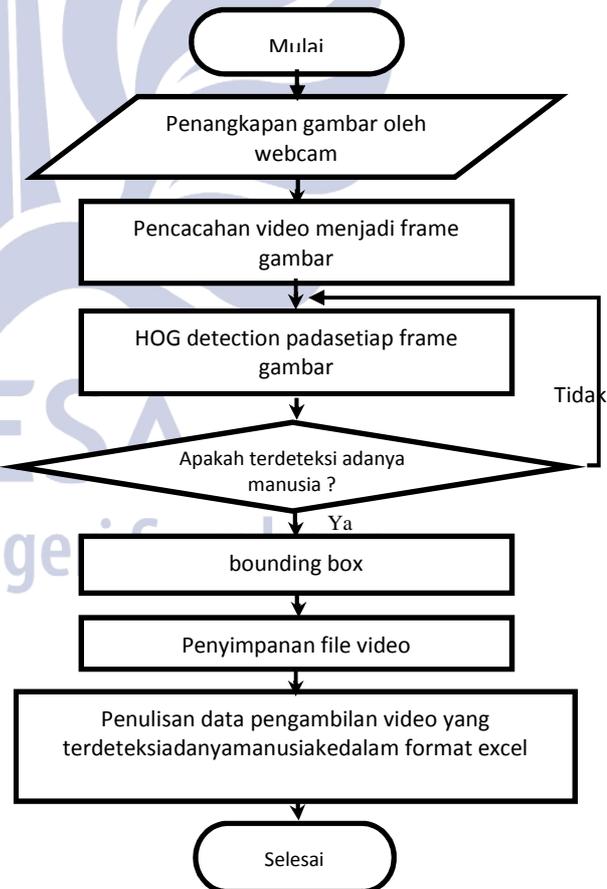
PENDAHULUAN

Proses deteksi pada pengolahan citra merupakan kajian yang menarik dan banyak dikembangkan oleh para peneliti, baik yang diaplikasikan pada system robotika, maupun pada bidang keamanan. Pada bidang keamanan penggunaan kamera direalisasikan sebagai sensor visual guna mengidentifikasi adanya kejahatan. Pada saat ini kamera yang banyak digunakan adalah jenis CCTV. Hampir disetiap sudut mulai dari fasilitas publik, pusat perbelanjaan, bank, perpustakaan, jalan raya sampai rumah pribadi dilengkapi dengan kamera pengaman. Kamera CCTV ini mampu mengidentifikasi seluruh obyek yang tertangkap oleh kamera ini, dan pada umumnya record data menggunakan CCTV dilakukan secara kontiyu berdasarkan waktu record. Pada sistem ini akan dibahas aplikasi kamera digital pada proses *humandetection* menggunakan metode *Histogram of Oriented Gradients* (HOG). Proses *human detection* menggunakan metode HOG bertujuan untuk mendeteksi kehadiran manusia didalam suatu area pantauan kamera. Sedangkan proses record data akan dilakukan secara otomatis, yaitu pada saat terdeteksi obyek (manusia) dan proses record data akan berhenti pada saat obyek sudah tidak terdeteksi oleh *camera*. Proses *record* data pada aplikasi ini selain dalam bentuk video juga akan merecord jumlah manusia yang terdeteksi oleh kamera tersebut. Aplikasi *human detection* menggunakan metode HOG bertujuan untuk mengoptimalkan kerja kamera pada proses deteksi tubuh manusia.

METODE

Sistem deteksi manusia merupakan proses deteksi adanya kehadiran manusia dalam sebuah video yang ditangkap oleh kamera. Referensi posisi manusia yang

akan dideteksi berdiri tegak lurus dengan kondisi diam. Pada sistem ini kecepatan proses deteksi dapat dipengaruhi oleh ketepatan pendeteksian. Gambar 1. Menunjukkan sistem kerja deteksi manusia menggunakan metode HOG berbasis OpenCv.



Gambar 1. Proses deteksi kehadiran manusia

Gambar 1. menunjukkan alur sistem pendeteksian kehadiran manusia dalam sebuah video dengan

menggunakan metode HOG *detection* berbasis OpenCV. Input berupa video yang ditangkap oleh webcam dengan format file AVI akan dicacah menjadi frame-frame gambar yang kemudian diproses oleh HOG *detection* sehingga diputuskan hasil pendeteksian apakah manusia atau bukan manusia. Hasil keputusan dari HOG *detection* jika manusia maka akan ditandai dengan bounding box berbentuk persegi (rectangle). Video yang tersusun dari frame-frame yang terdeteksi adanya kehadiran manusia tersebut lalu disimpan dalam format file AVI. Setelah itu hasil pendeteksian adanya kehadiran manusia tersebut disimpan dalam bentuk tabel dalam format file microsoft excel.

1. OpenCv

OpenCV merupakan open source library untuk computer vision. Library ini ditulis dalam bahasa C dan C++ dan dapat dijalankan dengan berbagai operating system yaitu Linux, Windows, dan mac OS X. OpenCV didesain untuk efektivitas dalam computer vision dan dengan focus aplikasi realtime yang kuat. Salah satu tujuan dari OpenCV adalah untuk menyediakan infrastruktur computer vision yang mudah penggunaannya sehingga dapat digunakan dengan cepat membangun aplikasi vision yang berharga. Library dari OpenCV mengandung sekitar 500 lebih fungsi yang terdapat pada berbagai area, termasuk *factory product inspection, medical imaging, security, user interface, camera calibration, stereo vision, dan robotics*. OpenCV juga mengandung *Machine Learning Library* (MLL) yang lengkap. Sublibrary ini difokuskan pada pengenalan pola secara statistik dan *clustering*. MLL sangat berguna untuk pekerjaan vision yang merupakan inti dari misi OpenCV dan secara umum, MLL cukup berguna untuk berbagai masalah *machine learning*.

2. Proses Pencacahan Sinyal Video

Proses pencacahan video dilakukan dengan cara mencacah sinyal video menjadi frame-frame gambar, pada sistem ini pengambilan gambar video oleh webcam menggunakan instruksi `frame = cvQueryFrame(capture)`. Instruksi OpenCV ini akan mencacah video yang sudah ditangkap oleh webcam menjadi beberapa frame yang sesuai dengan kapasitas kecepatan penangkapan gambar yang dimiliki oleh webcam. Proses pencacahan video menjadi frame gambar adalah,

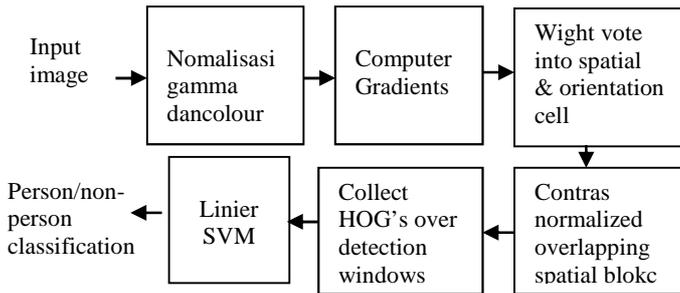
```
capture = cvCaptureFromCAM(0);
if(capture == 0){
    cout<<"ERROR: NO CAMERA
DETECTED!\n";
    cvReleaseCapture(&capture);
    cin>>i;
    return -1;
}
cout<<"Camera detected.\n";
```

```
cout<<"Creating hog object...\n";
hog.setSVMDetector(HOGDescriptor::getDefaultPe
opleDetector());
cout<<"Creating window...\n";
cvNamedWindow(windowName,CV_WINDOW_AUT
OSIZE);
while(true){
try{
frame = cvQueryFrame(capture);
if(frame == 0){
cvDestroyWindow(windowName);
cvReleaseCapture(&capture);
cout<<"ERROR: BAD FRAME!\n";
cin>>i;
}
}catch(exception e){
cout<<e.what()<<endl;
cin>>i;
return -1;
}
```

```
Mat mFrame(frame);
vector<Rect> found, found_filtered;
double t = (double)GetTickCount();
```

3. HOG Detection

HOG *detection* digunakan untuk memutuskan hasil pendeteksian apakah manusia atau bukan manusia. HOG *detection* dapat menganalisa apakah obyek yang ditangkap oleh kamera adalah manusia atau bukan manusia. Proses deteksi manusia pada sistem ini dilakukan per frame gambar sebagai hasil pencacahan file video. Ide dasar dari metode HOG *detection* adalah tampilan obyek lokal dan bentuk dapat dicirikan cukup baik oleh distribusi intensitas gradien lokal atau arah tepi, hal ini diimplementasikan dengan membagi jendela gambar ke dalam daerah ruang-ruang kecil ("sel"), untuk setiap sel mengumpulkan histogram arah gradien lokal 1-D nya atau berorientasi sudut terhadap pixel dari sel. Gabungan masukan histogram membentuk representasinya. Untuk invarian yang lebih baik terhadap pencahayaan, bayangan, juga berguna untuk menormalkan kontras tanggapan lokal sebelum menggunakannya. Hal ini dapat dilakukan dengan mengumpulkan suatu ukuran histogram lokal "energi" terhadap wilayah spasial yang lebih besar ("blok") dan menggunakan hasilnya untuk menormalkan semua sel didalam bloknya. Gambar 2 menunjukkan urutan kerja dari metode HOG.



Gambar 2. Rantai pendeteksian objek menggunakan metode HOG (Dalal, 2005:3)

Normalize gamma and color

Proses *Normalize gamma and color* atau normalisasi gamma dan warna pada frame gambar dari video terjadi dalam *framework HOG* pada OpenCV. Normalisasi gamma dan warna pada frame gambar pada sistem ini menggunakan instruksi : `void normalize_hists`. Proses normalisasi gamma dan warna dari frame gambar pada *framework HOG* dalam library OpenCV dilakukan pada frame-frame gambar dari video yang ditangkap oleh webcam. Instruksi OpenCV ini akan menormalisasikan gamma dan warna pada frame-frame gambar tersebut.

Compute Gradients

Proses *computer gradients* atau perhitungan gradien pada frame gambar dari video yang ditangkap oleh webcam terjadi dalam *framework HOG* pada OpenCV seperti yang terlihat pada rantai pendeteksian *HOG detection*. Komputasi gradien pada frame gambar menggunakan instruksi : `void cv::gpu::HOGDescriptor::computeGradient`. Proses komputasi gradien gambar pada *framework HOG* dalam library OpenCV dilakukan pada frame-frame gambar dari video yang ditangkap oleh webcam. Instruksi OpenCV ini akan mengkomputasikan gradien pada gambar-gambar tersebut.

Weighted vote into spatial and orientation cells

Pada rantai *HOG detection* terdapat proses *weighted vote into spatial and orientation cells* atau pemilihan bobot kedalam sel berorientasi dan spasial pada *frame* gambar dari video yang ditangkap oleh webcam yang terjadi dalam *framework HOG* pada OpenCV. Pemilihan bobot kedalam sel berorientasi dan spasial pada *frame* gambar dari video yang ditangkap oleh webcam menggunakan instruksi OpenCV : `cv::gpu::HOGDescriptor::HOGDescriptor`. Proses pemilihan bobot kedalam sel berorientasi dan spasial pada *framework HOG* dalam library OpenCV dilakukan pada frame-frame gambar dari video yang ditangkap oleh webcam. Instruksi OpenCV ini akan melakukan pemilihan bobot kedalam sel berorientasi dan spasial pada frame-frame gambar tersebut.

Contrast normalized over overlapping spatial blocks

Proses *contrast normalized over overlapping spatial blocks* atau normalisasi kontrasterhadap blok spasial yang bertumpukan pada *frame* gambar dari video yang ditangkap oleh webcam terjadi dalam *framework HOG* pada OpenCV seperti yang terlihat pada rantai pendeteksian *HOG detection*. Normalisasi kontrasterhadap blok spasial yang bertumpukan pada openCV menggunakan instruksi: `void cv::gpu::HOGDescriptor::computeBlockHistograms`. Proses normalisasi kontras terhadap blok spasial yang bertumpukan pada *framework HOG* dilakukan pada frame-frame gambar dari video yang ditangkap oleh webcam.

Collect HOG's over detection windows

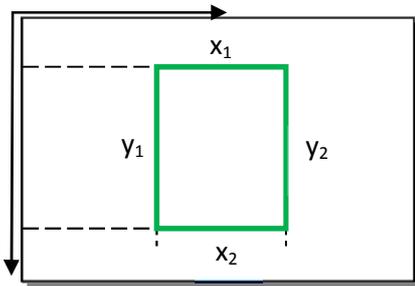
Pada rantai *HOG detection* terdapat proses *Collect HOG's over detection windows* atau pengumpulan HOG terhadap jendela deteksi pada *frame* gambar dari video yang ditangkap oleh webcam yang terjadi dalam *framework HOG* pada OpenCV. Pengumpulan HOG terhadap jendela deteksi pada *frame* gambar dari video yang ditangkap oleh webcam menggunakan instruksi OpenCV : `void cv::gpu::HOGDescriptor::detectMultiScale`. Proses pengumpulan HOG terhadap jendela deteksi gambar pada *framework HOG* dalam library OpenCV dilakukan pada keseluruhan *frame* gambar dari video yang ditangkap oleh webcam. Instruksi OpenCV ini akan melakukan pengumpulan HOG terhadap jendela deteksi pada gambar-gambar tersebut.

Linear SVM

Dalam penelitian ini proses *Linear SVM* atau *SVM linear* pada gambar dari video yang ditangkap oleh webcam terjadi dalam *framework HOG* pada OpenCV seperti yang terlihat pada rantai *HOG detection*. *SVM linear* pada gambar menggunakan instruksi : `void cv::gpu::HOGDescriptor::setSVMDetector`. Proses *SVM linear* dari gambar pada *framework HOG* dalam library OpenCV dilakukan pada frame-frame gambar dari video yang ditangkap oleh webcam. Instruksi OpenCV ini akan melinier *SVM*-kan gambar-gambar tersebut. Kemampuan melinier *SVM*-kan *HOG detection* diperoleh melalui training yang disimpan dalam database OpenCV. Pada proses ini tidak perlu melakukan training lagi karena sistem dapat menggunakan database training pada OpenCV.

4. Bounding Box

Proses *bounding box* digunakan untuk penandaan kehadiran adanya manusia. Pada OpenCV *bounding box* dibuat dengan menggunakan instruksi `Rect r` dan `rectangle`. Dimensi dari *bounding box* dibuat dengan menggunakan instruksi `cvRound(r.width*0.1)`. Berikut ini adalah bentuk dari suatu *bounding box*.



Gambar 3. Proses Bounding Box

5. Penyimpanan File Video

Pada penelitian ini file video yang terdeteksi adanya manusia disimpan dengan format AVI dengan durasi waktu hanya selama terdeteksi adanya kehadiran manusia. Format file video AVI dipilih karena kualitas video hasil penangkapan yang lebih baik dibandingkan format file video lainnya. Pada OpenCV penamaan file video ini dilakukan dengan menggunakan instruksi : *void generateAviFileName()*, dan proses penyimpanan video menggunakan instruksi *cvCreateVideoWriter*.

6. Penyimpanan Data

Pada penelitian ini kuantitas dari setiap kehadiran sejumlah tertentu manusia yang berhasil dideteksi disimpan dalam bentuk tabel pada format file microsoft excel. Data yang dituliskan terdiri dari jumlah record, nomor, waktu deteksi dan jumlah orang yang berhasil terdeteksi. Pada system ini penyimpanan file excel dilakukan secara otomatis setiap mendeteksi adanya kehadiran manusia dengan cara memanggil framework basic excel.cpp yaitu *void generateExcelFileName*.

HASIL DAN PEMBAHASAN

Sistem ini mendeteksi adanya kehadiran manusia dalam sebuah video yang ditangkap oleh camera secara online. Dalam penelitian ini pengujian dilakukan dalam beberapa posisi dengan kondisi objek tidak bergerak.

1. Posisi Obyek Tunggal tampak depan (180°)

Pengujian ini dilakukan untuk mengetahui pengaruh dari variasi posisi manusia terhadap keakurasian dan kecepatan pendeteksian. Berdasarkan gambar dan tabel hasil penyimpanan maka hasil dari pengujian ini adalah sistem mampu mendeteksi kehadiran obyek manusia dengan waktu pendeteksian yang sangat cepat yaitu sekitar 0,00259 detik. Kemampuan pendeteksian sistem ini tak terpengaruh oleh tipe posisi obyek karena sistem ini sudah ditraining dengan berbagai posisi obyek manusia sebelumnya. Hasil pendeteksian dapat digambarkan sbb :



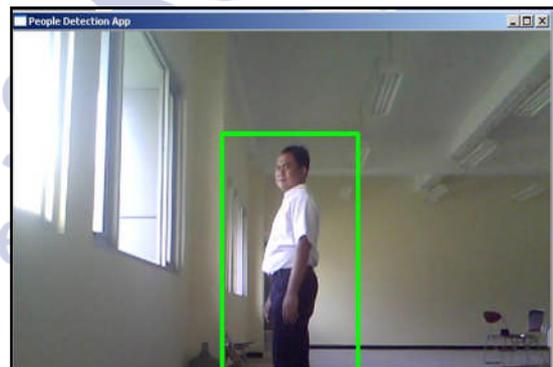
Gambar 4. Deteksi objek tunggal pada posisi 180° (depan)

Tabel 1. Hasil deteksi obyek tunggal pada posisi 180° (depan)

No.	Waktu Deteksi	Jumlah Orang
1	08:25:35	1
2	08:27:08	1
3	08:28:40	1
4	08:30:14	1
5	08:32:57	1
6	08:33:15	1
7	08:35:23	1
8	08:36:31	1
9	08:37:48	1
10	08:38:21	1

2. Posisi Obyek Tunggal tampak samping (+/- 90°)

Pada pengujian ini sistem mampu mendeteksi kehadiran obyek manusia berposisi +/- 90° (Samping) dengan waktu pendeteksian yang cukup cepat yaitu sekitar 2,85291 detik.



Gambar 5. Hasil deteksi obyek tunggal pada posisi tampak samping (+/- 90°)

Tabel 2. Hasil deteksi obyek tunggal pada posisi tampak samping (+/- 90°)

No	Waktu deteksi	Jumlah orang
1	17:05:05	1
2	17:06:48	1
3	17:08:10	1
4	17:09:38	1
5	10:11:21	1
6	10:12:55	1
7	10:14:16	1
8	10:17:31	1
9	10:19:29	1
10	10:20:52	1

3. Pengaruh Posisi objek tunggal (0°) tampak Belakang

Pengujian ini dilakukan pada posisi objek membelakangi kamera, atau obyek manusia berada pada posisi 0°. Hasil pengujian ini adalah sistem mampu mendeteksi kehadiran dengan waktu pendeteksian yang sangat cepat yaitu sekitar 0,00287 detik. Kemampuan pendeteksian sistem ini tak terpengaruh oleh tipe posisi obyek. Hasil pendeteksiannya dapat digambarkan sbb :



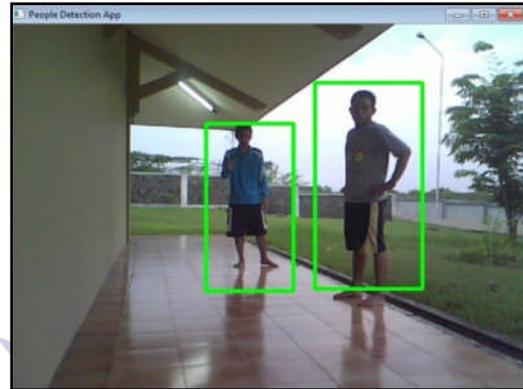
Gambar 6. Hasil deteksi obyek tunggal tampak belakang (0°)

Tabel 3. Hasil deteksi objek tunggal pada posisi 0° (belakang)

No.	Waktu deteksi	Jumlah orang
1	08:32:15	1
2	08:34:00	1
3	08:36:28	1
4	08:37:38	1
5	08:38:11	1
6	08:40:45	1
7	08:41:57	1
8	08:43:30	1
9	08:44:06	1
10	08:45:29	1

4. Pengaruh Jumlah Obyek Manusia Banyak Posisi Seragam

Pada pengujian ini sistem mampu mendeteksi kehadiran banyak obyek manusia dengan posisi yang seragam, misalnya posisi depan dengan waktu pendeteksian yang cukup cepat yaitu sekitar 2,80936 detik. Hasil pendeteksiannya dapat digambarkan sbb :



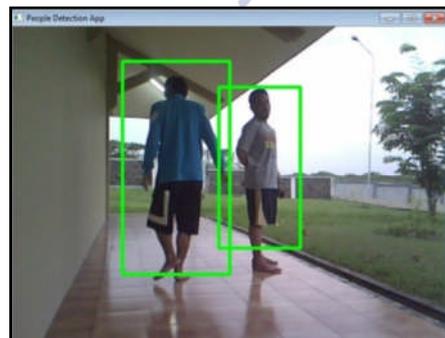
Gambar 7. Hasil deteksi dua obyek pada posisi seragam

Tabel 4. Hasil deteksi obyek pada posisi seragam

No.	Waktu deteksi	Jumlah orang
1	17:19:15	2
2	17:20:01	2
3	17:22:51	2
4	17:23:24	2
5	17:24:18	2
6	17:25:54	2
7	17:26:07	2
8	17:28:56	2
9	17:29:12	2
10	17:31:29	2

5. Pengaruh Jumlah Obyek (Manusia) Pada Posisi yang Bervariasi

Berdasarkan hasil pengujian sistem mampu mendeteksi kehadiran banyak obyek manusia dengan berbagai posisi yang bervariasi dengan waktu pendeteksian 2,80042 detik. Kemampuan pendeteksian sistem ini terpengaruh oleh jumlah dan tipe posisi obyek. Kecepatan pendeteksian ini terpengaruh oleh jumlah dan tipe posisi obyek manusia yang dideteksi. Hasil pendeteksiannya dapat digambarkan sbb :



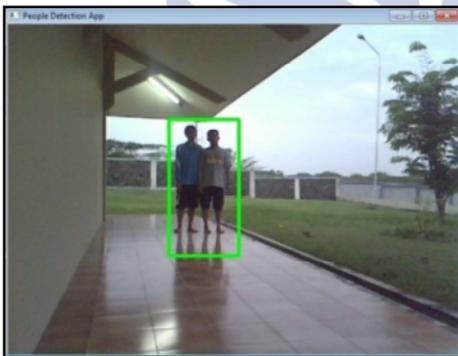
Gambar 8. Hasil deteksi 2 obyek manusia pada posisi bervariasi

Tabel 5. Hasil deteksi 2 obyek pada posisi yang bervariasi

No.	Waktu deteksi	Jumlah orang
1	17:20:42	2
2	17:21:15	2
3	17:23:28	2
4	17:24:32	2
5	17:25:46	2
6	17:27:21	2
7	17:28:07	2
8	17:30:53	2
9	17:31:14	2
10	17:35:39	2

6. Pengaruh Kerapatan Jumlah Obyek Manusia

Pengujian ini dilakukan untuk mengetahui pengaruh dari jumlah dan kerapatan obyek manusia terhadap keakurasian dan kecepatan pendeteksian obyek manusia. Berdasarkan gambar dan tabel hasil penyimpanan maka hasil dari pengujian ini adalah sistem mampu mendeteksi kehadiran obyek manusia dengan kondisi rapat dan membutuhkan waktu pendeteksian yang cukup cepat lagi yaitu sekitar 2,80936 detik. Obyek yang rapat dan saling berdekatan menyebabkan bounding box pendeteksian disatukan oleh HOG. Kecepatan pendeteksian ini terpengaruh oleh jumlah, tingkat kerapatan dan tipe posisi obyek manusia yang dideteksi. Hasil pendeteksian dapat digambarkan sbb :



Gambar 9. Hasil deteksi 2 buah obyek yang saling berimpit

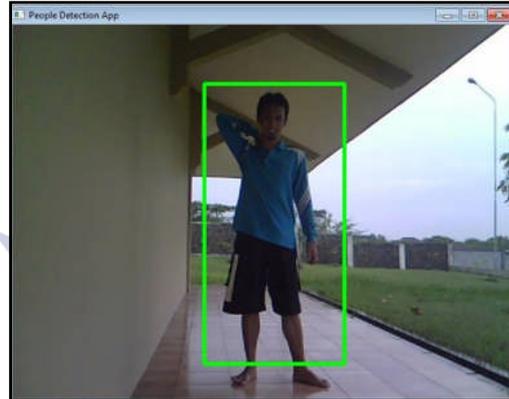
Tabel 6. Tabel hasil recording deteksi 2 obyek yang berimpit

No.	Waktu deteksi	Jumlah orang
1	17:25:57	2
2	17:30:32	2
3	17:31:21	2
4	17:33:18	2
5	17:35:45	2
6	17:38:12	2
7	17:32:06	2
8	17:35:40	2
9	17:37:36	2
10	17:40:29	2

4.1.10. Pengaruh Jarak Obyek Manusia

Pengujian ini dilakukan untuk mengetahui pengaruh dari jarak obyek manusia terhadap keakurasian dan kecepatan pendeteksian obyek manusia. Pada

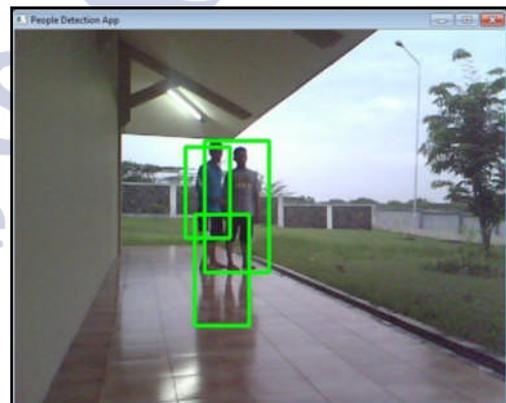
pengujian ini jarak obyek berada pada beberap posisi, Hasil pengujian pada obyek dengan jarak 2 m sistem dapat mendeteksi obyek manusia dengan baik. Sedangkan hasil pengujian pada obyek dengan jarak 8 m, sistem masih dapat mendeteksi obyek manusia dengan baik dengan kecepatan pendeteksian yang cukup cepat yaitu 2,80098 detik. Hasil pendeteksian dapat digambarkan sbb :



Gambar 10. Hasil pendeteksian obyek manusia pada jarak 2m

Tabel 7. Hasil deteksi obyek manusia pada jarak 2m

No.	Waktu deteksi	Jumlah orang
1	17:09:24	1
2	17:11:15	1
3	17:12:37	1
4	17:15:30	1
5	17:16:09	1
6	17:17:26	1
7	17:19:52	1
8	17:23:41	1
9	17:25:45	1
10	17:28:18	1



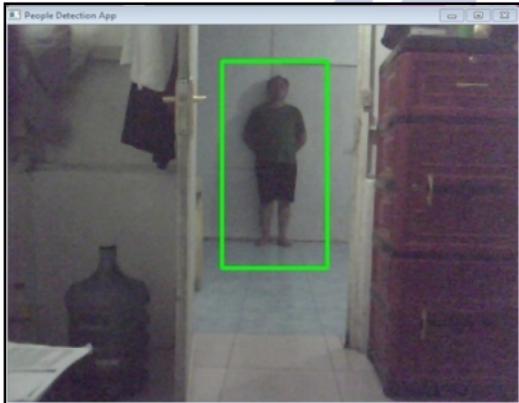
Gambar 11. Hasil pendeteksian obyek manusia pada jarak 8m

Tabel 8. Hasil deteksi obyek pada jarak 8m

No.	Waktu deteksi	Jumlah orang
1	17:37:09	2
2	17:38:25	2
3	17:39:07	2
4	17:41:30	2
5	17:43:14	2
6	17:45:58	2
7	17:46:42	2
8	17:49:13	2
9	17:50:51	2
10	17:51:35	2

7. Pengaruh Intensitas Cahaya

Pengujian ini dilakukan untuk mengetahui pengaruh intensitas cahaya terhadap keakuratan dan kecepatan pendeteksian obyek manusia. Hasil dari pengujian pada intensitas cahaya minimum adalah sistem ini masih tetap dapat mendeteksi obyek.



Gambar 12. Hasil deteksi obyek pada intensitas cahaya minimum

Tabel 4.10 Hasil deteksi pada intensitas cahaya minimum

No.	Waktu deteksi	Jumlah orang
1	20:35:54	1
2	20:37:26	1
3	20:38:45	1
4	20:39:37	1
5	20:41:12	1
6	20:42:38	1
7	20:45:24	1
8	20:46:09	1
9	20:47:43	1
10	20:49:16	1

- percobaan posisi objek tampak depan (180°), dan tampak belakang lebih cepat dibandingkan dengan objek tampak samping (90°)
- 3. Pada system ini jumlah objek tidak mempengaruhi kecepatan dan keakuratan system deteksi.
- 4. System deteksi dipengaruhi oleh intensitas cahaya dan posisi objek.

DAFTAR PUSTAKA

Dalal, N. and Triggs, B., 2005, *Histograms of oriented gradients for human detection* In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'2005)*. San Diego, CA. N.

D. M. Gavrilu, J. Giebel, and S. Munder, 2004, *Vision-based pedestrian detection: the protector+ system*. In *Proc. of the IEEE Intelligent Vehicles Symposium*. Parma, Italy.

Gary Bradsky and Adrian Kaehler., 1999, *Learning OpenCV*, 1st ed. Sebastopol : www.oreilly.com.

K. Mikolajczyk, C. Schmid, and A. Zisserman, 1993, *Human detection based on a probabilistic assembly of robust part detectors*. In *Proceedings of the 8th European Conference on Computer Vision*, volume I. Prague, Czech Republic. M. D. Yacoub, *Foundations of Mobile Radio Engineering*. Boca Raton, FL : CRC Press.

Viola, P., Jones, M. J., and Snow, D., *Detecting pedestrians using patterns of motion and appearance*, 3rd ed. In *Ninth International*

France.J. G. Proakis, 2003, *Digital Communications, Conference on Computer Vision (ICCV)*, Singapore: McGraw- Hill.

V. de Poortere, J. Cant, B. Van den Bosch, J. De Prins, F. Fransens, and L. Van Gool, 2002, *Efficient pedestrian detection : a test case for svm based categorization*. Workshop on Cognitive Vision,

X. Y. Wang, X. Han and S. C. Yan, An HOG-LBP, 2009, *human detector with partial occlusion handling*. In ICCV.

SIMPULAN

- 1. System kerja human *detection* menggunakan metode HOG, sinyal video yang ditangkap oleh webcam dicacah terlebih dahulu mejadi frame-frame gambar yang kemudian diproses oleh HOG *detection* yang bertujuan untuk memutuskan hasil deteksi apakah manusia atau bukan manusia.
- 2. Kecepatan deteksi pada system ini dipengaruhi oleh posisi objek terhadap posisi kamera. Berdasarkan hasil