

Sistem Pengawasan Berbasis Deteksi Gerak Menggunakan *Single Board Computer*

Afandi Nur Aziz Thohari¹, Rima Dias Ramadhani²

Abstract—Monitoring is the important thing for the security of an area. Based on the monitoring, the condition of the area, events, and objects can be observed. Monitoring of an area generally uses *Closed Circuit Television* (CCTV). But CCTV cameras only function as passive supervisor which are unable to detect the appearance of objects. Therefore, the motion detection techniques need to be applied to detect the appearance of the objects. In this paper, a Gaussian blur and accumulative frame difference method was applied to detect the appearance of the objects. The method works by comparing the reference frame as a benchmark with the target frame which is filled by the objects. Based on the results of the test, the system is able to detect the objects that appear by raising a segmentation line on the objects. Then, the time of objects appearance will be recorded in a *.csv file and the system visualizes the appearance of the objects in time-series graphs. Objects appearance examination at a distance of 1 to 10 meters can work well during a bright conditions. However, in dark environments (less than 40 lux), the system has not been able to detect the appearance of an object because it depends on the specification of camera used by the users. Then, in testing the number of the objects, the system can detect multiple objects. However, if there are several objects that are too close, those objects will be merged as one object.

Intisari—Pengawasan menjadi hal yang penting untuk keamanan suatu area. Berdasarkan pengawasan tersebut, dapat diketahui kondisi area, kejadian, dan objek yang muncul. Pengawasan suatu area umumnya dilakukan menggunakan *Closed Circuit Television* (CCTV). Namun, kamera CCTV hanya berfungsi sebagai pengawas pasif yang tidak mampu mendeteksi kemunculan objek. Oleh karena itu, teknik deteksi gerak perlu diterapkan untuk mendeteksi kemunculan objek. Pada makalah ini diterapkan metode *Gaussian blur* dan *accumulative frame difference* untuk mendeteksi kemunculan objek. Metode tersebut bekerja dengan membandingkan *frame* referensi sebagai patokan dengan *frame* target yang terisi objek. Berdasarkan hasil pengujian, sistem mampu untuk mendeteksi objek yang muncul dengan memunculkan garis segmentasi pada objek. Kemudian, waktu kemunculan objek akan terekam dalam sebuah berkas *.csv dan sistem memvisualisasikan kemunculan objek dalam grafik fungsi waktu. Pengujian kemunculan objek pada jarak 1 sampai 10 meter dapat bekerja dengan baik saat kondisi terang. Namun, untuk deteksi objek pada lingkungan yang gelap (kurang dari 40 lux), sistem belum mampu mendeteksi kemunculan objek

karena spesifikasi kamera yang dipakai. Kemudian pada pengujian jumlah objek, sistem dapat mendeteksi beberapa objek. Akan tetapi, jika terdapat objek yang sangat dekat jaraknya, maka akan digabung menjadi satu objek.

Kata Kunci—*frame*, deteksi, objek, pengawasan.

I. PENDAHULUAN

Sistem pengawasan berbasis deteksi gerak termasuk dalam rumpun ilmu visi komputer. Pengawasan digunakan untuk mengamati keadaan suatu area, termasuk objek dan lingkungan di sekitarnya. Sistem pengawasan biasanya digunakan untuk keamanan dalam berbagai bidang, seperti perbankan, pergudangan, perkantoran, dan berbagai macam fasilitas publik seperti bandara dan stasiun kereta, sampai tempat tinggal. Pengawasan suatu area umumnya dibantu oleh *closed circuit television* (CCTV). Namun, sistem pada CCTV hanya berfungsi sebagai pengawasan pasif. Kamera CCTV tidak dapat mengidentifikasi keadaan suatu objek dan durasi kemunculan objek tersebut.

Oleh karena itu, teknik deteksi gerak perlu diterapkan untuk memudahkan pengenalan suatu objek. Terdapat beberapa metode yang digunakan untuk deteksi gerak, seperti *background subtraction*, Sobel, *adaptive motion detection*, *accumulative difference images*, dan *frame differences* [1]. Di antara metode-metode tersebut, *frame differences* adalah metode yang umum digunakan untuk deteksi gerak secara *real-time*. Metode *frame difference* dilakukan dengan tiga langkah. Langkah pertama adalah proses menentukan gambar referensi atau *background*, langkah kedua adalah operasi *subtraction arithmetic*, dan yang terakhir adalah pengaturan *threshold* [2].

Beberapa penelitian tentang sistem keamanan menggunakan kamera video telah dilakukan. Salah satunya adalah penggunaan *IP Camera* untuk mendeteksi objek bergerak [3]. Hal tersebut dilakukan dengan menampilkan video pengawasan dalam sebuah server web secara *real-time*. Penelitian lainnya adalah deteksi gerak menggunakan *webcam* dengan metode *background subtraction* [4]. Sistem yang dibangun mampu mendeteksi objek dan mengirim pemberitahuan lewat *email*. Kemudian, penelitian yang terakhir adalah sistem pengawasan menggunakan kamera Raspberry Pi dan sensor *passive infrared* (PIR) [5]. Setiap gerakan yang terdeteksi sensor PIR dicatat kemudian pemberitahuan dikirimkan ke ponsel.

Berdasarkan penelitian-penelitian tersebut, terdapat beberapa keterbatasan yang terjadi, seperti pengawasan tidak dapat mencatat waktu kemunculan objek dan tidak dapat memvisualisasikan durasi objek dalam bentuk grafik. Oleh karena itu, makalah bermaksud membuat sistem pengawasan yang dapat mencatat tanggal dan waktu kemunculan objek.

¹Program Studi SI Rekayasa Perangkat Lunak, Fakultas Teknologi Industri dan Informatika, Institut Teknologi Telkom Purwokerto, Jl. DI Panjaitan No. 128, 53147 INDONESIA (telp: 0281-641629; afandi@ittelkom-pwt.ac.id)

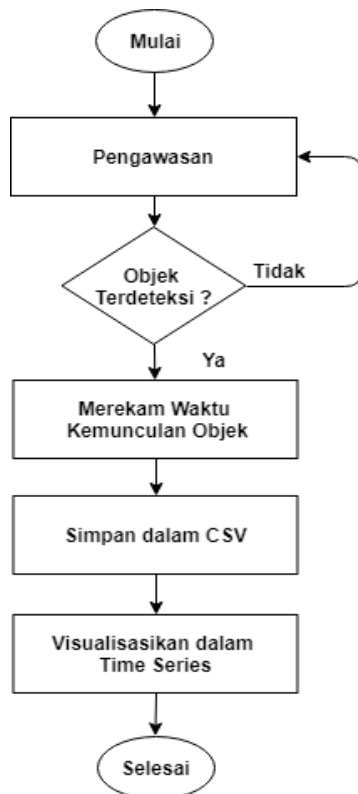
²Program Studi SI Informatika, Fakultas Teknologi Industri dan Informatika, Institut Teknologi Telkom Purwokerto, Jl. DI Panjaitan No. 128, 53147 INDONESIA (telp: 0281-641629; rima@ittelkom-pwt.ac.id)

Data kemunculan objek divisualisasikan ke dalam grafik yang mudah dibaca oleh pengguna. Pengambilan data objek dilakukan menggunakan kamera yang terhubung ke sebuah *Single Board Computer* (SBC) untuk mengolah data. Selanjutnya data yang telah diperoleh diproses dengan metode *frame difference* untuk dideteksi gerakannya. Video menampilkan garis *region of interest* (ROI) dari objek yang terdeteksi.

II. METODOLOGI

A. Rancangan Sistem

Masukan dari sistem yang dibangun pada makalah ini adalah sebuah video *real-time* yang diambil menggunakan kamera. Data dari video diproses dan dimanipulasi oleh *single board computer* (SBC). Pemrosesan citra dilakukan pada SBC menggunakan metode *accumulative frame differences*. Hasil dari pemrosesan citra adalah deteksi objek dan grafik kehadiran objek dalam fungsi waktu (*time-series*). Rancangan sistem ini ditunjukkan pada Gbr. 1.



Gbr. 1 Rancangan sistem.

Komponen utama yang digunakan untuk memproses data menjadi informasi adalah SBC. Ada berbagai macam tipe SBC, tetapi pada makalah ini digunakan Raspberry Pi untuk memproses data. Raspberry Pi dipilih karena memiliki ukuran yang kecil selayaknya kartu kredit, sehingga dapat diletakkan pada tempat yang tinggi yang dapat menjangkau seluruh area pengawasan. Selain itu, Raspberry Pi juga dapat terkoneksi dengan perangkat eksternal yang lain seperti kamera dan memiliki kemampuan menjadi server, sehingga dapat

dikendalikan dari jarak jauh dan dapat mengawasi situasi kapan pun dan di mana pun, selama terhubung dengan internet.

B. Konfigurasi Alat

Perangkat keras yang digunakan adalah Raspberry Pi 3 Model B dan Modul Raspberry Pi Camera V2 8 MP. Pemilihan Raspberry Pi 3 Model B dikarenakan versi ini sudah memiliki kemampuan yang cukup untuk melakukan komputasi algoritme pengolahan video. Selain itu, versi ini juga sudah memiliki modul Wi-Fi, sehingga dapat dikendalikan dari jarak jauh. Kamera yang digunakan sangat kompatibel dengan Raspberry Pi dan memiliki resolusi yang sudah baik, yaitu 8 MP. Tampilan perangkat keras yang telah terpasang ditunjukkan pada Gbr. 2.



Gbr. 2 Tampilan perangkat.

Sedangkan perangkat lunak dan pustaka yang harus terpasang adalah Raspbian OS, OpenCV, Pandas, dan Bokeh. Adapun fungsi dari masing-masing pustaka ditunjukkan pada Tabel I.

TABEL I
SPESISIKASI KEBUTUHAN PERANGKAT LUNAK DAN KERAS

No.	Spesifikasi Kebutuhan	Fungsi
1	Raspberry Pi 3 B+	Tempat pengolahan data
2	Camera Modul V2 8 MP	Mengambil video
3	Raspbian OS	Sistem operasi
4	OpenCV 3.4.0	Pustaka pengolahan citra
5	Pandas 0.22.0	Membaca data
6	Bokeh 0.12.13	Pustaka untuk visualisasi grafik

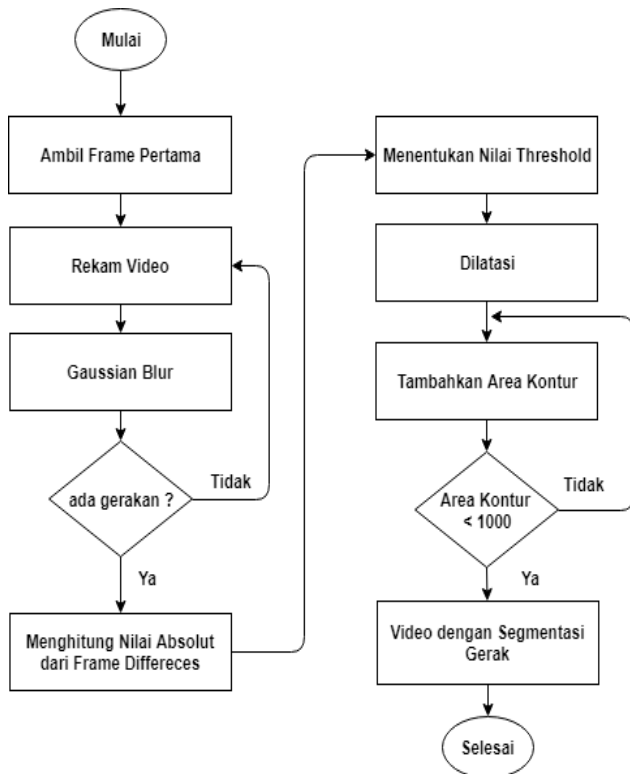
C. Metode Pemrosesan Data

Data yang diproses adalah video *real-time* dan diambil dari kamera Raspberry Pi. Data yang diperoleh diidentifikasi untuk dikenali pergerakan objeknya. Langkah-langkah deteksi gerak ditunjukkan pada Gbr. 3.

Gbr. 3 menunjukkan langkah-langkah untuk mengidentifikasi gerak dari sebuah objek. Penjelasan Gbr. 3 secara rinci adalah sebagai berikut.

1. Kamera terkoneksi ke Raspberry Pi untuk melakukan akuisisi video dan menghasilkan urutan *frame*.
2. Kamera membaca dan mengambil video secara *real-time*.

3. *Frame* yang pertama diambil digunakan sebagai *frame* referensi untuk dibandingkan dengan *frame* lainnya.
4. Mengubah video dalam bentuk *grayscale* agar mudah melakukan transformasi *frame*, sehingga objek mudah dideteksi.
5. Menerapkan teknik *Gaussian Blur* agar video yang dihasilkan memiliki tekstur yang halus.
6. Mengubah video menjadi bentuk biner (hitam putih), sehingga objek dapat secara jelas terdeteksi.
7. Beberapa *frame grayscale* kemudian dibandingkan dengan *frame* referensi dengan metode *accumulative frame differences*.
8. Nilai akumulasi perbedaan antara *frame* referensi dan *frame* hasil akuisisi dibandingkan dengan nilai *threshold*. Jika akumulasi perbedaan antara *frame* referensi dan *frame* akuisisi lebih besar dari *threshold*, maka sistem mendeteksi adanya objek dan menyeleksi objek tersebut.
9. Jika objek terdeteksi, maka sistem secara otomatis mengirim sinyal pergerakan, menyimpan video, dan mencatat waktu kehadiran objek.



Gbr. 3 Proses deteksi gerak.

D. Gaussian Blur

Gaussian blur merupakan teknik pengolahan citra yang digunakan untuk menghilangkan derau (*noise*). Teknik ini membuat citra pada video menjadi blur dan meningkatkan akurasi pendeteksian objek. Derau yang berupa titik-titik hitam dapat dihilangkan sehingga segmentasi yang dihasilkan dapat lebih akurat.

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2a^2}} \tag{1}$$

Persamaan (1) digunakan untuk menghitung atau menentukan nilai-nilai dari tiap elemen pada *Gaussian Blur* [6]. Parameter *x* dan *y* adalah koordinat dari piksel citra, sedangkan σ adalah standar deviasi yang dipakai. Hasil dari penerapan (1) adalah nilai piksel baru yang membuat tampilan objek semakin tebal.

E. Frame Difference

Pada makalah ini, pendeteksian gerak dilakukan dengan menggunakan metode *frame differences* dan *threshold*. Apabila dibandingkan dengan metode lain, *frame differences* memiliki ketelitian yang tergolong tinggi, sebab metode ini dilakukan dengan membandingkan nilai piksel citra. Teknik deteksi gerak tidak mengharuskan untuk mengenali objek yang direkam atau menelusuri gerakan objek tersebut, tetapi mengamati kemunculan suatu objek pada waktu tertentu.

Algoritme *frame differences* yang diterapkan melalui tiga tahap proses [2]. Tahap pertama adalah proses penentuan citra referensi atau citra latar (*background*), tahap kedua adalah proses operasi aritmetika *subtraction*, dan yang terakhir pengaturan ambang batas (*threshold*). *Threshold* merupakan bagian penting untuk penentuan ketelitian dari pendeteksian gerak. Nilai *threshold* yang terdeteksi berbanding terbalik dengan jumlah derau [7]. Semakin besar nilai *threshold* pada algoritme *frame differences*, semakin kecil jumlah derau yang terdeteksi. Adapun persamaan dari *frame differences* adalah sebagai berikut.

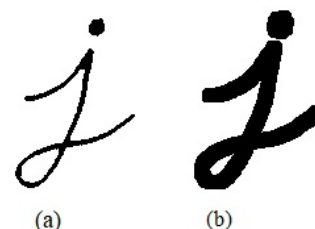
$$FD_K(x, y) = |I_k(x, y) - I_{k-1}(x, y)| \tag{2}$$

dengan *FD* adalah *frame differences*.

Mengacu pada (2), suatu *frame* yang dijadikan referensi dibandingkan dengan perubahan *frame* yang terjadi. Apabila objek muncul, maka pasti terjadi perubahan nilai piksel dari *frame*. Perubahan piksel yang terjadi direkam oleh sistem dan dibandingkan oleh lingkungan di sekitarnya (*background*).

F. Dilatasi

Dilatasi digunakan untuk menebalkan objek agar lebih mudah terdeteksi. Proses morfologi ini digunakan pula untuk memfokuskan objek agar lebih terlihat dibandingkan lingkungan sekelilingnya. Cara kerja dilatasi pada citra adalah dengan mengubah titik piksel di sekeliling garis batas objek menjadi bagian dari objek. Adapun ilustrasi dari penggunaan metode dilatasi ditunjukkan pada Gbr. 4.



Gbr. 4(a) Citra asli, (b) Citra hasil dilatasi.

G. Area Kontur

Setiap objek yang ditangkap kamera memiliki ukuran tinggi, lebar, dan juga ketebalan yang berbeda. Posisi objek yang

bergerak dapat menyebabkan terjadinya perubahan kontur. Program yang dibangun telah mampu mengidentifikasi pola kontur dari objek yang muncul. Berikut skrip program yang digunakan.

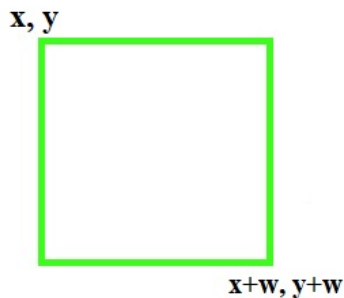
```

1. for contour in cnts:
2.   if cv2.contourArea(contour) < 1000:
3.     continue
4.   status=1
5.   cv2.rectangle(frame,(x, y), (x+w, y+h),
   0,255,0), 4)

```

Skrip nomor 1 menunjukkan proses pengulangan terhadap nilai kontur yang sebelumnya dideklarasikan pada variabel "cnts". Skrip nomor 2 dan 3 adalah suatu kondisi, apabila area kontur memiliki nilai kurang dari 1000, maka program akan berulang terus. Ketika akhirnya muncul objek, maka nilai kontur akan melebihi 1000 dan status berubah menjadi 1. Pada saat status = 1, maka objek terdeteksi dan akan disimpan dalam berkas *.csv. Kemudian, ketika terdapat objek, maka akan muncul garis persegi yang menyeleksi objek.

Skrip nomor 5 digunakan untuk membentuk sebuah *selector* yang digunakan untuk mendeteksi objek. Bentuk *selector* adalah persegi atau persegi panjang yang memiliki posisi koordinat sumbu x dan y , serta ditambah lebar (*width*) dari objek yang ditangkap. Kode (0,255,0) menghasilkan warna hijau dan nilai 4 adalah ketebalan garis dari *selector*. Ukuran *selector* akan bertambah mengikuti ukuran objek. Desain dari *selector* yang dibangun ditunjukkan pada Gbr. 5.



Gbr. 5 Desain *selector* pada objek.

III. HASIL

Pada bagian ini dibahas mengenai hasil yang telah diperoleh berdasarkan perancangan dan pengujian dari purwarupa yang telah dibangun. Pengujian terdiri atas pengujian deteksi gerak, pengujian jarak dan intensitas cahaya, dan pengujian jumlah objek yang dapat dideteksi.

A. Pengujian Deteksi Gerak

Objek yang bergerak melintasi daya jangkauan kamera akan dideteksi oleh sistem. Sistem menambahkan ROI dari objek dengan memberikan kotak warna hijau. Agar memudahkan dalam identifikasi objek, citra diubah ke dalam bentuk *grayscale* dan blur. Hal ini dikarenakan perubahan *frame* objek lebih mudah diketahui apabila berbeda dengan lingkungan di sekitar objek.

Implementasi metode *frame differences* menghasilkan segmentasi objek seperti ditunjukkan pada Gbr. 6. Ketika

objek muncul, maka secara otomatis sistem akan mengidentifikasi dan menampilkan garis ROI untuk menyeleksi objek. Kemudian, *Gaussian blur* digunakan untuk menghaluskan citra seperti yang ditunjukkan pada Gbr. 6(b). Pada Gbr. 6(c) dilakukan perubahan citra biner untuk mengetahui perbedaan piksel yang terjadi antara objek dengan lingkungan di sekitarnya.

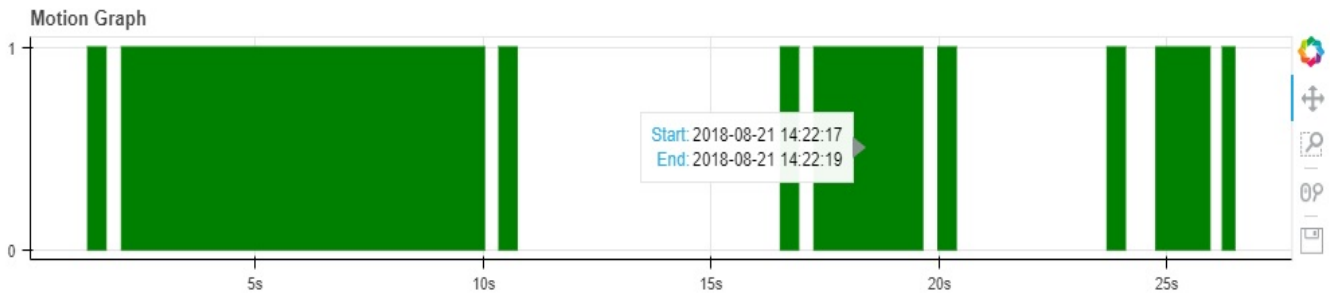


Gbr. 6(a) Deteksi objek dengan ROI, (b) Deteksi objek dengan *Gaussian blur*, (c) Deteksi objek dengan *binary threshold*.

B. Pengujian Deteksi Waktu Kehadiran Objek

Objek yang terekam oleh sistem akan dicatat waktu kemunculannya, sehingga dapat diketahui durasi dari kemunculan objek tersebut. Data rekaman tersebut disimpan ke dalam sebuah berkas dengan ekstensi *.csv. Berkas .csv muncul ketika program selesai dijalankan. Isi dari berkas .csv tersebut adalah tanggal, jam, menit, detik, dan milidetik dari kemunculan dan akhir waktu objek yang terekam.

Berdasarkan data rekaman kehadiran objek yang dicatat, dapat dibuat sebuah grafik fungsi waktu. Grafik tersebut otomatis muncul ketika program selesai dijalankan. Adapun tampilan dari grafik fungsi waktu yang terbentuk ketika program selesai dijalankan ditunjukkan pada Gbr. 7. Berdasarkan grafik tersebut, dapat diketahui pada detik ke berapa saja objek muncul. Pada tampilan grafik yang terbentuk dapat dilakukan pembesaran, sehingga dapat diketahui lebih detail detik dan milidetik kehadiran objek.



Gbr. 7 Grafik fungsi waktu kemunculan objek.

C. Pengujian Jarak dan Intensitas Cahaya

Pada pengujian ini diketahui nilai ambang batas mengenai jarak dan intensitas cahaya yang dapat ditangkap oleh kamera. Jarak objek dari kamera yang diuji adalah antara 1 sampai 10 meter, sedangkan intensitas cahaya dikategorikan ke dalam keadaan terang, redup, dan gelap. Berdasarkan *Illuminating Engineering Society* (IES), parameter untuk terang adalah lebih dari 120 lux, redup 40-119 lux, dan gelap berada pada parameter kurang dari 40 lux [8].

TABEL II
PENGUJIAN JARAK DAN INTENSITAS CAHAYA

Parameter	Terang	Redup	Gelap
	> 120 lux	40 – 119 lux	< 40 lux
1 meter	Terdeteksi	Terdeteksi	Terdeteksi
2 meter	Terdeteksi	Terdeteksi	Terdeteksi
3 meter	Terdeteksi	Terdeteksi	Tidak Terdeteksi
4 meter	Terdeteksi	Terdeteksi	Tidak Terdeteksi
5 meter	Terdeteksi	Terdeteksi	Tidak Terdeteksi
6 meter	Terdeteksi	Terdeteksi	Tidak Terdeteksi
7 meter	Terdeteksi	Terdeteksi	Tidak Terdeteksi
8 meter	Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
9 meter	Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi
10 meter	Terdeteksi	Tidak Terdeteksi	Tidak Terdeteksi

Berdasarkan Tabel II diketahui bahwa sistem tidak dapat mengidentifikasi objek dalam kondisi redup jika jarak lebih dari 7 meter. Sedangkan pada saat gelap, kamera tidak dapat mengidentifikasi objek pada jarak lebih dari 2 meter. Hal ini terjadi karena warna objek menyatu dengan lingkungan. Apabila dijadikan citra biner, maka tidak diketahui bagian yang putih (1) dan yang hitam (0).

Sedangkan pada perubahan lingkungan yang sangat ekstrem, seperti perubahan cahaya dari gelap ke terang atau sebaliknya, serta munculnya bayangan dari benda yang lewat, maka sistem dapat membedakannya. Hal ini dikarenakan sistem sudah diatur agar mengambil setiap *frame* selama 1 menit dahulu, kemudian diidentifikasi antara bayangan dan objek nyata.

D. Pengujian Deteksi Jumlah Objek

Pengujian deteksi jumlah objek dilakukan dengan melewati objek di depan kamera pada jarak 2 meter dan dengan pencahayaan yang terang. Percobaan dilakukan sebanyak sepuluh kali pada masing-masing jumlah objek. Hasil pendeteksian jumlah objek ditunjukkan pada Tabel III.

TABEL III
PENGUJIAN DETEKSI JUMLAH OBJEK

Jumlah Objek	Parameter		Akurasi
	Terdeteksi	Tidak Terdeteksi	
1	5	0	100%
2	5	0	100%
3	4	1	80%
4	3	2	60%
5	1	4	20%

Berdasarkan data pengujian jumlah objek pada Tabel III, diperoleh hasil bahwa semakin banyak objek yang terekam, maka akurasi objek yang terdeteksi semakin menurun. Terdapat objek yang tidak terdeteksi pada saat pengujian dengan jumlah objek 3, 4, dan 5. Ketika objek semakin banyak, maka sistem menggabungkan objek tersebut ke dalam satu segmen, berdasarkan kedekatan jarak dari tiap objek tersebut, sehingga yang teridentifikasi di sistem adalah gabungan dari beberapa objek yang muncul.

Objek yang ukurannya kecil akan tetap terdeteksi oleh sistem. Apabila objek tersebut bergerak dan masuk ke dalam *frame* kamera, maka akan terdeteksi. Hal ini karena terjadi perubahan nilai piksel ketika ada objek yang datang, walaupun objek tersebut kecil.

IV. KESIMPULAN

Sistem yang dibangun mampu mendeteksi gerakan sebuah objek dan mampu menampilkan visualisasi kehadiran objek dalam grafik fungsi waktu. Metode *frame differences* yang digunakan bekerja dengan membandingkan *frame* referensi dengan *frame* lain. Perbedaan nilai piksel menandakan terdapat objek yang terekam di kamera. Sedangkan *Gaussian blur* dapat menghilangkan derau pada *frame*, sehingga objek lebih mudah terdeteksi. Sistem pemantauan ini bersifat portabel, memiliki konsumsi daya yang kecil, dan dapat ditempatkan di area mana pun. Fitur tersebut muncul karena implementasi program dijalankan di Raspberry Pi yang ukurannya kecil. Sistem yang dibangun kurang maksimal untuk mendeteksi objek apabila

lingkungan dalam kondisi gelap. Kemudian, dari sisi deteksi jumlah objek, sistem akan menggabungkan objek yang jaraknya sangat dekat ke dalam satu bagian, tetapi masih tetap akurat apabila jumlah objek yang muncul kurang dari lima.

REFERENSI

- [1] D.I. Ramadhan, I.P. Sari, dan L.O. Sari, "Comparison of Background Substraction, Sobel, Adaptive Motion Detection, Frame Differences, and Accumulative Differences Images on Motion Detection," *SINERGI*, Vol. 22, No. 1, hal. 51–62, 2018.
- [2] P.L. Rosin dan T. Ellis, "Image Difference Threshold Strategies and Shadow Detection," *Proc. BMVC '95*, 1995, hal. 347-356.
- [3] M.I. Zul dan L.E. Nugroho, "Deteksi Gerak dengan Menggunakan Metode Frame Differences pada IP Camera," *Proceeding CITEE 2012*, 2012, hal. 52–56.
- [4] M. Harry, B. Pratama, A. Hidayatno, dan A. Zahra, "Menggunakan Metode Background Subtraction dengan Algoritma Gaussian Mixture Model," *TRANSIENT*, Vol. 6, No. 2, hal. 246–253, 2017.
- [5] R. Zakaria, "Smart Motion Detection: Security System Using Raspberry Pi," *J. Eng. Res. Inst.*, Vol. 30, hal. 1-8, 2017.
- [6] U. Ahmad, *Pengolahan Citra Digital & Teknik Pemrogramannya*, 1st ed., Yogyakarta, Indonesia: Graha Ilmu, 2005.
- [7] P. Gil, S. Maldonado, dan R. Gil, "Background Pixel Classification for Motion Detection in Video Image Sequences," *Neural Networks*, Vol. 1, hal. 718–725, 2003.
- [8] D.L. DiLaura, *An Introduction to the IES Lighting Handbook*, 10th ed., New York, USA: IES, 2011.