

PERANCANGAN APLIKASI DETEKSI GERAK DAN PERANGKAT PEMANTAUAN MENGGUNAKAN RASPBERRY PI DAN IP CAM

DESIGNING APPLICATION OF MOTION DETECTION AND MONITORING DEVICE USING RASPBERRY PI AND IP CAM

Wite Kurnia¹, Dr.-ing Fiky Yosef.S, S.T., M.T.², Ramdhan Nugraha, S.Pd, M.T.³

Prodi S1 Teknik Elektro, Fakultas Teknik Elektro, Universitas Telkom
Jl. Telekomunikasi, Dayeuhkolot Bandung 40257 Indonesia

¹witekurnia@student.telkomuniversity.ac.id, ²fikyosefs@telkomuniversity.ac.id,

³Ramdhannugraha@telkomuniversity.ac.id

Abstrak

Teknologi berkembang sangat pesat, seperti sistem pengawasan pada kamera. Dengan memanfaatkan teknologi *image processing* kita dapat membuat sistem dimana dapat mengawasi sebuah ruangan yang dapat mendeteksi gerakan dan mengirim pemberitahuan pada *mobile device* pemilik tempat tersebut. Sistem keamanan dibuat dengan algoritma *optical flow* ketika membandingkan *frame* untuk mendeteksi gerakan. Langkah yang digunakan pada deteksi ini yaitu *pre-processing*, *optical flow* dan *threshold* pada gambar. Hasil dari aplikasi berhasil menyimpan video hasil *capture* ketika gerakan terjadi lalu menyimpan pada *cloud storage* dan memberikan pemberitahuan. Hasil pengujian pada sistem menghasilkan nilai akurasi (*Percentage Correct Classification*) sebesar 83%, area yang benar terdeteksi (*True Positive Rate*) sebesar 0.97, nilai presisi (*F1 score*) sebesar 0.86 dan nilai kegagalan sistem (*False Alarm Rate*) sebesar 0.22, serta nilai rasio keberhasilan pesan pemberitahuan sebesar 98%.

Kata Kunci: Kamera pengawas, *imager processing*, *optical flow*.

Abstract

Technology developed very fast, like surveillance system on camera. With utilize image processing technology we can create system where we able to monitoring room with motion detection and send notification to user mobile device. Security system built with optical flow algorithm when compared frames to detect motion. The used step on this motion is pre-processing, optical flow and threshold on frames. Result from application is capable to save video motion then upload to cloud storage and send notification to mobile device. The amount of test result for accuracy (percentage correct classification) point is 83%, amount of true contour detected (True Positive Rate) is 0.96, amount of precision (F1 score) is 0.85 and amount of system failure (False Alarm Rate) is 0.22, also amount ratio of successfully notification system is 98%.

Keywords: *surveillance camera, image processing, optical flow.*

1. Pendahuluan

Teknologi berkembang dengan cepat. Dimana banyak kemudahan yang kita dapat dengan teknologi yang ada saat ini. Salah satunya, sistem keamanan yang sedang berkembang. Kini kebutuhan akan sistem kewanaman meningkat di berbagai sektor seperti pergudangan, perkantoran dan perumahan. Kewanaman sistem ini dibuat untuk meningkatkan bagian kewanaman dan produkifitas. Sistem kewanaman menggunakan deteksi gerak ini cocok digunakan untuk kewanaman rumah [1].

Sistem kewanaman ini memberikan solusi di berbagai sektor kewanaman. Dimana sistem akan selalu mengawasi secara berkala akan merekam segala aktivitas dan ketika terjadi sesuatu hal maka dapat segera diketahui dan ditangani dengan cepat. Penerapan sistem kewanaman ini juga memudahkan pengguna mengakses kejadian yang ada di rumah, kantor, gedung, hingga sarana

publik secara langsung. Dengan adanya sistem keamanan seperti ini, maka diharapkan adanya pengawasan pada suatu objek yang bergerak secara *real-time* [2].

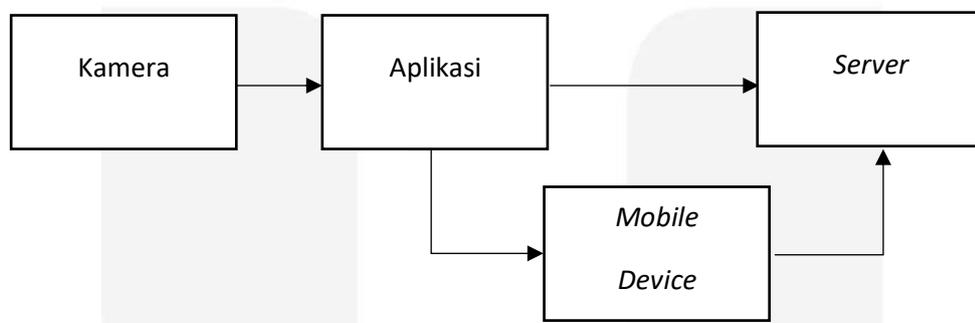
IP camera adalah salah satu perangkat elektronik yang banyak digunakan sebagai kamera pengawas. *IP camera* telah banyak digunakan sebagai kamera pengawas untuk rumah, bangunan, dan daerah yang membutuhkan pengawasan. *IP camera* juga telah dirancang untuk dapat berkomunikasi langsung melalui jaringan internet. Apabila terjadi sesuatu atau kejadian pada tempat yang diawasi, pemilik tempat tersebut tidak dapat segera mengetahui apa yang terjadi, sebelum melihat rekaman dari *IP camera*. Beberapa hal yang mempengaruhi hasil pendeteksian menggunakan *IP camera* ialah teknik penentuan citra referensi, resolusi citra, dan penggunaan *threshold* [1] [2].

Pada tugas akhir ini penulis membuat sebuah sistem yang memudahkan pengguna melakukan pengawasan. Sistem keamanan yang dirancang pada penelitian ini yaitu membuat sebuah aplikasi sistem keamanan yang terintegrasi dengan *IP camera* dan menggunakan algoritma *optical flow* pada sistem keamanan. Keunggulan dari *optical flow* yaitu memiliki akurasi yang tinggi untuk menangkap objek yang bergerak cepat [3]. Kemudian, apabila terdeteksi gerakan maka sistem akan langsung menyimpan gambar hasil deteksi gerak tersebut pada *cloud server*. Setelah itu, sistem akan mengirimkan pemberitahuan ke pengguna melalui aplikasi pada *mobile device* pengguna. Dengan demikian, sistem yang baru ini akan membantu pengguna untuk mengantisipasi kejadian-kejadian buruk dan mengatasinya dengan cepat.

2. Dasar Teori /Material dan Metodologi/perancangan

2.1 Konsep Solusi

Konsep solusi untuk mendapatkan sistem kamera pengawas yang dapat memantau dan memberikan pemberitahuan pada *mobile device* pengguna direpresentasikan dalam diagram blok pada Gambar 2.1



Gambar 2.1 Diagram Blok Sistem Kamera Pengawas

Prinsip kerja dari sistem kamera pengawas adalah sebagai berikut:

1. *Ip camera* yang sudah terpasang di dalam ruangan mulai mengambil gambar dan hasil gambar diproses pada Aplikasi.
2. Gerakan yang terjadi pada kamera akan memicu Aplikasi untuk merekam gambar sekaligus mengirim pesan pemberitahuan ketika gerakan terjadi.
3. Aplikasi berhenti merekam gambar ketika gerakan pada kamera berakhir.
4. Gambar yang tertangkap akan disimpan pada *storage*.
5. Hasil dari gambar yang tersimpan di *storage* dapat diakses pada *device*.

2.2 Computer Vision

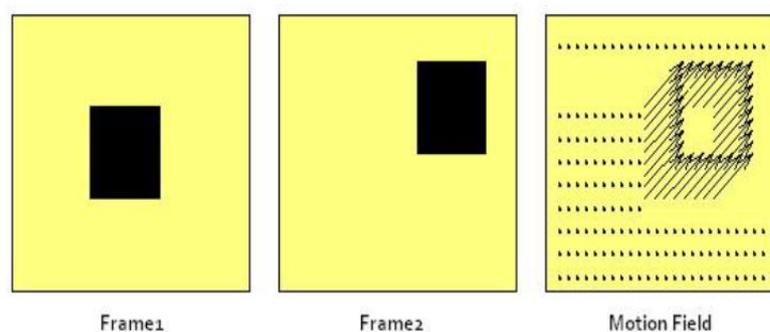
Computer vision adalah suatu ilmu dimana mesin dapat memproses dan mengekstrak gambar-gambar. *Computer vision* merupakan perpaduan antara pengolahan gambar dan pengenalan pola. Pengolahan gambar adalah proses mendapatkan gambar yang diolah seperti yang diinginkan dan pengenalan pola adalah proses untuk mengekstrak informasi dari gambar tersebut. Tidak ada batasan yang jelas antara *image processing* dan *computer vision*, bagaimana pun terdapat 3 jenis pengolahan pada hal tersebut yaitu: *low-level*, *mid-level* dan *high-level* dalam proses [4].

2.3 Image Processing

Image processing atau pengolahan citra merupakan proses transformasi gambar atau citra dengan melakukan masukan berupa gambar dan setelah diproses hasilnya berupa gambar juga. Awalnya pengolahan citra ini dilakukan untuk memperbaiki kualitas dari gambar atau citra. Dengan berkembangnya dunia komputasi serta munculnya ilmu-ilmu komputer yang dapat mengolah citra lebih jauh, maka *image processing* tidak dapat dipisahkan dari bidang *computer vision*. Citra atau gambar adalah kumpulan warna yang memiliki pola, berbentuk abstrak dan lain sebagainya. Dari segi ilmiah, citra merupakan gambar 3 dimensi dari suatu fungsi, biasanya intensitas warna sebagai fungsi *spatial x* dan *y*. Pada komputer, warna dapat dinyatakan sebagai angka dalam bentuk sekala RGB dan dapat diproses secara digital [2].

2.4 Optical Flow

Optical Flow merupakan suatu metode yang melakukan *tracking* objek atau menghitung perpindahan suatu objek. Melihat perubahan kecerahan suatu titik adalah konsep dasar pada *optical flow*. Pada frame atau gambar selanjutnya akan dibandingkan dengan kecerahan suatu titik. Dari hasil tersebut maka didapatkan nilai kecerahan yang menunjukkan posisi titik pada gambar selanjutnya atau waktu yang berbeda. Deteksi objek dengan metode *optical flow* dilakukan dengan mengubah gerakan ke dalam aliran vektor dan menghitung nilai koordinat centroid untuk menghitung kecepatan objek yang teridentifikasi [6].



Gambar 2.4 contoh aliran vector.

Pada kecerahan diamati dari setiap nilai titik objek konstan dari waktu ke waktu dan menentukan nilai titik terdekat pada bidang gambar bergerak dengan cara yang sama. Misalkan $F(x, y, t)$ dan $F(x + \Delta x, y + \Delta y, t + \Delta t)$ adalah intensitas piksel dalam dua frame berturut-turut yang diperoleh pada interval waktu Δt dengan posisi (x, y, t) dan $(x + dx, y + dy, t + dt)$ pada masing-masing nilai [7]. Sebagian besar asumsi yang dilakukan dari metode ini bahwa intensitas piksel bergerak konstan pada frame berikutnya, yaitu dengan asumsi nilai kecerahan konstan [8].

$$F(x, y, t) = F(x + \Delta x, y + \Delta y, t + \Delta t) \quad (2.5)$$

Dengan tidak mempertimbangkan istilah nonlinier, persamaan (2.5) dapat diperluas dengan bantuan metode perluasan Taylor dan dapat dinyatakan seperti:

$$I_x u + I_y v + I_t = 0 \quad (2.6)$$

Dimana (u, v) menyatakan vektor aliran optik $(\Delta x / \Delta t, \Delta y / \Delta t)$ dan (I_x, I_y, I_t)

Menyatakan turunan dari intensitas gambar pada kordinat (x, y, t) . Untuk mendapatkan gerak vektor untuk deteksi gerak, digunakan teknik *thresholding* pada nilai (u, v) . Pada *thresholding* ini, besaran gerak vektor dari $TH = \sqrt{u^2 + v^2}$ ditemukan dan diterapkan pada nilai TH . Dengan bantuan operasi *morphological* deteksi gerak dapat terdeteksi.

2.5 IP Camera

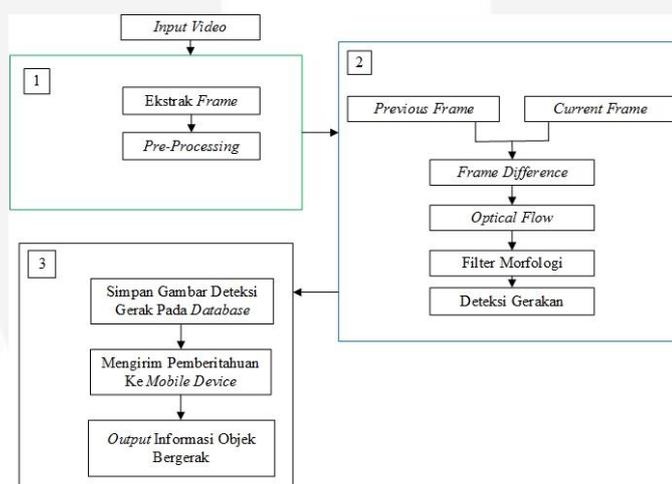
IP camera adalah sebuah kamera video digital berbasis *IP* yang digunakan sebagai kamera pengawas. *IP camera* dapat diibaratkan seperti sebuah komputer dan kamera digabungkan menjadi satu karena memiliki beberapa kesamaan. Pada *IP camera* terdapat beberapa komponen seperti sensor, lensa, *memory*, beberapa *processor* dan memiliki alamat *IP*.

Processor pada *IP camera* digunakan untuk kebutuhan seperti *video analysis*, *networking*, *compression* dan *image processing*. *IP camera* menyediakan fungsi *e-mail*, *FTP (File Transfer Protocol)*, *security protocol*, *webserver* dan masih banyak lagi protokol untuk *IP network*. *IP camera* juga memiliki kelebihan seperti kemampuan untuk memproses *visual* dan *audio* serta bisa diakses pada komputer secara langsung melalui *ethernet* dan internet. *IP camera* dapat dihubungkan langsung pada komputer dan pada saat yang sama dapat dihubungkan juga pada *router* atau perangkat lain yang bisa terhubung ke komputer. Kamera ini sangat baik karena fokus pada hasil gambar untuk pengawasan [9].

3. Pembahasan

3.1 Desain Sistem

cara kerja pada sistem ini pertama adalah menghubungkan *IP camera* pada aplikasi yang dibuat pada komputer. Setelah terhubung, *IP camera* akan melakukan *video stream* dan akan ditampilkan pada web server. Rekaman dari *IP camera* akan mulai mendeteksi gerakan dan akan ditampilkan secara *real time*.



Gambar 3.1 Desain Sistem.

Gerakan akan dideteksi dengan melakukan pengolahan gambar yang dimasukkan pada sistem. Proses video akan dilakukan oleh sistem ketika terdeteksi pergerakan pada rekaman. Ketika terdeteksi objek yang bergerak, *IP camera* akan memulai proses *capture* dan menyimpannya pada komputer. Setelah itu hasil *capture* akan dikirim ke *Android device* menggunakan *File Transfer Protocol (FTP)* dan akan membuat pemberitahuan pada *device* tersebut.

3.2.1 Pre-Processing

Setelah konversi dari video ke bentuk *frame* dilakukan, selanjutnya setiap *frame* dilakukan proses *preprocessing*. Tahap ini dilakukan sebelum memasuki proses awal yang berguna untuk mengurangi *noise*. Pertama mengubah *frame* berupa RGB ke *grayscale*. Perubahan ini diperlukan karena untuk melakukan proses *frame* dengan *optical flow* harus dalam bentuk HSV. Karena metode ini mengandalkan intensitas cahaya pada gambar maka dilakukan proses *grayscale* untuk memaksimalkan nilai cahaya. Setelah gambar berubah menjadi *grayscale*, maka gambar tersebut diubah lagi ke bentuk HSV untuk memenuhi syarat penggunaan metode *optical flow*.

3.2.2 Optical Flow

Pada proses ini *frame* yang telah berubah warna menjadi HSV diambil untuk dibandingkan dengan *frame* sebelumnya atau *background frame*, diasumsikan gambar I_1 dan I_2 . Contohnya jika gambar I memiliki ukuran 144×176 dan dibuat menjadi 4 tingkatan menjadi 72×88 , 36×44 , 18×22 , dan 9×11 . Setelah menentukan nilai tingkatan, derivatif parsial *Gaussian* dari komponen x , y dan t telah terhitung. Ketika semua komponen telah dihitung, nilai u dan v diperbarui seperti nilai intensitas pada *frame* sebelumnya. Setelah itu hasil dari *frame* yang berhasil diproses akan menjadi *frame* dengan perbedaan antara *background* dan *foreground* dalam bentuk HSV. Sehingga perbedaan *frame* yang didapat masih bernilai HSV dan perlu kita ubah ke nilai *grayscale* untuk melakukan proses selanjutnya.

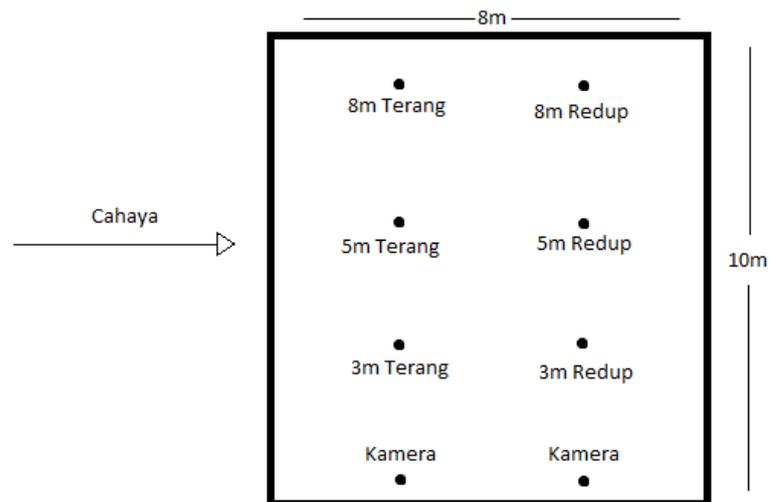
3.2.3 Segmentasi

Untuk memulai proses ini dibutuhkan gambar berupa *grayscale*. Gambar hasil dari *optical flow* masih berupa HSV, maka kita mengubahnya menjadi *grayscale* terlebih dahulu.

4. Hasil dan Analisis

4.1 Skenario Pengujian

Pada pengujian ini dilakukan dua skenario dimana skenario ini dilakukan pada ruangan 10×8 meter. skenario pertama dilakukan pada ruangan cahaya terang dan skenario kedua dilakukan pada ruangan dengan cahaya redup.



Gambar 4.1 Denah Lokasi Pengujian

Gambar di atas menunjukkan posisi skenario pada saat pengujian dilakukan. Terdapat tiga titik pengujian pada skenario cahaya redup dan cahaya terang. Pengujian ini menggunakan matahari sebagai sumber cahaya pada saat skenario cahaya terang dan menggunakan sisi ruangan yang kurang terkena sinar matahari sebagai skenario cahaya redup.

4.2 Analisis dan Hasil Pengujian Sistem.

4.2.1 Pengujian Parameter True Positive Rate (TPR)

Tujuan pengujian ini untuk mengevaluasi klasifikasi rasio dari jumlah piksel *foreground* yang terdeteksi benar terhadap jumlah piksel *foreground* pada *ground truth* (GT). Hasil yang diinginkan adalah hasil nilai TPR yang tinggi, karena hal tersebut menunjukkan hasil jumlah piksel *true positive* (TP) jauh lebih banyak dibandingkan dengan jumlah piksel *false negative* (FN).

Tabel 4.1 Hasil Pengujian True Positive Rate

Kondisi	Jarak	Lux Meter	Jumlah Sampel	<i>True Positive</i>	<i>False Negative</i>	Total Piksel	<i>True Positive Rate</i>
Redup	3 m	52	100	233	22	255	0,91
	5 m	176	100	210	15	225	0,93
	8 m	164	100	137	17	154	0,89
Terang	3 m	298	100	212	9	221	0,96
	5 m	320	100	174	5	179	0,97
	8 m	380	100	124	5	129	0,96

4.2.2 Pengujian Parameter Percentage Correct Classification (PCC)

Pada tahap ini pengujian dilakukan untuk melihat akurasi deteksi. Pada pengujian ini dilakukan skenario video percobaan dengan resolusi standar IP kamera yaitu 640x480 piksel, yang dilakukan di dalam ruangan. Jumlah sampel yang diambil adalah sebanyak n gambar, masing

masing n gambar sampel *foreground* dan n sampel *background*. Data yang digunakan adalah *true positive* (TP), *false negative* (FN), *true negative* (TN), *false positive* (FP).

Tujuan pengujian ini untuk pengukuran statistic dimana proses segmentasi dapat mendeteksi piksel *foreground*. Hasil yang diinginkan adalah hasil *percentage correct classification* (PCC) yang tinggi, karena hal tersebut menunjukkan bahwa presentasi jumlah piksel *foreground* yang terdeteksi *true positive* (TP) jauh lebih banyak dibandingkan dengan jumlah piksel lainnya.

Tabel 4.2 Hasil Pengujian Percentage Correct Classification

Kondisi	Jarak	Lux Meter	Jumlah Sampel	Total Piksel	<i>True Positive</i>	<i>False Negative</i>	<i>False Positive</i>	<i>True Negative</i>	<i>Percentage Correct Classification (%)</i>
Redup	3 m	52	100	453	233	22	98	100	74
	5 m	176	100	390	210	15	65	100	79
	8 m	164	100	293	137	17	39	100	81
Terang	3 m	298	100	418	212	9	97	100	75
	5 m	320	100	333	174	5	54	100	82
	8 m	380	100	269	124	5	40	100	83

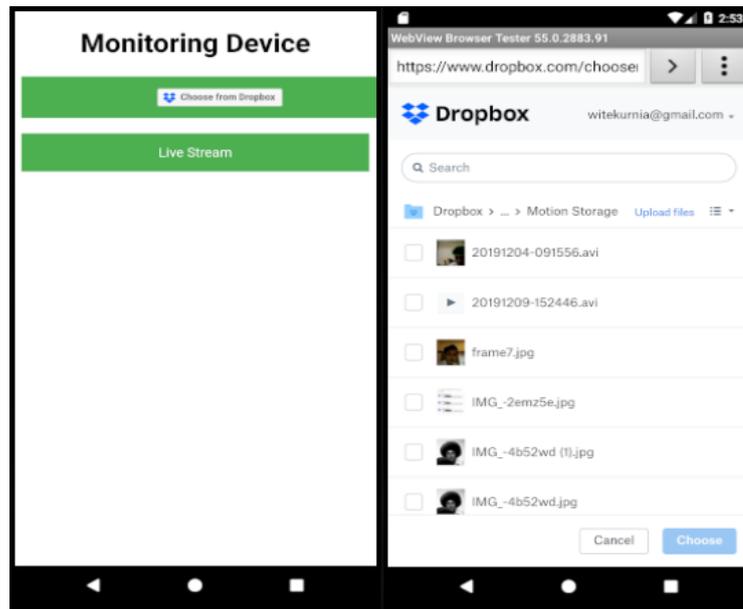
4.2.3 Pengujian Parameter F1 Score

Ditahap ini pengujian dilakukan sama seperti sebelumnya dengan kondisi yang sama. Namun data yang digunakan untuk pengujian ini adalah data *postivie predictive value* (PPV) dan *true predictive value* (TPR) yang sudah didapatkan sebelumnya, untuk mencari nilai *F1 score*. Tujuan dilakukannya pengujian ini untuk mengukur akurasi dari sistem deteksi gerak yang sudah dirancang. Dimana hasil *F1 score* terbaik adalah yang mencapai nilai 1 dan hasil terburuk adalah bernilai 0. Hasil yang diinginkan adalah nilai *F1 score* $\geq 80\%$.

Tabel 4.3 Hasil Pengujian F1 Score

Kondisi	Jarak	Lux Meter	Jumlah Sampel	<i>Positive Predictive Value</i>	<i>True Positive Rate</i>	<i>F1 Score</i>
Redup	3 m	52	100	0,70	0,91	0,80
	5 m	176	100	0,76	0,93	0,84
	8 m	164	100	0,78	0,89	0,83
Terang	3 m	298	100	0,69	0,96	0,80
	5 m	320	100	0,76	0,97	0,86
	8 m	380	100	0,76	0,96	0,85

4.3 Hasil Aplikasi Mobile Device



Gambar 4.3 User Interface Android

Aplikasi Android pada *mobile device* terdapat beberapa fitur seperti *cloud storage*, *live stream* dan *push notification*. Untuk *cloud storage* itu sendiri saya menggunakan media Dropbox sebagai *storage* aplikasi untuk kamera sehingga *user* dapat mengunduh file atau video output dari aplikasi *motion detection*.

5. Kesimpulan

Berdasarkan pengujian yang telah dilakukan pada sistem deteksi gerak pada perangkat pemantauan dengan metode *optical flow* menggunakan *IP Camera*, didapat kesimpulan:

1. Pengujian skenario pertama pada pencahayaan terang dimana nilai Lux meter 298 – 380 dengan hasil terbaik memiliki nilai akurasi (Percentage Correct Classification) sebesar 83% pada jarak 8 meter, area yang benar terdeteksi (True Positive Rate) sebesar 0.97 pada jarak 5 meter, nilai presisi (F1 score) sebesar 0.86 pada jarak 5 meter dan nilai kegagalan sistem (False Alarm Rate) sebesar 0.24 pada jarak 5 meter dan 8 meter.
2. Pengujian skenario kedua pada pencahayaan redup dimana nilai Lux meter 52 - 164 dengan hasil terbaik memiliki nilai akurasi (Percentage Correct Classification) sebesar 81% pada jarak 8 meter, area yang benar terdeteksi (True Positive Rate) sebesar 0.93 pada jarak 5 meter, nilai presisi (F1 score) sebesar 0.84 pada jarak 5 meter dan nilai kegagalan sistem (False Alarm Rate) sebesar 0.22 pada jarak 8 meter. Sistem transmisi data aplikasi menggunakan *web server* sebagai penerima respon dari *cloud storage* dan mengirimkan pemberitahuan pada aplikasi android menggunakan *cloud messaging service*. Menggunakan *port forwarding* sebagai akses untuk melihat *livestream* pada kamera.
3. Hasil terbaik dalam implementasi sistem deteksi gerak menggunakan metode *optical flow* dengan *IP camera* ini menghasilkan nilai akurasi (Percentage Correct Classification) sebesar 83%, area yang benar terdeteksi (True Positive Rate) sebesar 0.97, nilai presisi (F1 score) sebesar 0.86 dan nilai kegagalan sistem (False Alarm Rate) sebesar 0.22.
4. Pengujian skenario ketiga untuk pesan pemberitahuan pada sistem deteksi gerak device mendapatkan hasil rasio 98% menggunakan koneksi WiFi dan hasil rasio 92% menggunakan koneksi mobile data.

Daftar Pustaka:

- [1] M. I. Zul, Widyawan, and L. E. Nugroho, "Deteksi Gerak dengan Menggunakan Metode Frame Differences pada IP Camera," no. 2, 2012.
- [2] H. Mulyawan, M. Z. H. Samsono, and Setiawardhana, "Identifikasi Dan Tracking Objek

- Berbasis Image Processing” pp. 1–5, 2011.
- [3] D. Arioputra, “Analisa perbandingan akurasi metode optical flow dan gaussian mixture model untuk sistem pemantau lalu lintas berbasis computer vision.” Universitas Indonesia, 2012.
- [4] R. Gonzalez and R. Woods, *Digital image processing*. 2002.
- [5] V. Asari, *Wide Area Surveillance*. 2013.
- [6] E. Sutanty, “Analisis Perbandingan Algoritma Optical Flow dan Background Estimation untuk Pendeteksian Objek pada Video,” vol. 15, pp. 15–21, 2016.
- [7] S. S. Sengar and S. Mukhopadhyay, “Motion detection using block based bi-directional optical flow method,” *J. Vis. Commun. Image Represent.*, vol. 49, no. August, pp. 89–103, 2017.
- [8] D. Jansari and S. Parmar, “Novel Object Detection Method Based On Optical Flow,” pp. 197–201, 2013.
- [9] C. H. Kuo, H. M. Hsu, S. C. Ho, and W. T. Lee, “Universal Middleware Bridge System for IP cam networking,” *ISNE 2013 - IEEE Int. Symp. Next-Generation Electron. 2013*, pp. 291–295, 2013.
- [10] R. C. H. Silitonga, “Motion Detection with Background Substraction Methods Using an IP Camera.” Telkom University, 2017.
- [11] H. Nazruddin Safaat, *Android Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android*, Bandung: Informatika, 2012.
- [12] K. Grgić, I. Špeh and I. Heđi, "A web-based IoT solution for monitoring data using MQTT protocol," *2016 International Conference on Smart Systems and Technologies (SST)*, Osijek, 2016
- [13] J. Renno, N. Lazarevic-McManus, D. Makris and G. Jones, "*Evaluating Motion Detection Algorithms: Issues and Results*," *scienceandresearch*.