

Sistem Penghitung Jumlah Pengguna pada Ruang Kerja menggunakan *Background Subtraction*

Nico Kurniawan, Lily Puspa Dewi, Henry Novianus Palit
Program Studi Informatika, Fakultas Teknologi Industri, Universitas Kristen Petra
Jln. Siwalankerto 121-131 Surabaya 60236
Telp. (031) – 2983455, Fax. (031) - 8417658

nicokurniawan123@gmail.com, lily@peter.petra.ac.id, hnpalit@petra.ac.id

ABSTRAK

Pusat komputer di Universitas Kristen Petra belum bisa mengukur utilisasi dari ruang laptop, sehingga tidak dapat mengetahui jumlah pengguna yang sedang memakai fasilitas ruang laptop. Meskipun sudah adanya sistem registrasi dalam menggunakan fasilitas tersebut, jumlah daftar registrasi mahasiswa dengan jumlah pengguna yang memakai ruang laptop tersebut tidak sesuai karena ada mahasiswa yang tidak melakukan registrasi terlebih dahulu. Untuk itulah dirancang sistem penghitung jumlah pengguna menggunakan *background subtraction* dengan metode MOG2 dan GMG supaya dapat membantu mengetahui jumlah orang yang menggunakan fasilitas ruang laptop.

Pada skripsi ini akan membuat program penghitung jumlah pengguna dengan melakukan pendeteksian orang, serta membuat sistem program *website* untuk menampilkan data jumlah pengguna. Sistem ini dibuat dengan menggunakan bahasa C++, serta dibuat *website* menggunakan bahasa PHP dan database MySQL

Hasil dari skripsi ini menunjukkan bahwa faktor pencahayaan dan resolusi *frame* sangat berpengaruh terhadap keakuratan pendeteksi pengguna. Pengaruhnya pencahayaan pendeteksian dilihat dari hasil *confusion matrix* yang menunjukkan bahwa yang terdeteksi kebanyakan melebihi dari jumlah pengguna yang berada ruangan. Sedangkan pengaruhnya resolusi terhadap pendeteksian dilihat dari hasil *F1 score* pada salah satu video pencahayaan yang menunjukkan dengan menggunakan resolusi 1280x720 hasil metode MOG2 mencapai 68% dan metode GMG mencapai 40%. Setelah melakukan pengcilan *frame* ke 640x360, nilai rata-rata dari MOG2 mencapai 86% dan metode GMG mencapai 67%.

Kata Kunci: Perhitungan Orang, Pengurangan Latar Belakang, OpenCV.

ABSTRACT

The computer center at Petra Christian University has not been able to measure the utilization of laptop space, so it cannot know the number of users who are using laptop space facilities. Despite the existence of a registration system in using these facilities, the number of student registration lists with the number of users who use the laptop room is not appropriate because there are students who do not register first. For this reason, the system for calculating the number of users uses background subtraction using the MOG2 and GMG methods so

that it can help find out the number of people who use laptop room facilities.

In this study will create a program to count the number of users by detecting people, and create a website program system to display data on the number of users. This system was created using C ++, and a website was created using PHP and MySQL databases

The results of this study indicate that the lighting factor and frame resolution are very influential on the accuracy of the user's detection. The effect of detection lighting can be seen from the results of the confusion matrix which shows that what is detected mostly exceeds the number of users in the room. While the effect of resolution on detection can be seen from the results of F1 scores on one of the lighting videos that showed using a resolution of 1280x720 the results of the MOG2 method reached 68% and the GMG method reached 40%. After reducing the frame to 640x360, the average value of MOG2 reaches 86% and the GMG method reaches 67%.

Keywords: People Counting, Background Subtraction, OpenCV.

1. PENDAHULUAN

Teknologi dari awal hingga saat ini sudah mulai berkembang dan semakin canggih untuk digunakan, sehingga banyak manusia mulai memakai teknologi dalam melakukan suatu hal kegiatan hingga menjadi sebuah kebutuhan dalam sehari-hari. Bahkan dengan menggunakan teknologi, Manusia juga membuat berbagai macam teknologi dalam hal keamanan atau pun membuat laporan dengan teknologi tersebut.

Pusat komputer di Universitas Kristen Petra memiliki fasilitas ruang laptop yang disediakan meja laptop dan internet yang stabil. Dengan itu, mahasiswa yang membawa laptop dapat menggunakan fasilitas tersebut dalam melakukan suatu kegiatan dengan laptopnya, baik dalam hal melakukan pengerjaan tugas, hiburan, dan lainnya. Akan tetapi, pusat komputer di Universitas Kristen Petra belum bisa mengukur utilisasi dari ruang laptop, sehingga tidak dapat mengetahui jumlah pengguna yang sedang memakai fasilitas ruang laptop. Utilisasi merupakan pemanfaatan dengan tindakan menggunakan sesuatu. Istilah ini mengacu pada penggunaan sesuatu atau proses menggunakannya secara praktis. Meskipun sudah adanya sistem registrasi dalam menggunakan fasilitas tersebut, jumlah daftar registrasi mahasiswa dengan jumlah pengguna yang terpakai di ruang laptop tersebut tidak sesuai dikarenakan adanya mahasiswa yang

menggunakan fasilitas ruang laptop tanpa melakukan registrasi terlebih dahulu. Karena itu, dibutuhkan sebuah aplikasi untuk mengatasi permasalahan tersebut.

Untuk mengetahui jumlah pengguna yang terpakai, dilakukan perhitungan jumlah orang dengan pendeteksi orang untuk mengetahui berapa banyak orang yang berada di fasilitas tersebut.

2. LANDASAN TEORI

2.1 Webcam Logitech HD C310

Webcam (Web Camera) merupakan perangkat yang berupa sebuah kamera digital yang dihubungkan ke komputer atau laptop. Layaknya kamera pada umumnya, sebuah webcam dapat mengirimkan gambar-gambar secara *live* dari manapun ia berada ke seluruh penjuru dunia dengan bantuan internet.[3] Fungsi yang digunakan dalam memakai webcam yaitu untuk berkomunikasi, belajar jarak jauh, dan konferensi jarak jauh.

2.2 OpenCV

OpenCV adalah perpustakaan visi komputer sumber terbuka untuk pemrosesan gambar, pembelajaran mesin, dan deteksi waktu-nyata.[1]

2.3 Background Subtraction

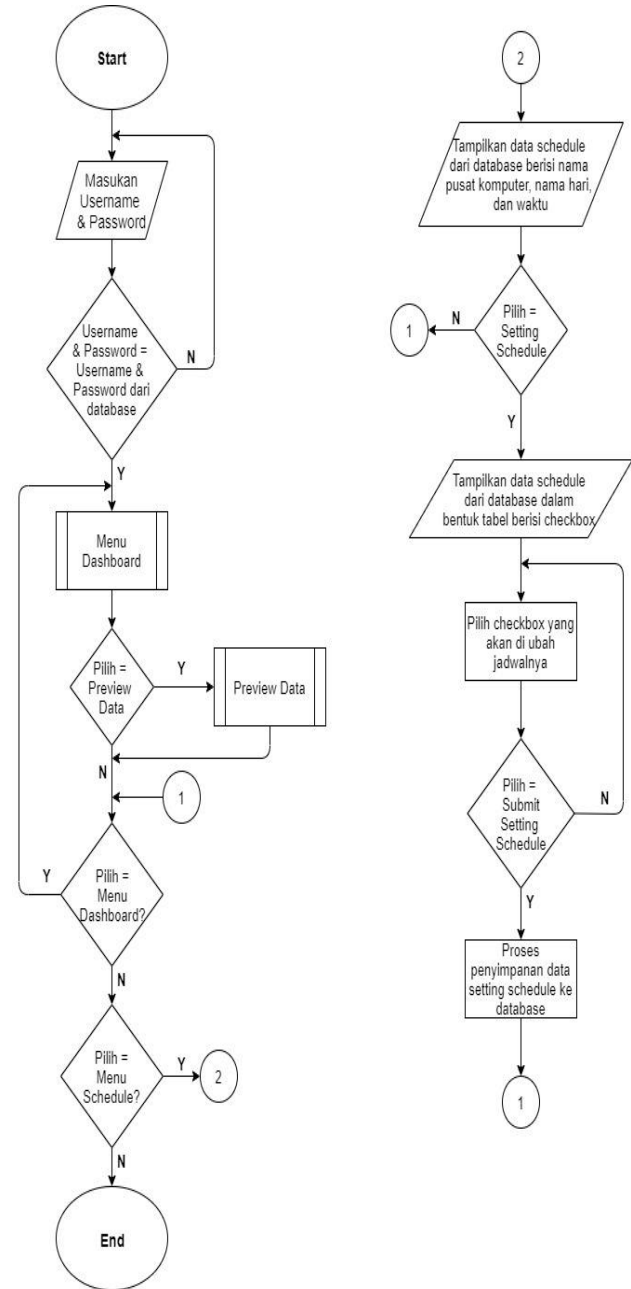
Background Subtraction merupakan salah satu teknik pada bidang pengolahan citra dan *computer vision* yang bertujuan untuk mendeteksi/mengambil *foreground* dari *background* untuk diproses lebih lanjut. Proses deteksi objek bergerak dengan metode *background subtraction* didasarkan pada perbedaan antara *background* referensi dengan *frame*. [6] Adanya beberapa metode pada *background subtraction* yang akan digunakan oleh penelitian ini, yaitu MOG2 (*Mixture of Gaussian 2*) dan GMG (*Godbehere Matsukawa Goldberg*). *Mixture of Gaussian 2* atau disingkat menjadi MOG2 merupakan algoritma yang menggunakan kepadatan probabilitas campuran Gaussian. Sedangkan GMG (*Godbehere Matsukawa Goldberg*) merupakan sebuah algoritma yang hanya berdasarkan studi yang menggabungkan estimasi gambar latar belakang statistik, segmentasi *Bayesian* per *pixel*, dan solusi perkiraan untuk masalah pelacakan *multi-target* menggunakan metode *Kalman-Filter* dan pencocokan *Gale-shapley*. Dalam algoritma GMG, *Bayesian* digunakan untuk menganalisis dan merekonstruksi gambar menggunakan cara *inference*. *Bayesian Inference* merupakan sebuah proses mendeduksi properti atau atribut dari suatu populasi/obyek atau distribusi populasi data menggunakan *Bayesian Theorem*[2] dan *Gale Shapley Pairing* merupakan suatu algoritma yang bertujuan untuk mencari pasangan yang telah memenuhi kriteria tertentu kemudian melakukan pengklasifikasian.[5]

2.4 Confusion Matrix

Confusion Matrix adalah suatu tabel yang digunakan untuk mengukur kinerja masalah klasifikasi pembelajaran mesin.[4] Adanya 4 perhitungan *confusion matrix* ini, yaitu *Accuracy*, *recall*, *precision*, dan *F1 Score*. *Accuracy* merupakan perhitungan jumlah prediksi yang benar dari ukuran *output*. *Precision* merupakan perhitungan rasio prediksi benar positif dibanding dengan keseluruhan hasil yang diprediksi positif. *Recall* merupakan perhitungan rasio prediksi benar positif

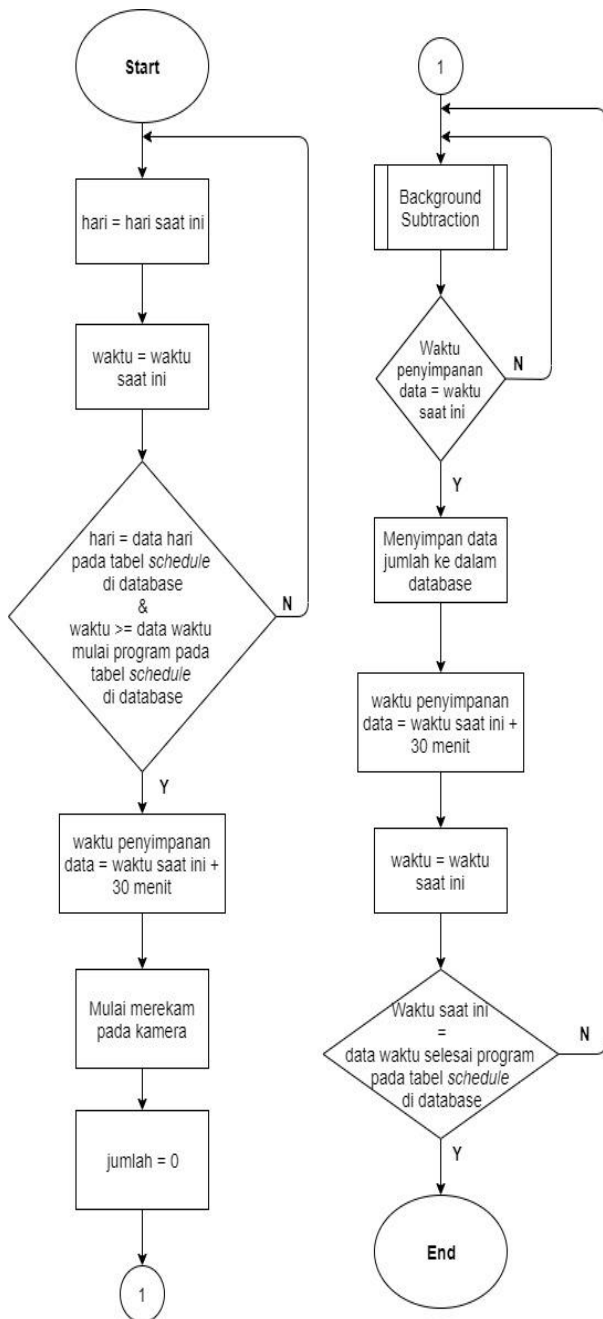
dibanding dengan keseluruhan data yang benar positif. Sedangkan *F1 Score* merupakan perhitungan rasio keseimbangan antara *precision* dan *recall*.

3. DESAIN SISTEM



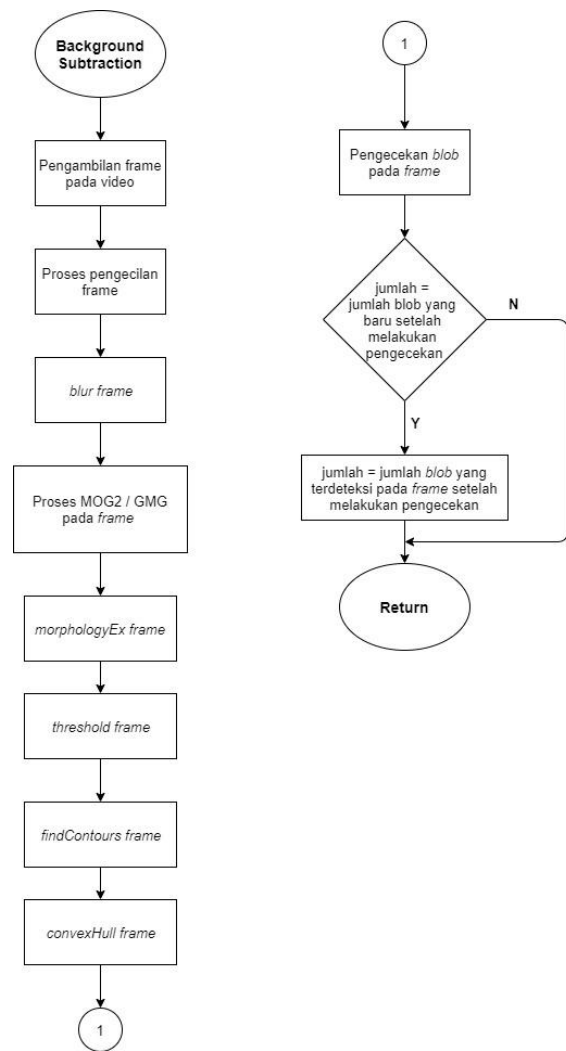
Gambar 1. Flowchart web

Pada Gambar 1 merupakan proses jalannya program *website* dalam bentuk *flowchart*. Proses ini dimulai dengan menampilkan *login* terlebih dahulu dengan memasukkan *username* dan *password*. Jika *login* berhasil, maka akan menampilkan *menu dashboard* terlebih dahulu. Lalu dapat memilih *menu schedule* untuk menampilkan jadwal mulainya program pendeteksian, serta dapat menyetel jadwal dengan memilih *checkbox* yang sudah disediakan. Jika sudah selesai menyetel jadwal, *checkbox* yang tercentang akan disimpan kedalam *database*.



Gambar 2. Flowchart aplikasi penghitung jumlah pengguna

Pada Gambar 2 merupakan proses jalannya program aplikasi penghitung jumlah pengguna dalam bentuk *flowchart*. Proses ini dimulai dengan melakukan pengecekan hari dan waktu saat ini. Jika hari saat ini sama dengan data hari pada tabel *schedule* di *database* dan waktu saat ini lebih dari sama dengan data waktu mulai program pada tabel *schedule* di *database*, maka webcam atau kamera mulai merekam dan menjalankan proses *background subtraction*. Penjelasan lebih detik mengenai proses *background subtraction* dapat dilihat pada Gambar 3.6. Data jumlah pengguna dapat disimpan kedalam *database* jika melewati waktu sampai 30 menit. Program akan selesai ketika waktu saat ini lebih dari sama dengan data waktu selesai program pada tabel *schedule* di *database*.



Gambar 3. Sub flowchart background subtraction

Pada Gambar 3 merupakan proses *background subtraction* dalam melakukan pendeteksian. Proses ini dimulai dengan melakukan pengambilan *frame*. Lalu melakukan pengecilan *frame*, pengecilan *frame*, dan melakukan proses *background subtraction* dengan memanggil metode MOG2 atau GMG pada *frame*. Kemudian dilakukan proses *morphologyEx* yang bertujuan untuk menutupi lubang kecil didalam obyek latar depan atau titik hitam pada obyek, proses *threshold* yang bertujuan untuk menghilangkan *noise*, proses *findContours* yang bertujuan untuk menghilangkan semua titik yang berlebihan dan memampatkan kontur, dan proses *threshold* yang bertujuan untuk menemukan cembung lambung dari titik 2D pada *frame*. Setelah itu melakukan pengecekan *blob* atau latar depan pada *frame* dengan mengecek ukuran *blob*. Jumlah *blob* akan terhitung jika sesuai dengan ukuran *blob* yang ditentukan dan menyimpan jumlah *blob* yang baru setelah melakukan pengecekan *blob*.

4. PENGUJIAN SISTEM

Evaluasi ini dilakukan dengan mengambil 16 frame dari masing-masing 6 video, serta dengan menggunakan 2 resolusi yang berbeda, yaitu resolusi original berukuran 1280x720 dan

640x360 setelah melakukan pengecilan *frame*. *Frame rate* yang didapat pada proses pendeteksian sekitar 8 *frame* per detik. 6 video tersebut merupakan 3 video dengan peletakan yang berbeda dan 3 video dengan pencahayaan ruangan yang berbeda.

Tabel 1. Hasil F1 Score Confusion Matrix Menggunakan Video Berisi Rekaman Mematikan Lampu Ruangan Bagian Depan

		1280x720		640x360	
		MOG2	GMG	MOG2	GMG
Frame	1	80%	57%	100%	100%
	2	80%	57%	100%	80%
	3	100%	57%	100%	80%
	4	100%	57%	100%	80%
	5	100%	67%	100%	80%
	6	100%	80%	100%	80%
	7	100%	80%	67%	50%
	8	100%	80%	67%	100%
	9	100%	100%	67%	100%
	10	80%	100%	67%	67%
	11	80%	100%	67%	67%
	12	80%	67%	67%	67%
	13	80%	67%	67%	100%
	14	80%	67%	67%	100%
	15	80%	67%	100%	100%
	16	80%	67%	100%	100%
Average		89%	73%	83%	84%

Hasil F1 Score pada Tabel 1 menunjukkan nilai rata-rata dengan menggunakan resolusi 1280x720 pada metode MOG2 mencapai 89% dan metode GMG mencapai 73%. Setelah melakukan pengecilan *frame* menjadi 640x360, nilai rata-rata metode MOG2 mencapai 83% dan metode GMG mencapai 84%.

Tabel 2. Hasil F1 Score Confusion Matrix Menggunakan Video Berisi Rekaman Mematikan Lampu Ruangan Bagian Belakang

		1280x720		640x360	
		MOG2	GMG	MOG2	GMG
Frame	1	44%	67%	67%	80%
	2	44%	67%	67%	80%
	3	44%	80%	67%	80%
	4	50%	80%	67%	80%
	5	44%	80%	67%	80%
	6	44%	80%	67%	80%
	7	44%	80%	67%	80%
	8	50%	80%	67%	100%
	9	50%	80%	67%	100%
	10	50%	100%	67%	100%
	11	50%	100%	67%	100%
	12	50%	100%	67%	80%
	13	50%	80%	67%	80%
	14	50%	80%	67%	80%
	15	50%	80%	67%	80%
	16	57%	80%	67%	80%
Average		48%	82%	67%	85%

Hasil F1 Score pada Tabel 2 menunjukkan nilai rata-rata dengan menggunakan resolusi 1280x720 pada metode MOG2 mencapai 48% dan metode GMG mencapai 82%. Setelah melakukan pengecilan *frame* menjadi 640x360, nilai rata-rata metode MOG2 mencapai 67% dan metode GMG mencapai 85%.

Tabel 3. Hasil F1 Score Confusion Matrix Menggunakan Video Berisi Rekaman Mematikan Semua Lampu Ruangan

		1280x720		640x360	
		MOG2	GMG	MOG2	GMG
Frame	1	57%	50%	80%	67%
	2	57%	40%	80%	67%
	3	67%	40%	80%	67%
	4	80%	40%	80%	67%
	5	50%	40%	80%	67%
	6	80%	40%	80%	67%
	7	80%	33%	80%	67%
	8	80%	40%	80%	67%
	9	67%	40%	80%	67%
	10	67%	40%	80%	67%
	11	67%	40%	80%	67%
	12	67%	40%	100%	67%
	13	67%	40%	100%	67%
	14	67%	40%	100%	67%
	15	67%	40%	100%	67%
	16	67%	40%	100%	67%
Average		68%	40%	86%	67%

Hasil F1 Score pada Tabel 3 menunjukkan nilai rata-rata dengan menggunakan resolusi 1280x720 pada metode MOG2 mencapai 68% dan metode GMG mencapai 40%. Setelah melakukan pengecilan *frame* menjadi 640x360, nilai rata-rata metode MOG2 mencapai 86% dan metode GMG mencapai 67%.

Tabel 4. Hasil F1 Score Confusion Matrix Menggunakan Video Peletakan Webcam Pada Bagian Belakang Di Sudut Kanan

		1280x720		640x360	
		MOG2	GMG	MOG2	GMG
Frame	1	40%	50%	50%	50%
	2	40%	50%	50%	50%
	3	40%	50%	50%	50%
	4	40%	50%	50%	50%
	5	40%	50%	50%	50%
	6	40%	50%	50%	50%
	7	40%	50%	50%	50%
	8	40%	40%	50%	50%
	9	40%	40%	50%	50%
	10	40%	33%	50%	50%
	11	40%	33%	50%	50%
	12	40%	33%	50%	50%
	13	40%	33%	50%	50%
	14	40%	33%	50%	50%
	15	40%	33%	50%	50%
	16	40%	33%	50%	50%
Average		40%	41%	50%	50%

Hasil F1 Score pada Tabel 4 menunjukkan nilai rata-rata dengan menggunakan resolusi 1280x720 pada metode MOG2 mencapai

40% dan metode GMG mencapai 41%. Setelah melakukan pengecilan *frame* menjadi 640x360, nilai rata-rata metode MOG2 mencapai 50% dan metode GMG mencapai 50%.

Tabel 5. Hasil F1 Score Confusion Matrix Menggunakan Video Peletakan Webcam Pada Bagian Depan Di Sudut Kiri

		1280x720		640x360	
		MOG2	GMG	MOG2	GMG
Frame	1	67%	57%	67%	50%
	2	67%	57%	67%	50%
	3	67%	67%	67%	50%
	4	67%	67%	67%	67%
	5	67%	67%	67%	67%
	6	67%	67%	67%	67%
	7	67%	67%	67%	67%
	8	67%	40%	67%	67%
	9	67%	40%	67%	67%
	10	67%	40%	67%	67%
	11	57%	67%	57%	67%
	12	57%	57%	57%	67%
	13	57%	50%	57%	67%
	14	57%	50%	57%	67%
	15	57%	50%	57%	67%
	16	57%	44%	57%	67%
Average		63%	55%	63%	64%

Hasil F1 Score pada Tabel 5 menunjukkan nilai rata-rata dengan menggunakan resolusi 1280x720 pada metode MOG2 mencapai 63% dan metode GMG mencapai 55%. Setelah melakukan pengecilan *frame* menjadi 640x360, nilai rata-rata metode MOG2 mencapai 63% dan metode GMG mencapai 64%.

Tabel 6. Hasil F1 Score Confusion Matrix Menggunakan Video Peletakan Webcam Pada Bagian Depan Di Sudut Kanan

		1280x720		640x360	
		MOG2	GMG	MOG2	GMG
Frame	1	100%	50%	100%	100%
	2	100%	50%	100%	100%
	3	100%	100%	100%	100%
	4	100%	100%	100%	100%
	5	100%	100%	100%	100%
	6	100%	100%	100%	100%
	7	100%	100%	100%	100%
	8	100%	50%	100%	100%
	9	100%	50%	100%	100%
	10	100%	50%	100%	100%
	11	100%	50%	100%	100%
	12	67%	50%	100%	100%
	13	67%	100%	100%	100%
	14	67%	100%	100%	100%
	15	67%	100%	100%	100%
	16	67%	100%	100%	100%
Average		90%	78%	100%	100%

Hasil F1 Score pada Tabel 6 menunjukkan nilai rata-rata dengan menggunakan resolusi 1280x720 pada metode MOG2 mencapai 90% dan metode GMG mencapai 78%. Setelah melakukan pengecilan *frame* menjadi 640x360, nilai rata-rata metode MOG2 mencapai 100% dan metode GMG mencapai 100%.

5. KESIMPULAN

Beberapa kesimpulan yang dapat diambil antara lain:

- Agar dapat memonitor seluruh tempat duduk yang tersedia pada ruangan tersebut, posisi *webcam* ditaruh pada tempat yang tinggi dan berada di bagian sudut ruangan, serta merubah posisi tempat duduk sesuai dengan radius *webcam* hingga dapat merekam semua tempat duduk yang tersedia.
- Dalam melakukan proses pendeteksian dianjurkan untuk *me-resize* atau diperkecil *frame* mulai dari resolusi 1280x720 pixel di-*resize* ke resolusi 640x360 pixel dan metode *background subtraction* dengan proses tercepat yaitu metode MOG2.
- Berdasarkan nilai rata-rata dari hasil *presicion*, *recall*, dan F1 *score*, faktor pencahayaan dan resolusi *frame* sangat berpengaruh terhadap keakurasian pendeteksi pengguna. Pengaruhnya pencahayaan pendeteksian dilihat dari hasil *confusion matrix* yang menunjukkan bahwa yang terdeteksi kebanyakan melebihi dari jumlah pengguna yang berada ruangan. Sedangkan pengaruhnya resolusi terhadap pendeteksian dilihat dari hasil F1 *score* pada salah satu video pencahayaan yang menunjukkan dengan menggunakan resolusi 1280x720 hasil metode MOG2 mencapai 68% dan metode GMG mencapai 40%. Setelah melakukan pengecilan *frame* ke 640x360, nilai rata-rata dari MOG2 mencapai 86% dan metode GMG mencapai 67%.
- Berdasarkan nilai rata-rata dari hasil *presicion*, *recall*, dan F1 *score*, metode MOG2 lebih baik daripada GMG.

6. DAFTAR PUSTAKA

- [1] Ajisafe, D. 2018. *Real-Time Object detection API using TensorFlow and OpenCV*. URI= <https://towardsdatascience.com/real-time-object-detection-api-using-tensorflow-and-opencv-47b505d745c4>
- [2] Brooks, J. 2018. *Probability concepts explained: Bayesian inference for parameter estimation*. URI= <https://towardsdatascience.com/probability-concepts-explained-bayesian-inference-for-parameter-estimation-90e8930e5348>
- [3] Hermawan. 2019. *Pengertian Webcam Beserta Fungsi dan Cara Kerja Webcam*. URI= <https://www.nesabamedia.com/pengertian-webcam-dan-fungsi-webcam/>
- [4] Narkhede, S. 2018. *Understanding Confusion Matrix*. URI= <https://towardsdatascience.com/understanding-confusion-matrix-a9ad42dcfd62>
- [5] Osipenko, A. 2016. *Gale-Shapley algorithm simply explained*. URI= <https://towardsdatascience.com/gale-shapley-algorithm-simply-explained-caa344e643c2>
- [6] Pamungkas, A. 2015. *Background Subtraction (Foreground Detection)*. URI= <https://pemrogramanmatlab.com/2015/10/12/background-subtraction-foreground-detection>

