

METODE BACKGROUND SUBTRACTION UNTUK MONITORING OBYEK BERGERAK MELALUI KAMERA WEBCAM

Gladys Julia Nuary Putri

Program Studi Teknik Informatika S1, Fakultas Teknologi Industri
Institut Teknologi Nasional Malang, Jalan Raya Karanglo km 2 Malang, Indonesia
gladisjulianuaryputri@gmail.com

ABSTRAK

Deteksi objek manusia merupakan salah satu arah penelitian yang penting untuk meningkatkan kemampuan sistem pengawasan. Penelitian ini bertujuan untuk merancang suatu program pengolahan citra yang mampu mendeteksi objek yang terekam di dalam video menggunakan video yang direkam. Penelitian ini menggunakan Metode background subtraction. Metode background subtraction dapat mendeteksi subtraksi pada background dengan mengubah citra menjadi citra biner dan menentukan tingkat kepekaan perubahan pixel background.

Di bidang ini banyak citra yang dapat diolah antara lain mencakup pada sebuah gambar dan video juga. Salah satu hal pengolahan citra pada video yang berkembang dengan pesat saat ini adalah sistem pendeteksian objek. Deteksi objek merupakan sebuah proses pendeteksian objek yang bergerak atau berpindah dalam suatu video yang telah dirubah kedalam bentuk image sequence yaitu potongan gambar yang terurut dari sebuah video.

Tujuan dari penelitian ini untuk mendeteksi objek yang bergerak yang terurut dari sebuah video dengan memanfaatkan teknologi pengolahan citra menggunakan MATLAB R2018b. Diharapkan dapat mempermudah proses deteksi objek yang akan dideteksi.

Kata kunci : *Monitoring, Deteksi Obyek Manusia, Background Subtraction, Threshold, Opening, Closing.*

1. PENDAHULUAN

1.1 Latar Belakang

Sistem monitoring adalah sistem yang digunakan untuk melakukan fungsi pengawasan secara berkala untuk mendapatkan informasi yang diinginkan. Hasil dari sistem monitoring tersebut berperan penting pada peningkatan berbagai aspek, yaitu informasi, keamanan, tingkat produktivitas, dan kinerja [3].

Salah satu bentuk sistem monitoring adalah aplikasi deteksi obyek. Aplikasi deteksi obyek merupakan pendeteksi obyek yang dapat mendefinisikan atau mengekstrak informasi dari obyek pada suatu citra. Salah satu contoh aplikasi deteksi obyek adalah deteksi obyek bergerak.

Deteksi obyek bergerak merupakan salah satu arah penelitian yang penting dalam rangka meningkatkan kemampuan sistem pengawasan di tempat-tempat umum. Secara umum, cara kerja deteksi obyek manusia ini mengadaptasi cara pengamatan apa yang ada pada dunia nyata melalui indra penglihatan manusia, dimulai dari suatu pembelajaran menganalisa gambar dan video untuk memperoleh hasil sebagaimana yang bisa dilakukan manusia. Dengan kata lain, pendeteksi obyek manusia mencoba meniru cara kerja sistem visual manusia (human vision) yang saat ini dikenal dengan teknologi computer vision [2].

Beberapa penelitian telah dilakukan dalam rangka penerapan teknik atau metode untuk mendeteksi obyek manusia. Salah satunya adalah

Metode Background Subtraction untuk Deteksi Obyek Manusia pada Lingkungan Statis [3]. Pada penelitian tersebut peneliti melakukan penerapan metode background subtraction untuk mendeteksi obyek manusia pada file video. Namun pada penelitian tersebut frame yang diekstrak dari video tersebut sangat terbatas yaitu hanya 5 frame yang diekstrak dalam 1 detik yang menghasilkan pendeteksian obyek yang kurang menyeluruh terhadap keseluruhan frame yang ada pada file video dan pendeteksian obyek manusia tidak dilakukan secara kontinyu sehingga penelitian tersebut relatif belum akurat dan belum bisa dijadikan dasar yang kuat untuk mendeteksi obyek manusia mengingat dalam satu detik file video berisi lebih besar dari 15 frame karena mata manusia melihat gambar sebagai suatu gerakan kontinyu apabila gambar-gambar tersebut kecepatannya melebihi 15 frame/detik [4]. Selain itu, penelitian yang dilakukan Solichin hanya menggunakan operasi dilasi dan region filling sebagai acuan untuk mendeteksi obyek manusia. Menurut penulis operasi yang digunakan tersebut masih belum optimal dan masih bisa disempurnakan dengan penambahan beberapa operasi lain.

Oleh karena itu penulis akan membuat aplikasi dengan menerapkan metode *Background Subtraction* untuk Deteksi Obyek bergerak pada Lingkungan Statis. Pada penelitian tersebut peneliti melakukan penerapan metode background subtraction untuk mendeteksi obyek manusia pada file video. Tujuan penelitian ini dilakukan adalah mengukur dan

menganalisa bagaimana penerapan metode background subtraction terhadap pendeteksian obyek manusia hingga pada hasil penelitian didapatkan nilai keberhasilan berdasarkan pengukuran dalam pendeteksian obyek manusia pada sebuah kamera video.

1.2 Rumusan Masalah

Berdasarkan latar belakang di atas dapat di rumuskan beberapa masalah, antara lain :

1. Bagaimana cara menerapkan metode *Background Subtraction* pada Matlab R2016b?
2. Bagaimana proses implementasi metode *Background Subtraction* untuk deteksi gerak pada Matlab R2016b?

1.3 Batasan Masalah

Penulis membuat beberapa batasan masalah, antara lain :

1. Kamera *Webcam* diletakkan pada lingkungan statis.
2. Menggunakan Bahasa pemrograman Matlab R2016b.
3. Metode yang digunakan untuk mendeteksi obyek adalah *Background Subtraction*.
4. Data penelitian yang digunakan merupakan data video kondisi lingkungan di dalam Gedung maupun luar Gedung.
5. Sistem pada penelitian ini belum dapat membedakan pergerakan manusia, hewan, atau benda.

1.4 Tujuan

Dari masalah-masalah yang telah dirumuskan, maka dapat diketahui dari laporan skripsi ini, yaitu :

1. Untuk menerapkan metode *Background Subtraction* pada Matlab R2016b.
2. Untuk menerapkan sistem deteksi gerakan dengan metode *Background Subtraction*.

1.5 Manfaat

Manfaat dari penyusunan skripsi ini adalah sebagai berikut:

1. Membantu user untuk mendokumentasikan data yang tertangkap oleh *Camera*.
2. Memberikan informasi kepada user jika terjadi pergerakan.

2. TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Deteksi gerak merupakan salah satu arah penelitian yang penting untuk meningkatkan kemampuan sistem pengawasan. Beberapa penelitian telah dilakukan dalam rangka penerapan teknik atau metode untuk mendeteksi obyek manusia, salah satunya adalah metode background subtraction untuk deteksi obyek manusia pada lingkungan statis. Permasalahan yang terjadi menerapkan metode

background subtraction dan operasi morfologi untuk mendeteksi obyek manusia pada sebuah database video. [3].

Operasi morfologi digunakan untuk menghilangkan noise yang dapat mengganggu keberhasilan deteksi pada video, sedangkan background subtraction akan mengubah citra biner dan menentukan tingkat perubahan background. Hasil penelitian Program dapat melakukan deteksi dan perhitungan secara otomatis, pendeteksian objek pada video dapat dihasilkan dalam kondisi exposure value statis. terlihat baik ketika pengujian dilakukan pada kondisi pencahayaan yang kurang (Harry, 2017).

2.2 Motion Detection

Deteksi gerak adalah proses mendeteksi perubahan posisi dari suatu objek relatif terhadap sekitarnya atau perubahan lingkungan *relative* terhadap suatu objek [1]. *Motion detection* melakukan pendekatan dengan membandingkan frame pada saat ini dengan frame sebelumnya. Mula-mula kamera video akan menangkap gambar dari ruangan yang sedang dipantau. Kemudian membandingkan warna yang terdapat pada frame saat ini dengan frame sebelumnya. Apabila terdapat perbedaan warna, maka objek tersebut terdeteksi sebagai gerakan [1].

Menurut penelitian yang dilakukan [1], ada tiga metode yang umum digunakan untuk mendeteksi gerak. Metode tersebut adalah *background subtraction*, *optical flow* dan *temporal differences*. *Background subtraction* dilakukan dengan membandingkan citra tertentu dengan citra yang dijadikan sebagai referensi. *Optical flow* merupakan sebuah metode segmentasi gerak yang menggunakan karakteristik pergerakan untuk mendeteksi gerak secara independen yang melewati sebuah kamera

Metode *temporal differences* juga dikenal dengan nama *frame differences*. Metode ini dilakukan dengan membandingkan frame - frame citra yang ditangkap sesuai dengan urutan waktu [4].

2.3 Tahap Preprocessing

Preprocessing merupakan tahap persiapan untuk proses awal. Input yang berupa video diproses dari RGB ke grayscale. Pada tahap ini merubah inputan yang berupa image frame yang masih RGB menjadi grayscale agar lebih mudah untuk diproses. Grayscale adalah warna-warna piksel yang berada dalam rentang gradasi warna hitam dan putih. Grayscale didapat dari konversi RGB image dengan Citra grayscale, yaitu citra yang nilai pixel-nya merepresentasikan derajat keabuan atau intensitas warna putih.

Nilai intensitas paling rendah merepresentasikan warna hitam dan nilai intensitas paling tinggi merepresentasikan warna putih. Pada umumnya citra grayscale memiliki kedalaman pixel 8 bit (256

derajat keabuan), tetapi ada juga citra grayscale yang kedalaman pixel-nya bukan 8 bit, misalnya 16 bit untuk penggunaan yang memerlukan ketelitian tinggi.

Citra grayscale merupakan citra satu kanal, dimana citra $f(x,y)$ merupakan fungsi tingkat keabuan dari hitam ke putih, x menyatakan variable kolom atau posisi pixel di garis jelajah dan y menyatakan variable kolom atau posisi pixel di garis jelajah. Intensitas f dari gambar hitam putih pada titik (x,y) disebut derajat keabuan (grey level), yang dalam hal ini derajat keabuannya bergerak dari hitam keputih. Derajat keabuan memiliki rentang nilai dari I_{min} sampai I_{max} , atau $I_{min} < f < I_{max}$, selang (I_{min}, I_{max}) disebut skala keabuan.

Biasanya selang (I_{min}, I_{max}) sering digeser untuk alasan-alasan praktis menjadi selang $[0,L]$, yang dalam hal ini nilai intensitas 0 menyatakan hitam, nilai intensitas L menyatakan putih, sedangkan nilai intensitas antara 0 sampai L bergeser dari hitam ke putih. Sebagai contoh citra grayscale dengan 256 level artinya mempunyai skala abu dari 0 sampai 255 atau $[0,255]$, yang dalam hal ini intensitas 0 menyatakan hitam, intensitas 255 menyatakan putih, dan nilai antara 0 sampai 255 menyatakan warna keabuan yang terletak antara hitam dan putih.

Proses konversi dari citra berwarna menjadi Grayscale menggunakan persamaan sebagai berikut :

$$Y = 0,2 * R + 0,5 * G + 0,2 * B$$

Dimana :

Y = derajat keabuan (uint8 atau uint16).

R = nilai pixel chanel Red

G = nilai pixel chanel Green

B = nilai pixel chanel Blue

2.4 Metode Background Substraction

Background subtraction adalah proses untuk mendeteksi pergerakan atau perbedaan signifikan yang terjadi didalam frame video ketika dibandingkan dengan citra referensi. Tujuan dari background subtraction adalah untuk memisahkan objek dan background sehingga gerakan dari sebuah objek terdeteksi.

Background subtraction banyak digunakan pada proyek-proyek berbasis pengolahan citra. Salah satu aplikasi yang sering menggunakan fungsi dari background subtraction ini adalah aplikasi penghitung jumlah pengunjung yang memasuki maupun meninggalkan ruangan ataupun kendaraan yang melewati suatu jalur dalam sistem informasi lalu lintas. Metode ini memisahkan manusia atau kendaraan dengan cara perbedaan latar belakang (*background*) dan manusia atau kendaraan (*foreground*) yang bergerak. Jika kondisi yang akan diamati oleh background subtraction hanya berupa latar belakang dan objek bergerak yang akan diamati, hal tersebut sangat mudah.

Namun, ketika terdapat objek lain yang juga berpindah dari titik satu ke titik yang lain misalnya bayangan dari objek tersebut hal ini akan diproses juga sebagai *foreground* atau masuk ke dalam objek yang diklasifikasikan. Ide dasar dari Background subtraction adalah $|frame(n) - background| > threshold$, bila piksel ke n memenuhi persamaan tersebut, maka piksel tersebut digolongkan kedalam kelompok piksel objek dan selain itu adalah latar.

$$S(n) = |F(n) - B| > T$$

Dimana:

$F(n)$ merupakan frame ke- n dari video masukan,

B merupakan citra background

$S(n)$ adalah citra selisih ken.

T adalah Threshold.

2.5 Tahap Thresholding

Threshold adalah metode di mana kita menetapkan suatu nilai, kemudian kita hanya mengambil nilai di atasnya saja atau di bawahnya saja, sedangkan nilai selainnya diabaikan (Bradski dan Kaehler, 2008). Hasil dari threshold adalah citra yang tampak nyata perbedaan intensitasnya, biasanya hitam dan putih. Metode ini biasanya digunakan untuk segmentasi atau pemisahan suatu objek dengan lainnya.

Untuk melakukan penghitungan pixel putih pada masing-masing region, maka data citra dikonversikan ke dalam citra biner dengan memanfaatkan thresholding. Thresholding adalah proses mengubah suatu citra berwarna atau berderajat keabuan (Grayscale) menjadi citra biner atau hitam putih, sehingga dapat diketahui daerah mana yang termasuk objek dan background dari citra secara jelas [3,6]. Citra hasil thresholding biasanya digunakan lebih lanjut untuk proses pengenalan obyek serta ekstraksi fitur. Tipe data dari hasil proses thresholding adalah tipe data float, yaitu antara 0 sampai dengan 1.

Dengan parameter yang di set sebelumnya makadada citra yang jika melebihi batas yang ditentukan akan dibuat menjadi 1 atau putih dan jika dibawah batas yang ditentukan maka akan dibuat menjadi 0 atau hitam. Berdasarkan nilai ambang (threshold) T dapat ditunjukkan seperti pada persamaan 3.

$$f(x,y)' = \begin{cases} \alpha_1, & f(x,y) < T \\ \alpha_2, & f(x,y) \geq T \end{cases}$$

Jika $\alpha_1 = 0$ dan $\alpha_2 = 1$, maka operasi pengambangan mentransformasikan citra hitam-putih ke citra biner. Dengan kata lain, nilai intensitas pixel semula dipetakan ke dua nilai saja: hitam dan putih.

3. ANALISIS DAN PERANCANGAN

3.1 Analisis Sistem

Pada saat ini keamanan adalah salah satu hal yang sangat penting. Dengan kemajuan teknologi dalam bidang keamanan sekarang ini, pengamatan suatu objek menjadi lebih praktis. Salah satu sistem keamanan pemantauan objek yang tersedia saat ini adalah sistem keamanan kamera pengawas atau lebih dikenal dengan sebutan *Closed Circuit Television* (CCTV). CCTV adalah sebuah kamera video digital yang difungsikan mengontrol semua kegiatan secara visual (audio visual) pada area tertentu yang secara langsung dapat mengawasi, mengamati serta merekam kejadian di suatu ruangan atau area tertentu dan mengirimkan sinyal video pada ruang tersebut yang kemudian akan diteruskan ke sebuah layar monitor.

Permasalahan yang timbul selama ini kamera hanya dapat menangkap suatu objek tetapi tidak dapat memberikan informasi tentang pergerakan objek tersebut. Begitu juga pada kebanyakan sistem keamanan CCTV. Hal ini sangat berguna jika diaplikasikan ke dalam sistem pendeteksi gerakan dengan *motion detection*. Ketika ada objek yang bergerak, maka secara otomatis aplikasi akan memberikan informasi objek asing yang terdeteksi tersebut. Ini akan memberikan kemudahan bagi pengguna untuk menganalisis lebih lanjut video dari rekaman kamera keamanan.

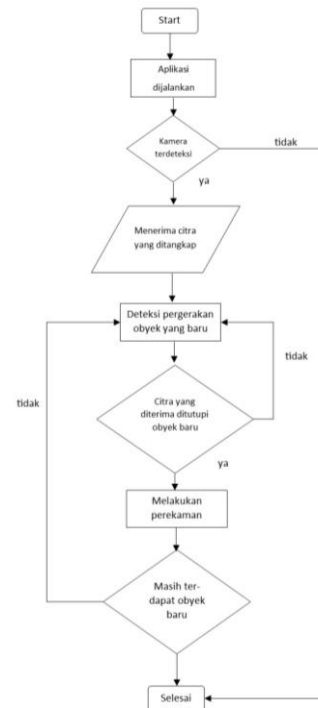
3.2 Sistem yang Akan di Bangun

Sistem yang dibangun pada penelitian ini merupakan perangkat lunak pendeteksi gerakan berbasis *desktop* yang dapat dioperasikan di sistem operasi *Windows*. Aplikasi ini dapat mendapatkan gambar yang dideteksi sebagai pergerakan secara *real time* dengan bantuan kamera yang tersambung dengan *server*.

Kemudian gambar yang telah dideteksi akan berikan notifikasi melalui aplikasi tersebut. Sistem akan memberikan Informasi berupa notifikasi hasil deteksi gerakan yang berupa gambar dari pergerakan sebuah objek yang ditangkap oleh *server*.

Cara kerja aplikasi yang ditunjukkan pada flowchart diatas ini diawali dengan melakukan pendeteksian ada atau tidaknya kamera yang terhubung dengan aplikasi, jika tidak ditemukan adanya kamera, maka aplikasi ini akan berhenti berjalan. Namun jika ditemukan adanya kamera yang dapat terbaca oleh aplikasi, maka alur kerja aplikasi akan berlanjut ke tahap pengambilan citra awal untuk dijadikan sebagai citra acuan. Setelah itu kamera akan terus menerus melakukan pengambilan citra, di mana pada tahap ini aplikasi mulai melakukan pendeteksian gerak berdasarkan perbandingan antara citra acuan dengan citra terbaru. Bila terdapat perbedaan antara citra acuan dengan citra terbaru dan perbedaan ini melebihi nilai *threshold* yang telah ditentukan, maka perbedaan itu dianggap sebagai pergerakan / objek baru, dengan begitu kamera akan melakukan perekaman citra secara terus-menerus sampai tidak

ditemukannya perbedaan antara citra acuan dengan citra terbaru dan selama kapasitas penyimpanan memori mencukupi. Terjadi pengolahan citra ketika menjalankan deteksi pergerakan objek baru yang ada pada flowchart.



Gambar 2. Flowchart Sistem

Pengolahan citra yang terjadi itu antara lain:

- Mengubah citra berwarna menjadi citra berskala keabuan. Warna dari citra yang direkam oleh kamera sebelum melalui pengolahan ialah RGB, namun untuk dilakukan pendeteksian gerak, maka warna RGB tersebut dikonversi menjadi warna berskala keabuan atau grayscale;
- Menghilangkan noise. Pada citra yang telah mengalami proses konversi kemudian akan dilakukan proses penghilangan noise-noise yang ada untuk memudahkan dalam pendeteksian gerak;
- Pembandingan citra. Pada tahap ini akan dilakukan pembandingan antara citra acuan dengan citra terbaru yang nampak pada kamera. Tahap ini dilakukan dengan cara menghitung selisih antara citra terbaru dengan citra acuan (citra terbaru dikurang dengan citra acuan);
- Thresholding. Pada tahap dilakukan pengecekan nilai dari hasil tahap sebelumnya dengan nilai *threshold*. Jika nilai hasil pada tahap sebelumnya lebih besar dibandingkan dengan nilai *threshold*, maka akan terdeteksi sebagai pergerakan atau objek baru

3.3 Desain Arsitektur

Dalam system yang dikembangkan menggunakan metode *Background Subtraction* yang

akan menghasilkan suatu hasil yang dapat mendeteksi gerakan

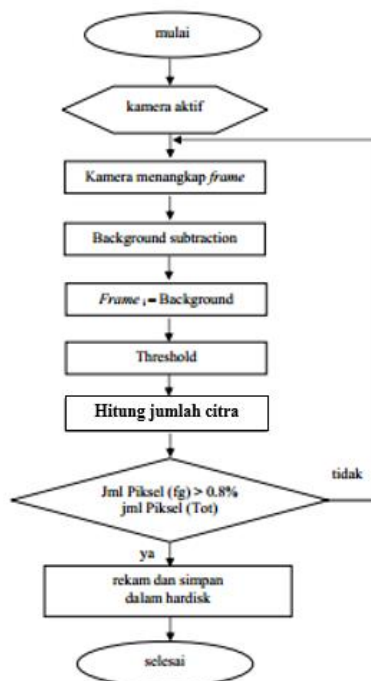


Gambar 3.2 Tampilan Tahapan Blok Diagram

Penjelasan diagram blok sistem keamanan rumah :

1. Proses penangkapan citra dari video atau kamera
2. Menentukan titik ambang batas citra beserta batas pergerakan citra.
3. Proses mengubah citra ke citra digital.
4. Menjadikan citra awal menjadi citra referensinya.
5. Menghitung jumlah nilai piksel RGB dari citra referensi.
6. Mengubah citra baru ke dalam citra digital.
7. Menghitung jumlah nilai piksel RGB dari citra baru.
8. Membandingkan jumlah nilai RGB citra referensi dengan citra baru.
9. Membandingkan nilai perbedaan dari citra referensi dengan citra baru dengan batas pergerakan yang telah ditentukan.
10. Hasil pendeteksian gerakan.

3.4 Flowchart



Gambar 3.4 Tampilan alur background subtraction

Flowchart adalah penggambaran secara grafik langkah-langkah dan urutan – urutan prosedur dari suatu program. Flowchart menolong analis dan

programmer untuk memecahkan masalah kedalam segmen – segmen yang lebih kecil dan menolong dalam menganalisis alternatif – alternatif lain dalam pengoperasian.

Pada Gambar 3.4 ditunjukkan alur dari flowchart sistem .

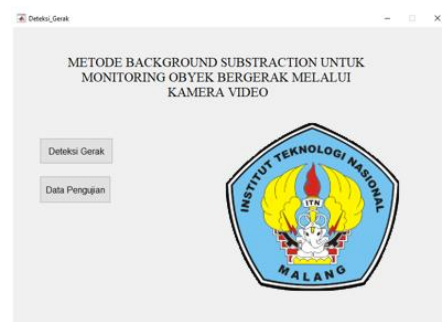
Cara kerja metode *background subtraction* pada flowchart di atas diawali dengan kamera mulai aktif merekam selanjutnya jika terdapat objek yang bergerak maka kamera akan menangkap background pertama selanjutnya akan dilakukan proses background subtraction. Alur selanjutnya background sebelumnya yang telah diambil akan dikurangkan atau dibandingkan dengan background yang baru ditangkap kamera. Maka akan dihitung nilai thresholdnya dan dihitung berdasarkan jumlah citra hasilnya akan disimpan pada ruang penyimpanan maka proses berakhir.

4. HASIL DAN PEMBAHASAN

Implementasi sistem adalah proses penerapan rancangan sistem yang telah dibuat menjadi suatu aplikasi yang bisa dijalankan. Disamping itu, pada pengujian akan dijelaskan hasil uji coba hasil perhitungan metode *background subtraction*. Pada tahapan implementasi sistem, pengembang membuat beberapa form untuk membantu penggunaan sistem ini seperti form menu utama, form sistem deteksi dan form data baru dan data pengujian. Setelah itu hasil uji coba yang dilakukan, akan dianalisa apakah rancangan ini dapat memenuhi tujuan yang akan dicapai seperti yang dipaparkan pada bab 1. Implementasi aplikasi ini dibagi menjadi beberapa bagian diantaranya:

4.1 Tampilan Form Utama

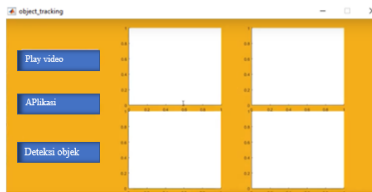
Form menu utama ini sebagai pengenalan dari sistem dan pengembangnya, isi dari form ini terdiri dari menu sistem deteksi dan menu data baru dan data pengujian, seperti pada gambar 4.1.



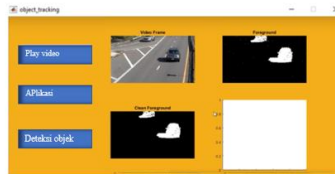
Gambar 4.1 Tampilan Tampilan Menu Utama Sistem.

4.2 Halaman Monitoring dan system Deteksi

Pada form ini, sistem deteksi akan langsung berjalan setelah form ini dibuka, sistem yang berjalan dapat mendeteksi obyek yang bergerak dari video yang telah ditambahkan seperti pada gambar 4.2.



Gambar 4.2 Tampilan Tampilan Form Sistem Deteksi Gerak menggunakan video



Gambar 4.3 Tampilan pengujian Form Sistem Deteksi Gerak

4.3 Pengujian Sistem

Pengujian merupakan tahapan bagian yang penting dalam perancangan perangkat lunak. Pengujian dilakukan untuk menjamin kualitas dan juga mengetahui kelemahan dari perangkat lunak yang sudah di bangun. Tujuan dari pengujian ini adalah untuk menjamin bahwa perangkat lunak yang dibangun memiliki kualitas yang handal, yaitu mampu merepresentasikan kajian pokok dari spesifikasi, analisis, perancangan dan pengkodean dari perangkat lunak itu sendiri.

4.3.1 Pengujian Jarak Sensitivitas Gerakan Terdeteksi

Tabel 4.1 merupakan pengujian jarak dari sistem pendeteksi pergerakan dapat menangkap adanya pergerakan, dimana pengujian ini mengujikan jarak pandang dari camera dan aplikasi mendeteksi adanya sebuah pergerakan..

Tabel 4.1 Pengujian sensitivitas gerak berdasarkan jarak

Jarak (meter)	Sukses	Gagal	Keterangan
1 – 3	[y]		Motion
4 – 6	[y]		Motion
7 – 9	[y]		Motion
10 – 12	[y]		Motion
13 – 15	[y]		Motion
16 – 18		[y]	Tidak Terdeteksi
19 – dst		[y]	Tidak Terdeteksi

Berdasarkan Tabel 4.1, diketahui bahwa jarak m maksimal gerakan yang dapat dideteksi adalah 15 meter

4.3.2 Pengujian berdasarkan nilai threshold pada kamera

Threshold atau ambang batas mempunyai pengaruh pada hasil pengujian dimana threshold

berhubungan dengan kepekaan kamera dalam mendeteksi objek, semakin rendah nilai thresholdnya maka semakin tinggi kepekaan kamera, sehingga jika ada perubahan sedikit saja kamera akan mendeteksi.

Dikarenakan perubahan cahaya sekitar dianggap gerakan maka tidak akan efektif apabila nilai threshold terlalu rendah. Sedangkan semakin tinggi thresholdnya maka semakin rendah kepekaan kamera sehingga kamera sulit untuk mendeteksi gerakan. Oleh karena itu nilai threshold diatur sesuai dengan keadaan sekitarnya sehingga kepekaan kamera menjadi efektif.

Tabel 4.2 Pengujian nilai threshold pada kamera

Ambang batas (Th)	Sukses	Gagal	Status
20		[x]	Tidak Terdeteksi
30		[x]	Tidak Terdeteksi
50	[x]		Deteksi
100		[x]	Deteksi
150		[x]	Deteksi
200			Aman

Berdasarkan Tabel 4.2, diketahui bahwa dengan nilai threshold dapat menghasilkan output yang berbeda terhadap nilai threshold yang berbeda.

4.3.3 Pengujian Fungsionalitas

Berikut merupakan tabel pengujian fungsionalitas sistem pendeteksi pergerakan, pengujian ini berkaitan dengan berjalan atau tidaknya fungsi – fungsi yang telah dibangun.

Tabel 4.4 Pengujian Fungsionalitas

Fungsi	Sukses	Gagal
Menu Utama	[x]	
Connect Camera	[x]	
Autocapture image when any motion	[x]	
Get last image that capture system	[x]	
Notification		[x]

Berdasarkan Tabel 4.4, diketahui bahwa secara fungsionalitas fitur sudah berjalan dengan sukses tetapi notifikasi belum berjalan karena belum dapat mengambil data.

4.4 Pengujian User

Pengujian ini dilakukan dengan maksud untuk mengetahui apakah system yang akan dibangun sudah sesuai dengan rumusan masalah yang diangkat dalam penelitian ini atau sebaliknya. Pengujian user

dilakukan dengan memberikan kuisiner kepada 6 responden. Hasil dari kuisiner dapat dilihat dari table 4.5

Tabel 4.5 Pengujian Studi Kelayakan Aplikasi

Pertanyaan	Sesuai/ mudah	Kurang sesuai/ lumayan	Tidak sesuai/ sulit
Apakah system deteksi ini layak untuk digunakan ?	6		
Apakah system deteksi ini mudah untuk dioperasikan ?	6		
Apakah aplikasi sudah sesuai untuk mendeteksi adanya pergerakan ?	4	2	
Apakah interface aplikasi menarik ?	4	2	

Berdasarkan Tabel 4.5, diketahui bahwa sistem layak diimplementasikan berdasarkan 6 responden yang melakukan uji coba kelayakan aplikasi.

5. Penutup

5.1 Kesimpulan

Beberapa kesimpulan yang dapat diambil dari penerapan sistem terhadap permasalahan yang ada beserta pengujian yang ada dalam perancangan sistem pendeteksi pergerakan adalah sebagai berikut:

1. Dari hasil pengujian fungsionalitas sistem terlihat bahwa semua fungsi aplikasi dapat berjalan seperti yang diharapkan tanpa error. Perancangan sistem pendeteksi pergerakan dapat memberikan solusi terhadap permasalahan yang ada saat ini.
2. Dari hasil pengujian sensitifitas penangkapan gambar berada diantara range hasil presentase 1 hingga 20 berdampak pada seberapa lama aplikasi mendeteksi adanya pergerakan pada video rekaman.
3. Dari hasil pengujian jarak pengambilan video camera terhadap sensitifitas aplikasi mendeteksi

gerak yang terlihat bahwa sistem mampu mendeteksi dengan jarak maksimal 15 meter.

5.2 Saran

Aplikasi sistem pendeksi pergerakan ini pun masih memiliki beberapa keterbatasan, sehingga disarankan untuk pengembangan aplikasi sistem pendeteksi pergerakan selanjutnya agar :

1. Tidak hanya menangkap gambar yang telah terdeteksi oleh sistem, akan tetapi dapat juga melakukan perekaman video secara live.
2. Dikembangkan pada perangkat *mobile* atau *android* agar lebih mudah digunakan oleh pengguna.
3. Proses deteksi gerakan dapat berjalan sesuai yang diharapkan di keadaan cahaya yang minim.
4. Di dalam sistem pendeteksi pergerakan ini dapat membedakan manusia atau benda dengan lebih spesifik

DAFTAR PUSTAKA

- [1] Darmawan, H., Setiawan, H. and Sirajuddin, S., 2013. Pemilihan Pemasok Bahan Baku Produksi Menggunakan Metode Data Envelopment Analysis. *Jurnal Teknik Industri Untirta*, 1(2).
- [2] Jati, H., 2015. Penilaian Efisiensi Universitas Lptk Di Indonesia Dengan Menggunakan Data Envelopment Analysis. *Elinvo (Electronics, Informatics, and Vocational Education)*, 1(1), pp.37-43.
- [3] Avkiran, N.K., 2006. Developing foreign bank efficiency models for DEA grounded in finance theory. *Socio-Economic Planning Sciences*, 40(4), pp.275-296.
- [4] Miski, A., Nandiroh, S. and Pratiwi, I., 2009. Analisis Efisiensi Distribusi Pemasaran Dengan Pendekatan Data Envelopment Analysis (DEA).
- [5] Rambe, I.H. and Syahputra, M.R., 2017. Aplikasi Data Envelopment Analysis (DEA) Untuk Pengukuran Efisiensi Aktivitas Produksi. *MES (Journal of Mathematics Education and Science)*, 2(2).
- [6] Irawan, J.D. and Adriantantri, E., 2019. Pendeteksi Mengantuk Menggunakan Library Python. *Jurnal Mnemonic*, 2(1), pp.22-27.