

# **TEPI JAMAN : *Sistem Deteksi Jatuh pada Manusia Secara Real Time Menggunakan Video Processing***

Untuk memenuhi salah satu persyaratan  
mendapatkan gelar Sarjana Teknik



Penyusun:

Muhammad Irfan Nugraha (17524014)

Muhammad Yusuf Kurniawan (17524064)

**Program Studi Teknik Elektro**

**Fakultas Teknologi Industri**

**Universitas Islam Indonesia**

**Yogyakarta**

**2021**

# HALAMAN PENGESAHAN

## TEPI JAMAN: *Sistem Deteksi Jatuh pada Manusia Secara Real Time Menggunakan Video Processing*

Penyusun:


Muhammad Irfan Nugraha (17524014)

Muhammad Yusuf Kurniawan (17524064)

Yogyakarta, 24 April 2021

Dosen Pembimbing 1

Dosen Pembimbing 2



Elvira Sukma Wahyuni, S.Pd., M.Eng.

Yusuf Aziz Amrulloh, Ph.D.

155231301

045240101

**Program Studi Teknik Elektro  
Fakultas Teknologi Industri  
Universitas Islam Indonesia  
Yogyakarta**

**2021**

## LEMBAR PENGESAHAN TUGAS AKHIR

### TEPI JAMAN: *Sistem Deteksi Jatuh pada Manusia Secara Real Time Menggunakan Video Processing*

**Disusun oleh:**

Muhammad Irfan Nugraha (17524014)

Muhammad Yusuf Kurniawan (17524064)

Telah dipertahankan di depan dewan penguji

Pada tanggal: 6 Mei 2021

Susunan dewan penguji

Ketua Penguji : Elvira Sukma Wahyuni, S.Pd., M.Eng., 

Anggota Penguji 1 : Sisdarmanto Adinandra, ST, M.Sc, Ph.D., 

Anggota Penguji 2 : Dthomas Hatta Fudholi, ST., M.Eng., Ph.D., 

Tugas Akhir ini telah disahkan sebagai salah satu persyaratan  
untuk memperoleh gelar Sarjana Teknik

Tanggal: 22 Mei 2021

Ketua Program Studi Teknik Elektro



Yusuf Aziz Amrulloh, Ph.D.

045240101

## PERNYATAAN

Dengan ini Kami menyatakan bahwa:

1. Tugas Akhir ini tidak mengandung karya yang diajukan untuk memperoleh gelar kesarjanaan di suatu Perguruan Tinggi lainnya, dan sepanjang pengetahuan Kami juga tidak mengandung karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka.
2. Informasi dan materi Tugas Akhir yang terkait hak milik, hak intelektual, dan paten merupakan milik bersama antara tiga pihak yaitu penulis, dosen pembimbing, dan Universitas Islam Indonesia. Dalam hal penggunaan informasi dan materi Tugas Akhir terkait paten maka akan diskusikan lebih lanjut untuk mendapatkan persetujuan dari ketiga pihak tersebut diatas.

Yogyakarta, 22 Mei 2021

Muhammad Irfan Nugraha (17524014) m



Muhammad Yusuf Kurniawan (17524064) m



# DAFTAR ISI

HALAMAN PENGESAHAN	2
LEMBAR PENGESAHAN TUGAS AKHIR	3
PERNYATAAN	4
DAFTAR ISI	5
RINGKASAN TUGAS AKHIR	6
BAB 1 : Definisi Permasalahan	7
BAB 2 : Observasi	8
BAB 3 : Usulan Perancangan Sistem	11
3.1 Usulan Rancangan Sistem	11
3.2 Metode Uji Coba dan Pengujian Usulan Rancangan Sistem	13
3.2.1 Alur Penelitian	13
3.2.2 Pengujian Usulan Rancangan Sistem	15
BAB 4 : Hasil Perancangan Sistem	16
4.1 Kesesuaian Usulan dan Hasil Perancangan Sistem	16
4.2 Kesesuaian Perencanaan dalam Manajemen Tim dan Realisasinya	16
4.3 Analisis dan Pembahasan Kesesuaian antara Perencanaan dan Realisasi	17
BAB 5 : Implementasi Sistem dan Analisis	19
5.1 Hasil dan Analisis Implementasi	19
5.1.1 Pengujian Kamera Webcam	19
5.1.2 Pengujian Parameter <i>Max Area</i>	20
5.1.3 Pengujian Algoritma Pendeteksi Jatuh	20
5.1.4 Pengujian Algoritma Terhadap Kondisi Berbeda.	21
5.2 Pengalaman Pengguna	29
5.3 Dampak Implementasi Sistem	30
5.3.1 Teknologi/Inovasi	30
5.3.2 Sosial	30
BAB 6 : Kesimpulan dan Saran	31
6.1 Kesimpulan	31
6.2 Saran	31
LAMPIRAN – LAMPIRAN	33

## RINGKASAN TUGAS AKHIR

Parkinson adalah salah satu penyakit yang biasa diderita oleh kalangan lansia. Suatu hal yang harus diperhatikan adalah penderita parkinson maupun lansia sering mengalami jatuh yang berisiko sangat berbahaya, sehingga dibutuhkan sistem pendeteksi jatuh untuk memantau keadaan penderita parkinson maupun lansia di rumah. Sistem ini menggunakan Raspberry Pi 3 sebagai *central unit processor* dan *web camera* sebagai masukan video. Sistem ini dipasang sesuai dengan prinsip dari pemasangan CCTV yaitu diletakkan di sudut atas ruangan dengan tinggi minimal 3 meter dengan tujuan dapat mendeteksi gerakan jatuh pada manusia secara lebih luas. Alat ini dapat mengirim pesan berupa gambar melalui aplikasi *telegram* untuk memberitahu anggota keluarga sebagai tanda *alarm* ketika objek terdeteksi jatuh. Sistem yang dibuat memiliki beberapa keunggulan dari sistem yang pernah dibuat sebelumnya yaitu sudah dilakukan secara *real time* dimana beberapa sistem yang sudah ada belum dilakukan secara *real time*. Sistem ini juga dapat membantu keluarga ketika memiliki anggota keluarga yang sudah lansia di rumahnya. Hal ini dikarenakan sistem dapat beroperasi sebagai penambah penjagaan terhadap penderita parkinson dan lansia.



## BAB 1 : Definisi Permasalahan

Parkinson adalah salah satu penyakit yang biasa diderita oleh kalangan lansia. Lansia (lanjut usia) merupakan seseorang yang telah memasuki usia 60 tahun ke atas berdasarkan UU Nomor 13 Tahun 1998. Pada tahun 2019 jumlah penduduk lanjut usia di Indonesia mencapai 25,64 juta orang, dimana pada tahun sebelumnya berjumlah 24,49 juta orang[1]. Menurut dokter spesialis saraf, dr Frandy Susatia, 1 dari 100 lansia adalah penderita parkinson. Dari data BPS tahun 2019, jika 1 dari 100 orang lansia penderita parkinson maka diperkirakan ada 25,64 ratus ribu lansia yang menderita parkinson, bahkan menurut data perhimpunan spesialis saraf indonesia mencatat sudah lebih dari 400.000 orang yang menderita parkinson. Parkinson merupakan gangguan sistem saraf pusat yang mempengaruhi sistem gerak pada manusia. Parkinson disebabkan oleh kurangnya zat dopamin yang ada di dalam otak, dimana kekurangan zat dopamin menyebabkan kerusakan otak, salah satu gejala parkinson adalah seringnya jatuh berulang karena ketidakstabilan berjalan dan keseimbangan berjalan[2]. Penelitian dari Goodwin dkk, membahas bahwa penderita parkinson mengalami jatuh sebanyak 577 kali dalam 20 minggu atau setiap 2 kali sehari[3].

Seiring berkembangnya teknologi saat ini, terdapat banyak metode untuk pengawasan jatuh pada manusia seperti pemasangan piranti elektronik pada tubuh manusia[4] dan penggunaan kamera[5]. Pemasangan piranti elektronik pada tubuh manusia memiliki kekurangan, faktor ingatan lansia yang sudah melemah ditambah penderita parkinson memiliki kesulitan dalam bergerak sehingga penggunaan piranti elektronik memiliki resiko hilang atau rusak. Selain itu sensor juga memiliki kelemahan akurasi dalam membedakan antara insiden jatuh dengan perilaku kegiatan sehari – hari seperti duduk dan membungkuk[4]. Penggunaan metode kamera yang digunakan oleh Wahyuni dkk, juga belum efektif karena pembacaan data tidak dilakukan secara langsung (*non real time*), sehingga keterlambatan dalam mengambil keputusan akan menjadi masalah besar dalam pengawasan jatuh manusia[5]. Dengan demikian, dari permasalahan tersebut dibutuhkan suatu sistem yang dapat mendeteksi gerak jatuh dari manusia secara yang lebih efektif dan tentunya nyaman digunakan oleh lansia dan penderita parkinson.

Langkah awal dalam *prototyping* sistem deteksi jatuh pada manusia menggunakan ruangan *indoor* yang tidak dalam kondisi gelap untuk pengambilan data sampel menggunakan satu kamera. Hal ini bertujuan agar kamera dapat mengambil citra dengan jelas dan dapat mendeteksi jatuh pada manusia. Usulan sistem ini memiliki tujuan sebagai solusi dari permasalahan pendeteksi gerak jatuh manusia dengan menggunakan *video processing* secara *real time*, sehingga dapat membantu dalam mencegah kondisi yang tidak diinginkan pada lansia dan penderita parkinson ketika mengalami jatuh terutama bagi yang berada di rumah seorang diri tanpa adanya pengawasan lebih dari keluarga.

## BAB 2 : Observasi

Proses observasi bertujuan untuk memastikan bahwa rancangan sistem yang diusulkan sesuai dengan batasan realistis yang ditentukan serta telah mengakomodasi kebutuhan awal prototyping yang telah disesuaikan dengan keinginan pengguna. Untuk mencapai hal tersebut, tahapan observasi ini diawali dengan mengumpulkan informasi-informasi dasar tentang kebutuhan sistem yang akan digunakan oleh pengguna, dalam hal ini adalah para anggota keluarga dari lansia dan penderita parkinson . Terdapat dua hal utama sebagai luaran dari proses observasi ini yaitu kumpulan informasi solusi yang memungkinkan dan spesifikasi sistem yang telah disesuaikan dengan kebutuhan pengguna.

Proses observasi diawali dengan mengumpulkan berbagai macam informasi yang berkaitan dengan solusi untuk menanggulangi rumusan permasalahan sebelumnya. Pada Tabel 2.1 menunjukkan beberapa kumpulan sumber informasi yang digunakan sebagai referensi dalam pembuatan sistem deteksi jatuh manusia secara real time.

Tabel 2.1. Kumpulan solusi yang identik dengan proyek tugas akhir

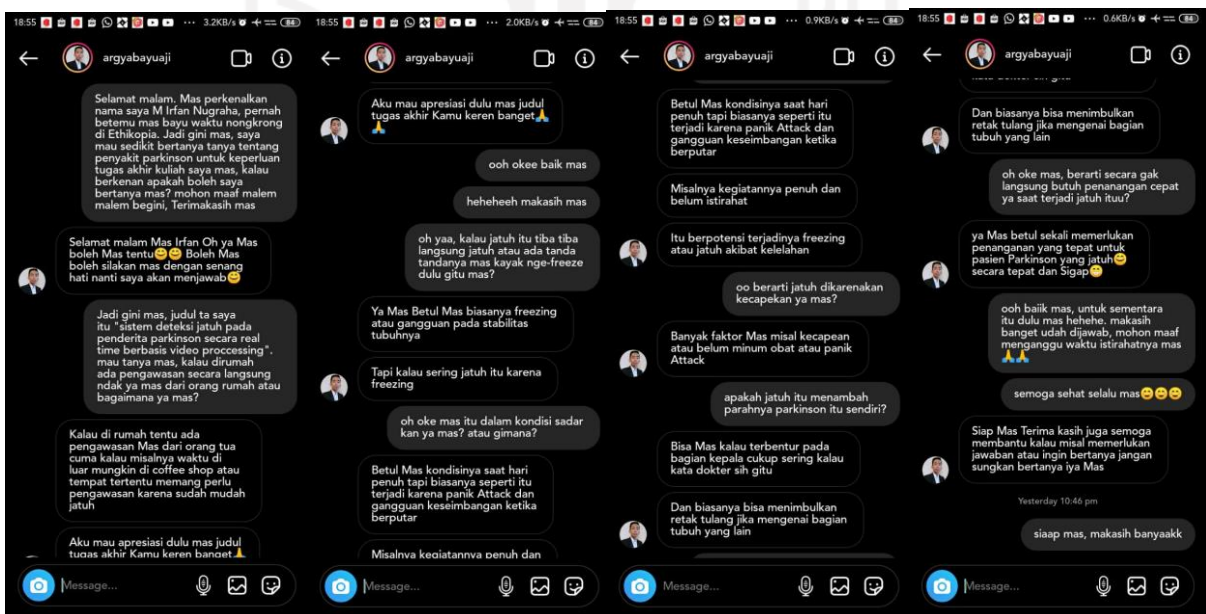
Penulis	Usulan Solusi	Hasil / Evaluasi
Wahyuni, dkk (2018) [5]	Mendeteksi jatuh seseorang menggunakan kamera dengan metode <i>Motion History Image (MHI)</i> dan <i>Approximaterd Ellipse</i> .	Metode yang digunakan sudah baik, namun sistem ini belum dilakukan secara real time.
Albawendi, dkk (2016) [6]	Perancangan sistem untuk mendeteksi jatuh manusia menggunakan video processing dengan metode <i>Timed Motion History Image (tMHI)</i> .	Pendekatan yang digunakan adalah kombinasi dari tMHI dan variasi bentuk manusia. Namun sistem ini belum dilakukan secara <i>real time</i>
Kreković, dkk (2012) [7]	Melakukan perancangan sistem untuk mendeteksi jatuh seseorang menggunakan kamera video secara real time. Metode yang digunakan adalah <i>Motion History Image (MHI)</i> , <i>Ellipse Moment</i> dengan bantuan dari C++ dan OpenCV.	Akurasi sudah baik sebesar 90% dan dilakukan secara <i>real time</i> , namun sistem yang dibuat belum ada keluaran pesan yang langsung diberikan kepada pengguna.
Umamakeswari, dkk (2015) [8]	Monitoring kelembaban tanah dengan menggunakan Arduino UNO. Media transmisi data menggunakan modul GSM mendeteksi gerak jatuh pada manusia menggunakan <i>video processing</i> secara <i>real time</i> . Alat yang digunakan adalah Raspberry Pi dengan modul komunikasi SIM800 GSM. Algoritma ditulis dalam bahasa C++ dan OpenCV	Mekanisme komunikasi menggunakan SIM800 GSM dengan keluaran pesan melalui pesan singkat (SMS) dan belum berbasis IoT.
Kumar (2019) [9]	Sistem deteksi jatuh berbasis video menggunakan Raspberry Pi secara	Konsep yang digunakan menggunakan Raspberry Pi 3 dengan keluaran output



	<i>real time</i> . Menggunakan metode background subtraction, siluet, kontur dan deteksi pergerakan manusia..	berupa pengiriman email dengan protokol SMTP.
--	---	---

Berdasarkan Tabel 2.1 dapat diketahui beberapa cara untuk melakukan deteksi jatuh pada manusia menggunakan *video processing* secara *real time*. Algoritma dari penelitian sebelumnya berbeda - beda, namun mayoritas menggunakan C++ dan Python yang telah dilengkapi dengan OpenCV. Peneliti sebelumnya juga ada yang menggunakan Raspberry Pi dan menggunakan Raspberry Pi. Berdasarkan daftar utama kebutuhan sistem dan dengan mengangkat isu *real time*, beberapa referensi diatas telah memenuhi kebutuhan tersebut. Konsep real time dari penelitian sebelumnya masih ditemukan beragam solusi, sehingga untuk menyelesaikan masalah tersebut, proses tahapan observasi perlu dilakukan untuk memenuhi kebutuhan pengguna melalui survey guna menentukan kebutuhan dan spesifikasi sistem yang sesuai.

Proses survey diawali dengan mencari salah satu penderita parkinson yang nantinya dijadikan narasumber. Setelah ditemukan narasumber yang tepat, dilakukan wawancara dengan narasumber tersebut yang nantinya digunakan untuk membantu menentukan spesifikasi sistem dan kebutuhan dari pengguna. Pada Gambar 2.1 menunjukan proses wawancara yang telah dilakukan dengan narasumber salah satu penderita parkinson dan pada Tabel 2.2 merupakan hasil transkrip dari wawancara tersebut.



Gambar 2.1 Proses Wawancara dengan Salah Satu Penderita Parkinson

Tabel 2.2. Hasil Transkrip Wawancara dengan Narasumber

Pertanyaan	Jawaban/tanggapan
Apakah dirumah ada pengawasan secara langsung dari keluarga?	Kalau dirumah tentu ada pengawasan dari orang tua, cuma kalau misalnya waktu di luar mungkin di <i>coffee shop</i> atau tempat tertentu memang perlu pengawasan karena saya mudah jatuh
Kalau jatuh itu tiba-tiba atau ada tanda-tandanya mas?	Iya mas ada. Biasanya freeze atau gangguan pada stabilitas tubuhnya. Tapi kalau sering jatuh itu karena freezing
Apakah hal itu terjadi dalam kondisi sadar mas?	Betul mas, biasanya seperti itu terjadi karena panik Attack dan gangguan keseimbangan ketika berputar. Misalnya kegiatan penuh dan belum istirahat, itu berpotensi terjadinya freezing atau jatuh akibat kelelahan.
Berarti jatuh dikarenakan kecapekan ya mas?	Banyak faktor mas, misal kecapekan, belum minum obat atau <i>Panic Attack</i>
Apakah jatuh itu menambahkan parahnya parkinson itu sendiri?	Bisa mas, kalau terbentur pada bagian kepala cukup sering bisa menimbulkan retak tulang jika mengenai bagian tubuh yang lain
Berarti secara tidak langsung butuh penanganan cepat ketika jatuh ya mas?	Iya mas, betul sekali memerlukan penanganan yang tepat untuk pasien Parkinson yang jatuh secara tepat dan sigap

Dari hasil survey/wawancara tersebut ditentukan beberapa spesifikasi dari sistem yang akan dikembangkan sebagai solusi permasalahan yang diangkat, yaitu sistem deteksi jatuh pada manusia secara *real time* menggunakan *video processing*. Berikut merupakan daftar spesifikasi lengkapnya.

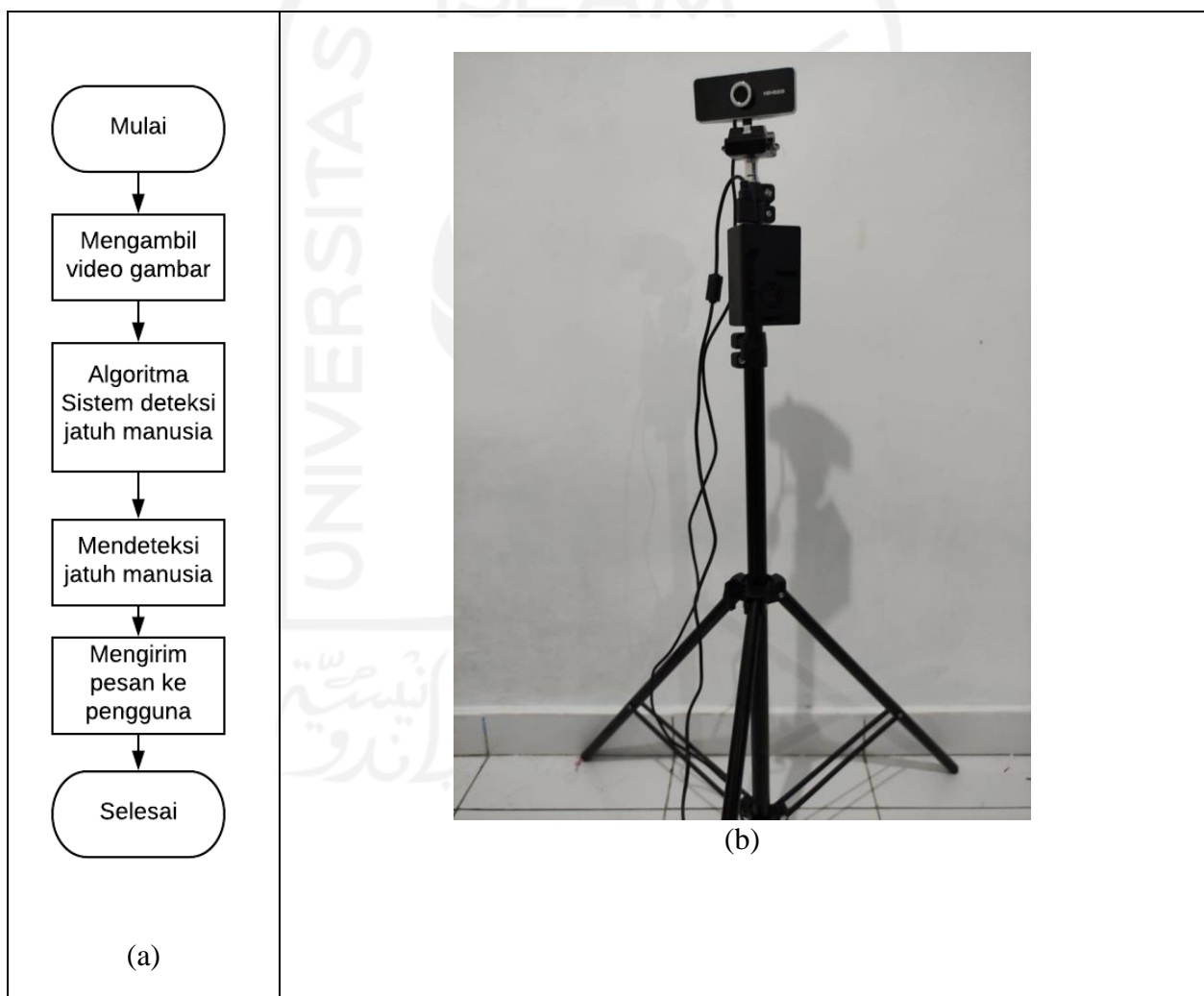
- *Central Processing Unit* berupa Raspberry pi 3 yang dapat menjalankan python dan OpenCV dan juga dapat terintegrasi dengan kamera.
- *Web Camera*.
- Koneksi internet untuk mengirim pesan.

Berdasarkan spesifikasi tersebut, maka selanjutnya akan dirancang usulan sistem yang memenuhi kriteria yang telah disebutkan diatas.

## BAB 3 : Usulan Perancangan Sistem

### 3.1 Usulan Rancangan Sistem

Sistem ini diberi nama TEPI JAMAN. TEPI JAMAN dirancang untuk memberi peringatan berupa pesan agar keluarga maupun orang terdekat bisa sesegera mungkin untuk menangani ketika lansia dan penderita parkinson mengalami jatuh. Gambar 3.1 adalah *flowchart* sistem yang akan dirancang. Perangkat *prototype* ini dapat dipasang didalam ruangan untuk mendeteksi jatuh dan mengirim pesan secara *real time*. Secara umum cara kerja sistem adalah saat sistem dinyalakan maka kamera akan otomatis mendeteksi pergerakan dari objek yang ada. Raspberry pi akan memproses citra yang di dapat, ketika objek jatuh maka akan mengirimkan pesan ke anggota keluarga maupun orang terdekat melalui aplikasi *telegram* yang sudah terintegrasi.







Gambar 3.1. Ilustrasi usulan rancangan sistem secara umum. (a)Diagram Blok Alur Penelitian, (b)implementasi alat yang digunakan

Sistem dibuat menggunakan *web camera* sebagai perangkat untuk menangkap video masukan yang telah terhubung dengan Raspberry Pi 3 berisi algoritma untuk mendeteksi gerak jatuh manusia. Video kemudian akan diproses menggunakan Raspberry Pi 3. Proses yang

dilakukan adalah mengubah citra menjadi *grayscale*, *background subtraction*, dan deteksi tepi, kemudian menentukan parameter jatuh menggunakan metode *max area*. Ketika objek terdeteksi jatuh maka sistem akan mengirim pesan ke pengguna melalui aplikasi bot Telegram sedangkan saat tidak terdeteksi jatuh maka proses akan berhenti dan memulai proses baru. Untuk dapat memenuhi usulan sistem tersebut, maka diperlukan inventarisasi kebutuhan sistem perangkat keras. Tabel 3.1 memperlihatkan kebutuhan sistem sesuai usulan dan spesifikasi yang dibutuhkan. Untuk dapat memenuhi usulan sistem tersebut, maka diperlukan inventarisasi kebutuhan sistem perangkat keras. Tabel 3.1 memperlihatkan kebutuhan sistem sesuai usulan dan spesifikasi yang dibutuhkan.

Tabel 3.1. Inventarisasi kebutuhan usulan sistem perangkat keras TEPI JAMAN

No	Nama Alat	Keterangan
1	Perangkat untuk kemasan alat 	Dibuat untuk menjadi tempat <i>mounting</i> alat yang telah didesain agar dapat melindungi dan terlihat lebih tertutup.
2	Raspberry pi 3 	Untuk <i>central processing unit</i> digunakan adalah Raspberry pi 3. karena bisa menjalankan python dan OpenCV yang jua dapat terintegrasi dengan kamera. Selain itu Raspberry Pi 3 juga memiliki ukuran yang kecil sehingga bisa dipasang dimana saja.
3	Web Camera 	<i>Web camera</i> menggunakan USB sebagai kabel Output sehingga bisa diakses oleh Raspberry Pi 3. <i>Web camera</i> ini memiliki spesifikasi HD 960P.
4	Power Supply 5V/2,5A 	Power supply yang digunakan adalah 5V/2,5A, sesuai dengan yang dibutuhkan Raspberry pi 3.

### 3.2 Metode Uji Coba dan Pengujian Usulan Rancangan Sistem

Sistem deteksi jatuh pada manusia menggunakan *video proccesing* secara *realtime* menggunakan pengolahan citra dengan bahasa pemograman *Python* dan sudah terintegrasi dengan OpenCV yang diproses dengan Raspberry Pi 3.

#### 3.2.1 Alur Penelitian

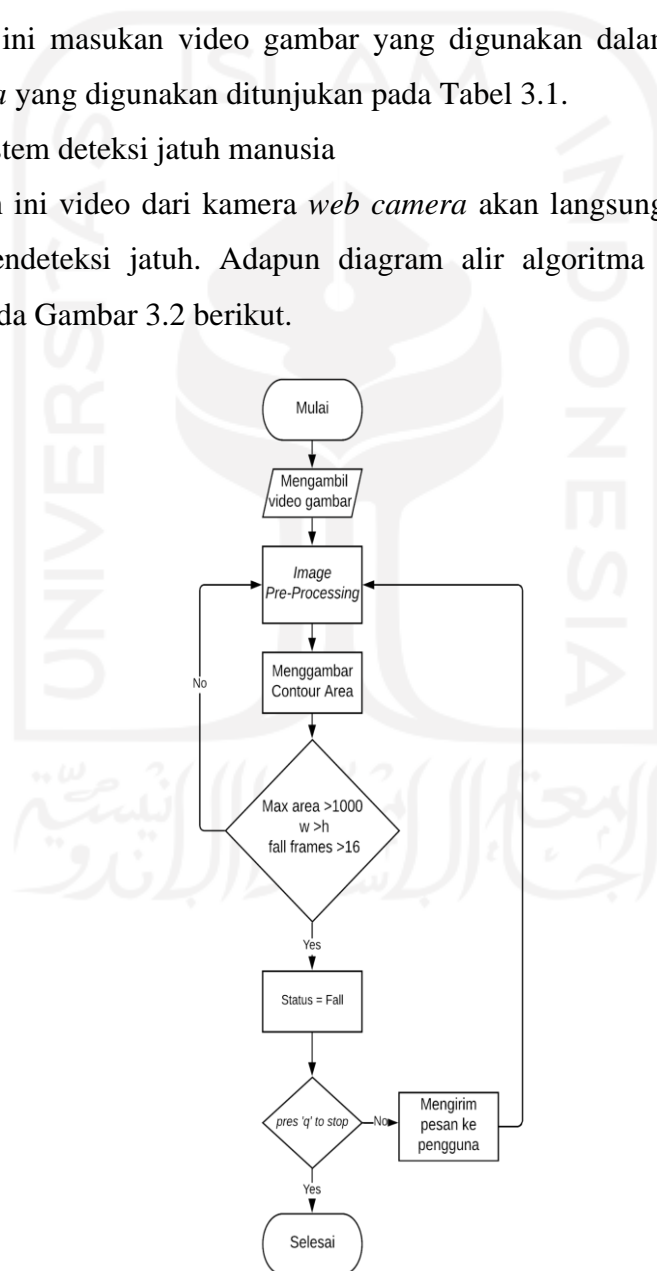
Penelitian sistem pendeteksi jatuh pada manusia ini memiliki beberapa tahap, dimana setiap tahap memiliki fungsi masing-masing untuk menyelesaikan sistem. Gambar 3.1.a merupakan diagram blok penelitian tugas akhir ini.

##### 1. Mengambil video gambar

Pada sistem ini masukan video gambar yang digunakan dalam sistem ini adalah *web camera*. *Web camera* yang digunakan ditunjukkan pada Tabel 3.1.

##### 2. Algoritma sistem deteksi jatuh manusia

Pada tahapan ini video dari kamera *web camera* akan langsung dianalisa menggunakan algoritma sistem pendeteksi jatuh. Adapun diagram alir algoritma sistem pendeteksi jatuh direpresentasikan pada Gambar 3.2 berikut.



Gambar 3.2 Flowchart Algoritma Sistem Deteksi Jatuh Manusia

### 3. Mendeteksi jatuh manusia

Proses deteksi jatuh manusia dilakukan sesuai dengan prosedur pada Gambar 3.2. Sistem deteksi jatuh manusia menggunakan perangkat Raspberry Pi 3 dengan bahasa *Python* yang telah terintegrasi dengan OpenCV. Berikut adalah prosedur yang dilakukan pada tahapan ini.

#### 3.1. Mengambil gambar masukan

*Input* dari sistem deteksi jatuh ini adalah citra video yang ditangkap pada *Web Camera* secara langsung, lalu dijadikan masukan untuk program *Python* yang telah dilengkapi dengan *library* OpenCV untuk membangun parameter yang akan digunakan sebagai acuan pendeteksi jatuh pada sistem.

#### 3.2 *Image Pre-Processing*

Proses pertama yang dilakukan pada data input video adalah *pre-processing*, proses ini dilakukan sebelum menentukan kondisi jatuh. Adapun proses-proses yang dilakukan terhadap data input video adalah *Grayscale*, *Background Subtraction* dan *Edge*.

##### 3.2.1 *Grayscale*

Sebelum melakukan pemrosesan citra pada video, dilakukan perubahan warna dengan metode *grayscale*. Metode *grayscale* digunakan untuk mengubah warna video gambar dari BGR ( *Blue, Green, Red* ) menjadi abu-abu.

##### 3.2.2 *Background Subtraction*

Video hasil *grayscale* digunakan sebagai masukan untuk tahapan berikutnya yaitu proses *Background Subtraction*. Metode *Background Subtraction* digunakan untuk mengolah citra dan memisahkan objek dengan latar. Hasil dari proses *Background Subtraction* adalah citra biner dimana objek yang dideteksi memiliki nilai piksel bernilai 1 (Putih) sedangkan background di sekitar objek memiliki nilai piksel bernilai 0 (hitam).

##### 3.2.3 *Edge*

Video hasil *grayscale* digunakan sebagai masukan untuk tahapan berikutnya yaitu proses *Edge*. Metode *Edge* menghasilkan tepi-tepi dari objek-objek citra dengan tujuan menandai bagian yang menjadi detail gambar/citra untuk memperbaiki detail dari gambar/citra yang blur, yang terjadi karena adanya efek dari proses akuisisi citra. Suatu titik (x,y) dikatakan sebagai tepi (*edge*) dari suatu citra bila titik tersebut mempunyai perbedaan yang tinggi dengan tetangganya.



### 3.3 Menentukan *Max area* dan Waktu ketika Jatuh

Setelah semua tahap sudah dilaksanakan maka hasil pendeteksi jatuh akan dianalisa. Status data input akan dianalisa apakah merupakan data jatuh atau tidak, penentuan tersebut berdasarkan status frames dari *web camera*. Ketika *max area*  $w$  (*width*)  $>$   $h$  (*high*) dan jatuh terdeteksi pada  $>$  16Fps, maka data tersebut akan dianggap sebagai data jatuh. Kondisi tersebut digunakan karena dianggap lebih bagus dari kondisi lain, yaitu ketika menggunakan kondisi “Jatuh terdeteksi  $>$  8Fps”, algoritma menjadi sangat sensitif dalam mendeteksi jatuh. Sedangkan ketika menggunakan kondisi “Jatuh terdeteksi  $>$  20Fps”, menjadi lebih lambat dan akurasi deteksi jatuh menjadi berkurang.

#### 4. Mengirim pesan ke pengguna

Setelah diketahui terjatuh, maka hasil akan dikirim melalui *bot Telegram*. Hasil yang dikirim berupa gambar dengan format JPG dan teks yang bertuliskan “*fall*”.

### 3.2.2 Pengujian Usulan Rancangan Sistem

Pengujian dilakukan dengan membandingkan *prototype* dengan spesifikasi yang telah dibuat sebelumnya. Setelah dilakukan pengujian ternyata spesifikasi yang ada tidak sesuai dengan hasil yang diinginkan sehingga dilakukan perubahan spesifikasi. Perubahan spesifikasi dilakukan dengan memodifikasi *prototype* sebelumnya baik komponen maupun algoritmanya. Setelah melakukan modifikasi, maka dilakukan pengujian untuk mengetahui apakah sistem pendeteksi jatuh dapat bekerja dengan baik. Pengujian ini dimulai dengan menguji masukan *web camera*, pengujian parameter objek dengan metode *max area* dan melihat hasil kinerja algoritma pendeteksi jatuh. Pengujian yang dilakukan meliputi berikut:

- Pengujian *web camera*
- Pengujian Parameter *Max area*
- Pengujian Algoritma Pendeteksi Jatuh
- Pengujian Algoritma Terhadap Kondisi Berbeda
  - b. Pengujian Algoritma Pendeteksi Jatuh Terhadap Posisi Objek yang Berbeda
  - c. Pengujian Algoritma Pendeteksi Jatuh Terhadap Posisi Jatuh yang Berbeda
  - d. Pengujian Algoritma Pendeteksi Jatuh Terhadap Intensitas Cahaya Berbeda
  - e. Pengujian Algoritma Pendeteksi Jatuh Terhadap Warna Objek Berbeda
  - f. Pengujian Algoritma Pendeteksi Jatuh Terhadap Objek Berbeda
  - g. Pengujian Algoritma Pendeteksi Jatuh Terhadap Jumlah Objek Berbeda
  - h. Pengujian Algoritma Pendeteksi Jatuh Terhadap Benda Lain
  - i. Pengujian pengiriman *Alarm* dengan aplikasi bot Telegram

## BAB 4 : Hasil Perancangan Sistem

### 4.1 Kesesuaian Usulan dan Hasil Perancangan Sistem

Sistem yang dibuat awalnya menggunakan CCTV sebagai perangkat untuk menangkap video masukan yang telah terhubung dengan Raspberry pi 3. Setelah diuji coba, masukan kabel dari CCTV adalah kabel video output bukan kabel USB sehingga tidak bisa digunakan pada Raspberry Pi 3 dan digantikan dengan Pi Camera sebagai masukan Video. Setelah dilakukan uji coba menggunakan Pi Camera, gambar yang dihasilkan tidak begitu cerah dan buram sehingga Pi Camera digantikan dengan *web camera* yang mempunyai kualitas cukup baik untuk menampilkan gambar.



Gambar 4.1 Rencana awal



Gambar 4.2 Realisasi

Tabel 4.1 Perbandingan usulan dan hasil perancangan sistem

No	Spesifikasi	Usulan	Realisasi
1	Kualitas Gambar	Full HD 1080p (tertulis)	Widescreen HD 960p
2	Fleksibilitas	1 tempat	Bisa ditaruh dimana saja

### 4.2 Kesesuaian Perencanaan dalam Manajemen Tim dan Realisasinya

Perencanaan perancangan sistem ditargetkan dikerjakan bulan April, namun sudah terealisasi sejak bulan Maret. Hal ini dikarenakan tim memutuskan untuk mengikuti program percepatan yang telah disusun oleh jurusan. Dengan ikut berpartisipasi program percepatan tersebut, membuat tim mempercepat dan mempersingkat waktu dengan tetap menjaga kualitas produk yang dibuat. Perbandingan *head-to-head* antara perencanaan dan realisasi *timeline* dapat dilihat pada Tabel 4.2.



Tabel 4.2 Kesesuaian antara usulan dan realisasi *timeline* pengerjaan Tugas Akhir 2

No	Kegiatan	Usulan waktu	Realisasi Pelaksanaan
1	Pembelian alat dan bahan	Maret – April	Maret
2	Perancangan sistem dengan usulan	April – Juni	Maret - April

Usulan dan realisasi pada RAB tugas akhir juga mengalami perubahan. Hal ini dikarenakan terdapat perubahan spesifikasi dari produk, sehingga diperlukan penyesuaian komponen untuk memenuhi spesifikasi hasil modifikasi yang dibuat. Perubahan RAB tersebut dapat dilihat pada Tabel 4.3. berikut:

Tabel 4.3 Kesesuaian RAB Tugas Akhir antara usulan dan realisasi

No	Jenis Pengeluaran	Usulan Biaya		Realisasi Biaya	
		Kuantitas	Total Harga	Kuantitas	Total Harga
1	Raspberry Pi 3 sepaket	1 paket	Rp. 1.000.000,-	1 pcs	Rp. 908.700,-
2	Case Raspberry Pi 3	-	-	1 pcs	Rp. 39.500,-
3	Tripod	-	-	1 pcs	Rp. 50.000,-
4	Tongsis	-	-	1 pcs	Rp. 55.000,-
	Total		Rp. 1.000.000,-		Rp. 1.053.200,-

#### 4.3 Analisis dan Pembahasan Kesesuaian antara Perencanaan dan Realisasi

Pengujian *prototype* dilakukan untuk mengetahui tingkat kesesuaian antara perencanaan dengan realisasi yang telah dibuat pada TA103. Setelah dilakukan pengujian, ternyata didapatkan beberapa hal yang harus diubah, sehingga dilakukan modifikasi spesifikasi sebelumnya. Berikut merupakan spesifikasi sebelum dilakukan perubahan:

- Digunakan kamera CCTV sebagai masukan.
- Menggunakan software Matlab sebagai aplikasi pengoprasian.
- Alarm sebagai keluaran.

Setelah dilakukannya pengujian dan konsultasi dengan peneliti sebelumnya, penggunaan program matlab dan CCTV jika digunakan dalam video processing secara real time ternyata memakan waktu yang cukup lama. Sehingga diputuskan untuk melakukan perubahan spesifikasi menjadi berikut:

**A. Input yang digunakan sebelumnya (Pi Camera) dirubah menjadi *web camera***

Sistem yang dibuat tim *capston design* tetap menggunakan prinsip dari CCTV dalam peletakan kamera meskipun menggunakan *web camera*. *Web camera* diletakkan pada sudut ruangan dengan tinggi 3 meter, hal ini dikarenakan program akan lebih efektif dalam mendeteksi gerak jatuh manusia ketika berada di sudut ruangan.

**B. Software yang digunakan sebelumnya (Matlab) diganti menjadi Python yang telah dilengkapi OpenCV dan terinstal pada Raspberry pi 3.**

Mayoritas *software* yang digunakan pada penelitian sebelumnya dalam mendeteksi gerak jatuh manusia dengan video processing secara real time adalah Python yang telah dilengkapi dengan OpenCV. Selain itu setelah dilakukan diskusi dengan peneliti judul ini sebelumnya (Moh. Brado), dimana beliau menggunakan *software* Matlab untuk pengolahan gambar yang memakan waktu cukup lama untuk pengolahan yaitu sebesar satu menit jika programnya dibuat real time. Maka dari itu diputuskan mengganti *software* matlab menjadi bahasa Python.

**C. Output yang digunakan tetap sama yaitu berupa alarm.**

*Alarm* yang direncanakan sebelumnya adalah mengirim notifikasi melalui Email. Setelah dilakukan pertimbangan, penggunaan Email dirasa kurang efektif. Hal ini dikarenakan penerimaan notifikasi pada email tidak cepat, sehingga jika terdapat notifikasi orang-orang tidak bisa langsung tau. Maka dari itu dilakukan perubahan *alarm* dari Email menjadi aplikasi Telegram yang memiliki kelebihan lebih cepat untuk mengirim dan menerima notifikasi.

## BAB 5 : Implementasi Sistem dan Analisis

### 5.1 Hasil dan Analisis Implementasi

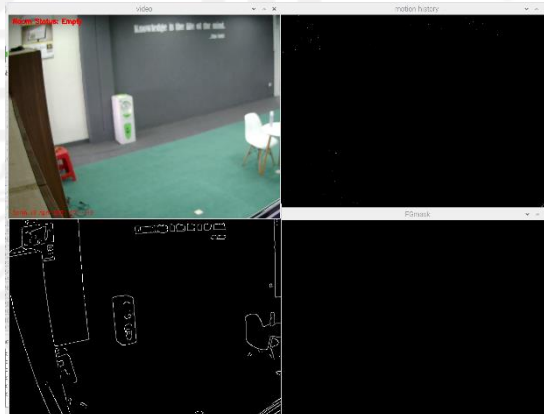
Setelah melakukan perubahan spesifikasi, dilakukan pengujian pada sistem yang telah dimodifikasi. Pengujian dilakukan untuk mengetahui apakah sistem pendeteksi jatuh dapat bekerja dengan baik. Pengujian ini dimulai dengan menguji masukan kamera Webcam, pengujian parameter objek dengan metode *max area* dan melihat hasil kinerja algoritma pendeteksi jatuh.

#### 5.1.1 Pengujian Kamera Webcam

Pengujian ini dilakukan dengan meletakkan Webcam pada sudut yang berbeda-beda. Percobaan menggunakan 2 sudut kamera yang berbeda, yaitu tengah ruangan seperti pada Gambar 5.1 dan sudut ruangan seperti yang ditunjukkan pada Gambar 5.2.



Gambar 5.1 Posisi Kamera Webcam di tengah ruangan

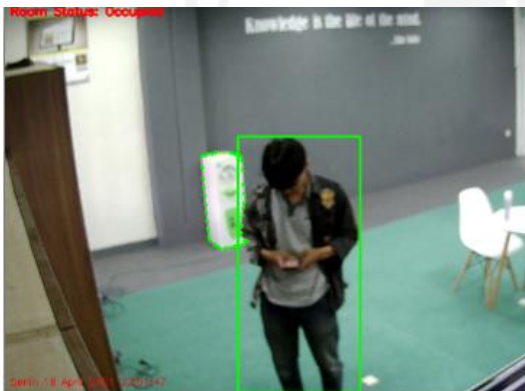


Gambar 5.2 Posisi kamera Webcam berada di sudut ruangan

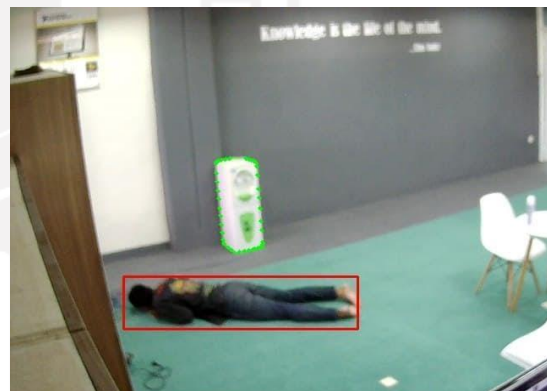
Hasil percobaan menunjukkan bahwa kamera yang diletakkan pada sudut ruangan lebih efektif dibandingkan di tengah ruangan. Hal ini dikarenakan sudut pandang pada sudut ruangan lebih lebar dibandingkan dengan di tengah ruangan, sehingga objek lebih mudah dideteksi.

### 5.1.2 Pengujian Parameter *Max Area*

Pengujian ini menggunakan *max area* sebagai acuan untuk mendeteksi apakah terdapat gerakan jatuh pada objek atau tidak. Parameter ini menyatakan objek dianggap jatuh ketika  $w > h$  (*width*) > *h* (*high*) dan jatuh terdeteksi ketika objek jatuh selama lebih dari 16Fps. Pengujian ini dilakukan dengan merubah nilai Fps ketika jatuh. Nilai *frame* kamera yang digunakan dalam pengujian adalah 8,16 dan 20. Ketika nilai Fps adalah 16, sistem akan mendeteksi dengan akurasi paling baik. Ketika nilai Fps adalah 8 akan membuat sistem terlalu sensitif, sehingga terlalu mudah objek didefinisikan mengalami *fall*. Ketika nilai Fps adalah 20, akan membuat sistem terlalu lama dalam mendefinisikan objek terjatuh. Pada pengujian ini juga didapatkan hasil bahwa parameter  $w > h$  masih kurang efektif digunakan untuk mendeteksi objek terjatuh. Hal ini dikarenakan tidak ada parameter tambahan yang digunakan untuk membedakan objek tersebut manusia atau bukan. Sehingga asalkan objek tersebut bergerak dan lebarnya lebih besar dari tingginya maka akan dinyatakan jatuh. Gambar 5.3 merupakan tampilan ketika objek tidak terdeteksi jatuh dan Gambar 5.4 merupakan tampilan ketika objek terdeteksi jatuh.



Gambar 5.3 Tampilan Tidak Jatuh



Gambar 5.4 Tampilan Saat Jatuh

### 5.1.3 Pengujian Algoritma Pendeteksi Jatuh

Pengujian ini dilakukan bertujuan untuk mengetahui apakah algoritma pendeteksi jatuh objek dapat bekerja. Algoritma pendeteksi jatuh menggunakan data input parameter yang dihasilkan oleh proses sebelumnya. Gambar 5.5 merupakan tampilan sistem pendeteksi jatuh Edges, Gambar 5.6 merupakan tampilan sistem pendeteksi jatuh Fgmask, dan Gambar 5.6 menunjukkan hasil dari sistem pendeteksi jatuh menggunakan metode MHI.



Gambar 5.5 Tampilan Edges



Gambar 5.6 Tampilan Fgmask

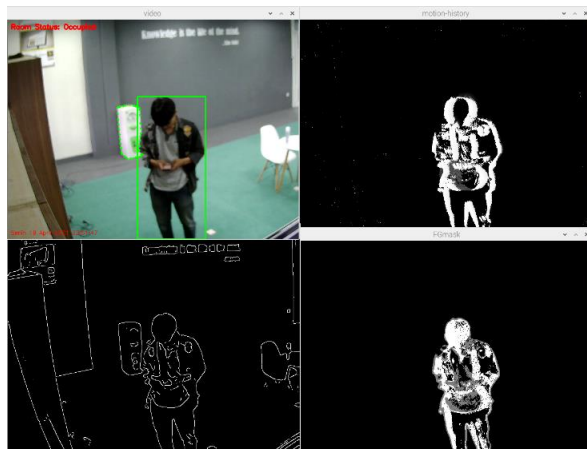


Gambar 5.7 Tampilan MHI

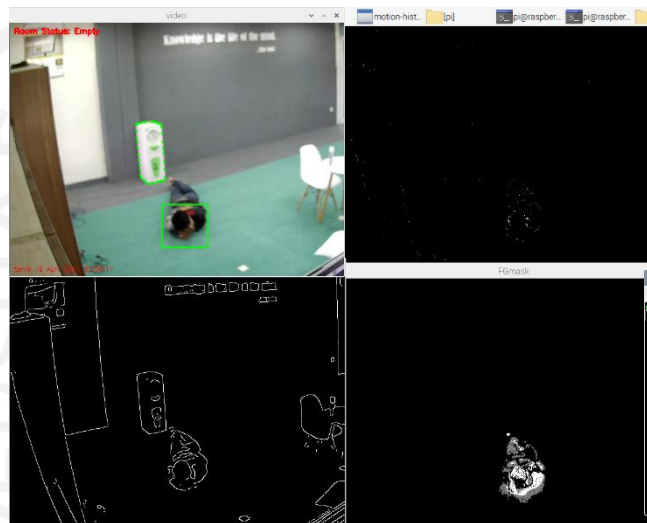
#### 5.1.4 Pengujian Algoritma Terhadap Kondisi Berbeda.

##### 1. Pengujian Algoritma Pendeteksi Jatuh Terhadap Posisi Objek yang Berbeda

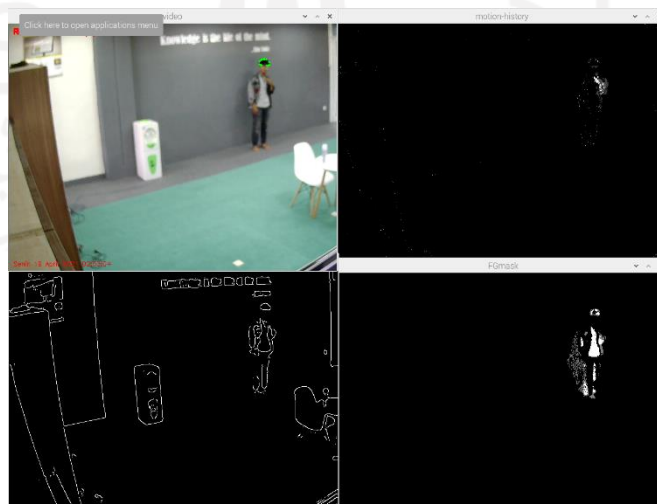
Pengujian ini bertujuan untuk mengetahui apakah posisi jatuh objek berpengaruh terhadap algoritma deteksi jatuh. Perbedaan posisi objek ini berhubungan dengan jarak antara objek terhadap kamera. Percobaan ini dilakukan dengan mengambil data jatuh pada posisi berbeda. Gambar 5.8 merupakan pengujian dengan objek berada di dekat kamera, Gambar 5.9 merupakan pengujian dengan objek berada di tengah ruangan dan Gambar 5.10 merupakan pengujian dengan objek yang berada pada posisi jauh dengan kamera.



Gambar 5.8 Objek berada didekat kamera



Gambar 5.9 Objek berada di tengah ruangan



Gambar 5.10 Objek pada posisi jauh dari kamera

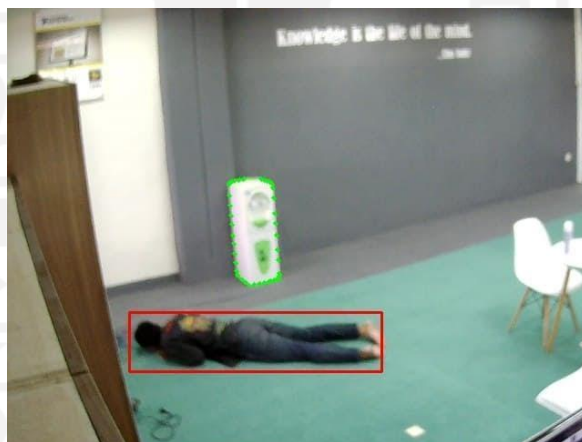
Gambar menunjukkan bahwa posisi objek yang berada di tengah ruangan merupakan posisi yang paling ideal untuk mendeteksi jatuh, hal ini dikarenakan semua pergerakan dari objek dapat terlihat jelas oleh kamera, sehingga data parameter yang didapat menjadi lebih akurat.

## 2. Pengujian Algoritma Pendeteksi Jatuh Terhadap Posisi Jatuh yang Berbeda.

Pengujian ini bertujuan untuk mengetahui bagaimana pengaruh posisi jatuh terhadap performa algoritma deteksi jatuh. Pengambilan data untuk percobaan ini dilakukan dengan objek jatuh kedepan dan kesamping. Gambar 5.11 menampilkan posisi ketika objek jatuh ke depan dan Gambar 5.12 menampilkan posisi ketika objek jatuh ke samping.



Gambar 5.11 Posisi objek jatuh ke depan.



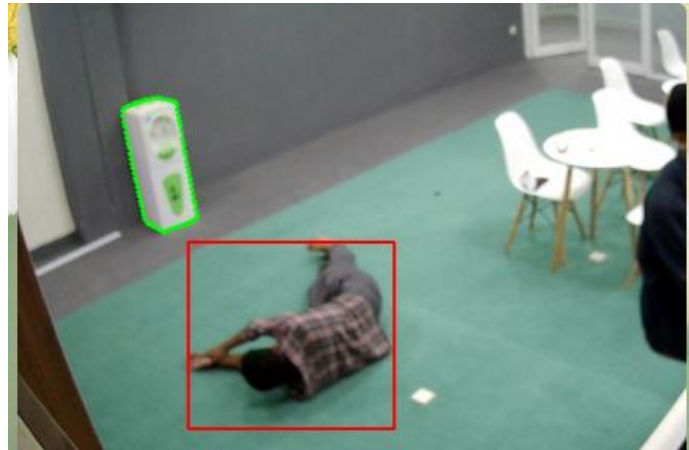
Gambar 5.12 Posisi objek jatuh ke samping.

Untuk posisi jatuh objek, pada Gambar 5.12 yang menampilkan posisi objek jatuh ke samping lebih akurat dibandingkan dengan sejajar dengan sudut kamera.

## 3. Pengujian Algoritma Pendeteksi Jatuh Terhadap Intensitas Cahaya Berbeda.

Pengujian ini bertujuan untuk mengetahui bagaimana pengaruh intensitas cahaya terhadap performa algoritma deteksi jatuh. Pengambilan data untuk percobaan ini dilakukan dengan menggunakan intensitas cahaya terang, redup dan cahaya gelap. Gambar 5.13 menampilkan objek dapat terdeteksi dengan baik saat cahaya terang dan Gambar 5.14 menampilkan objek tidak dapat terdeteksi saat cahaya gelap.





Gambar 5.13 Tampilan dalam kondisi terang



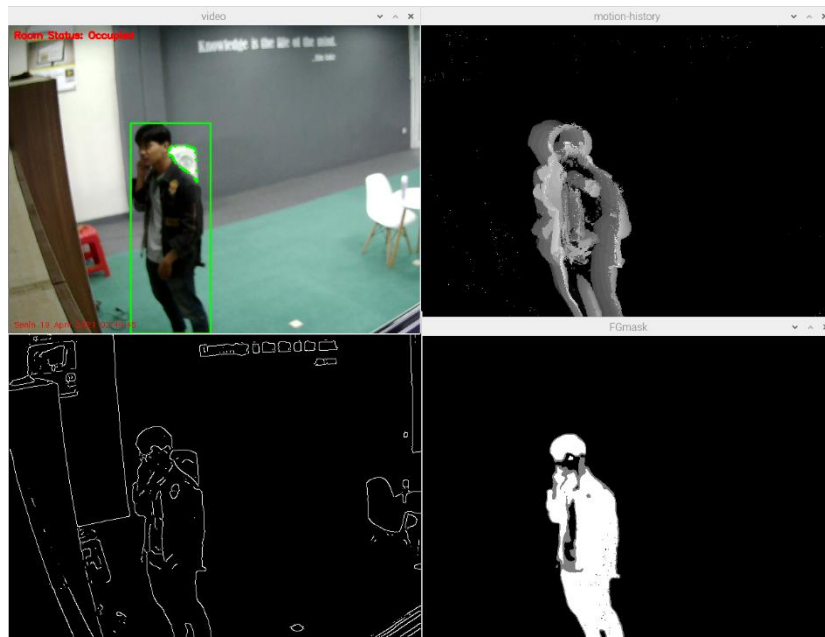
Gambar 5.14 Tampilan dalam kondisi gelap.

Hasil pengujian pada percobaan ini menunjukkan bahwa ketika cahaya terang algoritma dapat mendeteksi jatuh dengan baik, hal ini dikarenakan kamera dapat menangkap perubahan parameter dengan lebih jelas dan akurat.

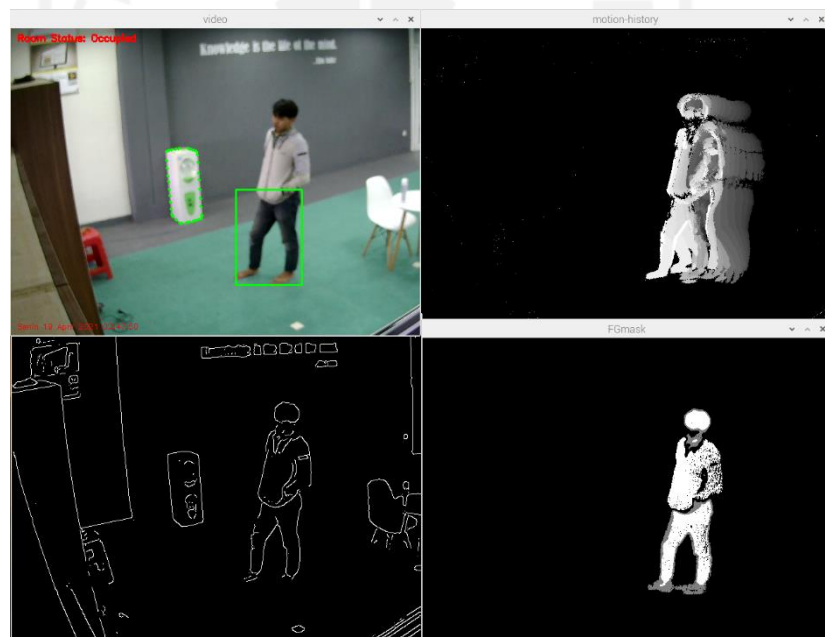
#### 4. Pengujian Algoritma Pendeteksi Jatuh Terhadap Warna Objek Berbeda.

Pengujian ini bertujuan untuk mengetahui bagaimana pengaruh perbedaan warna objek dengan background terhadap performa algoritma deteksi jatuh. Pengambilan data untuk percobaan ini dilakukan dengan menggunakan warna yang sama dan berbeda dengan background. Gambar 5.15 menunjukkan ketika objek memiliki warna yang berbeda dengan *background* sistem dapat mendeteksi objek dengan akurat, sedangkan Gambar 5.16 menunjukkan ketika objek memiliki warna yang sama dengan *background* sistem kurang akurat saat mendeteksi objek.





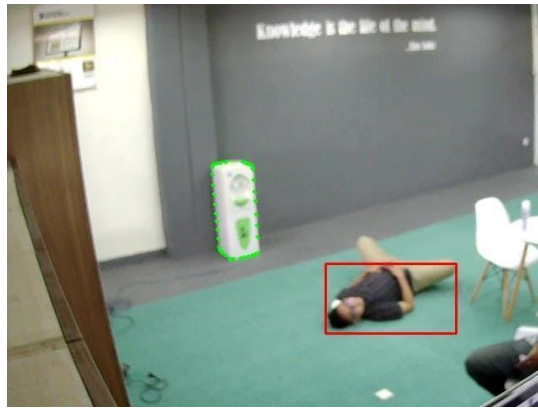
Gambar 5.15 Objek dengan warna berbeda dengan *background*



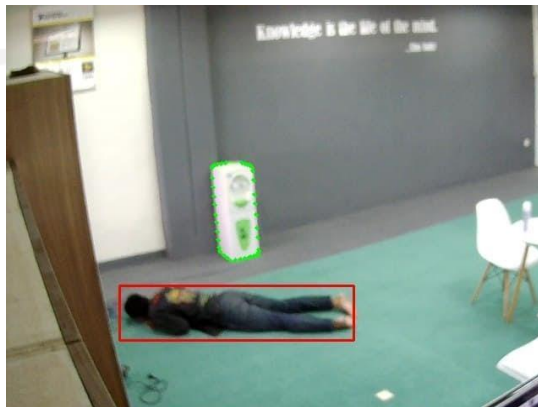
Gambar 5.16 Objek dengan warna sama dengan *background*

##### 5. Pengujian Algoritma Pendeteksi Jatuh Terhadap Objek Berbeda.

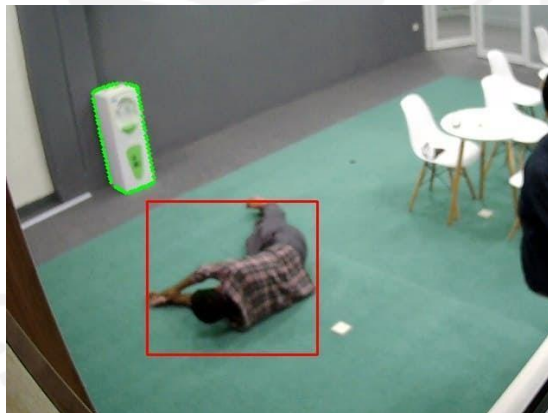
Pengujian ini bertujuan untuk mengetahui bagaimana pengaruh perbedaan objek yang ditangkap kamera terhadap performa algoritma deteksi jatuh. Pengambilan data untuk percobaan ini dilakukan dengan mengambil data 3 objek berbeda.



Gambar 5.17 Objek jatuh 1



Gambar 5.18 Objek jatuh 2



Gambar 5.19 Objek jatuh 3

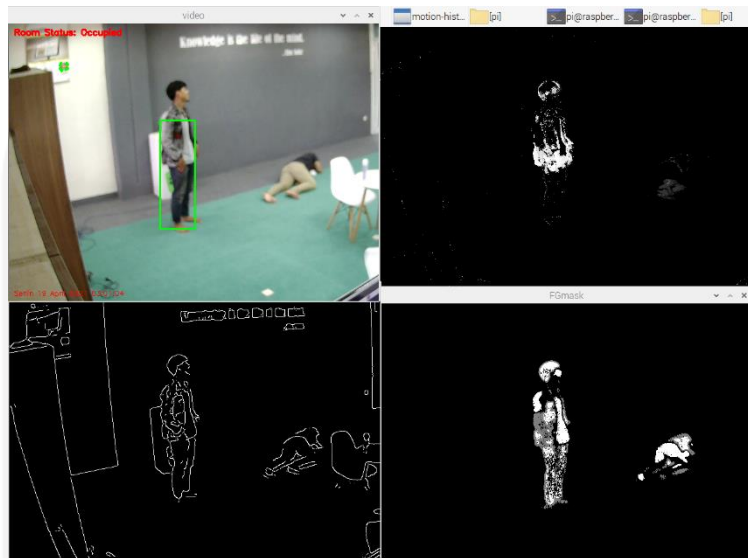
Hasil pengujian ini menunjukkan bahwa kamera dapat mendeteksi objek yang berbeda, pengujian dilakukan dengan mengambil beberapa data.

#### 6. Pengujian Algoritma Pendeteksi Jatuh Terhadap Jumlah Objek Berbeda.

Pengujian ini bertujuan untuk mengetahui bagaimana pengaruh perbedaan jumlah objek yang ditangkap kamera terhadap performa algoritma deteksi jatuh. Pengambilan data untuk percobaan ini dilakukan dengan mengambil data dengan jumlah objek berbeda. Gambar 5.20 menampilkan ketika hanya ada satu orang dalam *frame* kamera dan Gambar 5.21 menampilkan ada dua orang dalam *frame* kamera.



Gambar 5.20 Objek berjumlah 1 orang

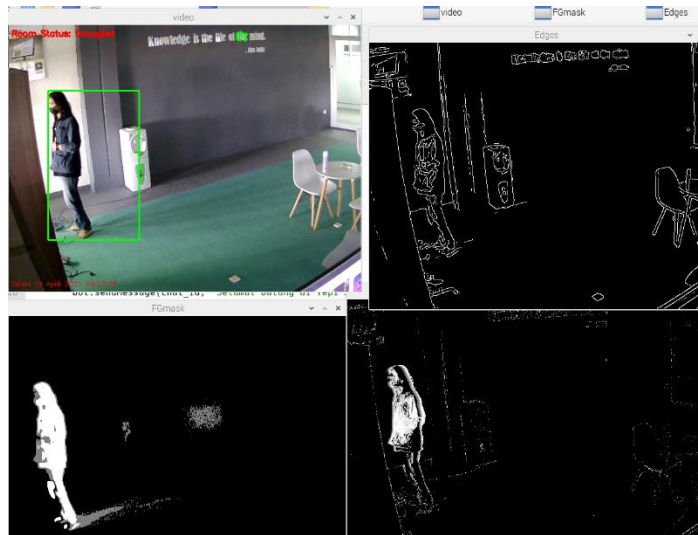


Gambar 5.21 Objek berjumlah 2 orang

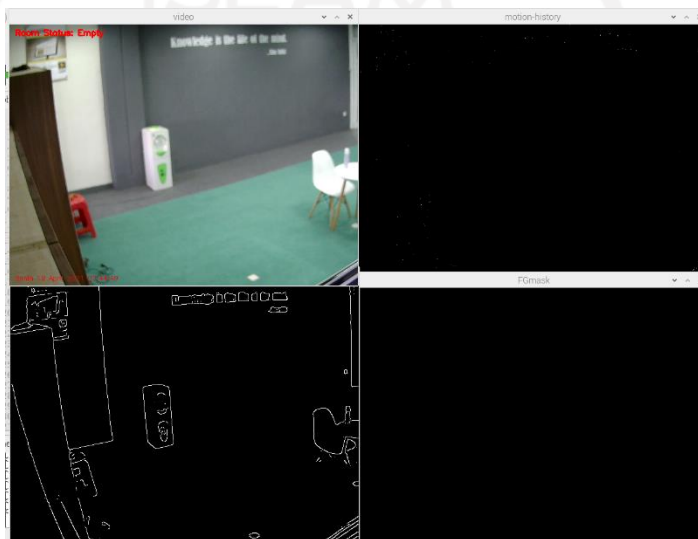
Hasil pengujian ini menunjukkan bahwa kamera tidak dapat mendeteksi objek dengan jumlah lebih dari satu.

#### 7. Pengujian Algoritma Pendeteksi Jatuh Terhadap Benda Lain

Pengujian ini bertujuan untuk mengetahui bagaimana pengaruh benda yang ditangkap kamera terhadap performa algoritma deteksi jatuh. Pada percobaan ini dilakukan dengan benda yang ada di ruangan dan bayangan objek. Gambar 5.22 menampilkan tampilan sistem ketika ada objek lain berupa bayangan dan Gambar 5.23 menampilkan tampilan sistem ketika tidak ada objek manusia sama sekali.



Gambar 5.22 Bayangan pada objek

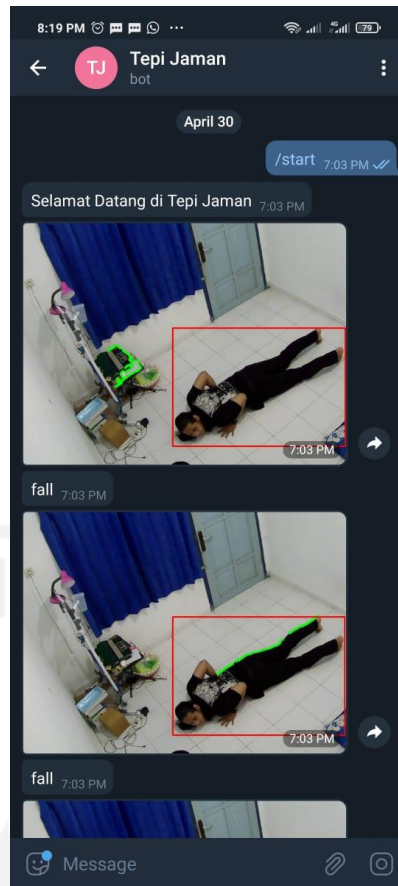


Gambar 5.23 Benda pada ruangan

Dari hasil pengujian didapatkan hasil bahwa bayangan objek dapat terdeteksi oleh algoritma. Sedangkan benda yang ada di dalam ruangan tidak terdeteksi oleh algoritma asalkan benda tersebut tidak bergerak.

#### 8. Pengujian Pengiriman *Alarm* dengan Aplikasi Bot Telegram

Pengujian ini bertujuan untuk mengetahui keberhasilan sistem yang dibuat dalam mengirimkan *alarm*. Pesan yang dikirimkan berisikan pesan singkat dan gambar ketika objek terdeteksi jatuh kepada pengguna seperti yang ditunjukkan pada Gambar 5.24.



Gambar 5.24 Tampilan Bot Telegram

Hasil dari pengujian adalah pesan yang dikirimkan memiliki *delay* minimal 4 detik. Hal ini dikarenakan untuk mengirim gambar memerlukan waktu lebih dibanding mengirim pesan singkat saja.

## 5.2 Pengalaman Pengguna

Implementasi sistem alat ini dapat digunakan langsung oleh penderita prkinson untuk pemantauan ketika di dalam ruangan. Dari pengalaman penggunaan sistem tersebut dihasilkan beberapa masukan dan kendala yang tentunya dibutuhkan perbaikan- perbaikan kembali untuk pengembangan alat ini agar lebih baik kedepannya. Berikut merupakan beberapa penjelasan singkat dari pengguna terhadap alat ini yang ditunjukkan pada Tabel 5.2.

Tabel 5.2 Pengalaman Pengguna

No	Fitur/Komponen	Capaian	Aksi/Perbaikan
1	Fungsi	Fungsi sebagai pendeteksi jatuh pada manusia menggunakan <i>video processing</i> secara <i>real time</i> sudah berjalan dengan baik.	Dipertahankan
2	Kemudahan	Pengoperasian <i>prototype</i> cukup mudah, hanya menancapkan adaptor pada stopkontak.	Dipertahankan

3	Kenyamanan	Penempatan prototype yang tidak mengganggu kenyamanan dalam penggunaannya, yaitu cukup diletakkan di ruangan.	Dirubah posisi peletakannya, tetap di ruangan namun berada di sudut.
4	Keamanan	Penggunaan <i>prototype</i> ini tidak memiliki efek yang membahayakan bagi pengguna, karena tidak memerlukan kontak fisik dengan <i>prototype</i> tersebut.	Dipertahankan

### 5.3 Dampak Implementasi Sistem

#### 5.3.1 Teknologi/Inovasi

Sistem yang dibuat memiliki beberapa keunggulan dari sistem yang pernah dibuat sebelumnya. Contohnya adalah sistem yang dibuat oleh Wahyuni, dkk dimana penulis menggunakan metode *video processing* dan belum *real time*. Kemudian sistem yang dibuat oleh Ali Hashim, dkk dimana penulis menggunakan sensor untuk mendeteksi jatuh pada seseorang yang memiliki ketahanan sistem yang rendah karena bisa saja rusak terkena air ataupun kabelnya putus ketika dipakai oleh pengguna. Yang terakhir adalah sistem yang dibuat oleh Suad Albawendi, dkk dimana penulis menggunakan metode *video processing* dan belum *real time*. Berikut perbandingan yang ditunjukkan pada Tabel 5.3.

Tabel 5.3 Perbandingan Inovasi dari Peneliti Terdahulu

No	Fitur/Komponen	Sistem yang dibuat	Sistem A (Wahyuni, dkk)	Sistem B (Hasim, dkk)	Sistem C (Albawendi, dkk)
1	Cara Kerja Sistem	Real time	Tidak real time	Real time	Tidak real time
2	Ketahanan Sistem	Tinggi	Tinggi	Rendah	Tinggi

#### 5.3.2 Sosial

Dengan adanya produk ini dapat membantu keluarga ketika memiliki anggota keluarga yang sudah lansia di rumahnya. Hal ini dikarenakan produk dapat beroperasi sebagai penambah penjagaan terhadap lansia. Produk ini menggunakan *video processing* yang nantinya akan mengirim pesan kepada anggota keluarga lainnya melalui telegram ketika ada lansia yang terjatuh. Sehingga anggota keluarga dapat bertindak lebih cepat saat terjadinya hal tersebut.



## BAB 6 : Kesimpulan dan Saran

### 6.1 Kesimpulan

Berdasarkan percobaan yang telah dilakukan, maka didapatkan kesimpulan berikut :

1. Sistem pendeteksi jatuh dipengaruhi oleh beberapa kondisi yang menyebabkan performa algoritma menjadi terganggu, yaitu :
  - Jarak antara kamera dengan objek terlalu jauh ataupun terlalu dekat.
  - Posisi jatuh sejajar dengan sudut kamera.
  - Keberadaan objek lain yang menghalangi objek yang akan dideteksi.
  - Intensitas cahaya ruangan.
  - Jumlah objek dalam ruangan.
2. Sistem pendeteksi jatuh memiliki performa yang baik ketika kondisi berikut :
  - Perbedaan pergerakan jatuh objek.
  - Perbedaan warna antara objek dengan background yang kontras.
3. Sistem masih belum bisa membedakan objek yang terjatuh manusia atau bukan.
4. Sistem memiliki delay saat mengirim *alarm* minimal 4 detik.

### 6.2 Saran

Berdasarkan hasil pelaksanaan tugas akhir, ada beberapa saran untuk penelitian kedepannya agar performa dari sistem menjadi lebih baik lagi, yaitu :

1. Algoritma yang digunakan masih memiliki akurasi yang rendah terhadap intensitas cahaya seperti mendeteksi jatuh saat gelap dan belum bisa membedakan objek yang terjatuh manusia atau bukan. Penelitian selanjutnya diharapkan dapat memperbaiki ini.
2. Produk yang dibuat menggunakan *web camera* yang memiliki kelemahan kabel tidak Panjang sehingga harus dekat dengan perangkat Raspberry Pi, diharapkan peneliti selanjutnya menggunakan *wireless camera* CCTV untuk menunjang keefektifitasan produk.
3. Algoritma yang digunakan masih memiliki delay saat mengirim pesan ke pengguna sehingga membuat video pada sistem menjadi patah patah, diharapkan peneliti selanjutnya dapat mengatasi masalah ini.

## Daftar Pustaka

- [1] Badan Pusat Statistik, “Statistik Penduduk Usia Lanjut 2019,” *Badan Pus. Stat.*, p. 4, 2019.
- [2] H. Ali Hashim, S. L. Mohammed, and S. K. Gharghan, “Accurate fall detection for patients with Parkinson’s disease based on a data event algorithm and wireless sensor nodes,” *Meas. J. Int. Meas. Confed.*, vol. 156, p. 107573, 2020, doi: 10.1016/j.measurement.2020.107573.
- [3] V. A. Goodwin, S. H. Richards, W. Henley, P. Ewings, A. H. Taylor, and J. L. Campbell, “An exercise intervention to prevent falls in people with Parkinson’s disease: A pragmatic randomised controlled trial,” *J. Neurol. Neurosurg. Psychiatry*, vol. 82, no. 11, pp. 1232–1238, 2011, doi: 10.1136/jnnp-2011-300919.
- [4] D. Yacchirema, J. S. De Puga, C. Palau, and M. Esteve, “Fall detection system for elderly people using IoT and Big Data,” *Procedia Comput. Sci.*, vol. 130, pp. 603–610, 2018, doi: 10.1016/j.procs.2018.04.110.
- [5] E. S. Wahyuni, M. B. Frasetyo, and H. Setiawan, “Combination of motion history image and approximated ellipse method for human fall detection system,” *Int. J. Simul. Syst. Sci. Technol.*, vol. 19, no. 3, pp. 13.1-13.6, 2018, doi: 10.5013/IJSSST.a.19.03.13.
- [6] S. Albawendi, K. Appiah, H. Powell, and A. Lotfi, “Video based fall detection with enhanced motion history images,” *ACM Int. Conf. Proceeding Ser.*, vol. 29-June-20, 2016, doi: 10.1145/2910674.2935832.
- [7] M. Kreković *et al.*, “A method for real-time detection of human fall from video,” *MIPRO 2012 - 35th Int. Conv. Inf. Commun. Technol. Electron. Microelectron. - Proc.*, no. September, pp. 1709–1712, 2012.
- [8] J. S. Madhubala and A. Umamakeswari, “A Vision based Fall Detection System for Elderly People,” *Indian J. Sci. Technol.*, vol. 8, no. 9, pp. 167–175, 2015, doi: 10.17485/ijst/2015/v8iS9/65545.
- [9] B. S. Kumar and C. Engineering, “Real Time Camera based Human Fall Detection Using Raspberry Pi,” no. May 2019.



# LAMPIRAN – LAMPIRAN

## 1. Logbook Kegiatan TA2



DEPARTMENT OF  
ELECTRICAL ENGINEERING

### LOGBOOK KEGIATAN *CAPSTONE PROJECT*

Judul Proyek : Sistem Deteksi Jatuh pada Manusia Secara Real Time Menggunakan Video Processing

Pengusul : Muhammad Irfan Nugraha (17524014), Muhammad Yusuf Kurniawan (17524064)

Hari, Tanggal	Deskripsi Kegiatan
5 Maret 2021	Pencarian Raspberry untuk dipinjam
15 Maret 2021	<ul style="list-style-type: none"><li>• Bimbingan dengan dosen pembimbing Bu Elvira</li><li>• Pembelian Raspberry serta alat yang dibutuhkan lainnya</li></ul>
20 Maret 2021	Instalasi <i>library</i> OpenCV pada Raspberry Pi 3
22 Maret 2021	Instalasi <i>library</i> OpenCV pada Raspberry Pi 3
24 Maret 2021	Instalasi <i>library</i> OpenCV pada Raspberry Pi 3
25 Maret 2021	Instalasi <i>library</i> OpenCV pada Raspberry Pi 3
27 Maret 2021	Instalasi <i>library</i> OpenCV pada Raspberry Pi 3
28 Maret 2021	Instalasi <i>library</i> lainnya yang dibutuhkan pada Raspberry Pi 3
29 Maret 2021	Implementasi program deteksi gerak jatuh pada manusia menggunakan Raspberry Pi 3
30 Maret 2021	<ul style="list-style-type: none"><li>• Implementasi program deteksi gerak jatuh pada manusia menggunakan Raspberry Pi 3 (Pengaktifan Kamera dan mendeteksi objek)</li><li>• Mengerjakan <i>technical report</i> tugas akhir 201</li></ul>
31 Maret 2021	<ul style="list-style-type: none"><li>• Implementasi program deteksi gerak jatuh pada manusia menggunakan Raspberry Pi 3 ( Uji coba menggunakan video untuk mendeteksi jatuh)</li><li>• Mengerjakan <i>technical report</i> tugas akhir 201</li></ul>
1 April 2021	<ul style="list-style-type: none"><li>• Implementasi program deteksi gerak jatuh pada manusia menggunakan Raspberry Pi 3 (koneksi internet untuk mengirim <i>alarm</i>)</li><li>• Mengerjakan <i>technical report</i> tugas akhir 201</li></ul>
2 April 2021	<ul style="list-style-type: none"><li>• Implementasi program deteksi gerak jatuh pada manusia menggunakan Raspberry Pi 3 (koneksi internet untuk mengirim <i>alarm</i>)</li><li>• Mengerjakan <i>technical report</i> tugas akhir 201</li></ul>

<b>6 April 2021</b>	Bimbingan dengan dosen pembimbing (Bu Elvira) <ul style="list-style-type: none"> <li>• Laporan Progres</li> <li>• Sosialisasi <i>timeline</i> Tugas Akhir oleh dosen pembimbing</li> </ul>
<b>8 April 2021</b>	Implementasi program deteksi gerak jatuh pada manusia menggunakan Raspberry Pi 3 (Uji coba pengaruh peletakan posisi kamera)
<b>9 April 2021</b>	Implementasi program deteksi gerak jatuh pada manusia menggunakan Raspberry Pi 3 (Uji coba pada luas daerah cakupan)
<b>10 April 2021</b>	Implementasi program deteksi gerak jatuh pada manusia menggunakan Raspberry Pi 3 (Uji coba pada perbedaan jatuh objek)
<b>11 April 2021</b>	Implementasi program deteksi gerak jatuh pada manusia menggunakan Raspberry Pi 3 (Uji coba pada objek dan jumlah yang berbeda)
<b>13 April 2021</b>	Bimbingan dengan dosen pembimbing (Bu Elvira) via zoom
<b>14 April 2021</b>	implementasi program deteksi gerak jatuh pada manusia menggunakan Raspberry Pi 3 (Uji coba pada perbedaan kondisi (gelap dan terang) dan pengaruh benda lain(bayangan))
<b>15 April 2021</b>	Bimbingan dengan dosen pembimbing (Bu Elvira) <ul style="list-style-type: none"> <li>• Laporan progres</li> </ul>
<b>18 April 2021</b>	Mengerjakan <i>technical report</i> tugas akhir 202
<b>19 April 2021</b>	<ul style="list-style-type: none"> <li>• Pengambilan data hasil uji coba</li> <li>• Mengerjakan <i>technical report</i> tugas akhir 202</li> </ul>
<b>21 April 2021</b>	<ul style="list-style-type: none"> <li>• Mengerjakan <i>technical report</i> tugas akhir 202</li> <li>• Mengerjakan laporan tugas akhir</li> </ul>
<b>22 April 2021</b>	Bimbingan dengan dosen pembimbing (Bu Elvira) <ul style="list-style-type: none"> <li>• Laporan progres</li> <li>• Sosialisasi Laporan,TA202, Video dan Poster</li> <li>• Mengerjakan laporan tugas akhir</li> </ul>
<b>23 April 2021</b>	Mengerjakan laporan tugas akhir
<b>24 April 2021</b>	Mengerjakan laporan tugas akhir

Yogyakarta, 24 April 2021

Dosen Pembimbing



Elvira Sukma Wahyuni, S.Pd., M.Eng.

155231301

## 2. Kode Program

```
import datetime
from cv2 import cv2
import time
import numpy as np
import telepot
from telepot.loop import MessageLoop

def action(msg):
    global chat_id

    chat_id = msg['chat']['id']
    command = msg['text']

    print ('Got command: %s' % command)

    if command == '/start':
        bot.sendMessage(chat_id, 'Selamat Datang di Tepi Jaman')

    else:
        bot.sendMessage(chat_id, "sorry, I don't know the command "+command)

bot = telepot.Bot('1748960930:AAFgCn4nmBaGlgRsjYadxRMhFBSPZdxTvg4')
MessageLoop(bot, action).run_as_thread()
print ('I am listening ...')

slowDown = 0
allSlowDown = 0

camera = cv2.VideoCapture(0)

fgbg = cv2.createBackgroundSubtractorKNN()

cX = 0
cY = 0

fall_frames = 0
fall_count = 0
fall_check = True

MHI_DURATION = 30
DEFAULT_THRESHOLD = 32
ret, frames = camera.read()
mhi_h, mhi_w = frames.shape[:2]
prev_frame = frames.copy()
motion_history = np.zeros((mhi_h, mhi_w), np.float32)
timestamp = 0

first_frame = None
status = "Empty"
run = True

while(run):
    ret, frames = camera.read()

    try:
        status = "Empty"
        frame_diff = cv2.absdiff(frames, prev_frame)
        gray_diff = cv2.cvtColor(frame_diff, cv2.COLOR_BGR2GRAY)
        ret, fgmask = cv2.threshold(gray_diff, DEFAULT_THRESHOLD, 1, cv2.THRESH_BINARY)
        timestamp += 1
```

```

prev_frame = frames.copy()

cv2.motempl.updateMotionHistory(fgmask, motion_history, timestamp, MHI_DURATION)
mh = np.uint8(np.clip((motion_history - (timestamp - MHI_DURATION)) / MHI_DURATION, 0, 1)
* 255)

gray = cv2.cvtColor(frames, cv2.COLOR_BGR2GRAY)
gray = cv2.blur(gray,(5,5))

if first_frame is None:
    first_frame = gray

frame_delta = cv2.absdiff(first_frame, gray)
print(type(frame_delta),type(gray))
edges = cv2.Canny(gray,75,75)

fgmask = fgbg.apply(gray)
fgmask = cv2.erode(fgmask,(3,3))
fgmask = cv2.dilate(fgmask,(5,5))

contours, _ = cv2.findContours(fgmask, cv2.RETR_TREE, cv2.CHAIN_APPROX_SIMPLE)
edge_contours, _ = cv2.findContours(edges, cv2.RETR_TREE, cv2.CHAIN_APPROX_SIMPLE)

if contours:
    areas = []
    edge_areas = []
    for contour in contours:
        ar = cv2.contourArea(contour)
        areas.append(ar)

    max_area = max(areas, default = 1)
    max_area_index = areas.index(max_area)

    for contour in edge_contours:
        ar = cv2.contourArea(contour)
        edge_areas.append(ar)

    edge_max_area = max(edge_areas, default = 1)
    edge_max_area_index = edge_areas.index(edge_max_area)

    cnt = contours[max_area_index]
    x, y, w, h = cv2.boundingRect(cnt)
    ed_cnt = edge_contours[edge_max_area_index]
    if edge_max_area > 3:
        cv2.drawContours(frames,ed_cnt,-1,(0,255,0),3)

if max_area > 1000:
    if w > h:
        fall_frames += 1

    if fall_frames > 16:
        if not fall_check:
            fall_count += 1
            file = open('FallTimmings.txt','a')
            file.write("{}\n".format(datetime.datetime.now().strftime("%A %d %B %Y
%i:%M:%S%p")))
            file.close()
            file = open('Fall.txt','w')
            file.write("Fall: {}\n".format(fall_count))
            file.close()
            fall_check = True
            status = "Fall"

```

```

cv2.rectangle(frames,(x,y),(x+w,y+h),(0,0,255),2)
#camera.capture('./capture.png')
#bot.sendPhoto(chat_id=chat_id, photo=open('./capture.png', 'rb'))
filename = "jatuh.png".format(time.strftime("%Y%m%d_%H%M%S"))
cv2.imwrite(filename, frames)
#bot.sendPhoto(chat_id, photo=open(filename + '.png', 'rb'))
bot.sendPhoto(chat_id, photo=open('/home/pi/Coba Fall/jatuh.png', 'rb'))
bot.sendMessage(chat_id, 'fall')

else:
    cv2.rectangle(frames,(x,y),(x+w,y+h),(0,255,0),2)

if h >= w:
    status = "Occupied"
    fall_frames = 0
    fall_check = False
    cv2.rectangle(frames,(x,y),(x+w,y+h),(0,255,0),2)

    cv2.putText(frames, "Room Status: {0}".format(status), (10, 20),cv2.FONT_HERSHEY_SIMPLEX,
0.5, (0, 0, 255), 2)
    cv2.putText(frames, datetime.datetime.now().strftime("%A %d %B %Y %I:%M:%S%p"),(10,
frames.shape[0] - 10), cv2.FONT_HERSHEY_SIMPLEX, 0.4, (0, 0, 255), 1)

    cv2.imshow('video', frames)
    cv2.imshow('FGmask', fgmask)
    cv2.imshow('Edges', edges)
    cv2.imshow('motion-history', mh)
    time.sleep(allSlowDown)

    if cv2.waitKey(1) & 0xFF == ord('q'):
        run = False
        break
except Exception as e:
    break

cv2.destroyAllWindows()

```

### 3. Laporan Keuangan

**Anggaran Pembelian BHP**  
**Tugas Akhir 2**  
**Semester Genap 2020/2021**  
**Program Studi Teknik Elektro FTI UII**

Kelompok : 20

- 1) Muhammad Irfan Nugraha (17524014)
- 2) Muhammad Yusuf Kurniawan (17524064)

Judul Tugas Akhir : Sistem Deteksi Jatuh pada Manusia Secara Real Time Menggunakan Video Processing

Pembimbing : 1) Elvira Sukma Wahyuni, S.Pd., M.Eng.  
2) Yusuf Aziz Amrulloh, Ph.D.

No	Tanggal pembelian	Keterangan	Jumlah	Harga Satuan	Subtotal	Bukti/No. nota
1	15 Maret 2021	Paket Lengkap Raspberry Pi 3b dan kamera raspberry	1	Rp 905,000	Rp 908,700	1
2	03 April 2021	Case Raspberry Pi 3	1	Rp 39,500	Rp 39,500	2
3	22 April 2021	Tripod	1	Rp 50,000	Rp 50,000	3
4	22 April 2021	Tongsis	1	Rp 55,000	Rp 55,000	3
Total					Rp 1,053,200	