

KLASIFIKASI SAMPAH ORGANIK DAN NON ORGANIK DENGAN METODE
SSD-MOBILENET

SKRIPSI

DICKY ARWANDA PUTRA MELIALA

171402131



PROGRAM STUDI S1 TEKNOLOGI INFORMASI
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS SUMATERA UTARA
MEDAN
2024

KLASIFIKASI SAMPAH ORGANIK DAN NON ORGANIK DENGAN METODE
SSD-MOBILENET

SKRIPSI

Diajukan untuk melengkapi tugas dan memenuhi syarat memperoleh ijazah Sarjana
Teknologi Informasi

DICKY ARWANDA PUTRA MELIALA

171402131



PROGRAM STUDI S1 TEKNOLOGI INFORMASI
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS SUMATERA UTARA

MEDAN

2024

PERSETUJUAN

Judul : Klasifikasi Sampah Organik Dan Non Organik Dengan
Metode Ssd-Mobilenet
Kategori : *Computer Vision*
Nama : Dicky Arwanda Putra Meliala
Nomor Induk Mahasiswa : 171402131
Program Studi : Teknologi Informasi
Fakultas : Ilmu Komputer dan Teknologi Informasi
Universitas Sumatera Utara
Komisi Pembimbing :
Medan, x x x

Pembimbing 2

Pembimbing 1

Fahrurrozi Lubis, B.IT., M.Sc.IT.
NIP. 198610122018052001

Baihaqi Siregar S.Si., M.T.
NIP. 197901082012121002

Diketahui/disetujui oleh
Program Studi Teknologi Informasi
Ketua,

Dedy Arisandi S.T., M.Kom.
NIP. 197908312009121002

PERNYATAAN

KLASIFIKASI SAMPAH ORGANIK DAN NON ORGANIK DENGAN METODE SSD-MOBILENET

SKRIPSI

Saya mengakui bahwa skripsi ini adalah hasil karya saya sendiri, kecuali beberapa kutipan dan ringkasan yang masing – masing telah disebutkan sumbernya

Medan, 12 Desember 2023

Dicky Arwanda Putra Meliala

171402131

UCAPAN TERIMA KASIH

Puji dan syukur kepada Tuhan Yang Maha Esa, atas berkat dan kasih-Nya penulis dapat menyelesaikan skripsi ini sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer pada Program Studi S1 Teknologi Informasi Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Sumatera Utara.

Skripsi ini ditulis dengan doa dan dukungan baik secara moril maupun materil dari keluarga penulis Bapak Tenang Ukur Sembirng dan Ibu Eli Erna S.Kep Ners, serta saudara penulis Agnes Thandania Blessky Meliala. Rasa terima kasih dan bersyukur penulis ucapkan kepada seluruh pihak yang turut serta terlibat dalam masa perkuliahan dan masa pengerjaan skripsi ini:

1. Bapak Dr. Muryanto Amin S.Sos., M.Si. selaku Rektor Universitas Sumatera Utara
2. Ibu Dr. Maya Silvi Lydia, B.Sc., M.Sc., selaku Dekan Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Sumatera Utara.
3. Ibu Sarah Purnamawati, ST., M.Sc., selaku Wakil Dekan Program Studi S1 Teknologi Informasi Universitas Sumatera Utara.
4. Bapak Dedy Arisandi, S.T., M.Kom., selaku Ketua Program Studi S1 Teknologi Informasi Universitas Sumatera Utara.
5. Bapak Fahrurrozi Lubis, B.IT., M.Sc.IT., selaku Dosen Pembimbing 1 yang telah mengarahkan dan berbagi ilmu kepada penulis selama proses pengerjaan skripsi hingga selesai.
6. Bapak Baihaqi Siregar, S.Si., MT., selaku Dosen Pembimbing 2 yang telah mengarahkan dan berbagi ilmu kepada penulis selama proses pengerjaan skripsi hingga selesai.
7. Para dosen dan staff akademik yang telah memfasilitasi segala aspek yang dapat menunjang proses perkuliahan penulis sampai dengan penulisan skripsi ini.
8. Teman-teman unit kegiatan mahasiswa GULLY Joshua Andrew, Syarfan Hasriansyah, Riyo Santo Yoseph, Maulana Pramisyah Ramadhan, Alfi Rayhananda, Frans samosir, Yusman Tri Klavier, Yonadab Laoli, Jonathan Simanjuntak, Moris Sormin, Frederiko, Aldo ndun, Mhd Irfan Fajar yang telah menjadi teman seperjuangan selama proses perkuliahan dan proses pembuatan skripsi.

9. Kepada senior yang telah membantu dan mengarahkan penulis dalam masa perkuliahan hingga tugas akhir.
10. Seluruh teman Angkatan 2017 terutama KOM B Teknologi Informasi 2017 yang sudah bersama dengan penulis dari awal perkuliahan, saling memotivasi serta saling berbagi ilmu selama perkuliahan.
11. Keluarga, kerabat dan saudara yang selalu mendukung dan mendoakan penulis.

Sekian ucapan dan rasa terima kasih yang dapat disampaikan semoga Tuhan Yang Maha Esa memberkahi apa-apa saja yang peneliti kerjakan dan diberikan kemudahan atas apa yang peneliti rencanakan.

Medan, 2024

Penulis

ABSTRAK

Salah satu tantangan utama dalam pengelolaan sampah adalah klasifikasi yang akurat antara sampah organik dan non-organik, yang merupakan langkah awal penting dalam proses daur ulang dan pengelolaan yang berkelanjutan agar tidak menyebabkan dampak negatif pada lingkungan tempat tinggal masyarakat. Penelitian ini bertujuan untuk mengembangkan dan menguji sebuah sistem klasifikasi otomatis yang dapat mengidentifikasi dengan tepat sampah organik dan non-organik. Sistem klasifikasi ini dikembangkan menggunakan algoritma pembelajaran mesin MobileNet yang merupakan suatu metode yang dirancang untuk memberikan akurasi yang lebih maksimal berdasarkan sumber daya yang diberikan dan dilakukan dengan cara efektif dan tentunya memiliki latensi yang rendah dan juga penggunaan daya ataupun source yang kecil sehingga lebih cepat dan ringan. Hasil penelitian menunjukkan bahwa sistem klasifikasi otomatis yang dibangun dapat mengenali dengan tingkat akurasi yang baik antara sampah organik dan non-organik. Hasil ini berpotensi untuk meningkatkan efisiensi dalam pengelolaan sampah, memungkinkan proses daur ulang yang lebih efektif, serta mengurangi dampak negatif lingkungan.

Kata Kunci : Sampah Anorganik, Sampah Organik, SSD-MOBILENET, IOT

CLASSIFICATION ORGANIC WASTE AND INORGANIC WASTE WITH SSD-MOBILENET METHOD

ABSTRACT

One of the main challenges in waste management is the accurate classification between organic and non-organic waste, which is an important first step in the recycling process and sustainable management so as not to cause negative impacts on the environment where people live. This research aims to develop and test an automatic classification system that can correctly identify organic and non-organic waste. This classification system was developed using the MobileNet machine learning algorithm which is a method designed to provide maximum accuracy based on the resources provided and is carried out in an effective way and of course has low latency and also uses small power or resources so it is faster and lighter. . The research results show that the automatic classification system that was built can recognize with a good level of accuracy between organic and non-organic waste. These results have the potential to increase efficiency in waste management, enable more effective recycling processes, and reduce negative environmental impacts.

Keywords: Inorganic Waste, Organic Waste, SSD-MOBILENET, IOT

DAFTAR ISI

PERSETUJUAN	i
PERNYATAAN	ii
UCAPAN TERIMA KASIH	iii
ABSTRAK	v
<i>ABSTRACT</i>	vi
DAFTAR ISI	vii
DAFTAR TABEL	x
DAFTAR GAMBAR	xi
BAB 1 PENDAHULUAN	13
1.1. Latar Belakang	13
1.2. Rumusan Masalah	15
1.3. Batasan Masalah	15
1.4. Tujuan Penelitian	16
1.5. Manfaat Penelitian	16
1.6. Metodologi Penelitian	16
1.7. Sistematika Penulisan	17
BAB 2 LANDASAN TEORI	19
2.1. Sampah	19
2.2. Citra	20
2.2.1. Citra RGB	23
2.2.2. Citra Biner	23
2.2.3. Citra Intensitas Keabuan	23
2.2.4. Format Citra Digital	24
2.3. <i>SSD Mobilenet V2</i>	26

2.4.	Mean Squared Error (MSE) dan Peak Signal to Noise Ratio (PSNR)	29
2.5.	<i>Raspberry Pi 3</i>	30
2.6.	Webcam Raspi	31
2.7.	ESP32	31
2.8.	Protokol MQTT	32
2.9.	Motor Servo	32
2.10.	Adaptor 12V	33
2.11.	Dinamo DC 5V	34
2.12.	Mengukur Kinerja Metode dengan <i>Confusion Matrix</i>	34
2.13.	Parameter Pengujian Sistem	37
2.14.	Penelitian Terkait	37
BAB 3 ANALISIS DAN PERANCANGAN		40
3.1.	Pengumpulan Data	40
3.2.	Arsitektur Umum	42
3.3.	Persiapan Training Data	43
3.4.	Instalasi Tensorflow	45
3.5.	Analisis Gambar dan Tabel	45
3.6.	Konversi Data XML ke CSV	46
3.7.	Generate File Record	48
3.8.	Training Data	49
3.9.	Testing Detection	53
BAB 4 IMPLEMENTASI DAN HASIL PENELITIAN		54
4.1.	Spesifikasi Kebutuhan Aplikasi	54
4.1.1.	Kebutuhan Perangkat Keras	54
4.1.2.	Kebutuhan Perangkat Lunak	54
4.2.	Penggunaan Data	55
4.3.	Pengujian	56

4.4. Hasil Pengujian	57
BAB 5 KESIMPULAN DAN SARAN	62
5.1. Kesimpulan	62
5.2. Saran	62
DAFTAR PUSTAKA	63

DAFTAR TABEL

Tabel 2.1 Jenis Model Warna	22
Tabel 2.2 Bitmap Info Header	24
Tabel 2.3 Bitmap Core Header	25
Tabel 2.4 Ukuran Parameter dan Komputasi pada Depthwise Separable Convolution	28
Tabel 2.5 Klasifikasi <i>confusion matrix</i>	36
Tabel 2.6 Penelitian Terdahulu	37
Tabel 3.1 Isi Dataset	40
Tabel 4.1 Sampel pengujian	58

DAFTAR GAMBAR

Gambar 2.1 Sistem Koordinat yang Dipergunakan untuk Mewakili Citra	20
Gambar 2.2 Proses Sampling dan Kuantisasi	21
Gambar 2.3 Nilai warna RGB pada Heksadesimal	22
Gambar 2.4 Citra RGB`	23
Gambar 2.5 Citra Grayscale	24
Gambar 2.6 Alur kerja MobileNetV2 (Sik-Ho Tsang, 2019)	27
Gambar 2.7 Konvolusi Depthwise (2-5) dan Pointwise (6-7)	28
Gambar 2.8 Raspberry Pi 3	30
Gambar 2.9 Raspi Webcam	31
Gambar 2.10 ESP32	31
Gambar 2.11 Alur Desain MQTT	32
Gambar 2.12 Motor Servo	33
Gambar 2.13 Adaptor	34
Gambar 2.14 Dinamo DC	34
Gambar 3.1 Arsitektur umum	43
Gambar 3.2 Labelling dengan Labeling	44
Gambar 3.3 Proses Instalasi Tensorflow	45
Gambar 3.4 Proses Analisis Citra	45
Gambar 3.5 Proses Analisis Label	46
Gambar 3.6 Proses Konversi XML ke CSV dan Hasilnya	47
Gambar 3.7 File Record	48
Gambar 3.8 Isi folder SSD MobileNet V2	49
Gambar 3.9 Konfigurasi Pipeline	50
Gambar 3.9 Konfigurasi Pipeline bagian 2	51

Gambar 3.10 Proses Training	52
Gambar 3.11 Proses testing	53
Gambar 4.1 Sampah Anorganik	55
Gambar 4.2 Sampah Organik	55
Gambar 4.3 Proses menjalankan program	56
Gambar 4.4 Tampilan Aplikasi	56
Gambar 4.5 Peralatan Pengujian	57
Gambar 4.6 Pengujian pada Sampah Organik	57
Gambar 4.7 Pengujian pada Sampah Anorganik	57
Gambar 4.8 Nilai Akurasi yang Didapatkan	61
Gambar 4.9 Nilai <i>F1 - Score</i> yang Didapatkan	61

BAB 1

PENDAHULUAN

1.1. Latar Belakang

Sampah saat ini menjadi masalah besar dimana - mana, kerugian besar selalu ditimbulkan dari sampah - sampah yang tidak benar pengelolaannya sehingga menimbulkan banyak sekali kerugian, seperti banjir, penyakit - penyakit kulit, dan tentunya pencemaran lingkungan yang diakibatkan. Salah satu jenis sampah yang sangat lama dalam proses urai ataupun penguraiannya disebut dengan sampah anorganik. Di dalam sampah jenis ini sangat banyak zat kimia yang sangat sulit ataupun membutuhkan proses yang sangat lama dalam penguraiannya. Contoh sampah anorganik ini pastinya sangat sering peneliti temui dan sangat familiar, seperti plastik, logam, kaca, kertas yang dimana membutuhkan waktu yang lama agar bisa terurai. Pemanfaatan daur ulang yang dilakukan untuk menanggulangi sampah-sampah ini tentunya sangat memudahkan apabila dipilah berdasarkan jenisnya.

Sampah diambil dari rumah ke rumah dengan menggunakan pengangkut sampah dan akan dikirim ke TPA sebagai tempat akhir pembuangan sampah masing- masing area. Sampah-sampah tersebut masih tercampur antara organik dan Anorganik, sehingga sampah proses penguraian sampah menjadi kurang maksimal dan memerlukan waktu yang lama. Sedangkan sampah organik akan tinggal di tanah yang menimbulkan pencemaran tanah.

Agar sampah dapat dikelola dengan baik, maka diperlukan pemisahan pada saat pembuangan ke tempat-tempat akhir setiap pembuangan. Karena banyaknya yang dikelola jumlahnya besar antara 10 sampai dengan 50 ton per hari, maka diperlukan teknik klasifikasi sampah agar pemisahan jenis sampah di atas dapat dilakukan dengan cepat dan efisien. Saat ini tentunya jika peneliti mengandalkan tenaga manual ataupun dikerjakan dengan cara memilah satu per satu sampah, tentunya akan memakan waktu yang sangat lama dan kurang efisien.

Aplikasi yang tentunya sesuai adalah aplikasi yang mampu untuk mengidentifikasi jenis-jenis sampah yang diberikan, dalam hal ini sebagai citra maupun gambar dengan menganalisisnya berdasarkan ciri maupun fitur dari citra tersebut. Salah satu bentuk identifikasi yaitu dengan memanfaatkan metode MobileNet yang menyediakan fitur-fitur tersebut.

MobileNet adalah suatu metode yang dirancang untuk memberikan akurasi yang lebih maksimal berdasarkan sumber daya yang diberikan dan dilakukan dengan cara efektif dan tentunya memiliki latensi yang rendah dan juga penggunaan daya ataupun source yang kecil sehingga lebih cepat dan ringan. Metode ini adalah pengembangan model dengan cara open source dan merupakan bagian dari CNN yang dapat peneliti gunakan pada platform mobile. MobileNet sendiri tentunya dapat peneliti gunakan untuk melakukan beberapa kinerja seperti deteksi, segmentasi, ataupun klasifikasi, bahkan embedding berdasarkan materi uji yang peneliti lakukan untuk mendapat akurasi yang baik.

Pada penelitian yang dilakukan oleh (Shindy, 2019) membuat arsitektur jaringan yang mampu untuk mendeteksi objek berupa manusia secara langsung ataupun realtime dengan memanfaatkan sebuah teknik deep learning tentunya dengan MobileNet-SSD juga. Intensitas cahaya yang diuji antara 50-550 lux dengan pengujian jarak antara 1-10 meter dengan kondisi bervariasi dan akurasi 91,67%.

Sela et al. 2017 dalam penelitiannya yang berjudul Deteksi Kualitas Telur Menggunakan Analisis Tekstur. Dalam penelitian ini bahwa telur merupakan barang yang sangat baik buat anak-anak. Subjek pada penelitian ini adalah mendeteksi kualitas telur menggunakan analisis tekstur pada kulit telur.

Penelitian (Trisnaningtyas et al. 2015) dalam penelitiannya yang membuat pengklasifikasian kebersihan dari telur memanfaatkan algoritma ataupun metode K-Nearest Neighbor. Telur merupakan salah satu makanan yang menyehatkan dan memiliki kandungan gizi yang baik dan tentunya harus dijaga kebersihannya. Telur juga sumber protein dan tentunya telur juga sangat mudah dicerna dan disiapkan. Subjek utama penelitian ini adalah aplikasi deteksi maupun klasifikasi mutu telur yang baik maupun buruk untuk memilih telur yang aman untuk digunakan maupun dikonsumsi.

Ramadhan, 2020 melakukan pengujian mengenai pengaturan dari neural network pada sisi penggunaan client untuk mendapatkan learning rate yang optimal dan baik. Dalam pengujiannya didapatkan kesimpulan bahwa pada client side melakukan prosesi

training lebih cepat dibandingkan dengan melakukannya pada server side, namun memiliki loss function yang lebih baik dibandingkan client side.

Selanjutnya melalui (Pramana et al., 2021) melakukan penelitian mengenai deteksi hal-hal yang berbau sepak bola, baik seperti bola, halangan, posisi gawang, dan objek-objek yang ada disepenuhnya dengan bantuan robot menggunakan metode single shot multibox detector (SSD). Pada hasil uji yang diterima didapatkan bahwa mAP rata-rata yang diperoleh yaitu 0.8 total loss 1.5. Selain itu validasi optimal terhadap objek-objek yang ada mendapatkan keberhasilan prediksi dengan angka rata-rata 98,45% yang menandakan prediksi yang dibuat telah baik. Dalam penelitian ini penulis melakukan klasifikasi Sampah Organik Dan Non Organik Dengan Metode MobileNet-SSD.

1.2. Rumusan Masalah

Sampah yang diambil dari rumah ke rumah dengan menggunakan pengangkut sampah dan akan dikirim ke TPA (Tempat Pembuangan Akhir) sebagai tempat akhir pembuangan sampah. Sampah-sampah tersebut masih tercampur antara organik dan non organik, sehingga proses penguraian sampah menjadi kurang maksimal dan memerlukan waktu yang lama. Sedangkan sampah organik akan tinggal di tanah yang menimbulkan pencemaran tanah. Agar sampah dan jenisnya dapat dikelola dengan baik, maka diperlukan pemisahan pada saat pembuangan ke tempat-tempat akhir setiap pembuangan, karena sampah yang dikelola jumlahnya besar antara 10 sampai dengan 50 ton per hari, maka diperlukan sebuah sistem klasifikasi sampah agar pemisahan jenis sampah di atas dapat dilakukan dengan cepat sesuai dengan jenisnya.

1.3. Batasan Masalah

Batasan-batasan yang dibuat antara lain:

1. Aplikasi ini hanya melakukan klasifikasi jenis sampah organik dan anorganik.
2. Tidak bisa memisahkan sampah yang terlalu kecil
3. Tidak dapat memisahkan sampah yang sudah ditumpuk jadi 1 di dalam plastik.
4. Tidak bisa menggunakan sampah yang terlalu besar dan berat karena alat masih berupa prototype.
5. Ekstensi citra uji dan citra latih dalam bentuk format .jpg/.png.

1.4. Tujuan Penelitian

Tujuan dilakukannya penelitian ini adalah untuk melakukan klasifikasi jenis sampah organik dan non organik dengan menggunakan metode SSD-Mobilenet.

1.5. Manfaat Penelitian

Manfaat yang diperoleh dari penelitian ini:

1. Membantu pengguna dalam pemisahan jenis sampah Organik dan Anorganik.
2. Implementasi metode SSD-MOBILENET dalam mengklasifikasi jenis sampah Organik dan Anorganik.
3. Menjadi referensi dan bahan pembelajaran bagi penelitian lain yang menerapkan metode SSD-MOBILENET.

1.6. Metodologi Penelitian

Metodologi yang dibuat memiliki beberapa urutan langkah yaitu:

1. Studi Literatur

Proses studi literatur merupakan tahapan mencari sebanyak-banyaknya referensi yang dapat menunjang penelitian. Studi literatur pada penelitian ini adalah mencari informasi terkait klasifikasi sampah organik dan non organik serta metode ssd- mobilenet dari berbagai sumber seperti skripsi, artikel serta jurnal dari berbagai sumber.

2. Identifikasi Masalah

Proses identifikasi masalah adalah tahapan memahami dan menganalisis semua sumber-sumber yang sudah terkumpul pada proses sebelumnya.

3. Perancangan Sistem

Proses ini adalah proses membuat arsitektur umum dari penelitian, mengumpulkan data yang dipakai pada penelitian, menentukan jumlah data latih dan data uji, juga merancang tampilan antarmuka sesuai dengan analisis setiap permasalahan pada tahap identifikasi masalah. Pengumpulan data berupa informasi terkait sampah organik dan non organik di lingkungan tempat tinggal masyarakat.

4. Implementasi Sistem

Proses ini dilakukan pengimplementasikan hasil rancangan yang sudah dibuat. Mulai dari merancang arsitektur sistem dan merancang metode yang akan digunakan dalam menganalisis masalah-masalah yang sudah ditentukan.

5. Pengujian Sistem

Proses ini melakukan beberapa pengujian berdasarkan kemampuan dari sistem yang sudah selesai dengan tujuan untuk memastikan kinerja sistem yang dibangun sudah mampu dimanfaatkan untuk mengklasifikasikan sampah organik dan sampah non organik.

6. Dokumentasi dan Penyusunan laporan

Proses laporan adalah kegiatan mengumpulkan dokumentasi berupa gambar pada masa penelitian serta menyusun laporan hasil analisis dan perancangan sistem terkait kemampuan metode SSD-MobileNet dalam melakukan klasifikasi sampah organik dan sampah non organik.

1.7. Sistematika Penulisan

Sistematika dalam penulisan skripsi ini akan dibagi menjadi beberapa bab sebagai berikut:

BAB 1 : PENDAHULUAN

Pada bab ini diuraikan hal-hal yang merupakan latar belakang masalah, pembuatan rumusan masalah, pembatasan masalah yang ada, pembuatan tujuan dan manfaat dari penelitian, metodologi penelitian yang dilakukan serta sistematika penulisan.`

BAB 2 : LANDASAN TEORI

Landasan teori membahas isu penting terkait pada penelitian. Teori penting yang berkaitan dengan skripsi ini seperti sampah organik dan non organik, *single shot detector* (SSD), *MobileNet* dan beberapa penelitian yang dijadikan sebagai referensi.

BAB 3 : ANALISIS DAN PERANCANGAN

Analisis dan perancangan akan membahas mengenai arsitektur umum dari metode SSD-MobileNet yang akan digunakan untuk memproses citra sampah organik dan non organik sampai menghasilkan output hasil klasifikasi sampah organik dan non organik. Dalam pembahasan ini akan menampilkan wireframe tampilan yang dibangun.

BAB 4 : IMPLEMENTASI DAN PENGUJIAN

Implementasi dan pengujian sistem akan membahas terkait implementasi dari analisis rancangan yang telah dijelaskan dan dirancang pada pembahasan sebelumnya. Dalam bagian ini juga membahas terkait hasil pengujian sistem yang sudah dibuat apakah sudah optimal dalam memecahkan masalah yang ada.

BAB 5 : KESIMPULAN DAN SARAN

Kesimpulan dan saran akan membahas tentang hasil analisis dari penelitian yang dilakukan yang biasanya akan dijabarkan poin per poin. Disini juga akan membahas terkait saran yang bisa dilakukan sebagai acuan agar meningkatkan penelitian dengan objek serupa.

BAB 2

LANDASAN TEORI

2.1. Sampah

Sampah tentunya sudah tidak asing di telinga masyarakat dewasa ini. Penyebab utama banjir yang ada dan faktor utama penyumbatan di tiap selokan ataupun parit bahkan bisa menutup sungai. Pada kasus ini tentunya akan menyumbat pada bagian parit dan membuat area di pemukiman warga basah dan tergenang air tentunya juga rentan dengan adanya nyamuk dan sumber - sumber penyakit yang akan datang baik pada kulit maupun pernafasan. Pada umumnya ada dua jenis dari sampah yang ada yaitu sampah organik dan satu lagi adalah sampah anorganik. Sampah organik adalah sampah yang mudah terurai berasal dari hal-hal alami yang hanya membutuhkan 0-6 bulan untuk dihancurkan atau dilenyapkan oleh pengurai, sementara itu sampah anorganik merupakan sampah yang mengandung banyak sekali zat kimia ataupun zat lainnya yang membuatnya sangat sulit terurai bahkan bisa memakan waktu puluhan tahun untuk mengurainya ataupun bahkan tidak bisa terurai sama sekali. Menurut data dari Badan Pusat Statistik pada tahun 2017, Indonesia menghasilkan sampah seberat 11,330 ton per hari dari jumlah total jiwa yang ada sepenelitian 128,5 juta jiwa dan banyaknya daerah perkotaan per orang mampu untuk menghasilkan 0.5kg sampah.

Sampah diambil dari rumah ke rumah dengan menggunakan pengangkut sampah dan akan dikirim ke TPA sebagai tempat akhir pembuangan sampah masing- masing area. Sampah - sampah tersebut masih tercampur antara organik dan organik, sehingga sampah proses penguraian sampah menjadi kurang maksimal dan memerlukan waktu yang lama. Sedangkan sampah organik akan tinggal di tanah yang menimbulkan pencemaran tanah. Agar sampah dan jenisnya dapat dikelola dengan baik, maka diperlukan pemisahan pada saat pembuangan ke tempat-tempat akhir setiap pembuangan.

Berdasarkan jenis sampah yang peneliti bahas di atas tentunya benda - benda ataupun bentuk dari sampah itu berbeda - beda. Pada sampah organik contohnya adalah sampah buah, benda - benda alami, sisa makanan, sampah hasil masakan, dedaunan, hewan - hewan yang mati. Sementara untuk sampah anorganik kaleng, botol plastik, kaca, produk sintetis, detergen, kantong plastik, keramik, logam, dan lain-lain. Tentunya membahas dari adanya sampah-sampah ini peneliti membutuhkan suatu teknologi terkini untuk menanggulangi ataupun mendaur ulang sampah - sampah yang ada menjadi benda - benda ataupun hal-hal yang berguna lainnya.

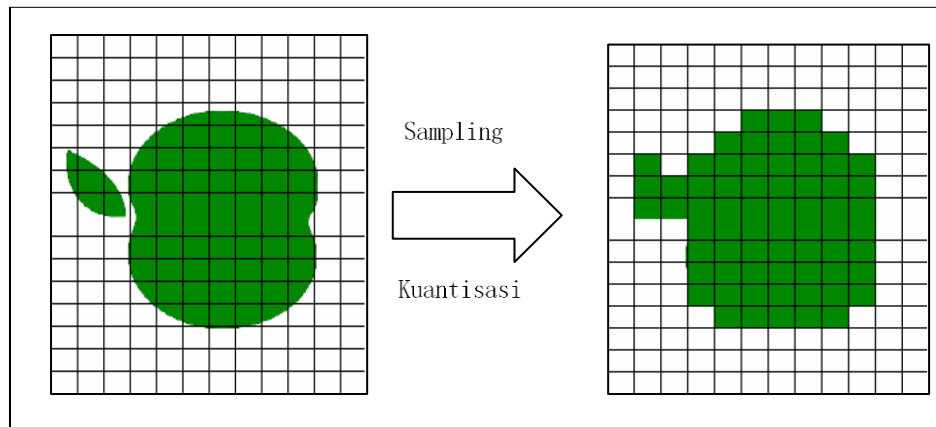


Gambar 2.1 Sistem Koordinat yang Dipergunakan untuk Mewakili Citra

2.2. Citra

Citra terbentuk berdasarkan ruang dua dimensi yang memiliki koordinat (x,y) di mana merupakan koordinat dan nilai intensitas cahaya yang juga memberikan informasi warna dan keceharan dari citra tersebut. Citra juga sering digunakan untuk menjadi data ataupun bahan untuk menjadi sampling pada penelitian (Gonzalez et al, 2002). Citra memiliki beberapa jenis yaitu citra dalam bentuk analog dan citra dalam bentuk digital. Citra analog dihasilkan dari intensitas yang berkelanjutan dan merupakan sinyal analog, sementara citra digital bersifat diskrit dan terbentuk dari sinyal digital yang dipancarkan. Pada umumnya kombinasi

warna yang ada terbagi menjadi tiga jenis yaitu RGB atau susunan warna yang biasa peneliti sesuaikan dalam mengedit maupun menyesuaikan warna. Dalam citra akan ada proses sampling untuk mendapatkan data dari citra yang peneliti input seperti pada Gambar 2.2.



Gambar 2.2 Proses Sampling dan Kuantisasi

Sampling memberikan informasi banyaknya blok *pixel* yang ada pada pendefinisian gambar, sementara kuantisasi pada gambar adalah banyaknya derajat nilai yang ada pada setiap *pixel* dan juga *pixel* akan menunjukkan jumlah banyaknya bit pada suatu citra apakah termasuk dalam *grayscale* ataupun *true color* yang masing-masing memiliki 8 dan 24 bit (Andono *et al.*, 2017).

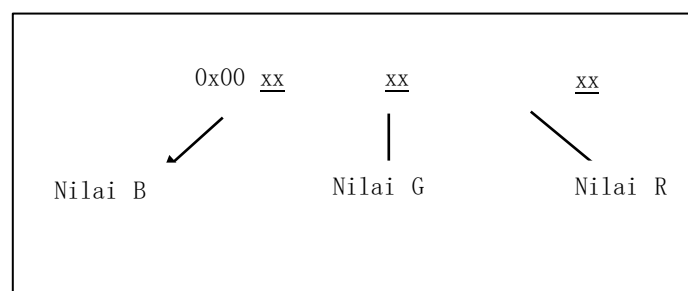
Warna merupakan representasi cahaya yang diterima oleh mata tepatnya pada bagian retina mata. 400-700nm adalah panjang gelombang yang mampu diterima oleh mata dan dapat di representasikan. Warna juga memiliki ruang untuk menilai komponen - komponen yang ada di dalamnya seperti, RGB, CMY, CIELAB, HSL, dan lain-lain.

Citra disusun oleh komponen titik-titik penyusun daripada citra itu sendiri yang biasa peneliti sebut dengan pixel. Dan banyaknya kumpulan dari pixel tersebut akan menjadi sebuah resolusi yang tentunya memiliki warna yang berbeda-beda. Berikut ini dipaparkan jenis model warna seperti pada Tabel 2.1:

Tabel 2.1 Jenis Model Warna

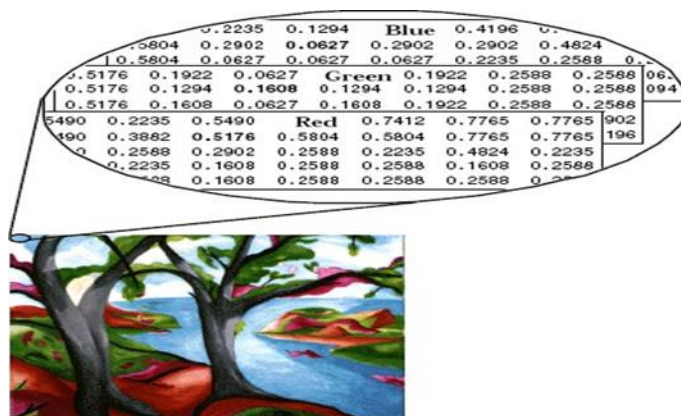
Jenis Mode Warna	Keterangan	Ukuran <i>bit depth</i>	Jumlah variasi warna
Grayscale	Warna abu-abu terdiri dari warna primer merah, hijau dan biru yang masing-masing memiliki nilai dasar yang sama. Misal = Merah = 67, Hijau = 67 dan Biru = 67. Dari nilai yang sama terdapat perbedaan warna abu-abu pada skala hitam putih	8 <i>bit depth</i>	$2^8 = 256$ variasi warna
Monokrom	Terdiri hanya hitam dan putih	1 bit	$2^1 = 2$ variasi warna
RGB	3 channel <i>Red, Green, Blue</i> yang masing – masing 8 <i>bit depth</i>	$8 \times 3 = 24$	$2^{24} = 16.777.216$ variasi warna
HSV	4 channel, <i>Cyan, Magenta, Yellow, Black</i> masing-masing 8 <i>bit depth</i>	$8 \times 4 = 32$	$2^{32} = 4.294.967.296$ variasi warna

Nilai heksadesimal direpresentasikan dari 0x00000000-0x00ffffff untuk pengolahan citra. Warna hitam 0x00000000 dan warna putih 0x00ffffff yang dimana variabel 0x00 menyatakan angka dibelakangnya adalah hexadecimal seperti pada penjelasan Gambar 2.3.

**Gambar 2.3** Nilai warna RGB pada Heksadesimal

2.2.1. Citra RGB

Gambar dengan citra RGB tentunya sudah sering dilihat dimana menggunakan tiga perhitungan lapisan warna. Matriks yang digunakan akan menghitung dan diwakili oleh warna biru, hijau, dan terakhir berwarna merah. Nilai nilai tersebut akan dihitung untuk setiap *pixel* yang ada sesuai dengan *lightness* warna yang dihasilkan oleh masing-masing layer. Intensitas cahaya dan warna juga penting dalam penggabungan menjadi suatu cahaya yang baru.



	0.2235	0.1294	Blue	0.4196				
0.5804	0.2902	0.0627		0.2902	0.4824			
0.5804	0.0627	0.0627		0.0627	0.2235	0.2588	0.0627	
0.5176	0.1922	0.0627	Green	0.1922	0.2588	0.2588	0.0627	
0.5176	0.1294	0.1608		0.1294	0.2588	0.2588	0.1608	0.0627
0.5176	0.1608	0.0627		0.1608	0.1922	0.2588	0.2588	0.0627
0.5490	0.2235	0.5490	Red	0.7412	0.7765	0.7765	0.7412	0.0627
0.5490	0.3882	0.5176		0.5804	0.5804	0.7765	0.7765	0.1922
0.5490	0.2588	0.2902		0.2588	0.2235	0.4824	0.2235	0.1608
0.5490	0.2235	0.1608		0.2588	0.2588	0.1608	0.2588	0.1608
0.5490	0.1608	0.1608		0.2588	0.2588	0.2588	0.2588	0.1608

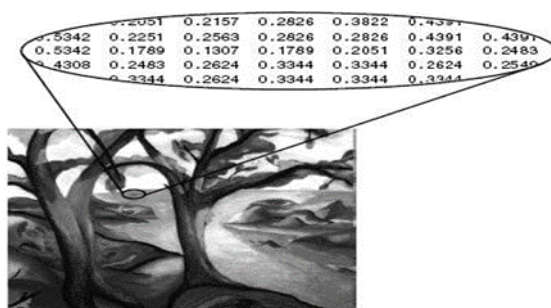
Gambar 2.4 Citra RGB`

2.2.2. Citra Biner

Citra biner merupakan citra yang memiliki hitungan berdasarkan dua intensitas utama yaitu 1 dan 0 untuk setiap pixelnya, dimana 0 akan menjadi perwakilan hitam dan 1 tentunya akan menjadi perwakilan putih. Gambar biner ini merupakan tingkatan terendah yang dapat peneliti buat untuk tingkat abu-abu pembuatan warna dan memiliki pencitraan digital yang cepat karena setiap pixelnya lebih rendah.

2.2.3. Citra Intensitas Keabuan

Gambar ini merupakan gambar dengan intensitas keabuan dan merupakan suatu gambar digital yang memiliki 1 lapisan skala dari RGB memiliki 8bit serta kedalaman dari 0-255. Penyebutan dari gambar ini juga sering disebut gambar grayscale, gambar skala abu-abu ditampilkan sebagai berikut Gambar 2.5.



Gambar 2.5 Citra Grayscale

2.2.4. Format Citra Digital

Format dari citra digital merupakan jenis dan kompresi yang digunakan untuk citra tersebut dan disesuaikan berdasarkan dengan karakteristiknya yang pastinya berbeda - beda (Yang et al, 2016).

1. Bitmap (BMP)

Format yang pertama yaitu format bitmap yang tentunya sudah banyak dijumpai dan digunakan untuk format windows dan biasanya berukuran besar. Besar ukurannya bisa sampai perhitungan megabyte. Dalam format gambar ini terdapat unsur RGB yang dicampurkan menjadi satu file utama. Format gambar ini tidak bisa atau jarang peneliti temui dalam penggunaan online karena ukurannya yang sangat besar. File ini dapat dibuka dengan beberapa aplikasi yaitu View, Paint, dan IrfanView. Untuk melihat detail dari BMP terlihat pada Tabel 2.2 dan 2.3.

Tabel 2.2 Bitmap Info Header

Nama	Ukuran	Penjelasan
"bfType"	2	BM sebagai indentifikasi tipe dari file yang ada
"bfSize"	4	Memori dari file
"bfReserved1"	2	Tidak digunakan
"bfReserved1"	2	Tidak digunakan
"bfOffBits"	4	Offset mulai data <i>pixel</i>

Tabel 2.3 Bitmap Core Header

Nama	Ukuran	Penjelasan
“bcSize”	4	<i>Header Memory</i>
“bcWidth”	2	Width
“bcHeight”	2	Height
“bcPlanes”	2	Harus 1
“bcBitCount”	2	<i>Bits per pixels – 1/4/8/24</i>

2. Joint Photographic Expert Group (JPEG/JPG)

JPEG merupakan ekstensi gambar yang sering dipakai baik secara online maupun offline. Jenis gambar ini juga sangat populer dan memiliki size yang tidak terlalu besar sehingga mudah untuk dioperasikan ataupun untuk digunakan dimanapun. Ukuran yang dibawa juga hanya ratusan Kilobyte sehingga sering digunakan di dunia fotografi.

3. GIF (*Graphics Interchange Format*)

Jenis dari file yang satu ini memungkinkan gambar untuk bergerak dan terdiri dari kumpulan gambar-gambar menjadi sebuah animasi yang sederhana. Kompresi file ini bernama lossless. Mengurangi jumlah isian warna menjadi hanya 256. Jika ingin mengedit gambar menggunakan file editor yang lebih baik seperti adobe photoshop ataupun editor lain yang menerima jenis file ini. File ini juga sangat sering digunakan saat membuat stiker baik di platform windows maupun mobile. Dikarenakan ukurannya yang tidak terlalu besar dan membuat gambar menjadi lebih interaktif. Penggunaan dari GIF ini tentunya ada banyak dan dapat dilihat secara detail dipenjelasan ini.

File GIF cocok digunakan untuk:

- Gambar dengan warna kurang dari 256.
- Logo Gradien.
- Animasi sederhana.
- Mencetak hasil printscreen sederhana.
- Gambar dengan banyak warna, seperti lanskap. F. Gambar dengan gradien atau rona (cahaya titik).

4. PNG (Portable Network Graphics)

PNG sangat sering digunnnakan untuk melengkapi ataupun alternatif dalam hal gambar-gambar trannsparan atau yang membutuhkan kontrol transparansi file ini sangat cocok untuk digunakan. Menggunakan kompresi yang sama dengan GIF yaitu lossless dan tentunya mendukung kedalaman warna hingga 48bit. File ini juga sangat cocok untuk grafik web dikarenakan transparansi dan keindahan yang ditawarkan yang tidak dimiliki oleh GIF maupun format JPEG. (Yang et al., 2016).

2.3. *SSD Mobilenet V2*

Mobilenet V2 SSD merupakan pengembangan lebih lanjut dari versi sebelumnya. SSD Mobilenet V1 Google dirancang untuk mendukung kebutuhan perangkat seluler, atau perangkat dengan daya komputasi rendah, jaringan SSD (one-shot multibox detector) yang ditujukan untuk deteksi objek. SSD Mobilenet V2 terdiri dari SSD yang beroperasi sebagai model dasar dan Mobilenet V2 sebagai model jaringan. SSD mengelola pengenalan objek dengan membuat kotak terikat dan Mobilenet V2 bertujuan untuk mengekstrak fitur yang akan diklasifikasikan nanti (Astuti et al., 2022).

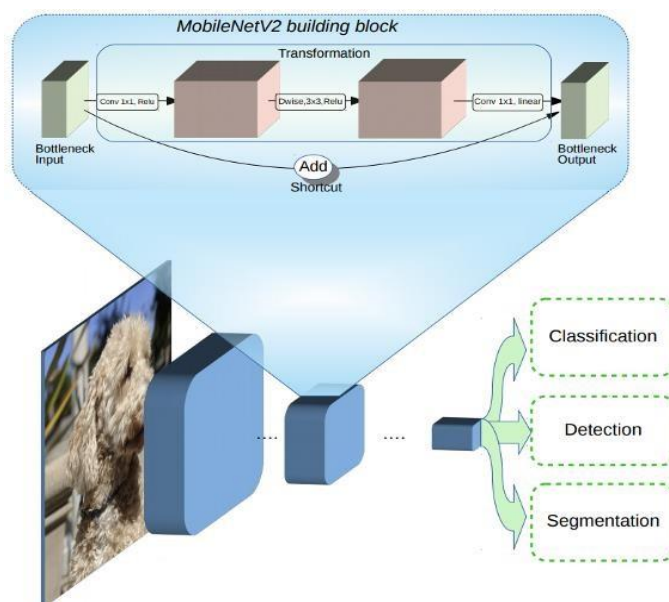
Bidang pandang komputer menggunakan teknologi pembelajaran mendalam untuk mengenali objek. Teknik ini merupakan salah satu bidang pembelajaran mesin yang menggunakan jaringan saraf tiruan untuk menyelesaikan masalah dengan data dalam jumlah besar. Deep learning menggunakan dua metode yang paling populer, yaitu convolutional neural network (CNN) dan recurrent neural network (RNN). Pengembangan arsitektur mendalam secara bertahap mengarah ke arsitektur yang dapat mengenali sistem pengenalan yang lebih cepat dan lebih efisien. Arsitektur yang memperhitungkan hal ini termasuk Single Shot Multibox Detector (SSD) dan (YOLO). (Liu et al., 2015).

Arsitektur Mobilenet V2 digunakan sebagai ekstraktor fitur. Pada langkah ini, gambar input melewati layer konvolusi filter. Pada arsitektur Mobilenet V2, konvolusi dibagi menjadi dua bagian yaitu kedalaman dan rangkaian titik. Depth convolution adalah lapisan convolution dengan ukuran kernel 3x3 dan setiap perhitungan konvolusi memiliki 2 pixel dan 1 pixel offset atau step, sedangkan point convolution memiliki ukuran kernel 1x1 dan 1 pixel atau step. (Howard et al., 2017).

Arsitektur MobileNet menggunakan Batch Normalization (BN) dan Rectified Linear Unit (ReLU). BN digunakan untuk menormalkan proporsi layer di setiap

minigroup. ReLU adalah layer aktivasi CNN yang menggunakan fungsi $f(x) = \max(0, x)$. Hasil akhir dari proses ini adalah fungsi gambar 14×14 dengan 512 filter. Pada fase deteksi target, klasifikasi dan lokalisasi dilakukan secara bersamaan. Fitur citra yang dihasilkan dari ekstraksi fitur memprediksi objek pada citra. Deteksi objek dilakukan menggunakan model Mobilenetv2 yang telah dilatih sebelumnya.

Orang yang terlatih mampu mengenali jenis objek dalam gambar. Untuk setiap citra masukan berukuran 300×300 , deteksi objek dan ekstraksi ciri dilakukan melalui 13 lapis konvolusi, sehingga hasil akhirnya adalah 14×14 . Prediksi bounding box dibuat dengan arsitektur SSD, prediksinya bisa mencapai 8732 prediksi. Di antara sekian banyak prediksi, metode non - maximum suppression digunakan untuk mengeliminasi hasil prediksi dengan nilai sebenarnya yang rendah sesuai ambang batas yang diterapkan sebesar 0,45. Kemudian nilai yang didapat dari prediksi adalah koordinat kotak pembatas, nilai kotak pembatas kepercayaan, objek kelas, dan nilai kepercayaan objek kelas. Hasil akhir dari pengenalan objek adalah citra input dengan hasil pengenalan dengan kotak pembatas, kepercayaan kelas dan objek kelas. (Nufus et al., 2021). Adapun alur kerja Mobilenetv2 dapat dilihat pada Gambar 2.6

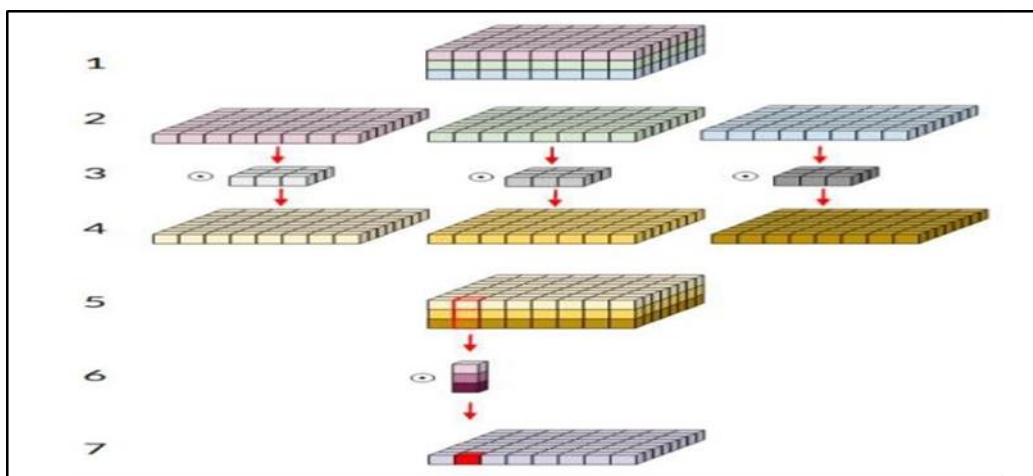


Gambar 2.6 Alur kerja MobileNetV2

(Sik-Ho Tsang, 2019)

Lipatan yang dapat dipisahkan dalam (DSC) menggantikan lipatan standar dengan dua langkah:

1. Konvolusi kedalaman, setiap filter $DF \times DF$ hanya memfilter peta fitur ke kedalaman input;
2. Konvolusi titik, lapisan konvolusional 1×1 yang digunakan untuk menggabungkan jalur informasi dari lapisan dalam. Gambar DSC terlihat Gambar 2.7.



Gambar 2.7 Konvolusi Depthwise (2-5) dan Pointwise (6-7)

Jalur konvolusi dengan parameter yang lebih sedikit menjadikan perhitungan menjadi lebih lebih efisien. Penggunaan parameter dan daya komputasi dapat dilihat pada Tabel 2.4.

Tabel 2.4 Ukuran Parameter dan Komputasi pada Depthwise Separable Convolution

Layer	Ukuran Parameter	Ukuran Komputasi
<i>Standard Conv</i>	$F \times F \times C_1 \times C_2$	$F \times F \times D_M \times D_M \times C_1 \times C_2$
Depthwise Separable	$F \times F \times C_{1+1} \times 1 \times C_1 \times C_2$	$F \times F \times D_M \times D_M \times C_{1+1} \times 1 \times C_1 \times C_2$

F= Filter

C=Filter Konvolusi

DM = Filter *Depthwise*

Perbedaan antara CNN dan SSD yang paling mencolok adalah detector objek yang klasik menjadikan itu sebagai perbedaan utama dan juga tugas-tugas yang diberikan untuk mengeksekusinya. Setelah tugas-tugas ini ditentukan, fungsi kerugian berjalan ujung ke ujung. Metode SSD mencocokkan objek dengan kotak pembatas standar, menggunakan skala dan rasio yang berbeda untuk posisi setiap objek peta. Setiap objek peta fitur memiliki beberapa kotak yang ditautkan. Setiap kotak pembatas dengan IoU lebih besar dari 0,5 dianggap cocok dengan objek dan akan diproses (Zhang, F. 2017).

2.4. Mean Squared Error (MSE) dan Peak Signal to Noise Ratio (PSNR)

Mean square error (MSE) adalah nilai dari error yang didapat dihitung dengan kuadrat dan rata rata dari nilai aslinya dan juga melakukan proses manipulasi dengan citra tersebut. Sedangkan PSNR adalah sinyal yang diukur dibandingkan dengan nilai maksimum yang didapat dan menghitung deras sinyal yang terpengaruh dengan derau. PSNR bila perhitungan desibelnya di bawah 30 db maka terjadi indikasi bahwa adanya kualitas yang rendah. Apabila terdapat nilai 40 db ataupun di atasnya maka dikatakan bahwa citra tersebut memiliki nilai yang bagus dan baik untuk digunakkan. Sebenarnya secara kasat mata peneliti tidak mampu untuk membedakan mana hasil citra yang asli dan dimana hasil dari steganografi, maka dari itu dilakukan perhitungan yang mampu untuk menjelaskannya (Sutoyo et al., 2010).

MSE dan PSNR dapat dihitung dengan persamaan (8) dan (9). Pada persamaan (8), $I(i,j)$ adalah nilai piksel di posisi (i,j) , I' adalah nilai nilai piksel citra hasil proses pada posisi (i,j) , x dan y adalah ukuran file citra. Pada persamaan (9), m adalah nilai piksel maksimum yang dimiliki oleh citra.

$$MSE = \frac{1}{XY} \sum_x \sum_y [I(x,y) - I'(x,y)]^2 \dots \dots \dots (8)$$

$I(i,j)$: menyatakan nilai piksel citra asli.

$I'(i,j)$: merupakan nilai piksel citra proses.

Nilai MSE yang besar, menyatakan penyimpangan atau selisih antara citra hasil penyisipan dengan citra aslinya cukup besar.

$$PSNR = 10 \frac{\log NM^2}{MSE} \quad (9)$$

N.M : dimensi citra (dalam satuan piksel)

PNSR adalah merupakan perhitungan toleransi yang menyatakan tingkat dari rasio derau dalam suatu sinyal ataupun gambar. PNSR itu sendiri adalah nilai utama yang menunjukkan adanya kualitas dari sinyal gambar yang diberikan. Noise merupakan kerusakan sinyal yang diterima oleh gambar di bagian-bagian tertentu yang pasti menurunkan sinyal dari gambar tersebut.

2.5. *Raspberry Pi 3*

Raspberry Pi 3 atau sering disingkat dengan Raspi merupakan sebuah jenis board yang dipasangkan berupa papan tunggal yang bisa digunakan untuk pemutar media, menjalankan program, beberapa permainan, dan aplikasi yang mendukung lainnya. Raspi ini diambil dari singkatan nama buah yaitu buah Raspberry dan digabungkan dengan bahasa python. Raspi memiliki dua jenis penyimpanan yaitu 256MB dan 512MB mempunyai ukuran kecil dan tentunya fleksibel untuk dibawa kemanapun dan bisa dengan praktis melakukan pemrograman yang diperintahkan (Rahmat, 2019). Board Raspberry pi 3 dapat dilihat pada gambar 2.8 di bawah.



Gambar 2.8 Raspberry Pi 3

2.6. Webcam Raspi

Webcam biasanya digunakan untuk menambahkan fungsionalitas kamera ke laptop/komputer. Biasanya perangkat tersebut digunakan pada komputer yang belum memiliki kamera built-in atau pada laptop yang sudah memiliki kamera namun masih kurang bagus sehingga diperlukan colokan tambahan dan kamera game. Raspberry Pi didasarkan pada sistem operasi Linux open source (biasanya menggunakan distribusi Raspbian), tetapi tidak semua driver perangkat keras tertanam di dalamnya. Gambar 2.9 menunjukkan jenis webcam Raspi



Gambar 2.9 Raspi Webcam

2.7. ESP32

ESP32 adalah mikrokontroler seperti Arduino. ESP32 juga dikenal sebagai mikrokontroler SoC (System on Chip) yang diproduksi oleh Espressif System. ESP32 memiliki modul Wi-Fi terintegrasi, Bluetooth dan lampu LED (Kevin, 2021). Fitur dan spesifikasi khusus ESP32 seperti: Processor: mikroprosesor Tensilica Xtensa Dual-Core 32-bit LX6, Frekuensi clock hingga 240 MHz. WiFi: 2.4GHz up to 150 Mbit/s. Bluetooth: BLE (Bluetooth Low Energy) dan Bluetooth legacy. ROM: 448 KB, SRAM: 520 KB, Low Power: bercatu daya rendah, bahkan dapat menjalankan fungsi ADC dalam mode sleep. ESP32 dapat dilihat pada Gambar 2.10 di bawah ini.



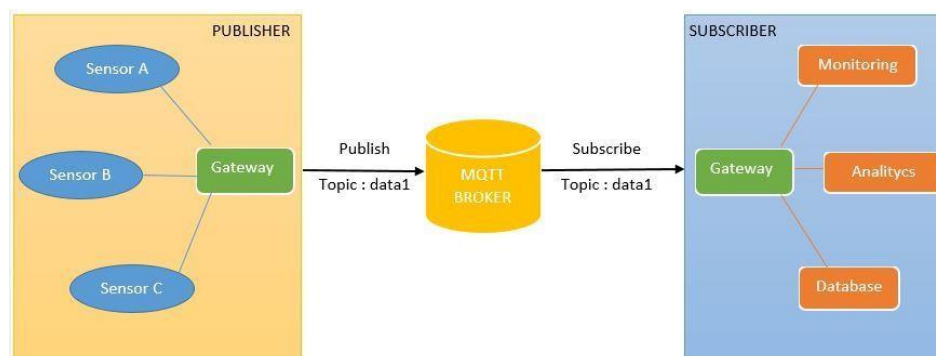
Gambar 2.10 ESP32

2.8. Protokol MQTT

MQTT (Message Queuing Telemetry Transport) adalah suatu sistem yang bekerja di atas tumpukan TCP/IP dan dikembangkan khusus untuk koneksi mesin ke mesin yang tidak memiliki alamat tertentu seperti Arduino dan Raspberry Pi. Sistem kerja MQTT menerapkan informasi publish dan subscribe. Praktik utamanya, perangkat akan langsung menghubungkan diri ke perantara yang ada dan memproses publikasi dari perangkat yang berbeda. IP khusus digunakan sebagai server agar broker dapat melakukan proses dengan baik. Mosquito dan Mosca merupakan beberapa broker yang sekarang sudah ada dan bisa beroperasi.

Saat melakukan proses penerbitan maka data akan dikirimkan ke pelanggan dengan perangkat terkait dan sensor yang sudah ditetapkan. Pelanggan akan mendapatkan informasi dari penerbit. Pelanggan juga bisa memantau dengan menggunakan sensor yang sudah tersedia dan bisa meminta kembali data kepada penerbit.

Topik adalah cara mengelompokkan informasi ke dalam focus kategori. Dalam kinerja utama dari sistem ini topik menjadi bahan yang paling utama dan penting, dimana setiap data transaksi yang ada antara penerbit dan pelanggan harus memiliki subjek. Gambar 2.11 menunjukkan proses perencanaan operasi MQTT.

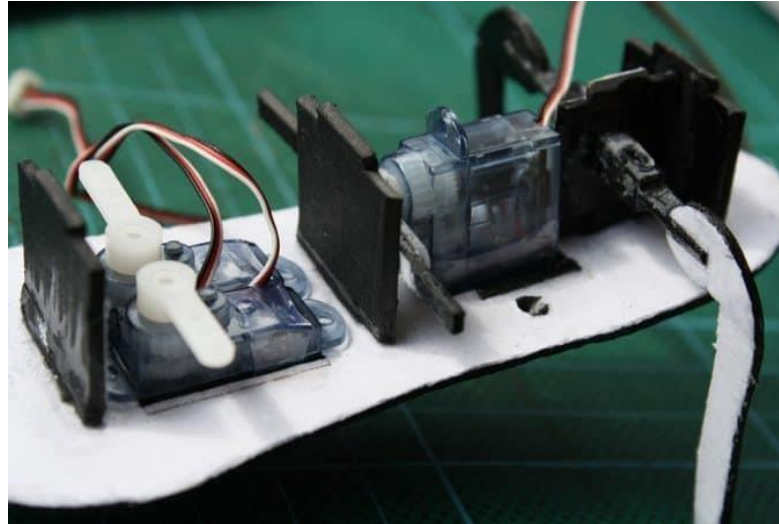


Gambar 2.11 Alur Desain MQTT

2.9. Motor Servo

Untuk memutar dan mendorong objek dengan presisi yang tinggi digunakanlah motor servo untuk memaksimalkan kinerja tersebut baik dalam hal sudut, maupun kecepatan (Schneider Electric, 2022). Moto servo terdiri dari 3 komponen yang wajib ada di dalamnya, yaitu motor, pengontrol dan potensiometer atau sensor. Motor bertindak

sebagai gearbox, memungkinkan perputaran dari poros motor sekunder berputar bersama. Sensor memberikan sinyal secara umpan balik terhadap kontrol dari system utama untuk posisi dari target (Sinaupedia, 2020). Pada gambar 2.12 Anda dapat melihat motor servo.



Gambar 2.12 Motor Servo

2.10. Adaptor 12V

Adaptor merupakan suatu rangkaian yang merupakan perangkat yang berguna untuk melakukan perubahan dari tegangan listrik yang awalnya merupakan tegangan yang tinggi menjadi tegangan listrik yang lebih rendah. Tentunya ada banyak adaptor yang digunakan sesuai dengan fungsi - fungsinya. Dengan adanya adaptor ini tentunya sangat berguna untuk menyesuaikan tegangan listrik yang dibutuhkan baik untuk perangkat maupun untuk rangkaian yang dibuat.

1. Adaptor DC Converter merubah tegangan DC tinggi menjadi DC rendah.
2. Adaptor Step Up dan Step Down yang dapat mengubah tegangan AC rendah menjadi AC tinggi.
3. Adaptor inverter adalah adaptor yang dapat mengubah tegangan DC kecil menjadi tegangan AC besar.
4. Catu daya adalah adaptor yang dapat mengubah tegangan AC tinggi menjadi tegangan DC rendah.



Gambar 2.13 Adaptor

2.11. Dinamo DC 5V

(DC) atau motor arus yang searah merupakan suatu perangkat keras elektromagnetis yang mampu merubah tenaga listrik menjadi tenaga mekanik. Dengan tegangan yang berbeda pada kedua terminal yang diberikan maka akan menentukan arah perputaran motor dan juga polaritas antar kutub yang diberikan tegangan akan menentukan kemana arah dari putaran motor yang akan dilakukan dan perbedaan tegangan akan menjadi penentu kecepatan dari motor.



Gambar 2.14 Dinamo DC

2.12. Mengukur Kinerja Metode dengan *Confusion Matrix*

Pengukuran terhadap kinerja suatu sistem merupakan hal yang penting dimana kinerja sistem menggambarkan seberapa baik sistem dalam pengolahan data. *Confusion*

matrix merupakan salah satu metode yang dapat digunakan untuk mengukur kinerja suatu metode. Pada dasarnya *confusion matrix* mengandung informasi yang membandingkan hasil klasifikasi yang dilakukan oleh sistem dengan hasil klasifikasi yang seharusnya.

Berdasarkan jumlah keluaran kelasnya, sistem dapat dibagi menjadi 4 (empat) jenis yaitu klasifikasi *binary*, *multi-class*, *multi-label* dan *hierarchical*. Pada klasifikasi *binary*, data masukan dikelompokkan ke dalam salah satu dari dua kelas. Jenis klasifikasi ini merupakan bentuk klasifikasi yang paling sederhana dan banyak digunakan. Contoh penggunaannya antara lain dalam sistem yang melakukan deteksi orang atau bukan, sistem deteksi kendaraan atau bukan, dan sistem deteksi pergerakan atau bukan.

Sementara itu, pada bentuk klasifikasi *multi-class*, data masukan diklasifikasikan menjadi beberapa kelas. Sebagai contoh sistem yang dapat mengklasifikasikan jenis kendaraan seperti sepeda, sepeda motor, mobil, bus, truk, dan sebagainya. Bentuk klasifikasi *multi-label* pada dasarnya sama dengan *multi-class* dimana data dikelompokkan menjadi beberapa kelas, namun pada klasifikasi *multi-label*, data dapat dimasukkan dalam beberapa kelas sekaligus. Bentuk klasifikasi yang terakhir adalah *hierarchical*. Data masukan dikelompokkan menjadi beberapa kelas, namun kelas tersebut dapat dikelompokkan kembali menjadi kelas-kelas yang lebih sederhana secara hirarkis. Contohnya dalam penelitian ini, arah pergerakan dikelompokkan menjadi 12 arah pergerakan yang tentunya dapat disederhanakan menjadi 4 arah. Pada pengukuran kinerja menggunakan *confusion matrix*, terdapat 4 (empat) istilah sebagai representasi hasil proses klasifikasi. Keempat istilah tersebut adalah *True Positive* (TP), *True Negative* (TN), *False Positive* (FP) dan *False Negative* (FN). Nilai *True Negative* (TN) merupakan jumlah data negatif yang terdeteksi dengan benar, sedangkan *False Positive* (FP) merupakan data negatif namun terdeteksi sebagai data positif. Sementara itu, *True Positive* (TP) merupakan data positif yang terdeteksi benar. *False Negative* (FN) merupakan kebalikan dari *True Positive*, sehingga data positif, namun terdeteksi sebagai data negatif. Pada jenis klasifikasi *binary* yang hanya memiliki 2 keluaran kelas, *confusion matrix* dapat disajikan seperti pada Tabel 2.5.

Tabel 2.5 Klasifikasi *confusion matrix*

Kelas	Terklasifikasi Positif	Terklasifikasi Negatif
Positif	TP (True Positive)	FN (False Negative)
Negatif	FP (False Positive)	TN (True Negative)

Berdasarkan nilai *True Negative* (TN), *False Positive* (FP), *False Negative* (FN), dan *True Positive* (TP) dapat diperoleh nilai akurasi, presisi dan *recall*. Nilai akurasi menggambarkan seberapa akurat sistem dapat mengklasifikasikan data secara benar. Dengan kata lain, nilai akurasi merupakan perbandingan antara data yang terklasifikasi benar dengan keseluruhan data. Nilai akurasi dapat diperoleh dengan Persamaan 2.3. Nilai presisi menggambarkan jumlah data kategori positif yang diklasifikasikan secara benar dibagi dengan total data yang diklasifikasi positif. Presisi dapat diperoleh dengan Persamaan 2.4 Sementara itu, *recall* menunjukkan berapa persen data kategori positif yang terklasifikasikan dengan benar oleh sistem. Nilai *recall* diperoleh dengan Persamaan berikut.

$$Akurasi = (TP + TN) / (TP + TN + FP + FN) * 100\% \dots\dots\dots (2.3)$$

$$Precision = (TP / (TP + FP)) * 100\%. \dots\dots\dots (2.4)$$

$$Recall = (TP / (TP + FN)) * 100\%. \dots\dots\dots (2.5)$$

dimana:

1. TP adalah *True Positive*, yaitu jumlah data positif yang terklasifikasi dengan benar oleh sistem.
2. TN adalah *True Negative*, yaitu jumlah data negatif yang terklasifikasi dengan benar oleh sistem.
3. FN adalah *False Negative*, yaitu jumlah data negatif namun terklasifikasi salah oleh sistem.
4. FP adalah *False Positive*, yaitu jumlah data positif namun terklasifikasi salah oleh sistem

2.13. Parameter Pengujian Sistem

Pengujian sistem pengenalan wajah bermasker perlu dilakukan untuk mengetahui tingkat akurasi dalam pemanfaatannya. Kondisi yang dilakukan memiliki parameter pengujian, seperti, Wajah tanpa masker, wajah mengenakan masker N95, wajah mengenakan masker jenis atau warna berbeda, wajah mengenakan masker dengan posisi kemiringan. Pengujian dilakukan membagi jumlah pengujian berhasil/jumlah sampel pengujian selanjutnya dikalikan dengan 100 persen. Berikut ini rumus yang digunakan untuk mengetahui hasil akurasi pengenalan wajah bermasker:

$$\text{Akurasi} = \frac{\text{Jumlah Data Uji Yang Sukses}}{\text{Total Data Uji}} \times 100\% \quad (2.6)$$

Selanjutnya, kesalahan sistem dalam mengenali wajah bermasker baik itu dalam pengenalan identitas dari dataset yang telah dilakukan pelatihan (*training*) sebelumnya maupun pengujian pengenalan identitas wajah bermasker yang tidak terdapat dalam dataset dapat dilakukan pengukuran parameter pengujian dengan menggunakan rumus *False Acceptance Rate (FAR)*, sebagai berikut :

$$\text{FAR} = \frac{\text{Jumlah Data Uji Yang Gagal}}{\text{Total Data Uji}} \times 100\% \quad (2.7)$$

2.14. Penelitian Terkait

Penelitian terdahulu dapat dilihat pada tabel 2.6 berikut :

Tabel 2.6 Penelitian Terdahulu

Peneliti	Metode	Keterangan
Sindy (2019)	MobileNet-SSD	Membuat arsitektur jaringan yang mampu untuk mendeteksi objek berupa manusia secara langsung ataupun <i>realtime</i> dengan memanfaatkan sebuah teknik <i>deep learning</i> tentunya dengan

		MobileNet-SSD juga. Intensitas cahaya yang diuji antara 50-550 lux dengan pengujian jarak antara 1-10 meter dengan kondisi bervariasi dan akurasi 91,67%.
Itje <i>et al.</i> (2017)	Analisis Tekstur	<p>Dalam penelitiannya yang berjudul Deteksi Kualitas Telur Menggunakan Analisis Tekstur. Dalam penelitian ini bahwa telur merupakan barang yang sangat baik buat anak-anak. Subjek pada penelitian ini adalah mendeteksi kualitas telur menggunakan analisis tekstur pada kulit telur. Sehingga bisa terlihat apakah ternyata tekstur yang ditampilkan baik ataupun bermasalah pada data yang diberikan.</p>
Trisnaningtyas <i>et al.</i> (2015)	K-Nearest Neighbor	<p>Dalam penelitiannya yang membuat pengklasifikasian kebersihan dari telur memanfaatkan algoritma ataupun metode K-Nearest Neighbor. Telur merupakan tentunya salah satu makanan yang menyehatkan dan memiliki kandungan gizi yang baik dan tentunya harus dijaga kebersihannya. Telur juga sumber protein dan tentunya telur juga sangat mudah dicerna dan disiapkan. Subjek utama penelitian ini adalah aplikasi deteksi maupun klasifikasi mutu telur yang baik maupun buruk untuk memilih</p>

		telur yang aman untuk digunakan maupun dikonsumsi. Tentunya telur yang bersih akan lebih sehat dan dapat dibedakan dengan kandungan baik didalamnya dikarenakan kebersihan yang baik
Ramadhan (2020)	Neural Network MobileNet	Melakukan pengujian mengenai pengaturan dari <i>neural network</i> pada sisi penggunaan <i>client</i> untuk mendapatkan <i>learning rate</i> yang optimal dan baik. Dalam pengujian nya didapatkan kesimpulan bahwa pada <i>client side</i> melakukan prosesi <i>training</i> lebih cepat dibandingkan dengan melakukannya pada <i>server side</i> , namun memiliki <i>loss function</i> yang lebih baik dibandingkan <i>client side</i> .
Pramana <i>et al.</i> , 2021	<i>Single shot multibox detector</i> (SSD)	Melakukan penelitian mengenai deteksi hal-hal yang berbau sepak bola, baik seperti bola, halangan, posisi gawang, dan objek-objek yang ada disepelitinnya dengan bantuan robot menggunakan metode <i>single shot multibox detector</i> (SSD). Pada hasil uji yang diterima didapatkan bahwa mAP rata-rata yang diperoleh yaitu 0.8 total total loss 1.5. Selain itu validasi optimal terhadap objek-objek yang ada mendapatkan keberhasilan prediksi 98,45% yang menandakan prediksi yang dibuat telah baik.




BAB 3

ANALISIS DAN PERANCANGAN

3.1. Pengumpulan Data

Dalam penelitian ini, hal yang utama adalah pengumpulan data yang disebut juga dataset berupa citra sampah yang diambil. Data citra sampah organik dan anorganik yang digunakan pada penelitian ini diambil dari kaggle dan beberapa diambil secara langsung di lingkungan rumah penulis. Pengambilan data dilakukan dengan menggunakan kamera *smartphone* 16 MP yang berekstensi .jpg/.png. Data citra sampah memiliki 6 atribut kelas (*class*) dataset, yakni : *cardboard* (393), *glass* (491), *metal* (400), *paper* (584), *plastic* (472) dan *trash* (127). Contoh dataset dapat dilihat pada tabel 3.1.

Tabel 3.1 Isi Dataset

ID	Nama Citra	Gambar	Jumlah	Label
1	<i>cardboard</i>		393	Anorganik
2	<i>glass</i>		491	Anorganik
3	<i>metal</i>		400	Anorganik

4

paper

584

Organik

5 *plastic*

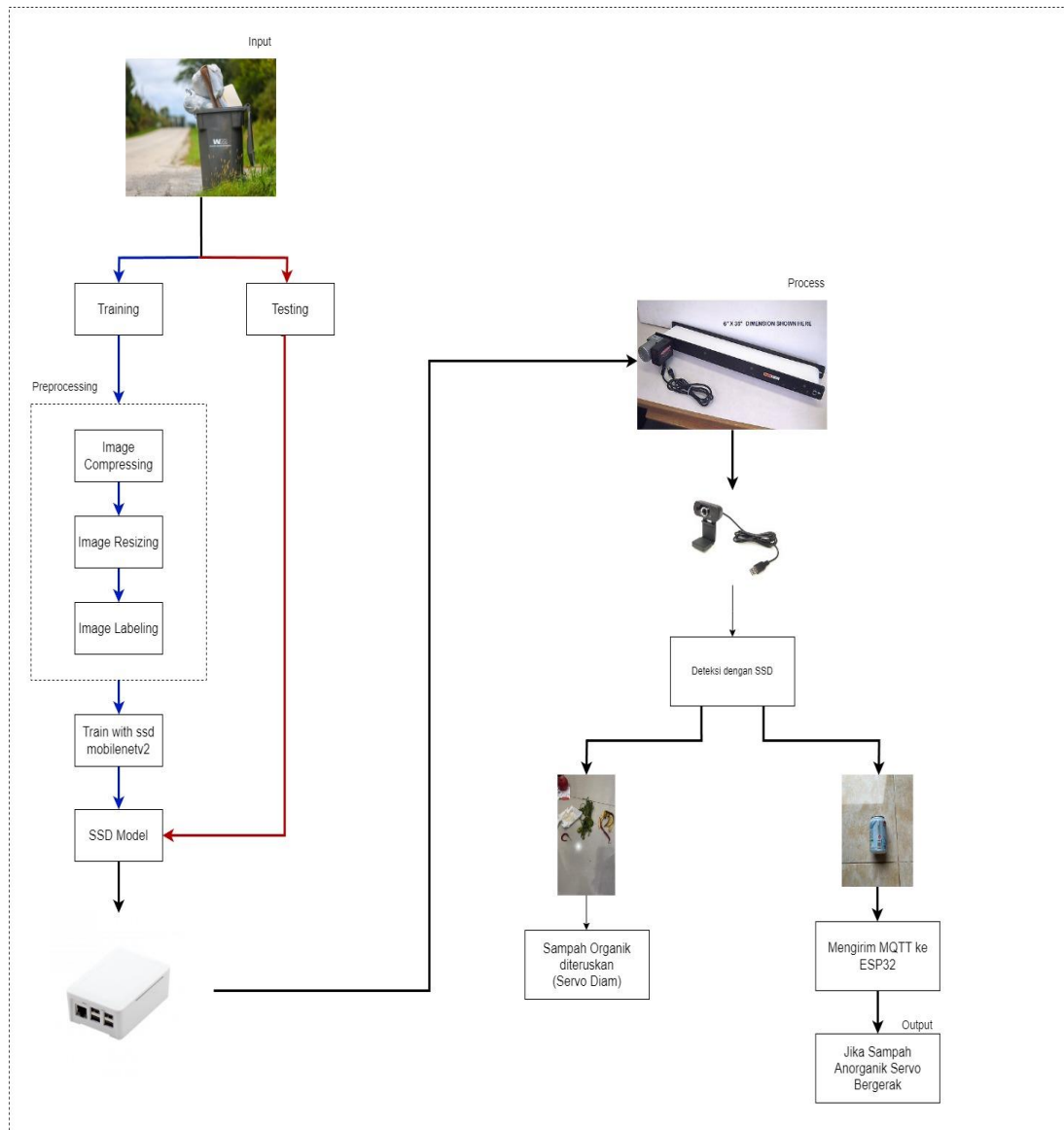
472 Anorganik

6 *trash*

127 Anorganik

3.2. Arsitektur Umum

Diawali dengan pengumpulan dataset sampah organik dan organik, kemudian dilakukan proses preprocessing dengan melakukan compressing, resizing, labeling image yang selanjutnya masuk ke tahap training dengan menggunakan SSDMobilenet V2. setelah selesai di training selanjutnya akan menghasilkan model, dimana model akan dimasukkan kedalam raspi 3b+, raspi terhubung dengan usb kamera yang diletakkan pada ujung conveyor belt. dimana nantinya sampah akan bergerak maju mendekati kamera menggunakan conveyor belt. apabila sampah sudah mendekati dengan jarak ≤ 20 meter maka raspi akan memberikan perintah ke esp32 melalui mqtt untuk menggerakkan motor servo sesuai dengan jenis sampah yang terdeteksi secara realtime. Serangkaian proses ini dapat dilihat pada gambar 3.1.



Gambar 3.1 Arsitektur umum

3.3. Persiapan Training Data

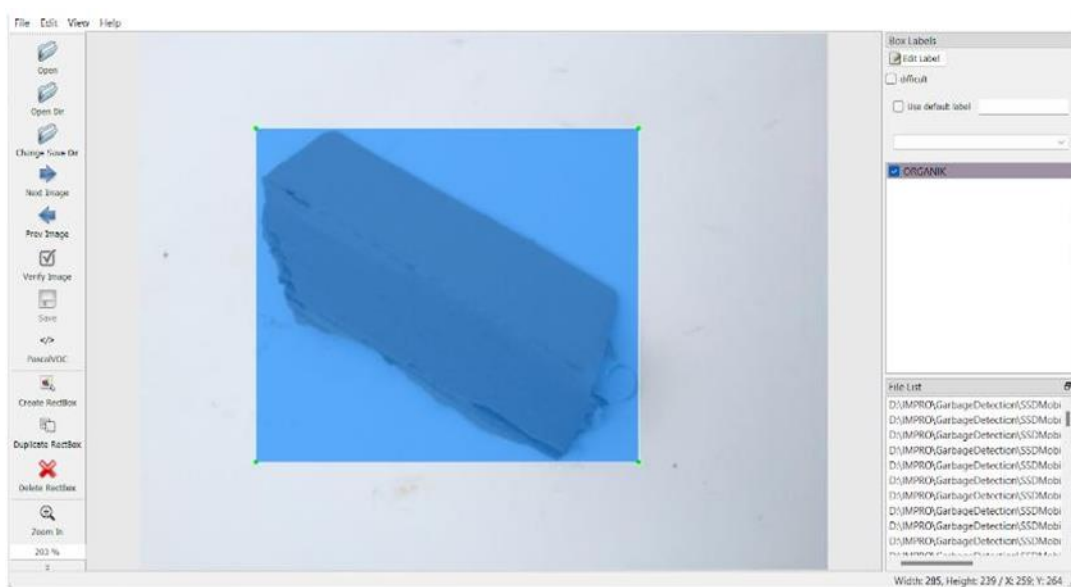
Dalam persiapan utama peneliti akan melakukan suatu step ataupun langkah awal peneliti akan melakukan pembagian dataset lalu melabeli citra untuk training data dengan menggunakan tools Labellmg.

Proses splitting data dapat dilihat pada pseudocode di bawah ini :

```

Function: Splitting Dataset
Input: Dataset
Output: Splitted Dataset
BEGIN
    GET Dataset
    THEN Split to train 80% and test 20%
END

```



Gambar 3.2 Labelling dengan Labelimg

Pada aplikasi terdapat beberapa menu yang dapat digunakan seperti *edit label*, *open file*, dll yang dapat peneliti gunakan sesuai fungsinya untuk melabel gambar data peneliti. Berikut ini jumlah keseluruhan data yang akan dipakai pada prosesi data training

Jumlah sampel data training:

- 2 kategori label (ORGANIK dan ANORGANIK)
- 1752 sampel data dengan pembagian 80:20 untuk dataset training & testing sehingga didapat 1402 citra untuk training & 350 citra untuk testing.
- 1350 data sampah ORGANIK
- 402 data sampah ANORGANIK

3.4. Instalasi Tensorflow

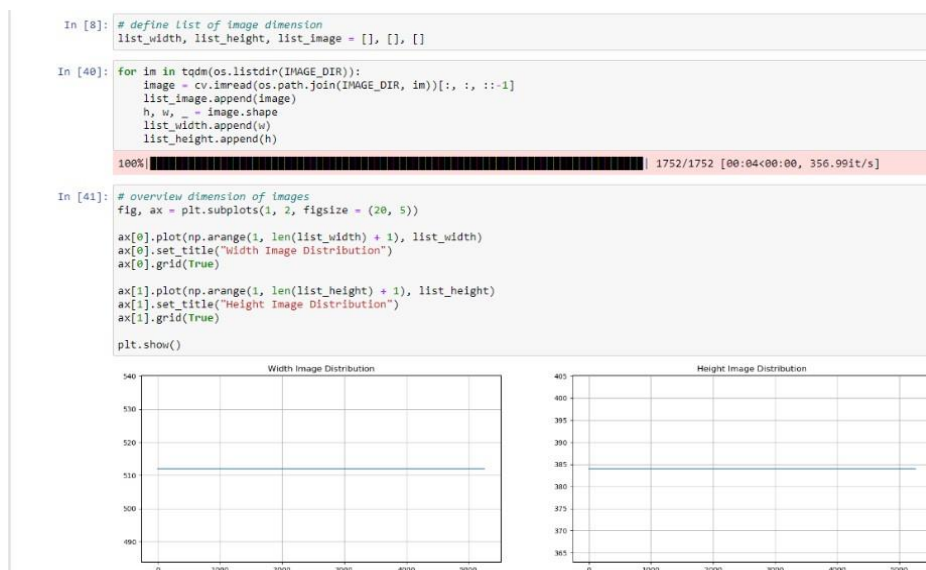
Pada instalasi tensorflow proses dilakukan di local karena google colabs sudah tidak mendukung tensorflow versi 1.14 sebagai environment untuk melakukan proses training ssdmobilenetv2

```
(garbageDet) C:\Users\Asus>pip install --ignore-installed --upgrade tensorflow==1.14
Collecting tensorflow==1.14
  Using cached tensorflow-1.14.0-cp37-cp37m-win_amd64.whl (68.3 MB)
Collecting protobuf>=3.6.1
  Downloading protobuf-4.21.12-cp37-cp37m-win_amd64.whl (526 kB)
    ----- 526.5/526.5 kB 6.6 MB/s eta 0:00:00
Collecting google-pasta>=0.1.6
  Using cached google_pasta-0.2.0-py3-none-any.whl (57 kB)
Collecting tensorboard<1.15.0,>=1.14.0
  Using cached tensorboard-1.14.0-py3-none-any.whl (3.1 MB)
Collecting six>=1.10.0
  Using cached six-1.16.0-py2.py3-none-any.whl (11 kB)
Collecting gast>=0.2.0
  Using cached gast-0.5.3-py3-none-any.whl (19 kB)
Collecting tensorflow-estimator<1.15.0rc0,>=1.14.0rc0
  Using cached tensorflow_estimator-1.14.0-py2.py3-none-any.whl (488 kB)
Collecting wrapt>=1.11.1
  Using cached wrapt-1.14.1-cp37-cp37m-win_amd64.whl (35 kB)
Collecting absl-py>=0.7.0
  Using cached absl_py-1.3.0-py3-none-any.whl (124 kB)
Collecting keras-applications>=1.0.6
  Using cached Keras_Applications-1.0.8-py3-none-any.whl (50 kB)
Collecting numpy<2.0,>=1.14.5
  Using cached numpy-1.21.6-cp37-cp37m-win_amd64.whl (14.0 MB)
Collecting wheel>=0.26
  Downloading wheel-0.38.4-py3-none-any.whl (36 kB)
Collecting astor>=0.6.0
  Using cached astor-0.8.1-py2.py3-none-any.whl (27 kB)
Collecting grpcio>=1.8.6
  Using cached grpcio-1.51.1-cp37-cp37m-win_amd64.whl (3.7 MB)
Collecting termcolor>=1.1.0
  Using cached termcolor-2.1.1-py3-none-any.whl (6.2 kB)
Collecting keras-preprocessing>=1.0.5
  Using cached Keras_Preprocessing-1.1.2-py2.py3-none-any.whl (42 kB)
Collecting h5py
  Using cached h5py-3.7.0-cp37-cp37m-win_amd64.whl (2.6 MB)
```

Gambar 3.3 Proses Instalasi Tensorflow

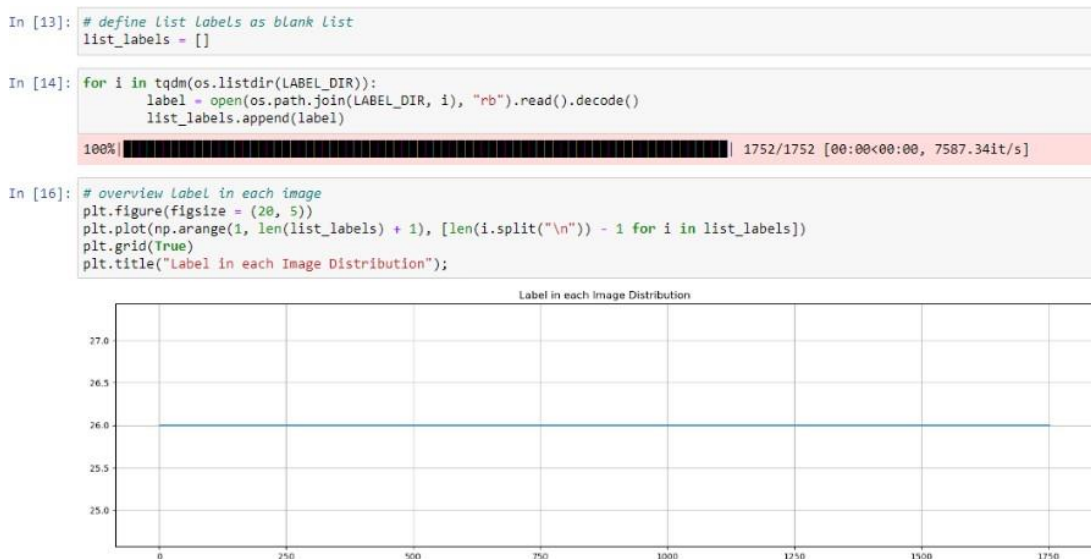
3.5. Analisis Gambar dan Tabel

Pada proses selanjutnya peneliti akan melakukan analisis pada gambar dan label lebih jelas akan ditampilkan di bawah ini :



Gambar 3.4 Proses Analisis Citra

Dari gambar 3.4 didapat bahwa seluruh citra memiliki ukuran yang sama yaitu 520 x 380 piksel dan berjumlah 1752 citra.



Gambar 3.5 Proses Analisis Label

Pada Gambar 3.5 didapat bahwa jumlah label yang peneliti buat dan pada perhitungan tersebut peneliti dapat label dengan jumlah 1752 *file*. Dapat disimpulkan bahwa jumlah gambar = jumlah label (1 gambar yang dibuat mewakili 1 label yang ada).

3.6. Konversi Data XML ke CSV

Setelah selesai proses pelabelan, dilakukan proses selanjutnya yaitu membuat file CSV yang berisi data seluruh nama file citra yang akan digunakan dalam proses training beserta kelas dan anotasinya, lalu file yang berisi data seluruh nama file citra yang akan digunakan dalam proses testing beserta kelas dan anotasinya dan file .pbt.txt yang berisi nama dari seluruh kelas yang ada dalam file XML. Proses ini dapat dilihat pada pseudocode di bawah :


```

Function: Convert xml to csv
Input: xml file
Output: csv file
BEGIN
    GET XML
    THEN Convert XML to CSV format
END

```

Proses dan hasil konversi ini dapat dilihat pada gambar 3.6

```

In [19]: import os
import glob
import pandas as pd
import xml.etree.ElementTree as ET

def xml_to_csv(path):
    xml_list = []
    for xml_file in glob.glob(path + '/*.xml'):
        tree = ET.parse(xml_file)
        root = tree.getroot()
        for member in root.findall('object'):
            value = (root.find('filename').text,
                    int(root.find('size')[0].text),
                    int(root.find('size')[1].text),
                    member[0].text,
                    int(float(member[4][0].text)),
                    int(float(member[4][1].text)),
                    int(float(member[4][2].text)),
                    int(float(member[4][3].text)))
            xml_list.append(value)
    column_name = ['filename', 'width', 'height', 'class', 'xmin', 'ymin', 'xmax', 'ymax']
    xml_df = pd.DataFrame(xml_list, columns=column_name)
    return xml_df

def main(): # this is the part we change to work with our setup
    for directory in ['train', 'test']:
        xml_path = os.path.join(os.getcwd(), 'annotations')
        xml_df = xml_to_csv(xml_path)
        xml_df.to_csv('data/{}.csv'.format(directory), index=None)
        print('Successfully converted xml to csv.')

main()

Successfully converted xml to csv.
Successfully converted xml to csv.

```

```

In [20]: # read dataset
dataset_train = pd.read_csv("data/train.csv")
dataset_train.head()

```

```

Out[20]:

```

	filename	width	height	class	xmin	ymin	xmax	ymax
0	cardboard1.jpg	512	384	ORGANIK	1	2	512	384
1	cardboard102.jpg	512	384	ORGANIK	1	1	512	384
2	cardboard104.jpg	512	384	ORGANIK	1	1	512	384
3	cardboard105.jpg	512	384	ORGANIK	1	1	512	384
4	cardboard106.jpg	512	384	ORGANIK	1	1	512	384

Gambar 3.6 Proses Konversi XML ke CSV dan Hasilnya

3.7. Generate File Record

Pada proses ini dilakukan generate file record yang dijelaskan pada pseudocode di bawah ini :

```
Function: Generate record file
Input: Labels in CSV and label_map
Output: record file
BEGIN
    GET labels and label_map
    THEN Generate record file
END
```

File record dapat dilihat pada gambar 3.7

```
In [38]: def main(_):
writer = tf.python_io.TFRecordWriter(FLAGS.output_path)
path = os.path.join(FLAGS.image_dir)
examples = pd.read_csv(FLAGS.csv_input)
grouped = split(examples, 'filename')
for group in grouped:
    tf_example = create_tf_example(group, path)
    writer.write(tf_example.SerializeToString())




writer.close()
output_path = os.path.join(os.getcwd(), FLAGS.output_path)
print('Successfully created the TFRecords: {}'.format(output_path))

In [39]: if __name__ == '__main__':
tf.app.run()

Successfully created the TFRecords: D:\IMPRO\GarbageDetection\SSD\mobilenetV2\data\train.record
An exception has occurred, use %tb to see the full traceback,

SystemExit
```

Hasil tfrecord disimpan dalam folder data

Name	Date modified	Type	Size
 test	12/20/2022 10:59 AM	Microsoft Excel Co...	79 KB
 train	12/20/2022 10:59 AM	Microsoft Excel Co...	79 KB
 train.record	12/20/2022 11:05 AM	RECORD File	30,081 KB

Gambar 3.7 File Record

Label map berisi dua item yang akan digunakan untuk pengklasifikasian sampah, berikut isi label map :

```
item {
    id: 1
    name: "ORGANIK"
```

```

        display_name: "ORGANIK"
    }
    item {
        id: 2
        name: "ANORGANIK"
        display_name: "ANORGANIK"
    }

```

Pada hasil akhir pembuatan *Tf Record* maka data akan disimpan dalam folder dan terbagi menjadi *testing* dan *training data*, selanjutnya akan tampil train.record yang menjadi hasil dari *tfrecord* yang dibuat sebelumnya.

3.8. Training Data

Model yang penulis gunakan adalah `ssd_mobilenet_v2` Model ini diambil dari tensorflow zoo. Berikut pseudocodenya :

```

Function: Get Pretrained Model
Output: SSD Mobilenet V2
BEGIN
    ACCESS Tensorflow Zoo
    GET SSD MobileNet V2
END

```

Berikut isi dari folder SSD MobileNet V2 yang diakses :

Name	Date modified	Type	Size
saved_model	12/20/2022 12:13 PM	File folder	
checkpoint	3/29/2018 7:48 PM	File	1 KB
frozen_inference_graph.pb	3/29/2018 7:48 PM	PB File	68,055 KB
model.ckpt.data-00000-of-00001	3/29/2018 7:48 PM	DATA-00000-OF-0...	65,924 KB
model.ckpt.index	3/29/2018 7:48 PM	INDEX File	15 KB
model.ckpt.meta	3/29/2018 7:48 PM	META File	3,415 KB
pipeline	3/29/2018 7:48 PM	Configuration Sou...	5 KB

Gambar 3.8 Isi folder SSD MobileNet V2

Selanjutnya adalah melakukan kustomisasi konfigurasi dengan mengubah isi dari file pipeline. Hasil konfigurasi ini dapat dilihat pada gambar 3.9.

```
model {  
  ssd {  
    num_classes: 2  
    image_resizer {  
      fixed_shape_resizer {  
        height: 300  
        width: 300  
      }  
    }  
  }  
  feature_extractor {  
    type: "ssd_mobilenet_v2"  
    depth_multiplier: 1.0  
    min_depth: 16  
    conv_hyperparams {  
      regularizer {  
        l2_regularizer {  
          weight: 3.99999989895e-05  
        }  
      }  
    }  
    initializer {  
      truncated_normal_initializer {  
        mean: 0.0  
        stddev: 0.02999999993294  
      }  
    }  
  }  
  activation: RELU_6  
  batch_norm {
```

Gambar 3.9 Konfigurasi Pipeline

```

        decay: 0.899999976158
        epsilon: 1.0
    }
}
fine_tune_checkpoint: "./nn-models/detection/model.ckpt"
num_steps: 2000
fine_tune_checkpoint_type: "detection"
}
train_input_reader {
  label_map_path: "./data/garbage_map.pbtxt"
  tf_record_input_reader {
    input_path: "./data/train.record"
  }
}
eval_config {
  num_examples: 5000
  max_evals: 10
  use_moving_averages: false
}
eval_input_reader {
  label_map_path: "./data/garbage_map.pbtxt"
  shuffle: false
  num_readers: 1
  tf_record_input_reader {
    input_path: "./data/val.record"
  }
}
}

```

Gambar 3.9 Konfigurasi Pipeline bagian 2

Pada penelitian ini digunakan proses training sejumlah 2000 steps. Proses training dapat digambarkan pada pseudocode di bawah ini :

```

Function: Training Model
Input: Previously Trained Model and pipeline config
Output: Tensorflow model
BEGIN
    GET Previously Trained Model and pipeline config
    THEN train model
END

```


3.9. Testing Detection

Pada tahap akhir akan mendapati hasil data dan sukses melakukan testing detection untuk mendapatkan hasil akhir. Peneliti menggunakan kode program berikut :



Gambar 3.11 Proses testing

BAB 4

IMPLEMENTASI DAN HASIL PENELITIAN

4.1. Spesifikasi Kebutuhan Aplikasi

Pengujian dilakukan dengan menggunakan *hardware* & perangkat lunak sebagai berikut :

4.1.1. Kebutuhan Perangkat Keras

Perangkat keras yang akan digunakan adalah :

1. Laptop dengan spesifikasi :
 - a. Laptop : Lenovo IdeaPad 5 14ABA7 RYZEN 5
 - b. Processor : AMD Ryzen 5 5625U
 - c. RAM : 16 GB
 - d. Storage : SSD 512 GB
 - e. Graphic Card : Integrated AMD Radeon Graphics

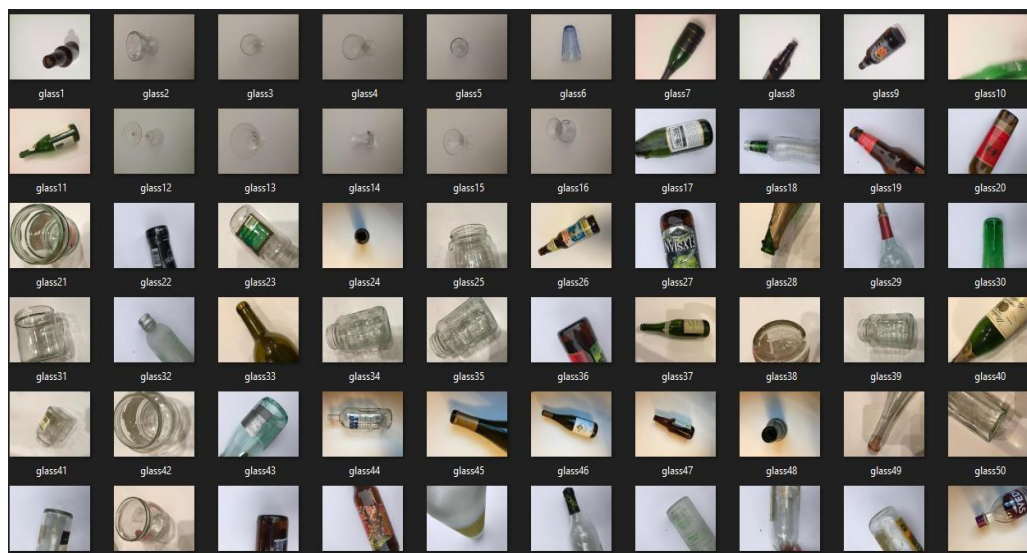
4.1.2. Kebutuhan Perangkat Lunak

Perangkat lunak yang dibutuhkan adalah :

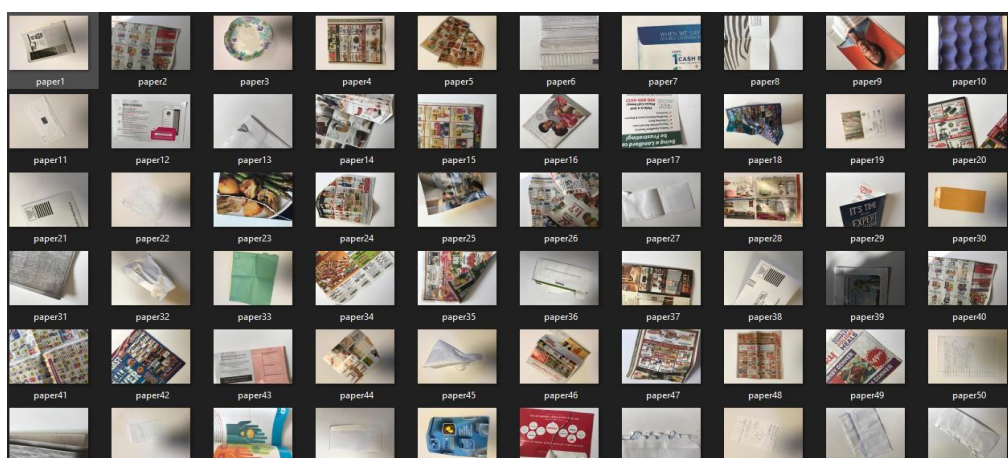
1. Sistem Operasi : Windows 11 Pro 64-bit
2. Bahasa Program : Python
3. *Library* : Tensorflow
Opencv
Tflite
4. *Text Editor* : Visual Studio Code
Google Colab

4.2. Penggunaan Data

Data citra sampah organik dan anorganik yang digunakan pada penelitian ini diambil dari kaggle dan beberapa diambil secara langsung di lingkungan rumah penulis. Pengambilan data dilakukan dengan menggunakan kamera smartphone 16 MP yang berekstensi .jpg/.png. Jumlah data yang dikumpulkan sebanyak 1500 data untuk sampah anorganik yang terbagi menjadi 4 kelas yaitu, glass, metal, plastic, dan trash. Jumlah data untuk sampah organik sebanyak 1000 data yang terbagi menjadi 2 kelas yaitu, cardboard dan paper. Adapun beberapa contoh data pada masing- masing jenis sampah terdapat pada gambar 4.1 dan 4.2.



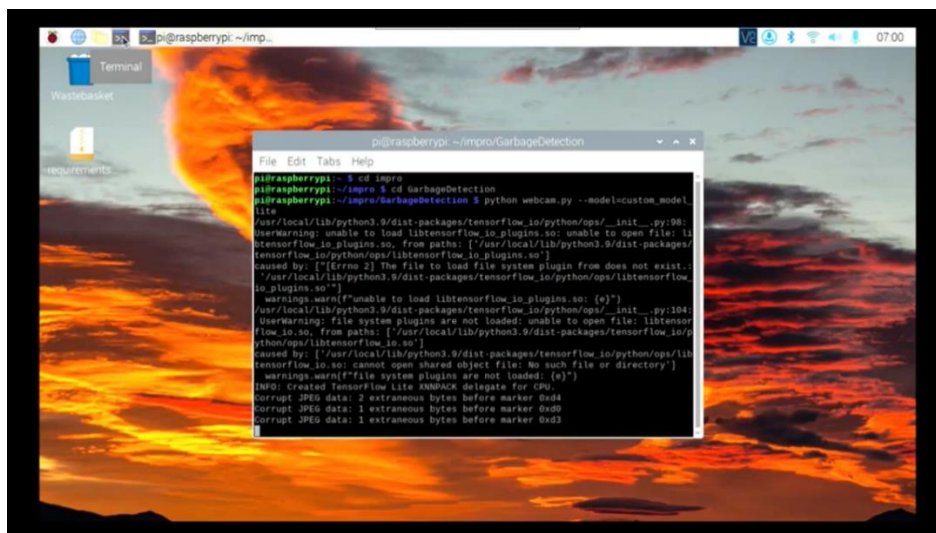
Gambar 4.1 Sampah Anorganik



Gambar 4.2 Sampah Organik

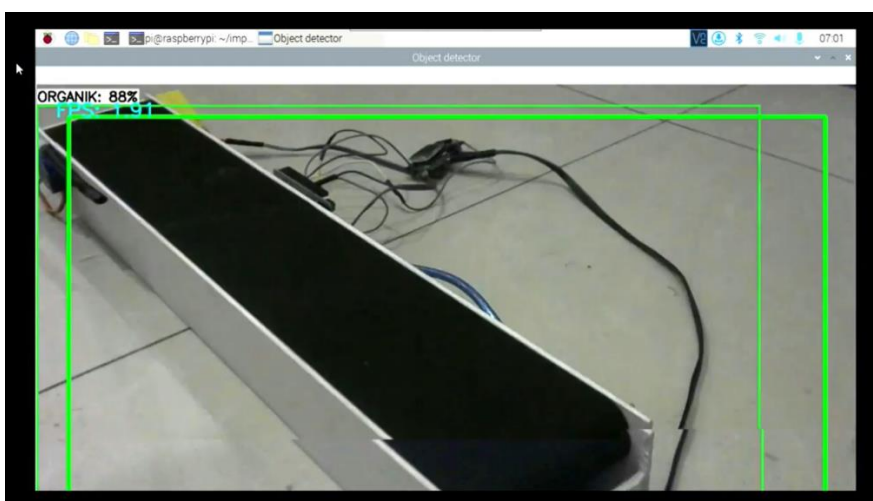
4.3. Pengujian

Pada pengujian ini peneliti menggunakan Raspi 3 dan *Webcam Raspi* untuk mendeteksi benda benda yang akan diuji tentunya. Sampah ataupun data uji akan diletakkan dan kamera akan mendeteksi benda tersebut. Program akan dijalankan di raspi. Proses menjalankan program dapat dilihat pada gambar 4.3



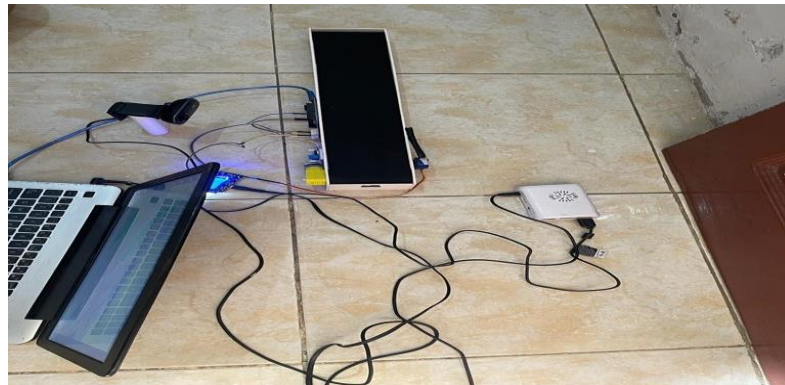
Gambar 4.3 Proses menjalankan program

Setelah jendela terbuka dan menampilkan kamera maka akan langsung mendeteksi objek. Tempat pengujian dilakukan didalam ruangan dengan intensitas cahaya yang cukup. Tampilan window aplikasi berjalan dapat dilihat pada gambar 4.4.



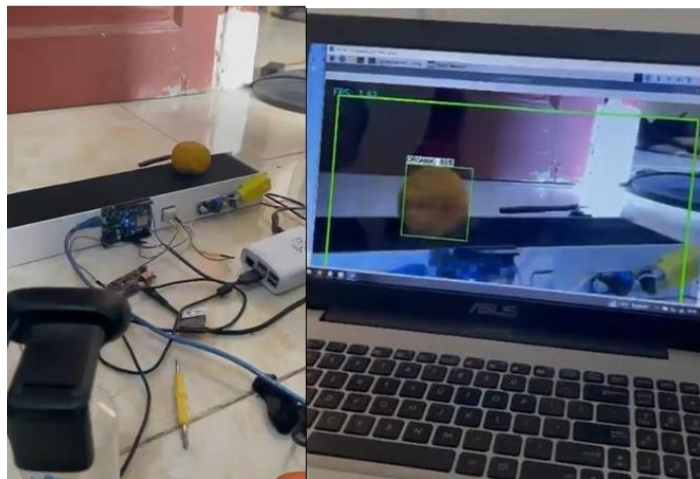
Gambar 4.4 Tampilan Aplikasi

Peralatan untuk pengujian dapat dilihat pada gambar 4.5.

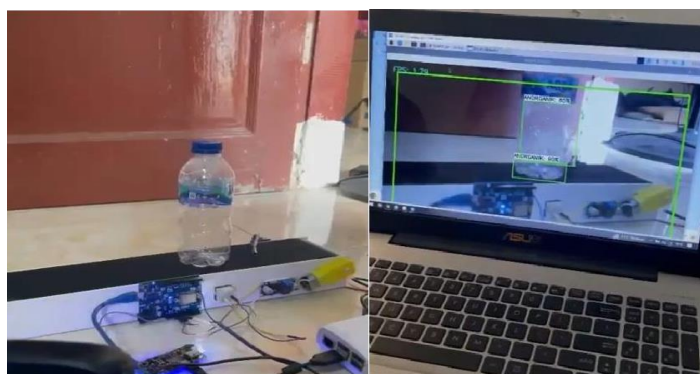


Gambar 4.5 Peralatan Pengujian

Proses pengujian pada sampah organik & anorganik dapat dilihat pada gambar 4.6 & 4.7



Gambar 4.6 Pengujian pada Sampah Organik

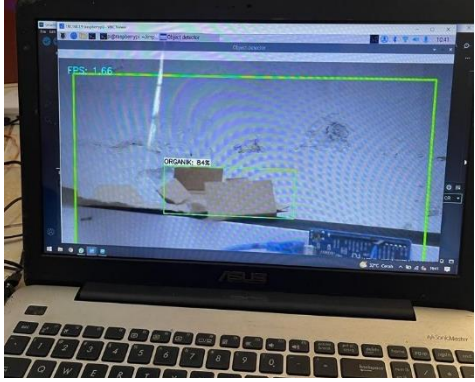
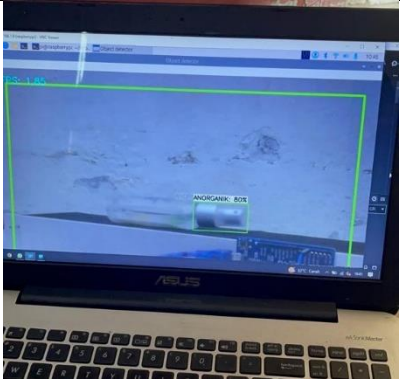
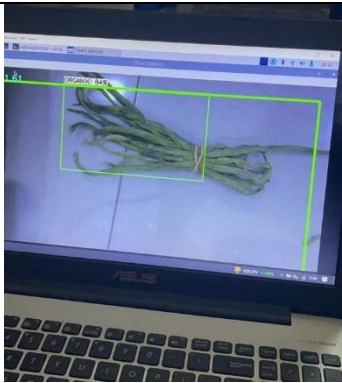


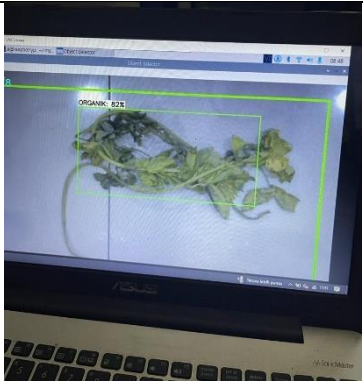
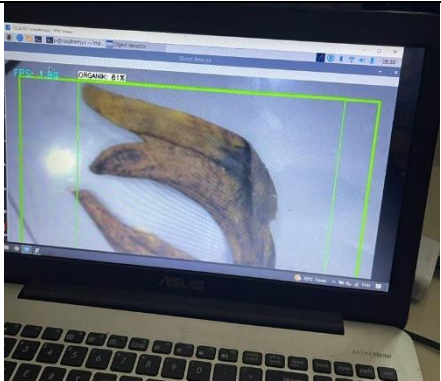
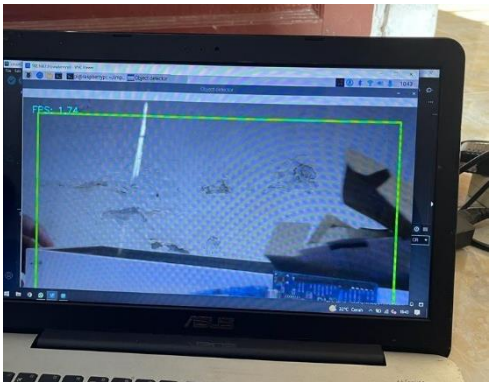
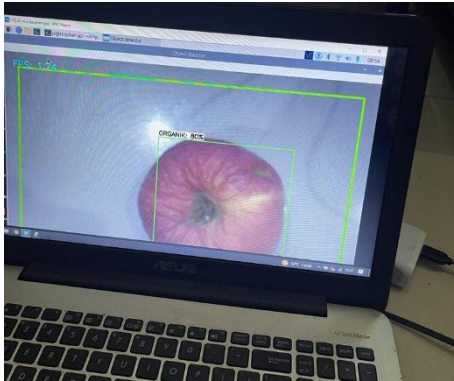
Gambar 4.7 Pengujian pada Sampah Anorganik

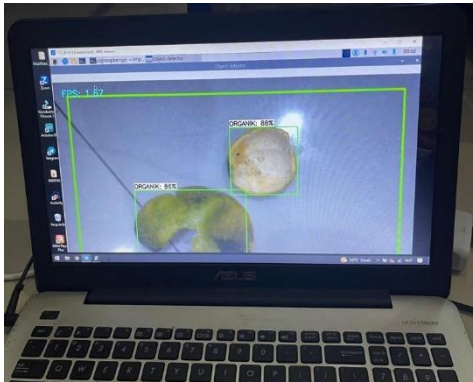
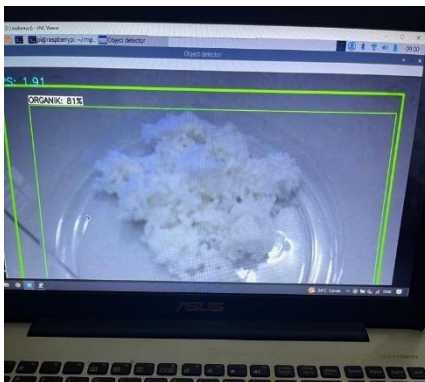
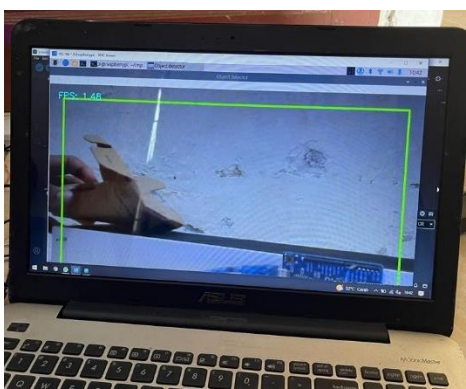
4.4. Hasil Pengujian

Beberapa sampel dari pengujian dapat dilihat pada tabel 4.1

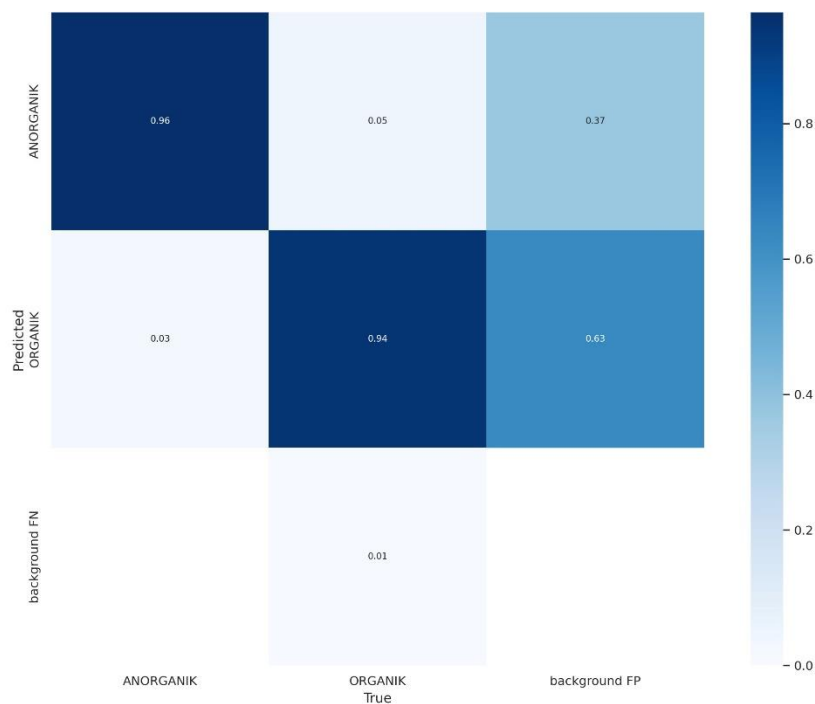
Tabel 4.1 Sampel pengujian

N o	Citra	Actual	Predicted	T P	T N	F P	F N
1		Organik	Organik	1	0	0	0
2		Anorganik	Anorganik	1	0	0	0
3		Organik	Organik	1	0	0	0

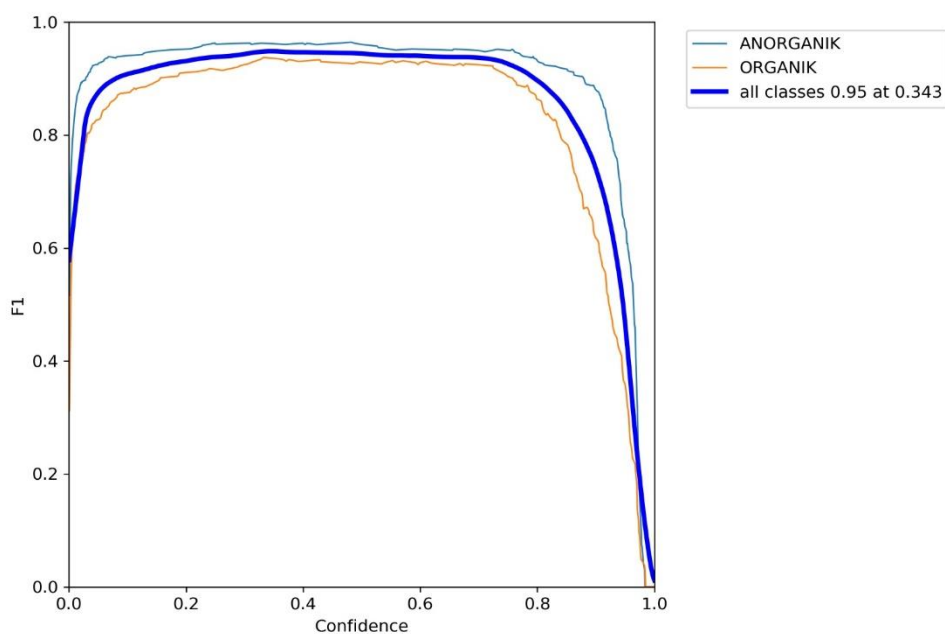
4		Organik	Organik	1	0	0	0
5		Organik	Organik	1	0	0	0
6		Anorganik	Tidak Terprediksi	0	0	0	1
7		Organik	Organik	1	0	0	0

8		Organik & Organik	Organik & Organik	2	0	0	0
9		Organik	Organik	1	0	0	0
10		Anorga nik	Tidak Terpredik si	0	0	0	1

Dari pengujian didapat nilai akurasi sebesar 0.96 atau 96% untuk class sampah anorganik dan 0.94 untuk class sampah organik dengan nilai *F1 - score* sebesar 0.95 atau 95% Hasil ini dapat dilihat pada grafik di gambar 4.8 & 4.9.



Gambar 4.8 Nilai Akurasi yang Didapatkan



Gambar 4.9 Nilai *F1 - Score* yang Didapatkan

BAB 5

KESIMPULAN DAN SARAN

5.1. Kesimpulan

Setelah dilakukan implementasi, didapat kesimpulan bahwa :

1. Sistem mampu menampilkan persentase prediksi jenis sampah yang dideteksi dan menunjukkan nilai benar pada pengujian tiap bendanya dan mendapatkan akurasi sebesar 0.96 atau 96% untuk class sampah anorganik dan 0.94 untuk class sampah organik dengan nilai *F1 - score* sebesar 0.95 atau 95%
2. Kesalahan sistem dalam mengklasifikasi jenis sampah dapat dipengaruhi oleh intensitas cahaya dan jarak kamera.
3. Penerapan sistem ini memiliki potensi besar dalam meningkatkan efisiensi sistem pengelolaan sampah, dengan memberikan solusi otomatis yang dapat membantu dalam proses pemilahan.

5.2. Saran

Aplikasi yang dibuat tentunya masih memiliki kekurangan maupun keterbatasan maka penulis memberikan beberapa saran untuk penelitian selanjutnya :

1. Pada penelitian selanjutnya diharapkan bisa membuat pemilahan menggunakan hand robot agar bisa dipilah secara bersamaan.
2. Pada penelitian selanjutnya diharapkan bisa memperbesar kapasitas conveyor belt.
3. Pada penelitian selanjutnya diharapkan bisa membuat pemilahan khusus untuk besi dan logam agar dapat didaur ulang.
4. Pada penelitian selanjutnya diharapkan bisa Melakukan penambahan dataset untuk sampah organik.

DAFTAR PUSTAKA

- Aliyah, A. N. 2022. Implementasi Metode Human Activity Recognition (Har) Menggunakan Mediapipe Holistics dan Algoritma Long Short Term Memory (Lstm) untuk Menerjemahkan Gerakan Bahasa Isyarat Menjadi Kosa Kata. Skripsi. UIN Syarif Hidayatullah Jakarta.
- Andari, N. 2009. Perbandingan Budaya Indonesia dan Jepang (Tinjauan Tradisi Penamaan dan Gerak Isyarat Tubuh). *Parafrase* Vol. 09 No.02 September 2009.
- Astuti, I., Ariestya, W. W., Solehudin, B. 2022. Deteksi Objek Daun Semanggi Secara Real Time Menggunakan CNN-Single Shot Multibox Detector (SSD). *JURNAL ILMIAH FIFO*. Vol. XIV No.1, 47-58.
- Damayanti, V. C. 2017. Rancang Bangun Sistem Pengunci Loker Otomatis Dengan Kendali Akses Menggunakan RFID. Politeknik Negri Sriwijaya.
- Gonzales, R.C., & Woods, R.E. (2002). *Digital Image Processing: 2nd Edition*. New Jersey: Prentice Hall.
- Grundmann, M. 2020. Mediapipe hands: On-device real-time hand tracking. *arXiv preprint arXiv:2006.10214*.
- Iksan H. Z., Saufian. 2017. Mobile Learning: Innovation In Teaching And Learning Using Telegram. *IJPTE*.
- Iskandar, N. M., Sumaryo, S., & Kurniawan, E. 2018. Implementasi Sistem Pengenal Pola Tangan Berbasis Pengolahan Citra Untuk Sistem Pertolongan Darurat. *eProceedings of Engineering*. 5(3).
- Kevin, K. 2021. Menggunakan Internal Sensor ESP32. Medium, di akses September 2023. <https://kenckevin.medium.com/menggunakan-internal-sensor-esp32-75867b40dfc9>.

- Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y. dan Berg, A.C., (2015). SSD: Single Shot Multibox Detector, International Journal of European Conference on Computer Vision, Vol 5.
- Makahaube, S., Sambul, A. M., & Sompie, S. R. 2021. Implementation of Gesture Recognition Technology for Self-Education Service Platform. *Jurnal Teknik Informatika*, 16(4), 465-472.
- Nufus, G. K., Mustafid, M., Gernowo, R. 2021. Sentiment Analysis for Video on Demand Application User Satisfaction with Long Short Term Memory Model. 3S Web of Conferences. Vol. 317.
- Pramana, C. G. W., Khrisne, D. C., Sastra, N. P. 2021. Rancang Bangun Object Detection Pada Robot Soccer Menggunakan Metode Single Shot Multibox Detector (SSD MobileNet V2). *Jurnal SPEKTRUM*. Vol. 8(2) Juni 2021.
- Rahmat, R. F., Pulungan, A. F., Faza, S and Budiarto, F. 2019. Image classification of ribbed smoked sheet using Learning Vector Quantization. 1 st Intl Conf on Computing and Applied Informatics.
- Ramadhan, F. E. 2020. Penerapan Image Classification Dengan Pre-Trained Model MobileNet Dalam Client-Side Machine Learning. Skripsi. UIN Syarif Hidayatullah Jakarta.
- Reslab. 2018. Robotics & Embedded System Laboratory Sistem Komputer Universitas Andalas. http://reslab.sk.fti.unand.ac.id/index.php?option=com_k2&view=item&id=229:mengenal-mqtt-protokol-untuk-iot&Itemid=303.
- Rizki, A. B., & Zuliarso, E. 2022. Klasifikasi Teknik Bulutangkis Berdasarkan Pose Dengan Convolutional Neural Network. *Jurnal Ilmiah Informatika*, 10(02), 96-101.
- Romindo, R., Barus, O. P., Pangaribuan, J. J., Pratama, Y. A., & Wiliem, E. 2022. Implementasi Algoritma Support Vector Machine Terhadap Klasifikasi Pose Balet. *Building of Informatics, Technology and Science (BITS)*, 4(3), 1558-1566.

- Schneider Electric. 2022. Servo Drives and Motors. Se.com, diakses pada September 2023. <https://www.se.com/id/en/product-subcategory/2935-servo-drives-and-motors/>.
- Sela, E. i., & Ihsan, M., 2017. Deteksi Kualitas Telur Menggunakan Analisis Tekstur. *Indonesian Journal of Computing and Cybernetics System (IJCCS)*. 11(2).
- Septerina, D. 2016. Rancang Bangun Conveyor Pada Alat Pengisi Minuman Otomatis Dengan Kecepatan Putaran Motor DC (Direct Current) Pada PLC (Programmable Logic Controller). <http://eprints.polsri.ac.id/3881/>.
- Shindy, F. 2019. Pendeteksian Objek Manusia Secara *Real Time* dengan Metode MobileNet SSD Menggunakan *Movidius Neural Compute Stick* pada *Raspberry Pi*. Universitas Sumatera Utara. Medan.
- Sinaupedia. 2020. Pengertian Motor Servo. Sinaupedia.com, diakses pada September 2023. <https://sinaupedia.com/pengertian-motor-servo/>.
- Sirait, H. 2022. Aplikasi Sistem Pemantauan Berbasis CCTV dan Perhitungan Kapasitas Media Penyimpanan DVR
- Surjono, & Herman, D. 1996. Eksperimen Pengiriman sinyal televisi dengan pemancar TV dan CCTV serta Pemanfaatannya dalam Pendidikan.
- Sutoyo. T, Mulyanto. Edy, Suhartono. Vincent, Dwi Nurhayati Oky, Wijanarto. 2010. Teori Pengolahan Citra Digital. UDINUS Semarang.
- Tanugraha, F. D., Pratikno, H., & Kusumawati, W. I. 2022. Pengenalan Gerakan Olahraga Berbasis (Long Short-Term Memory) Menggunakan Mediapipe. *Journal of Advances in Information and Industrial Technology*, 4(1), 37-46.
- Trisnaningtyas, P. R., & Maimunah. 2015. Klasifikasi Mutu Telur Berdasarkan Kebersihan Kerabang Telur Menggunakan *K-Nearest Neighbor*. Konferensi Nasional Informatika 2015.

- Tsang, S. 2019. Weight Model (Image Classification). Towardsdatascience.co m, diakses september 2023. <https://towardsdatascience.com/review- mobilenet v2- light-weight- model-image-classification-8febb490e61c>.
- Wahyudi, D. A., & Kartowisastro, I. H. 2011. Menghitung Kecepatan Menggunakan Computer Vision. Jakarta Barat: Computer Engineering Department.
- Yang, S., Zhang, Y., Su, X., Shi, E., Zhang. H. 2016. Automatic reading of domestic electric meter: an intelligent device based on image processing and ZigBee/Ethernet communication. *Journal of Real-Time Image Processing*. Vol. 12, 133-143 (2016).
- Yelmanova, E., & Romanyshyn, Y. 2017. Quantitative Evaluation of Contrast for a Complex Image by its Histogram. *XIII International Conference TCSET' 2017*. pp. 688-692, 2017.
- Zein, A. 2018. Pendeteksian Kantuk Secara Real Time Menggunakan PustakaOPENCV dan DLIB PYTHON. Tangerang: Program Studi Teknik Informatika, STMIK Eresha.
- Zhang, F. 2017. Neuroticism. In B. J. Carducci (Editor-in-Chief) & C. S. Nave (Vol. Ed.), *The Wiley-Blackwell encyclopedia of personality and individual differences: Vol. I. Models and theories*. Hoboken, NJ: John Wiley & Sons