**LAPORAN TUGAS AKHIR**

**IMPLEMENTASI PENGOLAHAN CITRA DIGITAL OBJEK MANUSIA DENGAN METODE YOLOV4-Tiny PADA ARM PROSESOR NANOPI M4V2**

Diajukan guna melengkapi sebagian syarat dalam mencapai gelar Sarjana Strata Satu (S1)

Logo, company name

Description automatically generated

Disusun Oleh :

Nama : Ahmad Faizin

N.I.M : 41420110145

Pembimbing : Rachmat Muwardi, B.Sc, ST., M.Sc

**PROGRAM STUDI TEKNIK ELEKTRO FAKULTAS TEKNIK UNIVERSITAS MERCU BUANA JAKARTA 2022**

# **LEMBAR PENGESAHAN**

**IMPLEMENTASI PENGOLAHAN CITRA DIGITAL OBJEK MANUSIA DENGAN METODE YOLOV4-Tiny PADA ARM PROSESOR NANOPI M4V2**

Disusun Oleh :

Nama : Ahmad Faizin

NIM : 41420110145

Jurusan : Teknik Elektro

Pembimbing

(Rachmat Muwardi, B.Sc., ST., M.Sc)

Mengetahui,

Ketua Program Studi Koordinator Tugas Akhir

(Dr. Eko Ihsanto, M.Eng) (Muhammad Hafizd Ibnu Hajar, ST. M.Sc)

# **LEMBAR PERNYATAAN**

Yang bertanda tangan di bawah ini,

Nama : Ahmad Faizin

N.I.M : 41420110145

Jurusan : Teknik Elektro

Fakultas : Teknik

Judul Skripsi : Implementasi Pengolahan Citra Digital Objek Manusia Dengan Metode YoloV4-Tiny Pada Arm Prosesor Nanopi M4v2

Dengan ini menyatakan bahwa hasil penelitian Tugas Akhir yang telah saya buat ini merupakan hasil karya sendiri dan benar keasliannya. Apabila ternyata di kemudian hari penelitian skripsi ini merupakan hasil plagiat atau penjiplakan terhadap karya orang lain, maka saya bersedia mempertanggung jawabkan sekaligus bersedia menerima sanksi berdasarkan aturan tata tertib Universitas Mercubuana .

Demikian, pernyataan ini saya buat dalam keadaan sadar dan tanpa paksaan.

Penulis,

( Ahmad Faizin )

# **ABSTRAK**

Pada saat ini sistem pendeteksian objek masih banyak menggunakan perangkat yang berukuran besar, seperti penggunaan PC sebagai perangkat pendukung untuk pendeteksian objek, sehingga perangkat ini sangat susah jika akan digunakan untuk menjadi sistem pengamanan di fasilitas umum yang berbasis deteksi objek manusia, sedangkan saat ini sudah banyak Mini PC yang sudah menggunakan Arm Processor dengan spesifikasi yang tinggi dan sudah mendukung sistem berbasis IoT.

Oleh sebab itu, untuk melakukan pendeteksian objek manusia penulis mencoba melakukan penelitian mengenai deteksi objek manusia dengan menggunakan pernagkat Mini PC Nanopi M4V2 yang memiliki kecepatan dalam pemrosesan dengan didukung oleh CPU Dual-Core Cortex-A72 (up to 2.0 GHz) + Cortex A53 (Up to 2.0 GHz) dan Ram 4 Gb DDR4 selain itu untuk sistem pendeteksian objek manusia penulis menggunakan metode *You only look once (*YOLO) dengan tipe YoloV4-Tiny dimana metode ini memiliki kecepatan mendeteksi dan akurasi yang tinggi.

Dengan menggunakan perangkat Nanopi M4V2 dan dengan menggunakan metode YOLOV4-Tiny simulasi pada pengenalan objek manusia mendapatkan tingkat pendeteksian 100% dengan nilai FPS 2.81-3.55.

**Kata Kunci** : *Arm Prosesor, NanoPi M4V2, YOLOv4-Tiny, Deteki Objek, Deteksi Objek Manusia*

# *ABSTRACT*

*At this time object detection systems still use large devices, such as the use of PCs as supporting devices for object detection, so this device is very difficult to use as a security system in public facilities based on human object detection, whereas currently there are many Mini PCs that already use an Arm Processor with high specifications and already support IoT-based systems.*

*Therefore, to detect human objects, the author tries to conduct research on human object detection using the Nanopi M4V2 Mini PC which has processing speed and is supported by a Dual-Core Cortex-A72 CPU (up to 2.0 GHz) + Cortex A53 (Up to 2.0 GHz). to 2.0 GHz) and 4 Gb DDR4 Ram. In addition, for the human object detection system, the author uses You only look once (YOLO) method with the YoloV4-Tiny type where this method has high detection speed and accuracy.*

*By using the Nanopi M4V2 device and by using the YOLOV4-Tiny method of human object recognition, we get a 100% detection rate with an FPS value of 2.81-3.55.*

**Keywords**: *Arm Processor, NanoPi M4V2, YOLOv4-Tiny, Object Detection, Human Object Detection*

# **DAFTAR ISI**

[**LEMBAR PENGESAHAN i**](#_LEMBAR_PENGESAHAN)

**[LEMBAR PERNYATAAN ii](#_LEMBAR_PERNYATAAN)**

**[ABSTRAK iii](#_ABSTRAK)**

**[ABSTRACT iv](#_ABSTCT)**

**[DAFTAR ISI v](#_DAFTAR_ISI)**

**[DAFTAR GAMBAR vii](#_DAFTAR_GAMBAR)**

**[DAFTAR TABEL viii](#_DAFTAR_TABEL)**

**[BAB I PENDAHULUAN 1](#_BAB_I_PENDAHULUAN)**

[1.1. Latar Belakang 1](#_Latar_Belakang)

[1.2. Rumusan Masalah 4](#_Rumusan_Masalah)

[1.3. Tujuan 4](#_Tujuan)

[1.4. Batasan Masalah 5](#_Batasan_Masalah)

[1.5. Metodologi Penelitian 5](#_Metodelogi_Penelitian)

[1.6. Sistematika Penulisan 6](#_Sistematika_Penulisan)

**[BAB II TINJAUAN PUSTAKA DAN LANDASAN TEORI 8](#_BAB_II_TINJAUAN)**

[2.1. Tinjauan Pustaka 8](#_Tinjauan_Pustaka)

*[2.2.](#_Arm_Processor)* [Arm Processore 17](#_Arm_Processor)

[2.3. Nanopi M4V2 18](#_Nanopi_M4V2)

[2.4. Pengolahan Citra Digital 20](#_Pengolahan_Citra_Digital)

[2.5. YoloV4-Tiny 21](#_YoloV4-Tiny)

*[2.6.](#_Python)**[Python](#_Python)* [23](#_Python)

*[2.7.](#_Open_CV)**[Open Cv](#_Open_CV)* [24](#_Open_CV)

*[2.8.](#_Webcam)**[Webcam](#_Webcam)* [24](#_Webcam)

[2.9. Visual Studi Code 25](#_Visual_Studio_Code)

[2.10. Solid State Disk (SSD) 26](#_SSD_(Solid_State)

[2.11. Layar 27](#_Layar)

[2.12. Vidio 27](#_Vidio)

[2.10. Real Time Vidio 28](#_Real_time_vidio)

[2.10. Streaming Vidio 28](#_Streaming_vidio)

**[BAB III METODOLOGI PENELITIAN 29](#_BAB_III_METODELOGI)**

[3.1. Metodelogi Penelitian 31  
3.2. Diagram Blok 34](#_Metodelogi_penelitian_1)

[3.3. Perancangan Perangkat Keras 35](#_Perancangan_Perangkat_Keras)

[3.4. Perancangan Perangkat Lunak 36](#_Perancangan_Perangakt_Lunak)

[3.5. Perancangan Program Perangkat Lunak 37](#_Perancangan_Program_Perangkat)

[3.6. Skema Penelitian 40](#_Skema_Penelitian)

**[BAB IV ANALISA DATA DAN PEMBAHASAN 42](#_BAB_IV_ANALISA)**

[4.1. Pengujian Tingkat Akurasi 42](#_Pengujian_Tingkat_Akurasi)

[4.1.1. Tujuan Pengujian Tingkat Akurasi 42](#_Tujuan_Pengujian_Tingkat)

[4.1.2. Prosedur Pengujian Tingkat Akurasi 42](#_Prosedur_Pengujian_Tingkat)

[4.1.3. Hasil Pengujian Tingkat Akurasi 43](#_Hasil_Pengujian_Tingkat)

[4.2. Perbandingan Tingkat Akurasi 46](#_Perbandingan_TIngakt_Akurasi)

[4.3. Performa CPU dan RAM 48](#_Performa_CPU_dan)

**[BAB V KESIMPULAN DAN SARAN 51](#_BAB_V_KESIMPULAN)**

[5.1. Kesimpulan 51](#_Kesimpulan)

[5.2. Saran 52](#_Saran)

**[DAFTAR PUSTAKA 53](#_DAPTAR_PUSTAKA)**

# **DAFTAR GAMBAR**

[Gambar 2.1 Nanopi M4V2 20](#GambarNanopi)

[Gambar 2.2 *Webcam* 25](#GambarWebcam)

[Gambar 2.9 Visual Studio Code](#GambarVisualstudioCode)  26

[Gambar 3.1 Grid algoritma dengan nilali confidance 30](#GambarAlgoritmaconfidace)

[Gambar 3.2 Grid algoritma non-maximum suppression 32](#GambarAlgoritmaNMS)

[Gambar 3.3 Grid algoritma hasil perbandingan true 32](#GambarAlgoritmaTrue)

[Gambar 3.4](#GambarAlgoritmabb22) *[Grid algoritma hasil perbandingan false bb22](#GambarAlgoritmabb22)* [33](#GambarAlgoritmabb22)

[Gambar 3.5](#GambarAlgoritmabb36) *[Grid algoritma hasil perbandingan false bb36](#GambarAlgoritmabb36)* [33](#GambarAlgoritmabb36)

[Gambar 3.6](#GambarAlgoritmayoloNMS) *[Hasil grid algoritma YOLO berdasarkan NMS](#GambarAlgoritmayoloNMS)* [34](#GambarAlgoritmayoloNMS)

[Gambar 3.7 Blok Diagram 34](#BlokDiagram)

[Gambar 3.8 Perancangan Perangkat Keras 36](#PrangaktKeras)

[Gambar 3.9 Perangkat Keras 36](#PrangaktKerasFisik)

[Gambar 3.10 Tampilan Visual Studio Code 37](#visualstudiocode3)

[Gambar 3.11 Kerangka Diagram Perogram Perangkat Lunak 38](#PerangkatLunak)

[Gambar 3.12 Skema Penelitian 18](#SkemaPenelitian)

[Gambar 4.1 Hasil 1 Objek Hadap Belakang 22](#Hasil1objek)

[Gambar 4.2 Hasil 2 Objek Hadap Depan 44](#Hasil2objek)

[Gambar 4.3 Hasil Banyak Objek 45](#Hasilbanyakobjek)

[Gambar 4.4 Kurva Perbandingan Vidio 1 18](#Kurvagambar1)

[Gambar 4.5 Kurva Perbandignan Vidio 2 22](#KurvaGambar2)

[Gambar 4.7 Kondisi CPU dan RAM Pada Saat Tidak Dijalankan 49](#PenggunaanCPU)

[Gambar 4.6 Kondisi CPU dan RAM Pada Saat Dijalankan 49](#PenggunaanCPU2)

# **DAFTAR TABEL**

[Tabel 2.1 Hasil dari Percobaan 10](#TabelHasilPercobaan1)

[Tabel 2.2](#TabelHasilPercobaan2) *[Hasil dari Percobaan](#TabelHasilPercobaan2)* [12](#TabelHasilPercobaan2)

[Tabel 2.3](#TabelHasilPercobaan3) *[Hasil dari Percobaan](#TabelHasilPercobaan3)* [13](#TabelHasilPercobaan3)

[Tabel 2.4](#TabelLiteraturriview) *[Literatur Riview](#TabelLiteraturriview)* [15](#TabelLiteraturriview)

[Tabel 2.5](#TabelSpesifikasiNanoPI) *[Spesifikasai dari Nanopi M4V2](#TabelSpesifikasiNanoPI)* [18](#TabelSpesifikasiNanoPI)

[Tabel 4.1 Hasil deteksi objek YOLOv4-Tiny 46](#Tabelberbandinganobjek)

[Tabel 4.2 Performa CPUpada deteksi Objek 50](#TabelPerformaCPU)

# **BAB I PENDAHULUAN**

## **Latar Belakang**

Pada zaman globalisasi saat ini perkembangan teknologi sangat pesat sehingga ketergantungan akan teknologi tidak dapat dipungkiri. Beberapa teknologi yang berkembang pesat saat ini seperti sistem teknologi pada microprosessor yang digunakan diberbagai perangkat elektronik, sistem keamanan yang menggunakan kamera dengan berbasis IoT, alat absensi dengan menggunakan pendeteksi wajah dan sistem keaamanan dengan mengikuti sebuah objek dan berbagai sistem teknologi yang lainnya. Salah satu sistem keamanan yang dikembangkan dan mengalami perkembangan pesat adalah sistem yang menggunakan identifikasi pengenalan objek, pada sistem ini program menganali objek yang tertangkap oleh kamera dengan kecepatan dan akurasi yang tinggi.

Penggunaan sistem pengenalan objek untuk saat ini sudah banyak dikembangkan dalam berbagai bidang, seperti pengenalan objek untuk mendeteksi kualitas tanaman pada lahan pertanian , penggunaan dalam sistem keamanan , dan dalam berbagai hal. Sistem pengenalan objek ini juga dapat digunakan untuk mengetahui seberapa banyak objek seperti objek manusia yang tertangkap oleh kamera pada suatu tempat, hal ini dapat bermanfaat untuk membatasi jumlah pengunjung atau sistem monitoring untuk survey seberapa banyak jumlah orang yang berkunjung dalam suatu waktu.

Didalam proses pengenalan objek dibutuhkan sebuah sistem perangkat keras yang memiliki spesifikasi yang tinggi , terutama dalam hal microprosessor

pada perangkat tersebut. Microprosessor merupakan sebuah chip IC yang menghubungkan fungsi inti dari central processing unit (CPU) sebuah komputer atau pusat sistem yang melakukan pemrosesan data pada komputer. Mikroprosesor dapat diprogram untuk menerima data digital sebagai masukan yang akan diproses sesuai dengan program yang diinstruksikan dan akan dikeluarkan sebagai sebuah data hasil. Mikroprosessor melakukan pembacaan program secara berurutan sehingga dapat berjalan sesuai keinginan pengguna atau programmer. Terdapat berbagai macam Bahasa pemrogramanan yang dapat digunakan untuk memberikan printah sehingga dapat dijalankan mikroprosessor seperti Bahasa pemrograman VHDL , Verilog, dan Python dll.

Algoritma Yolo merupakan algoritma deep learning yang dapat digunakan untuk mendeteksi suatu objek dengan menggunakan pendekatan yang berbeda dengan algoritma lainnya, dalam algoritma Yolo digunakan sebuah jaringan syaraf tunggal pada keseluruhan citra. Untuk mendeteksi sebuah objek YOLO melakukan pendekatan dimana YOLO membagi citra dalam region/grid dengan ukuran sxs. Grid-grid tersebut memiliki tugas untuk mendeteksi objek. Disetiap grid juga akan diprediksi bounding box beserta nilai confidence. Nilai confidence akan menunjuk seberapa akurasi bounding box dalam menentukan sebuah objek. Dalam setiap bounding box terdapat 5 nilai informasi yaitu x,y,w,h dan c. nilai x dan y merupakan koridnat dari titik tengah bounding box yang diperediksi, untuk nilai w dan h merupakan rasio dari ukuran lebar dan tinggi relative terhadap grid, dan c merupakan nilai dari confidence bounding box. Didalam algoritma YOLO, dalam setiap grid akan memprediksi nilai class probabilitas jika diprediksi terdapat objek didalamnya. Pada saat pengujian, algoritma YOLO akan mengkalikan nilai class probability dengan nilai confidence dari bounding box. Sehingga bisa menghasilkan nilai confidence kelas secara spesifik pada tiap bounding box. Selain YOLO terdapat berbagai macam metode yang dapat digunakan dalam pengenalan objek pada sistem citra digital seperti SSD (Single Shot Detection) , Haar case dan lain sebagainya.

Penelitian mengenai citra digital sudah banyak dilakukan seperti penelitian yang dilakukan oleh Bima Putra Gusti Pamungkas dkk (2021) untuk mengetahui performa dari YOLOv3 dan YOLO V2 dimana dalam penelitian ini YOLO v3 mendapatkan hasil yang lebih akurat mendekati nilai 1 dibandingkan dengan YOLOv2 dalam mendeteksi objek manusia, akan tetapi waktu yang dibutuhkan dalam menjalankan program dengan menggunakan YOLOv3 lebih lama jika dibandingkan dengan YOLOv2 [1]. Dipenelitian selanjutnya dilakukan oleh Moewardi R (2020) penelitian mengenai seberapa akurat dan lancar pendeteksian objek dimana hasil yang didapatkan bahwa pendeteksian wajah berjalan dengan lancar dengan menggunakan metode Haar Case [2] : Berikutnya penelitian yang dilakuka oleh Hidayatulloh, M. S. (2021) untuk melakukan pengenalan wajah dengan menggunakan metode YOLO akan didapatkaan hasil penurunan performa pada vidio yang digunakan dimana FPS yang didapatkan menurun menjadi 10 FPS, akan tetapi dalam proses pengenalan objek dengan foto , Algoritma yolo dan perangkat yang digunakan dapat mendeteksi objek dengan akurasi yang tinggi dari berbagai sudut pandang kamera [3].

Sejauh ini penerapan pengenalan objek dengan menggunakan pengolahan citra digital dengan menggunakan metode YOLO membutuhkan perangakat keras dengan spesifikasi yang tinggi terutama dibagian Arm Prosessornya, sehingga dalam pembuatannya dibutuhkan biaya yang terbilang cukup mahal, perangkat dengan spesifikasi tinggi ini bertujuan untuk mendapatkan hasil yang optimal pada saat melakukan pelatihan pada *dataset* yang telah dikumpulkan sehingga mendapatkan nilai akurasi yang tinggi pada saat penerapan, selain itu, penggunaan Arm Prosessor dengan spesifikasi yang tinggi dapat menghasilkan pengolahan citra digital pada vidio tidak mengalami penurunan kualitas seperti penurunan fps dari vidio tersebut, baik secara real time ataupun memasukkan hasil rekaman vidio secara manual.

Oleh karena itu untuk membuat sebuah perangkat yang dapat digunakan untuk mendeteksi objek dan mengetahui seberapa akurat perangkat tersebut mendeteksi objek dengan biaya yang terbilang terjangakau penulis mencoba menggunakan perangkat kamera dan Arm Prosessor dari NanoPi M4V2 yang dimana memiliki spesifikasi menengah keatas dan memiliki harga terbilang terjangkau selain itu NanoPi M4V2 juga sudah dapat dihubungkan kesebuah layar yang dapat menampilkan hasil dari kamera yang telah dilakukan pemerosesan pada mikroprosessor sehingga dapat menampilakan gambar dari pendeteksian objek , nilai akurasi dan besaran fps (frame per second) yang didapat dari kamera. Maka dari itu penulis membuat perangkat yang dapat mengidenttifikasi objek secara real time maupun masukan vidio dengan judul : implementasi proses citra digital objek manusia dengan metode YOLOv4 tinny menggunakan NanoPi M4V2.

## **Rumusan Masalah**

Bedasarkan dari latar belakang masalah, maka rumusan masalah dari perancangan alat ini adalah :

1. Dapatkah dibuat alat “Implementasi proses citra digital objek manusia dengan YOLOv4 tinny menggunakan NanoPi M4V2” ?
2. Apa yang akan digunakan untuk melakukan perekaman objek ?
3. Dapatkah NanoPi M4V2 melakukan pengolahan citra digital ?
4. Apa metode yang akan digunakan untuk mendeteksi objek pada citra digital ?
5. Bagaimana sistem pemantauan yang akan digunakan ?

## **Tujuan**

Tujuan dari pembuatan tugas akhir ini adaah :

1. Implementasi proses citra digital objek manusia dengan YOLOv4 tinny menggunakan NanoPi M4V2
2. Dapat mengetahui cara penggunaan kamera yang dihubungkan ke NanoPi M4V2 sehingga dapat melakukan perekaman objek
3. Dapat memahami penggunaan Arm Prosessor pada NanoPi M4V2 untuk melakukan pegolahan citra digital
4. Dapat menganalisa hasil pengolahan citra digital dengan menggunakan metode YOLOv4 tiny
5. Dapat memahami sistem pemantauan dengan menggunakan peragkat kamera dan Arm Prosessor NanoPi M4V2 dengan menggunakan Metode YOLOv4 tiny

## **Batasan Masalah**

Adapun Batasan masalah dari tugas akhir ini adalah :

1. Mikrokontroler yang digunakan dalam tugas akhir ini adalah NanoPi M4V2
2. Metode yang digunakan adalah metode YOLOv4 tinny
3. Menggunakan library open cv
4. Objek yang dideteksi adalah objek manusia
5. Bahasa pemrograman yang digunakan adalah bahasa pemrograman Python
6. Menggunakan kamera sebagai perekam objek
7. Pendeteksian objek dilakukan secara real time dan memasukan vidio hasil rekaman
8. Nilai yang ditampilkan berupa tingkat akurasi dan kualitas fps dari vidio yang telah diolah

## **Metodelogi Penelitian**

Adapun metodeogi yang digunakan pada tugas akhir ini adalah

1. Studi Literatur
2. Pencarian dan pengumpulan data
3. Pencarian dan pengkajian teori mengenai pembutan perangkat beserta cara kerjanya dari berbagai literatur serta sumber yang bermacam-macam seperti buku, internet, jurnal.
4. Pengumpulan data-data dan spesifikasi sistem yang dipakai untuk pembuatan perangkat sebagai pendukung sistem.
5. Analisa Masalah

Melakukan analisa dari teori yang telah didapat dengan bermacam macam sumber sehingga mendapatkan hasil yang semaksimal mungkin

1. Perancangan dan penambahan perangkat pendukung.

Pembuatan rancangan-rancangan kemudian mengimplementasikan rancangan tersebut ke dalam suatu perangkat pendukung.

1. Simulasi Sistem

Berdasarkan standar yang ada, tahap selanjutnya adalah melakukan simulasi sistem untuk melihat kinerja sistem tersebut.

1. Pengujian dan Perbaikan Sistem

Jika sistem telah berjalan, maka didapat keberhasilan maupun ketidakberhasilan dari simulasi sistem tersebut, sehingga dilakukan perbaikan sistem jika didapati sistem tersebut belum berjalan sesuai yang diharapkan. Semua tahapan dilakukan melalui tahap bimbingan (konseling) dengan pembimbing.

1. Kesimpulan

Penarikan kesimpulan harus didasarkan hal-hal yang relevan dari isi evaluasi dan juga tujuan penelitian.

## **Sistematika Penulisan**

Penulisan laporan akhir ini memiliki peranan sangat penting agar pembaca dapat mudah memahami terhadap isi yang terkandung didalamnya. Untuk mempermudah sistematika penulisan, penulis membagi dalam beberapa bab pembahasan dengan urutan sebagai berikut :

**BAB I PENDAHULUAN**

Merupakan bab yang berisi tentang gambaran umum dari permasalahan yang akan dibahas. Dalam pendahuluan ini terdiri dari enam sub bab, yaitu latar belakang masalah, perumusan masalah, tujuan penelitian, pembatasan masalah, metodologi penelitian, dan sistematika penulisan.

**BAB II LANDASAN TEORI**

Dalam landasan teori akan dijelaskan tentang tinjauan Pustaka dari penelitian sebelumnya, pengertian Monitoring, Vento, Sensor suhu, sensor kelembaban dan dasar teori – teori lainnya.

**BAB III METODOLOGI PENELITIAN**

Dalam bab ini berisi blok diagram sistem, flowchart, skema penelitian atau percobaan yang dilakukan untuk memperoleh hasil dan mencapai tujuan.

**BAB IV ANALISIS DATA DAN PEMBAHASAN**

Bab ini berisi tentang gambaran umum dari subyek penelitian, diskripsi data, analisa data, dan pembahasannya.

**BAB V PENUTUP**

Bab penutup berisi kesimpulan, serta saran-saran terhadap pengembangan sistem penelitian ini yang perlu untuk disampaikan.

# **BAB II TINJAUAN PUSTAKA DA LANDASAN TEORI**

## **Tinjauan Pustaka**

Setelah penulis melakukan telaah terhadap penelitian, ada beberapa penelitian yang berkaitan dengan penelitian yang penulis lakukan. Penelitan – penelitian yang berhasil penulis temukan masih jarang yang melakukan penelitian menggunakan Mini PC seperti NanoPi M4v4, sebagai Arm Prosessor yang digunakan untuk melakukan pengolahn citra digital. Peneltiian yang berhasil penulis temukan kebanyakan masih menggunakan perangkat keras berupa komputer ataupun laptop. Pada penelitian ini penulis membuat perangkat untuk melakukan implementasi citra digital dengan menggunakan NanoPi M4v4.

Penelitian pertama dilakukan oleh B.P.G. Pamungkas, (2021), Yang berjudul “DETEKSI DAN MENGHITUNG MANUSIA MENGGUNAKAN YOLO-CNN”. Penelitian ini memanfaatkan cctv pada lift yang bertujuan untuk mengetahui jumlah manusia yang tertangkap kamera. Sistem yang digunakan adalah memanfaatkan pengolahan citra digital dengan mendeteksi objek manusia yang kemudian akan menampilkan jumlah objek yang terdapat dalam lift dengan menggunakan YOLOv2 dan YOLOv3. Peneliti mencoba melakukan perbandingan hasil dari Yolo v2 dan Yolo V3. Hasil dari penelitian ini didapatkan kesimpulan berupa performa dari YOLOv3 mendapatkan hasil yang lebih akurat mendekati nilai 1 dibandingkan dengan YOLOv2 dalam mendeteksi objek manusia, akan tetapi waktu yang dibutuhkan dalam menjalankan program dengan menggunakan

YOLOv3 lebih lama jika dibandingkan dengan YOLOv2.

Penelitian kedua oleh J. Huang, (2021), yang berjudul “IMPROVE YOLOv3 MODEL FOR MINIATURE CAMERA DETECTION”. Tujuan dari penelitian ini mengetahui kemampuan dalam pendeteksian objek yang kecil dengan menggunakan model YOLOv3 yang sudah diitingkantkan yaitu YOLOv3-4L algoritma deteksi, dimana dalam algoritma ini memiliki banyak skala prediksi dan memiliki jaringan yang mendalam, selain itu dalam penelitian ini penulis memperkenalkan kamera miniature real time sistem deteksi (ALDS) dan YOLOv3-4L algoritma dengan hasil RCDS berfungsi dengan baik. Dalam penelitian ini selain membandingkan antara Yolov3 dengan YOLOv3-4L peneliti juga melakukan perbandingan dengan Faster R-CNN dan SSD. Dari hasil percobaan didapatkan hasil bahwa akurasi dari YOLOv3-L yang sudah ditingkatkan mendapatkan akurasi yang lebih tinggi dibandingkan dengan YOLOv3 versi awal , Faster CNN dan SSD.

Penelitian ketiga dilakukan oleh Muwardi, R., (2020). Dengan judul “Research and Design of Fast Special Human Face Recognition Sistem” Tujuan dari penelitian ini adalah memanfaatkan multiprosesor dengan arsitektur yang berbeda sehingga analisis bisa dilakukan dengan menjalankan perbandingan secara langsung dengan konfigurasi awal. Untuk meningkatkan deteksi wajah pada penelitian ini digunakan ARM Architecture, dengan membuat perangkat lunak menggunakan ARM Device (Nanopi M4V2), Bahasa program Python, Node Js dan OpenCV Library. Detection rate dan Tim dari metode pengenalan wajah didapatkan sebagai berikut :

Tabel 2.1 Hasil dari percobaan

(Muwardi, R., dkk, 2020)

|  |  |  |
| --- | --- | --- |
| Metode Pengenalan Wajah | Nilai Deteksi | Waktu Deteksi |
| Warna Kulit | 66% | 0.410 s |
| Haar Classifier | 87% | 0.443 s |
| Karakteristik Wajah | 93% | 1.669 s |

Hasil dari penelitian ini adalah pendeteksi wajah berjalan dengan lancar pada NanoPi M4V2, tidak terdapat error pada saat proses, manejemen CPU yang digunakan masi dibawah 80% dari 100% pada saat menjalankan program

Penelitian keempat dilakukan oleh Hidayatulloh, M. S. (2021). Yang berjudul *“*SISTEM PENGENALAN WAJAH MENGGUNAKAN METODE *YOLO (You Only Look Once)*. Penelitian ini bertujuan untuk merancang sebuah perangkat yang dapat digunakan untuk mengidentifikasi wajah dengan menggunakan metode YOLO dan OpenCV. Penelitian ini dilakukan dengan jumlah dataset yang digunakan adalah 70 citra dengan format jpg. dikarenakan spesifikasi hardware yang tidak optimal, nilai FPS yang didapatkan maksimal 10 FPS. Spesifikasi dari hardware yang penulis gunakan :

* Asus A412DA
* Processore AMD Ryzen 5 3500U @2.1Ghz
* RAM 8 GB
* Graphics Display Radeon Vega
* SSD 512 GB

Kesimpulan dari hasil penelitian ini adalah untuk menjalankan proses pengenalan wajah yang optimal dan dengan FPS yang baik dibutuhkan hardware yang sangat tinggi. Metode YOLO memiliki tingkat akurasi yang tinggi dalam pendeteksi wajah dari berbagai sudut pandang kamera.

Penelitian kelima dilakukan oleh Agung, W. A. S. (2021). Yang berjudul “SISTEM PENGHITUNG PENGUNJUNG PUSAT PERBELANJAAN DAN DETEKSI MASKER PADA PANDEMI COVID-19 MENGGUNAKAN LIBRARY OPENCV DAN YOLO” . Tujuan dari penelitian ini adalah untuk mengetahui jumlah pengunjung yang terdapat pada pusat perbelanjaan dan mendeteksi penggunaan masker. Kesimpulan dari penelitian ini mendapatakn bahwa sistem mampu mendeteksi pada kelas objek masker , tanpa masker dan orang, dengan menggunakan Algoritma YOLO v3 tipe tiny pada raspberry pi 4 degan mAP 77,92%. Dikarenakan kemampuan dari komputasi sistem nilai FPS dari vidio yang didapat menurun menjadi 0.77 FPS sehingga hasil perhitungan jumlah pengunjung tidak optimal.

Penelitian keenam dilakukan oleh Tan, L., (2021). Yang berjudul Comparison of YOLO v3, Faster R-CNN, and SSD for Real-Time Pill Identification. Tujuan dari penelitian ini adalah untuk mengidentifikasi obat pil yang benar untuk diberikan ke pasien, pendeteksian object dengan menggunakan tiga buah model algoritma deteksi objek diantaranya : Faster R-CNN , Single shot multi Box Detector (SSD) , and You Only Look Once V3 (YOLO v3), untuk mendeteksi pil dan membandingkan performanya. Metode dalam penelitian ini , memperkenalkan basic dari tiga object detection models. Hasil dari penelitian ini didapatkan bahwa penelitian ini YOLO v3 lebih baik dalam kecepatan mendeteksi dalam menjaga MAP dan dapat digunakan dalam real time pill detection.spesifikasi perangkat yang digunakan : OS: Win10, GPU: NVIDIA GeForce GTX 1080Ti, CPU: Intel(R) Core(TM) i7-7700K CPU @ 4.20GHz.

Penelitian ketujuh oleh Li, M., (2020). Dengan judul Agricultural greenhouses detection in high-resolution satellite images based on convolutional neural networks: Comparison of faster R-CNN, YOLO v3 and SSD.  Dalam penelitian ini peneliti mencoba membandingkan tiga buah metode pendeteksi objek yakni YOLO v3 , Faster R-CNN dan SSD dalam mendeteksi pertanian dengan memanfaatkan gambar dari satelit yang beresolusi tinggi berbasis pada CNN, hal ini bertujuan untuk mendapatkan metode terbaik dalam mendeteksi kondisi dari sebuah pertanian, metode dari penelitian ini adalah membandingkan nilai MAP dan FPS yang didapat dari ketiga model pendekteksi object dengan 2 buah gambar satelit yang berbeda yaitu gambar dari satelit 1 (GF1) dan gambar dari satelit 2 (GF2) dari hasil penelitian ini didapatkan kesimpulan bahwa nilai :

Tabel 2.2 Hasil dari Perocobaan

(Li, M., dkk, 2020)

|  |  |  |  |
| --- | --- | --- | --- |
|  | Faster R-CNN | YOLO v3 | SSD |
| MAP (GF1 & GF2) | 86.0% | 90.4% | 84.9% |
| MAP (GF1) | 64.0% | 73.0% | 60.9% |
| MAP (GF2) | 88.3% | 93.2% | 87.9% |
| FPS | 12 | 73 | 35 |

Dari table diatas dapat disimpulkan bahwa YOLO v3 adalah tipe model deteksi objek yang terbaik untuk dilakukan mendeteksi pertanian secara real time menggunakan gambar dari satelit.

Penelitian kedelapan oleh Gupta, P., (2021). People detection and counting using YOLOv3 and SSD models. *Materials Today: Proceedings.* Tujuan dari penelitian ini adalah untuk mendapatkan metode untuk mendeteksi dan menghitung objek yang terbaik pada gambar dan vidio dengan membandingkan nilai dari presisi , recall dan F1. Dari hasil penelitian ini didapatkan bahwa SSD mendapat hasil yang terbaik dalam hal Presisi, recall dan F1. Dengan SSD mendapatakn nilai presisi : 95 , recall : 84 , dan F1 : 90.

Penelitin kesembilan oleh Iyer, R., (2021). Yang berjudul “Comparison of YOLOv3, YOLOv5s and MobileNet-SSD V2 for Real-Time Mask Detection”. Penelitian ini bertujuan untuk membandingkan metode terbaik dalam mendeteksi penggunaan masker yang benar , salah dan tidak menggunakan masker. Dalam metode ini peneliti membandingkan algoritma antara YOLO v3 , YOLOv5s dan MobileNet SSD V2, dalam hal akurasi dan kaulitas FPS dari vidio yang diproses pada Real time. Hasil dari penelitian ini didapatakan seperti table dibawah ini ;

Tabel 2.3 Hasil dari Percobaan

(Iyer, R., 2021)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Model | mAP(%) | FPS pada 3 perangkat yang berbeda | | |
|  |  | Nvidia TeslaT4 | Nvidia GTS 1660Ti | Jetson Nano |
| YOLO v3 | 54.3 | 80 | 21 | 8 |
| YOLO v5 | 37.6 | 100 | 28 | 15 |
| Mobile-Net SSD | 33.7 | 94 | 26 | 15 |

Dari hasil penelitian tersebut dapat disimpulkan bahwa ketiga medel tersebut memiliki karakteristik masing masing, masing masing model berhasil mendeteksi objek bermasker.YOLO v5 memiliki hasil yang lebih optimal untuk dibuat real time untuk kecepatan dan accuracy.

Penelitian kesepuluh oleh Nepal, U., (2022). Dengan judul “Comparing YOLOv3, YOLOv4 and YOLOv5 for Autonomous Landing Spot Detection in Faulty UAVs”. Yang diupload pada *Sensors*, *22*(2), 464. Dalam penelitian ini,peneliti mencoba untuk menbandingkan tiga buah algoritma untuk mendeteksi tempat terbaik untuk landing pesawat autonomous UAVs, dalam penelitian ini didapatakan bawah YOLOv5 mendapatkan nilai terbaik dibandingkan YOLOv3 dan YOLOv4.

Penelitian kesebelas dilakukan oleh Y. Wang, (2018) penelitian yang dilakuakn untuk kendaraan trajectory data didalam kondisi lalu lintas, mereka melakukan penelitian mengenai pesawat tanpa awak dan mengambil vidio menggunakan vidio dari pesawat tanpa awak. Dalam penelitian ini menggunakan metode pendeteksi objek dengan dua model objek detection yaitu CNN model dan YOLO v3 untuk mendeteksi kendaraan, hasil dari percobaan sangat akurasi dan wise.

Penelitian kedua belas T. Stahl, (2019) melakukan penelitian mengenai Batasan pada CNN, penulis mendesai model dengan menggunakna YOLO v3 , yang coba di implementasikan untuk kecepatan deteksi pada PASCAL VOC dataset. Hasil dari penelitian ini, mereka DRAM yang merek gunakan mengalami kelebihan konsumsi.

Penelitian ketiga belas Z. Dai, (2019) melakukan penelitian mengenai deteksi manusia , Dataset yang mereka gunakan PDCS dataset, dengan memiliki vidio lebih dari 4500. Hasil dari objek deteksi dengan mengguankan model manusia, penulis mendapatka hasil FPS pada vidio 45fps dengan 1.7 GHz Prosessor.

Penelitian keempat belas R. Feng, (2020) melakukan penelitian untuk pendeteksian gambar dan deteksi objek, mereka melakukan penelitian dengan menggunakan HAAR dan juga YOLO v3 algoritma yang akan dikomparasikan, mereka melakuakan deteksi dengan objek palsu yang dibuat dengan menggunakan silicon untuk peralatan yang ada dilingkungan.

Penelitian kelima belas S.sun., (2019) melakukan penelitian untuk mendeteksi kendaraan dan roda. Batasan dalam CNN untuk pendeteksian ini telah dihilangkan dalam penelitian in, mereka menghasilkan novel optimized SSD algoritma. Penelitian ini menggunakan dataset PASCAL VOC 2007. Untuk mAP Mereka mendapatkan 1.5% lebih tinggi dari pada SSD.

Tabel 2.4 Literatur Riview

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Judul** | **Peneliti** | **Spesifikasi Sistem** | |
| ***Hardware*** | ***Interface*** |
| 1 | Deteksi Dan Menghitung Manusia Menggunakan Yolo-Cnn | 1. B.P.G. Pamungkas 2. Budi Nugroho 3. Fetty Anggraeny | * Laptop * Webcam | Simulasi |
| 2 | Improve Yolov3 Model For Miniature Camera Detection | 1. Jiahao Huang 2. Haiyang Zhang 3. Lin Wang 4. Zilong Zhang 5. Changming Zhao | * Komputer * Webcam | Simulasi |
| 3 | Research and Design of Fast Special Human Face Recognition Sistem | 1. Muwardi, R 2. Qin, H 3. Gao, H 4. Ghifarsyam, H. U 5. Hajar, M. H. I 6. Yunita, M | * Nanopi M4V2 * Kamera * Layar | *Prototype* |
| 4 | Sistem Pengenalan Wajah Menggunakan Metode *Yolo (You Only Look Once)* | 1. Hidayatulloh, M. | * Laptop | *Simulasi* |
| 5 | Sistem Penghitung Pengunjung Pusat Perbelanjaan Dan Deteksi Masker Pada Pandemi Covid-19 Menggunakan Library Opencv Dan Yolo | 1. Agung, W. A. S | * Komputer * Dataset * Buzzer | *Prototype* |
| 6 | Comparison of YOLO v3, Faster R-CNN, and SSD for Real-Time Pill Identification | 1. Tan, L Varsha 2. Huangfu, T 3. Wu, L 4. Chen, W | * Laptop | Simulasi |
| 7 | Agricultural greenhouses detection in high-resolution satellite images based on convolutional neural networks: Comparison of faster R-CNN, YOLO v3 and SSD | 1. Li, M 2. Zhang, Z 3. Lei, L 4. Wang, X 5. Guo, X | * Komputer * Dataset | *Prototype* |
| 8 | People detection and counting using YOLOv3 and SSD models. *Materials Today: Proceedings* | 1. Gupta, P., 2. Sharma, V. 3. Varma, S | * Laptop * Dataset | Simulasi |
| 9 | Comparison of YOLOv3, YOLOv5s and MobileNet-SSD V2 for Real-Time Mask Detection | 1. Iyer, R., 2. Shashikant Ringe, P., 3. Varadharajan Iyer, R. 4. Prabhulal Bhensdadiya, K | * Laptop * Dataset | Simulasi |
| 10 | Comparing YOLOv3, YOLOv4 and YOLOv5 for Autonomous Landing Spot Detection in Faulty UAVs | 1. Nepal, U. 2. Eslamiat, H | * Komputer * Drone | Simulasi |
| 11 | Manifold-Based Visual Object Counting | 1. Y. Wang 2. Y. Zou 3. W. Wang | * Laptop * Kamera | Simulasi |
| 12 | Divide and Count: Generic Object Counting by Image Divisions | 1. T. Stahl, S.L. 2. Pintea, J.C. 3. Van Gemert | * *Handphone* * Dataset | Simulasi |
| 13 | Vidio-based vehicle 361 counting framework | 1. Z. Dai, H. Song, 2. X. Wang, 3. Y. Fang, 4. X.u. Yun, 5. Z. Zhang 6. H. Li | * Laptop * Dataset | Prototype |
| 14 | Mixed Road User Trajectory Extraction from Moving Aerial Vidios Based on Convolution Neural Network Detection | 1. R. Feng, 2. C. Fan, 3. Z. Li, 4. X. Chen | * Laptop * Dataset * Drone | Prototype |
| 15 | Benchmark Data and Method for Real-Time People Counting in Cluttered Scenes Using Depth Sensors | 1. S. Sun, 2. N. Akhtar, 3. H. Song, 4. C. Zhang, 5. J. Li, 6. A. Mian | * Komputer * Kamera * Display | *Prototype* |

## **Arm Processor**

Arm prosessor merupakan salah satu keluarga dari CPU (*Komputer Processing Unit*) yang berbabis pada RISC mesin yang dikembangkan oleh ARM Limited. Arm arsitektektur menggunakan prosessor 32 bit dan 64bit RISC dengan multi coreprocessor. Prosesor RISC diracang untuk melakukan sedikit jenis intruksi komputer sehingga dapat beropesari pada kecepatan tinggi, dapat melakukan banyak instruksi dalam perdetik (MIPS). Dikarenakan menggunakan sedikit instruksi komputer dimana menghilangkan bagian yang tidak diperlukan sehingga sistem ini memberikan kinerja yang tinggi. Pada perangkat komputer ARM banyak digunakan pada keluarga X82, akan tetapi dikarenakan ARM sangat cocok digunakan pada aplikasi yang memiliki daya yang rendah. Sehingga kebanyak ARM banyak digunakna pada perankgat semart phone dan sistem embedded. Untuk penggunaan pada komputer ARM banyka digunakan pada perangkat mini pc karena memiliki sistem yang sederhana dan daya yang rendah. ARM memiliki 3 arsitektur yang berbeda dimana masing masing memiliki penggunaan yang berbeda, berikut adalah seri dari arsitektur ARM prosesor:

* 1. ARM arsitektur A sebagai prosesor aplikasi
  2. ARM cortex R sebagai prosessor real time
  3. ARM cortec M sebagai prosessor mikrokontroller

Arm memiliki kelebihan diantaranya merupakan arsitektur prosesor yang banyak digunakan didunia sehingga untuk pengembangannya akan selalu diperbaruhi,dan dapat diimplementasikan pada teknologi baru selain itu arsitektur ARM menggunakan seluruh teknologi prosesor , mulai dari penggunaan , aplikasi, mikrokontroler, dan aplikasi , dikarenakan penggunaan Arm prosessor yang banyak sehingga pengguna akan mendaatan bantuan yang lebih mudah apabila terdapat kendala didalam penggunaannya.

## **Nanopi M4V2**

Nanopi M4V2 adalah RK339 yang merupakan sebuah prosesor yang berbasis pada perangkat ARM. Nanopi M4V2 memiliki kesamaan dengan raspberry pi3. Nanopi M4V2 merupakan salah satu jenis mini pc yang dapat digunakan dalam menjalankan berbagai macam perangkat lunak. Dikarenakan memiki arm prosessor yang mumumpuni perangkat ini banyak digunakan dalam pembuatan sistem kecerdasaan buatan, pembelajaran mesin ataupun pengolahan data.

Nanopi M4V2 dapat mendukung dua buah kamera didalam penggunaannya melalui sambungan usb yangdimiliki dan telah mendukung tampilan 4K.

Nanopi M4V2 memiliki spesifikasi yang mumpuni dalam penggunaannya, berikut spesifikasi dari perangkat ini :

Tabel 2.5 Spesifikasi dari Nanopi M4V2

|  |  |
| --- | --- |
| CPU | Dual-core Cortext-A72(up to 2.0GHz) +Quad core Cortex A53 (up to 1.5 GHz) |
| RAM | Dual channel 4GB LPDDR4 |
| USB | 4 USB 3.0 tipe A, Ports USB type C : support USB 2.0 OTG dan power input |
| Wifi  Bluethooth | 802.11a/b/g/n/ac,  Bluetooth 4.1  Wi-Fi and Bluetooth combo module, 2x2 MIMO, dual antenna interface |
| Network | Native Gigabit Ethernet |
| Audio Input / output port | Audio Out: 3.5mm Dual channel headphone jack, or HDMI Audio In: one microphone input interface |
| Vidio input  Vidio output | one or two 4-Lane MIPI-CSI, dual ISP, up to 13MPix/ supports simultaneous input of dual camera data  HDMI: HDMI 2.0a, supports 4K@60Hz，HDCP 1.4/2.2, one 4-Lane MIPI-DSI |
| Storage | no Onboard eMMC, but has an eMMC socket |
| Micro SD slot | 1 slot |
| GPIO | * 40Pin GPIO Extension ports: * 3 X 3V/1.8V I2C, up to 1 x 3V UART, 1 X 3V SPI, 1 x SPDIF\_TX, up to 8 x 3V GPIOs * 1 x 1.8V 8 channels I2S * 24Pin Extension ports: * 2 independent native USB 2.0 Host * PCIe x2 * PWM x1, PowerKey |
| RTC | 2 Pin 1.27/1.25mm RTC battery input connector |
| Power | DC 5V/3A |
| Working Temperature | -20℃ to 70℃ |
| Software BSP Version | Linux-4.4-LTS U-boot-2014.10 |
| OS support | Android 8.1 Android 10 Lubuntu 16.04 (32-bit) FriendlyCore 18.04 (64-bit) FriendlyDesktop 18.04 (64-bit) FriendlyWrt 19.07.1 (64-bit) |
| Debug Serial Port / UART0 | one Debug UART, 4 Pin 2.54mm header, 3V level, 1500000bps |

A close-up of a computer chip

Description automatically generated with low confidence

Gambar 2.1 Nanopi V4M2

## **Pengolahan Citra Digital**

Teknik pengolahan citra digitalmengimplementasikan fungsi transformasi pada gambar dan vidio. Hasil keluaran yang didapatkan berupa gambar atau data yang telah disempurnakan. Pemrosesan gambar dan vidio banyak digunakan pada aplikasi streamer [2]. Proses pemrosesan pada pengolahan citra digital didasarkan atas kecepatan dan proses pengolahan data didasarkan atas spesifikasi perangkat keras yang digunaknan dan kompleksifitas dari algoritma program yang digunakan, sehingga untuk mendapatkan hasil yang optimal maka perangkat yang tertanam dan algoritma yang digunakan harus optimal. Citra tersusun oleh sekumpulan piksel yang memiliki kordinat (x,y) dan amplitude (x,y). untuk menunjukan posisi piksel mengginakan kordinat (x,y) dan amplitude (x,y) menunjukkan nilai untuk sebuah intensitas warna citra. Berdasarkan pada kombinasi warna pada piksel, citra dibagi menjadi 3 jenis yaitu citra RGB, citra grayscale, dan citra biner. Citra RGB merupakan citra yang terkombinasi dari tiga nilai intensitas pada kanal R , G, dan B. banyaknya nilai kombinasi pada pikes yang memungkina pada citra RGB 24bit adalah 256 x 256 x 256 = 16.777.216. berikut trepresentasi nilai dari intensitas piksel dengan kombinasi warna R, G, dan B.

Citra greyscale adalah citra yang nilai intensitas pikselnya berdasarkan pada derajat keabuan. Pada citragrayscale 8-bit, derajat pada warna hitam sampai dengan putih dibagi dengan 256 dari derajat keabuan , dengan nilai hitam yang sempurna diaplikasikan dengan nilai 0 dan warna putih sempurna diaplikasikan dengan niali 255. Gambar dengan kualitas RGB dapat di konversi menjadi gambar dengan kualitas greyscal sehingga hanya dapat menghasilkan satu kanal warna. Berikut persamaan yang biasa digunakan untuk mengkonversi gambar dengan kualitas RGB 24bit menjadi kualitas greyscale 8bit.

*Grayscale = 0.2989\*R + 0.5870\*G + 0.1140\*B*

Jenis citra yang ketiga adalah citra biner. Citra biner adalah citra yang pikselnya memiliki kedalaman bit sebesar 1 bit sehingga hanya memiliki dua nilai intensitas warna yaitu 0 (hitam) dan 1 (putih). Citra grayscale dapat dikonversi menjadi citra biner melalui proses thresholding. Dalam proses thresholding, dibutuhkan suatu nilai threshold sebagai nilai pembatas konversi. Nilai intensitas piksel yang lebih besar atau sama dengan nilai threshold akan dikonversi menjadi 1. Sedangkan nilai intensitas piksel yang kurang dari nilai threshold akan dikonversi menjadi 0. Misalnya nilai threshold yang digunakan adalah 128, maka piksel yang mempunyai intensitas kurang dari 128 akan diubah menjadi 0 (hitam) dan yang lebih dari atau sama dengan 128 akan diubah menjadi 1 (putih).

## **YoloV4-Tiny**

Algoritma Yolo merupakan algoritma deep learning yang dapat digunakan untuk mendeteksi suatu objek dengan menggunakan pendekatan yang berbeda dengan algoritma lainnya yang dikembangkan oleh Joseph redmo. Yolo memiliki berbagai macam versi mulai dari viz, YOLO v2, YOLO v3, YOLO v4, dan yang terbaru YOLO v5. Didalam YOLO v4 terdapat versi terbaru dimana disebut YOLOv4 Tiny. YOLOv4 tiny adalah hasil kompresi dari YOLOv4 yang dimana memiliki jaringan yang lebih sederhana dan terdapat pengurangan parameter yang tidak diperlukan sehingga layak untuk dikembangkan pada perangkat seluler sederhana. Yolov4 tiny dapat digunakan untuk mendaptkan hasil pelatihan dan deteksi objek yang lebih cepat. Untuk pendeteksian objek secara langsung yolov4 tiny merupakan pilihan yang terbaik dikarenakan waktu pendekteksian yang lebih cepat.

Sistem pendeteksian pada algoritma yolov4 tiny sama dengan algoritma Yolo pada umumnya yang dimana menggunakan sebuah jaringan syaraf tunggal pada keseluruhan citra. Dalam pendeteksian sebuah Objek Yolo menggunakan beberapa tahapan, Yaitu :

1. Membagi citra dalam region/grid dengan ukuran sxs. Grid-grid tersebut memiliki tugas untuk mendeteksi objek. Disetiap grid juga akan diprediksi bounding box beserta nilai confidence. Nilai confidence akan menunjuk seberapa akurasi bounding box dalam menentukan sebuah objek. Perhitungan nilai confidence didapatkan dari persamaan berikut :

(2.1)

Dimana :

P(Class) adalah sebuah probabilitas suatu objek yang muncul dalam suatu grid dan rasio dari intersection over union (IOU) diantara kotak prediksi dan kotak ground truth.

Pred merupakan luas area dalam kotak prediksi dan,

Truth merupakan area dalam ground Truth. Semakin tinggi nilai IOU maka makin tinggi akurasi dari pendeteksinya.

1. Dalam setiap bounding box terdapat 5 nilai informasi yaitu x,y,w,h dan c. nilai x dan y merupakan kordinat dari titik tengah bounding box yang diperdiksi, untuk nilai w dan h merupakan rasio dari ukuran lebar dan tinggi relative terhadap grid, dan c merupakan nilai dari confidence bounding box.
2. Didalam algoritma YOLO, dalam setiap grid akan memprediksi nilai class probabilitas jika diprediksi terdapat objek didalamnya. Pada saat pengujian, algoritma YOLO akan mengkalikan nilai class probability dengan nilai confidence dari bounding box. Sehingga bisa menghasilkan nilai confidence kelas secara spesifik pada tiap bounding box.

## **Python**

Python merupakan Bahasa pemrograman tingkat tinggi yang sangat mudah dipahami oleh manusia, Bahasa pemrograman ini dikembangkan oleh programmer asal belanda Guido Van Rossum pada tahun 1989 yang kemudian diperkenalkan secara resmi pada tahun 1991. Bahasa pemrograman python sangat banyak digunakan seperti dalam hal pembelajaran mesin, membangun situs web dan dalam pembuatan sebuah aplikasi maupun melakukan pengujian sebuah aplikasi. Karena kemudahan dalam penggunaannya Bahasa pemrograman python dapat digunakan oleh seseorang yang memiliki basic programmer ataupun yang tidak memiliki basic programmer. Penggunaan python dalam kecerdasan buatan maupun pengolahan citra digital akhir – akhir ini sangat sering digunakan. Python menyediakan library yang sangat banyak yang dapat digunakan untuk melakukan pengolahan data maupun dalam pengolahan citra digital sehingga dapat memudahkan pengguna. Keunggulan dalam penggunaan python diantaranya :

1. Mudah dipelajari

Bahasa pemrograman python sangat mudah untuk dipelajri dan dipahai dikarenakan memiliki sintaks yang cukup sederhana

1. Mudah diaplikasikan

Dalam pengembangannya Bahasa python sangat mudah diaplikasikan karena memiliki banyak library yang dapat digunakan dalam berbagai hal, misalanya dalam pengolahan data, pembelajaran mesin, kecerdasan buatan dll.

1. Mendukung IoT

Dalam penggunaan dibidang *internet of things* Bahasa pemrograman python sangat mendukung dikarenakan banyak board yang menjalankan sistem dengan metode *internet of thing* menggunakan Bahasa python.

1. Open source

Bersifat open source dimana yang artinya Bahasa pemrograman ini dapat digunakan, dikembangkan dan dikomersikan secara gratis.

## **Open CV**

Open CV adalah sebuah perpustakaan terbuka yang dikembangkan oleh intel yang dimana pengelolaannya saat ini dikelola oleh Wollow Garage , Open CV berfokus untuk menyederhanakan programming dalam hal citra digital. Fitur yang terdapat didalam open cv diantaranya pengenalan wajah, deteksi wajah, pelacakan wajah, dan berbagai jenis kecerdasan buatan lainnya, selain itu open cv juga menyediakan berbagai algoritma sederhana terkait komputer vision untuk level API yang rendah. Pada awalnya open cv memerlukan perpustakaan dari intel image processing yang dimana kemudian hal tersebut dihilangkan sehingga menghasilkan open cv yang seperti saat ini,dimana perpustakaannya berdiri sendiri. Open cv mendukung multiplatform dimana dapat mendukung windows, macos, linus dan android.

Open cv memiliki berbagai macam fitur yang dapat dimanfaatkan , berikut ini adalah fitur utama pada open cv :

1. Komputer vision dan pengolahan citra digital dengan lebel API rendah maupun tinggi
2. Gambar dan vidio baik berupa input maupun output
3. Metode dalam pengembangangan AI maupun pembelajaran mesin
4. Modul Komputer vision dengan level tinggi dalam pengenalan wajah ,dan detek wajah
5. Metode untuk menghitungkan permodelan pada 3D yang dapat digunakan dalah melakukan pemetaan dan visualisais

## **Webcam**

Webcam merupakan sebuah kamera yang berukuran kecil yang dapat dihibungkan dengan perangkat komputer dengan menggunakan kabel usb. Webcam dapat digunakan untuk mengambil sebuah gambar atau vidio secara langsung. Dengan menggunakan sebuah perangkat lunak webcam dapat mengirimkan hasil dari tangkapan baik berupa vidio atau gambar melalui internet secara lagnsung ataupun tidak langsung.

Perangkat webcam, tidak memiliki penyimpanan sendiri seperti kamera digital ataupun kamera recorder yang pada umumnya, untuk proses penyimpanan hasil tangkapan dari webcam akan disimpan pada penyimpanan komputer sehingga webcam akan terus terhubung dengan perangkat komuter untuk penggunaannya.Dibawah ini merupakan contoh gambr webcam.

Graphical user interface, application

Description automatically generated

Gambar 2.2 Webcam

(Logitech.com)

## **Visual Studio Code**

Visual studio code merupakan sebuah perangkat lunak yang dapat digunakan untuk membaut atau menjalankan sebuah koding. Visual studio code dapat digunakan pada sitem operarsi windows, linux ataupun macos. Dalam penggunaannya visual studio dapat berjalan dengan lancar dan juga ringan pada saat melakukan pembacaan ataupun mengedit code yang bersumber dari berbagai macam Bahasa pemrograman, seperti : python, java script, node js, dan type script.

Visual studio code memiliki berbagai macam fitur yang dapat membantu dalam penggunaannya, sehingga programmer dapat membut sebuah program dengan mudah, berbagai fitur tersebut sepert , visual studio code dapat meyarankan penggunaan kata berdasarkan sintaks yang ditulis sehingga programmer dapat dengan mudah memilih sintaks yang sesuai dengan sintaks yang ditulis, selain itu visual studio dapat menjalankan *debugging* yang dapat digunakan untuk mencari tahu kesalahan didalam penulisan sintaks , sehingga kita dapat dengan mudah dalam mengedit , mengcomplie dan megekseuki code berulangkali. Dan selanjtunya visual studio code memiliki *extensens market* yang dapat digunakan untuk menambahkan fitur atau library dari pengembang yang lain.

A screenshot of a computer

Description automatically generated

Gambar 2.3 Visual studio code

## **SSD (Solid State Disk)**

SSD (*Solid state disk*) merupakan perangkat penyimpanan data yang menggunakan serangkaian IC sebagai memori penyimpanan dan pembacaan data, perangkat ini memiliki penyimpanan yang besar sehingga dapat dikatakan bahwa perangkat ini merupakan perangkat dengan versi lebih canggih dari USB *flash drive* dan perangkat ini lebih ringan jika dibandingkan dengagn *harddisk. Chip* pada SSD memiliki kecepatan yang lebih tinggi jika dibandingkan dengan *harddisk* maupun *flashdisk.* Proses pembacaan data pada ssd tidak menggunakan proses mekanik seperti halnya pada *harddisk*,pada ssd proses pembacaan data menggunakan rangkaian elektronik seperti ic (*integrated circuit*), *micro chip*, serta kompunen elektronik lainnya.

Memori yang digunakan pada SSD merupakan memori yang berbasis NAND dimana memori ini merupakan memori Non-Volatile. Penyimpanan data pada ssd dapat disimpan meskipun penghubung perangkat dimatikan dan ssd dapat menyimpan data aman hingga 200 tahun. Perangkat ssd tidak menimpulkan panas yang berlebih dikarenakan perangkat ini tidak menggunakan putaran pada piringan seperti halnya di harddisk selain itu tegangan yang digunakan juga tidak terlalu tinggi.

## **Layar**

Layar merupakan sebuah perangkat yang dapat digunakan untuk menampilkan sebuah teks gambar atau vidio secara visual. Layer merupakan perangakat yang dapat dihubungkan ke perankat komputer, *mini* PC, tablet, dan perangkat lainnya yang dimana berfungsi sebagai keluaran dari hasil proses pada perangkat proses. Layar memiliki resolusi yang berbeda – beda dalam menampilkan hasil proses. Ukuran diagonal pda monitor disebut dengan inchi dan resolusi merupakan lebar dan tinggi piksel yang akan menentukan kualitas dari gambar dan vidio yang dihasilkan. Layar memiliki berbagai macam jenis, berikut adalah jenis jenis dari layar :

* 1. *Chatode ray tube* (CRT)

Layar CRT bekerja dengan menggunakan tabung sinar katode, cara kerja pada layar ini dengan cara memancarkan sinarr electron ke titik – titik kecil pada layar.

* 1. *Liquid cristal display* (LCD)

Layar LCD menggunakan cairan kristal yang memancarkan sinar selanjutnya dipancarkan secara elektrik yang akan membentuk sebuah panel panel kecil yang datar. Sistem pencahayaan pada lcd menggunakan lampu neon berjenis fluorescent, ukuran LCD lebih kecil dari pada crt tetapi memiliki resolusi yang lebih tinggi.

* 1. *Light emiting diode* (LED)

Sistem kerja pada led memiliki kesamaan dengan LCD yang menjadi perbedaan terletak pada penggunakan teknologi pancaran dari LED backlight sebagai sumber pencahayaan.

## **Vidio**

Vidio adalah sebuah tampilan pada gambar yang bergerak yang tertangkap oleh kamera. Vidio dapat diaplikasikan pada perangkat televisi, layar yang terhubung dengan perangkat komputer, dan perangkat vidiotape maupun media non non komputer lainnya. Setiap frame tersebut dipresentasikan menggunakan signal listrik yang disebut dengan gelombang analog atau vidio komposit yang telah mempunyai komponenkomponen dalam vidio seperti warna, penerangan dan kesingkronan dari setiap gambar nya (Purnama, 2013).

Kualitas dari vidio yang ditampilkan bergantung dari kualitas perekam yang digunakan. Perpaduan antara suara dan gambar dapat membentuk sebuah karakter yang sama dengan objek aslinya. Alat yang termasuk kedalam kategori vidio adalah TV, Sound slide, film dan VCD (Sanaky, 2011). Keuntungan dari vidio adalah dapat menampilkan gambar yang bergerak beserta suara yang hasil tangkapan dari perangkat yang digunakan untuk merekam.

## ***Real time vidio***

*Real time* vidio adalah sebuah vidio yang ditampilkan secara langsung ke perangkat media yang dapat digunakan untuk menampilkan vidio. *Real time* vidio digunakan dalam berbagai hal seperti tampilan rekaman dari kamera cctv ke layar yang ada pada ruangan kontrol untuk proses keamanan, kualitas dari vidio yang ditampilkan tergantung dari perangkat yang digunakan untuk merekam. Vidio yang dihasilkan melalui *real time* vidio lebih bagus jika dibandingkan melalu *live streaming* dikarenakan vidio yang ditampilkan secara langsung tidak membutuhkan jaringan yang bagus, tetapi hanya tergantung dari kualitas perangkat dan kabel penghubung yang digunakan.

## ***Streaming vidio***

*Streaming vidio* adalah vidio yang ditampilkan secara langsung yang memanfaatkan jaringan internet dengan memanfaatkan aplikasi ataupun perangkat lainnya. Dikarenakan pengiriman vidio memanfaatkan jaringan internet maka kualitas dari vidio yang dihasilkan tergantung dari jaringan internet yang digunakan dan perangkat yang digunakan untuk melakukan perekamn. Seiring dengan kemajuan teknologi *streaming* vidio saat ini banyak digunakan untuk mendeteksi objek secara langsung yang kemudian dikirimkan ke perangkat secara jauh hal ini biasanya digunakan pada perangkat kamera yang terpasang dijalan untuk mendeteksi pelanggaran maupun pelaku kejahatan.

# **BAB III METODELOGI PENELITIAN**

## **Metodelogi penelitian**

Pada penelitian ini menggunakan metode penelitian eksperimen. Metode eksperimen ini adalah salah satu metode kuantitatif, digunakan terutama apabila penelitian ingin melakukan percoban untuk mencari pengaruh variabel independen/*treatmen*/ perlakuan tertentu terhadap variabel depenten/hasil/*output* dalam kondisi yang terkendalikan. (Dr. Sigioyono, 2019;110). Metode eksperimen yang digunakan pada penelitian ini adalah dengan pengumpulan data menggunakan teknik pengambilan gambar atau vidio menggunakan webcam atau penginputan vidio atau gambar yang sudah terekam yang selanjutnya dilakukan pemrosesan. Proses selanjutnya adalah melakukan pembelajaran mesin, Machine learning merupakan serangkaian teknik yang dapat membantu dalam menangani dan memprediksi data yang sangat besar dengan cara mempresentasikan data-data tersebut dengan algoritma pembelajaran (Danukusumo, 2017).

Tahapan selanjutnya data yang telah dikumpulkan akan dilatih sehingga perangkat yang kita gunakan dapat menentukan objek yang ada pada vidio. Pada tahapan ini menggunakan metode algoritma Yolov4 Tiny. Yolov4 Tiny merupakan salah satu bagian dari algoritma YOLO yang dapat diguakan untuk mendeteksi objek. Vidio atau gambar yang dimasukkan dilakukan Pendeteksian objek dimana gambar atau vidio yang dimasukkan tersebut lalu didapatkan vector kotak pembatas dan prediksi kelas dalam keluaran. Setelah pendeteksian objek selesai maka akan disamakan dengan objek yang ada di dataset yang telah dilatih yang selanjutnya akan ditampilkan pada Komputer *vision*.

Pendeteksiann sebuah objek pada algoritma yolov4-tiny memiliki kesamaan dengan metode yolo dimana menggunakan klasifikasi dengan lokalisaisi, yaitu terdapat tambahan pemberian lokasi objek dalam bentuk bounding box(bx,by,bh, dan bw). Terdapat tahapan yang digunakan pada algortima yolov4-tiny, sebagai berikut :

1. Pembacaan data citra dengan ukuran acak.
2. Mengubah ukuran pada citra menjadi 448 x 448, setelah itu membuat grid pada citra dengan menggunaka ukuran *s \* s* grids. Jika S = 7, maka tiap grid cell memiliki ukuran 64 x 64. Sehingga terdapat sebanyak 49 grid cell. Dan jika pada citra memiliki 2 kelas (nC=2), yaitu “Manusia” (c1), “Kuda” (c2), dan “Background”.
3. Tiap grid cell, contoh nB=2, terdapat B yang berisi 5 nilai, yaitu lokasi dari kordinat x diasumsikan berdasarkan baris (x\_br), kordinat y berdasarkan kolom (y\_kol), ukuran dan nilai confidence (x,y,w,h,cf) terhadap nB bbox yang ada, yaitu B1 dan B2.

Titik grid cell dari citra yang sudah dinormalisasi kordinat x dan y-nya menjadi ukuran [0;1] dan [0;1]

Graphical user interface, text

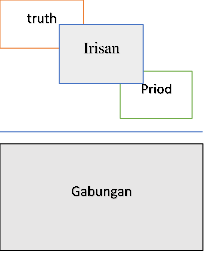
Description automatically generated

Gambar 3.1 Grid algoritma dengan nilai confidence

(Sisco Jupiyadi,2019)

Dari gambar diatas didapatkan hasil

(2.1)



(2.2)

, dimana nB adalah banyaknya bbox. *bbox a*dalah banyaknya *bounding box*,B1 untuk *bbox 1,* B2 untuk *bbox 2.* Persamaan 2.2 merupakan *Intersection Over Union (IoU).*

Perhitungan pada :

Untuk P(objek) dapat dilewati terlebih dahulu, sehingga cukup dengan *cf = IoU.*

1. Melakukan Non-Maximum Suppression (NMS)

* Daftar semua bbox.

Graphical user interface, text, application

Description automatically generated

Gambar 3.2 Grid algoritma non-maximum suppression

* Mengatur bbox dengan skor maks, sebagai “bbox\_max”. Misal bb64 merupakan bbox\_max.
* Membandingkan “bbox\_max” dengan bbox lainnya sebagai “bbox\_cut” yang memiliki skor dibawahnya dan tidak nol. Jika IoU (bbox\_max,bbox\_cut) > 0.5, maka set skor nol untuk bbox\_cut. Dari hasil perbandingan, jika bbox\_max = bb64, bbox\_cut = bb24, maka IoU (bbox\_max,bbox\_cut) > 0.5 (*true*), maka set skor bb24 = 0

Graphical user interface, text

Description automatically generated.

Gambar 3.3 Grid algoritma hasil perbandingan true

(Sisco Jupiyandi, 2019)

* Pada bb ke-22, dari hasil perbandingan, jika bbox\_max = bb64, bbox\_cut = bb22, maka IoU(bbox\_max,bbox\_cut) > 0.5 (*false*), maka skor bb22 tetap.

Graphical user interface, text, application

Description automatically generated

Gambar 3.4 Grid algoritma hasil perbandingan *false* bb22

(Sisco Jupiyandi, 2019)

* Pada bb ke-36, dari hasil perbandingan, jika bbox\_max = bb64, bbox\_cut = bb36, maka IoU(bbox\_max,bbox\_cut) > 0.5 (*false*), maka skor bb36 tetap.

Graphical user interface, text

Description automatically generated

Gambar 3.5 Grid algoritma hasil perbandingan *false* bb36

(Sisco Jupiyandi, 2019)

* Selanjutnya melakukan pembandingan sampai bbox\_max = bb64, bbox\_cut = bb98. Setelah itu, set bbox\_max = bb22, dan bbox\_cut = bb36, dst.
* Plot *bounding box* berdasarkan hasil NMS

Graphical user interface

Description automatically generated

Gambar 3.6 Hasil grid algoritma YOLO berdasarkan NMS

(Sisco Jupiyandi, 2019)

## **Diagram Blok**

Sebelum membuat perancangan alat, terlebih dahulu dilakukan pembuatan blok diagram yang akan digunakan sebagai gambaran dalam pembuatan alat sesusi dengan yang diinginkan. Berikut ini adalah blok diagram alat yang akan digunakan.

Kamera USB

Arm Prosessore

Layar

* Python
* OpenCV
* YOLOv4 Tiny

Program

Masukan

Proses

Keluaran

Inputan Vidio Manual

Gambar 3.7 Blok Diagram

Dari gambar 3.1 dapat diketahui kamera marupakan masukan yang akan digunaka, yang dimana selanjutnya masukan sebagai awal yang akan diproses. Sedangkan arm processor sebagai pemroses hasil dari pembacaan masukan tersebut, dalam pemrosesan pada arm processor akan dilakukan pemrosesna dari program python yang digunakaan pemroesan pada program python akan dilakukan berupa pendeteksian objek dengan menggunakan algoritma yolov4 tiny dan dibantu oleh komputer vision berupa Open CV yang berfungsi untuk melakukan penyederhanaan program, jika proses pengolahan data dari masukan telah selesai, maka hasil pemrosesan akan di tampilkan pada layar, tampilan yang keluar pada layar akan menunjukkan hasil berupa objek yang terdeteksi, nilai akurasi dari objek dan FPS yang dihasilkan dari vidio yang terekam. Penjelasan mengenai masing – masing bagian pada blok diagram sebagai berikut :

* + - 1. Kamera

Kamerea adalah perangkat yang akan digunakan untuk merekam vidio yang akan digunakan sebagai masukan selanjutnya akan diproses pada Arm prosesor

* + - 1. Inputan Vidio Manual

Vidio dimasukkan secara manual dalam format .mp4 yang telah direkam sebelumnya.

* + - 1. Arm Prosesor

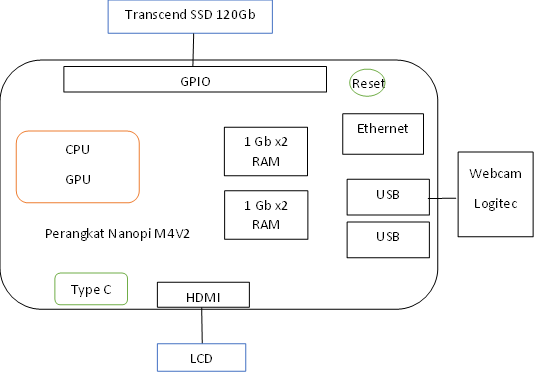
Arm prosessor merupakan sebuah IC yang terpasang pada perangkat Nanopi M4V2 yang akan digunakna untuk melakukan pemrosesan dari hasil masukan kamera, pemerosesan pada Arm prosesor berupa mendeteksian objek menggunakan bahas pemrograman python dan menggunkan algoritma yolv4 tiny yang dibantu oleh komputer vision berupa opencv.

* + - 1. Layar

Layar merupakan perangkat yang digunakan untuk menampilkan hasil dari pengolahan yang dilakukan pada Arm prosesor Nnaopi M4V2.

## **Perancangan Perangkat Keras**

Berikut adalah gambaran dari pemodelan alat yang akan digunakan



Gambar 3.8 Perancangan perangkat keras

A computer on a desk

Description automatically generated with low confidence

Gambar 3.9 Perangkat Keras

## **Perancangan Perangakt Lunak**

Pada tahap akhir ini dilakukan pemrograman perangkat lunak. Dalam proses pemrograman ini dilakukan setelah perancangan alat sudah selesai dilakukan sehingga pemrograman bisa diterapkan pada perangkat. Dalam pemrograman ini program yang telah dibuat akan dimasukkan ke perangkat keras Nanopi M4V2 selanjutnya akan melakukan pemrosesan sehingga diharapkan alat dapat berfungsi sesuai dengan perencanaan. Pemrograman pada perangkat keras Nanopi M4V2 menggunakan perangkat lunak berupa visual studio code dengan menggunakan Bahasa pemrograman python. Penggunaan visual studio code dikarenakan perangkat lunak ini memiliki tampilan yang cukup sederhana dan lebih mudah digunakan, selain itu visual code studio juga memiliki perpustakaan yang banyak. Berikut ini tampilan dari visual studio code yang digunakan.

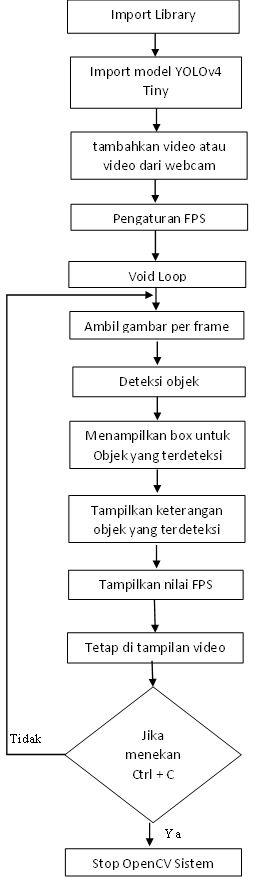
A screenshot of a computer

Description automatically generated

Gambar 3.10 Tampilan viual studio code

## **Perancangan Program Perangkat Lunak**

Pada perancangan program ini penullis menggunakan bahasa pemrograman python dengan menggunakan algoritma yolov4 tiny untuk melakukan pelatihan objek yang akan dideteksi dan dibantu oleh komputer vision yang akan melakukan pendeteksian objek. Berikut adalah kerangka diagram dari program yang dibuat.



Gambar 3.11 Kerangka diagram pemrograman perangkat lunak

Dari kerangka diagram diatas, secara garis besar dapat dijelaskan sebagai berikut:

1. Diawali dengan memasukkan library yang akan digunakan
2. Selanjutnya memasukkan dataset dari yolov4 tiny yang telah dibuat
3. Pada tahap selanjutnya, dilakukan pengaturan jenis vidio yang akan digunakan, apakah menggunakan vidio dari rekaman atau menggunakan vidio secara real time.
4. Pada tahapan selanjutnya akan dilakukan pengaturan FPS pada vidio yang ditampilakan pada layar.
5. Selanjtunya deprogram void loop, akan dilakukan pengmabilan gambar yang terdapat pada vidio, dimana pengambilan gambar ini dilakukan perframe.
6. Setelah itu akan dilakukan pendeteksian objek yang terdapat pada vidio
7. Apabila objek sudah terdeteksi maka objek akan ditandai dengan menggunakan kotak segi empat beserta dengan nama objek yang terdeteksi
8. Penampilan nila FPS pada vidio yang ditampilkan
9. Selanjutnya akan ada fungsi if yang berfungsi untuk mengetahui apakah operator menekan ctrl + C atau tidak.
10. Apabila perator menekan ctrl + C maka vidio akan keluar dan sistem open cv akan berhenti bekerja.
11. Apabial tidak menekan ctrl + C maka program akan terus berjalan dari void loop.

## **Skema Penelitian**

Ya

Tidak

Studi Literatur

Analisa Masalah

Perancangan dan Penambahan perangkat pendukung

Simulasi sistem

Evalusai

Kesimpulan

Gambar 3.12 Skema penelitian

Dari gambar 3.5 Diatas digambarkan penelitian implementasi citra digital pada objek manusia dengan menggunakan Nanopi M4V2, yang dapat dijelaskan sebagai berikut.

1. Mulai

Pada tahapan ini penulis memulai mencari dan mengumpulkan berbagai macam jurnal sehingga penulis mendapatkan judul “implementasi citra digital objek manusia dengan menggunakan Arm Processor Nanopi M4V2”

1. Studi Literatur
2. Pencarian dan pengkajian teori mengenai pembutan perangkat beserta cara kerjanya dari berbagai literatur serta sumber yang bermacam-macam seperti buku, internet, jurnal.
3. Pengumpulan data-data dan spesifikasi sistem yang dipakai untuk pembuatan perangkat sebagai pendukung sistem.
4. Analisa Masalah

Melakukan analisa dari teori yang telah didapat dengan bermacam macam sumber sehingga mendapatkan hasil yang semaksimal mungkin

1. Perancangan dan penambahan perangkat pendukung

Pembuatan rancangan-rancangan kemudian mengimplementasikan rancangan tersebut ke dalam suatu perangkat pendukung.

1. Simulasi sistem

Berdasarkan standar yang ada, tahap selanjutnya adalah melakukan simulasi sistem untuk melihat kinerja sistem tersebut.

1. Evaluasi

Jika sistem telah berjalan, maka didapat keberhasilan maupun ketidakberhasilan dari simulasi sistem tersebut, sehingga dilakukan perbaikan sistem jika didapati sistem tersebut belum berjalan sesuai yang diharapkan. Semua tahapan dilakukan melalui tahap bimbingan (konseling) dengan pembimbing.

1. Kesimpulan

Penarikan kesimpulan harus didasarkan hal-hal yang relevan dari isi evaluasi dan juga tujuan penelitian.

# **BAB IV ANALISA DATA DAN PEMBAHASAN**

Hasil pembahasan dan pengujian yang dilakukan oleh penulis akan dijelaskan pada bab ini. Tujuan dari bab ini adalah untuk mengetahui tingkat keberhasilan dari perancangan sistem yang telah dikerjakan. Dimana pengujian yang dilakukan meliputi, tingkat akurasi objek yang terdeteksi, nilai dari FPS (Frame per second), dan jumlah objek yang terdeteksi.

## **Pengujian Tingkat Akurasi**

Pada prose pengujian tingkat akurasi ini digunakan untuk mengetahui seberapa tingkat akurasi dari perangkat yang telah dibuat sehingga dapat dikatakan bahwa perangkat terebut dapat digunakan secara baik dengan tingkat akurasi yang sempurna.

### **Tujuan Pengujian Tingkat Akurasi**

Pada penelitian ini pengujian tingkat akurasi pada penelitian ini bertujuan untuk apakah perangkat dapat mendeteksi objek dengan sempurna. Pada pengujian ini apabila webcam atau vidio yang dimasukkan terdapat objek manusia maka sistem akan mendeteksi dan memberikan sebuah tanda berupa rectangle tepat dibagian wajah objek dan akan menampilkan nilai tingkat akurasi

### **Prosedur Pengujian Tingkat Akurasi**

Dalam melakukan pengujian tingkat akurasi pada sistem pendeteksian objek dibutuhkan beberapa prosedur yang perlu dilakukan sehingga perangkat bisa bekerja optimal. berikut prosedur tersebut :

* + - 1. Memasukkan data training yang berformat weights, kedalam program
      2. Menjalankan program yang sudah dibuat dalam format \*Py

### **Hasil Pengujian Tingkat Akurasi**

Pada saat melakukan pemrosesan, vidio rekaman yang berbentuk vidio, dimasukkan kedalam program yang telah dibuat selanjut, selanjutnya vidio akan diproses oleh program, hasil pemrosesan akan ditampilkan berupa nilai fps dari vidio tersebut, tingkat akurasi dan nama objek yang terdeteksi. Berikut merupakn contoh dari pemrosesan vidio dengan masukan secara manual.

Graphical user interface

Description automatically generated

Gambar 4.1 Hasil 1 objek hadap belakang

Pada hasil tangkapan layar diatas, dilakukan percobaan dengan objek membelakangi kamera dimana didapatkan hasil bahwa objek terdeteksi dengan dengan tingkat akurasi yang tinggi dan nilai dari FPS yang tidak terlalu rendah.

Graphical user interface

Description automatically generated

Gambar 4.2 Hasill 2 objek hadap depan

Pada tangkapan layar diatas, dilakukan percobaan dengan menggunakan 2 buah objek manusia yang dimana terdapat objek yang menggunakan masker dan yang tidak menggunakan masker, dengan hasil kedua objek tersebut terderteksi oleh perangkat dan mendapatkan tingkat akurasi yang tinggi dengan nilai FPS yang menurun.

Graphical user interface

Description automatically generated

Gambar 4.3 Hasil banyaknya objek

Pada tangkapan layar diatas, dilakukan percobaan dengan menggunakan objek manusia yang berjalan dijalan raya dengan terdapat objek seperti tiang sebagai objek yang menyerupai manusia, dari percobaan tersebut didapatkan hasil objek manusia yang berjalan terderteksi oleh perangkat dengan tingkat akurasi yang tinggi dan objek yang lain tidak terdeteksi sedangkan nilai FPS yang sedikit tinggi.

Dari ketiga gambar diatas didapatkan tingkat akurasi yang tinggi dan sedikit penurunan nilai FPS. Selain itu tingkat akurasi pada pendeteksisan objek tidak terpengaruh dengan posisi objek, apakah membelakangi kamera atau menghadap kamera, dikrenakan nilai akurasi masih diatas 90%. Berikut table dari tangkapan layar diatas.

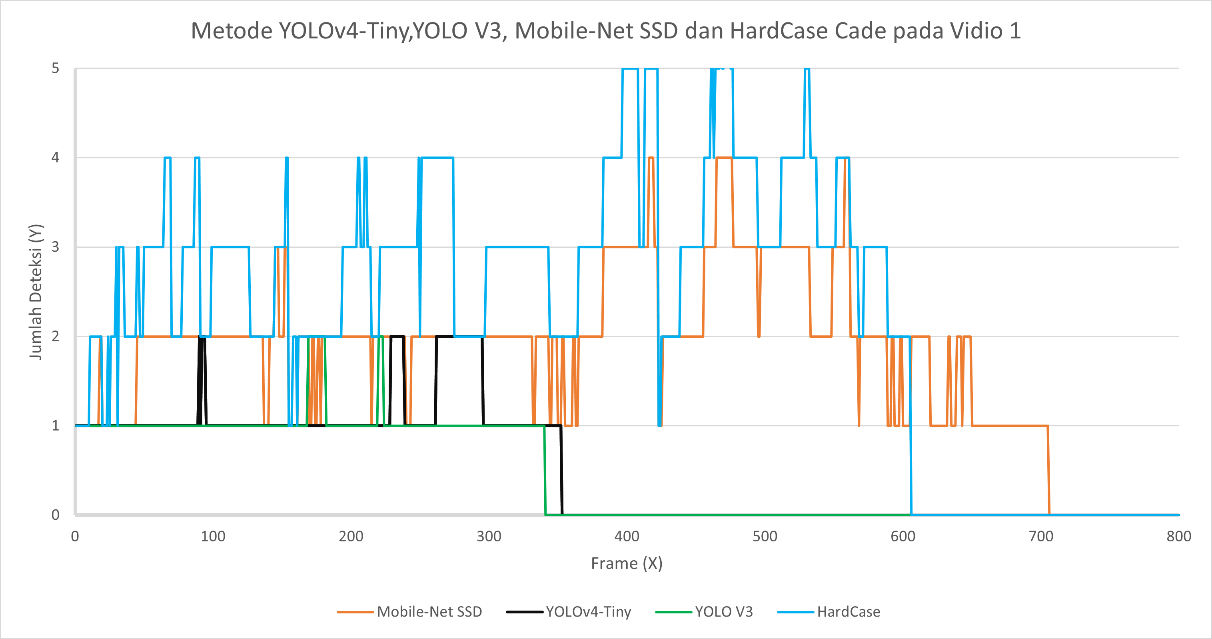
Tabel 4.1 Deteksi objek YOLOv4-Tiny

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Objek | Akurasi (%) | Jumlah Terdeteksi | Jumlah Objek |
| 1 | 1 Objek | 0.99 | 1 | 1 |
| 2 | 2 Objek | * 1.0 * 0.86 | 2 | 2 |
| 3 | Banyak Objek | * 0.81 * 0.87 * 0.6 * 0.84 * 0.81 | 5 | 5 |
| FPS | | 2.81 – 3.55 | | |

Dari tabel diatas dapat dilihat bahwa dengan menggunakan metode YOLOv4 Tiny kita dapat melakukan pemrosesan citra digital dengan mendeteksi objek khususnya objek manusia dengan tingkat akurasi yang tinggi dan tidak terpengaruh dengan objek yang lainnya.

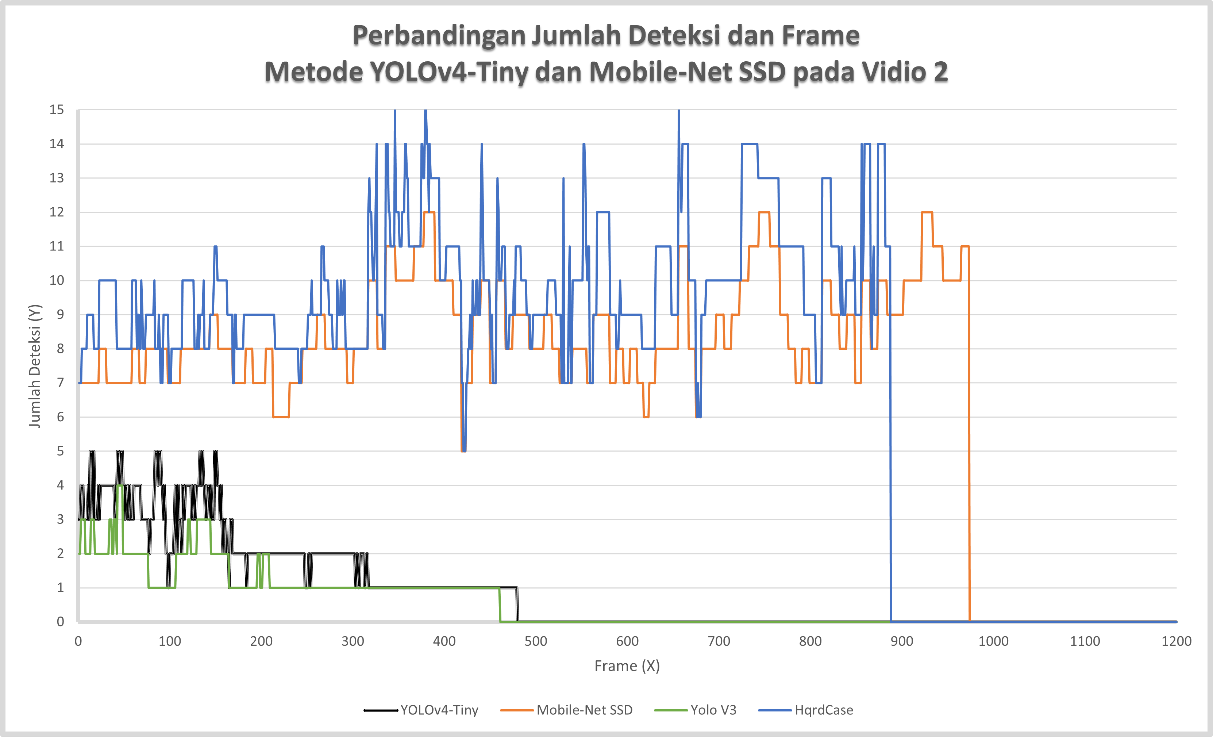
## **Perbandingan Tingkat Akurasi**

Dalam penelitian ini, penulis juga melakukan perbandingan dengan 4 metode pendeteksian objek yang berbeda diantaranya : YOLOv4-Tiny, Yolo V3, HardCase dan Mobile-Net SSD. Tujuan dari perbandingan ini adalah untuk mengetahui bagaimana kualitas dari Arm Prosesore pada 4 metode yang berbeda, dan untuk mengatahui apakah metode dari YOLOv4-Tiny yang penulis gunakan mendapatkan hasil yang lebih baik dari ketiga metode tersebut. Berikut curve dari hasil perbandingan.



Gambar 4.4 Kurva Perbandingan Pada Vidio 1

Pada gambar diatas terlihat perbedaan dari 4 metode yang digunakan dimana metode YoloV4-Tiny mendapatkan hasil yang stabil dengan jumlah objek yang sesuia dengan objek pada vidio, sedangkan hasil pada Mobile-Net SSD didapatkan hasil dengan objek yang terdeteksi lebih banyak dari pada objek yang terdapat pada vidio, selain itu pada metode YoloV3 didapatkan penurunan jumlah objek yang terdeteksi dari pada jumlah objek yang terdapat pada vidio dan yang terakhir degan menggunakan metode HardCase didapatkan hasil terdapat banyak objek ayng terdeteksi jika dibandingkan dengan ketiga metode. Sehingga didapatkan hasil bahwa YoloV4 Tiny lebih baik jika dibandingakan dengan ketiga metode diatas. Akan tetapi jumlah FPS (jumlah frame perdetik) pada YoloV4-Tiny lebih rendah daripada metode Mobile-Net SSD dan Hardcase, sedangkan dengan metode Yolo V3 FPS.



Gambar 4.5 Kurva Perbandingan Pada Vidio 2

Pada gambar diatas terlihat perbedaan dari 4 metode yang digunakan dimana metode YoloV4-Tiny mendapatkan hasil yang stabil dengan jumlah objek yang sesuai dengan objek pada vidio, sedangkan hasil pada Mobile-Net SSD didapatkan hasil dengan objek yang terdeteksi lebih banyak dari pada objek yang terdapat pada vidio, selain itu pada metode YoloV3 didapatkan penurunan jumlah objek yang terdeteksi dari pada jumlah objek yang terdapat pada vidio dan yang terakhir degan menggunakan metode HardCase didapatkan hasil terdapat banyak objek ayng terdeteksi jika dibandingkan dengan ketiga metode. Sehingga didapatkan hasil bahwa YoloV4 Tiny lebih baik jika dibandingakan dengan ketiga metode diatas. Akan tetapi jumlah FPS (jumlah frame perdetik) pada YoloV4-Tiny lebih rendah daripada metode Mobile-Net SSD dan Hardcase, sedangkan dengan metode Yolo V3 FPS.

## **Performa CPU dan RAM**

Pada proses ini penulis akan membahas bagaimana performa dari RAM (Random Access Memory) dan CPU (Central Processore Unit) ketika pengolahan citra digital dijalankan dan pada saat perangkat belum digunakan untuk melakukan pengolahan citra digital.

Graphical user interface

Description automatically generated

Gambar 4.6 (Pada saat program tidak dijalankan)

Graphical user interface

Description automatically generated

Gambar 4.7 (pada saat program dijalankan)

Dari kedua gambar diatas, dapat dilihat bahwa penggunaan CPU pada saat program tidak dijalankan lebih sedikit jika dibandingakan ketika program dijalankan.

Dibawah ini adalah tabel perbandingan dari kedua gambar diatas.

Tabel 4.2 Performa CPU pada deteksi objek dengna metode YOLOv4 Tiny

|  |  |  |  |
| --- | --- | --- | --- |
| Nama | CPU Frequency | Tanpa program | Program dijalankan |
| Little Core | 1.42 GHz | 5.3 % | 65.8 % |
| 1.42 GHz | 0 % | 42.1 % |
| 1.42 GHz | 4 % | 59.9 % |
| 1.42 GHz | 0 % | 51.6 % |
| Big Core | 1.80 GHz | 0 % | 75.2 % |
| 1.80 GHz | 2.6 % | 74.8 % |
| RAM | 4 GB | 10 % | 25 % |
| Suhu CPU | | 38 °C | 62 °C |

Dari data yang didapatkan penggunaan CPU pada perangkat tidak terlalu tinggi atau tidak sampai 100% dan RAM yang terpakai hanya 25% dari 100% atau 1,02Gb dari 4Gb Ram yang tersedia selain itu suhu pada perangkat juga tidak mengalami kepanasan yang terlalu berlebihan. Sehingga secara keseluruhan program dan perangkat berjalan dengan baik dan tidak terdapat masalah dalam penggunaanya.

# **BAB V KESIMPULAN DAN SARAN**

Pada bab ini akan dipaparkan hasil kesimpulan dari perangkat implementasi proses citra digital objek manusia dengan YOLOv4 tinny menggunakan NanoPi M4V2 yang telah di analisa dan juga akan diberikan saran untuk pengembangan atau peningkatan dari sistem yang akan dibuat untuk penelitian selanjutnya.

## **Kesimpulan**

Kesimpulan dari hasil Analisa pada deteksi objek manusia dengan menggunakan metode YOLOV4 Tiny pada perangkat NanoPi M4V2 antara lain :

Hasil dari pengujian yang telah dilakukan, dapat disimpulkan bahwa perangkat dan sistem yang telah dibuat dapat bekerja dengan baik untuk mendeteksi objek manusia dengan 2 macam kondisi yaitu dengan 2 objek hadap kamera dan membelakangi kamera dan dengan objek acak yang lebih banyak.

Perangkat NanoPi M4V2 memiliki Tingkat FPS (Frame Per Second) tidak terlalu rendah pada pengolahan objek citra digital dalam format vidio dengan metode YOLOV4 Tiny.

Nanopi M4V2 dapat digunakan dengan baik untuk melakukan pengolahan citra digital dengan menggunakan metode YOLOV4-Tiny.

Tingkat akurasi pada metode YOLOV4 Tiny dengan menggunakan perangkat NanoPI M4V2 terlihat memiliki tingkat akurasi yang tinggi dengan inputan vidio yang memiliki pencahayaan yang lebih terang.

Pada saat perangkat dijalankan tidak terdapat masalah dengan penggunaan CPU pada NanoPi M4V2 dengan menggunakan metode YOLOV4-Tiny untuk mendetesi objek manusia

Dari keempat metode yang dicoba, dalam penelitian ini metode YOLOV4-Tiny memiliki tingkat akurasi yang paling tepat dan setabil jika dibandingkan dengan tiga metode yang lain.

Kualitas dari vidio dapat mempengaruhi tingkat akurasi dari objek yang dideteksi.

## **Saran**

Untuk penelitian selanjutnya terdapat beberapa saran yang dapat diimplementasikan untuk mengembangkan aplikasi lain :

* + - 1. Untuk pengembangan selanjutnya, penelitian selanjutnya dapat mengembangkan pendeteksian objek ini untuk membuat autonomous robot, dimana robto dapat menghindari dan mendeteksi objek yang direkam oleh kamera.
      2. Penelitian disarankan untuk menggunakan perangkat yang lebih terbaru dan memiliki spesifikasi yang lebih tinggi dari perangkat NanoPi-M4V2 yang penulis gunakan sehingga mendapatkan hasil yang lebih maksimal terutama pada hal tangkapan FPS dari inputan vidio.
      3. Penggunaan metode objek deteksi yang terbaru, dapat digunakan sehingga tingkat akursai dapat lebih optimal lagi.
      4. Disarankan untuk menggunakan kamera yang memiliki resolusi tinggi sehingga mendapatkan kualitas vidio yang lebih bagus.

# **DAPTAR PUSTAKA**

B.P.G. Pamungkas, B. Nugroho, & F. Anggraeny (2021). DETEKSI DAN MENGHITUNG MANUSIA MENGGUNAKAN YOLO-CNN. jurnal informatika dan sistem informasi (JIFoSI), volume 2, No.1

J. Huang, H.Zhang, L. Wang, Z. Zhang, & C. Zhao (2021). IMPROVED YOLOv3 MODEL FOR MINIATURE CAMERA DETECTION. Optics and Laser Technology, volume 142, No.107133

Muwardi, R., Qin, H., Gao, H., Ghifarsyam, H. U., Hajar, M. H. I., & Yunita, M. (2020, September). Research and Design of Fast Special Human Face Recognition Sistem. In *2020 2nd International Conference on Broadband Communications, Wireless Sensors and Powering (BCWSP)* (pp. 68-73). IEEE.

Hidayatulloh, M. S. (2021). *“*SISTEM PENGENALAN WAJAH MENGGUNAKAN METODE *YOLO (You Only Look Once)* yang diupload pada Doctoral dissertation, Universitas Dinamika.

Agung, W. A. S. (2021). “SISTEM PENGHITUNG PENGUNJUNG PUSAT PERBELANJAAN DAN DETEKSI MASKER PADA PANDEMI COVID-19 MENGGUNAKAN LIBRARY OPENCV DAN YOLO” (Doctoral dissertation, Universitas Andalas).

Li, M., Zhang, Z., Lei, L., Wang, X., & Guo, X. (2020). Agricultural greenhouses detection in high-resolution satellite images based on convolutional neural networks: Comparison of faster R-CNN, YOLO v3 and SSD. *Sensors*, *20*(17), 4938.

Tan, L., Huangfu, T., Wu, L., & Chen, W. (2021). Yang berjudul Comparison of YOLO v3, Faster R-CNN, and SSD for Real-Time Pill Identification.

Gupta, P., Sharma, V., & Varma, S. (2021). People detection and counting using YOLOv3 and SSD models. *Materials Today: Proceedings*.

Nepal, U., & Eslamiat, H. (2022). Comparing YOLOv3, YOLOv4 and YOLOv5 for Autonomous Landing Spot Detection in Faulty UAVs. *Sensors*, *22*(2), 464.

Y. Wang, Y. Zou and W. Wang, "Manifold-Based Visual Object Counting," in IEEE Transactions on Image Processing, vol. 27, no. 7, pp. 3248-3263, July 2018, doi: 10.1109/TIP.2018.2799328.

T. Stahl, S.L. Pintea, J.C. Van Gemert, Divide and Count: Generic Object 358 Counting by Image Divisions, IEEE Trans. Image Process. 28 (2) (2019) 1035– 359 1044, <https://doi.org/10.1109/TIP.2018.2875353>.

Z. Dai, H. Song, X. Wang, Y. Fang, X.u. Yun, Z. Zhang, H. Li, Vidio-based vehicle 361 counting framework, IEEE Access 7 (2019) 64460–64470, [https://doi.org/ 362 10.1109/Access.628763910.1109/ACCESS.2019.2914254](https://doi.org/%20362%2010.1109/Access.628763910.1109/ACCESS.2019.2914254)

R. Feng, C. Fan, Z. Li, X. Chen, Mixed Road User Trajectory Extraction from 364 Moving Aerial Vidios Based on Convolution Neural Network Detection, IEEE 365 Access 8 (2020) 43508–43519, [https://doi.org/10.1109/Access.628763910. 366 1109/ACCESS.2020.2976890.](https://doi.org/10.1109/Access.628763910.%20366%201109/ACCESS.2020.2976890.)

S. Sun, N. Akhtar, H. Song, C. Zhang, J. Li, A. Mian, Benchmark Data and Method 372 for Real-Time People Counting in Cluttered Scenes Using Depth Sensors, IEEE 373 Trans. Intell. Transp. Syst. 20 (10) (2019) 3599–3612, [https://doi.org/10.1109/ 374 TITS.697910.1109/TITS.2019.2911128](https://doi.org/10.1109/%20374%20TITS.697910.1109/TITS.2019.2911128).

FITRI RAHMADANI, S. I. S. K. A. (2020). *Vidio Tutorial Pembuatan Kompres Kayu Manis (Cinnamomum burmanii) Segabai Upaya Menurunkan Nyeri Sendi Pada Lansia Hak Kekayaan Intelektual (HKI)* (Doctoral dissertation, Universitas Muhammadiyah Ponorogo).

Shafiee, M. J., Chywl, B., Li, F., & Wong, A. (2017). Fast YOLO: A fast you only look once sistem for real-time embedded object detection in vidio. *arXiv preprint arXiv:1709.05943*.

Fujitake, M., & Sugimoto, A. (2021, July). Real-time object detection by feature map forecast for live streaming vidio. In *2021 IEEE International Conference on Multimedia and Expo (ICME)* (pp. 1-6). IEEE.

Hidayat, F., Hamami, F., Dahlan, I. A., Supangkat, S. H., Fadillah, A., & Hidayatuloh, A. (2020, July). Real Time Vidio Analytics Based on Deep Learning and Big Data for Smart Station. In *Journal of Physics: Conference Series* (Vol. 1577, No. 1, p. 012019). IOP Publishing.

Jupiyandi, S., Saniputra, F. R., Pratama, Y., Dharmawan, M. R., & Cholissodin, I. (2019). Pengembangan deteksi citra mobil untuk mengetahui jumlah tempat parkir menggunakan CUDA dan modified YOLO. *Jurnal Teknologi Informasi dan Ilmu Komputer*, *6*(4), 413-419.