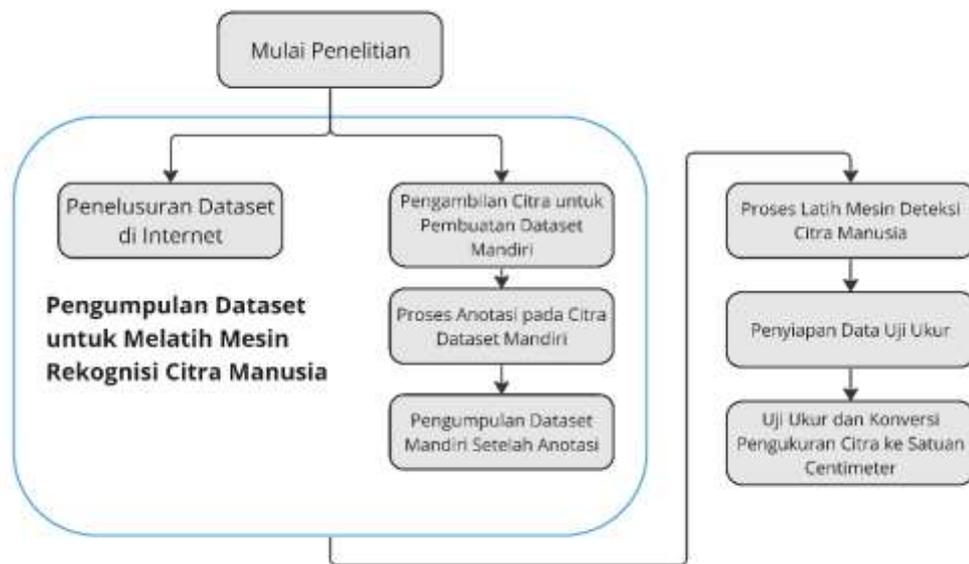


BAB III METODOLOGI PENELITIAN

3.1 Prosedur Penelitian

Riset dilakukan untuk mengembangkan perangkat lunak yang terintegrasi dengan hasil kerja perangkat keras berupa kamera. Pada penelitian ini dilakukan prosedur berikut beserta tujuannya. Secara umum terdapat dua bagian, yaitu bagian pelatihan mesin deteksi citra manusia, dan bagian perhitungan posisi manusia berdasarkan citra manusia yang terdeteksi.



Gambar 3. 1 Alur Penelitian

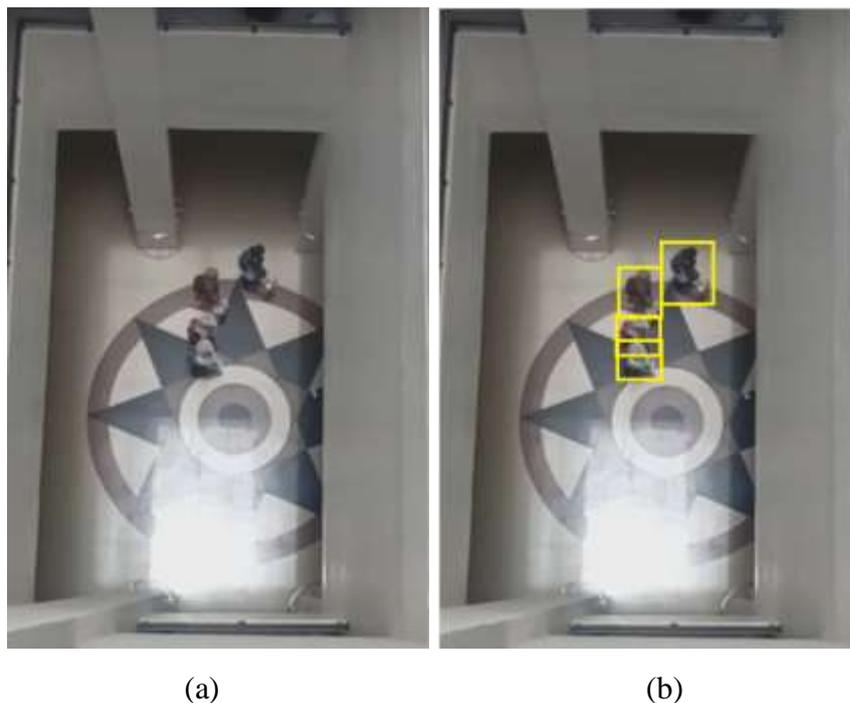
Pada bagian pelatihan mesin rekognisi citra manusia, terdapat empat tahapan. Penelitian dimulai dengan penelusuran internet maupun pengambilan foto untuk mendapatkan data citra sebagai modal data latih. Data citra yang telah didapatkan perlu dianotasi dengan spesifikasi yang sesuai dengan algoritma pendeteksi dan rekognisi citra yang dipakai, yaitu algoritma YOLO v4. Data yang telah dianotasi harus dihimpun menjadi susunan data dengan nama yang teratur disertai fail konfigurasi sesuai spesifikasi algoritma YOLO v4. Setelah konfigurasi sesuai, mesin rekognisi dan pendeteksi citra manusia dapat dilatih dengan data yang telah dikumpulkan.

Bagian perhitungan posisi manusia berdasarkan citra manusia terdeteksi memiliki dua tahapan. Tahapan pertama adalah penyiapan data ukur dan data citra. Data ukur adalah data berupa hasil pengukuran dengan meteran berdasarkan

kerangka acuan yang dibuat. Adapun data citra yang dimaksud pada tahap ini adalah data citra yang akan dibandingkan secara langsung dengan hasil pengukuran. Ketika dua data tersebut telah terkumpul, tahap selanjutnya adalah melakukan perhitungan dan perbandingan hasil pengukuran dari data ukur dan hasil perhitungan berdasarkan data citra.

3.2 Penelusuran Internet dan Pengambilan Foto untuk Data Citra

Data citra pada bagian ini didefinisikan sebagai data berupa fail gambar dengan format JPEG atau JPG. Data citra disyaratkan untuk mengandung citra manusia yang tampak dari atas. Data citra yang ditelusuri dari internet adalah data citra yang telah dianotasi dan dihimpun sehingga memungkinkan untuk langsung digunakan sebagai data latih, sehingga dapat disebut sebagai *dataset*. Adapun data citra yang diambil dengan kamera atau difoto adalah data yang belum dianotasi, sehingga akan dilanjutkan pada tahap anotasi.



Gambar 3. 2 Data citra dengan format JPG (a) tanpa anotasi, dan (b) dengan anotasi

3.2.1 Penelusuran Internet untuk *Dataset*

Penelusuran di internet dilakukan dengan mesin pencari Google, forum *data enthusiast* Kaggle, dan forum pengembang perangkat lunak Github. Penelusuran dilakukan pada Google dan Kaggle dengan frasa “human top-view”, “person top-view”, “human overhead”, dan “person overhead”. Proses pencarian memerlukan kriteria untuk memilih *dataset* yang akan dilanjutkan ke proses selanjutnya. Kriteria pertama adalah *dataset* tersebut mengandung citra manusia dari sudut pandang atas kepala seperti pada Gambar 3. 2. Kriteria kedua adalah *dataset* tersebut telah disusun dalam format YOLO. Selain itu, data tanpa distorsi lensa optis lebih baik untuk dipilih. *Dataset* dengan jumlah data lebih dari 2000 data lebih disukai karena lebih sesuai untuk algoritma latih mesin rekognisi citra.

3.2.2 Pengambilan Foto untuk Data Citra

Tahapan pengambilan foto dilakukan karena data latih perlu dilengkapi dengan ragam yang berbeda dari data citra yang bersumber dari internet. Hal ini dilakukan untuk memperkaya variasi fitur atau ciri manusia yang akan dipelajari oleh mesin, sehingga mesin lebih adaptif dan menghindari *overfit*.

Pengambilan foto dilakukan di dua tempat yang berbeda. Lokasi pertama pengambilan foto adalah Lobi Gedung JICA-FPMIPA A, Universitas Pendidikan Indonesia. Lokasi kedua adalah Hall Gedung FPMIPA B, Universitas Pendidikan Indonesia. Kedua lokasi tersebut dipilih karena merupakan lokasi yang memenuhi kriteria yang diharapkan, yaitu kemudahan pengambilan data dan ramai dilalui orang. Data mudah diambil karena dua lokasi tersebut dapat dipantau dari lantai tingkat atasnya. Data untuk kedua lokasi tersebut diambil dari lantai dua Gedung tersebut dengan menyorot pandangan kamera menuju lantai satu atau lantai dasar. Ketinggian tempat pengambilan foto adalah sekitar 4 meter dari permukaan lantai dasar.



Gambar 3. 3 Lokasi pengambilan foto di (a) Gedung JICA-FPMIPA A, dan (b) Gedung FPMIPA B

Data citra berupa foto diambil menggunakan *smartphone* Samsung A22 5G. Alasan dari penggunaan gawai ini adalah kesesuaian spesifikasi minimum untuk mengambil data dengan resolusi yang baik dan juga *frame rate* yang cukup baik.

Tabel 3.1 Spesifikasi Kamera Samsung A22 5G

Seri Smartphone	Samsung A22 5G
Resolusi	48 MP (4:3 ratio)
<i>Frame Rate Video</i>	30 Frame per detik (<i>FPS/Frame per Second</i>)

Data diambil dalam bentuk video karena video adalah pengambilan foto berulang berdasarkan fungsi waktu. Sehingga untuk dilanjutkan pada proses selanjutnya, video perlu dipecah menjadi *frame-frame* yang tercatat dalam perekaman. Proses pemecahan video tersebut menggunakan program Python dengan pustaka *decord*. Terdapat fungsi yang memungkinkan pengguna untuk menyimpan foto atau *frame* setiap 15, 10, atau 5 *frame*.

Setiap video yang telah dipecah disimpan dalam suatu folder dengan nama yang sama dengan nama fail video yang dipecah. Fail video diberi nama dengan kode “[huruf mewakili lokasi][lokasi lantai gedung][dua angka indeks].mp4”, dengan contoh “a200.mp4” yang bermakna bahwa video tersebut adalah video yang diambil di lokasi Gedung A lantai dua dan merupakan video pertama dalam sesi pengambilan pertama dengan indeks 00. Setiap *frame* yang telah dipecah juga

diganti namanya dengan sistem nama berupa kode. Sistem penamaannya adalah “[nama video]_v0(indeks).jpg” dengan contoh “a200_v0(0).jpg” yang berarti ini adalah *frame* dari fail video a200.mp4 yang merupakan *frame* pertama yang disimpan. Contoh penamaan ditunjukkan juga pada Gambar 3. 4.

Nama Gedung : JICA-FPMIPA A	Nama Video : a202
Kode Gedung : a	Indeks frame : 20
Lantai ke- : 2	
Video ke- (index): 02	Nama frame : a202_v0(20).jpg
Nama frame : a202.mp4	

(a) (b)

Gambar 3. 4 Contoh penamaan untuk (a) video, dan (b) gambar per frame dari video

Dapat disimpulkan bahwa setiap video memiliki *folder slicing* dengan nama folder yang sama dengan nama video. Setiap *folder slicing* memiliki *frame* dengan nama *frame* yang mengandung nama identik mewakili nama *folder slicing* dan nama video. Setiap nama video dan nama *frame* memiliki nomenklatur yang ditentukan. Simpulan ini dinyatakan secara visual dengan Gambar 3. 5.

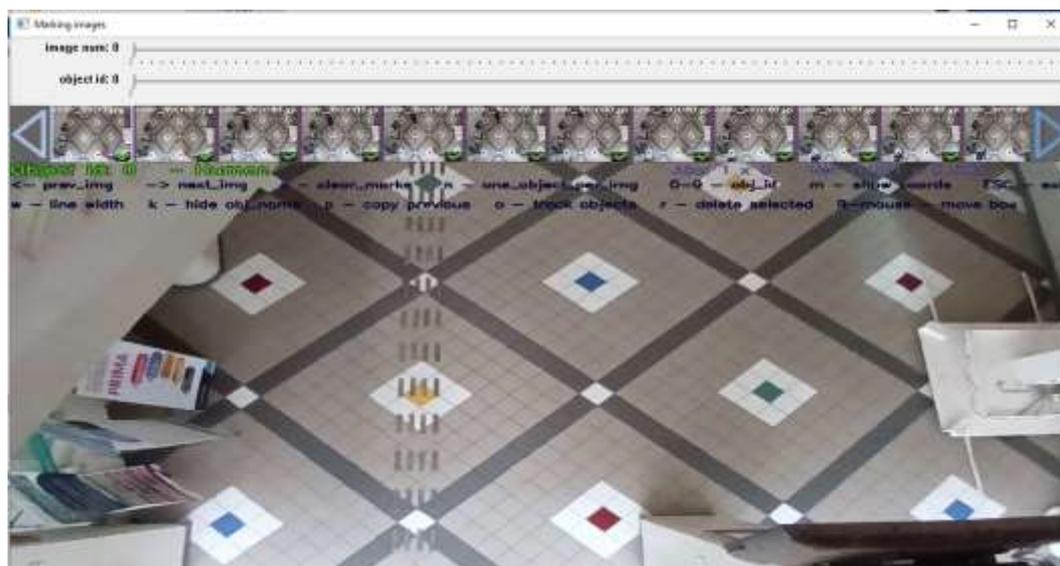


Gambar 3. 5 Visualisasi hubungan penamaan fail video dengan nama folder *slicing* dan data citra hasil *slicing*

3.3 Anotasi Foto untuk Data Citra

Prosedur yang dilakukan mengikuti ketentuan anotasi YOLO dengan bantuan software YOLO Mark. YOLO Mark yang telah terinstall dapat digunakan untuk memberi anotasi sesuai dengan instruksi manual. Berdasarkan instruksi manual, peneliti mengumpulkan data-data gambar yang akan dianotasi pada folder *img* yang diinduki oleh folder Yolo_mark/x64/Release/data. Pada tahap ini, setiap *folder slicing* yang telah dibuat di tahap sebelumnya dikumpulkan pada folder *img* tersebut. Setelah itu, program Yolo Mark dijalankan. Program akan menampilkan daftar gambar yang terdaftar pada folder *img*, gambar yang akan dianotasi, dan juga tampilan lain yang membantu.

Peneliti menandai setiap objek manusia. Penandaan dilakukan dengan mengarahkan mouse pada titik sudut dari kotak yang akan menjadi penanda. Kemudian, peneliti menekan tahan tombol kiri pada mouse dan menariknya sehingga terbentuk kotak. Kursor ditarik saat menahan tombol kiri sampai terbentuk kotak dengan area yang diharapkan. Jumlah tanda yang dibuat pada tiap gambar tidak terbatas.



Gambar 3.6 Tampilan YOLO Mark menunjukkan opsi-opsi shortcut dan gambar Lobi Gedung JICA-FPMIPA A

Setelah selesai menandai sebuah gambar, peneliti menekan tombol panah ke kanan pada *keyboard* untuk melanjutkan ke gambar selanjutnya. Prosedur penandaan dilakukan kembali sampai selesai dan dilanjutkan dengan menekan tombol panah ke kanan untuk melanjutkan. Prosedur terus diulang hingga gambar yang ingin dianotasi habis. Prosedur anotasi diakhiri dengan menekan tombol ESC pada *keyboard*. Hasil anotasi berupa fail *text* akan disimpan secara otomatis menggunakan nama yang sama dengan nama *frame* yang dianotasi.

3.4 Penghimpunan Data Citra Pasca Anotasi

Data citra yang telah dianotasi perlu dikumpulkan dalam satu folder untuk kemudian didaftarkan pada daftar nama data yang akan dibaca oleh mesin rekognisi yang akan memanggil data tersebut. Data citra dan hasil anotasi berformat *text* dikumpulkan pada folder google drive dengan alamat lokasi “Drive Saya/Skripsi/data/v0/img” yang kemudian akan disebut folder *img* dan dikumpulkan dalam fail beralamat di “Drive Saya/Skripsi/data/v0/” dengan nama *train.txt*.

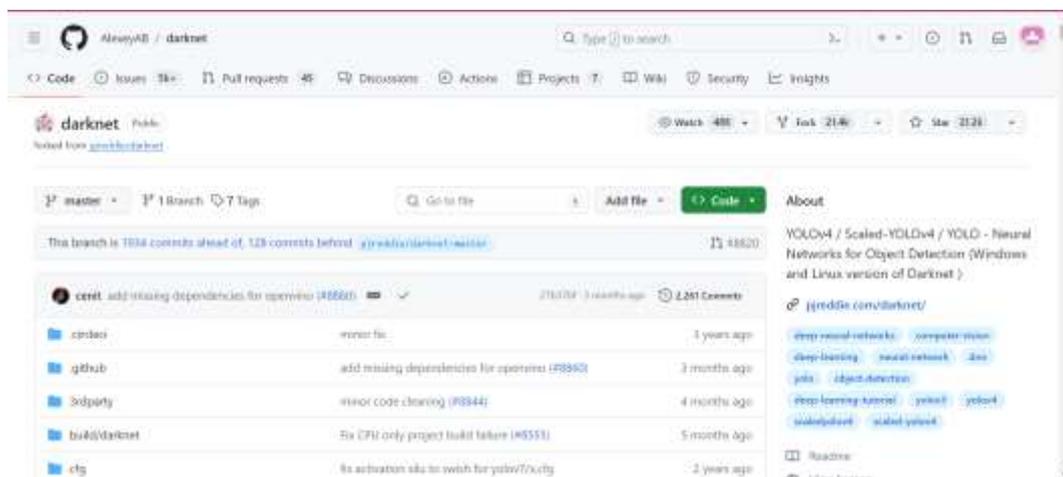
Tahap pertama adalah menyalin seluruh gambar dan hasil anotasi dari tiap *folder slicing* ke dalam folder *img*. Kedua, dilanjutkan dengan pembuatan fail *train.txt* yang berisi seluruh data latih di folder *img*. Setelah dua tahap ini, maka dilanjutkan dengan membuat fail konfigurasi dengan nama *obj.data* dan *obj.names*. Fail *obj.data* berisi lima baris teks. Masing-masing baris secara berturut-turut adalah *classes* yang diisi jumlah kategori objek yaitu 1, *train* yang berisi deklarasi alamat tempat fail *train.txt* disimpan, *valid* berisi deklarasi alamat tempat fail *test.txt* disimpan dan dapat digantikan dengan *train.txt*, *names* berisi lokasi fail *obj.names*, dan *backup* berisi lokasi direktori tempat menyimpan hasil pelatihan.

```
1 classes= 1
2 train = data/train.txt
3 valid = data/train.txt
4 names = data/obj.names
5 backup = backup/
```

Gambar 3.7 Isi dari fail "obj.data" untuk proses latih mesin deteksi

3.5 Proses Latih Mesin Rekognisi Citra Manusia

Proses training model dilakukan pada platform Google Colab yang merupakan *development environment* berbasis Jupyter Notebook dengan akses online pada *Remote Computer* milik Google. Proses training melibatkan penggunaan Pustaka Drive oleh Google. Pustaka tersebut digunakan untuk menghubungkan Colab Notebook dengan penyimpanan di Google Drive. Google Drive digunakan sebagai direktori utama pengembangan program ini. Direktori utama tersebut berisi serangkaian program YOLO yang telah disalin dari situs Github Alexey AB. Di dalam direktori tersebut juga terdapat *dataset* yang disiapkan untuk dilatih dan juga hasil pelatihan model kecerdasan buatan.



Gambar 3. 8 Tampilan laman github pada repository darknet

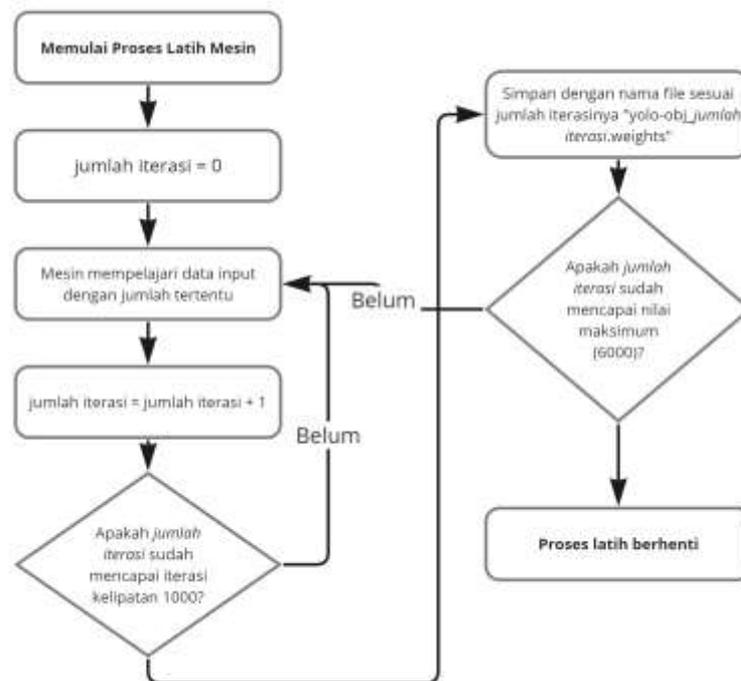
Sebelum proses pelatihan model kecerdasan buatan dilakukan, dibutuhkan penyesuaian fail konfigurasi layer kecerdasan buatan. Fail konfigurasi adalah fail yang berisi parameter pelatihan model kecerdasan buatan dan deklarasi tiap layer yang dibuat beserta parameternya. Bagian yang disesuaikan adalah *network size*, jumlah kelas dari objek yang ingin dideteksi, dan jumlah filter.

Proses training dilakukan dengan *dataset* yang telah disebutkan sebelumnya. Pelatihan model dimulai dengan memberikan instruksi sebagaimana pada Gambar 3.9 di *command line* milik *remote computer* Google Colab. Instruksi tersebut dimulai dengan memanggil program darknet yang disimpan di folder darknet, menjalankan fungsi detector train, menambahkan parameter lokasi dokumen

dengan ekstensi “.data”, lokasi fail konfigurasi layer kecerdasan buatan yang akan dilatih, dan lokasi *weight* bawaan dari model YOLO v4.

```
!darknet/darknet detector train
darknet/build/darknet/x64/data/obj.data
darknet/build/darknet/x64/cfg/yolo-obj.cfg
darknet/build/darknet/x64/yolov4.conv.137
```

Gambar 3.9 *Command Line* Inisiasi Pelatihan Model



Gambar 3. 10 Alur Proses Latih Mesin Pendeteksi Citra Manusia

Model telah dilatih sebanyak 3000 iterasi menggunakan data UCU Apps, sedangkan pada 3000 iterasi kemudian menggunakan data yang dibuat penulis. Proses pelatihan dilakukan secara bertahap. Setiap 1000 iterasi, algoritma secara otomatis menyimpan fail backup hasil pelatihan. Ketika proses latih terhenti, algoritma juga menyimpan fail dengan nama “yolo-obj_last.weights”. Untuk melanjutkan proses latih, dapat menggunakan instruksi yang sama seperti Gambar 3.9 dengan mengubah fail *weight* yang digunakan, misalnya dengan weight terbaru

menggunakan fail “yolo-obj_last.weights” atau memulai Kembali proses latih dari 3001 dengan fail “yolo-obj_3000.weights” sehingga tampak seperti Gambar 3.11.

```
!darknet/darknet detector train
darknet/build/darknet/x64/data/obj.data
darknet/build/darknet/x64/cfg/yolo-obj.cfg
darknet/build/darknet/x64/yolo-obj_last.weights
```

Gambar 3.11 *Command Line* Melanjutkan Proses Pelatihan Model

Mesin rekognisi citra manusia akan dilatih hingga 6000 iterasi. Setelah pelatihan, hasil proses latih berupa mAP dapat ditinjau sebagai parameter kualitas pendeteksian. Model dengan parameter mAP yang lebih tinggi adalah model yang akan digunakan sebagai pendeteksi citra manusia pada penelitian ini.

3.6 Persiapan Data Ukur Posisi Manusia dan Data Citra

Model akhir pada penelitian ini ditujukan sebagai alat ukur Panjang yang mengukur posisi manusia yang terdeteksi berdasarkan kerangka acuan. Sehingga, setelah pendeteksian dilakukan, dapat dilakukan perhitungan pengukuran. Dengan demikian, data pengukuran posisi dengan meteran dan dengan citra perlu diambil. Data pengukuran posisi dilakukan dengan meteran dan dengan citra, dengan *point of reference* terletak pada titik tengah kamera yang ditandai dengan marker *midpoint* pada aplikasi kamera di Windows. Pada dasarnya, pengukuran dilakukan dari titik kerangka acuan menuju titik tengah. Pengukuran dilakukan dengan membentangkan pita ukur atau meteran sepanjang satu meter. Lalu objek yang diukur posisinya berpindah menuju titik lain dengan jarak yang tetap namun sudutnya relatif berubah terhadap titik tengah sebagai acuan.

3.7 Perhitungan Posisi Manusia berdasarkan Citra Manusia

Posisi dihitung berdasarkan titik acu yang ditentukan. Pada penelitian ini, titik acuan ditentukan pada titik tengah citra yang tergambar. Adapun pengukuran di dunia nyata, titik acunya mengikuti titik tengah pada citra. Titik tengah pada citra ini yang kemudian dicari koordinat pixelnya. Setelah ditemukan koordinat pixel

titik tengah, ditentukan formulasi untuk menghasilkan koordinat baru yang terbagi pada titik tengah.

Pencarian formulasi menggunakan metode substraktif relatif. Koordinat pada gambar umumnya dimulai dari titik paling kiri dan paling atas (atau pojok kiri atas). Sehingga, semakin ke kanan maka semakin mendekati jangkauan terjauh pixel yaitu 1920, dan semakin ke bawah maka mendekati 1080, hal ini dikarenakan citra yang dihasilkan memiliki resolusi 1920x1080 dengan rasio 16:9. Dengan demikian, titik tengah seharusnya ada pada 960,5 dan 540,5. Namun karena tidak ada nilai tersebut dan tidak memungkinkan untuk menandai titik tersebut pada citra, pendekatan yang dilakukan adalah mencari rerata secara kalkulasi. Untuk kebutuhan penjelasan, titik tengah ditetapkan sebagai 960 dan 540.



Gambar 3.12 Titik-titik penting pada citra di tengah, titik (0, 0) dan titik maksimal (1920, 1080)

Nilai titik tengah yang didapat kemudian menjadi suku substraktif. Pada formula, dapat dipahami bahwa hasil alamat di tiap sumbu akan dikurangi masing-masing suku substraktifnya. Pada sumbu x, suku substraktifnya adalah 960 dan 540 pada sumbu y. Sehingga, jika pada citra terdeteksi seorang manusia pada titik

(1035, 756) dalam koordinat citra, maka dalam koordinat pengukuran, ia berada pada (75, 216). Jika pada citra terdeteksi seorang manusia pada titik (549, 381), maka pada koordinat pengukuran ia ada pada (-411, -159).

$$X(x) = x - 960$$

$$Y(y) = y - 540$$

Gambar 3.6 – Formula ekstraksi posisi titik dari titik tengah

Dengan formula seperti ini, koordinat pengukuran akan terbagi menjadi empat (4) kuadran dengan konfigurasi yang berbeda dari diagram dua sumbu yang biasa dibuat. Pada bagian kiri atas, koordinat berkonfigurasi (negatif, negatif), kiri bawah adalah (negatif, positif), kanan bawah adalah (positif, positif), kanan atas adalah (positif, negatif).