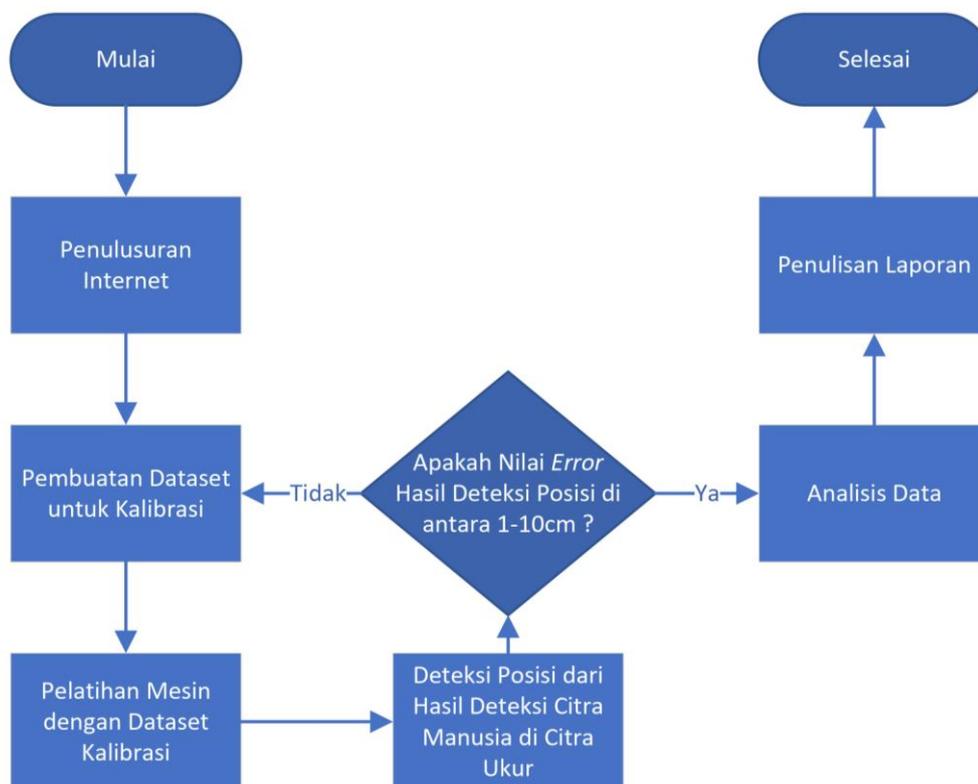


BAB III METODE PENELITIAN

3.1 Prosedur Penelitian

Kalibrasi pada Penelitian ini dilakukan dengan menggunakan *dataset* beranotasi di kaki. *Dataset* dikumpulkan dari Penelusuran Internet dan *Dataset* penelitian Kurniawan, *Dataset* yang dikumpulkan lalu dianotasi di bagian kaki dengan bantuan alat berbasis web makesense.ai. Setelah proses anotasi selesai, *dataset* dipakai dalam proses pelatihan model. Setelah model dilatih, model akan dipakai untuk mendeteksi citra manusia dari data ukur. Hasil dari pendeteksian kemudian diolah dalam microsoft excel yang menjadi jarak dari titik tengah gambar ke hasil prediksi posisi manusia, jarak prediksi posisi kemudian dibandingkan dengan nilai riil nya yaitu 100cm untuk dikehui tingkat keakurasian dari hasil kalibrasi. Performa dari mesin hasil kalibrasi ini kemudian di bandingkan dengan performa mesin yolov4-csp-swish dari pustaka darknet.



Gambar 3.1 Alur Penelitian

3.2 Alat dan Bahan

Personal Computer (PC) Asus A409UJ adalah alat yang digunakan dalam penelitian ini dengan aplikasi pencatat program daring *Google Colab* yang terintegrasi dengan *Cloud Computer* milik *Google*. Karena spesifikasi PC yang kurang mumpuni, penggunaan mesin deteksi citra tidak dilakukan dalam PC melainkan dalam *Google Colab Pro* yang memiliki spesifikasi yang lebih tinggi. Adapun pencatatan luring pada PC dilakukan dengan perangkat lunak *Visual Studio Code* dengan format *Development Environment* Jupyter Notebook. *Microsoft Excel* dipakai untuk mengolah data.

Tabel 3.1 Spesifikasi *Cloud Computer* Google Collab Pro yang Dipakai

CPU	2 vCPU Intel Xeon 2,2 GHz
RAM	12,7 GB
GPU	Nvidia Tesla T4 15 GB
Penyimpanan	201.2 GB

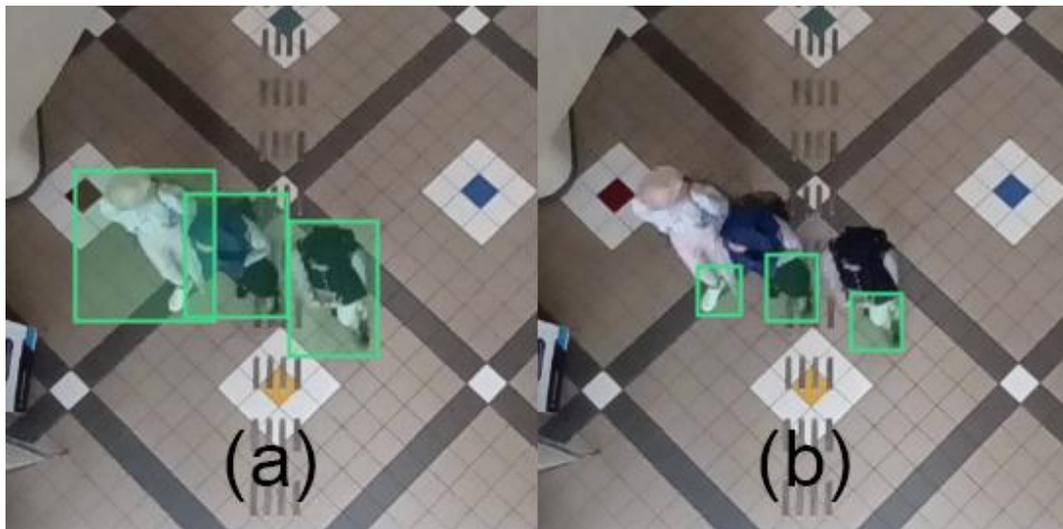
Bahan dari penelitian ini adalah *Dataset* tampak atas yang dikumpulkan dari hasil penelusuran internet, *dataset* UCUapp, dan *dataset* dari penelitian Kurniawan, semua citra akan di anotasi ulang. *Dataset* yang dikumpulkan akan berjumlah 2000.

3.3 Pembuatan *Dataset* untuk Kalibrasi

Sebelum gambar dipakai untuk anotasi, setiap gambar akan dibuat menjadi *potrait* dalam *preprocessing*. Anotasi dilakukan dengan ketentuan YOLO dengan bantuan laman *makesense.ai*. Setelah laman dibuka, pilih tombol “*Get Started*”, masukan foto dengan cara menyeret *file* foto ke box “*Drop Image*” atau klik box tersebut dan pilih foto sesuai alamat penyimpanan foto. Setelah itu akan diminta label apa yang digunakan dengan cara menuliskan di *box* atau dengan memasukan *file* label yang berisi label yang akan dipakai. Laman akan memunculkan citra foto yang akan dianotasi dengan daftar gambar dan alat-alat untuk membantu anotasi.

Penandaan dilakukan dengan mengarahkan *mouse* pada titik sudut dari kotak yang akan dibuat sebagai penanda, kemudian tahan klik kiri dan menariknya hingga

membentuk kotak. Arahkan *cursor* ke tengah kotak anotasi sehingga titik tengah diketahui. Jika titik tidak berada di tengah-tengah dua kaki, maka sesuaikan kotak dengan cara tahan klik kiri pada salah satu sudut hingga titik tengah sesuai keinginan, lalu isi label setiap kotak deteksi di sebelah kiri dengan label yang diinginkan.

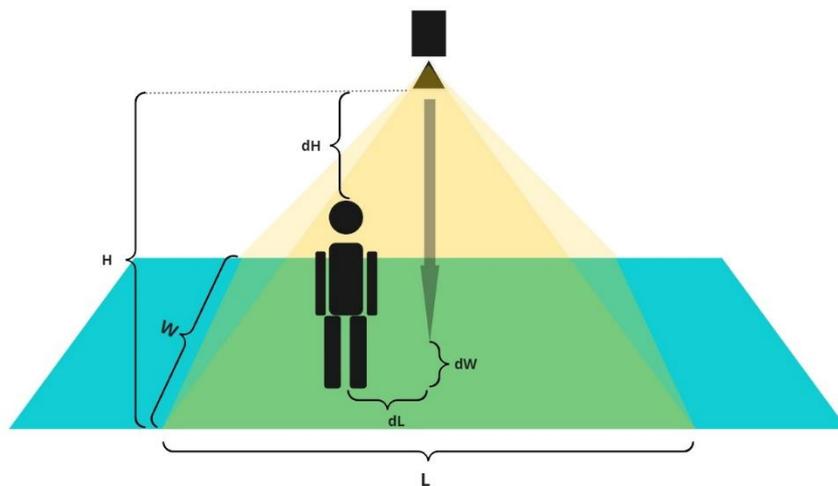


Gambar 3.2 Gambar dari *Dataset* Penelitian Kurniawan, Gambar Diambil di Gedung JICA-FPMIPA A, Anotasi Pada yang Digunakan di Mesin yolov4-csp-swish (a) dan Anotasi pada Mesin Penelitian Ini (b)

Setelah semua gambar dianotasi, klik menu *export* dan pilih format YOLO, lalu unduh *file* anotasi. Kumpulan *file* anotasi ini akan di himpun di *file* anotasi.zip, kemudian ekstrak *file* zip di folder yang sama dengan gambar untuk *dataset*.

3.3.1 Penelusuran Internet

Data citra yang akan dipakai sebagai *dataset* adalah *file* gambar dengan format JPEG atau JPG. Citra yang dipakai harus citra dengan tampak dari atas dan data mentahan. Proses penelusuran data di internet dilakukan di mesin pencari *Google*. Penelusuran dilakukan dengan pencaharian frasa “*overhead*”, “*human overhead*”, “*top view*” dengan opsi untuk menghilangkan kata “*fisheye*”.



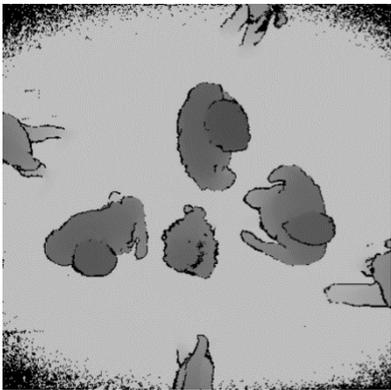
Gambar 3.3 Visualisasi Pengambilan Citra Tampak Atas

Diluar dari hasil temuan, peneliti menemukan empat *dataset* dengan satu tidak memiliki halaman web, yaitu data dari *Institute of Communication & Computer Systems* yang berasal dari riset “Self Configurable Cognitive Video Supervision”. Meski data ini tidak memiliki laman web, banyak penelitian menggunakan *dataset* ini yang diperoleh dengan cara meminta izin untuk mengakses data lewat email insititusi tersebut.

Empat *dataset* memiliki tampilan planar, dan satu dengan tampilan agak bulat karena merupakan data dari citra kamera yang berlensa *fisheye*. Dari semua data, hanya satu yang menggunakan kamera yang paling berbeda, yaitu *dataset* ke-3 dengan kamera *Kinect V2* yang memiliki mode lain, yaitu inframerah, sehingga *dataset* ini berisi gambar citra inframerah yang memiliki tampilan gradasi abu dan citra berwarna dari kamera RGB. Rata-rata data yang temukan adalah untuk keperluan menghitung jumlah manusia atau *people counting*. Terdapat dua *dataset* untuk keperluan ini yaitu “Overhead view Computer Vision Project” dari grup riset Chelkautun Nauka yang diperoleh dari rekaman kamera dalam sebuah bis dalam kota di Santiago Chile dan “Top View Person Re-identification - 2nd edition” dari Polytechnic University of Marche Via Brece Bianche. Data ini direkam dengan skenario per orang akan jalan dari sisi kiri ke kanan dan sebaliknya sampai hilang dari citra. Saat mendekati titik tengah subjek akan berhenti

sejenak lalu melanjutkan perjalanannya. Dalam rekaman citra manusia yang ada tidak lebih dari satu, dan subjek yang mengikuti perekaman data ini menggunakan pakaian dan aksesoris sehari-hari yang berbeda. Total dalam pembuatan data ini membutuhkan waktu 8 hari dengan 200 kali sesi pengambilan video yang melibatkan 2000 orang. Sementara itu lingkungan *dataset* SCOVIS sangat berbeda.

Tabel 3.2 Daftar *dataset* yang ditemukan di Internet

No	Nama - Penerbit	Contoh Data	Alamat Tautan
1	Overhead view Computer Vision Project – Chelkautun Nauka		https://universe.roboflow.com/chelkautun-nauka/overhead-view
2	TVPR\2 (Top View Person Re-identification - 2nd edition) – Polytechnic University of Marche Via Breccia Bianche		https://vrai.dii.univpm.it/content/tvpr2-dataset
3	Overhead Depth Images People Detection (GOTPD) – Electronic Engineering Applied to Intelligent Spaces		https://www.kaggle.com/datasets/lehomme/overhead-depth-images-people-

	and Transport Group		detection-gotpd1
4	SCOVIS (Self Configurable Cognitive Video Supervision) – Institute of Communication & Computer Systems		-

Setiap *dataset* ini dapat diunduh dengan cara yang berbeda. Pada *dataset* SCOVIS memerlukan mengirimkan email untuk meminta izin untuk mengunduh. Hal yang sama juga dibutuhkan untuk *dataset* TVPR/2.

Dataset GOTPD paling mudah diunduh. Pengguna hanya perlu mendaftar akun Kaggle lalu men-klik opsi *download*. Untuk *dataset* Chelkautun Nauka, akses untuk mengunduhnya diperoleh dengan menyalin citra kedalam penyimpanan akun Roboflow. Kemudian masukan semua citra kedalam seksi anotasi dan klik *mark as null* di logo “Ø”. Tunggu sampai proses selesai. Setelah itu pergi ke profil Roboflow dan pilih opsi “Project”, lalu pilih *dataset* Chelkautun Nauka dan klik opsi ekspor dengan format anotasi YOLOv4 Keras agar *file* yang diperoleh hanya memiliki satu *file* anotasi. *Dataset* lain yang dipakai adalah *dataset* Chelkautun Nauka, data citra ini di ambil dari ketinggian antara langit-langit dan lantai yang mencapai sekitar 3 m.

3.3.2 Penghimpunan *Dataset*

Data citra yang sudah dianotasi dikumpulkan dalam satu folder untuk bisa didaftarkan pada *file* .data untuk bisa dibaca oleh mesin, sehingga dapat dipakai untuk proses latih. Data citra dan *file* hasil anotasi dikumpulkan dalam satu folder di Google Drive dengan alamat lokasi ”MyDrive/Skripsi/anotasibiasa/darknet/data/v1/img”, buat *file* train.txt

untuk memberi alamat setiap citra dan data anotasi sehingga dapat dipakai dan simpan di "My Drive/Skripsi/anotasibiasa/darknet/data/v1".

```
data/v1/img/a200_v0(129).jpg
data/v1/img/a200_v0(130).jpg
data/v1/img/a200_v0(131).jpg
data/v1/img/a200_v0(132).jpg
data/v1/img/a200_v0(133).jpg
```

Gambar 3.4 Isi dari *File* train.txt

Kemudian buat *file.data* dan *v1.names*. untuk *v1.data*. Karena kelas yang digunakan hanya satu (human atau manusia), maka isi classes dengan angka satu, alamat train untuk *file* train.txt sebagai alamat *dataset* untuk latihan, isi valid dengan alamat *file.txt* yang diisi alamat citra dan anotasi untuk dijadikan perbandingan untuk mengukur akurasi prediksi, dan backup sebagai alamat penyimpanan hasil pelatihan mesin YOLO.

```
1 classes= 1
2 train = data/v1/train.txt
3 valid = data/v1/train.txt
4 names = data/v1/v1.names
5 backup = train
```

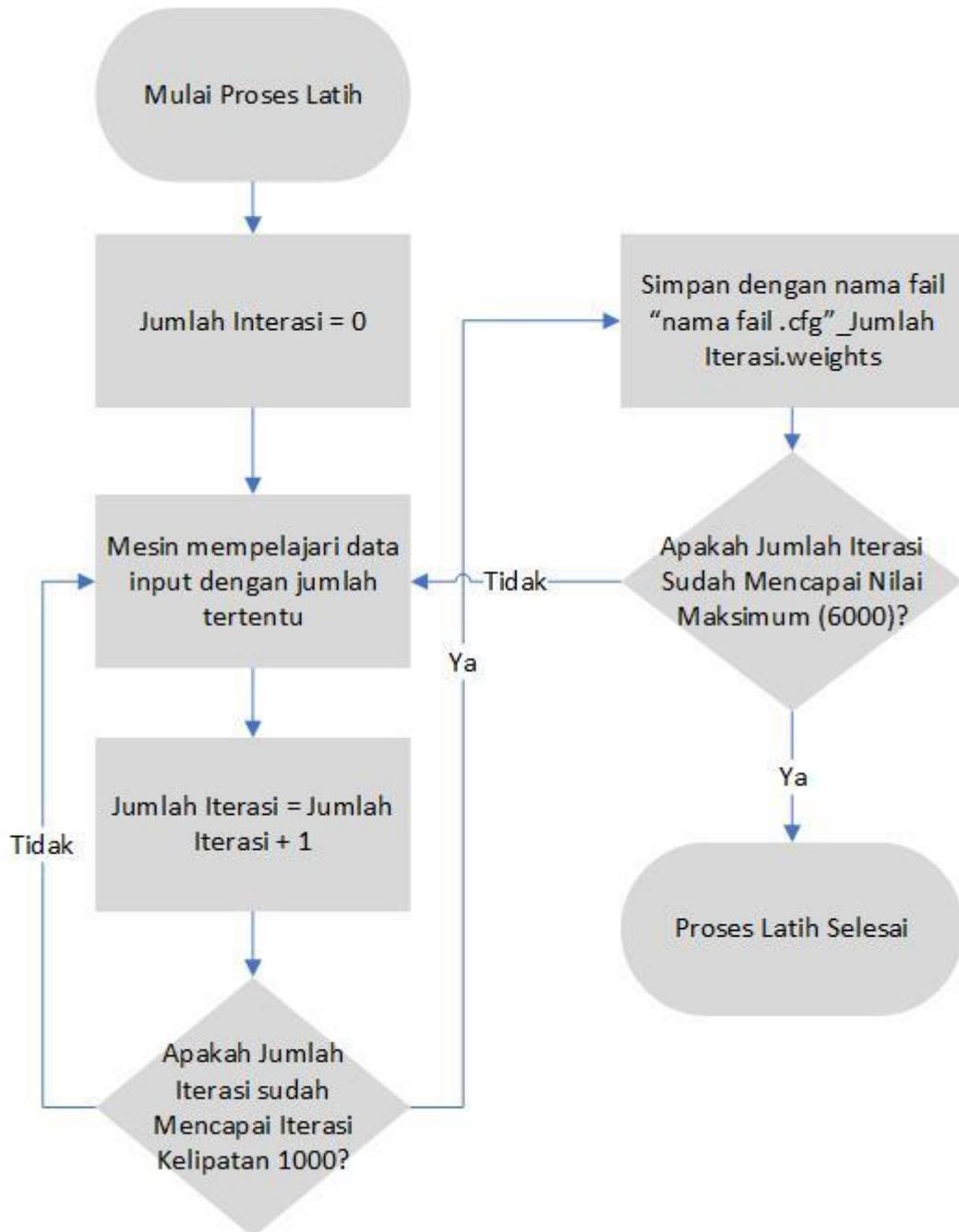
Gambar 3.5 Isi Dari *File* v1.data

3.4 Proses Latih Mesin yang Dikalibrasi

Sebelum proses latih mesin dilakukan dalam Google Colab, buat *file* colab pada *Google Drive*, lalu salin pustaka dan *file* weights yolov4.conv.137 Darknet dari Github Alexey AB. Lalu modifikasi *file* yolov4-custom.cfg sesuai dengan jumlah kelas berdasarkan ketentuan yang disebut di *Github Alexey AB*, kemudian buat perintah sebagai berikut. Setelah perintah dijalankan tunggu proses latih hingga selesai

```
!./darknet detector train data/v1/v1.data cfg/yolo4-khusus-cal.cfg yolov4.conv.137
```

Gambar 3.6 Perintah untuk Memulai Proses Latih



Gambar 3.7 Proses Latih Mesin

Model dilatih dengan 6000 iterasi dari *dataset* yang sudah dibuat. Setiap 1000 iterasi, algoritma akan otomatis menyimpan *file backup* hasil latihan, sehingga ketika proses latih terhenti, proses latih bisa dilanjutkan berdasarkan *file* "(nama *file* .cfg)_last.weights". Setelah mesin melakukan proses latih hingga 6000 iterasi, mesin akan meninjau hasil latihan berupa mAP(*mean average precision*) yang

merupakan parameter kualitas pendeteksian. Model dengan mAP yang tinggi akan digunakan sebagai pendeteksi citra manusia dalam penelitian ini.

```
!./darknet detector train data/v1/v1.data cfg/yolo4-khusus-cal.cfg train/yolo4-khusus-cal_last.weights
```

Gambar 3.8 Perintah untuk Melanjutkan Proses Latih

3.5 Perhitungan Posisi Manusia dan Perbandingan Hasil

Posisi dari titik tengah prediksi dicari berdasarkan titik acu yang ditentukan, titik acu ini adalah titik tengah gambar. Pada pengukuran di dunia nyata, titik acuannya mengikuti titik acu pada gambar. Titik tengah pada gambar ini dicari koordinat *pixel*nya, lalu ditentukan formulasi untuk menghasilkan koordinat baru yang terbagi pada titik tengah.

Pencarian formulasi menggunakan metode substraktif relatif. Koordinat pada gambar umumnya dimulai dari titik pojok kiri atas. Artinya jika resolusi gambar adalah 1920x1080, semakin ke kanan, *pixel* terjauhnya yaitu 1920, dan semakin ke bawah *pixel* terjauhnya 1080. Titik tengah pada citra seharusnya berada pada 960,5 dan 540,5. Namun titik tersebut tidak memungkinkan untuk ditandai pada citra. Oleh karena itu pendekatan yang dilakukan adalah mencari rata-rata secara kalkulasi. Titik tengah ditetapkan sebagai 960 dan 540 demi keperluan penjelasan.

Pada formula ini, hasil alamat di tiap sumbu akan dikurangi masing-masing suku substraktifnya dan nilainya dimutlakkan. Nilai titik tengah citra yang diperoleh menjadi suku substraktifnya. Pada kedua sumbu x dan y, suku substraktifnya adalah 960 dan 540. Sehingga jika citra manusia terdeteksi pada titik (910, 590) dalam koordinat citra, maka dalam koordinat pengukurannya berada pada (-50, 40). Data ukur yang digunakan dengan sebelas gambar ukur yang jarak dari citra manusia ke titik tengah gambar sebesar satu meter dengan orientasi dan lokasi yang berbeda-beda. Terdapat foto citra manusia yang berada di pojok kanan atas, kanan bawah, kiri atas, kiri bawah, dan kiri. Nama dari *file* gambar ukur diubah berdasarkan posisi dari citra manusia yang akan diukur. Posisi ini akan dinamai berdasarkan gambar dengan grid 3x3 dan urutan iterasi di setiap posisi. Contoh: TR-

1 (posisi *Top Right* dan iterasi ke-1). Data ukur memiliki dimensi 1280x720, titik tengah nya berada di (640, 360) dalam satuan *pixel*

Top Left	Top Center	Top Right
Middle Left	Center	Middle Right
Bottom Left	Bottom Center	Bottom Right

Gambar 3.9 Grid 3x3 dengan nama posisi disetiap sel

$$X(x) = x - 640 \quad (3.1)$$

$$Y(y) = y - 360 \quad (3.2)$$

Dalam hasil pendeteksian posisi citra akan diperoleh nilai *left_x*, *top_y*, *width*, dan *height* yang digunakan untuk mendapat koordinat titik tengah dari *bounding box*. Nilai tersebut diperoleh dari hasil prediksi. Setelah itu nilai-nilai hasil prediksi dimasukan dalam rumus untuk dicari titik tengahnya. Setelah titik tengah diperoleh, jarak antara titik tengah hasil prediksi dan titik tengah gambar bisa diperoleh. Setelah nilai ini diperoleh, bandingkan dengan nilai riilnya.

$$\text{Koordinat Hasil deteksi}(x,y) = \left(x_{\text{left}} + \left(\frac{\text{width}}{2} \right), y_{\text{top}} + \left(\frac{\text{height}}{2} \right) \right) \quad (3.3)$$

Selanjutnya konversi nilai jarak riil ke *pixel*. Konversi ini memakai nilai rasio konversi yang terukur pada gambar ukur secara riil dan *pixel*. Terukur nilai riil dengan panjang 395,0 cm dan lebar 221,8 cm dan nilai panjang pada gambar 1280 *pixel* dengan lebar 720 *pixel*.

$$\text{Rasio Konversi} \left(\frac{\text{cm}}{\text{pixel}} \right) = \frac{\text{Nilai Ukur (cm)}}{\text{Nilai Ukur (pixel)}} \quad (3.4)$$

Tabel 3.3 Rasio Konversi yang Dipakai dalam Penelitian

Rasio Sumbu X	Rasio Sumbu Y
0,30859375	0,308055556

Dari hasil deteksi citra manusia di kesebelas gambar ukur diperoleh koordinat masing-masing deteksi dengan label “Human”, hasil dari deteksi ini adalah *bounding box* dengan titik tengah atau yang biasa disebut *anchor point*. Dari koordinat *anchor point* jarak hasil deteksi dengan titik acu yaitu titik tengah gambar dapat diperoleh dengan menggunakan Pythagoras dengan persamaan 3.5 untuk dicari jarak dalam satuan *pixel*, persamaan ini dikonversi ke satuan centimeter dengan menggunakan rasio konversi pada tabel 3.3 untuk membuat persamaan 3.6 yang digunakan untuk mencari jarak dengan koordinat dalam satuan centimeter.

$$\text{Jarak(px)} = \sqrt{(x_{\text{px}} - 640)^2 + (y_{\text{px}} - 360)^2} \quad (3.5)$$

$$\text{Jarak(cm)} = \sqrt{(x_{\text{cm}} - 197,5)^2 + (y_{\text{cm}} - 110,9)^2} \quad (3.6)$$

3.5.1 Analisis Akurasi Mesin

Setelah jarak dari anchor point ke titik tengah semua gambar ukur diperoleh, akurasi dari setiap prediksi dapat diperoleh dari pengurangan 100% oleh presentase *error*. Presentase *error* diperoleh dari perbandingan jarak riil (100cm) dengan selisih jarak hasil prediksi dengan jarak riil.

$$\text{Error}\% = \left| \frac{\text{Jarak(cm)} - \text{Jarak riil}}{\text{Jarak riil}} \right| \times 100\% \quad (3.7)$$

$$\text{Akurasi}\% = 100\% - \text{Error}\% \quad (3.8)$$

Untuk menilai apakah mesin sudah sesuai untuk digunakan dalam gudang dengan barang kecil, nilai *error* dari yang terkecil dan terbesar dibandingkan dengan nilai akurasi yang dibutuhkan dalam gudang dengan barang kecil yaitu 1 sampai 10cm (Richards, 2017). Bila mesin sudah memiliki akurasi dengan *error* diantara 1 sampai 10cm, mesin pendeteksi

citra manusia sudah bisa digunakan dalam *indoor positioning system* gudang dengan barang kecil.

3.5.2 Analisis Perbandingan Mesin Kalibrasi dan Mesin Darknet

Mesin hasil kalibrasi dan yolov4-csp-swish dari pustaka darknet dibandingkan dalam hal keakurasian hasil deteksi, rata-rata *mean avarage precision* semua iterasi untuk menilai karakteristik hasil latihan, dan uji deteksi gambar baru untuk menilai apakah mesin bisa beradaptasi pada gambar dengan karakteristik yang berbeda.