

BAB IV HASIL DAN PEMBAHASAN

4.1 Instalasi *Raspberry Pi*

4.1.1 Instalasi *Operating System (OS)* pada *Raspberry Pi*

Sebuah sistem operasi adalah satu set program dasar dan program kegunaan (*utility*) yang membuat *Raspberry Pi* dapat bekerja. Ada beberapa jenis OS yang dapat diinstall pada *Raspberry Pi*, diantaranya adalah NOOBS, *Raspbian*, *Pidora*, *OpenELEC*, *RaspBMC*, dan *RISC OS*. Dari beberapa OS tersebut, sistem operasi yang digunakan pada *Raspberry Pi* penulis adalah *Raspbian*.

Raspbian adalah sistem operasi gratis berbasis *Debian GNU / Linux* dan dioptimalkan untuk perangkat keras *Raspberry Pi* (arsitektur prosesor *armhf*). *Raspbian* dilengkapi dengan lebih dari 35.000 paket, atau perangkat lunak *pre-compiled* paket dalam format yang bagus untuk kemudahan instalasi pada *Raspberry Pi*. Berikut ini cara instalasi OS *Raspbian* pada *Raspberry Pi*.

- a. *Download* file ISO *Raspbian* pada link berikut <https://www.raspberrypi.org/downloads/raspbian/> dan *extract* file zip tersebut sehingga mengeluarkan file dalam bentuk *.img* *Raspbian Jessie*.
- b. *Download* Win32 Disk Imager. Selanjutnya *install* hingga selesai.
- c. Memasukkan *micro* SD Card yang akan dijadikan *hard disk* bagi *Raspberry Pi* ke dalam laptop melalui *card reader*. Selanjutnya format *micro* SD Card terlebih dahulu sebelum digunakan.

- d. Menjalankan *Win32 Disk Imager* lalu *browse* file *.img Raspbian Jessie* yang telah *diextract* tadi
- e. Pilih *drive path storage* dari *SD Card* yang digunakan.
- f. Memilih *write* dan menunggu hingga proses selesai.
- g. Saat proses telah selesai, melepaskan *micro SD Card* dari laptop lalu memasukkan ke dalam slot *SD Card* pada *Raspberry Pi*.
- h. Selanjutnya *Raspberry Pi* sudah siap untuk digunakan.

4.1.2 Uji Fungsi Webcam dengan menggunakan Raspberry Pi

Apabila *SD Card* telah *diinstall* OS maka selanjutnya menguji apakah *Raspberry Pi* sudah siap digunakan atau belum dengan cara sebagai berikut.

- a. Memasukkan *micro SD Card* ke dalam slot *SD Card* pada *Raspberry Pi*, kemudian menghubungkan *keyboard*, *mouse*, kabel konektor HDMI, dan *webcam*.
- b. Menyalakan *Raspberry Pi* dengan cara menghubungkan kabel *power* pada *Raspberry Pi* dengan keluaran 1 Ampere.
- c. *Raspberry Pi* akan *booting* dan kemudian akan masuk ke tampilan awal yaitu *desktop*
- d. Mencoba apakah *webcam* dapat digunakan dengan baik, karena *webcam* merupakan komponen utama yang digunakan dalam pembuatan modul, jadi *webcam* harus terinstal dalam *Raspberry Pi*. Untuk menginstal *webcam* pada *Raspberry Pi* dengan memasukkan *listing* program berikut pada terminal *Raspberry Pi*.

```
sudo apt-get install fswebcam
```

Listing 4.1 Program untuk *Install Webcam*

Selanjutnya dilakukan percobaan cara kerja *webcam* dengan memasukkan Listing 4.2. hal tersebut dilakukan untuk mencoba menjalankan apakah *webcam* sudah dapat diakses pada *Raspberry* atau belum. Program tersebut akan menangkap gambar dari *webcam* dan menghasilkan gambar dengan ukuran 640x480 dengan nama *image.jpg*.

```
fswebcam -r 640x480 --no-banner image.jpg
```

Listing 4.2 Program untuk Mencoba *Webcam*

Listing 4.3 bertujuan untuk melihat gambar yang ditangkap oleh kamera *webcam* yang telah tersimpan otomatis pada *Raspberry* di folder *pi*.

```
ls -l          # verify "image.jpg" is there
```

Listing 4.3 Program untuk melihat hasil gambar

4.1.3 Instalasi *Plug in OpenCV* pada *Raspberry Pi*

OpenCV adalah suatu *plug in* atau *software* tambahan yang dapat digunakan dalam pemrograman-pemrograman yang berbasis C ++, C, *Python* dan *Java interface* dan mendukung *Windows*, *Linux*, *Mac OS*, *iOS* dan *Android*. *OpenCV* akan digunakan pada *Raspberry* agar kamera dapat menangkap bentuk lingkaran sesuai yang dibutuhkan. Berikut ini adalah *listing* program untuk penginstallan *OpenCV* pada *Raspberry Pi*.

Langkah pertama adalah melakukan *update* dan *upgrade* pada *Raspberry Pi*, hal ini bertujuan agar *software* yang telah terinstall pada *Raspberry Pi* mendapatkan versi terbaru. Berikut ini *listing* programnya.

```
sudo apt-get update
sudo apt-get upgrade
```

Listing 4.4 *Update dan Upgrade Raspberry Pi*

Kemudian dilakukan penginstalan *OpenCV* pada *Raspberry Pi*. *OpenCV* sangat penting untuk di *install* pada *Raspberry Pi* karena *OpenCV* merupakan *plug in* yang dapat membantu untuk mendeteksi lingkaran sebagai awal untuk menentukan keliling pada lingkaran kepala bayi. Untuk menginstall *OpenCV* dilakukan dengan memasukkan perintah pada terminal sebagai berikut.

```
sudo apt-get install libopencv-dev python-opencv
```

Listing 4.5 Instalasi *OpenCV*

Langkah selanjutnya adalah *download* versi terbaru dari *library* pada *OpenCV* yaitu *OpenCV* dengan versi 3.0.0 dengan memasukkan perintah berikut.

```
wget http://sourceforge.net/projects/opencvlibrary/files/opencv-unix/3.0.0/opencv-3.0.0.zip
```

Listing 4.6 *Download Versi Terbaru OpenCV*

Setelah itu melakukan *reboot* pada *Raspberry Pi* dengan memasukkan perintah berikut. *Reboot* ini bertujuan agar *software* yang baru saja di *install* dapat dijalankan dengan baik setelah di *reboot*.

```
sudo shutdown -r now
```

Listing 4.7 Reboot Raspberry Pi

4.1.4 Instalasi Software Qt Creator pada Raspberry Pi

Qt Creator adalah *cross-platform* yang mana *library* di dalam Qt itu dituliskan dalam bentuk C++ dan QML. Kelebihan dari *software* ini adalah mempunyai kemampuan dalam membentuk suatu aplikasi baru yang berbasis pada *Graphic User Interface* (GUI). GUI merupakan tampilan grafis yang mengandung alat-alat atau komponen-komponen yang memungkinkan pengguna untuk melakukan pekerjaan interaktif.

Dalam membuat program deteksi keliling lingkaran, penulis menggunakan *software Qt Creator*. Oleh karena itu, pada *Raspberry Pi* perlu di *install software Qt Creator* tersebut dengan *listing* program sebagai berikut.

```
sudo apt-get install qt4-dev-tools  
sudo apt-get install qtcreator
```

Listing 4.8 Instalasi Qt Creator

4.1.5 Instalasi LCD 4,3 inch pada Raspberry Pi

LCD 4,3 *inch* merupakan LCD HDMI dengan layar *touchscreen* yang digunakan penulis untuk menampilkan data hasil pengukuran dari modul. Untuk dapat menghubungkan *Raspberry Pi* dengan LCD 4,3 *inch*, perlu adanya instalasi LCD 4,3 *inch* pada *Raspberry Pi*.

Langkah pertama yang harus dilakukan adalah *download* master file lcd 4,3 *inch* dengan bentuk file *tar.gz* dan kemudian di *extract* pada *Raspberry Pi*. Langkah pertama, masuk ke dalam folder LCD-show dengan perintah berikut.

```
cd LCD-show
```

Listing 4.9 Masuk ke Folder LCD-show

Langkah selanjutnya adalah *install driver* LCD 4,3 *inch* dengan memasukkan perintah berikut pada terminal.

```
./LCD43-show
```

Listing 4.10 Program Instalasi LCD 4,3 *inch*

Setelah itu *Raspberry* akan *reboot* secara otomatis. Kemudian LCD 4,3 *inch* sudah terinstal pada *Raspberry Pi*, sehingga keduanya dapat digunakan.

4.2 Pemrograman

Sebuah lingkaran adalah himpunan semua titik pada bidang dalam jarak tertentu, yang disebut jari-jari atau radius, dari suatu titik tertentu, yang disebut pusat. Sebuah lingkaran memiliki jari-jari, diameter dan keliling. Jari-jari lingkaran merupakan garis lurus yang menghubungkan titik pusat dengan lingkaran. Sedangkan diameter merupakan tali busur terbesar yang panjangnya adalah dua kali dari jari-jarinya. Diameter ini membagi lingkaran sama luas. Jika kita menganalogikan sebuah tali sebagai lingkaran, maka saat lingkaran tersebut dibuka, maka keliling lingkaran tersebut merupakan panjang dari tali yang membentuk lingkaran tersebut. Berikut ini merupakan Rumus keliling Lingkaran.

$$K = 2\pi r \quad (4-3)$$

Untuk melakukan pengukuran lingkar kepala bayi, diperlukan program untuk mendeteksi lingkaran yang selanjutnya akan dimasukkan rumus keliling lingkaran agar hasil kelilingnya dapat ditampilkan. Berikut ini merupakan program yang digunakan dalam modul.

```
CvCapture *capture_ = cvCaptureFromCAM( 0 );
frame = cvQueryFrame( capture_ );
```

Listing 4.11 Fungsi *Real Time* Video

Fungsi `cvQueryFrame` merupakan fungsi untuk mendapatkan *live* video agar video yang didapatkan dari kamera *webcam* adalah video *real time*. Jika fungsi tersebut tidak ada, maka yang didapatkan hanya *capture* atau foto saja bukan video *real time*. Sedangkan yang dibutuhkan adalah video *real time* yang dapat terus menerus memonitor lingkaran agar didapatkan keliling lingkaran yang sesuai. Fungsi tersebut adalah proses awal dalam modul ini, karena yang dibutuhkan adalah video *real time* untuk mendeteksi lingkaran yang didapatkan dari tangkapan kamera *webcam*.

```
qt_img_ =IplImage2QImage( img );
```

Listing 4.12 Mengubah dari `IplImage` ke `QImage`

`IplImage2QImage` merupakan suatu fungsi yang digunakan untuk mengubah dari `IplImage` ke `QImage`. `IplImage` adalah struktur dasar dalam OpenCV yang digunakan untuk mengkodekan suatu citra. Sedangkan `QImage` merupakan *library* dalam *Qt Creator* yang menyediakan perwakilan gambar yang memungkinkan akses langsung ke data piksel dan dapat digunakan sebagai perangkat melukis.

Dari Listing 4.12 diatas, `qt_img_` merupakan obyek dari `QImage`. Sedangkan `img` merupakan obyek dari `IplImage`. Konversi dari `IplImage` ke

QImage dilakukan karena citra data yang ada pada *library* OpenCV tidak cocok dan tidak dapat diaplikasikan pada *library* QImage yang ada pada Qt Creator.

```
InputLabel->setPixmap(QPixmap::fromImage(qt_img_));
```

Listing 4.13 Menampilkan Gambar pada Layar dengan Menggunakan QLabel

Setelah gambar telah diubah mengikuti aturan yang ada pada Qt Creator, kemudian menampilkan gambar pada layar. `InputLabel` merupakan suatu obyek dari `QLabel` yang digunakan untuk mengeluarkan hasil pada sebuah label pada layar *Grafik User Interface* (GUI). `QPixmap` merupakan suatu *library* yang dirancang dan dioptimalkan untuk menampilkan gambar pada layar. Jadi maksud dari Listing 4.13 diatas adalah untuk menampilkan gambar yang diambil dari `qt_img_` dan ditampilkan pada obyek `InputLabel`.

```
circles = getCirclesInImage(frame, storage, grayscaleImg);
```

Listing 4.14 Mendeteksi Lingkaran

Fungsi `getCirclesInImage` adalah sebagai fungsi untuk mendeteksi lingkaran dari gambar yang ditangkap oleh kamera *webcam*. Kata `circles` diatas sebagai obyek dari `CvSeq` yang merupakan sebuah fungsi untuk mendeteksi sesuatu, pada modul ini `CvSeq` digunakan untuk dapat mendeteksi lingkaran. Ketika fungsi `getCirclesInImage` ini dihilangkan, maka lingkaran yang tertangkap oleh kamera *webcam* tidak dapat dideteksi.

```
sprintf(label, "color: %s", "red");
drawCircleAndLabel(frame, p, label);
```

Listing 4.15 Menggambar Lingkaran saat Lingkaran Terdeteksi

Sedangkan *listing* program 4.15 diatas digunakan untuk menandai lingkaran yang dideteksi oleh kamera *webcam* dengan garis berwarna merah disekeliling lingkaran sehingga membentuk gambar lingkaran berwarna merah.

```
float *p = (float*)cvGetSeqElem( circles, i );
r = p[2];
r = p[2]*0.102;
```

Listing 4.16 Deteksi Diameter pada Lingkaran

Listing program diatas merupakan suatu program untuk dapat mengetahui nilai diameter pada lingkaran yang akan dideteksi oleh kamera *webcam*. Untuk menampilkan hasil dari perhitungan keliling tersebut, digunakan perintah sebagai berikut.

```
void HeadDetection::KelilingLing(float x)
{
    QFont font = Keliling->font();
    font.setPointSize(50);
    font.setBold(true);
    Keliling->setText("<font color='red'>" +
                    QString::number(x, 'f', 1) + "</font>" +
                    "<font color='yellow'>" + "cm" + "</font>");
}
```

Listing 4.17 Program Menampilkan Hasil Keliling pada Layar

`QFont` merupakan suatu *library* yang digunakan untuk menentukan *font* yang akan digunakan pada suatu teks. Fungsi pada `setText`, adalah untuk menerima pemrograman dari HTML. Sedangkan `Keliling` merupakan suatu obyek dari `QLabel`. Pendeklarasian tersebut dapat dilihat pada Listing dibawah ini.

```
QLabel *Keliling;
```

Listing 4.18 Deklarasi obyek Keliling

Pada Listing 4.17 teks yang akan ditentukan *font*-nya adalah teks yang ada pada label *Keliling*. Yang diatur adalah *font size* sebesar 50 pixel dan teks diatur dalam keadaan *bold*.

Kemudian fungsi `setText` diatas merupakan untuk mengatur huruf yang akan ditampilkan. Sehingga teks pada label yang muncul adalah sesuai dengan yang telah ditetapkan. Perintah tersebut memanfaatkan susunan perintah seperti pada pemrograman HTML yang digunakan untuk menambah *style* atau format pada teks.

Dalam penampilannya teks pertama yang merupakan bilangan dari hasil keliling lingkaran ditulis dengan menggunakan *font* berwarna merah, yang dalam program ditulis dalam bentuk `""`. Sedangkan teks kedua merupakan satuan cm yang ditulis dengan *font* berwarna kuning yang dalam program ditulis dalam bentuk `"<font color='yellow'"`.

4.3 Cara Perakitan Alat Secara Keseluruhan

Dalam pembuatan modul ini, langkah-langkah yang dilakukan untuk perakitan modul adalah sebagai berikut.

- a. Membuat box akrilik dengan ukuran 14,5cm x 13cm x 11cm
- b. Merangkai *Raspberry Pi* dengan *LCD Touchscreen* dengan cara menghubungkan pin GPIO pada *Raspberry* dengan konektor pada LCD
- c. Merakit box akrilik dengan cara di lem pada sisi bawah, samping dan belakang
- d. Memasukkan bahan-bahan yang dibutuhkan ke dalam box, seperti *Raspberry Pi*, *LCD Touchscreen*, *webcam*, baterai dan saklar

- e. Menghubungkan kamera *webcam*, catu daya dan konektor tambahan (untuk menghubungkan *mouse* dan/atau *keyboard*) pada *Raspberry Pi*
- f. Menyempurnakan modul dengan cara memasang box bagian atas dan kemudian di lem
- g. Modul siap digunakan dengan cara menekan saklar on/off ke posisi on untuk menyalakan modul

4.4 Standar Operasional Prosedur Penggunaan Alat

Dalam penggunaan alat, perlu adanya prosedur penggunaan alat, agar dapat memperpanjang umur alat dan mencegah kerusakan pada alat. Berikut ini merupakan standar operasional alat.

- a. Menyalakan alat dengan menekan saklar on/off ke posisi on
- b. Menempatkan alat sejajar dengan obyek yang akan diukur dengan jarak pengukuran antara alat dan obyek sejauh 65 cm
- c. Klik 2x pada file “Tayes2” yang ada pada *desktop*, setelah itu pilih *execute*
- d. Memasukkan data obyek yang akan diukur, berupa nama dan umur
- e. Melakukan pengukuran dan obyek yang diukur akan tampil pada layar dan nilai keliling lingkaran yang diukur akan tampil pada layar di bagian kiri atas
- f. Setelah obyek telah terukur dapat menekan tombol *pause* agar nilai yang dihasilkan dapat dilihat dengan stabil
- g. Setelah pengukuran selesai, pilih tombol keluar kemudian memilih *shutdown* untuk mematikan alat dan menekan saklar on/off ke posisi off

4.5 Pengukuran Lingkaran

Pada pengukuran modul ini menggunakan ukuran lingkaran dari 25,5 cm sampai dengan 48,2 cm. Hal ini didasarkan pada ukuran lingkaran kepala bayi normal usia 0-12 bulan adalah 31 cm - 47,5 cm menurut tabel lingkaran kepala bayi normal versi *World Health Organization* (WHO). Sedangkan untuk ukuran lingkaran dibawah ukuran normal tersebut hanyalah sebagai ukuran bahwa modul dapat mendeteksi lingkaran kepala bayi yang lebih kecil dari ukuran lingkaran kepala bayi normal (*mikrosefali*).

Pengukuran dilakukan pada jarak 45 cm, 60 cm dan 65 cm. Hal tersebut didasarkan pada resolusi LCD yang hanya 480x272 *pixel*, karena pada pengukuran dibawah jarak 45 cm lingkaran yang memiliki diameter paling kecil pun tidak terlihat secara sempurna dan modul tidak dapat menghitung nilai keliling lingkarannya. Pada saat pengukuran di jarak 45 cm, 60 cm dan 65 cm akan ditentukan jarak terbaik untuk pengukuran.

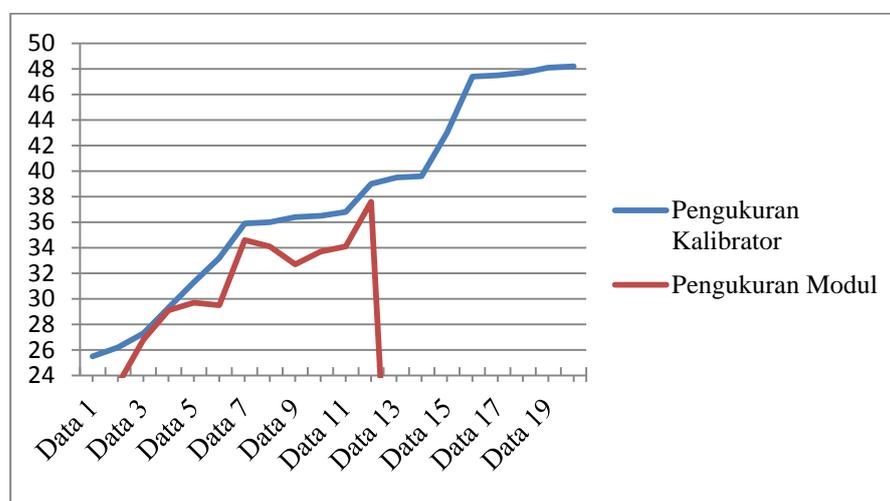
Pada saat melakukan pengukuran keliling lingkaran, diperlukan pencahayaan yang cukup, agar obyek yang dideteksi dapat terlihat dengan baik dan agar modul dapat mendeteksi lingkaran dari obyek dan dapat ditentukan nilai keliling lingkarannya. Pencahayaan yang kurang, akan menghambat pendeteksian obyek lingkaran oleh modul. Hal tersebut karena kamera *webcam* yang digunakan dengan spesifikasi resolusi kameranya hanya sekitar 0,5 *megapixel*. Berikut ini hasil pengukuran modul terhadap obyek lingkaran.

4.5.1 Pengukuran pada Jarak 45 cm

Tabel 4.1 Hasil Pengukuran Modul pada Jarak 45 cm

No	Data	Pengukuran Kalibrator (cm)	Pengukuran Modul (cm)			Rata-rata
			I	II	III	
1.	Data 1	25,5	23,2	22,1	22,6	22,6
2.	Data 2	26,2	23,3	23,3	23,4	23,3
3.	Data 3	27,3	26,7	26,3	27,5	26,8
4.	Data 4	29,3	29,3	29,4	28,6	29,1
5.	Data 5	31,3	30,4	29,4	29,4	29,7
6.	Data 6	33,2	29,4	29,8	29,2	29,5
7.	Data 7	35,9	35,3	34,3	34,2	34,6
8.	Data 8	36,0	31,9	31,9	31,4	34,1
9.	Data 9	36,4	32,2	32,1	33,9	32,7
10.	Data 10	36,5	34,5	33,6	33,1	33,7
11.	Data 11	36,8	34,2	34,0	34,2	34,1
12.	Data 12	39,0	37,7	37,6	37,4	37,6
13.	Data 13	39,5	-	-	-	-
14.	Data 14	39,6	-	-	-	-
15.	Data 15	43,0	-	-	-	-
16.	Data 16	47,4	-	-	-	-
17.	Data 17	47,5	-	-	-	-
18.	Data 18	47,7	-	-	-	-
19.	Data 19	48,1	-	-	-	-
20.	Data 20	48,2	-	-	-	-

a. Grafik Hasil Pengukuran Modul pada Jarak 45 cm



Gambar 4.1 Grafik Hasil Pengukuran pada Jarak 45 cm

b. Analisa Perhitungan

1. Rata-rata Data Kalibrator

$$\bar{X} = \frac{\sum Xi}{n}$$

$$(\bar{X}) = \frac{25,5+26,2+27,3+29,3+31,3+33,2+35,9+36,0+36,4+36,5+}{20}$$

$$\frac{36,8 + 39,0 + 39,5 + 39,6 + 43,0 + 47,4 + 47,5 + 47,7 + 48,1 + 48,2}{20}$$

$$(\bar{X}) = \frac{754,4}{20}$$

$$(\bar{X}) = 37,72$$

2. Rata-rata Data Modul

$$\bar{X} = \frac{\sum Xi}{n}$$

$$(\bar{X}) = \frac{22,6+23,3+26,8+29,1+29,7+29,5+34,6+34,1+32,7+33,7+}{20}$$

$$\frac{34,1 + 37,6}{20}$$

$$(\bar{X}) = \frac{367,8}{20}$$

$$(\bar{X}) = 18,39$$

3. Error

$$\% \text{ Error} = \frac{Xn - (\bar{X})}{Xn} \times 100\%$$

$$\% \text{ Error} = \frac{37,72 - (18,39)}{37,72} \times 100\%$$

$$\% \text{ Error} = \frac{19,33}{37,72} \times 100\%$$

$$\% \text{ Error} = 0,5125 \times 100\%$$

$$\% \text{ Error} = 51,25 \%$$

c. Uraian Hasil Pengukuran

Tabel 4.1 diatas merupakan tabel perbandingan hasil pengukuran data modul dan data kalibrator (pita meteran) pada jarak 45 cm. Pada pengukuran dengan jarak 45 cm didapat nilai error sebesar 51,25 %.

Sedangkan Gambar 4.1 adalah gambar grafik hasil perbandingan pengukuran antara modul dengan pita meteran dengan jarak pengukuran 45 cm. Dari gambar tersebut, dapat dilihat bahwa selisih yang dihasilkan oleh modul terhadap pita meteran cukup jauh.

Nilai error yang dihasilkan pada jarak pengukuran 45 cm sangat besar, hal tersebut menjadi pertimbangan bahwa pada jarak pengukuran antara obyek dengan modul sebesar 45 cm mendapatkan hasil yang kurang memuaskan, karena nilai error yang dihasilkan terlalu besar. Hal tersebut juga karena modul tidak dapat mendeteksi keliling lingkaran yang lebih besar dari 39,0 cm, sedangkan lingkar kepala bayi normal usia 0-12 bulan adalah 31 cm – 47,5 cm.

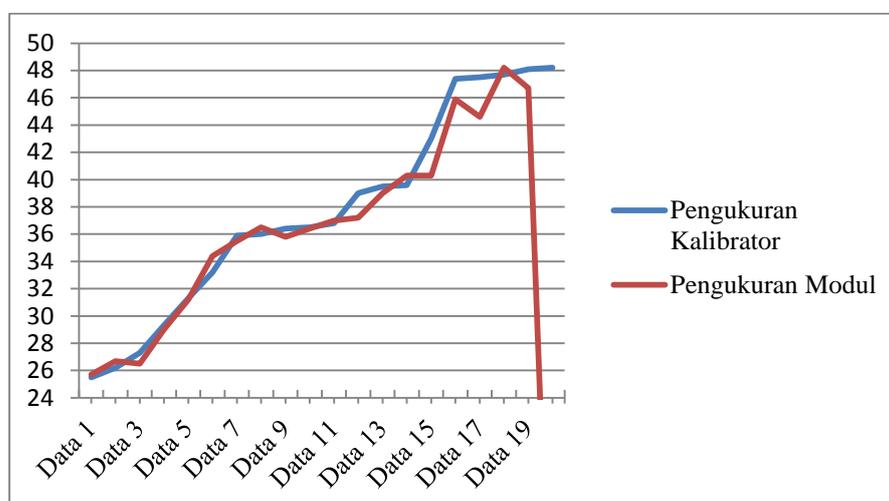
Dari data hasil pengukuran tersebut, menentukan pada jarak pengukuran antara modul dan obyek sejauh 45 cm merupakan jarak pengukuran yang kurang efisien.

4.5.2 Pengukuran pada Jarak 60 cm

Tabel 4.2 Hasil Pengukuran Modul pada Jarak 60 cm

No	Data	Pengukuran Kalibrator (cm)	Pengukuran Modul (cm)			Rata-rata
			I	II	III	
1.	Data 1	25,5	24,7	25,9	26,4	25,7
2.	Data 2	26,2	26,6	26,9	26,6	26,7
3.	Data 3	27,3	26,3	26,7	26,4	26,5
4.	Data 4	29,3	29,0	28,8	29,3	29,0
5.	Data 5	31,3	30,9	31,6	31,0	31,2
6.	Data 6	33,2	33,8	34,5	35,0	34,4
7.	Data 7	35,9	35,5	35,4	35,5	35,5
8.	Data 8	36,0	36,3	37,1	36,2	36,5
9.	Data 9	36,4	36,7	35,7	35,0	35,8
10.	Data 10	36,5	36,0	36,5	36,7	36,4
11.	Data 11	36,8	36,4	37,1	37,4	37,0
12.	Data 12	39,0	37,1	37,3	37,2	37,2
13.	Data 13	39,5	37,2	38,8	41,0	39,0
14.	Data 14	39,6	39,9	40,7	40,2	40,3
15.	Data 15	43,0	40,3	40,3	40,4	40,3
16.	Data 16	47,4	45,3	46,1	46,2	45,9
17.	Data 17	47,5	45,3	43,6	44,9	44,6
18.	Data 18	47,7	47,2	48,6	48,4	48,2
19.	Data 19	48,1	47,3	46,3	46,6	46,7
20.	Data 20	48,2	-	-	-	-

a. Grafik Hasil Pengukuran Modul pada Jarak 60 cm



Gambar 4.2 Grafik Hasil Pengukuran Modul pada Jarak 60 cm

b. Analisa Perhitungan

1. Rata-rata Data Kalibrator

$$\bar{X} = \frac{\sum Xi}{n}$$

$$\bar{X} = \frac{25,5+26,2+27,3+29,3+31,3+33,2+35,9+36,0+36,4+36,5+36,8+39,0+39,5+39,6+43,0+47,4+47,5+47,7+48,1+48,2}{20}$$

$$\bar{X} = \frac{754,4}{20}$$

$$\bar{X} = 37,72$$

2. Rata-rata Data Modul

$$\bar{X} = \frac{\sum Xi}{n}$$

$$\bar{X} = \frac{25,7+26,7+26,5+29,0+31,2+34,4+35,5+36,5+35,8+36,4+37,0+37,2+39,0+40,3+40,3+45,9+44,6+48,2+46,7}{20}$$

$$\bar{X} = \frac{696,9}{20}$$

$$\bar{X} = 34,845$$

3. Error

$$\% \text{ Error} = \frac{Xn - (\bar{X})}{Xn} \times 100\%$$

$$\% \text{ Error} = \frac{37,72 - (34,845)}{37,72} \times 100\%$$

$$\% \text{ Error} = \frac{2,875}{37,72} \times 100\%$$

$$\% \text{ Error} = 0,0762 \times 100\%$$

$$\% \text{ Error} = 7,62 \%$$

c. Uraian Hasil Pengukuran

Tabel 4.2 diatas merupakan tabel perbandingan hasil pengukuran data modul dan data kalibrator (pita meteran) pada jarak 60 cm. Pada jarak pengukuran 60 cm ini didapatkan nilai error sebesar 7,62 %.

Sedangkan Gambar 4.2 adalah gambar grafik hasil perbandingan pengukuran antara modul dengan pita meteran dengan jarak pengukuran 60 cm. Dari gambar tersebut, dapat dilihat bahwa selisih yang dihasilkan oleh modul terhadap pita meteran tidak terlalu jauh. Data yang dihasilkan lebih mendekati dengan pengukuran aslinya.

Pengukuran pada jarak antara obyek dengan modul sebesar 60 cm mendapatkan hasil yang lebih baik dibandingkan pada pengukuran dengan jarak 45 cm, akan tetapi nilai error yang dihasilkan masih terlalu besar. Pada jarak pengukuran 60 cm ini juga data ke-20 dengan ukuran keliling 48,2 tidak dapat dideteksi oleh modul.

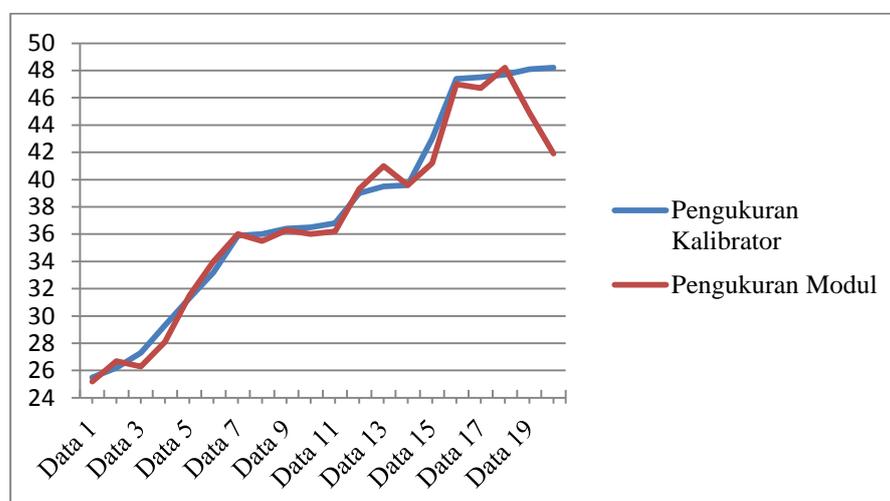
Dari hasil data pengukuran tersebut, penulis menentukan pada jarak pengukuran 60 cm ini, masih belum mendapatkan hasil pengukuran yang baik.

4.5.3 Pengukuran pada Jarak 65 cm

Tabel 4.3 Hasil Pengukuran Modul pada Jarak 65 cm

No	Data	Pengukuran Kalibrator (cm)	Pengukuran Modul (cm)			Rata-rata
			I	II	III	
1.	Data 1	25,5	25,1	25,3	25,3	25,2
2.	Data 2	26,2	26,7	26,7	26,8	26,7
3.	Data 3	27,3	26,2	26,3	26,8	26,4
4.	Data 4	29,3	28,1	28,0	28,1	28,1
5.	Data 5	31,3	31,6	31,9	31,0	31,5
6.	Data 6	33,2	33,8	34,2	34,1	34,0
7.	Data 7	35,9	36,0	35,7	36,2	36,0
8.	Data 8	36,0	35,7	35,4	35,4	35,5
9.	Data 9	36,4	36,2	36,1	36,5	36,3
10.	Data 10	36,5	35,7	36,2	35,9	36,0
11.	Data 11	36,8	36,3	36,2	36,0	36,2
12.	Data 12	39,0	39,4	39,3	39,3	39,3
13.	Data 13	39,5	41,1	41,0	41,0	41,0
14.	Data 14	39,6	39,5	40,1	39,1	39,6
15.	Data 15	43,0	41,5	41,0	41,0	41,2
16.	Data 16	47,4	47,2	46,9	47,6	47,0
17.	Data 17	47,5	46,7	46,7	46,7	46,7
18.	Data 18	47,7	48,2	48,3	48,1	48,2
19.	Data 19	48,1	45,4	44,7	44,5	44,9
20.	Data 20	48,2	42,1	42,0	41,5	41,9

a. Grafik Hasil Pengukuran pada Jarak 65 cm



Gambar 4.3 Grafik Hasil Pengukuran pada Jarak 65 cm

b. Analisa Perhitungan

1. Rata-rata Data Kalibrator

$$\bar{X} = \frac{\sum Xi}{n}$$

$$(\bar{X}) = \frac{25,5+26,2+27,3+29,3+31,3+33,2+35,9+36,0+36,4+36,5+}{20}$$

$$\frac{36,8 + 39,0 + 39,5 + 39,6 + 43,0 + 47,4 + 47,5 + 47,7 + 48,1 + 48,2}{20}$$

$$(\bar{X}) = \frac{754,4}{20}$$

$$(\bar{X}) = 37,72$$

2. Rata-rata Data Modul

$$\bar{X} = \frac{\sum Xi}{n}$$

$$(\bar{X}) = \frac{25,2+26,7+26,4+28,1+31,5+34,0+36,0+35,5+36,3+36,0+36,2}{20}$$

$$\frac{39,3 + 41,0 + 39,6 + 41,2 + 47,0 + 46,7 + 48,2 + 44,9 + 41,9}{20}$$

$$(\bar{X}) = \frac{741,7}{20}$$

$$(\bar{X}) = 37,085$$

3. Error

$$\% \text{ Error} = \frac{Xn - (\bar{X})}{Xn} \times 100\%$$

$$\% \text{ Error} = \frac{37,72 - (37,085)}{37,72} \times 100\%$$

$$\% \text{ Error} = \frac{0,635}{37,72} \times 100\%$$

$$\% \text{ Error} = 0,0168 \times 100\%$$

$$\% \text{ Error} = 1,70 \%$$

c. Uraian Hasil Pengukuran

Tabel 4.3 diatas merupakan tabel perbandingan hasil pengukuran data modul dan data kalibrator (pita meteran) pada jarak 65 cm. Pada jarak pengukuran 65 cm ini data yang didapatkan tidak memiliki selisih yang terlalu banyak dengan data kalibrator. Berdasarkan perhitungan error diatas didapat nilai error sebesar 1,70 %.

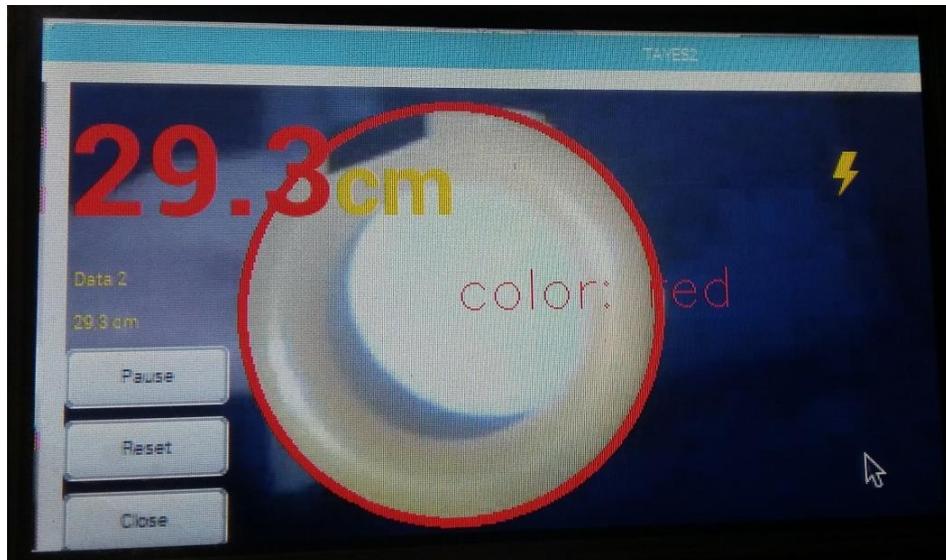
Sedangkan Gambar 4.3 adalah gambar grafik hasil perbandingan pengukuran antara modul dengan pita meteran dengan jarak pengukuran 65 cm. Dari gambar tersebut, dapat dilihat bahwa selisih yang dihasilkan oleh modul terhadap pita meteran tidak terlalu jauh. Data yang dihasilkan lebih mendekati dengan pengukuran aslinya.

Hal tersebut menjadi pertimbangan bahwa pada jarak antara obyek dengan modul sebesar 65 cm mendapatkan hasil yang lebih baik dibandingkan data pengukuran modul pada jarak 45 cm dan 60 cm.

Dari hasil pengukuran tersebut, penulis menentukan jarak pengukuran antara modul dengan obyek sejauh 65 cm. Hal tersebut dikarenakan nilai error yang dihasilkan adalah nilai error yang paling kecil dibandingkan pada jarak pengukuran yang lainnya.

4.6 Contoh Gambar Hasil Pengukuran

4.6.1 Hasil Pengukuran pada Jarak 45 cm



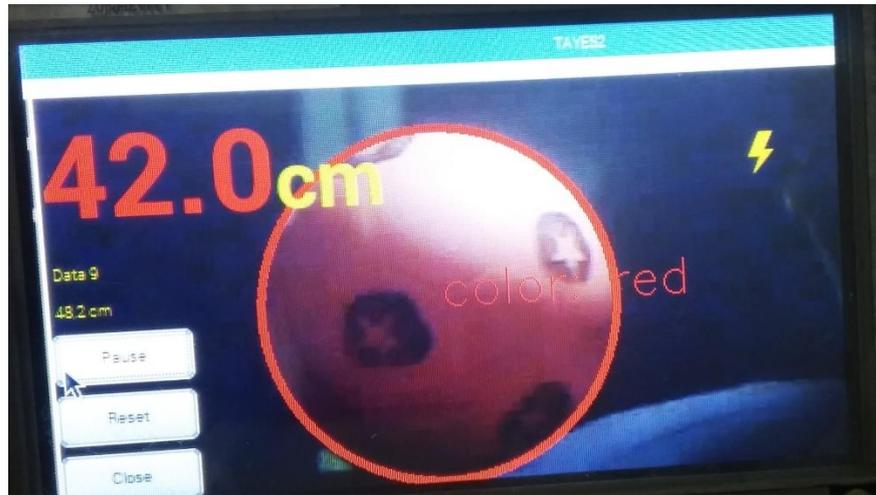
Gambar 4.4 Hasil Pengukuran pada Jarak 45 cm

4.6.2 Hasil Pengukuran pada Jarak 60 cm



Gambar 4.5 Hasil Pengukuran pada Jarak 60 cm

4.6.3 Hasil Pengukuran pada Jarak 65 cm



Gambar 4.6 Hasil Pengukuran pada Jarak 65 cm