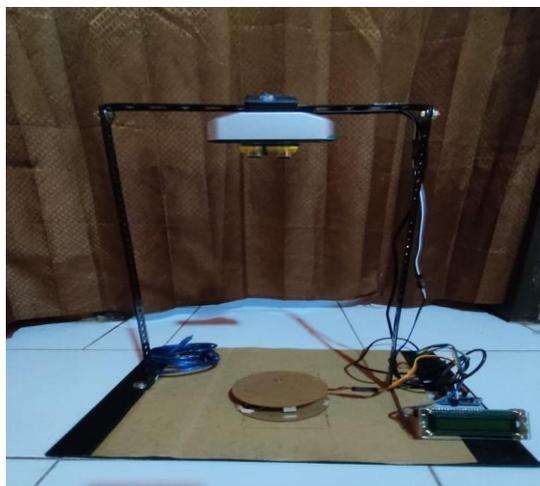


BAB V

IMPLEMENTASI DAN PENGUJIAN SISTEM

5.1 HASIL IMPLEMENTASI ALAT DAN PROGRAM

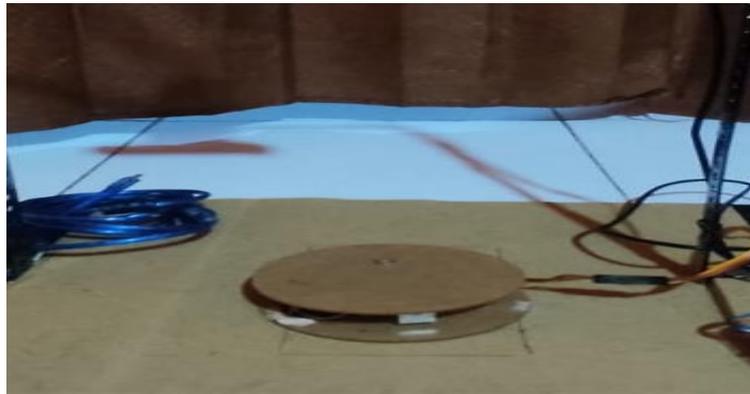
Hasil implementasi dari perancangan alat penyortir alat penyortir getah karet menggunakan pengolahan citra ini terdiri perancangan *hardware* dan *software*. Perancangan *hardware* yang digunakan yaitu *Arduino Nano* sebagai mikrokontroler, kamera digunakan sebagai alat pengambil gambar dari getah karet, modul sensor *HX711* digunakan sebagai sensor pendeteksi berat getah karet, *HC-SR04* digunakan sebagai pendeteksi ketinggian dari getah karet, *LCD* digunakan sebagai alat penampil hasil perbandingan berat getah karet. Sedangkan perancangan *software* dalam penelitian ini yaitu *visual studio code* yang digunakan sebagai *text editor* yang digunakan dalam pengkodean pengolahan citra, *Arduino IDE* digunakan untuk membuat *sket* program yang akan dimasukkan ke dalam *board* yang ingin digunakan. Berikut hasil implementasi alat keseluruhan dapat dilihat pada gambar 5.1



Gambar 5.1 Hasil Implementasi Alat Secara Keseluruhan

5.1.1 Implementasi *Loadcell*

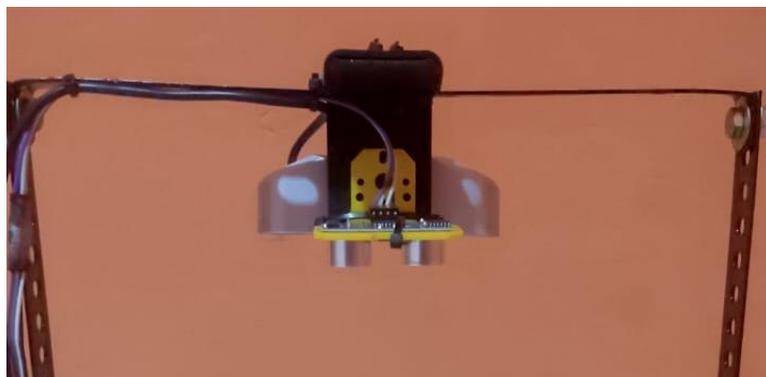
Pada *Loadcell* yang digunakan akan membaca berat pada getah karet. Untuk mengetahui berat getah karet yang didapat dapat dilihat dengan meletakkan getah karet diatas *Laodcell*. Implementasi *Loadcell* dapat dilihat pada gambar 5.2 :



Gambar 5.2 Implementasi *Loadcell*

5.1.2 Implementasi sensor *HC-SR04*

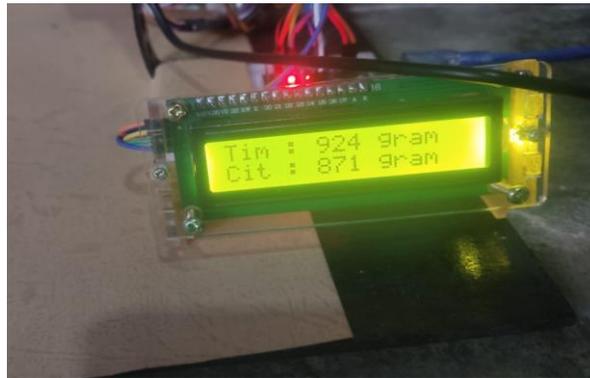
Pada sensor *HC-SR04* digunakan untuk mendapatkan tinggi dari getah karet, dengan perhitungan tinggi awal dikurang tinggi sekarang maka akan didapatkan tinggi dari getah karet tersebut. Hasil dari implementasi dari sensor *HC-SR04* dapat dilihat pada gambar 5.3 :



Gambar 5.3 Implementasi *HC-SR04*

5.1.3 Implementasi LCD

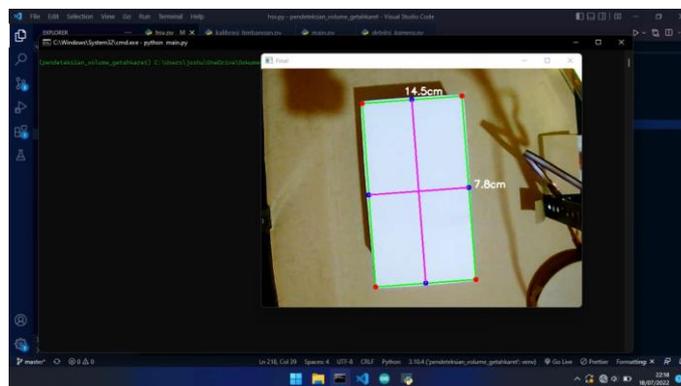
Pada LCD yang digunakan akan menampilkan berat getah telah dibaca oleh sensor *Loadcell*, nilai yang ditampilkan pada LCD tersebut akan dibandingkan dengan nilai dari pengolahan citra. Hasil dari implementasi pada tampilan LCD dapat dilihat pada gambar 5.4 :



Gambar 5.4 Implementasi Pada Tampilan LCD

5.1.4 Implementasi Program

Sistem pendeteksi objek getah karet menggunakan *OpenCV* dengan metode *HSV (Hue, Saturation, Value)*, perancangan program menggunakan bahasa pemrograman *Python* yang dijalankan pada sistem operasi *Windows* pada *software Visual Studio code*. Hasil implementasi *OpenCV* dapat dilihat pada gambar 5.5 :

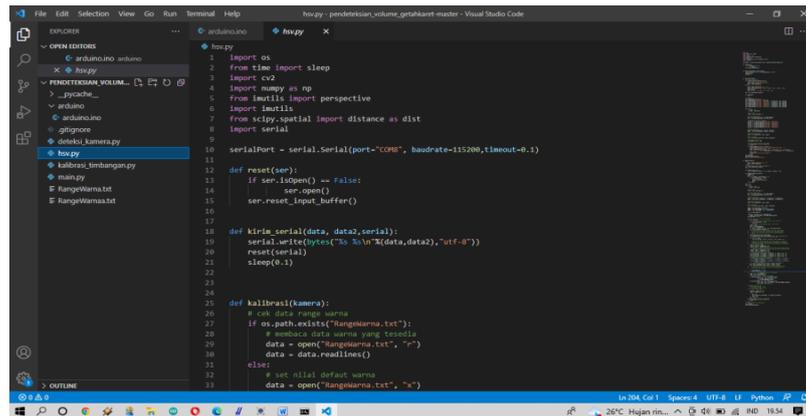


5.2.2 Pengujian Perangkat Lunak

5.2.2.1 Pengujian *Visual Studio Code*

Untuk pengujian dapat dilihat pada saat menjalankan program.

Pengujian dapat dilihat pada gambar 5.8 :



```

1 import os
2 from time import sleep
3 import cv2
4 import numpy as np
5 from imutils import perspective
6 import imutils
7 from scipy.spatial import distance as dist
8 import serial
9
10 serialPort = serial.Serial(port="/dev/ttyUSB0", baudrate=115200, timeout=0.1)
11
12 def reset(serial):
13     if serial.isopen() == False:
14         serial.open()
15         serial.reset_input_buffer()
16
17
18 def kirim_serial(data, data2, serial):
19     serial.write(bytes("%s %s\n" % (data, data2), "utf-8"))
20     reset(serial)
21     sleep(0.1)
22
23
24
25 def kalibrasi(kamera):
26     # cek data range warna
27     if os.path.exists("Rangasarna.txt"):
28         # membaca data warna yang tersedia
29         data = open("Rangasarna.txt", "r")
30         data = data.readlines()
31     else:
32         # set nilai default warna
33         data = open("Rangasarna.txt", "w")

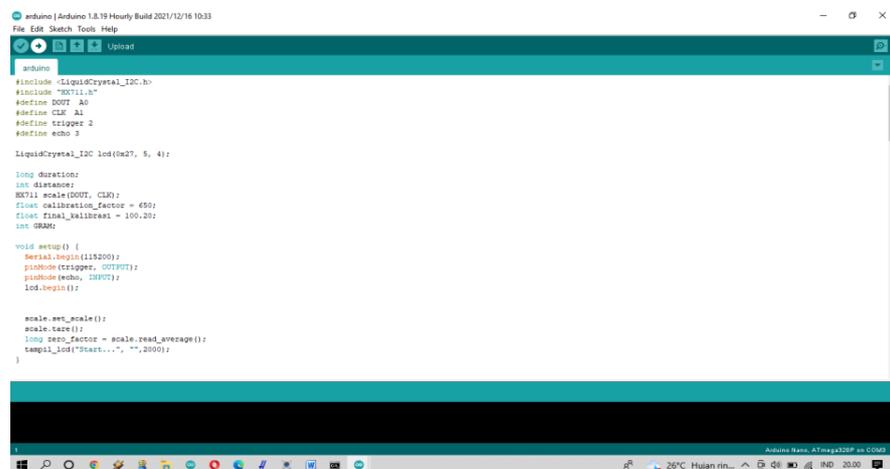
```

Gambar 5.7 Tampilan Pengujian *Visual Studio Code*

5.2.2.2 Pengujian *Arduino IDE*

Untuk pengujian dapat dilihat pada pengaplotan program ke dalam

board Arduino Nano. Pengujian dapat dilihat pada gambar 5.9 :



```

Arduino | Arduino 1.8.19 Hourly Build 2021/12/16 10:33
File Edit Sketch Tools Help
Upload
arduino
#include <LiquidCrystal_I2C.h>
#include <RTClib.h>
#define DOUT_A0
#define CLK_A1
#define trigger 2
#define echo 3

LiquidCrystal_I2C lcd(0x27, 5, 4);

long duration;
int distanc;
RTC * rca = new RTC(DOUT, CLK);
float calibration_factor = 650;
float final_calibration = 100.20;
int GRAM;

void setup() {
  Serial.begin(115200);
  pinMode(trigger, OUTPUT);
  pinMode(echo, INPUT);
  lcd.begin(16);

  scale_set_scale();
  scale_tare();
  long zero_factor = scale_read_average();
  temp1_ind("Dstart... ", "", 2000);
}

```

Gambar 5.8 Pengujian *Arduino IDE* Meng-*Upload* Data Ke *Arduinon Nano*

5.3 PENGUJIAN ALAT

Adapun rancangan alat yang digunakan dalam simulasi ini adalah sebagai berikut :

5.3.1 Pengujian Tegangan *Arduino Nano*

Rangkaian ini merupakan otak dari seluruh rangkaian. Semua rangkaian yang ada dikendalikan *input,output*-nya oleh rangkaian *mikrokontroler* ini. Adapun hasil dari pengujian tegangan *Arduino Nano* dapat dilihat pada table 5.1:

Table 5.1 Tegangan *Arduino Nano*

Sumber	Tegangan <i>Input</i>	Tegangan <i>Output</i>
<i>Adaptor</i>	5v	3.3v

5.3.2 Pengujian Tegangan *HX711*

Pengujian yang dilakukan untuk mendapatkan nilai berat pada *Loadcell* yaitu dengan meletakkan beban diatas *Loadcell*. Kemudian hasil dari pengecekan nilai bera dapat dilihat pada *serial monitor arduino IDE*. Hasil pengujian *Loadcell* dapat dilihat pada tabel 5.2 :

Table 5.2 Tegangan *HX711*

Sumber	Tegangan <i>Input</i>
<i>Adaptor</i>	5v

5.3.3 Pengujian Tegangan *HC-SR04*

HC-SR04 digunakan untuk mendeteksi ketinggian objek. Hasil pengujian tegangan *HC-SR04* dapat dilihat pada tabet 5.3 :

Table 5.3 Tegangan HC-SR04

Sumber	Tegangan <i>Input</i>
<i>Adaptor</i>	5v

5.3.4 Pengujian Tegangan LCD I2C

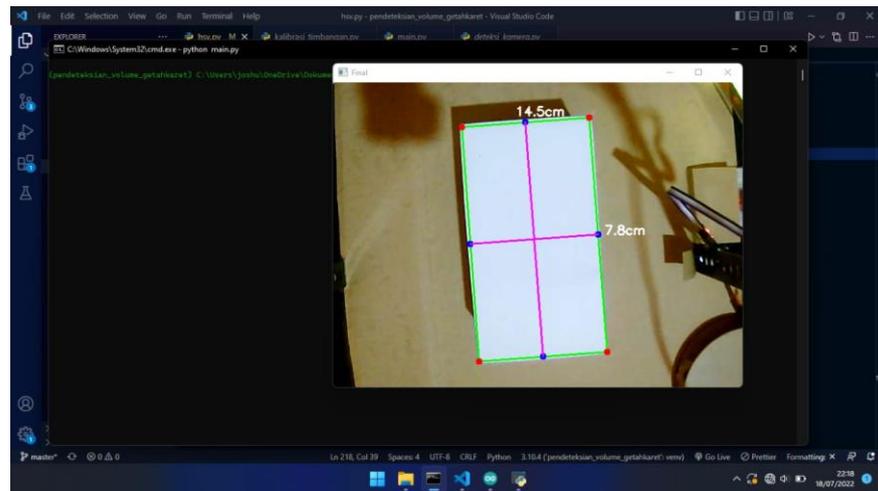
LCD digunakan sebagai alat untuk menampilkan hasil perbandingan antara berat getah dari pengolahan citra dan berat getah dari *Loadcell*. Hasil pengujian *LCD* dapat dilihat pada tabel 5.4 :

Table 5.4 Tegangan LCD I2C

Sumber	Tegangan <i>Input</i>
<i>Adaptor</i>	5v

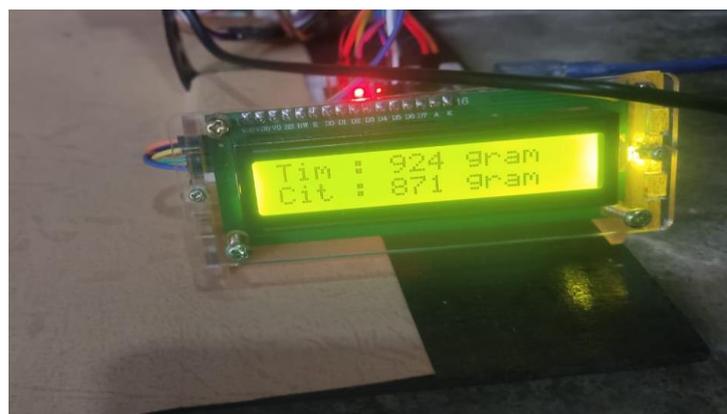
5.4 ANALISIS SISTEM SECARA KESELURUHAN

Untuk mendeteksi apabila terjadi kesalahan setelah uji coba, maka perlu dilakukan analisa rangkaian secara keseluruhan. Dari seluruh proses yang telah dilakukan, baik pengujian perangkat keras maupun perangkat lunak, dapat dikatakan bahwa alat ini dapat berfungsi sebagaimana yang diinginkan. Pengujian dilakukan dengan melakukan percobaan meletakkan getah karet dengan ukuran 15cm x 8cm x 8cm diatas *loadcell*. Yang mana *Loadcell* terletak dibagian bawah kamera dan sensor *HC-SR04*. Kamera dan *Loadcell* akan mendeteksi objek getah karet secara bersamaan, kemudian hasil visual dari kamera akan diproses oleh *OpenCV* untuk mendapatkan panjang dan lebar dari getah karet. Hasil dari pemrosesan oleh *OpenCV* dapat dilihat pada gambar 5.10 :



Gambar 5.9 Hasil pemrosesan dari *OpenCV*

Lalu data tersebut dikirim ke *Arduino Nano* untuk dilakukan perhitungan dengan data tinggi dari *HC-SR04* untuk mendapatkan berat dari getah karet tersebut. Kemudian data berat dari pengolahan citra ditampilkan bersama dengan data berta dari *Loadcell* di *Lcd* untuk mengetahui berapa selisih beratnya. Hasil dari tampilan *Lcd* dapat dilihat pada gambar 5.11 :



Gambar 5.10 Hasil dari tampilan pada *Lcd*