



# MANUAL BOOK SIHACOV

<https://web-sihacov.vercel.app/>

---

PREPARED BY:

Mukhlis Maulana Al-Fakih

Firgi Aditya

Ari Fahrezi

A free & fully editable template by:

<https://usedtotech.com>



# CONTENTS

<b>Introduction .....</b>	<b>3</b>
<b>Section 1: Register .....</b>	<b>4</b>
<b>Register .....</b>	<b>4</b>
1. Tombol Pause/ Resume .....	4
2. Input Field Iteration.....	4
3. Tombol Good/ Not Good .....	4
4. Tombol Exit .....	5
<b>Panduan menggunakan Register .....</b>	<b>5</b>
1. Masukan Objek yang akan diregist.....	5
2. Masukan Nilai Iterasi.....	5
3. Pilih Kelas .....	5
<b>Section 2: Inspeksi.....</b>	<b>6</b>
<b>Inspeksi .....</b>	<b>6</b>
1. Label Kategori .....	6
2. Bounding Box.....	6
3. Label Jumlah Objek .....	6
<b>Section 3: Problem Solving.....</b>	<b>7</b>
<b>Problem Solve .....</b>	<b>7</b>

# SIHACOV

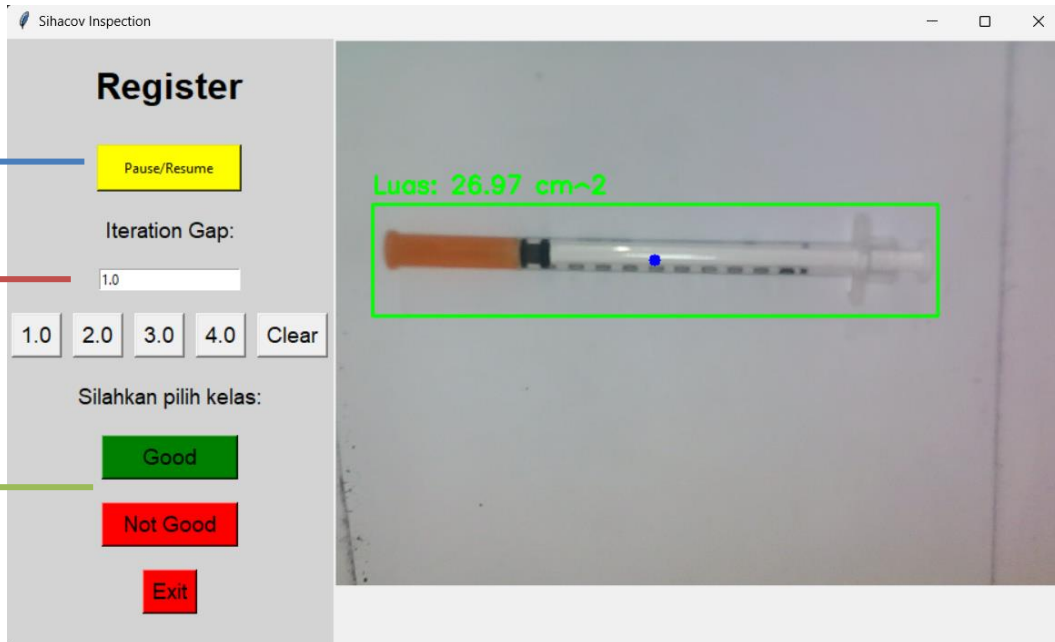
Sistem Inspeksi Alat Kesehatan menggunakan Computer Vision pada Proses  
Quality Control.

## INTRODUCTION

Sistem Inspeksi Alat Kesehatan menggunakan Computer Vision pada proses Quality Control (SIHACOV), merupakan alat inspeksi yang menggunakan beberapa teknologi didalamnya. Pengolahan Citra Digital (PCD) adalah salah satu dari teknologi yang digunakan pada alat ini. Dengan menggunakan PCD sistem ini mampu mendeteksi estimasi luas objek dengan satuan cm<sup>2</sup>. Setelah PCD mendeteksi, kemudian klasifikasi dilakukan dengan menggunakan algoritma *MLPClassifier* untuk dikategorikan sebagai “Good” atau “Not Good”. Algoritma klasifikasi ini berbasis *Artificial Neural Network* (ANN) yang dapat dilatih menggunakan dataset.

**Buku Panduan ini memiliki 3 Section**

Section 1	Register
Section 2	Inspeksi
Section 3	Problem Solving



## SECTION 1: REGISTER

### Register

Fitur Register ini merupakan fitur yang digunakan untuk membuat dataset pada alat ini. Dengan adanya Register *user* dapat mendaftarkan produk untuk diinspeksi, dengan syarat *user* harus memiliki 2 kondisi produk yaitu “Good” dan “Not Good”.

#### 1. Tombol Pause/ Resume

Berfungsi untuk menjeda frame agar *user* dapat mengetahui berapa nilai luas objek agar tidak terganggu deteksi PCD yang terus-menerus berubah.

#### 2. Input Field Iteration

*Input field iteration* ini berfungsi untuk memasukkan nilai gap produk yang akan diinspeksi. Di bawah *input field* terdapat tombol untuk memasukkan nilai secara langsung. Contoh : Objek dapat diklasifikasikan sebagai kelas “Good” sampai rentan 1.0 cm<sup>2</sup> ke atas.

#### 3. Tombol Good/ Not Good

Tombol ini berfungsi untuk memilih kelas manakan objek tersebut akan dikategorikan antara “Good” atau “Not Good”. (pastikan telah mengisi nilai iterasi)



#### 4. Tombol Exit

Berfungsi untuk keluar dari sistem register.

## Panduan menggunakan Register

### 1. Masukan Objek yang akan diregist

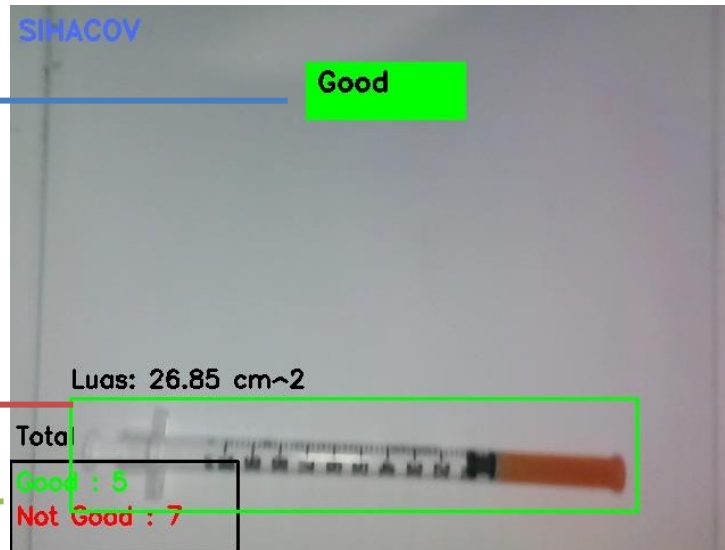
Masukan objek yang akan diregist ke conveyor. Pastikan objek tersebut berada di tengah *frame* kamera.

### 2. Masukan Nilai Iterasi

Masukan nilai gap yang digunakan untuk melakukan iterasi nilai luas yang terdeteksi hingga nilai yang dimasukan ke dalam *input field*. Pilih sesuai dengan tombol yang tersedia.

### 3. Pilih Kelas

Setelah memasukan nilai iterasi, selanjutnya *user* memilih kategori manakan atau kelas manakan objek tersebut akan dikategorikan (“*Good*” atau “*Not Good*”).



## SECTION 2: INSPEKSI

### Inspeksi

Gambar di atas merupakan tampilan proses inspeksi berlangsung. Objek yang terdeteksi akan dibuat bounding box dan diukur luasnya. Alat ini dilengkapi dengan sensor Proximity Photoelectric untuk mendeteksi objek. Ketika sensor tersebut mendeteksi maka sistem akan melakukan klasifikasi dan menghitung berapa jumlah objek yang diinspeksi berdasarkan kategorinya.

#### 1. Label Kategori

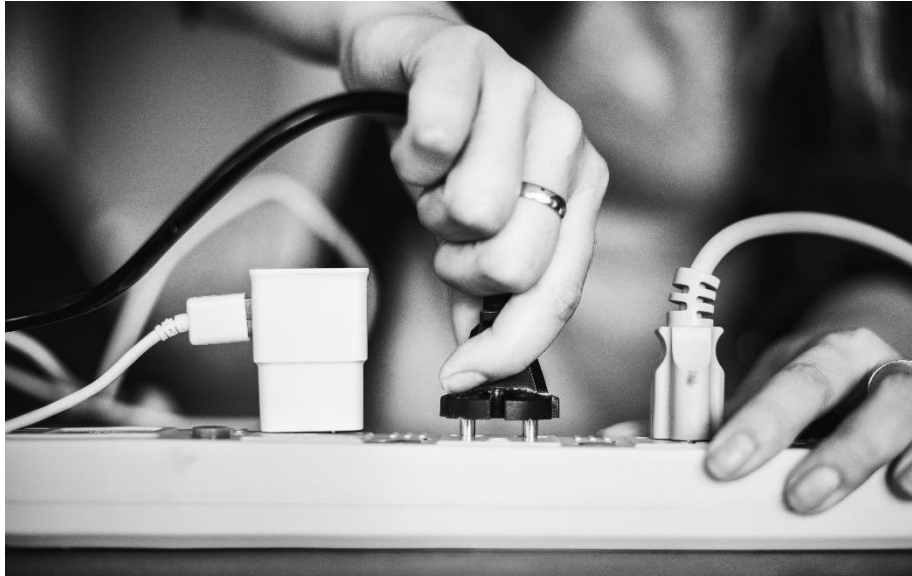
Terdapat label kategori untuk memberikan keterangan objek yang terdeteksi diklasifikasikan sebagai "Good" atau "Not Good".

#### 2. Bounding Box

Objek yang terdeteksi akan dibuat bounding box untuk mengukur berapa luas objek tersebut sebelum nantinya akan diklasifikasikan.

#### 3. Label Jumlah Objek

Hasil jumlah objek yang terdeteksi akan diakumulasi dan ditampilkan pada *frame* kamera.



## SECTION 3: PROBLEM SOLVING

### Problem Solve

Error	Solve
Tidak dapat membaca kamera	Pastikan Kabel kamera terpasang dengan baik
Warning Voltage Raspberry Pi	Gunakan power adapter yang sesuai yaitu 5V 3A
LED tidak menyala	Pastikan kabel Arduino nano terpasang dengan baik
Salah satu komponen tidak merespon	