

## **BAB III**

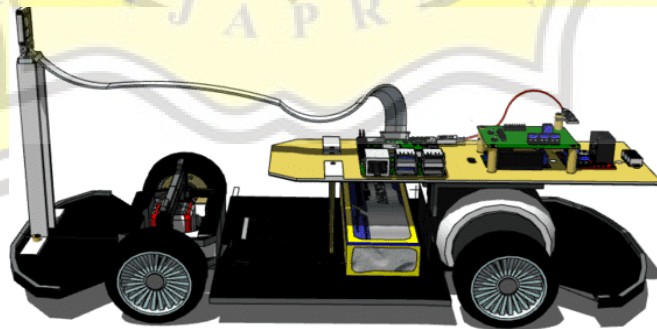
### **PERANCANGAN ALAT**

#### **3.1. Pendahuluan**

Pada Bab ini akan dijelaskan mengenai perancangan AGV. Perancangan alat mulai dari desain 3D AGV, wiring diagram, blok diagram kinerja AGV, proses kinerja AGV yang dijelaskan dengan *flowchart*, kemudian pola lintasan yang dipakai AGV, pendeteksian teks, dan yang terakhir membuat metodologi penelitian.

#### **3.2. Desain 3D**

Desain 3D AGV diperlukan untuk memberikan gambaran rancangan robot yang akan dibuat. Selain itu, proses perakitan akan dimudahkan dengan adanya desain 3D. Untuk membuat desain menggunakan aplikasi SketchUp Pro. Desain 3D ini merupakan desain yang diusulkan sesuai dengan keinginan perancang sehingga memudahkan dalam proses perakitan maupun peletakan komponen. Untuk itu, ukuran komponen harus diperhatikan agar dapat terpasang dengan baik. Gambar 3.1 menunjukkan desain 3D AGV tampak samping.

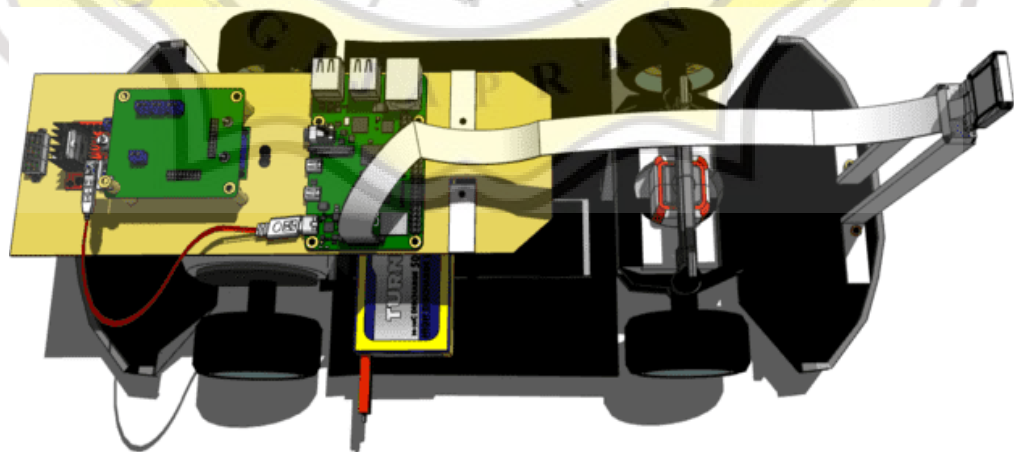


**Gambar 3.1 Desain 3D AGV Tampak Samping**

Proses desain diperlukan untuk memberikan gambaran terkait perancangan alat dalam hal ini perakitan AGV. Untuk Desain 3D AGV tampak depan dapat diamati Pada Gambar 3.2. Sedangkan untuk Desain 3D AGV tampak atas dapat dilihat pada Gambar 3.3.



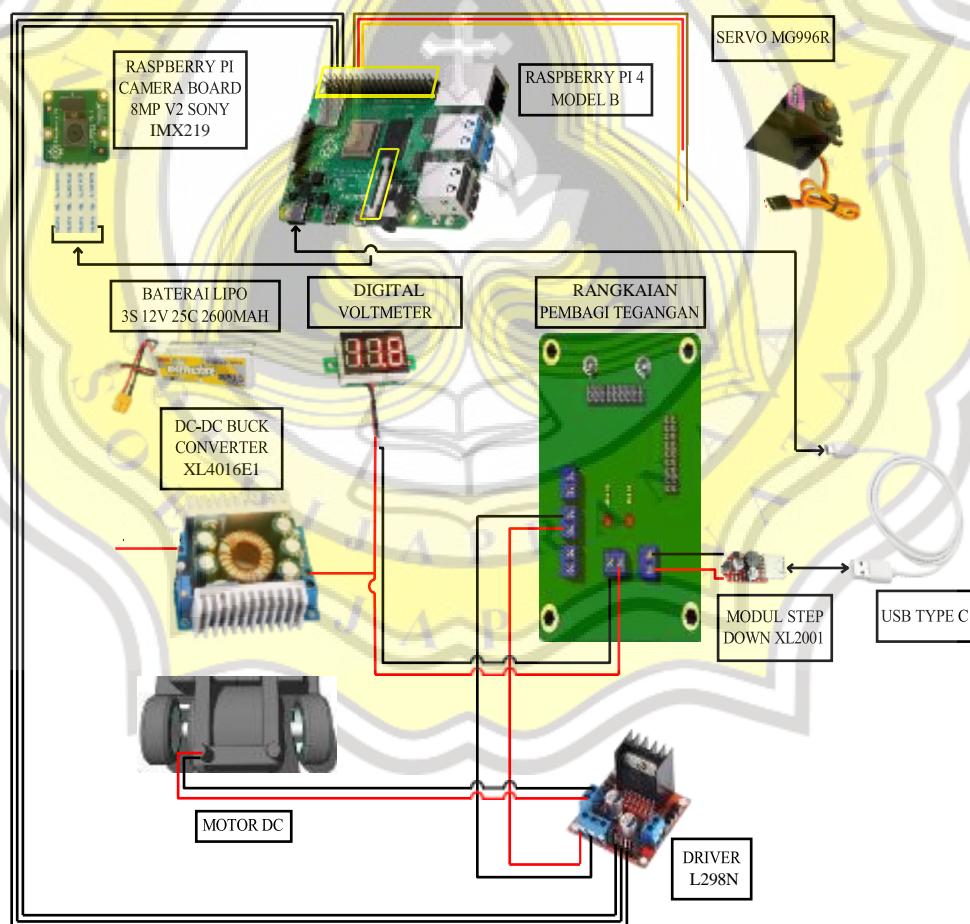
**Gambar 3.2 Desain 3D AGV Tampak Depan**



**Gambar 3.3 Desain 3D AGV Tampak Atas**

### 3.3. Wiring Diagram

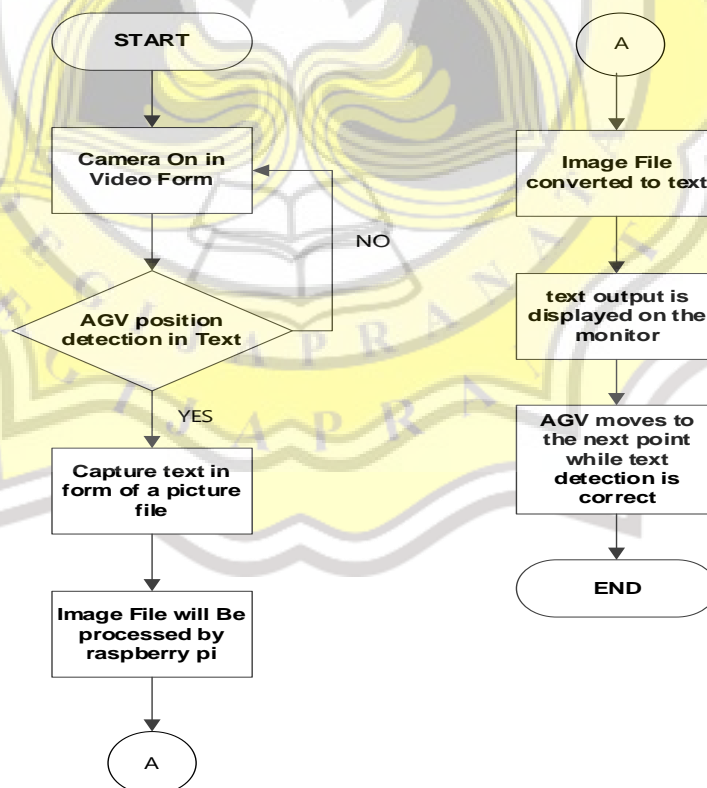
Pada proses *wiring* yang dilakukan seperti pada Gambar 3.4 yang perlu difokuskan yaitu komponen yang digunakan pada saat perakitan robot AGV. Perancangan skematik ini meminimalisir apabila adanya terjadi kesalahan pemasangan komponen dan mempermudah ketika merakit komponen ini hingga menjadi AGV. AGV ini menggunakan suplai tegangan 12 Volt DC. Kemudian tegangan tersebut disalurkan melalui rangkaian pembagi tegangan/*voltage divider*. Pada robot AGV ini juga memakai modul *step down* untuk menurunkan tegangan dari 12 Volt menjadi 5 Volt untuk mensuplai Raspberry Pi yang terhubung melalui kabel *usb type-C*.



Gambar 3.4 Wiring Diagram

### 3.4. Blok Diagram Kinerja Pembacaan Teks Pada AGV

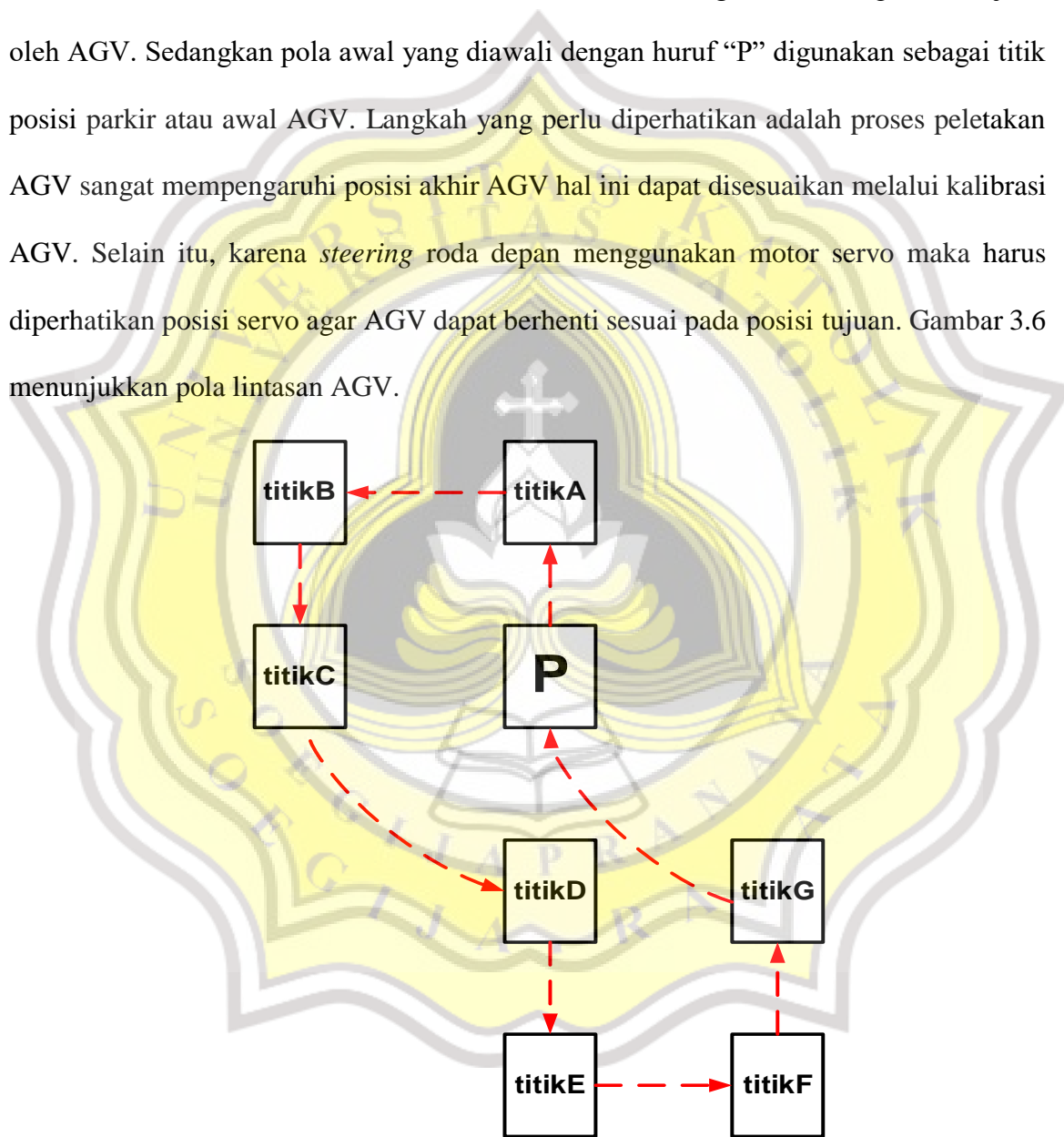
Blok diagram berikut ini menunjukkan sistem pembacaan teks pada AGV. Input yang dipakai berupa kamera untuk mendeteksi titik awal yang dioperasikan menggunakan Raspberry Pi. Pada penelitian ini berfokus pada sistem pembacaan AGV yang digunakan untuk mendeteksi keberadaan AGV ini. Pembacaan AGV ini menggunakan OCR (*Optical Character Recognition*) yang sudah dimodifikasi sehingga dapat mendeteksi teks sebagai detector robot AGV terhadap pola lintasan sehingga dapat dicapai akurasi AGV dengan *error* yang minim. Untuk kedua roda depan menggunakan motor servo untuk proses *steering* berbelok kanan maupun kiri. Sedangkan untuk penggerak roda belakang memakai motor DC yang mudah diatur kecepatan dan arah putarannya. AGV dapat berpindah posisi sesuai dengan perintah yang diproses melalui program.



Gambar 3.5 Flowchart Proses Kinerja AGV

### 3.5. Pola Lintasan

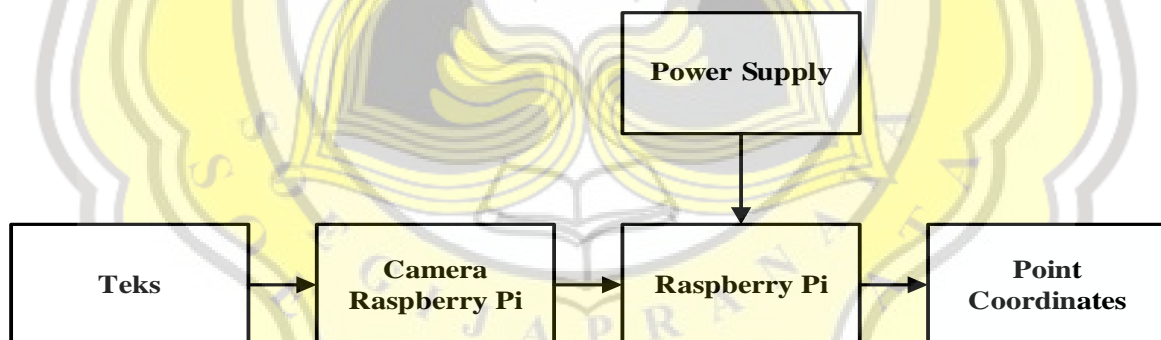
Trek yang digunakan dalam AGV ini akan ditunjukkan pada Gambar 3.7 . Robot AGV ini diuji di dalam ruangan dan pencahayaannya cukup terang. Saat mengambil gambar dalam bentuk kertas HVS dengan menggunakan kata-kata “titikA”, “titikB”, “titikC”, “titikD”, “titikE”, “titikF”, “titikG”, dan “titikP” digunakan sebagai titik tujuan oleh AGV. Sedangkan pola awal yang diawali dengan huruf “P” digunakan sebagai titik posisi parkir atau awal AGV. Langkah yang perlu diperhatikan adalah proses peletakan AGV sangat mempengaruhi posisi akhir AGV hal ini dapat disesuaikan melalui kalibrasi AGV. Selain itu, karena *steering* roda depan menggunakan motor servo maka harus diperhatikan posisi servo agar AGV dapat berhenti sesuai pada posisi tujuan. Gambar 3.6 menunjukkan pola lintasan AGV.



Gambar 3.6 Pola Lintasan

### 3.6. Block Diagram of Pattern Recognition

Gambar 3 menunjukkan diagram blok dari sistem pengenalan teks. Diagram blok ini merupakan urutan kerja teks bacaan yang akan digunakan untuk mendeteksi lokasi yang akan diterapkan pada AGV. Alat ini dilengkapi dengan sensor kamera *built-in* dengan raspberry pi yang akan berfungsi untuk *streaming* video yang kemudian akan membaca teks yang telah ditentukan. Catu Daya diperoleh dari baterai yang dikonversi menggunakan chopper step-down untuk mengontrol komputer mini atau raspberry ini. Teks terbaca oleh kamera saat video stream akan diproses oleh raspberry yang telah diprogram sebelumnya. Setelah membaca teks ini tepat di lokasi yang kita inginkan, AGV ini akan menunjukkan dimana letak AGV tersebut, jadi raspberry ini akan menampilkannya dalam bentuk koordinat (x,y) yang kemudian akan menginstruksikan kita untuk menggerakkan motor yang tadi diprogram sebelumnya untuk sampai ke titik itu.



Gambar 3.7 Model kinematik robot AGV

### 3.7. Invariant Moment and Zernike Moment

Dalam pengenalan pola atau objek, invarian momen dapat digunakan sebagai metode yang mengekstraksi suatu pola atau objek. Metode ini menggunakan fungsi dua dimensi diskrit, yaitu  $f(x,y)$  dalam bentuk digital momen biasa  $(p + q)$ . Makalah ini mengadopsi momen invarian seperti pada (1).

$$m_{pq} = \sum x (x - \bar{x})^p (y - \bar{y})^q f(x, y) \quad (1)$$

Dari (1),  $\bar{x} = \sum_x x, \bar{y} = \sum_y y$  mewakili massa pusat benda. Asal gambar digital pertama  $f(x,y)$  diasumsikan dalam menerjemahkan ke pusat massa suatu objek untuk mencapai terjemahan invarian. Mirip dengan momen biasa, momen Zernike (m,n) digunakan untuk gambar digital yang ditentukan seperti pada (2) berikut.

$$A_{nm} = \frac{n+1}{\pi} \sum_x \sum_y f(x, y) V_{nm}(x, y), x^2 + y^2 \leq 1 \quad (2)$$

Dari (2),  $V_{nm}(x, y) = V_{nm}(\rho, \theta) = R_{nm}(\rho) e^{jm}$ ,  $|m| \leq n$  dan  $n - |m|$  adalah bilangan genap. Kemudian  $R_{nm}(\rho)$  dapat didefinisikan seperti pada (3).

$$R_{nm}(\rho) = \sum_{s=0}^{\frac{n-|m|}{2}} (-1)^s \frac{(n-s)!}{s! \left(\frac{n+m-s}{2}\right)! \left(\frac{n-m-s}{2}\right)!} \rho^{n-2s} \quad (3)$$

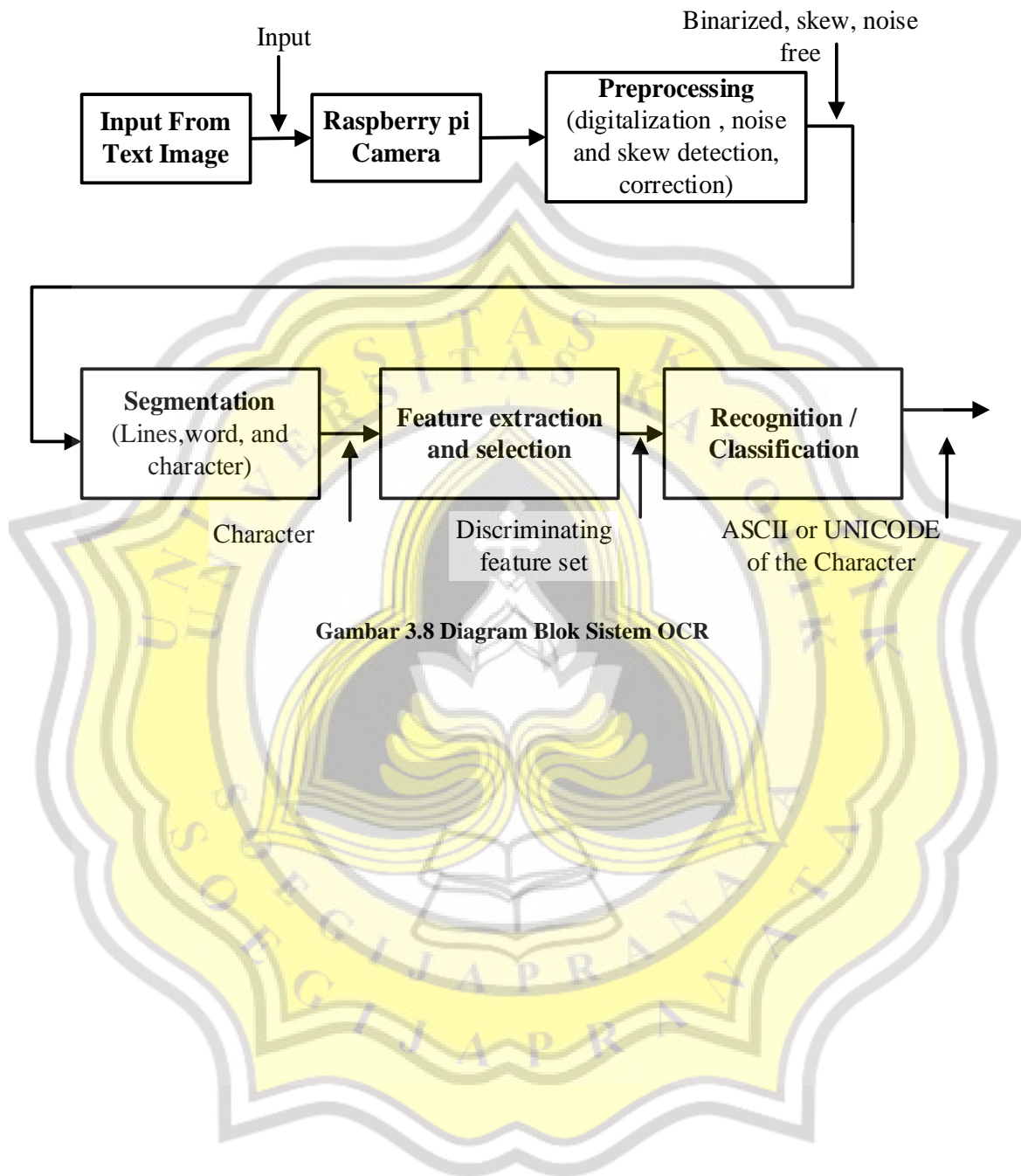
Untuk fungsi gambar nyata  $f(x, y)$ , hanya suku  $V_{nm}(\rho, \theta)$  yang merupakan bilangan kompleks seperti pada (4)

$$A_{nm} = \frac{n+1}{\pi} \sum_{xi} \sum_{yj} f(x_i, y_j) \quad (4)$$

$$[VR_{nm}(x, y) + jVI_{nm}(x, y)] x^2 + y^2 \leq 1$$

Dengan VR dan VI adalah fungsi real yang mewakili komponen real dan imajiner dari basis fungsi  $V_{nm}(\rho, \theta)$ , yang diubah menjadi bidang (x,y). Sistem yang digunakan oleh OCR terdiri dari akuisisi citra, preprocessing, segmentasi, ekstraksi fitur & seleksi, dan komponen pengenalan seperti yang ditunjukkan pada Gambar 3.8. Teks karakter atau gambar diperoleh dengan menggunakan kamera yang terhubung ke komputer mini. Gambar yang ditangkap oleh kamera disimpan dalam format skala abu-abu, tetapi apa yang dilihat diproses menjadi gambar biner. Modul pra-pemrosesan mengubah warna/skala abu-abu menjadi gambar biner, koreksi deteksi kemiringan, dan digitalisasi. Proses ini juga mendeteksi adanya noise dan mengoreksi noise pada citra. Noise ini bisa disebabkan karena kualitas kamera yang

merekam tidak memiliki kualitas yang baik. Berikut adalah gambar blok diagram untuk sistem pembacaan OCR



Gambar 3.8 Diagram Blok Sistem OCR