

BAB II

KAJIAN PUSTAKA

2.1 Tinjauan Pustaka

Sistem *Quality Control bag semen* yang ada pada lapangan saat ini menggunakan SDM (pekerja pabrik) sebagai pengontrol utama. Pada penelitian ini penulis berusaha memberikan inovasi menggunakan implementasi Teknologi Visual Process dan kecerdasan buatan. Teknologi Visual Process dengan Image Processing dengan metode NN.

Data sistem monitoring yang dibuat pada penelitian ini yaitu proses konversi dari RGB ke grayscale kemudian diolah menggunakan metode NN menggunakan Matlab. Ekstraksi fitur menghasilkan data berupa kontras, korelasi, energy dan homogenitas sehingga di dapatkan sebuah nilai yang akan dilakukan pengklasifikasian menggunakan NN.

Untuk itu agar terwujudnya visi PT. Cemindo Gemilang maka penulis melakukan penelitian ini sehingga dapat membantu meningkatkan kualitas proses packaging bag semen yang tergolong sulit dikarenakan faktor panas, berdebu dan berat. Dengan menggunakan inovasi visual sebagai alat sortir secara real time yang nantinya siap dikembangkan menuju high quality packaging visual automatic control. Selain itu untuk mendukung keberhasilan penelitian ini, digunakan beberapa komponen pendukung. Dasar – dasar dari beberapa komponen yang digunakan pada tugas akhir ini akan dibahas pada bagian ini.^[3]

2.2 Dasar Teori

2.2.1 Kamera Logitech C270 HD

Pada penelitian ini kamera digunakan untuk mengambil citra objek kemudian akan diproses ke dalam grayscale. Ketika dalam suatu produk packaging terdapat suatu benturan pada saat proses produksi bag semen sehingga dapat menyebabkan perubahan bentuk fisik pada produk maka bisa disebut sebagai cacat fisik atau sobek. Kamera diletakkan pada atas konveyor karena 1 unit kamera dapat menjangkau 180⁰. Untuk spesifikasi kamera Logitech C270 yaitu memiliki fitur

autofocus dengan 3 megapixel dengan ditingkatkan menggunakan perangkat lunak dan dapat mempertahankan ketajaman hingga sedekat 10 cm.^[4] Pada tugas akhir ini digunakan kamera Logitech seperti pada Gambar 2.1.



Gambar 2.1 Logitech C270 HD Camera

2.2.2 Ekstraksi Fitur

Ekstraksi fitur dilakukan untuk mengambil informasi pokok dari suatu data tertentu sebelum digunakan dalam proses tertentu. Analisis tekstur lazim dimanfaatkan sebagai proses antara untuk melakukan klasifikasi dan interpretasi citra. Suatu proses klasifikasi citra berbasis analisis tekstur pada umumnya membutuhkan tahapan ekstraksi fitur, yang dapat terbagi dalam tiga macam metode berikut :

1. Metode Statistik

Metode statistik menggunakan perhitungan statistik distribusi derajat keabuan (histogram) dengan mengukur tingkat kekontrasan, granularitas, dan kekasaran suatu daerah dari hubungan ketetanggaan antar piksel di dalam citra. Paradigma statistik ini penggunaannya tidak terbatas, sehingga sesuai untuk tekstur-tekstur alami yang tidak terstruktur dari sub pola dan himpunan aturan (mikrostruktur).

2. Metode Spektral

Metode spektral berdasarkan pada fungsi autokorelasi suatu daerah atau power distribution pada domain transformasi Fourier dalam mendeteksi periodisitas tekstur.

3. Metode Struktural

Analisis dengan metode ini menggunakan deskripsi primitif tekstur dan aturan sintaktik. Metode struktural banyak.

Pada tugas akhir ini setelah memperoleh matriks kookurensi tersebut, kita dapat menghitung ciri statistik orde dua dengan cara merepresentasikan citra yang diamati dengan formula seperti pada Tabel 2.1.

Tabel 2.1 Formula ekstraksi fitur

No.	Fitur	Formula
1.	<i>Contras</i>	$\sum_i^n \sum_j^m (i-j)^2 C(i,j)$ (2.1)
2.	<i>Energy</i>	$\sum_i^n \sum_j^m C^2(i,j)$ (2.2)
3.	<i>Entropy</i>	$\sum_i^n \sum_j^m C(i,j) \log C(i,j)$ (2.3)
4.	<i>Homogenitas</i>	$\sum_i^n \sum_j^m \frac{C(i,j)}{1+ i+j }$ (2.4)
5.	<i>Correlation</i>	$\frac{\sum_i^n \sum_j^m (i,j) \cdot C(i,j) - \mu_i \mu_j}{\sqrt{\sigma_i^2 \sigma_j^2}}$ (2.4.1)
		$\mu_i = \sum_i^n \sum_j^n (i) (C(i,j))$ (2.4.2)
		$\mu_j = \sum_i^n \sum_j^n (j) (C(i,j))$ (2.4.3)
		$\sigma_i^2 = \sum_i^n \sum_j^n ((C(i,j))(i - \mu_i)^2)$ (2.4.4)
		$\sigma_j = \frac{1}{2} \sqrt{\sum_i^n \sum_j^n ((C(i,j))(j - \mu_j)^2)}$ (2.4.5)

Kegunaan dari 6 fitur dari hasil ekstraksi fitur adalah sebagai berikut :

1. Entropy
Mengukur kompleksitas (keacakan) citra. Entropy akan bernilai tinggi ketika citra tidak seragam.
2. Dissimilarity

Mengukur ketidakmiripan suatu tekstur, yang akan bernilai besar bila acak dan sebaliknya akan bernilai kecil bila seragam.

3. Contrast / inertia

Mengukur frekuensi spasial dari citra dan perbedaan moment GLCM. Perbedaan yang dimaksudkan adalah perbedaan tinggi dan rendah nya pixel. Contrast akan bernilai 0 jika pixel ketetanggaan mempunyai nilai yang sama.

4. Correlation

Mengukur linearitas (the joint probability) dari sejumlah pasangan pixel (*pairs*).

5. Energy

Menunjukkan kehomogenan citra yang berderajat keabuan sejenis

6. Homogeneity / inversedifference moment

Mengukur homogenitas. Nilai ini sangat sensitif terhadap nilai disekitar diagonal utama. Bernilai tinggi ketika semua pixel mempunyai nilai yang sama/ seragam. Kebalikan dari contrast yaitu akan bernilai besar jika mempunyai nilai pixel yang sama pada saat energy bernilai tetap.^[2]

2.2.3 Neural Network (NN)

Neural network merupakan bagian dari sistem kecerdasan buatan yang digunakan untuk memproses informasi yang didesain dengan menirukan cara kerja otak manusia dalam menyelesaikan masalah dengan melakukan proses belajar melalui perubahan bobot sinapsisnya. Pada Gambar 2.4 menunjukkan salah satu contoh syaraf secara biologis dimana setiap sel syaraf (neuron) akan memiliki satu inti sel yang bertugas untuk melakukan pemrosesan informasi. Informasi yang datang akan diterima oleh dendrit. Sistem tersebut seperti ditunjukkan pada Gambar

2.4. Sistem tersebut hampir sama dengan sistem syaraf secara biologi.^[4]



Gambar 2.2 Syaraf secara biologi

Selain menerima informasi, *dendrit* juga menyertai *axon* sebagai keluaran dari suatu pemrosesan informasi. Informasi hasil olahan ini akan menjadi masukan bagi neuron lain dimana antar *dendrit* kedua sel tersebut dipertemukan dengan sinapsisnya. Informasi yang dikirimkan antar neuron ini berupa rangsangan yang dilewatkan melalui *dendrit*. Informasi yang datang dan diterima oleh *dendrit* akan dijumlahkan dan dikirim melalui *axon* lain. Informasi ini akan diterima oleh neuron lain jika memenuhi batasan tertentu dikenal dengan nilai ambang (*threshold*) yang dikatakan teraktivasi.

Pada umumnya jaringan syaraf tiruan memiliki tiga lapisan, yaitu *Input layer*, *hidden layer*, dan *Output layer*. Berikut penjelasan mengenai layer pada NN.

1. *Input layer*

Input layer berisi *neuron-neuron* yang masing-masing menyimpan sebuah nilai masukan yang tidak berubah pada fase latih dan hanya dapat berubah jika diberikan nilai masukan baru. *Neuron* pada lapisan ini tergantung pada banyaknya *Input* dari suatu pola.

2. *Hidden layer*

Lapisan ini tidak pernah muncul sehingga dinamakan *hidden layer*. Akan tetapi semua proses pada fase pelatihan dan fase pengenalan dijalankan di lapisan ini. Jumlah lapisan ini tergantung dari arsitektur yang akan dirancang, tetapi pada umumnya terdiri dari satu lapisan *hidden layer*.

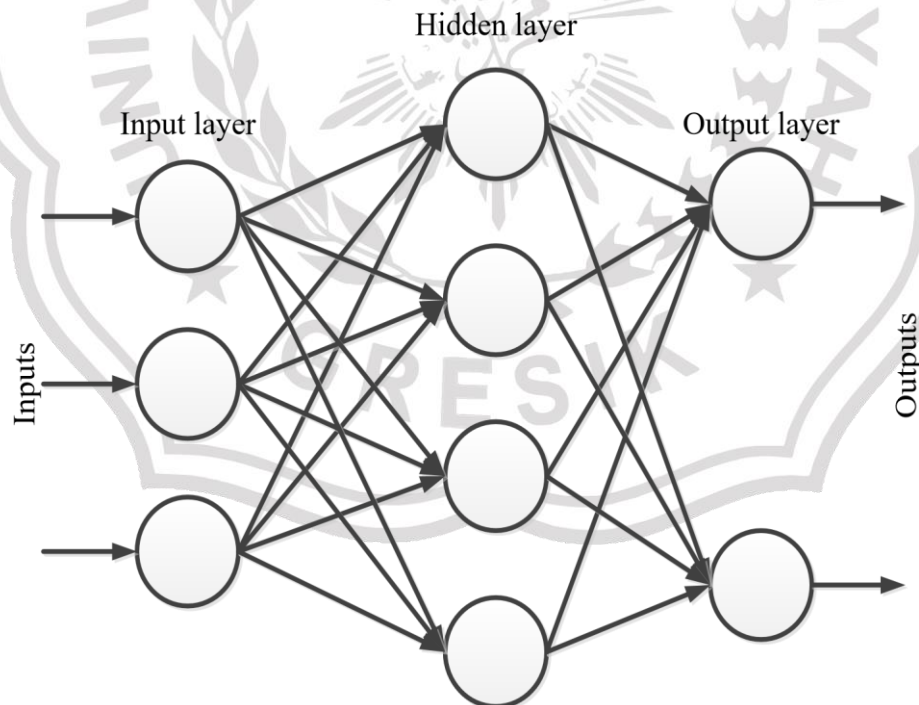
3. *Output layer*

Output layer berfungsi untuk menampilkan hasil perhitungan sistem oleh fungsi aktivasi pada lapisan *hidden layer* berdasarkan *Input* yang diterima. *Neural network* ditentukan oleh tiga hal, yaitu pola hubungan antar neuron yang disebut jaringan. Metode untuk menentukan bobot penghubung yang disebut metode *Training / learning / algoritma*, dan fungsi aktivasi atau fungsi transfer. Salah satu arsitektur *NN* yang sangat populer adalah multi layer *feedforward networks*. Secara umum, jaringan seperti ini terdiri dari sejumlah unit neuron sebagai lapisan *Input*, satu atau lebih lapisan simpul-simpul neuron komputasi *hidden* (lapisan tersembunyi), dan sebuah lapisan simpul-simpul neuron komputasi *Output*. Sinyal *Input* dipropagasikan kearah depan (arah lapisan *Output*), lapisan demi lapisan.

Jenis jaringan ini adalah hasil generalisasi dari arsitektur perceptron satu lapisan, jadi biasa disebut sebagai *multi layer perceptron* (MLP).

Propagasi balik (ke arah lapisan *Input*) terjadi setelah jaringan menghasilkan *Output* yang mengandung *Error*. Pada fase ini seluruh bobot *synaptic* (yang tidak memiliki aktivasi nol) dalam jaringan akan disesuaikan untuk mengoreksi/memperkecil *Error* yang terjadi (*Error correction rule*). Untuk pelatihan jaringan, pasangan fase propagasi ke depan dan balik dilakukan secara berulang untuk satu set data latihan, kemudian diulangi untuk sejumlah *epoch* (satu sesi lewatan untuk seluruh data latihan dalam sebuah proses pelatihan jaringan) sampai *Error* yang terjadi mencapai batas kecil toleransi tertentu atau nol.

MLP terdiri dari beberapa unit pemroses (neuron) seperti pada Gambar 2.5, yang terhubung dan mempunyai beberapa masukan serta memiliki satu atau beberapa *Output*. *Perceptron* digunakan untuk menghitung jumlah nilai perkalian penimbang dan masukan dari parameter yang kemudian dibandingkan dengan nilai *threshold*, jika keluaran lebih besar dari *threshold* maka keluaran adalah satu, sebaliknya adalah nol. [5]



Gambar 2.3 Struktur *Multi Layer Perception*

Pernyataan ini merupakan hasil proses pelatihan yang dalam bentuk bahasanya adalah pernyataan iya atau tidak. Secara matematis dapat ditulis dengan persamaan seperti pada Persamaan 2.1:

Jumlah perkalian penimbang dan parameter masukan adalah :

$$I = w_{ji} x_i \quad (2.1)$$

Dengan :

I = Input

x_i = sinyal masukan

w_i = penimbang

Dari Persamaan 2.1 Bila $I > T$ maka keluaran $O = 1$, dengan T adalah *threshold*. Pelatihan pada *perceptron* dilakukan dengan merubah nilai penimbangannya sehingga sesuai dengan kebutuhan, dilakukan dengan membandingkan keluaran dari *JST* dengan targetnya, proses tersebut dapat dituliskan pada Persamaan 2.2

$$w_{baru_{ji}} = w_{lama_{ji}} + \alpha(t_j - O_j)x_i \quad (2.2)$$

Dengan :

t_j = target

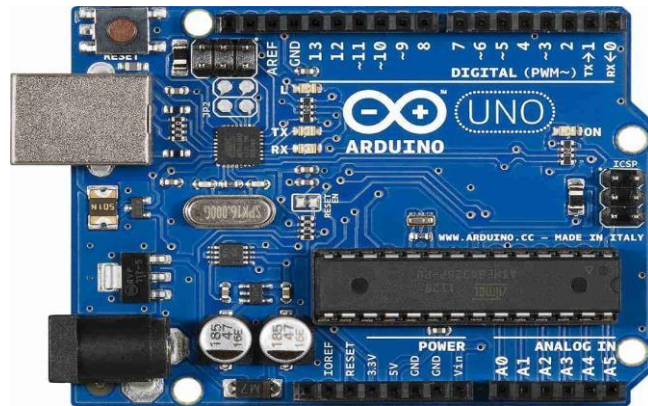
α = bilangan konstan

O_j = Output

Proses diatas dijalankan pada setiap *neuron* yang ada pada setiap layer sampai penimbang tersebut sesuai dengan yang diinginkan. Nilai awal penimbang adalah bilangan kecil yang diambil secara acak.

Pada tugas akhir ini *NN* digunakan pelatihan dan *Testing* untuk klasifikasinya. Yang dimana setelah proses konvolusi dan subsampling selesai, proses akhir akan dikombinasikan dengan *NN* untuk mendapatkan nilai *hidden layer*.

2.2.4 Arduino Uno



Gambar 2II.4 Arduino UNO

Arduino adalah sebuah mikrokontroler dengan ATmega 328 yang memiliki tegangan operasi 5V yang digunakan sebagai pengendali proses dari sensor dan aktuator yang kemudian akan memberikan sebuah data pada pengendalian proses suatu system. Mikrokontroler ini memiliki beberapa fitur yaitu 14 digital input / output pin (6 pin digunakan sebagai output PWM), 6 input analog, sebuah osilator 16 MHz. ^[5]

2.2.5 Motor DC

Motor DC memerlukan suplai tegangan yang searah pada kumparan medan untuk diubah menjadi energi mekanik. Dalam motor DC terdapat dua kumparan yaitu kumparan medan yang berfungsi untuk menghasilkan medan magnet dan kumparan jangkar yang berfungsi sebagai tempat terbentuknya gaya gerak listrik (GGL).



Gambar 2.5 Motor DC

Jika arus dalam kumparan jangkar berinteraksi dengan medan magnet, akan timbul torsi (T) yang akan memutar motor. Gambar 2.9 merupakan gambar motor DC. Akan motor DC ini digunakan untuk menggerakkan konveyor yang dimana akan memindahkan *Packing* rokok ketempat pengecekan (*Sorting*).^[6]

2.2.6 Motor Servo



Gambar 2.6 Motor Servo

Motor servo adalah sebuah motor DC yang dilengkapi rangkaian kendali dengan sistem closed feedback yang terintegrasi dalam motor tersebut. Pada motor servo posisi putaran sumbu (axis) dari motor akan diinformasikan kembali ke rangkaian kontrol yang ada di dalam motor servo selain itu juga dapat bergerak bekerja dua arah (CW dan CCW) dimana arah dan sudut pergerakan rotornya dapat dikendalikan dengan memberikan variasi lebar pulsa (duty cycle) sinyal PWM pada bagian pin kontrolnya.^[7]

2.2.7 Sensor Proximity

Proximity capacitive mempunyai prinsip kerja memantulkan infrared dan diterima receiver yang ada pada ujung proximity. Proximity digunakan untuk mendeteksi drum yang lewat di depan kamera ataupun ejector dengan memanfaatkan kondisi Low (+0V) saat mendeteksi drum yang melewati batas pantulan proximity, saat keadaan stand by proximity mengirim sinyal High (+5V).^[9]



Gambar 2.7 Sensor Proximity

2.2.8 Driver L298N



Gambar 2.8 Driver L298N

L298 adalah driver motor berbasis H-Bridge yang mampu menangani beban hingga 4A pada tegangan 6V – 46V, driver ini memiliki kemampuan untuk mengendalikan 2 untuk motor dc namun hanya dapat mengendalikan 1 motor stepper. L928 akan digunakan untuk mengendalikan motor yang ada di konveyor pada prototype sortir asphalt drum (kedairobot.com, 2015). Spesifikasi dari modul driver motor L298 antara lain : 1. Menggunakan IC L298N (Double H bridge Drive Chip) 2. Tegangan minimal untuk masukan power antara 5V-35V 3. Tegangan operasional 5V 4. Arus untuk masukan antara 0-36 mA 5. Arus maksimal untuk keluaran per output yaitu 2A 6. Daya maksimal yaitu 25W.^[10]