

Penerapan Sistem Pakar Berbasis Aturan dan Metode *Freeman Chain Code* pada Aplikasi Pengukuran Isi Botol

Ken Ratri Retno Wardani^{#1}, Ari Setiawan^{#2}, Yohanes Setiawan^{#3}

^{1,3}*Departemen Informatika, Institut Teknologi Harapan Bangsa
Jl. Dipatiukur No. 80-84, 40132 Indonesia*

¹ken_ratri@ithb.ac.id

³yohanes.s447@gmail.com

²*Departemen Teknik Industri, Institut Teknologi Harapan Bangsa
Jl. Dipatiukur No. 80-84, 40132 Indonesia*

²ari_setiawan@ithb.ac.id

Abstract— *In the field of the food or beverage industry, particularly in the beverage industry, a tool is needed to measure the contents of the bottle volume which is useful to supervise the process of bottle filling so that the contents of the bottle is not less or no more than the company limit. Rule-based expert systems can be used to calculate the estimated volume contents of a bottle by giving facts to the bottle and the fact of the bottle tolerance. From these facts will be calculated the estimated volume of the image to be tested and determine whether the image is included in the company standard or not. The chain code method can also be used in a wide screening process that serves to filter unwanted pixels for the purpose of searching the region over the bottle and surface area of the bottle contents. Testing the threshold value of the chain code is performed to find the most optimal threshold value for widespread filtering. Based on the results of testing the most optimal threshold value is 200 and 300 with 100% accuracy and 0.89 milliliter accuracy.*

Keywords— *bottle, volume, rule based, chain code.*

Abstrak— Pada bidang industri makanan atau minuman, khususnya pada industri minuman dibutuhkan alat untuk mengukur isi volume botol yang berguna untuk mengatasi masalah dalam pengisian botol sehingga isi botol tidak kurang atau tidak lebih dari batas perusahaan. Sistem pakar berbasis aturan dapat digunakan dalam menghitung perkiraan isi volume suatu botol dengan memberi fakta berupa citra botol berisi volume neto dan fakta berupa citra botol berisi volume toleransi. Dari fakta-fakta tersebut akan dihitung perkiraan volume dari citra yang akan diuji dan ditentukan apakah citra tersebut masuk dalam standar perusahaan atau tidak. Metode kode rantai juga dapat digunakan dalam proses penyaringan luas yang berfungsi menyaring piksel-piksel yang tidak diinginkan untuk keperluan proses pencarian wilayah atas botol dan wilayah permukaan isi botol. Pengujian nilai *threshold* luas area kode rantai dilakukan untuk mencari nilai *threshold* paling optimal untuk penyaringan luas. Berdasarkan hasil pengujian nilai *threshold* paling optimal adalah 200 dan 300 dengan akurasi 100% dan tingkat ketelitian 0,89 mililiter.

Kata Kunci— *bottle, volume, rule based, chain code.*

I. PENDAHULUAN

Teknologi informasi berkembang sangat pesat saat ini. Misalnya dalam bidang industri makanan dan minuman untuk tujuan pengawasan. Salah satu industri yang paling sering ditemui adalah industri minuman. Pemrosesan dalam industri minuman bermacam-macam dimulai dari pembuatan botol kemasan, pengisian botol, dan pemasangan label kemasan.

Kepercayaan konsumen (*trust*) merupakan hal yang sangat penting yang ditentukan oleh kualitas kemasan, rasa (kualitas isi), komposisi minuman, dan ketelitian berat atau volume minuman. Konsumen akan kecewa jika volume minuman yang dibeli kurang dari standar yang dicantumkan pada kemasan. Sebaliknya, produsen akan rugi jika volume minuman yang diisi terlalu banyak, melebihi volume yang dicantumkan pada kemasan [1].

Penelitian yang sudah untuk mengatasi masalah pengisian botol adalah untuk menganalisis apakah sebuah botol memiliki isi atau kosong. Penelitian tersebut menggunakan metode *canny edge detection* untuk proses deteksi tepi dan metode *morphological opening* untuk perbaikan kualitas citra. Hasil pengujian menunjukkan 88% dari total data sesuai dengan target yang diinginkan. Aplikasi mengeluarkan hasil kosong untuk botol yang kosong dan mengeluarkan hasil isi untuk botol yang berisi.

Penelitian terkait selanjutnya bertujuan untuk mengawasi proses pengisian botol dengan mencari permukaan isi botol dan menentukan apakah tinggi isi botol sesuai standar atau tidak. Metode yang dipakai pada penelitian tersebut adalah *ISEF edge detection* untuk mendeteksi tepi dan *Region of Interest (ROI)* untuk menentukan wilayah penting untuk diproses. Proses penentuan didapat dengan mengukur jarak antara atas botol dengan permukaan isi botol. Jika jarak sesuai dengan yang diharapkan, maka isi botol memenuhi standar yang diberikan. Jika jarak terlalu jauh, maka isi botol kurang dari standar yang diberikan.

Pada penelitian ini akan digunakan sistem pakar berbasis aturan sehingga dapat mengukur volume isi botol dan menentukan isi botol tersebut masuk dalam standar atau tidak. Sistem pakar berbasis aturan berfungsi untuk menjalankan proses-proses yang ada mulai dari *pre-processing* sampai

proses perhitungan akhir. Proses paling awal merupakan pengambilan citra batas atas dan batas bawah untuk keperluan perhitungan di proses akhir.

Penggunaan metode kode rantai digunakan untuk proses pencarian wilayah botol dan wilayah isi botol. Tujuan dari kode rantai adalah untuk menentukan/mencari representasi batasan dari suatu objek. Kode rantai merupakan algoritme sederhana tetapi memiliki kinerja yang tinggi. Kode rantai didasarkan pada kenyataan bahwa titik berurutan pada kurva kontinyu yang saling berdekatan satu sama lain [2].

II. DASAR TEORI

A. Citra Digital

Definisi citra adalah suatu bentuk optikal objek yang dihasilkan oleh intensitas cahaya yang dipantulkan dari objek tersebut kemudian ditampilkan oleh lensa kamera. Definisi citra digital adalah representasi numerik berupa fungsi dua dimensi. x dan y adalah koordinat ruang yang terbatas dan diskrit, dan f adalah derajat keabuan pada titik tertentu yang terbatas dan diskrit [3].

B. Citra Grayscale (skala keabuan)

Citra *grayscale* mempunyai *range* diantara warna hitam dan warna putih. Banyaknya warna tergantung pada jumlah bit yang disediakan di memori untuk menampung kebutuhan warna tersebut. Semakin besar jumlah bit warna yang disediakan di memori, maka semakin halus gradasi warna yang terbentuk [3].

C. Kode Rantai (Freeman Chain Code)

Kode rantai digunakan untuk memberitahukan representasi batasan dari suatu objek. Kode rantai merupakan algoritme sederhana tetapi memiliki kinerja yang tinggi, didasarkan pada kenyataan bahwa titik berurutan pada kurva kontinyu yang saling berdekatan satu sama lain. Masing-masing titik data secara berurutan berbatasan dengan salah satu dari delapan kode rantai yang mengelilingi titik data tersebut. Kode rantai direpresentasikan dengan 4 arah atau 8 arah mata angin. Arah dari suatu mata angin dikodekan dengan menggunakan skema penomoran [4].

Kode rantai dimulai dengan mengubah citra RGB menjadi citra biner. Setelah itu, ditentukan titik awal sebagai titik awal penelusuran dengan arah mata angin. Setelah ditelusur, hasilnya akan berupa kode-kode. Perimeter menyatakan ukuran panjang dari keliling atau luas objek yang ditelusuri.

Metode kode rantai memiliki 2 jenis pendekatan, yaitu [5]:

1. Kode rantai pada umumnya dengan penelusuran tepian objek dimana perimeter panjang akan menunjukkan panjang keliling objek tersebut.
2. Kode rantai dengan penelusuran berdasarkan arah jarum jam dan dimulai dari arah kanan dahulu. Perimeter panjang yang dihasilkan pada teknik ini menunjukkan luas dari objek yang ditelusuri.

$$P = \text{jumlah kode genap} + \sqrt{2} \times \text{jumlah kode ganjil} \quad (1)$$

Pada persamaan 1, kode genap dan ganjil adalah *counter* yang didapatkan dari penelusuran objek. Jika arah penelusuran menunjukkan arah diagonal, maka kode ganjil bertambah 1 dan jika arah penelusuran menunjukkan arah diagonal, maka kode genap bertambah 1. Misalkan contoh kasus berikut: Piksel bernilai 0 merupakan daerah *background* dan piksel bernilai 1 merupakan piksel dari objek. Langkah selanjutnya melakukan penentuan arah dari piksel yang ditemukan pertama kali, seperti yang ditunjukkan pada Gambar 1. Setelah didapatkan piksel pertama, dilanjutkan dengan mencari piksel tetangga dengan urutan mata angin. Aplikasi akan mengecek piksel tetangga dimulai dari kanan terlebih dahulu, kemudian kanan bawah, bawah, kiri bawah, kiri, kiri atas, atas, dan terakhir kanan atas. Urutan proses akan diperlihatkan pada Gambar 2. Masing-masing arah akan akan dihitung jumlah arah ganjil dan genap sesuai arah mata angin dimulai dari kanan (0), kanan bawah (1), dan seterusnya. Setelah itu, akan didapat kode:

00 111 2 44444 5 00000 5 4444 7 00

Jumlah arah genap = 19 dan jumlah arah ganjil = 6. Kemudian jumlah genap dan ganjil dimasukkan ke persamaan 1:

$$\begin{aligned} \text{Total} &= 19 + \sqrt{2} * 6 \\ &= 27,4853 = 27 \text{ unit} \end{aligned}$$

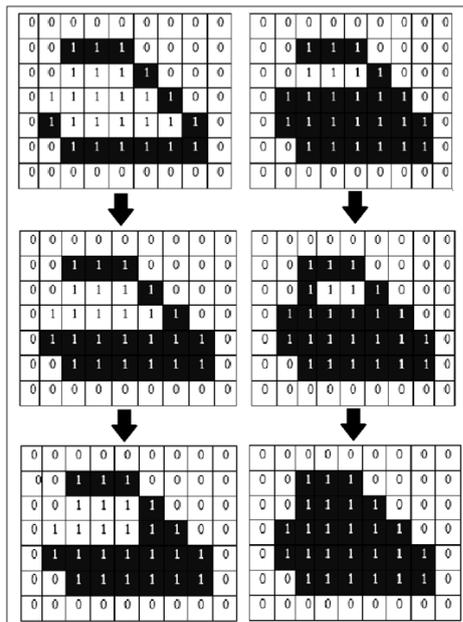
D. Rule Based System

Rule Based System (RBS) merupakan suatu sistem pakar yang menggunakan aturan-aturan untuk menyajikan pengetahuannya. Sistem berbasis aturan adalah suatu perangkat lunak yang menyajikan keahlian pakar dalam bentuk aturan-aturan pada suatu domain tertentu untuk menyelesaikan suatu permasalahan. *RBS* adalah model sederhana yang bisa diadaptasi kedalam masalah. Namun, jika aturan terlalu banyak, pemeliharaan sistem akan rumit dan terdapat banyak kesalahan dalam kerjanya [6].

Teori *RBS* ini menggunakan teknik yang sederhana. Dimulai dengan dasar aturan yang berisi semua pengetahuan dari permasalahan yang dihadapi kemudian dikodekan ke dalam aturan *if-then* yang mengandung data, pernyataan dan informasi awal. Sistem akan memeriksa semua aturan kondisi *if* yang menentukan subset, set konflik yang ada. Jika ditemukan, maka sistem akan melakukan kondisi *then*. Perulangan ini akan terus berlanjut hingga salah satu atau dua kondisi bertemu. Jika aturan tidak diketemukan, maka sistem tersebut harus keluar dari perulangan (*terminate*).

0	0	0	0	0	0	0	0	0	0
0	0	1	1	1	0	0	0	0	0
0	0	1	1	1	1	0	0	0	0
0	1	1	1	1	1	1	0	0	0
0	1	1	1	1	1	1	1	0	0
0	0	1	1	1	1	1	1	0	0
0	0	0	0	0	0	0	0	0	0

Gambar 1 Objek dalam citra biner



Gambar 2 Proses kode rantai

Aturan dapat direpresentasikan dalam bentuk:

$$if < condition > then < action > \quad (2)$$

Untuk mengelola aturan, terdapat dua pendekatan, yaitu [6]:

1. *Forward Chaining*: aturan diproses berdasarkan sejumlah fakta yang ada, dan didapatkan konklusi sesuai dengan fakta-fakta tersebut. Pendekatan *forward chaining* disebut juga *data driven*.
2. *Backward Chaining*: diberikan target, kemudian aturan yang aksinya mengandung target di-trigger. *Backward chaining* ini cocok untuk menelusuri fakta yang masih belum lengkap, disebut juga *goal driven*.

Alur proses metode *rule based* dapat direpresentasikan pada Tabel I.

E. Toleransi Batas

Adanya ketidaktepatan pada proses produksi yang tidak dapat dihindari, maka suatu alat tidak dapat dibuat setepat ukuran yang diminta. Agar persyaratan dapat dipenuhi maka ukuran pada sebuah benda kerja boleh terletak antara 2 batasan. Perbedaan 2 batas ukuran tersebut disebut toleransi [7]. Teori toleransi ini sering digunakan pada proses pengukuran dalam bidang industri. Toleransi terdiri dari batas atas dan batas bawah. Batas atas dinamakan dengan ukuran maksimum dan batas bawah dinamakan dengan ukuran minimum.

Ukuran maksimum dan ukuran minimum yang nantinya menjadi *rule* atau aturan yang dipakai saat proses penentuan. Jika perkiraan volume kurang dari ukuran minimum maka isi botol kurang, jika perkiraan volume lebih dari ukuran maksimum maka isi botol terlalu penuh, dan jika perkiraan

TABEL I
TAHAP *RULE BASED*

Tahap	Parameter	Proses
1	Citra neto	a) Segmentasi botol b) Segmentasi isi botol c) Deteksi atas botol dan permukaan isi d) Hitung jarak antara atas botol dengan permukaan isi Simpan jarak sebagai <i>rule</i> positif
2	Citra toleransi	a) Segmentasi botol b) Segmentasi isi botol c) Deteksi atas botol dan permukaan isi d) Hitung jarak antara atas botol dengan permukaan isi Simpan jarak sebagai <i>rule</i> negatif
3	Citra uji	a) Segmentasi botol b) Segmentasi isi botol c) Deteksi atas botol dan permukaan isi d) Hitung jarak antara atas botol dengan permukaan isi Simpan jarak sebagai jarak uji
4	a) <i>Rule</i> positif b) <i>Rule</i> negative c) Jarak uji	If (jarak uji > <i>rule</i> negatif) then (proses isi botol kurang) Else (lanjut ke proses selanjutnya)
5	a) <i>Rule</i> positif b) <i>Rule</i> negative c) Jarak uji	If (jarak uji < <i>rule</i> positif) then (proses isi botol lebih) Else (lanjut ke proses selanjutnya)
6	a) <i>Rule</i> positif b) <i>Rule</i> negative c) Jarak uji	If ((jarak uji ≥ <i>rule</i> positif) & (jarak uji ≤ <i>rule</i> negatif)) then (proses isi sesuai standar)

volume terletak di antara ukuran maksimum dan ukuran minimum maka botol tersebut memiliki volume yang ideal.

F. Standar Deviasi (*simpangan baku*)

Varian dan standar deviasi (*simpangan baku*) adalah ukuran-ukuran keragaman (*variasi*) data statistik yang paling sering digunakan. Standar deviasi adalah nilai statistik yang digunakan untuk menentukan bagaimana sebaran data dalam *sample* dan seberapa dekat titik data individu ke *mean* atau rata-rata.

Sebuah standar deviasi dari kumpulan data sama dengan nol menunjukkan bahwa semua nilai-nilai dalam himpunan tersebut adalah sama. Sebuah nilai deviasi yang lebih besar akan memberikan makna bahwa titik data individu jauh dari nilai rata-rata. Berikut adalah rumus standar deviasi:

$$S = \sqrt{\frac{\sum f(X - \bar{X})^2}{n - 1}} \quad (3)$$

- S = standar deviasi
- X = data uji ke-*n*
- \bar{x} = rata-rata data
- n = jumlah data

G. Permukaan Isi Botol

Pada industri minuman, takaran standar isi suatu produk minuman harus tepat. Mesin pengisi botol minuman sudah memiliki takaran dalam mengisi botol-botol tersebut sesuai dengan kapasitas botolnya. Akan tetapi, mesin-mesin akan membutuhkan *set up* ulang setelah digunakan dalam jangka waktu tertentu.

Permukaan isi suatu botol adalah indikator utama apakah botol terisi dengan benar atau tidak. Setelah mengetahui permukaan dari isi botol maka dapat diketahui jarak antara bagian atas botol dengan permukaan isi botol. Saat permukaan isi dari suatu botol menunjukkan perbedaan yang signifikan dari standar yang ada, maka sudah saatnya mesin dilakukan *set up* ulang.

III. HASIL DAN PEMBAHASAN

A. Urutan Proses Global

Gambar 3 adalah penjelasan mengenai urutan proses global dalam penelitian ini. Proses dimulai dari *start* dan berhenti pada *end*.

B. Analisis Masalah

Masalah yang timbul pada penelitian ini akan berkisar pada kondisi pencahayaan. Jika cahaya terlalu terang (*over exposure*), maka akan timbul efek pantulan (*refleksi*) dari botol. Akibatnya warna *background* citra terganggu dan hasil proses segmentasi terpengaruhi, yaitu saat ingin memisahkan wilayah kemungkinan botol (*foreground*) dengan wilayah bukan botol (*background*). Masalah ini dapat diatasi dengan mengatur sedemikian rupa kondisi tempat pengambilan citra, sehingga cahaya tidak melebihi batas normal (*over exposure*). Pada penelitian ini digunakan media yang terbuat dari *triplex* dengan ukuran alas 60x30cm dan ukuran *background* 60x50cm. Keduanya disusun 90 derajat sebagai *background* dan alas dari objek yang memiliki berwarna hitam.

C. Pengambilan Data Sampling

Data sampling pada penelitian ini terdiri dari 2 jenis, yaitu *data sampling* citra batas dan *data sampling* citra uji. *Data sampling* citra batas merupakan pemasukan citra yang akan dijadikan batas maksimum (neto botol) dan batas minimum (toleransi). *Data sampling* citra uji adalah proses pemasukan citra yang akan diuji. Proses pengukuran volume pada citra

menggunakan alat bantu gelas ukur dan alat suntik agar tepat pada volume yang ditargetkan.

Masukan citra diambil menggunakan kamera *action-cam Xiaomi yi 2* dengan resolusi 1920x2560 piksel dan berorientasi citra tegak (*potrait*). *Background* yang dipilih merupakan warna hitam. Alasan penggunaan warna hitam pada *background* citra karena warna hitam dalam model warna RGB 24 bit akan menghasilkan nilai komponen *red*, *green*, dan *blue* yang cukup kecil. Hal ini memudahkan dalam memisahkan wilayah kemungkinan botol (*foreground*) dengan wilayah bukan botol (*background*). Warna dari isi botol dapat berupa warna apa saja, kecuali warna hitam dan *transparan* (tembus pandang) karena *background* yang dipilih merupakan warna hitam dan warna botol yang digunakan adalah *transparan*/tembus pandang. Data yang digunakan dalam penelitian adalah citra RGB 24 bit dengan format *JPG*. Dalam 1 citra terdapat 1 objek botol.

Posisi kamera berada tegak lurus dengan garis permukaan isi sehingga permukaan tidak membentuk *curve* atau melengkung. Perubahan ukuran botol akan berpengaruh pada posisi kamera karena dibutuhkan pengaturan ulang agar posisi kamera tegak lurus dengan permukaan isi botol. Citra masukan direpresentasikan dalam RGB dengan ukuran 3x3 piksel.

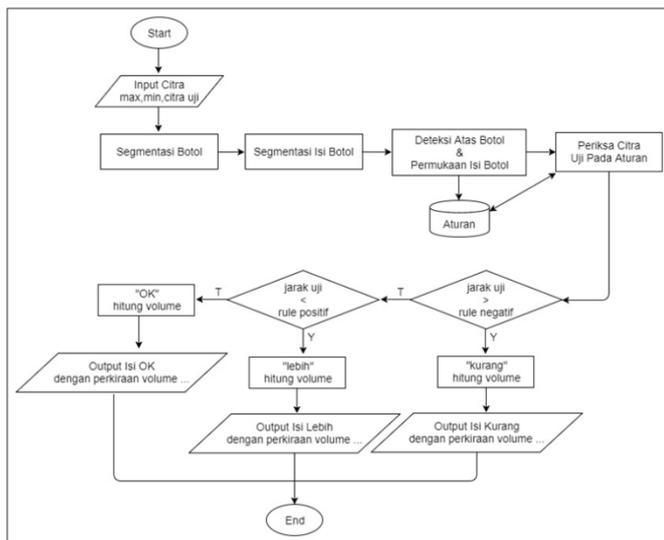
D. Segmentasi Botol

Tahap ini bertujuan untuk memisahkan wilayah kemungkinan botol (*foreground*) dari wilayah bukan botol (*background*) dengan menggunakan model warna RGB 24 bit dan memberi batas (*threshold*) pada setiap komponen warna merah, hijau, dan biru dari setiap piksel citra. Jika piksel citra memiliki nilai merah, hijau, dan biru ≤ 75 , maka piksel tersebut akan diubah menjadi warna putih.

Contoh perhitungan diketahui piksel dengan nilai ($R = 198, G = 140, B = 116$). Pertama-tama nilai RGB didapatkan dengan menggunakan fungsi *getRed()*, *getGreen()*, dan *getBlue()*. Setelah didapatkan nilai dari komponen warna *red*, *green*, dan *blue*, maka masuk pada aturan pertama: jika nilai dari ketiganya ≤ 75 , maka piksel tersebut akan diubah menjadi warna putih menggunakan fungsi *setRGB(x, y, 255)*

E. Segmentasi Isi Botol

Setelah tahap segmentasi botol dilakukan kemudian dilanjutkan dengan proses pemisahan wilayah kemungkinan isi botol dengan wilayah bukan isi botol (botol atau *backgro-*



Gambar 3 Urutan Proses Global

TABEL II
MASUKAN CITRA

R: 198	R: 110	R: 106
G: 140	G: 95	G: 115
B: 116	B: 76	B: 68
R: 188	R: 218	R: 94
G: 168	G: 128	G: 79
B: 144	B: 128	B: 56
R: 143	R: 254	R: 140
G: 153	G: 147	G: 106
B: 152	B: 169	B: 96

und). Tahap pertama adalah mengubah citra hasil segmentasi botol menjadi citra abu-abu menggunakan persamaan 2. Setelah didapatkan nilai rata-rata dari komponen *red*, *green*, dan *blue* maka masuk pada aturan: jika nilai rata-rata lebih kecil dari 70 (cenderung gelap), maka piksel tersebut diubah menjadi warna hijau. Warna hijau dipilih karena warna hijau unik untuk dijadikan tanda untuk proses berikutnya.

Proses ini dimulai dengan mengkonversi RGB ke arah abu-abu 32bit. Setelah didapatkan nilai keabuan dari setiap piksel, masuk ke peraturan yang ada, yaitu jika nilai rata-rata keabuan lebih kecil dari 70 maka piksel tersebut akan diubah menjadi warna hijau.

F. Deteksi Atas Botol

Tahap ini akan mencari bagian atas botol. Bagian atas botol didapatkan dengan mencari piksel dengan warna bukan putih pertama. Setelah itu, digunakan metode kode rantai untuk memastikan piksel tersebut merupakan wilayah atas botol atau bukan, bisa saja ada piksel lain yang bukan putih yang tidak merupakan wilayah botol. Aplikasi akan terus mencari sampai dapat atau sampai iterasi habis.

G. Deteksi Permukaan Isi Botol

Tahap ini akan mencari bagian atas permukaan isi botol. Bagian atas permukaan isi botol didapatkan dengan mencari piksel berwarna hijau pertama. Berikutnya dilakukan proses kode rantai untuk memastikan piksel tersebut merupakan permukaan isi botol atau bukan, bisa saja ada piksel lain yang berwarna hijau yang tidak merupakan wilayah botol. Aplikasi akan terus mencari sampai dapat atau sampai iterasi selesai.

H. Pengukuran (measurement)

Tahap ini merupakan proses terakhir dari aplikasi di mana akan dilakukan proses perhitungan volume dari citra uji. Proses perhitungan membutuhkan data dari fakta-fakta sebelumnya.

Contoh:

1. Fakta 1 (neto botol):
botol berisi 330ml (neto).
Setelah citra masuk ke aplikasi didapatkan nilai 600 piksel (misalkan)
2. Fakta 2 (toleransi):
botol berisi 315ml (toleransi).
Setelah citra masuk ke aplikasi didapatkan nilai 700 piksel (misalkan)
3. Fakta 3:
Ada 3 citra uji dengan jarak atas botol dengan permukaannya: 500px, 650px, 800px. Didapatkan setelah melalui proses segmentasi, deteksi, dan pengukuran.

Berdasarkan data di atas, maka dapat disimpulkan:

1. Citra 1: sebuah citra botol dengan jarak atas botol dan permukaan isi 500px
"isi botol terlalu penuh" karena (jarak permukaan uji < *rule* positif)
Perkiraan isi :
$$= [(500 - 600) / (700 - 600)] * (330-315)$$

$$= 100/100 * 15$$

$$= 15 \text{ mililiter}$$

$$= 330 + 15$$

$$= 345 \text{ mililiter}$$

2. Citra 2: sebuah citra botol dengan jarak atas botol dan permukaan isi 650px.

"isi botol pas" karena (*rule* positif \leq jarak permukaan uji \leq *rule* negatif)

Perkiraan isi :

$$= [(650 - 600) / (700 - 600)] * (330-315)$$

$$= 50/100 * 15$$

$$= 7.5 \text{ mililiter}$$

$$= 330 - 7.5$$

$$= 322.5 \text{ mililiter}$$

3. Citra 3: sebuah citra botol dengan jarak atas botol dan permukaan isi 800px.

"isi botol kurang" karena (jarak permukaan uji > *rule* negatif)

Perkiraan isi :

$$= [(800 - 600) / (700 - 600)] * (330-315)$$

$$= 200/100 * 15$$

$$= 30 \text{ mililiter}$$

$$= 330 - 30$$

$$= 300 \text{ mililiter}$$

I. Pengujian

Hasil akhir dari pengujian ini adalah kondisi penentuan akhir apakah citra uji masuk pada standar yang sudah ada pada proses pengambilan *rule* atas dan *rule* bawah. Tahap pengujian dilakukan pada 46 citra botol dengan kriteria: 15 citra dengan isi kurang dari batas minimum yaitu antara 300 – 314 dengan kenaikan 1 mililiter, 16 citra dengan isi memenuhi standar yaitu antara 315 – 330 dengan kenaikan 1 mililiter, 15 citra dengan isi lebih dari batas maksimum yaitu 331 – 345 dengan kenaikan 1 mililiter. Proses pengukuran volume pada citra menggunakan alat bantu gelas ukur dan alat suntik agar tepat. Untuk mengetahui tingkat akurasi aplikasi akan dibuat 3 buah pengujian yaitu sebagai berikut:

1. Pengujian dengan mengganti nilai *threshold* luas pada metode *Freeman Chain Code* untuk mengetahui nilai batasan yang paling cocok untuk penyaringan luas yang akan dibandingkan berdasarkan pengamatan manual.
2. Pengujian untuk mengukur tingkat ketelitian aplikasi dalam menentukan apakah isi botol memenuhi standar atau tidak.
3. Pengujian keandalan alat (sistem) untuk mengukur ketepatan aplikasi dalam menentukan apakah isi botol memenuhi standar atau tidak.

Pengujian menggunakan perbedaan warna, tekstur isi, dan jenis botol.

J. Analisis Hasil Pengujian

Berdasarkan penelitian dan pengujian yang telah dilakukan, dapat disimpulkan beberapa hal sebagai berikut:

1. Penggunaan metode *Freeman Chain Code* dalam penyaringan luas terbukti sangat berpengaruh pada akurasi aplikasi. Jika tidak memakai metode *Freeman*

Chain Code maka piksel-piksel yang tidak diinginkan akan ikut masuk pada proses dan akan mempengaruhi akurasi dari aplikasi. Berdasarkan pengujian yang sudah dilaksanakan. Akurasi pada nilai *threshold* 100 dan 200 adalah 100% dengan tingkat ketelitian 1,40. Akurasi pada nilai *threshold* 200 dan 300 adalah 100% dengan tingkat ketelitian 0,89. Akurasi pada nilai *threshold* 300 dan 400 adalah 95,7% dengan tingkat ketelitian 2,03. Akurasi pada nilai *threshold* 400 dan 500 adalah 91,3% dengan tingkat ketelitian 2,44. Jadi nilai *threshold* yang paling baik adalah 200 dan 300 dengan tingkat ketepatan 100% dan ketelitian 0,89 mililiter.

2. Metode *Rule Based System* dalam penelitian ini berperan penting. Akurasi tertinggi yang dicapai aplikasi pada pengujian yang sudah dilakukan adalah 100% dengan tingkat ketelitian 0,89 mililiter.
3. Hasil pengujian keandalan alat (sistem) 310 mililiter menghasilkan nilai rata-rata 311,98 mililiter dengan standar deviasi 1,13 sebelum dilakukan proses kalibrasi. Kalibrasi dilakukan dengan mengurangi atau menambah hasil perkiraan volume dengan nilai pergeseran antara rata-rata dengan nilai aslinya. Setelah kalibrasi nilai rata-rata yang didapat adalah 309,95 mililiter dengan standar deviasi 0,65. Kalibrasi terbukti mempengaruhi tingkat ketelitian alat.
4. Hasil pengujian isi botol berwarna hijau dari kelima citra dapat diproses dengan tepat dengan tingkat ketelitian 0,86. Hasil pengujian isi botol saus sambal dari kelima citra dapat diproses dengan tepat dengan tingkat ketelitian 0,96. Hasil pengujian botol 400 mililiter dari kelima citra dapat diproses dengan tepat dengan tingkat ketelitian 0,52.

IV. KESIMPULAN

Perangkat lunak dan perangkat keras telah berhasil dibuat dengan menggunakan metode *Freeman Chain Code* untuk penyaringan luas dan *Rule Based System* untuk alur kerja aplikasi, dengan tingkat akurasi paling tinggi adalah 100% dan tingkat ketelitian alat 0,89 mililiter.

Penggunaan metode *Freeman Chain Code* dalam penyaringan luas terbukti sangat berpengaruh pada akurasi aplikasi. Akurasi tertinggi yang berhasil didapat adalah pada nilai *threshold* 200 dan 300, yaitu 100% dengan tingkat ketelitian 0,89 mililiter. Metode *Rule Based System* dalam penelitian ini berperan penting. Akurasi tertinggi yang dicapai aplikasi pada pengujian yang sudah dilakukan adalah 100% dengan tingkat ketelitian 0,89 mililiter.

Berdasarkan hasil pencapaian dan pengujian yang telah dilakukan, terdapat beberapa saran untuk pengembangan yang

lebih lanjut untuk menghasilkan ketepatan yang lebih baik dalam pengecekan isi botol, seperti:

- Untuk meningkatkan kualitas citra hasil *pre-processing* sehingga wilayah botol dan isi botol yang didapatkan dapat maksimal, dapat dilakukan *noise reduction* untuk memperbaiki kualitas citra agar dapat memaksimalkan hasil segmentasi.
- Pada bagian proses segmentasi dapat ditambahkan proses deteksi tepi menggunakan metode *canny edge detector*. Metode deteksi tepi *canny* menggunakan *gaussian filter* untuk mengurangi *noise* sehingga akurasi aplikasi dapat bertambah.
- Untuk penelitian lebih lanjut, aplikasi dapat dikembangkan secara real time agar proses pengujian dapat lebih efektif.

DAFTAR REFERENSI

- [1] D. Taylor, D. Brunt. *Manufacturing Operations and Supply Chain Management*, 1st edition, Thomson Learning, 2001
- [2] A. F. Hastawan, A. Hidayatno, dan R. RizalIsnanto. "Deteksi Sudut Menggunakan Kode Rantai Untuk Pengenalan Bangun Datar Dua Dimensi," *Transmisi*, Vol. 15, No. 1, p. 1-7, 2013.
- [3] G. Rafael and W. Richard. *Digital Image Processing*, 3rd Edition, Pearson Education International, 2007.
- [4] J. F. Martinez, J. A. C. Ochoa. *Progressing Pattern Recognition, Image Analysis Recognition and Application*, Springer, 2006.
- [5] Y. Rijal, R. D. Ariefianto. "Deteksi wajah berbasis segmentasi model warna menggunakan *template matching* pada objek bergerak," *Seminar Nasional Aplikasi Teknologi Informasi 2008 (SNATI 2008)*, 21 Juni 2008, pp. 35-41.
- [6] Ligeza. *Logical Foundations for Rule-Based System*, Springer, 2006.
- [7] T. Sato dan N. S. Hartanto. *Menggambar Mesin Menurut Standar ISO*, PT Pradnya Paramita, 2003.

Ken Ratri Retno Wardani, menerima gelar Sarjana Teknik Informatika dari Sekolah Tinggi Sains Dan Teknologi Indonesia pada tahun 1993. Menyelesaikan studi Magister di Institut Teknologi Bandung jurusan Teknologi Informasi tahun 2004. Saat ini aktif sebagai pengajar di Departement Teknik Informatika, Institut Teknologi Harapan Bangsa di Bandung. Minat penelitian pada pengolahan citra dan Interaksi manusia Komputer.

Ari Setiawan, lahir pada tahun 1966 di Bandung, menerima gelar Sarjana Teknik dari ITB jurusan Teknik Mesin pada tahun 1990 dan gelar Magister Teknik dari ITB jurusan Teknik dan Manajemen Industri pada tahun 1997. Saat ini aktif sebagai Dosen Tetap di Departemen Teknik Industri ITHB. Minat penelitian: CIM, Proses Produksi, dan *Maintenance*.

Yohanes Setiawan, menempuh jenjang pendidikan S1 di Departemen Informatika ITHB dan lulus pada tahun 2018. Minat penelitian pada pemrograman.