

Klasifikasi Jenis Buah dan Sayuran Menggunakan SVM Dengan Fitur Saliency-HOG dan Color Moments

Yohannes^{1*)}, Muhammad Rizky Pribadi²⁾, and Leo Chandra³⁾

^{1, 2, 3)} Teknik Informatika, STMIK Global Informatika MDP, Indonesia

Corresponding Email: ^{*)}yohannesmasterous@mdp.ac.id

Abstract – Fruit is part of a plant that comes from the flower or pistil of the plant and usually has seeds. Meanwhile, vegetables are leaves, legumes, or seeds that can be cooked. Fruits and vegetables have many variants that can be distinguished based on color, shape, and texture. The Saliency-HOG feature and Color moments were used in this study to extract shapes and colors features in fruit and vegetable images. In this study, the Support Vector Machine (SVM) method was used to classify the types of fruit and vegetables. The dataset used in this study is a public dataset consisting of 114 images of fruit and vegetables. Each type of fruit and vegetable contains 100 images consisting of 70 images as training data and 30 images as testing data. There are 4 saliency features used in the testing phase, namely Region Contrast (RC), Frequency-tuned (FT), Histogram Contrast (HC), and Spectral Residual (SR). Based on the test results, the Saliency-HOG and Color Moments features were able to provide good results with the best precision, recall, and accuracy being 98.57%, 98.55%, and 99.120%, respectively.

Key words: Saliency-HOG, Color Moments, SVM, Fruits and Vegetables, Classification

I. PENDAHULUAN

Buah merupakan bagian dari suatu tumbuhan yang berasal dari bunga atau putik dari tumbuhan tersebut dan biasanya memiliki biji sedangkan sayur merupakan daun, tumbuhan polong, atau biji-bijian yang dapat dimasak. Buah dan sayuran mengandung berbagai macam vitamin, mineral, dan serat pangan yang berperan untuk membantu proses metabolisme dalam tubuh [1]. Vitamin dan mineral pada buah dan sayuran juga tidak dapat digantikan oleh bahan pangan lainnya [2].

Buah dan sayuran memiliki berbagai macam jenis yang memiliki bentuk dan warna yang berbeda dari setiap jenisnya [1]. Di antara keanekaragamannya, tentu ada beberapa jenis buah maupun sayuran yang memiliki bentuk dan warna yang hampir serupa namun memiliki kandungan vitamin dan gizi yang berbeda, misalnya buah apel dengan apel yang lain, masing-masing memiliki ukuran, warna, tekstur, bentuk, dan lain-lain. Namun terkadang masih banyak kesulitan yang terjadi dalam membedakan antar jenis buah dan sayur yang memiliki bentuk dan warna yang hampir serupa tersebut. Perkembangan teknologi tentunya dapat dikembangkan untuk membantu melakukan klasifikasi suatu objek

berdasarkan karakteristik yang dimiliki oleh objek tersebut.

Salah satu objek yang dapat dikenali dengan karakteristiknya adalah buah dan sayuran. Karakteristik dari buah dan sayuran sangat beragam sehingga dibutuhkan suatu metode untuk melakukan klasifikasi terhadap jenis-jenisnya. Klasifikasi buah merupakan salah satu penelitian yang cukup kompleks karena terdapat jenis tertentu memiliki bentuk yang serupa satu dengan yang lain. Buah dan sayuran memiliki karakteristik yang berbeda untuk setiap jenisnya [1]. Namun perlu diketahui bahwa karakteristik tersebut seringkali memiliki kemiripan sehingga hampir serupa [3]. Klasifikasi buah dan sayuran tentunya dapat dilakukan dengan mengenali karakteristik berupa bentuk dan warna yang tentunya berbeda antara satu dengan yang lain. Untuk membantu dalam mengidentifikasi buah dan sayuran, *computer vision* dapat digunakan untuk mengenali sebuah objek berdasarkan analisis fitur-fitur tertentu.

Telah banyak metode yang diterapkan untuk klasifikasi, seperti *Support Vector Machine* (SVM), *k-Nearest Neighbor* (k-NN), dan *Convolutional Neural Network* (CNN), dan lain-lain. Penggunaan metode SVM memiliki tingkat akurasi yang lebih besar dibandingkan dengan metode *k-Nearest Neighbor* (k-NN) [4], [5], [6]. Untuk mengenali objek dari citra *digital* diperlukan fitur-fitur pendukung yang dapat membantu mengklasifikasikan objek, fitur pertama adalah fitur bentuk dan tekstur, seperti *Histogram of Oriented Gradients* (HOG), *Gaussian Mixture Model* (GMM), *Local Binary Pattern* (LBP), dan *Saliency*.

HOG merupakan fitur yang biasa digunakan untuk merepresentasikan bentuk dari sebuah citra berupa histogram. HOG bekerja dengan menangkap struktur tepi atau gradien yang merupakan karakteristik dari bentuk lokal sebuah citra [7]. Bentuk lokal yang dihasilkan oleh HOG diperoleh dengan cara yang relatif sehingga tidak terpengaruh oleh perubahan tampilan pada sebuah citra. Hal ini membuat HOG menjadi unggul untuk dapat mengambil karakteristik bentuk objek yang ada pada sebuah citra. Tidak hanya itu, HOG mampu menjadi fitur untuk deteksi manusia dan lebih unggul dibandingkan GMM [8]. HOG juga telah digunakan untuk identifikasi jenis mangga yang mempunyai akurasi sebesar 90% [9], dan pengenalan plat kendaraan dengan akurasi tinggi [10].

Pada umumnya, banyak sekali objek yang terkandung pada sebuah citra baik itu objek utama, latar objek yang

bervariasi dengan warna yang kompleks sampai pada objek yang tidak diinginkan ada pada citra tersebut. Hal ini membuat sulitnya untuk mengambil fitur pada objek yang diinginkan dalam sebuah citra, sehingga dibutuhkan proses pemisahan objek utama dengan objek lain untuk mendapatkan objek yang diinginkan agar fitur dapat bekerja sesuai dengan objek utama. Salah satu metode untuk mendapatkan objek utama dari sebuah citra adalah *saliency*. *Saliency* bekerja dengan mendapatkan wilayah yang dominan dari keseluruhan objek pada sebuah citra berdasarkan sebaran intensitas *pixel*. Dominasi objek pada *saliency* ditentukan dengan perhitungan *distance* pada setiap wilayah intensitas *pixel* masing-masing. *Saliency* mampu menonjolkan objek utama dan meniadakan objek lain seperti *background* dan *noise* pada sebuah citra dengan variasi atribut warna, gradien, tepi, dan batas-batasnya [11].

Fitur selanjutnya adalah fitur warna. Salah satu fitur warna adalah *Color Moments* yang digunakan untuk mengukur kemiripan warna antar citra *digital*. Komponen pada *color moments* mampu menjadi fitur untuk mencocokkan sebuah citra berdasarkan variasi warna yang terkandung pada konten citra tersebut.

Saliency dan HOG telah digunakan sebagai fitur pada klasifikasi wajah mamalia [12], [13], [14]. Fitur *Saliency-HOG* mampu mengklasifikasi wajah mamalia tampak depan sehingga fitur ini akan digunakan untuk klasifikasi jenis buah dan sayuran dimana algoritma yang dapat bekerja paling efisien adalah SVM, dan akan ditambahkan fitur warna *Color Moments* untuk membantu ekstraksi warna pada klasifikasi jenis buah dan sayuran ini. Pada penelitian ini terdapat perbandingan *saliency* [15], [16], dimana fitur HOG dilakukan pada citra hasil *saliency*. Beberapa metode *saliency* yang telah digunakan adalah *Region Contrast* (RC), *Frequency-tuned* (FT), *Histogram Contrast* (HC), dan *Spectral Residual* (SR). Keempat *saliency* ini telah digunakan namun untuk kasus deteksi penyakit malaria [17].

Berdasarkan penelitian sebelumnya, klasifikasi jenis buah dan sayuran hanya menggunakan fitur bentuk dan warna. Namun pada penelitian ini digunakan fitur bentuk berbasis *saliency* (*Saliency-HOG*) dan *Color Moments* sebagai fitur warna untuk klasifikasi buah dan sayuran menggunakan metode klasifikasi *Support Vector Machine* (SVM). Objek yang digunakan pada penelitian ini adalah citra buah dan sayuran.

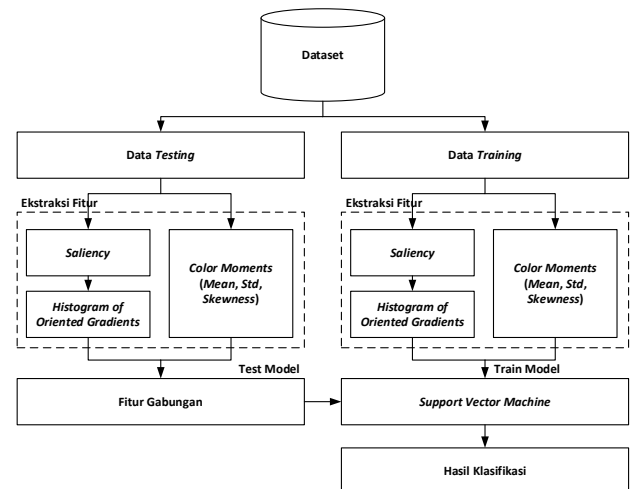
II. METODOLOGI

Pada penelitian ini, data yang digunakan merupakan objek berupa citra buah dan sayuran yang diambil dari dataset Fruit 360 [18]. Versi dataset Fruit 360 yang digunakan adalah versi 57 (2019.07.07.0) yang berjumlah 11.400 citra dengan 114 jenis buah dan sayuran [19]. Untuk setiap jenis buah dan sayuran yang digunakan adalah 100 citra dengan ukuran 100 x 100 *pixel* dengan format JPG. Citra buah dan sayuran yang digunakan pada dataset Fruit 360 dapat dilihat pada **Gambar 1**.



Gambar 1. Dataset Fruit 360

Tahapan yang dilakukan pada penelitian ini dapat dilihat pada **Gambar 2**. Tahapan penelitian dimulai dengan melakukan ekstraksi fitur bentuk dan warna. Ekstraksi fitur bentuk yang digunakan adalah metode *Saliency-HOG* dimana input citra dilakukan *saliency* terlebih dahulu kemudian dilanjutkan dengan mengambil bentuk menggunakan metode HOG dari hasil *saliency*. Kemudian untuk fitur warna yang digunakan adalah *color moments* dimana proses untuk ekstraksi fitur ini terpisah dengan *Saliency-HOG*. *Color Moments* diambil dari input citra tanpa *saliency* melalui ruang warna *red*, *green*, dan *blue*. Kedua fitur ini kemudian dibentuk menjadi satu fitur, yaitu fitur gabungan *Saliency-HOG* dan *Color Moments*. Fitur gabungan *Saliency-HOG* dan *Color Moments* ini akan digunakan untuk tahapan pelatihan dan pengujian.



Gambar 2. Metode yang diusulkan

Pada tahapan pelatihan, data latih yang digunakan merupakan data fitur gabungan *Saliency-HOG* dan *Color Moments*. Tahap pelatihan (*training*) data akan menghasilkan model klasifikasi. Metode klasifikasi yang digunakan adalah *Support Vector Machine* (SVM). Setelah model klasifikasi dihasilkan, maka dilanjutkan dengan tahap pengujian (*testing*). Pada tahap pengujian, model klasifikasi akan diuji untuk mendapatkan hasil klasifikasi. Metode *Saliency* yang digunakan pada *Saliency-HOG* sebagai komparasi perbandingan adalah *Region Contrast* (RC), *Frequency-tuned* (FT), *Histogram Contrast* (HC), dan *Spectral Residual* (SR).

A. Region Contrast Saliency

Region Contrast merupakan metode *saliency* yang memetakan hubungan spasial pada *region* citra berdasarkan nilai kontras. *Region contrast* mula-mula membagi sebuah citra ke dalam beberapa *region* dan membentuk histogram warna pada tiap *region*-nya [15]. Kontras warna pada setiap *region* dihitung menggunakan Persamaan (1).

$$S(r_k) = \sum_{r_k \neq r_i} w(r_i) D_r(r_k, r_i) \quad (1)$$

Nilai *saliency* $S(r_k)$ pada Persamaan (1) dihitung berdasarkan bobot tiap *region* $w(r_i)$ dan jarak warna $D_r(r_k, r_i)$ antar *region* r_k terhadap *region* r_i lainnya. Jarak warna antar *region* $D_r(r_k, r_i)$ dapat dihitung menggunakan Persamaan (2) dimana $f(c_{k,i})$ adalah frekuensi kemunculan warna c_k ke- i terhadap semua warna n_k dan $D(c_{1,i}, c_{2,j})$ adalah jarak warna antara *pixel* $c_{k,i}$ pada *region* yang satu terhadap lainnya.

$$D_r(r_1, r_2) = \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} f(c_{1,i}) f(c_{2,j}) D(c_{1,i}, c_{2,j}) \quad (2)$$

B. Histogram Contrast Saliency

Histogram Contrast merupakan metode perhitungan *saliency* berdasarkan statistik warna dimana hasil *saliency* mengacu pada perbedaan kontras warna terhadap keseluruhan wilayah lainnya pada sebuah citra. Karena metode ini melibatkan semua *pixel* pada citra maka dapat dilakukan pengurangan jumlah warna *pixel* pada citra sehingga waktu komputasi *saliency* menjadi lebih cepat dan efisien. *Histogram Contrast* merupakan metode *saliency* yang sederhana dan mampu memberikan hasil *saliency* yang baik dibandingkan dengan *saliency* lainnya [15]. Secara khusus, nilai-nilai *pixel saliency* dari sebuah *pixel* I_k pada citra I didefinisikan dengan Persamaan (3).

$$S(I_k) = \sum_{I_i \in I} D(I_k, I_i) \quad (3)$$

Dimana $D(I_k, I_i)$ adalah matriks jarak warna antara *pixel* I_k dan I_i di ruang warna L^*a^*b dan diperluas dengan urutan *pixel* yang didefinisikan dengan Persamaan (4).

$$S(I_k) = D(I_k, I_1) + D(I_k, I_2) + \dots + D(I_k, I_N) \quad (4)$$

Dimana N adalah jumlah *pixel* dalam citra I . *Pixel* dengan warna memiliki *saliency* yang sama dengan hubungan spasial nilai warna yang sama C_j dikelompokkan bersama-sama, dan mendapatkan nilai *saliency* untuk setiap warna yang didefinisikan dengan Persamaan (5).

$$S(I_k) = S(c_i) = \sum_{j=1}^N f_j D(c_i, c_j) \quad (5)$$

Dimana c_i adalah nilai warna *pixel* I_k , N adalah nilai warna *pixel* yang berbeda, dan f_j adalah probabilitas *pixel* warna c_j pada citra I .

C. Spectral Residual Saliency

Spectral Residual merupakan metode *saliency* yang menggunakan *log-spectrum* sebagai dasar untuk menghasilkan *saliency map*. Pembentukan *spectral residual* dilakukan dengan melakukan transformasi *log-spectrum* yang didapatkan menggunakan prinsip *pre-attentive visual search* dalam bentuk *spatial domain*.

Berawal dari prinsip *natural image* yang selalu mengandalkan *visual search* maka *spectral residual saliency* mampu menjadi metode *saliency* dengan performa yang lebih cepat dengan waktu komputasi FFT yang cukup singkat [16]. FFT dilakukan menggunakan *mean filter* dengan ukuran 3×3 . Operasi ini dilakukan dalam satu dimensi dengan membentuk residu $R(k)$ yang didefinisikan menggunakan Persamaan (6).

$$R(k) = \ln|X(k)| - g_n * \ln|X(k)| \quad (6)$$

Dengan $g_n = [\frac{1}{3}, \frac{1}{3}, \frac{1}{3}]$, dan $*$ menunjukkan operasi konvolusi. Operasi sederhana ini menunjukkan bahwa (1-D) residu spektral $R(k)$ didefinisikan dengan persamaan (7).

$$R(k) = \frac{1}{3} \ln \left[\frac{|X(k)|^2}{|X(k-1)||X(k+1)|} \right] \quad (7)$$

D. Frequency-tuned Saliency

Frequency-tuned merupakan metode *saliency* yang dihasilkan dengan menghitung perbedaan warna terhadap rata-rata dan pencahayaan pada sebuah citra. Dengan perbedaan warna yang dihasilkan, *Frequency-tuned saliency* mampu menentukan wilayah tiap objek pada citra dengan baik, bekerja pada resolusi citra yang tinggi, dan memiliki waktu komputasi yang efisien [20]. Metode *saliency* ini memiliki hasil yang sangat baik karena frekuensi perbedaan penyebaran warna yang unggul terhadap wilayah warna secara keseluruhan. Metode ini dapat menentukan *saliency map* dengan Persamaan (8).

$$S(x, y) = |I_\mu - I_{whc}(x, y)| \quad (8)$$

Dimana I_μ adalah nilai *pixel* rata-rata aritmatik dari citra dan I_{whc} adalah versi Gaussian dari citra asli untuk menghilangkan detil tekstur halus serta *noise* dan *coding artifacts*. Pada metode ini, *Norm* digunakan karena hanya tertarik pada besarnya perbedaan. Komputasi ini cukup memakan waktu karena beroperasi pada citra asli tanpa *downsampling* dimana memperoleh *saliency map* secara resolusi penuh.

E. Histogram of Oriented Gradient

Histogram of Oriented Gradients (HOG) merupakan salah satu metode ekstraksi ciri untuk mendeteksi suatu objek dengan representasi bentuk. HOG bekerja dengan menangkap histogram dari orientasi gradien pada setiap *cell* yang terbagi dari sebuah citra [7]. Pembentukan fitur HOG dimulai dengan menghitung nilai *gradient image*, orientasi *binning*, dan normalisasi blok pada tiap *bin*. Persamaan (9) dan (10) merupakan persamaan untuk menghitung *gradient image* I dimana r dan c masing-masing menyatakan posisi baris dan kolom dari *image*.

$$I_x(r, c) = (r, c + 1) - (r, c - 1) \quad (9)$$

$$I_y(r, c) = (r + 1, c) - (r - 1, c) \quad (10)$$

Besar *gradient image* (μ) dan orientasi (θ) selanjutnya dapat dihitung berdasarkan *gradient image* masing-masing, yaitu I_x dan I_y yang telah diperoleh menggunakan Persamaan (11) dan (12).

$$\mu = \sqrt{I_x^2 + I_y^2} \quad (11)$$

$$\theta = \frac{180}{\pi} (\tan^{-1}(I_y, I_x) \bmod \pi) \quad (12)$$

Selanjutnya nilai tiap orientasi sel histogram untuk bin j , C_i , dan w ditentukan menggunakan Persamaan (13), (14), dan (15) dengan batasan w untuk selang interval tertutup w_i dan w_{i+1} .

$$W = \frac{180}{B} \quad (13)$$

$$bin\ j = \left[\frac{\theta}{w} - \frac{1}{2} \right] \bmod B \quad (14)$$

$$C_i = w(i + \frac{1}{2}) \quad (15)$$

Kemudian untuk nilai pada tiap orientasi sel histogram dilakukan *vote bin* V_j dan V_{j+1} sesuai pada Persamaan (16) dan (17) dimana B adalah jumlah *bin*.

$$V_j = \mu \frac{c_j + 1}{w} \quad (16)$$

$$V_{j+1} = \mu \frac{\theta - c_j}{w} \quad (17)$$

Setelah proses *vote bin* maka dilakukan tahap blok normalisasi untuk mengatasi *overlapping* dengan Persamaan (18) dimana b merupakan nilai pada *bin* dalam sebuah blok.

$$b \leftarrow \frac{b}{\sqrt{|b|^2 + \varepsilon}} \quad (18)$$

F. Color Moments

Color moments merupakan metode untuk mendeskripsikan citra berdasarkan distribusi warna dengan menggunakan tiga *moment* utama, yaitu *mean*, *standard deviation*, dan *skewness*. Tiga *moment* ini merupakan *moment* yang mampu mewakili informasi warna pada sebuah citra [21]. *Mean*, *standard deviation*, dan *skewness* pada *color moments* masing-masing dapat dihitung pada Persamaan (19), (20), dan (21).

$$E_i = \frac{1}{N} \sum_{j=1}^N P_{i,j} \quad (19)$$

$$\sigma_i = \sqrt{\left(\frac{1}{N} \sum_{j=1}^N (P_{i,j} - E_i)^2 \right)} \quad (20)$$

$$s_i = \sqrt[3]{\left(\frac{1}{N} \sum_{j=1}^N (P_{i,j} - E_i)^3 \right)} \quad (21)$$

Dimana :

P_{ij} : nilai warna pada *pixel* ke- j pada *channel* warna ke- i

N : banyaknya *pixel* yang terdapat pada citra

E_i : nilai *mean* dari *channel* warna ke- i

σ_i : nilai *standard deviation* dari *channel* warna ke- i

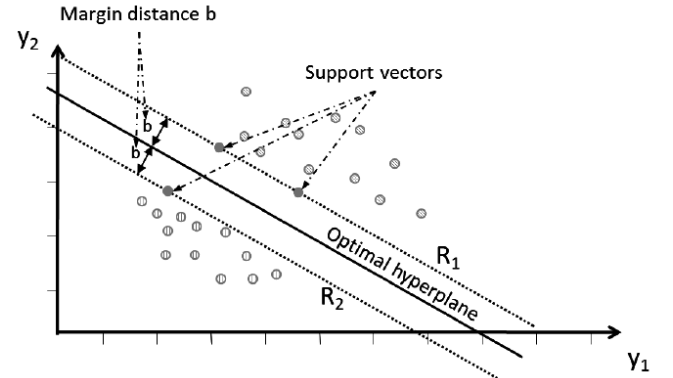
s_i : nilai *skewness* dari *channel* warna ke- i

G. Support Vector Machine

Support Vector Machine (SVM) merupakan metode pembelajaran mesin yang banyak digunakan untuk masalah klasifikasi dan regresi. Metode ini termasuk dalam metode klasifikasi linier biner diskriminatif (non-probabilistik). SVM sering kali digunakan daripada metode pembelajaran mesin lainnya karena lebih sederhana, memiliki komputasi yang relatif lebih sedikit dan mampu menghasilkan akurasi yang baik [22].

Prinsip dasar SVM adalah menemukan *hyperplane* dalam ruang berdimensi N yang mampu memisahkan titik data dalam dua kelas, dimana N adalah jumlah fitur. Jika data dapat dipisahkan secara linier maka dapat ditemukan dua *hyperplane* paralel yang memisahkan dua kelas data.

Gambar 3. menunjukkan dua *hyperplane* R_1 dan R_2 yang diwakili dengan garis putus-putus. Titik data yang berada pada dua *hyperplane* disebut sebagai *support vector* yang mana akan mempengaruhi posisi dan orientasi *hyperplane*. *Hyperplane* yang terletak di tengah dua *hyperplane* adalah *hyperplane* dengan *margin* maksimum.



Gambar 3. Optimal Hyperplane SVM [22]

Metode SVM bertujuan untuk menemukan *hyperplane* dengan jarak maksimum dari data pelatihan terdekat. *Hyperplane* dapat didefinisikan pada Persamaan (22) dimana w^T adalah vektor bobot, x adalah vektor input, dan b adalah bias.

$$w^T x + b = 0 \quad (22)$$

Untuk vektor bobot w dan bias b tertentu, maka dapat ditentukan dengan Persamaan (23) dan (24) dimana d adalah *margin* pemisah antara *hyperplane* dan titik data terdekat.

$$w^T x + b \geq 0, \text{ untuk } d_i = +1 \quad (23)$$

$$w^T x + b < 0, \text{ untuk } d_i = -1 \quad (24)$$

SVM mampu mempelajari data untuk menghasilkan model klasifikasi dengan melibatkan fungsi *kernel* untuk menemukan *hyperplane* terbaik [23]. *Kernel* yang digunakan pada penelitian ini, yaitu *linear* yang ditunjukkan pada Persamaan (25).

$$K(x, x_i) = x \cdot x_i \quad (25)$$

H. Confusion Matrix

Cara umum untuk memvisualisasikan hasil klasifikasi adalah dengan menggunakan tabel kontingensi yang disebut *confusion matrix*. Baris dari *confusion matrix* mewakili nilai yang diprediksi sedangkan kolom mewakili nilai sebenarnya. *Confusion matrix* bisa memiliki dua kolom atau lebih bergantung pada jumlah *label* atau kelas [22]. Contoh *confusion matrix* dengan 2 kelas dapat dilihat pada **Tabel 1**.

Tabel 1. *Confusion matrix* untuk 2 kelas

		Actual class	
		Kelas A	Kelas B
Predicted class	Kelas A	True Positive (TP)	False Positive (FP)
	Kelas B	False Negative (FN)	True Negative (TN)

Dimana :

TP : banyaknya data positif yang dianggap benar oleh sistem

TN : banyaknya data negatif yang dianggap benar oleh sistem

FN : banyaknya data negatif yang dianggap salah oleh sistem

FP : banyaknya data positif yang dianggap salah oleh sistem

Pengukuran kinerja hasil klasifikasi menggunakan *confusion matrix* dilakukan dengan menghitung nilai *precision*, *recall*, dan *accuracy* dengan Persamaan (26), (27), dan (28).

$$Precision = \frac{TP}{TP+FP} \quad (26)$$

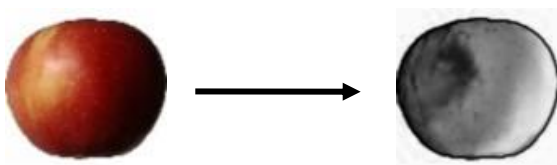
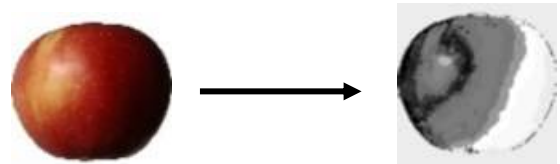
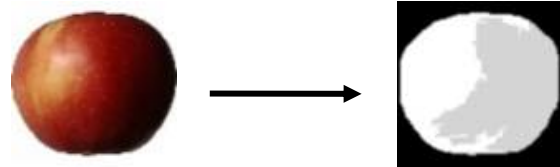
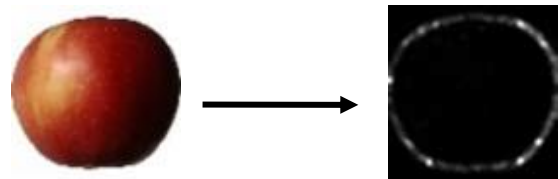
$$Recall = \frac{TP}{TP+FN} \quad (27)$$

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (28)$$

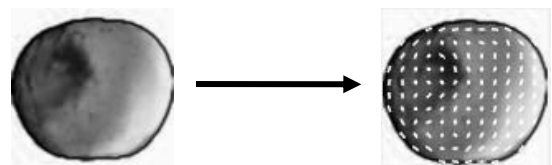
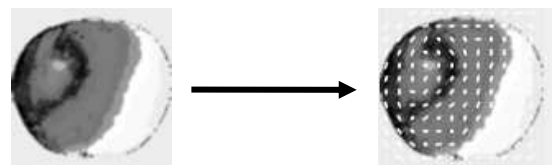
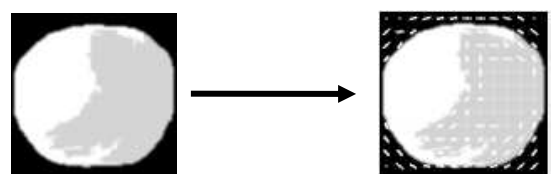
III. HASIL DAN PEMBAHASAN

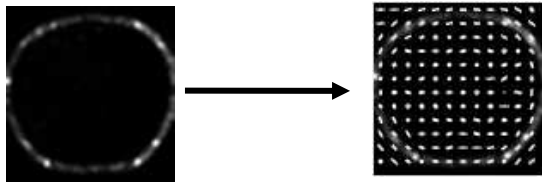
Perbandingan data yang digunakan pada penelitian ini adalah 70% untuk data training dan 30% untuk data testing. Dengan perbandingan tersebut maka dari 100 citra pada setiap jenis buah dan sayuran terdapat 70 citra untuk data training dan 30 citra untuk data testing sehingga secara keseluruhan terdapat 7.980 untuk data training dan 3.420 untuk data testing dengan 114 jenis buah dan sayuran. Untuk tahap pengujian digunakan 4 skenario, yaitu penggunaan kombinasi *saliency* FT-HOG, HC-HOG, RC-HOG, dan SR-HOG terhadap *Color Moments*.

Hasil beberapa metode *Saliency* pada citra buah dan sayuran, yaitu *Frequency-tuned* (FT), *Histogram Contrast* (HC), *Region Contrast* (RC), *Spectral Residual* (SR) dapat dilihat pada **Gambar 4**, **Gambar 5**, **Gambar 6**, dan **Gambar 7**. Berdasarkan perbandingan hasil metode *saliency* terhadap citra buah dan sayuran, metode *saliency* FT terlihat lebih mendapatkan hasil bentuk yang lebih baik dibandingkan dengan *saliency* HC, RC, dan SR. Hasil *saliency* SR terlihat kurang baik dari ketiga *saliency* lainnya.

**Gambar 4.** *Saliency* FT**Gambar 5.** *Saliency* HC**Gambar 6.** *Saliency* RC**Gambar 7.** *Saliency* SR

Citra *saliency* baik dari hasil *saliency* FT, HC, RC, maupun SR selanjutnya dilakukan ekstraksi fitur menggunakan metode HOG. Fitur HOG akan mengambil histogram dari gradien dan orientasi citra hasil *saliency* dengan ukuran *cell* sebesar 4 dan *block* berjumlah 11x11. Karena satu *block* terdiri dari sembilan bin, maka didapatkan 4.356 data fitur untuk setiap citra *saliency* dalam bentuk vektor. Visualisasi hasil metode HOG terhadap *saliency* FT, HC, RC, dan SR berturut-turut dapat dilihat pada **Gambar 8**, **Gambar 9**, **Gambar 10**, dan **Gambar 11**.

**Gambar 8.** Visualisasi *Saliency* FT-HOG**Gambar 9.** Visualisasi *Saliency* HC-HOG**Gambar 10.** Visualisasi *Saliency* RC-HOG



Gambar 11. Visualisasi *Saliency* SR-HOG

Pada tahap ekstraksi fitur warna, *Color Moments* mengambil nilai *pixel* dalam bentuk RGB dari citra asli dan mengkonversi nilai RGB dengan cara menghitung *Moment* (mean, standar deviasi, dan *skewness*) yang dapat dihitung pada masing-masing ruang warna RGB pada setiap warna. Selanjutnya dibandingkan antar gambar pada citra asli dengan *color moments*, perbandingan citra dilakukan dengan menghitung jarak *moments*.

Hasil ekstraksi fitur *Saliency*-HOG berupa *feature vector* digabungkan dengan tiga komponen utama *color moments* pada ruang warna RGB, yaitu *mean* (meanR, meanG, meanB), standar deviasi (stdR, stdG, stdB), dan *skewness* (skewR, skewG, skewB) sehingga membentuk fitur gabungan *Saliency*-HOG dan *Color Moments*. Fitur gabungan ini digunakan pada proses *training* menggunakan metode SVM untuk membentuk model klasifikasi. Pengujian model klasifikasi dilakukan dengan fitur yang sama, yaitu *Saliency*-HOG dan *Color Moments* pada data *testing*.

Dari hasil penelitian yang telah dilakukan pada 4 skenario didapatkan nilai *precision*, *recall*, dan *accuracy* yang dihitung menggunakan *confusion matrix* untuk mendapatkan tingkat keberhasilan klasifikasi. Hasil pengujian didapatkan dari perhitungan rata-rata *precision*, *recall*, dan *accuracy* untuk jenis buah dan sayuran. Hasil yang didapatkan pada pengujian menggunakan gabungan fitur *saliency* yang berbeda, yaitu FT-HOG, HC-HOG, RC-HOG, dan SR-HOG terhadap *Color Moments* dapat dilihat pada Tabel 2.

Tabel 2. Hasil *Precision*, *Recall*, dan *Accuracy* pada tiap Fitur

	FT-HOG dan Color Moments	HC-HOG dan Color Moments	RC-HOG dan Color Moments	SR-HOG dan Color Moments
Precision	98,57%	98,54%	98,26%	98,51%
Recall	98,55%	98,52%	98,23%	98,49%
Accuracy	99,120%	99,119%	99,114%	99,119%

Walaupun metode *Saliency* FT, HC, RC, dan SR bekerja dengan cara yang berbeda untuk menemukan wilayah dominan dari sebuah citra, hasil *saliency* tersebut secara keseluruhan mampu memberikan bentuk luar dari citra buah dan sayuran sehingga ekstraksi bentuk yang didapatkan dari HOG memberikan hasil yang baik. Hal ini dapat dilihat dari hasil *precision*, *recall*, dan *accuracy* untuk *saliency* FT, HC, RC, dan SR yang tidak terlalu berbeda jauh. Di samping itu, tiga komponen utama pada *Color Moments* yang dilibatkan mampu memberikan hasil klasifikasi yang baik.

IV. KESIMPULAN

Berdasarkan hasil pengujian dapat disimpulkan bahwa metode *Saliency*-HOG dan *Color Moments* mampu mengekstraksi fitur pada buah dan sayuran dengan baik dan metode SVM mampu mengklasifikasikan buah dan sayuran dengan tingkat akurasi yang berbeda untuk setiap *saliency* yang digunakan. Penggunaan *Saliency*-HOG sebagai fitur bentuk berbasis *saliency* mampu menjadi salah satu kombinasi fitur yang dapat memberikan representasi bentuk berdasarkan wilayah dominan dari sebuah citra. Performa klasifikasi jenis buah dan sayuran menggunakan fitur *Saliency*-HOG dan *Color Moments* mampu memberikan hasil yang baik dengan *precision*, *recall*, dan *accuracy* terbaik masing-masing adalah 98,57%, 98,55%, dan 99,120%. Penggunaan metode segmentasi di luar metode *saliency* dapat digunakan sebagai alternatif kombinasi fitur bentuk untuk saran penelitian selanjutnya.

DAFTAR PUSTAKA

- [1] Kementerian Kesehatan RI, "Tingkatkan Konsumsi Sayur dan Buah Nusantara Menuju Masyarakat Hidup Sehat," *Biro Komunikasi dan Pelayanan Masyarakat Sekretariat Jenderal Kementerian Kesehatan RI*, 2017. [Online]. Available: <https://www.kemkes.go.id/article/view/17012500002/tingkatkan-konsumsi-sayur-dan-buah-nusantara-menuju-masyarakat-hidup-sehat-.html>. [Accessed: 03-Jan-2020].
- [2] Y. Takaoka and N. Kawakami, "Fruit and vegetable consumption in adolescence and health in early adulthood: a longitudinal analysis of the Statistics Canada's National Population Health Survey," *BMC Public Health*, vol. 13, p. 1206, 2013, doi: 10.1186/1471-2458-13-1206.
- [3] Meiriyama, "Klasifikasi Citra Buah Berbasis Fitur Warna HSV dengan Klasifikator SVM," *J. Komput. Terap.*, vol. 4, no. 1, pp. 50–61, 2018.
- [4] S. R. G. Pertiwi, "Perbandingan metode k-nearest neighbor dan support vector machine dalam analisis sentimen twitter terhadap stasiun televisi berita Indonesia," Universitas Gadjah Mada, 2018.
- [5] D. Nurdiah and I. A. Muwahid, "Perbandingan Support Vector Machine dan K-Nearest Neighbor untuk Klasifikasi Telur Fertil dan Infertil Berdasarkan Analisis Texture GLCM," *J. Transform.*, vol. 13, no. 2, p. 29, 2016, doi: 10.26623/transformatika.v13i2.324.
- [6] M. Ichwan, I. A. Dewi, and Z. M. S., "Klasifikasi Support Vector Machine (SVM) Untuk Menentukan Tingkat Kemanisan Mangga Berdasarkan Fitur Warna," *MIND J.*, vol. 3, no. 2, pp. 16–23, 2018.
- [7] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, 2005, vol. 1, pp. 886–893, doi: 10.1109/CVPR.2005.177.
- [8] Y. F. Munawaroh and I. Salamah, "Analisa Perbandingan Algoritma Histogram of Oriented Gradient (HOG) dan Gaussian Mixture Model (GMM) Dalam Mendeteksi Manusia," *Semin. Nas. Inov. dan Apl. Teknol. Di Ind.* 2018, vol. 4, no. 2, pp. 251–255, 2018.

- [9] E. Utama, F. Yapputra, and Gasim, "Identifikasi Jenis Mangga Berdasarkan Bentuk Menggunakan Fitur HOG dan Jaringan Syaraf Tiruan," *J. Ilm. Inform. Glob.*, vol. 9, no. 1, pp. 1–6, 2018, doi: 10.36982/jig.v9i1.437.
- [10] G. Ning, "Vehicle License Plate Detection and Recognition," University of Missouri, Columbia, 2013.
- [11] A. Borji and L. Itti, "State-of-the-Art in Visual Attention Modeling," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 185–207, 2013, doi: 10.1109/TPAMI.2012.89.
- [12] Y. Yohannes, Y. P. Sari, and I. Feristyani, "Klasifikasi Wajah Hewan Mamalia Tampak Depan Menggunakan k-Nearest Neighbor Dengan Ekstraksi Fitur HOG," *J. Tek. Inform. dan Sist. Inf.*, vol. 5, no. 1, pp. 84–97, 2019, doi: 10.28932/jutisi.v5i1.1584.
- [13] M. E. Al Rivan and Y. Yohannes, "Klasifikasi Mamalia Berdasarkan Bentuk Wajah Dengan K-NN Menggunakan Fitur CAS Dan HOG," *J. Tek. Inform. dan Sist. Inf.*, vol. 5, no. 2, pp. 173–180, 2019.
- [14] Y. Yohannes and M. E. Al Rivan, "Penggunaan Global Contrast Saliency dan Histogram of Oriented Gradient Sebagai Fitur untuk Klasifikasi Jenis Hewan Mamalia," *Petir*, vol. 13, no. 1, pp. 80–85, 2020, doi: 10.33322/petir.v13i1.908.
- [15] M.-M. Cheng, N. J. Mitra, X. Huang, P. H. S. Torr, and S.-M. Hu, "Global contrast based salient region detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 3, pp. 569–582, 2015, doi: 10.1109/TPAMI.2014.2345401.
- [16] X. Hou and L. Zhang, "Saliency Detection: A Spectral Residual Approach," *IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2007, doi: <https://doi.org/10.1109/CVPR.2007.383267>.
- [17] Y. Yohannes, S. Devella, and K. Arianto, "Deteksi Penyakit Malaria Menggunakan Convolutional Neural Network Berbasis Saliency," *JUITA J. Inform.*, vol. 8, no. 1, p. 37, 2020, doi: 10.30595/juita.v8i1.6671.
- [18] H. Mureşan and M. Oltean, "Fruit recognition from images using deep learning," *Acta Univ. Sapientiae, Inform.*, vol. 10, no. 1, pp. 26–42, 2018, doi: 10.2478/ausi-2018-0002.
- [19] M. Oltean, "Fruits 360 dataset: A dataset of images containing fruits and vegetables," *Kaggle*, 2019. [Online]. Available: <https://www.kaggle.com/moltean/fruits/version/57>. [Accessed: 07-Jul-2019].
- [20] R. Achanta, S. Hemami, F. Estrada, and S. Susstrunk, "Frequency-tuned Salient Region Detection," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 1597–1604, 2010, doi: <https://doi.org/10.1109/cvpr.2009.5206596>.
- [21] B. Ustubioglu, V. Nabiyev, G. Ulutas, and M. Ulutas, "Image Forgery Detection Using Colour Moments," *2015 38th Int. Conf. Telecommun. Signal Process. TSP 2015*, pp. 540–544, 2015, doi: 10.1109/TSP.2015.7296321.
- [22] P. Włodarczak, *Machine Learning and its Applications*. University of Southern Queensland, Toowoomba, Queensland, Australia: CRC Press, 2020.
- [23] J. Dean, *Big Data, Data Mining, and Machine Learning*. Wiley, 2014.