

# SISTEM TEMU KEMBALI CITRA DAUN MENGGUNAKAN METODE REDUCED MULTI SCALE ARCH HEIGHT (R-MARCH) PADA SMARTPHONE

Mirza Galih Kurniawan<sup>1)</sup>, Nanik Suciati<sup>2)</sup>, Chastine Fatichah<sup>3)</sup>  
<sup>1,2,3)</sup> Jurusan Informatika, Fakultas Teknologi Informasi, Kampus ITS Keputih, Sukolilo, Surabaya  
e-mail: mirza\_mail\_box@yahoo.co.id<sup>1)</sup>, nanik@if.its.ac.id<sup>2)</sup>, chastine@if.its.ac.id<sup>3)</sup>

## ABSTRAK

Metode yang digunakan dalam sistem temu kembali citra daun harus efisien agar dapat berjalan baik pada smartphone yang memiliki sumber daya terbatas. Salah satu metode sistem temu kembali citra daun yang cukup efisien dan dapat diterapkan pada smartphone adalah Multiscale Arch Height (MARCH). MARCH menggunakan karakteristik tinggi lengkungan pada kontur daun sebagai fitur untuk proses temu kembali citra daun. Pada metode MARCH sampel titik pada kontur yang digunakan cukup banyak sehingga masih dimungkinkan mengurangi komputasi metode MARCH dengan cara mereduksi titik acuan Arch Height yang digunakan. Pada penelitian ini diusulkan metode sistem temu kembali citra daun pada smartphone menggunakan metode reduced multi scale arch height (R-MARCH) yang lebih efisien dibanding metode MARCH. Dari percobaan yang sudah dilakukan, didapatkan waktu komputasi yang dibutuhkan metode MARCH adalah 864 milidetik sedangkan R-MARCH 632 milidetik.

**Kata Kunci:** CBIR, Citra Daun, MARCH, Reduced MARCH

## ABSTRACT

In order to be able to run well on smartphones that have limited resources, the method used in leaves image retrieval should be efficient. One of leaves image retrieval method that is efficient enough to be applied on a smartphone is Multiscale Arch Height (MARCH). MARCH using collection of high curvature from contour as feature. To get this feature MARCH using a lot of reference point from the leaf contour. There is a possibility to decrease MARCH complexity by reducing the reference point to be used. This research propose the method of leaves image retrieval on a smartphone using reduce multi-scale arch height (R-MARCH) that more efficient than MARCH. From the experiments that have been conducted, it was found that the computing time needed by MARCH method is 864 milliseconds while the R-MARCH is 632 milliseconds.

**Keywords:** CBIR, Leaves Image, MARCH, Reduced MARCH.

## I. PENDAHULUAN

Sistem temu kembali citra memiliki peran penting dalam lingkup botani. Kurang lebih terdapat 400.000 spesies tumbuhan yang telah dikenali di mana 270.000 di antaranya sudah diidentifikasi dan diberi nama [8]. Dengan besarnya jumlah spesies yang ada maka pengenalan spesies tumbuhan secara manual akan sulit dilakukan. Sistem temu kembali citra dapat membantu proses pengenalan spesies tumbuhan dengan memberikan informasi terkait spesies tumbuhan yang relevan dan telah dikenali sebelumnya dalam *database* berdasarkan citra kueri dari spesies yang dicari.

Dasar pengenalan spesies tumbuhan adalah kesamaan ciri biometrik yang dapat dibandingkan antara satu spesies tumbuhan dengan spesies tumbuhan lainnya. Dalam ilmu taksonomi, klasifikasi tumbuhan umumnya dilakukan berdasarkan daun dan bunga. Proses pengenalan tumbuhan secara otomatis banyak menggunakan daun sebagai ciri biometrik yang dipakai. Hal ini dikarenakan daun cenderung memiliki bentuk 2 dimensi yang akan lebih mudah diolah dan akan memerlukan komputasi relatif lebih rendah dibanding untuk pengolahan objek tiga dimensi [9].

Alat komputasi yang ideal digunakan untuk penerapan aplikasi sistem temu kembali citra daun adalah *smartphone*. Hampir semua *smartphone* sudah dilengkapi dengan kamera sehingga dapat dijadikan alat komputasi yang efektif untuk sistem temu kembali citra daun. Berbeda dengan komputer pada umumnya, *smartphone* memiliki sumber daya yang terbatas. Oleh karena itu, tidak setiap metode identifikasi tumbuhan yang unggul diterapkan pada komputer dapat berjalan dengan baik pula jika diterapkan pada *smartphone*. Metode yang diterapkan untuk identifikasi tumbuhan pada smart phone harus memperhatikan batasan sumber daya yang ada antara lain jumlah RAM, kemampuan komputasi dan daya. Untuk itu diperlukan metode yang efisien dan efektif.

Beberapa metode temu kembali citra daun berdasarkan bentuk daun telah dikembangkan, di antaranya adalah penelitian yang dilakukan oleh [1],[2],[3],[9]. Meskipun sudah banyak penelitian yang dilakukan, namun masih sedikit metode untuk penerapan pada *smartphone*.

Penelitian sebelumnya dilakukan oleh [9] menggunakan metode *Multi Scale Arch Height* (MARCH) di mana fitur bentuk daun direpresentasikan dengan tinggi lengkungan pada tiap titik pada kontur daun. Metode ini cukup baik diimplementasikan pada *smartphone* karena hanya menghasilkan vektor fitur yang sedikit (kurang dari 150 fitur) dan waktu komputasi yang rendah.

Walaupun metode MARCH cukup baik diimplementasikan pada *smartphone*, namun metode MARCH menggunakan cukup banyak titik acuan pada kontur sebagai masukan *descriptor*. Komputasi yang diperlukan metode MARCH dimungkinkan dapat dikurangi dengan mengurangi titik acuan pada kontur yang digunakan namun tanpa menghilangkan informasi bentuk kontur.

Pada penelitian ini diusulkan metode sistem temu kembali citra daun pada *smartphone* menggunakan metode *Reduced Multi Scale Arch Height* (R-MARCH) yang lebih efisien dibanding metode MARCH. Pada metode R-MARCH dilakukan reduksi terhadap titik acuan yang digunakan pada metode MARCH dengan hanya menggunakan titik *convex hull* pada kontur sehingga dapat diperoleh metode identifikasi tumbuhan yang membutuhkan komputasi lebih sedikit dibanding metode MARCH dan akan berjalan lebih efisien pada *smartphone*.

## II. TINJAUAN PUSTAKA

### 1. *Convex hull* pada Kumpulan Koordinat

*Convex hull* digambarkan secara sederhana dalam sebuah bidang sebagai pencarian *subset* dari himpunan titik pada bidang tersebut, sehingga jika titik-titik tersebut dijadikan poligon maka akan membentuk poligon yang konveks. Suatu poligon dikatakan konveks jika garis yang menghubungkan antar kedua titik dalam poligon tersebut tidak memotong garis batas dari poligon. Pencarian *convex hull* dari sebuah himpunan titik  $Q$  ( $CH(Q)$ ) adalah mencari sebuah *convex set* terkecil yang memuat seluruh titik pada  $Q$ . *Convex hull* dari dari sebuah himpunan titik  $Q$  ( $CH(Q)$ ) pada  $n$  dimensi adalah seluruh irisan dari semua *convex set* yang mengandung  $Q$ . Terlebih lanjut, untuk  $N$  buah titik  $p_1, p_2, \dots, p_N$ . *Convex hull* merupakan himpunan *convex combination* yang dinyatakan dengan Persamaan 1.

$$CH(Q) = \sum_{j=1}^N \lambda_j p_j; \lambda \geq 0; \sum_{j=1}^N \lambda_j = 1 \quad (1)$$

Metode *convex hull* yang digunakan pada penelitian ini adalah metode yang diperkenalkan oleh [6]

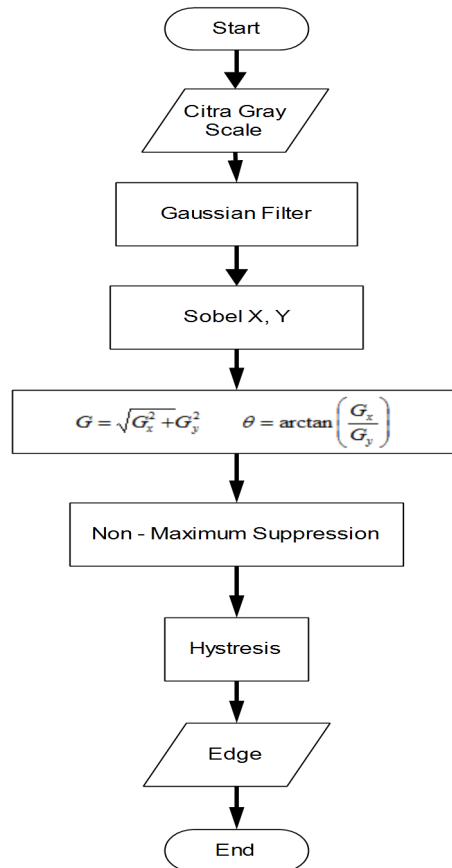
### 2. Ekstraksi Kontur Menggunakan Metode Canny

Kontur adalah garis batas dari bentuk geometris dalam gambar digital [4]. Karena identifikasi kontur sangat penting untuk menganalisis isi gambar, ekstraksi kontur menjadi salah satu bagian penting dalam masalah *computer vision* dan pengenalan pola. Pada penelitian ini digunakan metode Canny untuk proses Ekstraksi Kontur.

Deteksi tepi Canny mampu mendeteksi tepian dengan tingkat kesalahan minimum [5]. Berbeda dengan operator lainnya, deteksi tepi Canny menggunakan *Gaussian Derivative Kernel* untuk memperhalus tampilan sebuah citra. Keunggulan Canny di banding dengan deteksi tepi lainnya

- Good detection*, memaksimalkan *signal to noise ratio* (SNR) agar semua tepi dapat terdeteksi dengan baik.
- Good location*, untuk meminimalkan jarak deteksi tepi yang sebenarnya dengan tepi yang dihasilkan melalui pemrosesan, sehingga lokasi tepi terdeteksi menyerupai tepi secara nyata. Semakin besar nilai *Loc*, maka semakin besar kualitas deteksi yang dimiliki.
- One respon to single edge*, untuk menghasilkan tepi tunggal/tidak memberikan tepi yang bukan tepi.

Alur proses metode Canny terdapat pada Gambar 1. Hasil dari ekstraksi kontur adalah vektor koordinat titik yang berada pada garis tepi daun.



Gambar 1. Alur Proses Metode Canny

Penjelasan tiap tahapan pada alur proses metode Canny adalah sebagai berikut:

- Gaussian filter/smoothing* dilakukan untuk menghilangkan *noise*. Salah satu *filter Gaussian* dengan menggunakan  $\sigma = 1, 4$  yang didapat dari Persamaan *Gaussian*

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

adalah:

$$\frac{1}{159} \begin{bmatrix} 2 & 4 & 5 & 4 & 2 \\ 4 & 9 & 12 & 9 & 4 \\ 5 & 12 & 15 & 12 & 5 \\ 4 & 9 & 12 & 9 & 4 \\ 2 & 4 & 5 & 4 & 2 \end{bmatrix}$$

Yang selanjutnya *filter* tersebut digunakan untuk mendapatkan nilai baru dari titik pada citra dari nilai tetangga pada piksel tersebut.

- Setelah dilakukan operasi *gaussian filter*, operasi sobel dilakukan untuk mendapatkan gradien x dan y. operator sobel yang digunakan adalah:

$$G_x = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ +1 & +2 & +2 \end{bmatrix} * A \text{ dan } G_y = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} * A$$

- Perhitungan gradien dan arah gradien dilakukan setelah nilai gradien x dan y didapatkan, perhitungan gradien dan arah gradien dihitung dengan Persamaan 4.

$$G = \sqrt{G_x^2 + G_y^2}; \theta = \arctan\left(\frac{G_y}{G_x}\right)$$

(4)

- d. Informasi arah dan gradien yang didapat digunakan pada operasi *non-maximum suppression*. Pada operasi ini diambil nilai piksel dengan gradien yang besar.
- e. Tahapan terakhir pada Canny adalah *thresholding* untuk menampilkan tepi dari citra, di mana tepi adalah piksel yang memiliki gradien besar yang sudah diseleksi pada tahapan *maximum suppression*.

### 3. MARCH

Multi Scale Arch Height [7] yang disingkat dengan MARCH merupakan suatu deskriptor yang dikhususkan untuk ekstraksi fitur daun berdasarkan *multi scale arch height* (beberapa skala tinggi lengkungan pada daun). Tujuan utama dari metode ini adalah untuk mendapatkan deskriptor bentuk daun yang efisien dan efektif agar dapat diterapkan dengan baik pada perangkat *smartphone*.

## III. METODE PENELITIAN

Penentuan desain model sistem ini akan memberikan gambaran mengenai apa dan bagaimana penelitian ini dilaksanakan. Metode yang diusulkan adalah *Reduced MARCH* di mana garis besar alur proses pada metode ini terdapat pada Gambar 2. Pada Gambar 2 terdapat proses perhitungan *convex hull* yang merupakan tambahan proses pada metode R-MARCH. Tahapan proses R-MARCH adalah sebagai berikut:

#### a. Input Sistem

Input sistem berupa citra digital daun yang sudah tersimpan sebelumnya, dalam hal ini adalah *data test* dari *dataset*. Citra daun yang diambil merupakan citra daun tunggal dan harus memiliki *background* putih yang kontras dengan objek daun.

#### b. Konversi *Gray Scale*

Dalam penelitian ini nilai keabuan didapat dari *channel V* pada ruang warna HSV. *Channel V* digunakan karena *channel* ini merupakan nilai maksimum dari nilai R/G/B sehingga hampir tidak terpengaruh oleh refleksi cahaya.

#### c. Ekstraksi Kontur Daun

Ekstraksi kontur daun merupakan proses untuk mendapatkan kontur dari citra input daun yang akan digunakan pada operasi MARCH. Input pada tahapan ekstraksi kontur daun ini adalah citra *gray scale* yang telah didapatkan pada proses sebelumnya. Beberapa tahapan untuk mendapatkan kontur daun diantaranya adalah konversi *blur* atau pengaburan citra, *binary thresholding* dan penerapan metode Canny.

#### c. *Convex hull* Ruas Tulang Daun

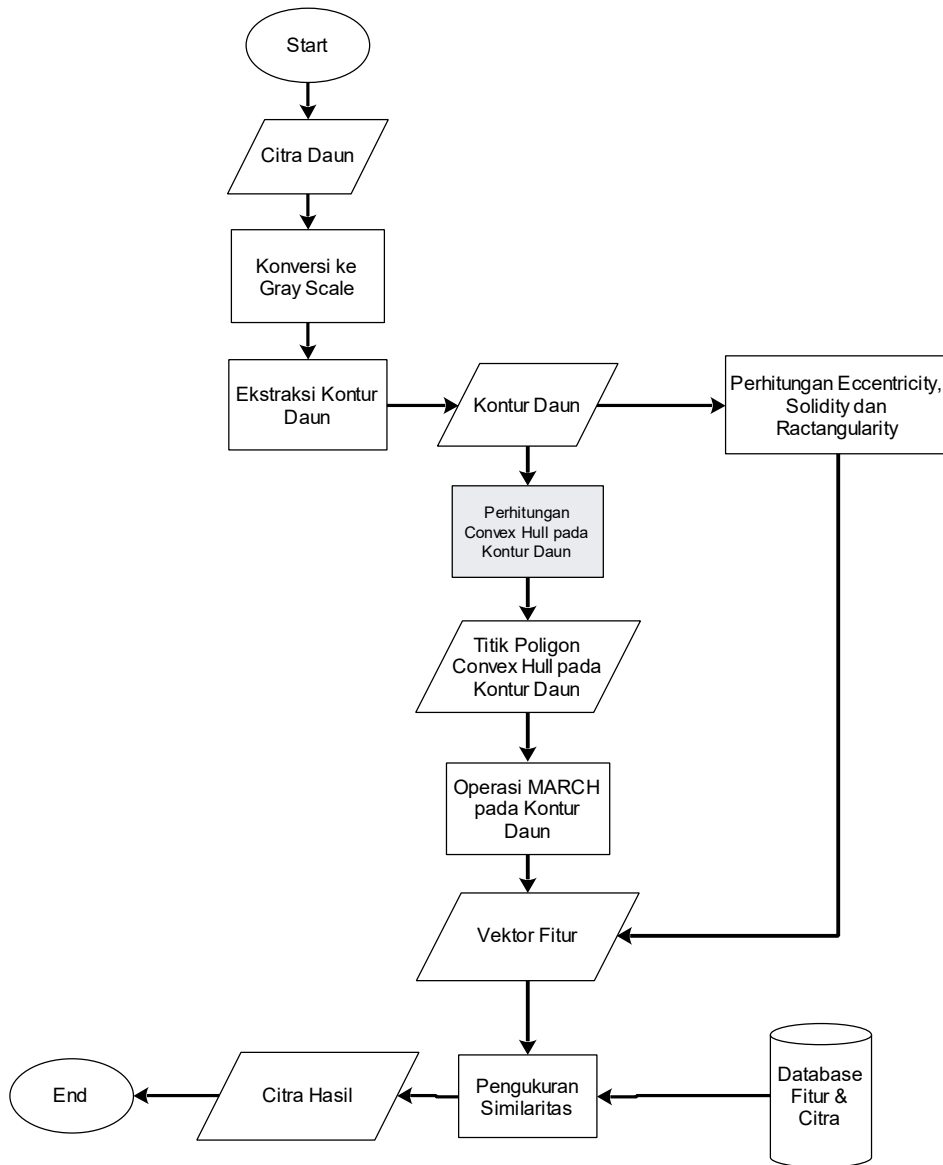
Titik terluar ruas tulang daun merupakan titik *convex hull* dari kumpulan titik yang membentuk ruas tulang daun. Untuk mendapatkan titik *convex hull* digunakan algoritma yang dikembangkan oleh [6].

#### d. *Operasi March*

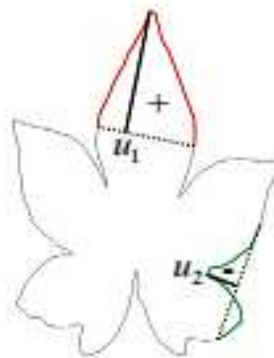
Setiap titik  $u$  pada kontur *convex hull*, lengkung dari ukuran skala  $S$  merupakan kumpulan titik pada kontur  $A_u^S = t | u - S/2 \leq t \leq u + S/2$ , di mana  $S \in (0,1)$ . Tinggi dari lengkungan  $h_u^S$  dari lengkungan  $A_u^S$  merupakan bilangan real di mana nilai absolutnya dapat dihitung dari Persamaan 5.

$$|h_u^S| = \frac{|(x(u)-x(u+\frac{S}{2})) (y(u+\frac{S}{2})-y(u-\frac{S}{2})) - (y(u)-y(u-\frac{S}{2})) (x(u+\frac{S}{2})-x(u-\frac{S}{2}))|}{\sqrt{(x(u+\frac{S}{2})-x(u-\frac{S}{2}))^2 + (y(u+\frac{S}{2})-y(u-\frac{S}{2}))^2}} \quad (5)$$

Di mana jika nilai  $h_u^S = 0$  maka titik  $A_u^S$  berada pada kontur lengkungan sedangkan nilai positif atau negatif dari tinggi lengkungan melambangkan lengkungan tersebut cekung atau cembung.



Gambar 2. Alur Proses Sistem Temu Kembali Citra Daun dengan Metode R-MARCH



Gambar 3. Ilustrasi Metode MARCH

Setiap titik  $u$  pada kontur,  $K$  merupakan sebuah integer positif. Di mana tinggi lengkungan  $h_u^s$  dari  $s = \frac{1}{2^n}$  yang merupakan  $K$ -scale dari tinggi lengkungan. Semakin tinggi nilai  $K$  maka akan semakin pendek panjang busur dari lengkungan. Pada bentuk kontur  $C(u) = (X(u), y(u)), u \in [0,1]$ , melintasi dari titik awal  $C(0) = (X(0), y(0))$  ke titik akhir  $C(1) = (X(1), y(1))$  sepanjang kontur. Panjang lengkungan pada skala- $K$   $h_u^{\frac{1}{2^k}}$  di setiap nilai  $u$  dapat diformulasikan pada fungsi  $f^k(u)$ . Di mana nilai  $K$  bergantung pada jumlah titik ( $N$ ) dalam

kontur di mana nilai maksimal dari  $K$ , dilambangkan dengan  $T = \log_2 N - 1$ . Dari fungsi  $f^k(u)$  dapat diturunkan menjadi fungsi  $\alpha^k(u)$  dan  $\beta^k(u)$  yang masing-masing fungsi tersebut dapat didefinisikan sebagai:

$$\begin{aligned} \alpha^k(u) &= |f^k(u)| \\ \beta^k(u) &= 1 \text{ jika } f^k(u) > 0; \beta^k(u) = 0 \text{ jika } f^k(u) \leq 0 \end{aligned} \tag{6}$$

Dari Persamaan 6 dapat disimpulkan bahwa nilai  $\beta^k(u)$  melambangkan cekung atau cembung lengkungan. Sedangkan nilai  $\alpha^k(u)$  adalah nilai absolut dari tinggi lengkungan. Karena menggunakan nilai tinggi, maka metode MARCH dapat dikatakan *invariance* pada translasi. Agar *invariance* terhadap penskalaan, maka nilai tinggi yang didapatkan dapat dibagi dengan nilai tinggi maksimalnya.

Untuk mendapatkan fitur yang *invariance* terhadap rotasi, tiap nilai dari tiap fungsi yang didapat diaplikasikan pada transformasi *fourier* dan didapatkan nilai besaran dari *discrete fourier transform* sebagai berikut:

$$\begin{aligned} \tau^K(v) &= \left| \frac{1}{N} \sum_{i=0}^{N-1} \alpha^k(u_i) \exp\left(\frac{-j2\pi vi}{N}\right) \right| \text{ dimana } v = 0,1,2,3 \dots N - 1 \\ \text{Dan} \\ \psi^K(v) &= \left| \frac{1}{N} \sum_{i=0}^{N-1} \beta^k(u_i) \exp\left(\frac{-j2\pi vi}{N}\right) \right| \text{ dimana } v = 0,1,2,3 \dots N - 1 \end{aligned} \tag{7}$$

Nilai  $N$  merupakan jumlah titik sample pada kontur Sehingga deskriptor akhir yang didapat dapat didefinisikan seperti yang terdapat pada Persamaan 8.

$$MARCH = \{\tau^K(v). \psi^K(v) | K = 1,2 \dots \log_2 N - 1; v = 0,1,2 \dots M - 1\} \tag{8}$$

e *Eccentricity, Solidity dan Rectangularity*

*Eccentricity* merupakan perbandingan/ratio dari *major axis* terhadap *minor axis* dalam sebuah batas berbentuk *elips*. *Solidity* merupakan ukuran sebuah bentuk apakah cembung atau cekung. Nilai *Solidity* adalah perbandingan antara luas bentuk benda terhadap luas area *convex hull* dari bentuk tersebut. *Rectangularity* merupakan gambaran seberapa kotak bentuk dari suatu benda. Nilai *rectangularity* merupakan perbandingan antara luas area dengan luas minimum suatu kotak yang dapat melingkupi bentuk tersebut. *Eccentricity, Solidity dan Rectangularity* merupakan fitur tambahan dalam metode MARCH.

f Pengukuran jarak vektor fitur pada MARCH

Jika ada dua vektor fitur MARCH, misal vektor A dan vektor B didefinisikan sebagai:

$$\begin{aligned} MARCH_A &= \{\tau_A^K(v). \psi_A^K(v) | K = 1,2 \dots \log_2 N - 1; v = 0,1,2 \dots M - 1\} \\ MARCH_B &= \{\tau_B^K(v). \psi_B^K(v) | K = 1,2 \dots \log_2 N - 1; v = 0,1,2 \dots M - 1\} \end{aligned}$$

Maka jarak antara A dan B adalah:

$$D(A, B) = \sum_{K=1}^{\log_2 N - 1} \left( \sum_{v=0}^{M-1} |\tau_A^K(v) - \tau_B^K(v)| + \lambda |\psi_A^K(v) - \psi_B^K(v)| \right) \tag{9}$$

Di mana  $\lambda$  merupakan parameter bobot. Fitur tambahan yang digunakan pada metode MARCH adalah *Eccentricity, Solidity dan Rectangularity* sehingga persamaan pengukuran jarak vektor fitur adalah seperti yang terdapat pada Persamaan 10.

$$D_g = D(A, B) + \mu (|E_a - E_b| + |S_a - S_b| + |R_a - R_b|) \tag{10}$$

*Ea: Eccentricity citra daun A*

*Eb: Eccentricity citra daun B*

*Sa*: Solidity citra daun A  
*Sb*: Solidity citra daun B  
*Ra*: Rectangularity citra daun A  
*Rb*: Rectangularity citra daun B

#### IV. HASIL UJI COBA DAN PEMBAHASAN

*Dataset* yang digunakan adalah *dataset* Flavia yang terdiri dari 1907 gambar yang terdiri dari 32 spesies daun, tiap spesies terdiri 50 hingga 77 gambar. Uji coba pada penelitian ini terdapat dua skenario. Skenario pertama untuk mengukur akurasi dengan MAP (*Mean Average Precission*) sedangkan skenario kedua untuk mengukur waktu komputasi metode *Reduced MARCH*.

Skenario pengujian pertama *dataset* dibagi menjadi dua bagian, tiap spesies diambil 40 data untuk *training* dan sisanya untuk data *testing*. Pada cara ini dilakukan pengujian untuk *hasil uji* sebanyak 10 peringkat teratas. Skenario pengujian pertama dilakukan pada aplikasi *desktop*.

Skenario uji coba kedua adalah pengukuran waktu komputasi pada *smartphone*, pengukuran waktu komputasi dilakukan dengan mengambil nilai rata-rata waktu komputasi *query* dari tiap spesies. Total waktu komputasi adalah waktu ekstraksi (pada *smartphone*) ditambah dengan waktu pengukuran similarias vektor fitur (pada *server*). Pengujian pada *smartphone* dilakukan pada emulator *Oracle Virtual Box* untuk menjalankan sistem operasi Android 4.0 (*ice cream sandwich*) dengan memori sebesar 512 MB.

Parameter evaluasi hasil yang digunakan adalah nilai *Average Precission* (AP). AP digunakan karena nilai AP memperhitungkan peringkat hasil *query* sehingga semakin relevan peringkat teratas dokumen *terretrive* maka nilai AP semakin tinggi. Selain nilai *Average Precission*, juga dianalisa performa metode dengan mengukur waktu komputasi yang dibutuhkan pada *smartphone*.

Nilai *Average precision* adalah suatu ukuran evaluasi yang diperoleh dengan menghitung rata-rata tingkat *precision* pada berbagai tingkat *recall*. Nilai *Average precission* berada pada rentang antara nol sampai dengan satu. Oleh karena itu, suatu sistem temu kembali yang baik adalah yang dapat memberikan nilai *Average precision* mendekati atau sama dengan satu. Nilai *Average precision* dirumuskan dalam Persamaan 11.

$$AP(Q) = \frac{\sum_{k=1}^M (P(k) \times f(k))}{N} \quad (11)$$

Di mana  $P(k)$  merupakan *precision* dari peringkat ke  $-k$  daftar dokumen *terretrive*,  $f(k)$  bernilai 1 jika dokumen ke  $-k$  tersebut valid dan bernilai 0 jika tidak.

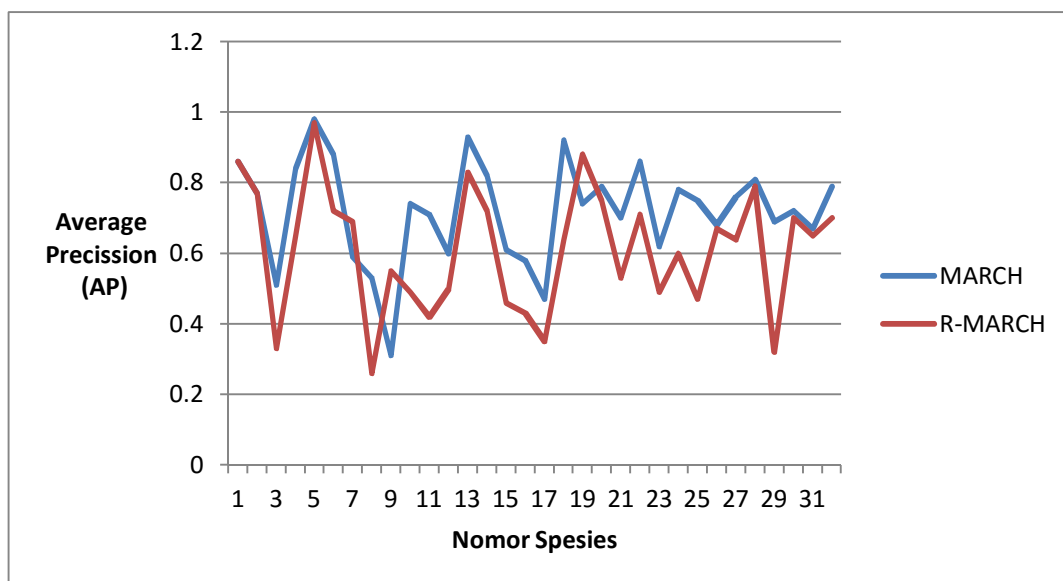
Setelah didapatkan nilai *Average precision* dari tiap pengujian, selanjutnya dihitung nilai *Mean Average Precisson* (MAP) dan rata-rata waktu komputasi untuk tiap dataset. MAP merupakan rata-rata dari seluruh nilai AP pada tiap *query* uji coba. Hasil uji coba untuk pengukuran MAP dan waktu komputasi terdapat pada Tabel 1, sedangkan untuk hasil berupa grafik terdapat pada Gambar 4 dan 5. Pada Tabel 1 terdapat 32 spesies daun yang menjadi data ujicoba serta hasil uji coba untuk mengukur efektifitas metode dengan MAP serta waktu komputasi dalam satuan milidetik (ms). Gambar 4 dan Gambar 5 merupakan bentuk grafik dari Tabel 1. Di mana sumbu x pada grafik merupakan urutan nomor spesies (sesuai pada Tabel 1) dan sumbu y pada Gambar 4 merupakan nilai *Average Precission* (AP) tiap spesies, sedangkan sumbu y pada Gambar 5. merupakan rata-rata waktu komputasi untuk pengujian tiap spesies dalam satuan milidetik.

Rata-rata jumlah fitur berupa titik acuan untuk menghitung *Arch Height* pada metode MARCH sebanyak 256 titik sedangkan pada R-MARCH kurang dari 64 titik acuan. Pada metode MARCH digunakan simple titik sedangkan pada R-MARCH digunakan titik poligon *convex hull* pada kontur daun.

Dari pengujian yang dilakukan didapatkan rata-rata waktu komputasi yang dibutuhkan metode R-MARCH dibanding MARCH adalah 632 milidetik dibanding 864 ms. Sedangkan rata – rata nilai MAP R-MARCH dibanding MARCH adalah 61%:72%.

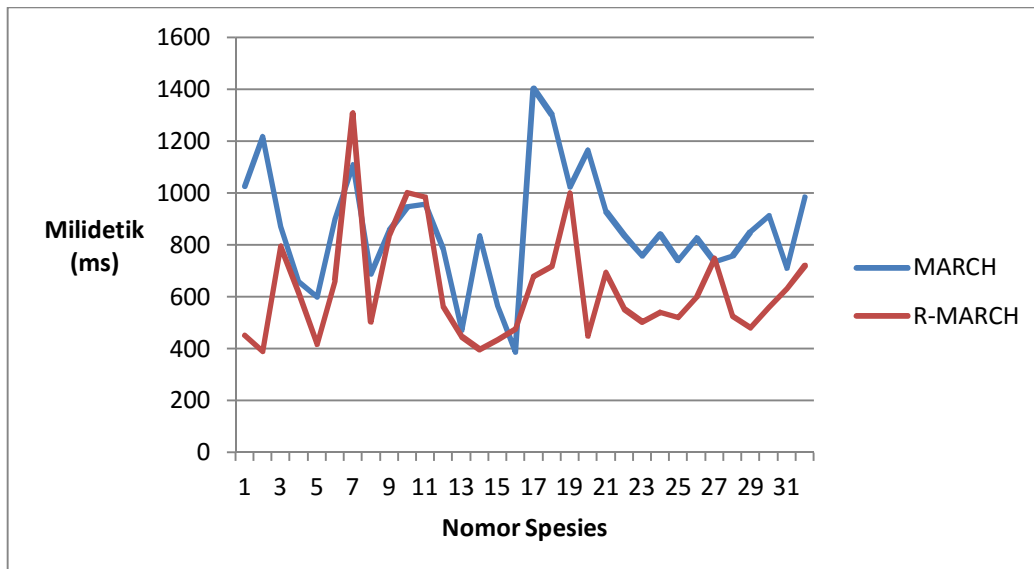
TABEL I  
HASIL Uji COBA

No	Spesies	R-MARCH (AP)	MARCH (AP)	R-MARCH (milidetik)	MARCH (milidetik)
1	Acer buergerianum Miq - trident maple	0.86	0.86	451	1028
2	Acer Palmatum - Japanese maple	0.77	0.77	391	1218
3	Aesculus chinensis - Chinese horse chestnut	0.33	0.51	796	871
4	Berberis anhweiensis Ahrendt - Anhui Barberry	0.65	0.84	618	660
5	Cedrus deodara (Roxb.) G. Don – deodar	0.97	0.98	417	600
6	Cercis chinensis - Chinese redbud	0.72	0.88	661	900
7	Chimonanthus praecox L. – wintersweet	0.69	0.59	1308	1109
8	Cinnamomum camphora (L.) J. Presl – camphortree	0.26	0.53	504	689
9	Cinnamomum japonicum Sieb. - Chinese cinnamon	0.55	0.31	839	858
10	Citrus reticulata Blanco – tangerine	0.49	0.74	1003	947
11	Ginkgo biloba L. - ginkgo, maidenhair tree	0.42	0.71	986	959
12	Ilex macrocarpa Oliv. - Big-fruited Holly	0.50	0.60	564	784
13	Indigofera tinctoria L. - Japanese maple	0.83	0.93	446	471
14	Kalopanax septemlobus (Thunb. ex A.Murr.) Koidz. - castor aralia	0.72	0.82	398	834
15	Koelreuteria paniculata Laxm. - goldenrain tree	0.46	0.61	435	566
16	Lagerstroemia indica (L.) Pers. - Crape myrtle, Crepe myrtle	0.43	0.58	478	387
17	Ligustrum lucidum Ait. f. - Glossy Privet	0.35	0.47	679	1405
18	Liriodendron chinense (Hemsl.) Sarg. - Chinese tulip tree	0.64	0.92	718	1300
19	Magnolia grandiflora L - southern magnolia	0.88	0.74	1000	1024
20	Mahonia bealei (Fortune) Carr - Beales barberry	0.75	0.79	450	1164
21	Manglietia fordiana Oliv. - Ford Woodlotus	0.53	0.70	694	931
22	Nerium oleander L. – oleander	0.71	0.86	554	837
23	Osmanthus fragrans Lour. - sweet osmanthus	0.49	0.62	503	759
24	Phoebe nanmu (Oliv.) Gamble – Nanmu	0.60	0.78	540	841
25	Phyllostachys edulis (Carr.) Houz. - pubescent bamboo	0.47	0.75	522	741
26	Pittosporum tobira (Thunb.) Ait. f. - Japanese cheesewood	0.67	0.68	601	828
27	Podocarpus macrophyllus (Thunb.) Sweet - yew plum pine	0.64	0.76	749	736
28	Populus canadensis Moench - Canadian poplar	0.79	0.81	527	758
29	Prunus persica (L.) Batsch – peach	0.32	0.69	482	853
30	Prunus serrulata Lindl. var. lannesiana auct. - Japanese Flowering Cherry	0.70	0.72	560	914
31	Tonna sinensis M. Roem. - Chinese Toon	0.65	0.67	633	712
32	Viburnum awabuki K.Koch - Japan Arrowwood	0.70	0.79	722	984
	Rata – Rata (MAP)	<b>0.61</b>	<b>0.72</b>	<b>632</b>	<b>864</b>



Gambar 4. Grafik MAP MARCH dan R-MARCH





Gambar 5. Grafik Waktu Komputasi MARCH dan R-MARCH dalam milidetik

## V. KESIMPULAN DAN SARAN

Hasil percobaan menunjukkan bahwa metode R-MARCH dapat mengurangi rata-rata waktu komputasi yang dibutuhkan oleh metode MARCH sebesar 232 ms dengan perbandingan waktu komputasi antara MARCH dengan R-MARCH adalah 864 ms dibanding 632ms. Dengan kata lain metode R-MARCH memiliki efisiensi yang lebih tinggi dibanding MARCH sehingga dapat diterapkan lebih baik pada *smartphone*. Walaupun demikian nilai MAP metode R-MARCH lebih rendah dibanding metode MARCH. Untuk penelitian lebih lanjut diperlukan penambahan fitur untuk masukan metode R-MARCH agar dapat meningkatkan nilai MAP yang dihasilkan seperti halnya fitur ruas tulang daun atau tekstur daun. Di mana fitur tersebut diharapkan juga tidak menambah banyak kompleksitas komputasi yang dibutuhkan.

## DAFTAR PUSTAKA

- [1] Abbasi, S., Mokhtarian, F. & Kittler. 1999. *Reliable classification of chrysanthemum leaves through curvature scale space*. Lect. Notes Comput. Sci. 1252. Pages 284–295.
- [2] Du, J.X., Wang, X.F. & Zhang, G.J. 2007. *Leaf shape based plant species recognition*, Appl. Math. Comput. 185 Pages 883–89.
- [3] Lee, C.L. & Chen, S.Y. 2003. *Classification for leaf images*. in: Proc. 16th IPPR Conf. Comput. Vision Graphics Image. Process. Pages 355–362.
- [4] Pedro, J. T., Qi, X. & Jiang, M. 2009. *Computational Geometry of Contour*, CCCG 2009, Vancouver
- [5] Putra, D. 2010. *Pengolahan Citra Digital*. Andi: Yogyakarta.
- [6] Skalansky, J. 1982. *Finding the convex hull of a simple polygon*. Pattern Recognition Letters 1 79-83
- [7] Wang, B., Brown, D., Gao, Y & Salle, J. L. 2015. *MARCH: Multiscale-arch-height description for mobile retrieval of leaf images*. Journal Information Sciences volume 302. Pages 132–148.
- [8] Wang. J.W., Bai, X., You, X.G., Liu, W.Y., & Latecki, L.J. 2012. *Shape matching and classification using height functions*. Pattern Recogn. Lett. Pages 134–143.
- [9] Wang. X. F., Huang, D. S., Du, J. X., Xu, H. & Heutte, L. 2008. *Classification of plant leaf images with complicated background*. Appl. Math. Comput. 205 . Pages 916–926.