

# Sistem Identifikasi Relief pada Situs Bersejarah Menggunakan Perangkat Mobile Android (Studi Kasus Candi Borobudur)

Rajif Agung Yunmar\*<sup>1</sup>, Agus Harjoko<sup>2</sup>

<sup>1</sup>Prodi S2/S3 Ilmu Komputer dan Elektronika, FMIPA UGM, Yogyakarta

<sup>2</sup>Jurusan Ilmu Komputer dan Elektronika, FMIPA UGM, Yogyakarta

e-mail: \*<sup>1</sup>rajif.agung.y@mail.ugm.ac.id, <sup>2</sup>aharjoko@ugm.ac.id

## Abstrak

Bangsa yang besar adalah bangsa yang menghargai sejarah dan asal usulnya. Relief yang terdapat pada candi Borobudur menggambarkan banyak cerita, termasuk sejarah dan asal usul bangsa ini. Mulai dari cerita kehidupan kalangan kerajaan, kehidupan masyarakat, dan adat istiadat pada saat bangunan candi tersebut dibuat, dsb.

Penelitian ini merancang sebuah perangkat lunak mobile Android untuk mengidentifikasi citra relief Candi Borobudur sehingga dapat membantu wisatawan dalam menerjemahkan pesan dan informasi yang terkandung didalamnya. Metode ekstraksi ciri yang digunakan adalah Speeded-Up Robust Feature (SURF) dan hierarchical k-means tree nearest-neighbor untuk identifikasinya.

Pengujian identifikasi citra relief dilakukan dengan berbagai macam variasi, yaitu sudut, jarak, rotasi, intensitas cahaya dan keutuhan citra masukan untuk melihat pengaruhnya terhadap hasil pengenalan citra relief tersebut. Metode identifikasi yang diajukan memberikan hasil pengenalan sebesar 93.30% dengan rata-rata waktu komputasi 59.55 detik.

**Kata kunci**— Candi Borobudur, relief, Android, SURF, hierarchical k-means tree

## Abstract

The great nation built from people who can respects they history and origins. Reliefs at Borobudur temple contained many stories, including the history and origins of this nation. Starting from the life story of the royal, society, and customs at the time of the building of the temple was made, and so on.

This study develops mobile Android software for identification of Borobudur Temple relief image object so that it can help travelers in translating the story and the information contained therein. Feature extraction method used is speeded-Up Robust Feature (SURF) and hierarchical k-means tree nearest-neighbor for identification.

Identification testing of relief images is done by different variations, ie angle, distance, rotation, intensity of the light and wholeness of image input to see the effect on the relief image recognition results. The proposed identification method gives recognition results of 93.30% and the average computation time for 59.55 seconds.

**Keywords**— Borobudur temple, relief, Android, SURF, hierarchical k-means tree

## 1. PENDAHULUAN

Bangsa yang besar adalah bangsa yang menghargai sejarah dan asal usulnya. Mengunjungi situs sejarah adalah salah satu cara yang dapat dilakukan untuk menggali sejarah bangsa ini. Pada beberapa situs wisata candi yang tersebar di Daerah Istimewa Yogyakarta dan Jawa

Tengah, seringkali terdapat relief yang menggambarkan sebuah cerita. Mulai dari cerita pewayangan, keagamaan, kehidupan kalangan kerajaan, kehidupan masyarakat, adat istiadat bahkan sampai pada kekuatan militer kerajaan pada saat bangunan candi tersebut dibuat, dan lain sebagainya [1,2].

Tidak setiap wisatawan mengetahui apa pesan yang disampaikan dari sebuah relief. Untuk itu, seorang wisatawan perlu ditemani oleh seorang pemandu wisata. Pemandu wisata adalah orang pertama yang diajak bicara oleh para wisatawan dan seringkali dianggap sebagai wakil atau representasi dari suatu tempat [3]. Namun dengan beberapa alasan, terkadang pemandu wisata tidak dapat setiap saat menemani wisatawan untuk menerjemahkan pesan yang disampaikan dari sebuah relief. Menurut Purwaningsih [4] salah satu penyebabnya adalah terus meningkatnya jumlah wisatawan yang tidak diimbangi oleh jumlah pemandu wisata.

Penelitian ini mengembangkan sebuah perangkat lunak untuk identifikasi objek relief pada suatu situs wisata, sehingga dapat membantu wisatawan dalam menerjemahkan pesan dan informasi yang terkandung didalamnya. Pengenalan sebuah objek, dalam hal ini adalah objek relief Candi Borobudur menggunakan perangkat *mobile* memiliki permasalahan tersendiri, yaitu setiap wisatawan dapat mengambil citra relief dari posisi yang fleksibel, kapan saja, beragam kondisi cuaca dan intensitas cahaya, dan seringkali bidang relief tidak diambil secara utuh, dsb. Pengembang perangkat lunak dalam hal ini tidak dapat mencegah atau mengatur seorang wisatawan untuk mengambil citra relief dari posisi yang ditentukannya sendiri secara sepihak.

Penelitian ini merancang perangkat lunak berbasis Android yang dapat membantu wisatawan untuk menerjemahkan cerita dari sebuah relief. Metode ekstraksi ciri yang digunakan adalah *Speeded-Up Robust Feature* (SURF) dengan metode identifikasi *hierarchical k-means nearest-neighbor*. Disamping itu, juga akan dilakukan pengujian terhadap faktor-faktor yang mempengaruhi pengenalan diantaranya adalah skala, rotasi, masukan bidang relief yang tidak utuh, dan lain sebagainya.

## 2. METODE PENELITIAN

### 2.1 Analisa Kebutuhan

#### 2.1.1 Pengumpulan Data

Keseluruhan, Candi Borobudur memiliki 1460 relief. Untuk data pelatihan, diambil 50 *sample* data citra, yang terdiri dari 22 relief bentuk persegi panjang dan 18 relief bentuk bujur sangkar. Masing-masing *sample* data citra pelatihan tersebut diambil secara utuh dalam format JPEG. Untuk relief dengan bentuk persegi panjang citra objek pelatihan diambil dengan resolusi 800x300. Sedangkan untuk relief dengan bentuk bujur sangkar diambil dalam resolusi 350x350.

#### 2.1.2 Masukan & Keluaran Sistem

Masukan sistem ini berupa citra dalam format JPEG dengan resolusi 400x240. Sebagai keluaran, sistem akan menampilkan hasil pengenalan relief kepada user melalui perangkat *mobile* Android dalam bentuk informasi atau cerita.

#### 2.1.3 Perangkat Keras

Perangkat keras yang digunakan dalam penelitian ini terdiri dari sebuah komputer server yang digunakan dalam proses training data, sebuah ponsel berbasis Android yang digunakan untuk melakukan pengujian citra inputan dan sebuah alat ukur intensitas cahaya. Spesifikasi komputer server yang digunakan dalam penelitian ini adalah sebagai berikut:

- Prosesor AMD Quad Core 2378 CPU @ 2.4 GHz
- Sistem operasi CentOS GNU/Linux 5.9 32bit
- Memori internal 512 MB
- Harddisk kapasitas 50 GB

Sedangkan spesifikasi perangkat *mobile* berbasis sistem operasi Android yang

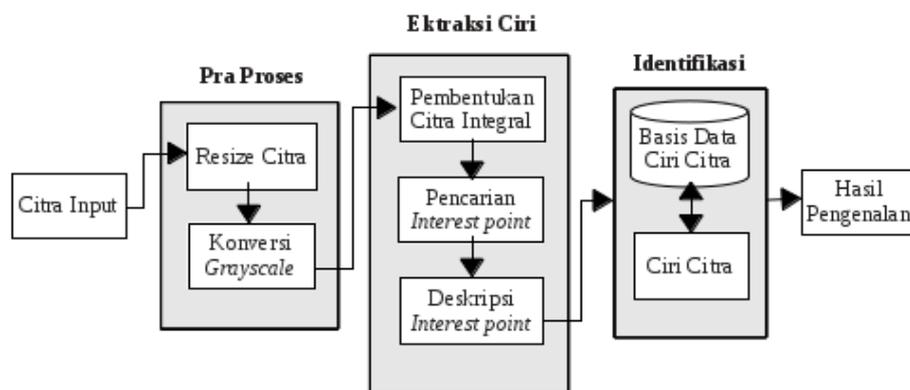
digunakan adalah sebagai berikut:

- Prosesor Cortex Quad Core 1.4 GHz
- Sistem operasi Android Ice Cream Sandwich 4.0.4
- Memori internal 1 GB
- Kamera 8 MP dengan resolusi 3232x2424 piksel

Selain itu juga digunakan alat untuk pengukuran intensitas cahaya. Penggunaan alat ini ditujukan untuk mengukur intensitas cahaya ada saat pengujian sistem yang dirancang. Spesifikasi dari perangkat keras pengukur cahaya adalah Lutron LX-100.

## 2.2 Perancangan Sistem

Proses identifikasi ini terbagi atas tiga bagian utama, yaitu: pra proses, ekstraksi ciri dan proses identifikasi. Secara umum urutan proses dari sistem identifikasi yang akan dibuat dapat kita lihat pada Gambar 1.



Gambar 1 Urutan proses sistem pengenalan objek relief

Peneliti mengusulkan pra proses yang terdiri dari pengaturan ulang ukuran citra (*resize*) dan konversi citra hasil akuisisi dari format RGB ke *grayscale*. Selain itu peneliti juga mengusulkan metode identifikasi dengan metode *hierarchical k-means tree nearest-neighbor*. Metode ini melakukan pengindeksan terhadap vektor ciri sebagai langkah yang dibuat untuk mempercepat proses identifikasi. Hasil pengindeksan adalah sebuah tree dengan tiga buah level. Masing-masing level mengandung sebanyak k-kluster yang dibentuk dengan algoritma *k-means clustering*. Pada proses ekstraksi ciri digunakan algoritma yang diusulkan oleh Bay, dkk [5].

### 2.2.1 Pra Proses

Pada tahapan ini citra relief masukan akan di proses sedemikian rupa sebagai persiapannya untuk memasuki tahapan ekstraksi ciri. Hal ini dimaksudkan agar kompleksitas komputasi pada tahapan ekstraksi ciri dapat lebih ditekan.

Pada tahapan ini, citra hasil akuisisi perangkat *mobile* Android pertama kali akan dikenai operasi pengaturan ulang skala (*resize*) ke ukuran yang lebih kecil. Menurut Chu [6], pengaturan ulang skala ini ditujukan untuk untuk menjaga waktu komputasi yang tetap rendah namun tetap nilai ciri yang tetap baik. Kemudian citra akan diubah dalam mode *grayscale*. Dalam pengubahan citra RGB kedalam bentuk *grayscale*, digunakan persamaan yang merupakan standar dari PAL dan NTSC berikut [7] :

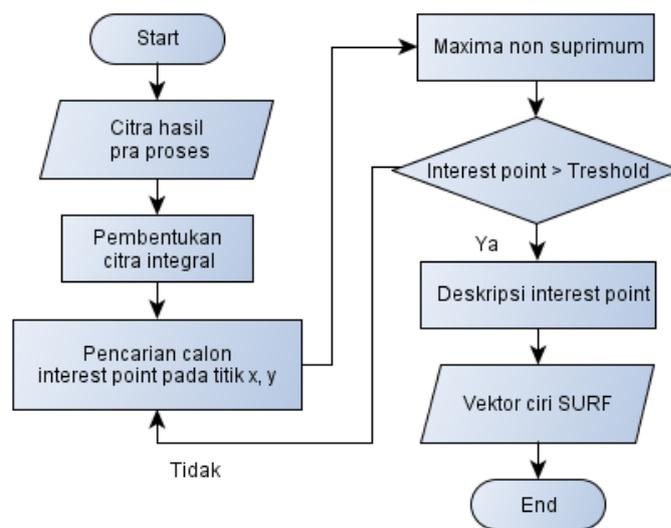
$$Y' = 0.299R + 0.587G + 0.114B \quad (1)$$

Citra yang telah dikenai tahapan pra proses tersebut selanjutnya akan dikirimkan ke server untuk dilakukan operasi ekstraksi ciri dan pengenalan objek.

### 2.2.2 Ekstraksi Ciri

Pada tahapan ini citra input hasil pra proses akan diambil cirinya menggunakan metode ekstraksi fitur SURF (*Speeded-Up Robust Feature*). Metode ini dipilih karena ketahanannya terhadap berbagai variasi input citra (*invariant*) seperti sudut, jarak, rotasi, intensitas cahaya sampai dengan kondisi citra input yang tidak utuh.

Selanjutnya, citra input akan diubah ke dalam bentuk citra integral untuk mengurangi kompleksitas komputasi [8]. Langkah selanjutnya adalah mencari calon ciri pada citra menggunakan matriks Fast Hessian yang terkenal akan kehandalan performa dan akurasinya [8]. Agar nilai ciri yang dihasilkan memiliki ketahanan terhadap skala, maka pencarian terhadap ciri dilakukan pada *scale space*. *Space scale* dapat dibentuk dengan melakukan iterasi konvolusi terhadap citra masukan dengan tapis kernel Gaussian diskret dan berulang kali melakukan sub-sampling yang akhirnya mereduksi ukuran citra. Pada akhirnya, piramida citra terbentuk karena dari setiap perulangan yang terjadi, ukuran citra pada *scale space* akan terus berkurang [9].



Gambar 2 Tahapan ekstraksi ciri dengan metode SURF

Pencarian ciri (*interest point*) pada *scale space* tersebut nantinya ditentukan berdasarkan dari nilai ekstremum yang merupakan hasil perbandingan nilai antar titik piksel tetangga lintas *scale*. Sebuah titik ditentukan sebagai *interest point*, apabila nilai perhitungannya lebih besar dari pada treshold. Langkah terakhir dari tahapan ekstraksi ciri dengan metode SURF adalah melakukan deskripsi terhadap ciri atau *interest point* yang didapatkan. Pendeskripsian ini bertujuan agar *interest point* lebih tahan terhadap rotasi, sudut pengambilan citra dan intensitas cahaya. Hasil yang didapatkan dari tahapan ekstraksi ciri ini adalah vektor ciri yang mendeskripsikan citra itu sendiri. Ilustrasi pencarian ciri *interest point* dapat dilihat pada Gambar 2.

#### 4.2.3 Pelatihan Vektor Ciri

Tujuan dari tahapan pelatihan adalah melakukan konversi vektor ciri yang ada pada citra sample menjadi sebuah *tree*. Pembentukan *tree* ini dimaksudkan agar proses identifikasi objek menjadi lebih mudah dan cepat. Metode yang digunakan dalam pelatihan ini adalah *heirarchical tree clustering*.

Output dari pelatihan adalah sebuah *tree* dengan tiga level. Level pertama terdiri dari kumpulan ciri yang dikelompokkan kedalam K kluster. Masing-masing kluster pada level pertama tersebut memiliki sub anggota pada level kedua sebanyak K kluster. Masing-masing kluster pada level kedua tersebut juga memiliki sekumpulan anggota *interest point*. Sekumpulan anggota dalam bentuk *interest point* inilah yang dijadikan sebagai *tree* level ketiga.

Pembentukan level pertama didahului dengan menentukan centroid level pertama secara

acak. Langkah kedua adalah mencari jarak antara centroid kluster yang telah terbentuk dengan setiap interest point dengan persamaan[10]:

$$D = \sqrt{\sum_{i=1}^{i \leq 64} (U_i - V_i)^2} \quad (2)$$

dimana :

- D : jarak antara centroid kluster dengan sebuah ciri  
 i : elemen deskriptor SURF ke-i  
 U<sub>i</sub> : deskriptor ke-i pada centroid  
 V<sub>i</sub> : deskriptor ke-i pada ciri yang dibandingkan

Kluster dengan jarak terdekat akan dijadikan sebagai kluster utama dari interest point tersebut. Setiap kluster yang terbentuk akan dicari nilai reratanya. Keseluruhan interest point yang ada pada setiap kluster akan dikenai proses perhitungan dengan persamaan[10]:

$$avg(i) = \frac{\sum_{x=1}^{x \leq n} f(x,i)}{n} \quad (3)$$

dimana:

- i : elemen deskriptor ke-i  
 x : ciri ke-x dari sebuah objek  
 n : jumlah ciri dalam level  
 avg(i) : rata-rata nilai untuk centroid pada elemen deskriptor ke-i  
 f(x,i) : nilai deskriptor SURF pada ciri ke-x dan elemen deskriptor ke-i

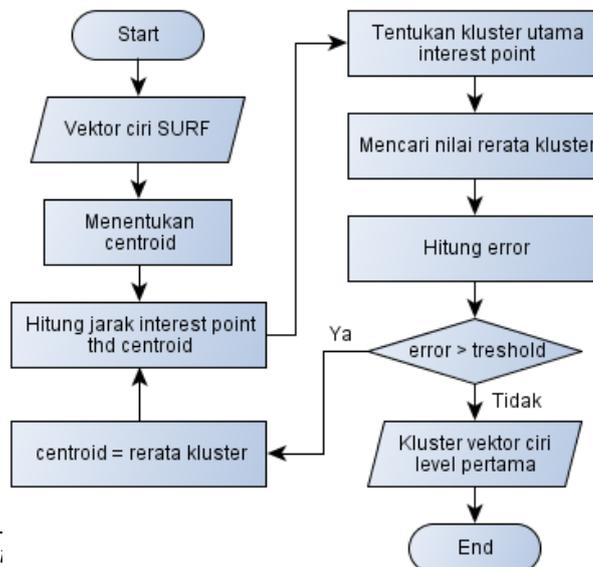
Nilai rerata kluster dibutuhkan dalam perhitungan error. Perhitungan error dilakukan dengan mencari jumlahan dari jarak antara centroid setiap kluster dengan nilai reratanya, proses ini dilakukan dengan persamaan[10]:

$$error = \sum_{i=1}^{i \leq k} D(c(i), avg(i)) \quad (5)$$

dimana:

- k : jumlah kluster yang akan terbentuk  
 c(i) : centroid kluster ke-i  
 avg(i) : nilai rerata dari centroid kluster ke-i  
 D(c(i), avg(i)) : fungsi pencarian selisih antara centroid ke-i dan nilai rerata kluster ke-i menggunakan persamaan (2)

Jika nilai error masih lebih besar dari *threshold* yang ditentukan, maka proses *training* akan kembali dilakukan mulai dari langkah kedua. Begitu seterusnya hingga nilai error telah lebih kecil dari *threshold*.

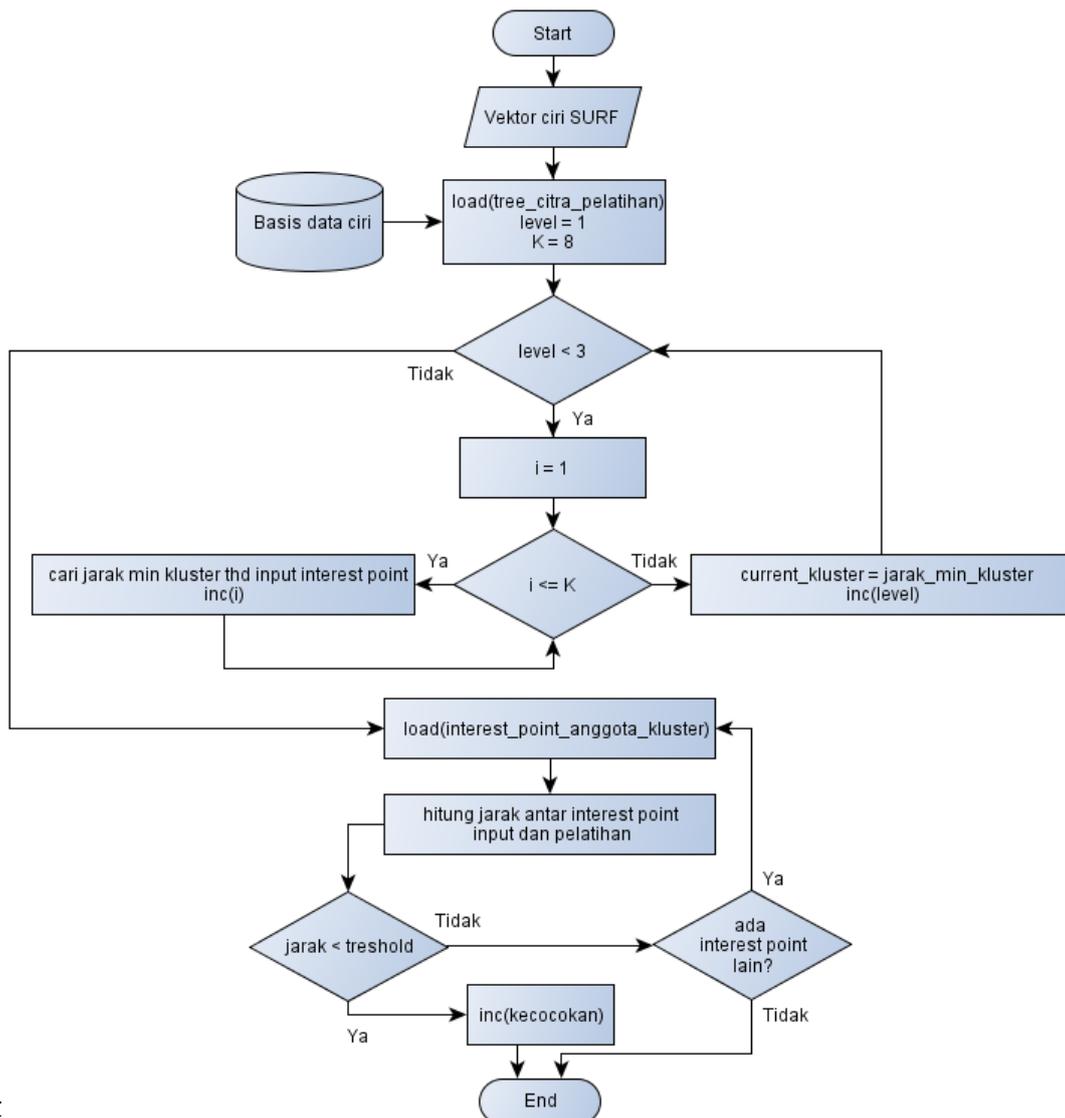


Gambar 3 Tahapan pembentukan kluster pada pelatihan

Menggunakan cara yang sama, pelatihan level kedua dilakukan terhadap setiap kluster hasil pelatihan pada level pertama. Diagram alir pada Gambar 3 menunjukkan proses pelatihan dan pembentukan *tree* level pertama dan kedua pada vektor ciri *interest point* citra *sample*. Sekumpulan ciri *interest point* yang terdapat pada level kedua akan dijadikan sebagai daun dari *tree*. Daun *tree* inilah yang dianggap sebagai *tree* level ketiga.

2.2.4 Identifikasi Citra Relief

Pada tahapan ini, vektor ciri citra input yang didapatkan dari tahapan ekstraksi ciri akan dibandingkan dengan basis data ciri yang ada pada server. Setiap anggota dari basis data pada server tersebut berbentuk sebuah *tree* yang dihasilkan dari pelatihan sekumpulan pada vektor ciri menggunakan algoritma *hierarchical k-means tree nearest-neighbor*.



Gambar 4 Tahapan identifikasi citra relief

Tahapan identifikasi citra relief dapat dilakukan dengan cara membandingkan item vektor ciri citra masukan dengan *tree* yang telah terbentuk. Perbandingan ini dilakukan dengan cara melakukan penelusuran yang dimulai dari titik *tree* level pertama yang dilanjutkan ke level selanjutnya sampai dengan ditemukannya daun pada *tree* pada level 3.

Pada *tree* level pertama, setiap *interest point* atau ciri input yang datang akan dihitung jaraknya terhadap masing-masing centroid kluster menggunakan persamaan (2). Centroid kluster yang memiliki jarak paling kecil pada level pertama tersebut akan menjadi dasar atau *parent* bagi pencarian centroid pada level ke dua.

Pada level kedua juga dilakukan hal yang sama, yaitu menghitung jarak centroid setiap kluster dengan *interest point* yang datang. Centroid kluster pada level dua yang memiliki jarak paling kecil akan menjadi dasar bagi penghitungan kecocokan *interest point* pada level ketiga.

Level ketiga adalah level terakhir pada *tree*, dimana tidak ada lagi cabang setelahnya. Level ini terdiri dari daun *tree* yang anggotanya adalah *interest point* citra hasil pelatihan. Pada level ini dilakukan perhitungan jarak antara *interest point* yang datang dengan setiap *interest point* anggota kluster. Anggota kluster yang memiliki perhitungan jarak dibawah tetapan *threshold* yang ditentukan, akan dihitung sebagai satu *interest point* yang cocok.

Ketiga langkah tersebut diatas akan dikenakan kepada seluruh objek citra pelatihan dalam basis data. Objek citra pada basis data yang memiliki kecocokan ciri *interest point* yang paling banyak, akan diajukan sebagai hasil dari pengenalan. Diagram alir pada Gambar 4 menunjukkan proses pencocokan masing-masing pada *interest point* citra masukan yang datang dengan citra latih yang terdapat pada basis data yang telah berbentuk *tree*.

### 2.2.5 Pengujian

Objek yang digunakan untuk melakukan pengujian seluruhnya berjumlah 30 buah dengan resolusi 400x240 piksel. Masing-masing objek tersebut akan dikenai berbagai pengujian, yaitu: jarak 1 meter, jarak 2 meter, sudut 45°, sudut 135°, rotasi, variasi intensitas cahaya, bidang relief terpotong 25%, bidang relief terpotong 50%, bidang relief terpotong 75%.

Untuk mengukur akurasi hasil pengenalan objek untuk masing-masing pengujian pada setiap variasi yang ada digunakan persamaan sebagai berikut:

$$akurasi(variasi) = \frac{jumlahdatayangbenar}{jumlahseluruhdatauji} \times 100\% \quad (6)$$

Secara keseluruhan, akurasi arsitektur sistem yang telah dibangun dapat dilakukan dengan persamaan sebagai berikut:

$$akurasi(total) = \frac{\sum akurasi(variasi)}{9} \quad (7)$$

## 3. HASIL DAN PEMBAHASAN

Pada bagian ini dijelaskan mengenai rangkaian uji coba dan evaluasi terhadap metode yang diajukan. Uji coba tersebut ditujukan untuk melihat sejauh mana kinerja yang dihasilkan dari proses implementasi. Evaluasi dilakukan dengan cara melakukan analisa terhadap hasil uji coba tersebut, untuk kemudian dapat diberikan kesimpulan dan saran bagi implementasi perangkat lunak selanjutnya.

### 3.1 Tahapan Pembentukan Basis Data

Proses pembentukan basis data dimulai dari ekstraksi ciri yang dikenakan pada citra

latih. Hasil dari ekstraksi ciri adalah kumpulan vektor ciri yang disimpan kedalam sebuah file dengan nama data-ciri.txt. Selanjutnya vektor ciri tersebut akan dikenai algoritma *k-means hierarchical tree clustering*.

Data pelatihan yang digunakan dalam penelitian seluruhnya berjumlah 50 citra latih. Jumlah K yang dibangun untuk identifikasi sebanyak 8 kluster dengan nilai toleransi error sebesar 0.0015. Hasil pelatihan dari masing-masing citra latih adalah sebuah array multidimensi yang disimpan dalam sebuah file dengan nama data-training.php. Waktu keseluruhan yang diperlukan dalam proses pelatihan dengan kebutuhan tersebut adalah 15 Menit.

### 3.2 Tahapan Pengujian

Kondisi yang diterapkan pada setiap pengujian adalah ketinggian kamera dari lantai dasar Candi Borobudur adalah 150-160CM. Kemudian, kecuali pada pengujian intensitas cahaya, setiap jenis pengujian diatas dilakukan saat cuaca cerah dan intensitas cahaya sedang dengan nilai 2250-4000Lux pada alat pengukuran. Pengujian dilakukan antara pukul 08.00 sampai dengan 10.30 pada bulan Juli-September 2013.

#### 3.2.1 Pengujian Jarak 2 Meter

Pengujian pada variasi ini dilakukan dengan cara melakukan akuisisi citra uji menggunakan perangkat lunak yang telah dibangun. Citra uji diambil pada jarak 2 meter dengan sudut tegak lurus terhadap bagian tengah objek relief ( $90^\circ$ ). Contoh *screenshot* pengujian sistem ditunjukkan pada Gambar 5.

#### 3.2.2 Pengujian Jarak 1 Meter

Pada pengujian ini, dilakukan akuisisi citra menggunakan perangkat lunak hasil penelitian pada jarak 1 meter dengan sudut pengambilan adalah tegak lurus ( $90^\circ$ ).

#### 3.2.3 Pengujian Sudut $45^\circ$

Pengujian ini dilakukan dengan melakukan akuisisi citra melalui perangkat lunak yang dibangun pada sudut  $45^\circ$  (sebelah kiri objek) pada jarak 2.5 meter terhadap bagian tengah objek relief.



Gambar 5 *Screenshot* pengujian jarak 2 meter

#### 3.2.4 Pengujian Sudut $135^\circ$

Dalam pengujian ini citra relief diambil dari sudut  $135^\circ$  (sebelah kanan objek) pada jarak 2.5 meter terhadap bagian tengah objek relief. Contoh *screenshot* pengujian sistem ditunjukkan pada Gambar 8, sedangkan data hasil pengujian ini dapat dilihat pada Tabel 5.

#### 3.2.5 Pengujian Rotasi

Pengujian pada variasi rotasi dilakukan dengan cara mengubah orientasi *default* dari akuisisi citra relief. Dimana orientasi *default* pada tahapan pengujian untuk objek relief persegi panjang adalah horizontal (*landscape*) dan vertikal (*portrait*) untuk relief dengan bentuk bujur sangkar. Maka pengujian variasi rotasi untuk masing-masing bentuk relief adalah kebalikannya. Citra uji diambil pada jarak 2 meter dengan sudut tegak lurus terhadap bagian tengah objek relief ( $90^\circ$ ).

### 3.2.6 Pengujian Intensitas Cahaya

Pengujian terhadap variasi intensitas cahaya dilakukan dengan cara melakukan akuisisi citra dengan intensitas cahaya yang berbeda dari citra yang ada pada citra pelatihan. Perbedaan ini dapat disebabkan oleh waktu pengambilan atau cuaca (mendung, gerimis dan hujan).

Pada penelitian ini, citra relief diambil pada waktu pagi hari pukul 06.05 sampai dengan pukul 06.30 di bulan Juli-September. Angka yang ditunjukkan oleh alat ukur intensitas cahaya Lux-100 adalah 2.250-4.000 Lux. Citra uji diambil pada jarak 2 meter dengan sudut tegak lurus terhadap bagian tengah objek relief ( $90^\circ$ ). *Screenshot* contoh pengujian sistem ditunjukkan oleh Gambar 6.6. Tabel 7 menunjukkan hasil pengujian objek relief untuk jenis variasi ini:

### 3.2.7 Pengujian Terpotong 25%

Pengujian ini dilakukan dengan cara melakukan akuisisi citra dengan hanya mengambil 75% dari keseluruhan bidang objek relief. Citra uji ini diambil dari jarak 2 meter dengan sudut tegak lurus terhadap bagian tengah objek relief ( $90^\circ$ ). Melalui pengaturan *zoom* yang terdapat pada fasilitas kamera ponsel Android, bidang relief yang akan diambil dikondisikan sedemikian rupa sehingga terpotong sebesar 25%.

### 3.2.8 Pengujian Terpotong 50%

Pengujian ini dilakukan dengan cara melakukan akuisisi citra dengan hanya mengambil 50% dari keseluruhan bidang objek relief. Citra uji ini diambil dari jarak 2 meter dengan sudut tegak lurus terhadap bagian tengah objek relief (sudut  $90^\circ$ ). Melalui pengaturan *zoom* yang terdapat pada fasilitas kamera ponsel Android, bidang relief yang akan diambil dikondisikan sedemikian rupa sehingga terpotong sebesar 50%.

### 3.2.9 Pengujian Terpotong 75%

Pengujian ini dilakukan dengan cara melakukan akuisisi citra dengan hanya mengambil 25% dari keseluruhan bidang objek relief. Citra uji ini diambil dari jarak 2 meter dengan sudut tegak lurus terhadap bagian tengah objek relief (sudut  $90^\circ$ ). Melalui pengaturan *zoom* yang terdapat pada fasilitas kamera ponsel Android, bidang relief yang akan diambil dikondisikan sedemikian rupa sehingga terpotong sebesar 75%.

## 3.3 Hasil Pengujian

Tabel 10 menunjukkan detail hasil pengujian yang dikenakan terhadap citra relief Candi Borobudur. Akurasi sistem pada setiap pengujian dihitung berdasarkan persamaan (6).

Tabel 10 Detail informasi pengujian bidang relief terpotong 75%

Pengujian	Citra Uji	Berhasil Dikenali	Dikenali sbg objek lain	Rerata kecocokan ciri	Rerata waktu pengenalan	Akurasi
Jarak 2 Meter	30	30	-	44.60	62.40s	100%
Jarak 1 Meter	30	30	-	71.33	65.88s	100%
Sudut $45^\circ$	30	25	5	29.63	60.20s	83.33%
Sudut $135^\circ$	30	28	2	34.06	57.84s	93.33%
Rotasi	30	28	2	35.33	57.05s	93.33%
Intensitas Cahaya	30	29	1	38.83	56.05s	96.66%
Terpotong 25%	30	28	2	49.10	62.43s	93.33%
Terpotong 50%	30	30	-	60.96	63.45s	100%
Terpotong 75%	30	25	5	29.80	54.66s	83.33%

## 3.4 Pembahasan Penelitian

Pengujian yang dilakukan pada akhirnya memberikan hasil yang dapat digunakan untuk melakukan pengukuran keberhasilan dari metodologi penelitian yang diajukan. Dari serangkaian

pengujian yang dilakukan terlihat bahwa pengujian jarak 2 meter memberikan pengenalan sebesar 100%, pengujian jarak 1 meter 100%, pengujian citra bidang relief terpotong 25% memberikan akurasi 93.33%, pengujian citra bidang relief terpotong 50% memberikan akurasi 100%, pengujian citra bidang relief terpotong 75% dengan akurasi 83.33%, pengujian sudut 135° dengan akurasi 93.33%, sudut 45° dengan akurasi 83.33%, pengujian rotasi dengan hasil 93.33% dan pengujian perbedaan intensitas cahaya adalah 96.66%. Secara umum, berdasarkan persamaan (7) akurasi keseluruhan sistem yang dibangun adalah:

$$\begin{aligned} \text{akurasi (total)} &= \frac{100\% + 100\% + 93.33\% + 100\% + 83.33\% + 93.33\% + 83.33\% + 93.33\% + 96.66\%}{9} \\ &= 93.30\% \end{aligned}$$

Pengujian jarak 2 meter memberikan hasil yang sempurna dikarenakan pada jarak ini bidang objek relief dapat diambil seluruhnyasehingga memaksimalkan proses identifikasi.

Nilai paling tinggi untuk rata-rata kecocokan ciri interest point ditunjukkan oleh pengujian jarak 1 meter. Hal ini menunjukkan pengenalan dari jarak yang lebih dekat memberikan hasil yang lebih baik. Karena semakin dekat jarak akuisisi akan menghasil citra dengan tekstur yang lebih detail, walaupun dengan jarak yang semakin dekat dapat dipastikan tidak keseluruhan bagian dari sebuah objek terambil keseluruhannya.

Akurasi paling rendah ditunjukkan pada pengujian variasi sudut 45° dan pengujian dengan citra terpotong 75%. Dimana kedua pengujian berhasil mengenali objek sebesar 83.33%. Penurunan akurasi pada pengujian sudut 45° diakibatkan oleh penurunan kualitas ciri karena pengambilan citra dari sudut yang berbeda dapat mempengaruhi tekstur yang dihasilkan, hal ini senada dengan pendapat Mäenpää [11] dimana sebuah tekstur dalam pengolahan citra dapat dipengaruhi oleh sudut dan perbedaan pencahayaan.

Pengujian objek relief dengan kondisi pengambilan citra relief yang tidak utuh dibawah 50% tidak terlalu mempengaruhi hasil pengenalan. Namun untuk citra objek relief dengan bagian terpotong lebih dari nilai tersebut dapat mengakibatkan hasil pengenalan menjadi sedikit lebih rendah. Hal ini dimungkinkan oleh akibat hilangnya informasi yang hilang yang mewakili relief tersebut.

Pengujian rotasi, sudut dan intensitas cahaya menghasilkan akurasi yang hampir sama. Hal ini menandakan bahwa variasi tersebut tidak berpengaruh kepada hasil pengenalan citra relief. Latar belakang dan letak objek relief pada Candi Borobudur yang saling berdekatan sedikit banyak mempengaruhi hasil pengenalan.

Rata-rata waktu yang dibutuhkan dalam identifikasi citra relief adalah 59.55 detik. Untuk waktu komputasi yang lebih cepat dapat digunakan citra masukan dengan ukuran piksel yang lebih kecil. Namun hal ini dapat menurunkan kualitas pengenalan itu sendiri. Untuk hasil pengenalan yang lebih baik, dapat digunakan citra masukan dengan ukuran piksel yang lebih besar, namun hal ini berdampak pada waktu komputasi yang semakin besar.

Melihat kepada penelitian berkaitan identifikasi objek dengan perangkat mobile sebelumnya, khususnya yang memakai metode ekstraksi ciri yang sama (SURF), maka penelitian ini mengalami beberapa kemajuan. Pada penelitian yang dilakukan Temmermans, dkk [12] dengan objek pelatihan sebanyak 17 buah menghasilkan akurasi 84%. Sedangkan penelitian Bay, dkk [6] dengan objek pelatihan sebanyak 20 buah menghasilkan akurasi 83.8% dengan waktu komputasi sebesar 38 detik.

Sedangkan penelitian yang menggunakan metode ekstraksi ciri Linier Binary Pattern (LBP) seperti penelitian [13,14,15] memiliki waktu komputasi yang lebih baik, namun ketahanan terhadap pengujian lain termasuk terhadap citra masukan yang tidak utuh belum diketahui. Jika merujuk kepada penelitian Zou, dkk [16] maka penelitian menggunakan LBP memiliki kelemahan setidaknya pada pengujian variasi skala dan intensitas cahaya.

Hal yang tidak ditemui pada penelitian sebelumnya adalah melihat pengaruh faktor sudut, jarak, rotasi, intensitas cahaya dan keutuhan citra masukan dalam pengenalan sebuah objek. Selain itu, rerata waktu komputasi juga menjadi perhatian, waktu komputasi yang hampir

mencapai 1 menit seringkali membuat koneksi antara aplikasi dengan server terputus (*timeout*).

#### 4. KESIMPULAN

Pada bagian ini diuraikan kesimpulan yang dapat ditarik dari penelitian yang telah dilakukan, yaitu:

1. Hasil pengujian menunjukkan pendekatan metode SURF untuk ekstraksi ciri dan *hierachical k-means nearest-neighbor* dalam pengenalan relief Candi Borobudur memiliki ketahanan yang baik terhadap faktor-faktor yang diujikan, yaitu: jarak, sudut, rotasi, intensitas cahaya dan keutuhan bidang citra relief yang menjadi masukan sistem. Hal ini ditandai dengan tingkat akurasi pengenalan rata-rata sebesar 93.30%.
2. Pengujian dengan jarak 1 meter memberikan nilai kecocokan ciri interest point yang lebih tinggi dari pengujian jarak 2 meter. Jarak pengujian yang lebih dekat memberikan hasil pengenalan yang lebih baik karena tekstur yang didapatkan dari sebuah objek relief bisa lebih mendetail.
3. Pengambilan citra uji yang terpotong tidak mempengaruhi hasil pengenalan obek relief secara signifikan. Namun bidang citra relief masukan yang tidak utuh atau terpotong lebih dari 50% menurunkan tingkat akurasi pengenalan relief tersebut sampai ke prosentase 83.33%.
4. Pengujian pada variasi rotasi dan sudut menghasilkan akurasi yang baik dengan nilai yang hampir sama, yaitu 83.33% sampai dengan 93.33%. Hal ini menandakan bahwa pengenalan citra dengan metode yang diajukan dapat mengatasi pengaruh saat akuisi citra khususnya yang diakibatkan oleh variasi rotasi maupun sudut tertentu saat pengambilan citra.
5. Waktu rata-rata komputasi untuk pengenalan adalah 59.55 detik. Tingginya kebutuhan komputasi membuat hasil penelitian belum dapat secara langsung digunakan oleh wisatawan, karena masih besarnya waktu tunggu yang diperlukan untuk mengidentifikasi sebuah citra relief.

#### 5. SARAN

Pada bagian ini diberikan saran-saran untuk penelitian selanjutnya yang mungkin dapat berguna untuk menutup kekurangan penelitian ini, yaitu:

1. Penggunaan sensor kompas yang umum ada pada setiap perangkat *smartphone* sangat disarankan apabila aplikasi hendak diterapkan di lapangan. Dengan sensor kompas, proses identifikasi akan lebih cepat, karena tidak perlu menyertakan semua data pelatihan dalam basis data ciri untuk proses perbandingan (pencarian objek relief yang cocok), melainkan hanya data ciri objek relief pada mata angin tertentu saja. Penyertaan sensor kompas dapat memotong waktu komputasi berkisar antara 50% sampai dengan 75%.
2. Waktu komputasi dapat ditekan dengan memberikan masukan sistem berupa citra dengan resolusi lebih rendah. Namun hal ini dapat menurunkan akurasi dari sistem secara keseluruhan.

#### DAFTAR PUSTAKA

- [1] Agusmiyati, M., 1996, Variasi Bentuk Pakaian Wanita Abad X-XV Masehi Berdasarkan Penggambaran Pada Relief Candi di Jawa, *Skripsi*, Jurusan Ilmu Arkeologi Fakultas Sastra, Universitas Gadjah Mada.

- 
- [2] Rahman, 1994, *Jenis Mata Pencapaian pada Relief Karmawibhangga Candi Borobudur*, Balai Studi dan Konservasi Borobudur.
- [3] Cole, S., 2008, *Tourism, Culture ad Development: Hopes, Dream and Realities in East Indonesia*, Cromwell Press, Clevedon.
- [4] Purwaningsih, R.M., 2012, Pengaruh Kualitas Pelayanan Pemandu Wisata Terhadap Kepuasan Wisatawan di Candi Prambanan (Tinjauan Khusus Pada Kemampuan Bahasa Verbal), *Tesis*, Program Studi Magister Kajian Pariwisata, Sekolah Pascasarjana Universitas Gadjah Mada.
- [5] Bay, H., Fasel, B. dan Gool, L.V., 2006, Interactive Museum Guide: Fast and Robust Recognition of Museum Objects, *Proceedings of the first international workshop on mobile vision*.
- [6] Chu, E., Hsu, E. dan Yu, S., 2007, *Image-Guided Tours : Fast-Approximated SIFT With U-SURF Features*, Standford University.
- [7] Bay, H., Andreas E., Tinne, T. dan Gool, L.V., 2008, Speeded-Up Robust Features (SURF), *J. of Computer Vision and Image Understanding 110*, pp. 346-359.
- [8] Evans, C., 2009, Notes on the OpenSURF Library, <http://opensurf1.googlecode.com/files/opensurf.pdf>, diakses 26 Mei 2013.
- [9] Tzacheva, A.A., 2013, Preparing for Final Exam, <http://faculty.uscupstate.edu/atzacheva/SHIM450/PreparingForFinalExam.doc>, diakses 15 Juli 2013.
- [10] Mäenpää, T., 2003, The Local Binary Pattern Approach to Texture Analysis, *Ph.D Thesis*, Department of Electrical and Information Engineering, Oulu University.
- [11] Temmermans, F., 2011, The Mobile Museum Guide: Artwork Recognition With Eigenpaintings And SURF, *Proceedings of the 12th International Workshop on Image Analysis for Multimedia Interactive Services*.
- [12] Pravista, D.S., 2012, MEDLEAF: Aplikasi Mobile Untuk Identifikasi Tumbuhan Obat Berbasis Citra Dan Teks, *Skripsi*, Jurusan Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam, Institut Pertanian Bogor.
- [13] Wahyuni, N.K.S., 2012, Penggabungan Fitur Fuzzy Local Binary Pattern Dan Fuzzy Color Histogram Untuk Aplikasi Mobile Identifikasi Citra Tumbuhan Obat, *Skripsi*, Jurusan Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam, Institut Pertanian Bogor.
- [14] Santoni, M.M., 2012, Penggabungan Fitur Morfologi, Local Binary Pattern Variance dan Color Moments untuk Aplikasi Mobile Identifikasi Citra Tumbuhan Obat, *Skripsi*, Jurusan Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam, Institut Pertanian Bogor.
- [15] Zou, J., Ji, Q. dan Nagy, G., 2007, A Comparative Study of Local Matching Approach for Face Recognition, *IEEE Transactions on Image Processing*, vol.16, no.10, pp. 2617-2628.