

# Pembuatan Protoype Aplikasi Pengenalan Logo Berbasis Android

Ferry Tanggo<sup>1</sup>, Justinus Andjarwirawan<sup>2</sup>, Ibnu Gunawan<sup>3</sup>

Program Studi Teknik Informatika Fakultas Teknologi Industri Universitas Kristen Petra

Jl. Siwalankerto 121 – 131 Surabaya 60236

Telp. (031) – 2983455, Fax. (031) – 8417658

E-mail: hnbs\_h\_rcc@yahoo.com, justin@petra.ac.id, ibnu@petra.ac.id

## ABSTRAK

Dewasa ini banyak produsen berlomba-lomba memperkenalkan produknya. Hal ini memicu kreatifitas di bidang periklanan. Iklan-iklan ini dibuat dalam berbagai format, mulai dari media cetak, media video maupun media lainnya.

Tetapi dengan munculnya berbagai format media iklan ini, muncul juga berbagai kendala. Salah satu kendalanya adalah biaya operasional yang cukup tinggi sehingga berimbas pula terhadap harga produk yang semakin tinggi pula.

Cara kerja aplikasi ini adalah memindai objek gambar logo lalu mengirimkan hasil pindaian kepada *server* untuk diproses. Hasil proses akan segera dikirim kembali kepada pengguna *smartphone* dalam bentuk informasi gambar. Fungsi lainnya adalah dapat berbagi informasi tentang logo kepada orang lain dengan menggunakan media sosial yang terhubung dengan internet.

Berdasarkan hasil pengujian, aplikasi ini dapat melakukan pengenalan objek dalam bentuk logo dan menampilkan informasi yang terkait. Akurasi kesuksesan *scan* normal sebesar 68,75%, akurasi pada keadaan redup 0%, pada kemiringan 5°-30° sebesar 60%, pada keadaan logo terpotong sebesar 20%, pada orientasi landscape sebesar 40% dan pada logo kecil sebesar 20%. Pengguna juga dapat menambahkan logo dan informasi yang dibutuhkan, serta dapat pula melakukan interaksi dengan pengguna lainnya melalui fitur komentar.

Kata kunci:

Pengenalan Logo, Android, Media Sosial.

## ABSTRACT

Today many manufacturers are competing to introduce their products. This triggers creativity in advertising. These ads are made in a variety of formats, ranging from print, video or any other media. But with the advent of various media formats of this ads, there is also a variety of constraints. One problem is that operating costs are also high enough to affect the price of the product is higher as well.

The workings of this application is to scan the logo image objects and sent to the server for processing. The results of the process will be immediately sent back to the user of smartphones in the form of image information. Another function is to share information about the logos to others by using social media which connected with the internet.

Based on the test results, the application can perform object recognition in the form of logos and displays related information. Accuracy of the normal scan success of 68.75%, accuracy in dim state of 0%, on a slope of 5 ° -30 ° by 60%, on the state of the

logo is cut off by 20%, in landscape orientation by 40% and the small logo by 20%. Users can also add logos and information needed, and can also interact with other users through the comment feature.

Keywords:

Logo Recognition, Android, Social Media.

## 1. PENDAHULUAN

Dewasa ini banyak produsen berlomba-lomba memasarkan produknya, baik itu informasi tentang produk baru atau pun promosi produk-produk yang sedang trend. Hal ini memicu kreatifitas di bidang periklanan, baik itu di luar maupun di dalam negeri. Iklan-iklan ini dibuat dalam berbagai format, mulai dari media cetak, media video maupun media lainnya.

Tetapi dengan munculnya berbagai format media iklan ini, muncul juga berbagai kendala. Salah satu kendalanya adalah biaya operasional yang cukup tinggi sehingga berimbas pula terhadap harga produk yang semakin tinggi pula.

Perkembangan internet dan pengguna *smartphone* di dunia berkembang sangat pesat, terutama android yang pada 2013 memegang 61,9% market share *smartphone* di dunia. Perkembangan ini juga berpengaruh pada trend iklan saat ini, banyak orang menggunakan *social media* sebagai sarana beriklan, hal ini dikarenakan banyak orang yang juga menggunakan media sosial, sehingga secara tidak langsung orang-orang terdekat maupun kerabatnya dapat melihat iklan tersebut, factor lainnya adalah tidak memerlukan biaya operasional yang besar.

Dari latar belakang yang telah disebutkan di atas, maka dibuatlah *prototype* aplikasi pengenalan logo berbasis android sebagai alternatif lain media cetak yang bertujuan membantu penggunaannya untuk mendapatkan informasi. Aplikasi ini dibuat untuk sistem operasi android pada *smartphone* dan *web server* untuk menyimpan data logo dan informasi.

Cara kerja aplikasi ini adalah melakukan pemindaian terhadap objek gambar logo yang sebelumnya telah disimpan di database, beserta dengan informasi terkait yang merupakan informasi dari logo yang telah disimpan. Hasil pindaian akan dikirim kepada server dan akan langsung diproses, hasil proses oleh server akan dikirim kembali ke *smartphone* pengguna dalam bentuk gambar yang berisi informasi terkait dengan logo yang dipindai.

Dengan menggunakan aplikasi ini, maka diharapkan pengguna dapat menggunakan ponselnya untuk mendapatkan informasi selain dari menggunakan media cetak, membagikan informasi tersebut kepada orang lain dengan menggunakan media sosial yang terhubung dengan internet dan yang terpenting, dapat mengurangi biaya operasional iklan.

## 2. TINJAUAN PUSTAKA

### 2.1 Augmented Reality

Secara garis besar pengertian *augmented reality* adalah penggabungan benda-benda yang ada di dunia maya (virtual) ke dalam dunia nyata dalam bentuk dua dimensi maupun tiga dimensi yang dapat disentuh, maupun dilihat, dan juga dapat didengar.

Benda-benda maya berfungsi menampilkan informasi yang tidak dapat diterima oleh manusia. Hal ini membuat realitas bertambah berguna sebagai alat untuk membantu persepsi dan interaksi penggunaannya dengan dunia nyata. Informasi yang ditampilkan oleh benda maya membantu pengguna melaksanakan kegiatan-kegiatan dalam dunia nyata.[1][2][3]

### 2.2 Android

Android merupakan sistem operasi berbasis Linux dan menyediakan *platform* terbuka bagi *user* untuk mengembangkan dan menciptakan aplikasi yang meraka inginkan. Pada awalnya, Google membeli Android, setelah diyakinkan oleh pendiri android Andy Rubin bahwa google akan tertarik dengan android, dan google pun mulai membuat *software* untuk *smartphone*.

Kemudian untuk pengembangan Android, dibentuklah Open Handset Alliance, yang terdiri dari 34 perusahaan hardware, software, dan telekomunikasi, termasuk Google, HTC, Intel, Motorola, Qualcomm, NVIDIA, dll. Android bersama Open Handset Alliance menyatakan untuk mendukung pengembangan *open source* pada perangkat *mobile* melalui Android.

Di dunia ini terdapat dua jenis distributor sistem operasi android, yang pertama mendapat dukungan penuh dari Google atau Google Mail Service (GMS) dan yang kedua adalah benar-benar bebas distribusinya tanpa dukungan langsung dari Google atau dikenal dengan Open Handset Distribution (OHD). [4]

#### 2.2.1 ADT (Android Development Tools)

*Android Development Tools* (ADT) adalah *plugin* yang didesain untuk IDE Eclipse yang memberikan kita kemudahan dalam mengembangkan aplikasi Android dengan menggunakan IDE Eclipse, membuat *user Interface*, menambahkan komponen-komponen berdasarkan *framework* API Android, *debug* aplikasi, dan pemaketan aplikasi Android. Selain itu, dengan ADT kita dapat membuat *package* Android (\*.apk) yang digunakan untuk mendistribusikan aplikasi Android yang telah kita buat.[4]

### 2.3 Thresholding

*Thresholding* digunakan untuk mengatur *Gray-level* yang ada pada gambar. Misalkan pada sebuah *gambar*,  $f(x,y)$  tersusun dari objek yang terang pada sebuah *background* yang gelap. *Gray-level* milik objek dan milik *background* terkumpul menjadi 2 grup yang dominan. Salah satu cara untuk mengambil objek dari *background*nya adalah dengan memilih sebuah nilai *threshold*  $T$  yang memisahkan grup yang satu dengan grup yang lain. Maka semua *pixel* yang memiliki nilai  $> T$  disebut titik objek, yang lain disebut titik *background*. Proses ini disebut *thresholding*. Sebuah gambar yang telah di *threshold*  $g(x,y)$  dapat didefinisikan :

Nilai  $T$  dapat ditentukan melalui perhitungan rata-rata dari keseluruhan nilai warna yang ada pada gambar. Pada perhitungan ini, nilai  $T$  yang didapat tetap disimpan dalam bilangan *real*. nilai  $T$  yang didapat untuk gambar yang memiliki *histogram* yang telah *ter-equalize* adalah berkisar antara 127 dan 128. Nilai maksimum

dari  $T$  adalah nilai tertinggi dari sistem warna yang digunakan dan nilai minimum dari  $T$  adalah nilai terendah dari sistem warna yang digunakan. Untuk *256-graylevel* maka nilai tertinggi  $T$  adalah 255 dan nilai terendahnya adalah 0. [5]

### 2.4 Grayscale

Proses awal yang banyak dilakukan dalam *image processing* adalah mengubah *image* berwarna menjadi gambar *gray-scale*, hal ini digunakan untuk menyederhanakan model *image*.

*Gray-level* adalah tingkat warna abu-abu dari sebuah *pixel*. Dengan kata lain nilai yang terkandung dalam *pixel* yang menunjukkan tingkat keabu-abuan *pixel* tersebut dari hitam ke putih. Biasanya nilainya ditetapkan antara 0 hingga 255 ( untuk *256-graylevel* ), dengan 0 adalah hitam dan 255 adalah putih. Karena hanya terbatas 1 *byte* saja maka untuk mempresentasikan

$$g(x,y) = \begin{cases} 1 & \text{jika } f(x,y) > T \\ 0 & \text{jika } f(x,y) \leq T \end{cases}$$

nilai *pixel* cukup 8 bit saja. *Grayscale* adalah *image* yang dari tiap *pixel*-nya memiliki *gray-level* sebagai nilai.

*Grayscale* adalah proses perubahan nilai *pixel* dari warna (RGB) menjadi *gray-level*. Pada dasarnya proses ini dilakukan dengan meratakan nilai *pixel* dari 3 nilai RGB menjadi 1 nilai. Untuk memperoleh hasil yang lebih baik, nilai *pixel* tidak langsung dibagi menjadi 3 melainkan terdapat persentasi dari masing-masing nilai. Salah satu persentasi yang sering digunakan adalah 29,9% dari warna merah (*Red*), 58,7% dari warna hijau (*Green*), dan 11,4% dari warna biru (*Blue*). Nilai *pixel* didapat dari jumlah persentasi 3 nilai tersebut. [5]

### 2.5 Histogram

*Histogram* adalah grafik yang menunjukkan distribusi dari intensitas sebuah *image* [5]. *Histogram* dari sebuah *image digital* berupa sebuah fungsi  $h(r_k) = n_k$  dimana  $r_k$  adalah nilai warna ke- $k$  dan  $n_k$  adalah jumlah *pixel* dalam gambar yang memiliki nilai tersebut. Pada *gray-level*,  $r_k$  adalah tingkat *gray-level* ke- $k$ .  $k=0, 1, 2, \dots, L-1$ .  $L$  adalah batas maksimum nilai.

### 2.6 Joint Photographic Experts Group (JPEG)

*JPEG* (Acharya Tinku, Ray Ajoy, 2005) adalah standar kompresi citra internasional pertama bagi *continuous-tone* gambar diam, baik untuk *grayscale* dan gambar berwarna. *JPEG* adalah akronim *Joint Photographic Experts Group*. Standar kompresi citra ini adalah hasil kerjasama dengan *International Telecommunication Union* (ITU), Organisasi Internasional untuk Standardisasi (ISO), dan *International Electrotechnical Commission* (IEC). *JPEG* standar resmi disebut sebagai ISO / IEC IS (International Standard) 10918-1: *Digital Compression and Coding of Continuous-tone Still Images*, dan juga *ITU-T Recommendation T.81*. [6]

### 2.7 JSON

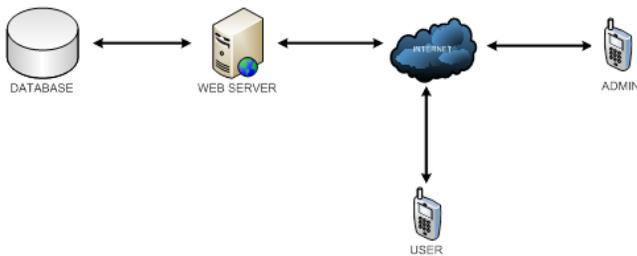
*JSON* (*Javascript Object Notation*) adalah format pertukaran data yang ringan mudah dibaca dan ditulis oleh manusia, serta mudah diterjemahkan dan di *degenerate* oleh komputer. *JSON* merupakan format teks yang tidak bergantung pada bahasa pemrograman apapun karena menggunakan gaya bahasa yang umum digunakan oleh programmer keluarga C termasuk C,C++,C#,Java,JavaScript,dll. [7]

### 3. ANALISA DAN DESAIN SISTEM

#### 3.1 Desain Sistem

Sistem yang akan dibuat terdiri dari dua bagian yaitu *mobile phone* dan *web service*. *Web Service* digunakan sebagai fasilitas penyedia *service*, menerima inputan berupa logo dan informasi baru dalam bentuk *JPEG*, juga melakukan proses *image comparing*, aplikasi ini dapat digunakan oleh *user* untuk melakukan *scan*, menambahkan logo dan informasi, *add friend* dan *group*.

Aplikasi *mobile phone* ini bertindak sebagai *client*, di mana setiap inputan logo dan informasi yang diinputkan oleh *user* langsung dikirim ke dalam *server*, disimpan dan diproses. Hasil dari pencarian yang dilakukan oleh *server* lalu dikirim ke dalam aplikasi *client*. Secara keseluruhan sistem yang akan dibuat dapat dilihat pada Gambar 1.



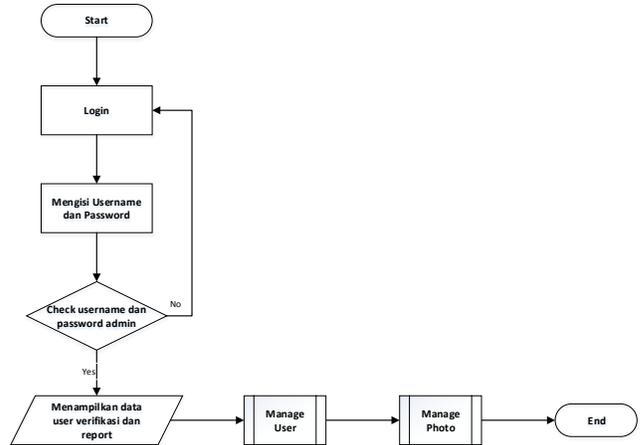
Gambar 1. Desain Cara Kerja Sistem

#### 3.2 Flowchart Sistem pada Administrator

Sistem pada *admin* dimulai dengan proses *login*, dimana *admin* diminta untuk mengisi *username* dan *password*. Setelah menekan tombol *login*, *admin* akan masuk pada aplikasi LogoLayar, *admin* dapat menggunakan semua fungsi standar di aplikasi ini, tapi memiliki 2 tab baru, yaitu *manage user* dan *manage photo*.

Dalam tab *manage user*, *admin* melakukan proses verifikasi sebagai ijin agar *user* yang baru mendaftar dapat menggunakan aplikasi LogoLayar. Tab yang kedua yaitu tab *manage photo*,

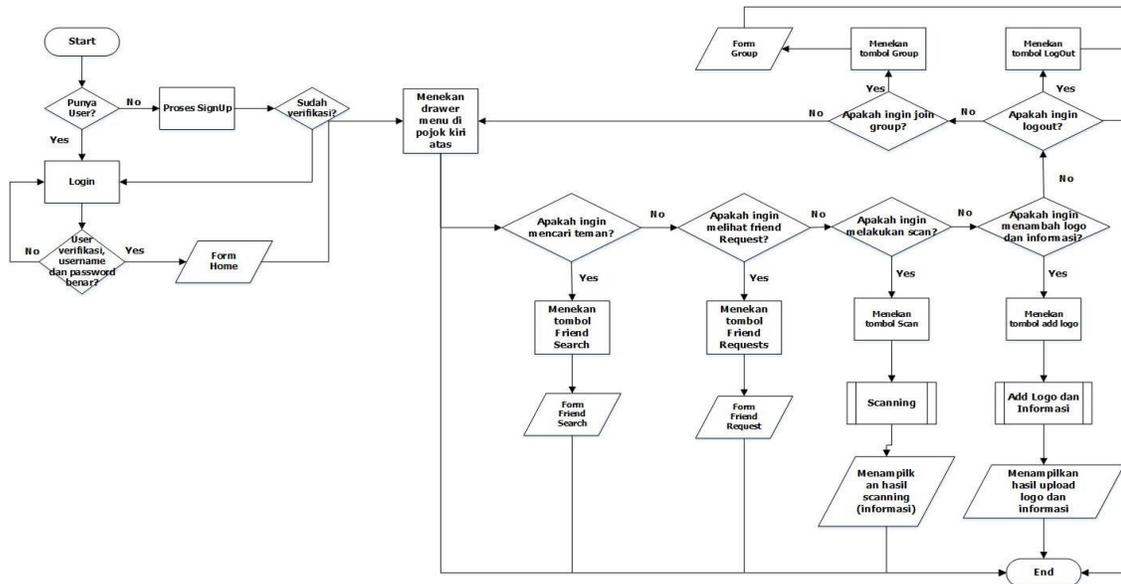
disini *admin* dapat mengecek apakah ada logo/informasi yang tidak sesuai yang di *report* oleh *user*, serta dapat menghapusnya apabila memang tidak sesuai. Desain sistem dari *admin* dapat dilihat pada Gambar 2.



Gambar 2. Desain Sistem Administrasi

#### 3.3 Flowchart Sistem pada Android

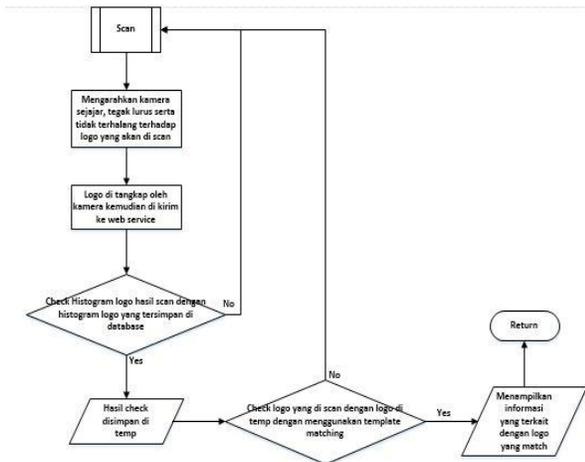
Sistem pada android dimulai dengan memanggil halaman *home*, di *home* berisi daftar *groups* dan daftar postingan *user*. *User* dapat mengakses *drawer menu* pada bagian pojok kiri atas, pilihan menu yang dapat dipilih *user* antara lain : *Home, Search Friend, Friend Request, Scan, Add Logo, Group, dan Logout*. Desain dari sistem android dapat dilihat pada Gambar 3.



Gambar 3. Desain Sistem Android

### 3.4 Flowchart Scan

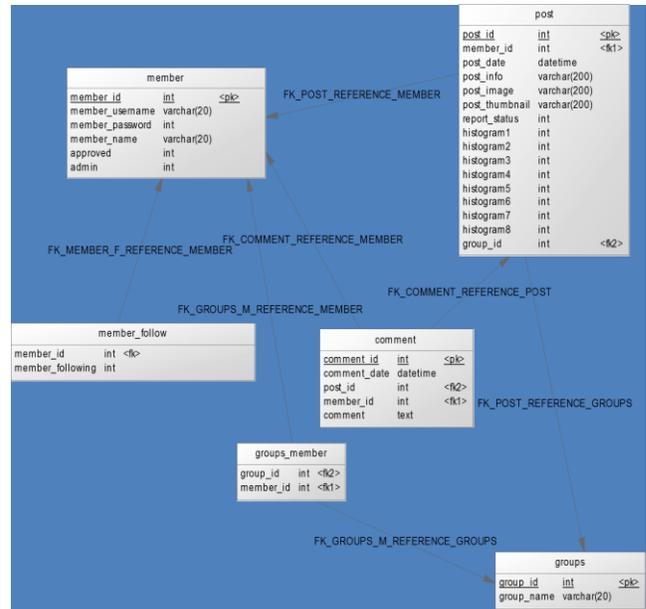
Pada sub proses ini, sistem pada android menggunakan 2 metode *image comparing* yaitu *histogram* dan *template matching*. *Histogram* dibuat agar mempercepat proses *scan*, tidak perlu membandingkan semua logo yang ada di *database* dengan *template matching* sehingga menghabiskan banyak waktu, tapi cukup beberapa logo yang punya *histogram* mirip yang akan di *template matching*, tabel *histogram* pada tiap logo dibuat ketika proses *add* logo pada android dan disimpan di *web server* bersama logo, *histogram* yang digunakan menggunakan 8 kolom untuk menyimpan. *Histogram* ini akan dibandingkan dengan *histogram* dari logo yang ada di *database* yang sudah di *resize*, hasilnya akan diambil 10 terbaik, kemudian hasilnya dibandingkan lagi dengan menggunakan *template matching*, proses *template matching* dilakukan dengan membandingkan *pixel per pixel* dari 10 logo dengan *histogram* terbaik dengan logo yang dikirim, jika ditemukan bahwa logo yang di *scan* dan yang ada di *database match*, maka informasi yang terkait dengan logo di *database* tersebut akan ditampilkan melalui layar *user*.



Gambar 4. Scan

### 3.5 Entity Relationship Diagram

Database yang dibuat terdiri dari 6 tabel, yaitu *member*, *member\_follow*, *post*, *comment*, *groups*, *group\_member*. Relasi keenam tabel tersebut dapat terlihat pada Gambar 5.



Gambar 5. ERD

## 4. HASIL PENGUJIAN

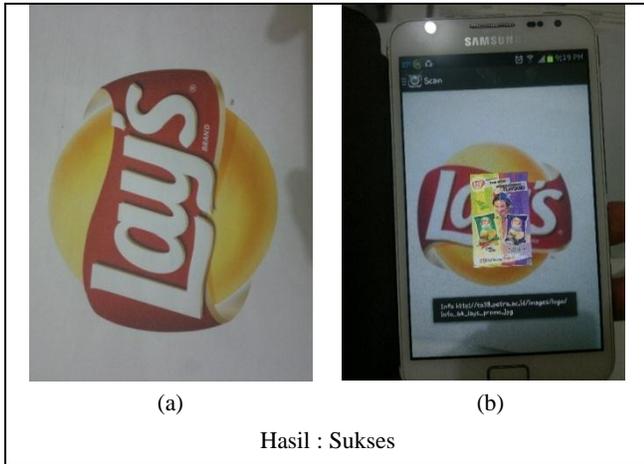
Pengujian dilakukan dengan melakukan scan pada logo dengan posisi miring, *landscape*, logo kecil, logo terpotong. Pengujian dapat dilihat pada Gambar 6. sampai Gambar 7.



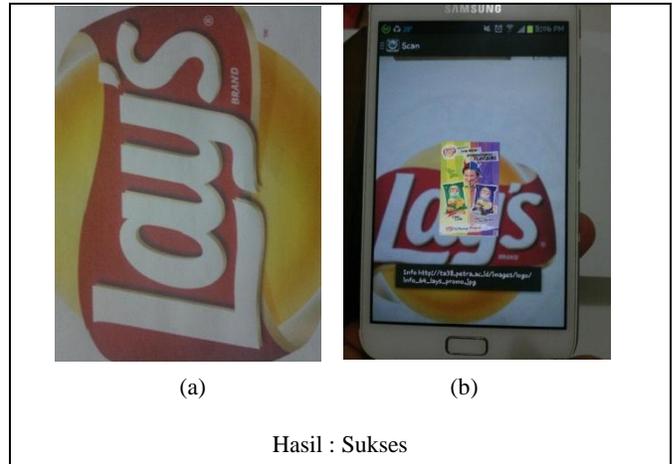
(a) (b)

Hasil : Logo tidak ditemukan

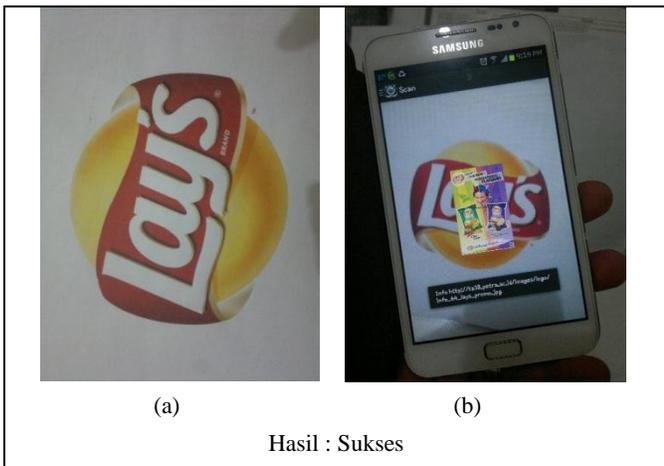
Gambar 6. (a) Logo cahaya redup, (b) Pengujian logo cahaya redup



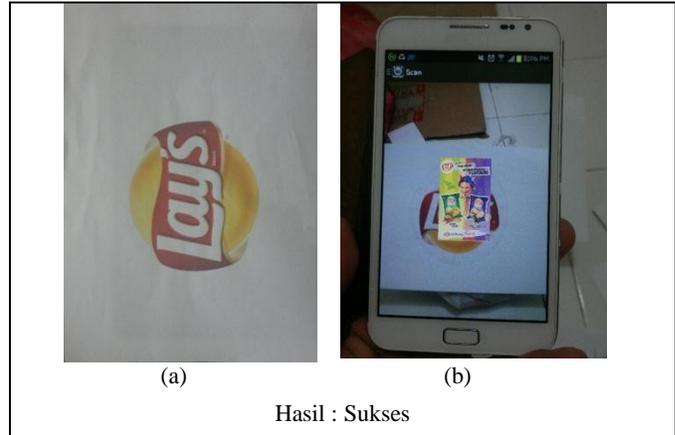
**Gambar 6. (a) Logo miring  $5^\circ$ ,  
(b) Pengujian logo miring  $5^\circ$**



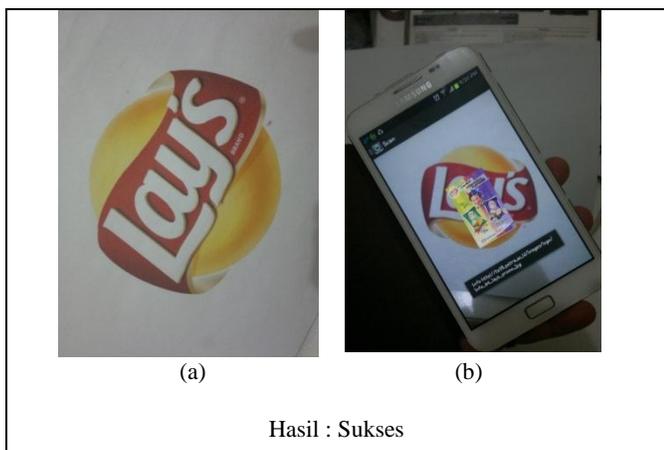
**Gambar 9. (a) Logo terpotong,  
(b) Pengujian logo terpotong**



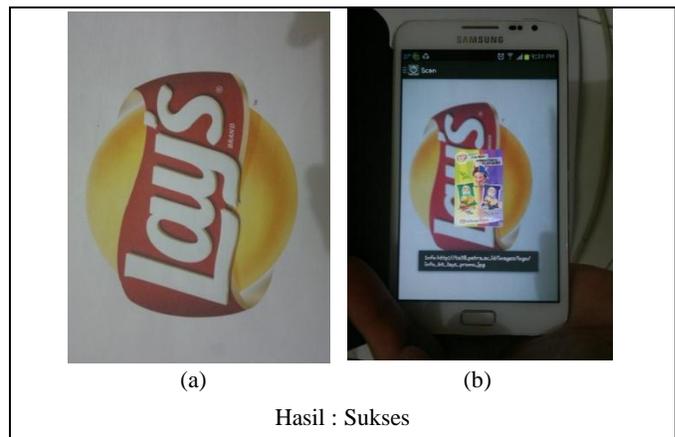
**Gambar 7. (a) Logo miring  $15^\circ$ ,  
(b) Pengujian logo miring  $15^\circ$**



**Gambar 10. (a) Logo kecil,  
(b) Pengujian logo kecil**



**Gambar 8. (a) Logo miring  $30^\circ$ ,  
(b) Pengujian logo miring  $30^\circ$**



**Gambar 11. (a) Logo landscape,  
(b) Pengujian logo landscape**



Gambar 12. Pengujian fitur Scan

Hasil Pengujian :

Menggunakan 16 logo, logo yang berhasil sebanyak 11 logo, akurasi 68,75%.

## 5. KESIMPULAN

Dari proses perancangan sistem hingga pengujian sistem yang sudah dilakukan pada skripsi ini, dapat ditarik beberapa kesimpulan yaitu sebagai berikut.

- Kecepatan *internet* mempengaruhi proses *upload*.
- Kadang terjadi kesalahan menampilkan informasi yang sesuai dengan logo yang di *scan*, apabila *histogram* nya mirip, karena itu perlu pengembangan algoritma yang lebih akurat untuk *image comparing*.

- Cahaya sangat berpengaruh pada proses *scan*, karena penggunaan *histogram*, sehingga cahaya yang terlalu redup mempengaruhi akurasi, terbukti dari pengujian 5 logo, tidak ada logo yang match dalam kondisi cahaya redup.
- Ukuran juga sangat berpengaruh pada akurasi, karena proses *scan* setelah *histogram* menggunakan *template matching*, dengan membandingkan *pixel per pixel*, sehingga perbedaan ukuran membuat akurasi berkurang.
- Batas kemiringan maksimum 30° untuk logo yang berbentuk bulat dan 5° untuk logo berbentuk lain.
- Orientasi *landscape* memiliki akurasi yang lebih kecil daripada *portrait*, terbukti dari pengujian 5 logo hanya 2 logo yang berbentuk bulat yang berhasil.
- Logo yang berbentuk bulat memiliki akurasi lebih tinggi daripada bentuk lainnya.

## 6. DAFTAR PUSTAKA

- [1] Lester Madden. (2012). Augmented Reality Browsers for Smartphones PROGRAMMING FOR JUNAIO, LAYAR, AND WIKITUDE.
- [2] Te-Lien Chou a , Lih-Juan ChanLin. (2012). *Augmented reality smartphone environment orientation application: a case study of the Fu-Jen University mobile campus touring system*.
- [3] Jens Grubert, Tobias Langlotz, Raphael Grasset. (2011). *Augmented Reality Browser Survey*, Desember 2013.
- [4] Safaat, Nazruddin. (2012). Pemograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android.
- [5] Sigit, Riyanto. (2001). *Digital Image Processing with MFC*. Jakarta: PT. Gramedia Pustaka Utama.
- [6] Acharya Tinku, Ray Ajoy. (2005). *Image Processing Principles and Applications*.
- [7] JSON. (2013). *Introduction JSON*. Diambil pada 22 Oktober 2013, dari <http://json.org>.