

Pengenalan Objek pada Computer Vision dengan Pencocokan Fitur Menggunakan Algoritma SIFT

Studi Kasus: Deteksi Penyakit Kulit Sederhana

Tjatur Kandaga Gautama¹, Antonius Hendrik², Riskadewi³

Jurusan Teknik Informatika, Fakultas Teknologi Informasi,

Universitas Kristen Maranatha, Jl. Surya Sumantri No. 65, Bandung

¹tjatur.kandaga@itmaranatha.org, ²antoniushendrik@live.com, ³riskadewi@yahoo.com

Abstract - Human vision can do amazing things such as recognizing people or objects, navigating through obstacles, recognizing the mood in a scene, and imagining stories. To do mimicry of the human vision, the computer requires a sensor that functions like the human eye and a computer program that serves as a data processor from the sensor. Computer vision is the science that uses image processing to make decisions based on images obtained from sensors. In other words, computer vision aims to build an intelligent machine that can "see". Computer vision can be used to detect skin diseases, for example, to detect disease Shingles (Herpes Zoster), Hives (Urticaria), Psoriasis, Eczema, Rosacea, Cold Sores (Fever Blisters), Rash, Razor Bumps, Skin Tags, Acne, Athlete's Foot, moles, Age or Liver Spots, Pityriasis Rosea, Melasma (Pregnancy Mask), Warts, and Seborrheic keratosis. Prewitt, Sobel, Roberts, and Canny operator are used to detect the edges of one or more objects. Then the results will be match with the results of edge detection image data base to determine the type of disease using Scale invariant Feature Transform (SIFT) algorithm. Skin Disease Detection Expert System will be implemented with C++ programming language, IDE MS Visual Studio 2010 and OpenCV 2.4 library.

Keywords— computer vision, edge detection, SIFT algorithm, skin disease

I. PENDAHULUAN

Penglihatan manusia (*human vision*) dapat melakukan hal-hal yang menakjubkan seperti mengenali orang/objek, navigasi dalam menghindari rintangan, mengenali mood di dalam sebuah adegan (*scene*), dan membayangkan cerita (*imagine stories*). Lain halnya dengan mesin, yang dalam konteks ini adalah komputer. Untuk melakukan mimikri terhadap penglihatan manusia, komputer memerlukan sensor yang berfungsi layaknya mata pada manusia dan program komputer yang berfungsi sebagai pemroses data dari sensor. *Computer vision* merupakan ilmu yang menggunakan *image processing* untuk membuat keputusan berdasarkan citra yang didapat dari sensor [1]. Dengan kata lain, *computer vision* bertujuan untuk membangun sebuah

mesin pandai yang dapat "melihat". Kerangka kerja umum yang biasa dilakukan dalam *computer vision* adalah: *image acquisition* (akuisisi citra), *pre-processing* (pra pemrosesan), *feature extraction* (ekstraksi fitur), deteksi/segmentasi, pemrosesan tingkat tinggi, dan pengambilan keputusan [2].

Melihat kompleksitas proses akuisisi sampai dengan pengambilan keputusan, maka penelitian meliputi dua tahap besar, yaitu: Ekstraksi fitur untuk deteksi tepi antar obyek; dan pengaplikasiannya untuk pengenalan obyek, yaitu untuk mendeteksi penyakit kulit, misalnya Shingles (Herpes Zoster), Hives (Urticaria), Psoriasis, Eczema, Rosacea, Cold Sores (Fever Blisters), Rash, Razor Bumps, Skin Tags, Acne, Athlete's Foot, Moles, Age or Liver Spots, Pityriasis Rosea, Melasma (Pregnancy Mask), Warts, dan Seborrheic Keratosis.[3].

Dalam penelitian ini ditekankan pada tahap kedua yaitu pengenalan obyek untuk mendeteksi penyakit kulit. Pada penelitian ini akan dihasilkan sistem pakar deteksi penyakit kulit. Sistem pakar ini menerima input berupa citra digital penyakit kulit, identitas pasien seperti umur, jenis kelamin, dan sejarah penyakit kulit yang pernah diderita, dan gejala dari penyakit kulit seperti lokasi, *leison*, *surface*, warna, dan kegatalan. Citra digital kemudian akan dilakukan *pre-processing*, dan mendeteksi tepi dari citra. Hasil deteksi tepi dengan operator yang optimal akan dicocokkan dengan hasil deteksi tepi basis data citra [4]. Pengenalan obyek dilakukan dengan algoritma *Scale Invariant Feature Transform* (SIFT). Output dari sistem pakar adalah jenis penyakit kulit yang diderita oleh pasien.

II. LANDASAN TEORI

Masalah klasik dalam *computer vision*, *image processing*, dan *machine vision* adalah untuk menentukan apakah citra mengandung objek, *feature*, dan aktivitas tertentu. Tugas ini biasanya dapat diselesaikan secara cepat dan mudah oleh manusia, tetapi masih belum terselesaikan dengan baik oleh *computer vision*, misalnya pengenalan obyek tertentu pada

situasi tertentu. Metode yang telah ada untuk menangani masalah ini hanya dapat mengenali obyek tertentu, misalnya obyek geometri sederhana, wajah manusia, karakter hasil cetak, karakter tulisan tangan, atau kendaraan, dan dalam situasi tertentu, biasanya dideskripsikan dalam pencahayaan yang baik, latar belakang yang dapat dikenali, dan pose objek yang umum yang dapat dikenali relatif terhadap posisi kamera. [5]

Masalah pengenalan obyek adalah sebagai berikut:

1. Pengenalan obyek
Satu atau beberapa obyek yang telah ditetapkan sebelumnya, atau kelas objek yang dapat dikenali, berikut posisi 2 dimensi dalam citra atau pose 3D pada suatu adegan (*scene*). Contoh: program pengenalan objek pada Google Goggles.
2. Identifikasi
Pengenalan sebuah instansiasi (*instance*) dari sebuah objek. Contoh: identifikasi wajah, identifikasi sidik jari, atau identifikasi kendaraan tertentu.
3. Deteksi
Image dipindai untuk kondisi tertentu (*scanned for specific condition*). Contoh: deteksi sel atau jaringan abnormal dalam citra medis, deteksi kendaraan di sebuah sistem jalan tol otomatis (*automatic road toll system*). Deteksi yang berdasarkan pada perhitungan yang relatif sederhana dan cepat, biasanya digunakan untuk mencari daerah kecil dari gambar yang menarik, yang hasilnya dapat dianalisa lebih lanjut dengan menggunakan teknik yang lebih baik untuk menghasilkan interpretasi yang benar.

Fungsi yang biasa ditemukan dalam sistem *computer vision* adalah sebagai berikut:

1. Image acquisition

Sebuah citra digital dihasilkan oleh satu atau beberapa sensor citra, di mana, selain beberapa tipe kamera yang sensitif terhadap cahaya (*light-sensitive cameras*), termasuk sensor kedalaman, perangkat tomografi, radar, kamera ultrasonik, dan lain-lain. Berdasarkan jenis sensor, citra yang dihasilkan biasanya adalah citra 2D, citra 3D, atau gambar yang berurutan. Nilai *pixel* biasanya berkorespondensi dengan intensitas cahaya dalam satu atau beberapa *spectral band* (citra abu-abu atau citra berwarna), tetapi dapat juga berhubungan dengan berbagai ukuran fisik (*physical measures*), seperti kedalaman, penyerapan atau pemantulan gelombang sonik, penyerapan atau pemantulan gelombang elektromagnetik, atau resonansi magnetik nuklir (*nuclear magnetic resonance*).

2. Pre-processing

Sebelum sebuah metode *computer vision* diterapkan pada citra, untuk mendapatkan beberapa bagian informasi yang spesifik, biasanya diperlukan pemrosesan data yang bertujuan untuk memastikan data tersebut memenuhi asumsi

tertentu yang berhubungan dengan metode tersebut. Misalnya,

- *Re-Sampling* untuk menjamin bahwa sistem koordinat citra sudah benar.
- Pengurangan noise (*noise reduction*) untuk menjamin bahwa sensor noise tidak mengandung informasi palsu.
- Perbaikan kontras untuk memastikan bahwa informasi yang relevan dapat dideteksi.
- Representasi skala-ruang untuk meningkatkan struktur gambar agar sesuai dengan skala lokal.

3. Feature extraction

Feature citra dengan berbagai tingkat kompleksitas, yang diekstrak dari citra. Misalnya,

- Garis, tepi, dan tepian.
- Sekumpulan titik yang menarik, seperti sudut, gumpalan (*blob*), atau titik.
- Feature yang lebih kompleks, yang berhubungan dengan tekstur, bentuk, atau pergerakan.

4. Deteksi/segmentasi

Pada titik tertentu dalam pemrosesan, sebuah keputusan dibuat untuk menentukan titik atau daerah pada citra yang relevan untuk pemrosesan lebih lanjut. Misalnya,

- Pemilihan sekumpulan titik yang menarik.
- Segmentasi dari satu atau beberapa daerah citra, yang mengandung objek tertentu yang menarik.

5. Pemrosesan tingkat tinggi

Masukan pada langkah ini biasanya adalah satu kumpulan kecil data, misalnya satu kumpulan titik atau daerah citra yang mengandung objek tertentu yang menarik. Pemrosesan ini berhubungan dengan :

- Verifikasi bahwa data memenuhi asumsi spesifik berbasis pada model dan aplikasi.
- Estimasi parameter aplikasi yang spesifik, seperti pose atau ukuran obyek.
- Pengenalan citra, dengan mengelompokkan objek yang dideteksi ke berbagai kategori.
- Pendaftaran (*registration*) citra, dengan membandingkan dan menggabungkan dua pandangan (*view*) yang berbeda dari objek yang sama.
- Pengambilan keputusan (*decision making*)

Pengambilan keputusan akhir dibutuhkan untuk aplikasi, misalnya

- Lulus/tidak (*pass/fail*) lulus pada aplikasi otomatisasi inspeksi.
- Sama (*match*) /tidak sama pada aplikasi pengenalan.
- Penandaan untuk diperiksa lebih lanjut oleh manusia pada aplikasi medis, militer, keamanan, dan pengenalan.

Pengenalan objek dalam citra adalah salah satu fungsi utama dari pengolahan citra dan *computer vision*. Biasanya, skema yang berhubungan dengan pengenalan objek adalah

satu atau lebih citra yang dievaluasi untuk menentukan objek apa saja yang terdapat di dalamnya dan posisi dari objek tersebut. Agar dapat mengenali objek, dibutuhkan pengetahuan mengenai tampilan (*apperance*) dari objek yang akan dicari (model yang telah dibuat sebelumnya).

Invariance bertujuan untuk meningkatkan sensitivitas yang berhubungan dengan informasi yang menekankan pada perbedaan objek dengan kelas yang berbeda (*inter-class variance*), dan mengurangi sensitivitas yang berhubungan dengan informasi yang menekankan pada perbedaan objek dalam kelas yang sama (*intra-class variance*). Invariance berhubungan dengan pencahayaan, ukuran, rotasi, latar belakang yang bertumpuk (*background clutter*), *partial occlusion*, dan perubahan sudut pandang. [6] [7]

Feature extraction berhubungan dengan menemukan bentuk (*shape*) dalam citra digital. Agar dapat mengenali wajah secara otomatis, sebagai contoh, salah satu pendekatan yang dipakai adalah mengeluarkan feature komponen. Pendekatan ini memerlukan pengekstrakan mata, telinga, dan hidung, yang merupakan ciri-ciri wajah. Untuk mencarinya, gunakan bentuk dari feature tersebut, misalnya bentuk mata adalah elipsoid, bentuk mulut adalah dua garis, sama halnya dengan alis. Dalam banyak aplikasi, analisis dapat dipandu oleh posisi dari bentuk tersebut, misalnya posisi mata di atas dan berada di antara hidung, dan posisi mulut berada di bawah hidung. [8]

Dalam pengekstrakan feature, biasanya dilakukan pencarian properti konstan (*invariance properties*) sehingga proses ekstraksi tidak berbeda berdasarkan kondisi yang dispesifikasikan. Dengan kata lain, teknik ekstraksi harus dapat menemukan bentuk tanpa memperdulikan nilai parameter yang mengontrol tampilan dari sebuah bentuk. misalnya perubahan tingkat pencahayaan (kontras antara bentuk dan latar belakang), posisi (*translation invariance*), rotasi (*orientation invariance*), dan ukuran (jarak antara kamera dengan objek tertentu).

Untuk mengekstrak bentuk dari sebuah citra, diperlukan cara untuk mengidentifikasi bentuk tersebut dari elemen latar belakang. Hal ini dapat dilakukan dengan mempertimbangkan informasi intensitas atau membandingkan pixel dengan template tertentu. Template matching adalah pendekatan yang berbasis model dimana bentuk diekstrak dengan mencari korelasi terbaik antara model yang diketahui sebelumnya dengan pixel dalam sebuah citra.

Template matching dilakukan dengan menempatkan template pada sebuah titik dalam sebuah citra, kemudian dihitung jumlah titik pada template yang cocok dengan titik pada citra. Prosedur ini dilakukan berulang-ulang hingga seluruh bagian dari citra. Titik yang menghasilkan kecocokan yang terbaik (yang memiliki jumlah pencocokan

yang maksimum), adalah titik dimana bentuk berada dalam citra.

OpenCV (*Open Source Computer Vision*) adalah sebuah pustaka (*library*) perangkat lunak *computer vision* dan *machine learning* yang bersifat *open source*. OpenCV dibangun untuk menyediakan infrastruktur umum untuk aplikasi *computer vision* dan untuk mempercepat penggunaan persepsi mesin (*machine perception*) dalam produk komersial. OpenCV berlisensi BSD (*Berkeley Software Distribution*), sehingga memudahkan bagi pelaku bisnis untuk memanfaatkan dan mengubah kode.

OpenCV *library* memiliki lebih dari 2500 algoritma optimal, yang mencakup sekumpulan algoritma *computer vision* dan *mesin learning* bertipe klasik dan canggih (*state-of-the-art*). Algoritma-algoritma ini dapat digunakan untuk mendeteksi dan mengenali wajah, mengidentifikasi objek, mengklasifikasikan tindakan manusia dalam video, melacak gerakan kamera, melacak obyek yang bergerak, ekstrak model 3D dari obyek, menghasilkan 3D *point clouds* dari kamera stereo, menggabungkan citra untuk menghasilkan citra dengan resolusi tinggi dari seluruh adegan, menemukan citra yang sama dari database citra, menghapus mata merah dari citra yang diambil menggunakan flash, mengikuti gerakan mata, mengenali pemandangan dan membuat penanda (*marker*) untuk melapisi (*overlay*) penanda dengan *augmented reality*, dan lain-lain. [9]

Operasi paling mendasar dalam pengembangan aplikasi OpenCV adalah menambahkan sebuah citra input dari file, menampilkan sebuah citra pada sebuah window, dan menyimpan output sebuah citra pada disk. Hal pertama yang harus dilakukan adalah mendeklarasikan variabel yang akan menampung citra, yaitu sebuah objek dari kelas `cv::Mat`.

```
cv::Mat image;
```

Definisi ini menghasilkan citra dengan ukuran (*size*) 0. Untuk mengujinya, gunakan method `size()` untuk membaca ukuran citra saat ini. Method ini mengembalikan struktur yang mengandung tinggi dan lebar dari citra:

```
std::cout << "size: " <<
image.size().height << " , " <<
image.size().width << std::endl;
```

Untuk membaca citra dari file, *decode* citra, dan mengalokasikan memori, gunakan perintah sebagai berikut:

```
image= cv::imread("img.jpg");
```

Untuk mengecek validitas citra, gunakan perintah berikut ini:

```
// no image has been created..
if (!image.data) {
}
```

Citra dikategorikan tidak valid, jika file tidak ditemukan, file rusak, atau file tidak dikenali formatnya.

Untuk menampilkan citra, gunakan modul `highgui` yang disediakan oleh `OpenCV`. Mula-mula deklarasikan window mana yang akan dipakai untuk menampilkan citra. Kemudian tentukan citra yang akan ditampilkan pada window tersebut. Contoh kode dapat dilihat pada bagian berikut :

```
// define the window
cv::namedWindow("Original Image");
// show the image
cv::imshow("Original Image", image);
```

Transformasi citra dalam `OpenCV` dapat dilakukan *in-place*, artinya transformasi diterapkan langsung pada citra hasil input (tidak ada citra baru dihasilkan). Atau menggunakan matriks lain untuk menyimpan hasil pemrosesan. Perhatikan contoh penggunaan method `flip` berikut ini:

```
cv::Mat result;
cv::flip(image, result, 1);
// positive for horizontal
// 0 for vertical,
// negative for both
```

Kemudian hasilnya ditampilkan di window lain seperti contoh kode berikut ini:

```
cv::namedWindow("Output Image");
cv::imshow("Output Image", result);
```

Untuk menunggu pengguna sebelum mengakhiri program, gunakan perintah berikut ini:

```
cv::waitKey(0);
```

Untuk menyimpan citra yang telah diproses pada disk, gunakan method `highgui` berikut ini: [10]

```
cv::imwrite("output.bmp", result);
```

Algoritma `SIFT` mengubah data citra menjadi koordinat *scale-invariant* yang bersifat relatif terhadap feature lokal. Aspek yang paling penting pada pendekatan ini adalah pembangunan sejumlah besar feature yang melingkupi seluruh citra untuk setiap skala dan lokasi. Citra dengan ukuran 500 x 500 pixel akan menghasilkan 2000 feature. Jumlah feature sangat penting untuk pengenalan objek yang handal, di mana kemampuan untuk mendeteksi objek yang kecil dalam latar belakang yang bertumpuk membutuhkan sedikitnya tiga feature berhasil dicocokkan untuk setiap objek.

Mula-mula feature `SIFT` diekstrak dari sekumpulan citra dan disimpan dalam basis data. Sebuah citra dicocokkan dengan membandingkan setiap feature dari citra tersebut dengan feature dalam basis data, dan pencarian setiap kandidat feature yang cocok berdasar pada jarak *Euclidean* dari vektor feature. Algoritma `Fast Nearest-Neighbor` dipakai agar dapat melakukan komputasi yang cepat terhadap basis data yang besar.

Penggunaan `Keypoint descriptor` mengakibatkan sebuah feature dapat menemukan pencocokan yang benar (*correct match*) dengan probabilitas yang besar pada basis data feature yang besar. Tetapi pada citra yang bertumpuk, banyak feature dari latar belakang tidak memiliki pencocokan yang benar dalam basis data. Hal ini menghasilkan peningkatan kesalahan pada pencocokan. Pencocokan yang benar dapat dipisahkan dari sekumpulan pencocokan dengan mengidentifikasi bagian dari `keypoint` yang cocok dengan objek dan lokasi, ukuran, dan orientasi objek tersebut dalam citra. Penetapan cluster yang konsisten dapat dilakukan dengan mengimplementasikan *hash table* yang efisien dari `Generalized Hough Transform`.

Setiap cluster yang terdiri dari atau lebih feature yang cocok dengan objek dan pose dari objek tersebut akan diverifikasi lebih lanjut. Mula-mula dilakukan estimasi *least-square* sebagai pendekatan nilai tertentu (*affine approximation*) untuk pose dari objek. Setiap fitur dari citra yang konsisten dengan pose ini akan diidentifikasi, dan sisanya dibuang. Kemudian, komputasi detail dibuat berdasarkan probabilitas bahwa sekumpulan feature mengindikasikan adanya sebuah objek, jika diberikan akurasi pencocokan yang tepat dan tidak tepat. Objek yang memenuhi pengujian di atas dapat diidentifikasi sebagai objek yang tepat dengan tingkat kepercayaan yang tinggi. [11]

Penerapan `SIFT` pada pengenalan obyek dilakukan dengan terlebih dahulu mencocokkan setiap *keypoint* dengan basis data *keypoint* hasil ekstraksi dari citra pelatihan. Banyak dari pencocokan awal akan salah karena fitur ambigu atau fitur yang muncul dari latar belakang yang kacau. Oleh karena itu, cluster dengan minimal tiga fitur yang mula-mula diidentifikasi adalah yang menyerupai objek dan pose dari objek, cluster seperti ini memiliki kemungkinan lebih tinggi daripada pencocokan fitur individual. Kemudian, masing-masing cluster diperiksa dengan melakukan pencocokan geometris rinci pada model, dan hasilnya digunakan untuk menerima atau menolak interpretasi dari objek.

Pencocokan kandidat terbaik untuk setiap *keypoint* ditemukan dengan mengidentifikasi tetangga terdekat dalam basis data *keypoint* dari citra pelatihan. Tetangga terdekat didefinisikan sebagai *keypoint* yang memiliki jarak *Euclidean* minimum untuk vektor deskriptor invarian.

Banyak fitur dari sebuah citra tidak akan memiliki pencocokan dalam basis data pelatihan akibat dari kekacauan latar belakang atau tidak terdeteksi dalam citra pelatihan. Oleh karena itu, fitur yang tidak memiliki pencocokan dalam basis data lebih baik dibuang. Ambang batas global terhadap jarak ke fitur terdekat tidak akan bekerja dengan baik, karena beberapa deskriptor jauh lebih diskriminatif daripada yang lain. Ukuran efektivitas lainnya

diperoleh dengan membandingkan jarak tetangga terdekat dengan yang ada dengan tetangga kedua terdekat. Jika ada beberapa citra pelatihan dari objek yang sama, maka tetangga terdekat kedua didefinisikan sebagai tetangga terdekat yang diketahui berasal dari objek yang berbeda dari yang pertama, misalnya hanya menggunakan citra yang diketahui mengandung objek yang berbeda. Langkah ini berkinerja baik karena pencocokan yang tepat perlu memiliki tetangga terdekat secara signifikan lebih dekat dari pencocokan terdekat yang tidak tepat untuk mencapai pencocokan handal. Pencocokan kedua terdekat memberikan perkiraan kepadatan pencocokan tidak tepat dari ruang fitur dan pada saat yang sama mengidentifikasi contoh-contoh spesifik dari fitur ambigu. [12]

III. ANALISIS & DESAIN

Beberapa faktor yang berpengaruh dalam menentukan jenis penyakit kulit adalah gender (jenis kelamin), age (usia), color (warna), history (sejarah penyakit dalam keluarga), tingkat kegatalan, dan site (lokasi terjadinya penyakit kulit) [13]. Decision table penentuan jenis penyakit kulit dapat dilihat pada tabel I berikut ini.

TABEL I
DECISION TABLE

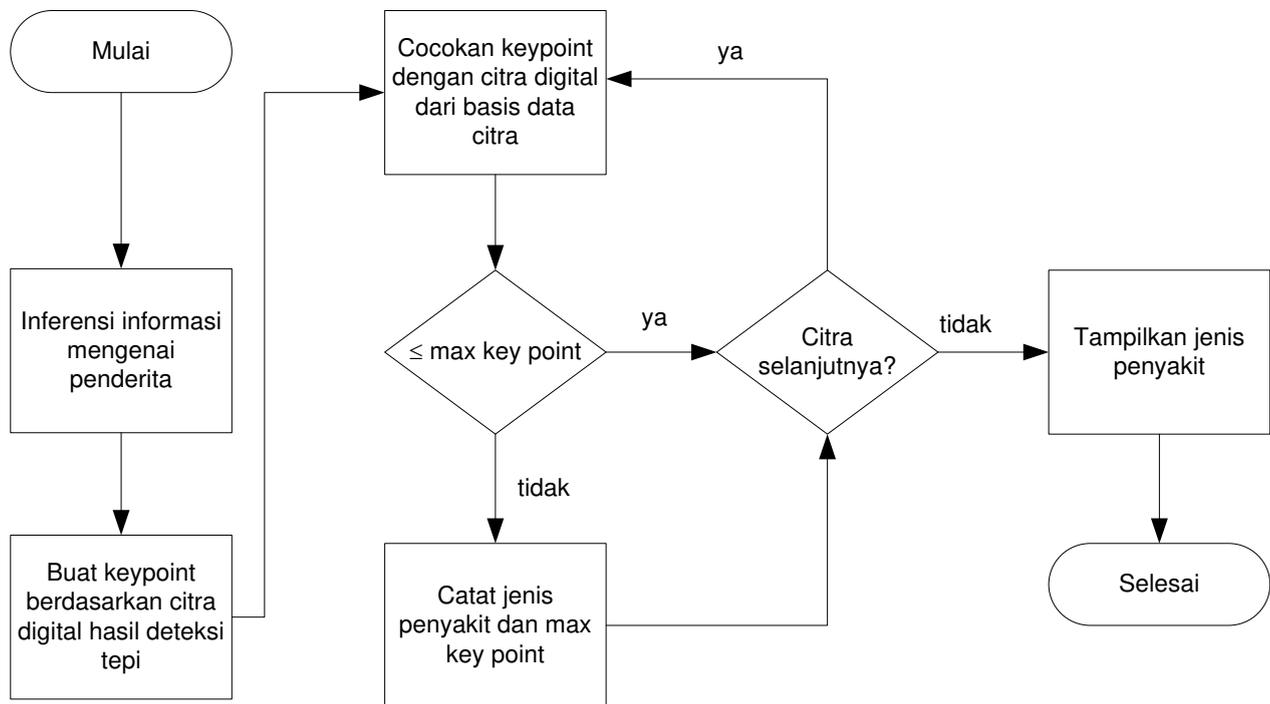
Name	Site	Gender	Age	Color	History	Itchiness
Acne	face, back, chest, neck	All	young	red	Family / personal	non-itchy
Age or Liver Spots	face, forehead, forearm, shoulder, back, hands	All	old	brown, gray	personal	non-itchy
Athlete's Foot	Feet	All	all	red	personal	Itchy
Cold Sores (Fever Blisters)	lip, mouth, nose, chin	All	young	red	personal	non-itchy
Eczema	face, elbows and knees, and around the wrists and neck, hand, feet	All	infant	red, brown	family	Itchy
Hives (Urticaria)	All	All	all	red	personal	Itchy

Name	Site	Gender	Age	Color	History	Itchiness
Melasma (Pregnancy Mask)	Face	Female	adult, old	brown	personal	non-itchy
Moles	All	All	all	brown, black, blue, white, red	personal	non-itchy
Pityriasis Rosea/Rosacea	nose, cheeks, chin, and forehead	All	adult, old	red	personal	non-itchy
Psoriasis	scalp, nail, arm, leg, trunk, arm, genital	All	all	red	personal	non-itchy
Razor Bumps	face, neck, genital area	All	young, adult, old	red	personal	non-itchy
Scabies	all	All	all	red	personal	itchy
Seborrheic Keratosis	all	All	all	brown, white, black	family	non-itchy
Shingles (Herpes Zoster)	forehead, cheek, nose, and around one eye	All	All	red	personal	itchy
Skin Tags	neck, chest, back, armpits, under the breasts, genital area	All	All	brown	personal	non-itchy
Warts (common)	All	All	Young	gray, brown	personal	non-itchy

Data input berupa citra digital hasil deteksi tepi dari penyakit kulit (hasil penelitian tahap 1), informasi mengenai penderita penyakit kulit seperti jenis kelamin, usia, warna penyakit kulit, sejarah penyakit dalam keluarga, tingkat kegatalan, dan lokasi terjadinya penyakit kulit. Mula-mula akan dilakukan inferensi informasi mengenai penderita penyakit kulit dengan metode *Forward Chaining* untuk menentukan kandidat jenis penyakit kulit yang akan dicocokkan pada proses berikutnya. Berdasarkan citra digital

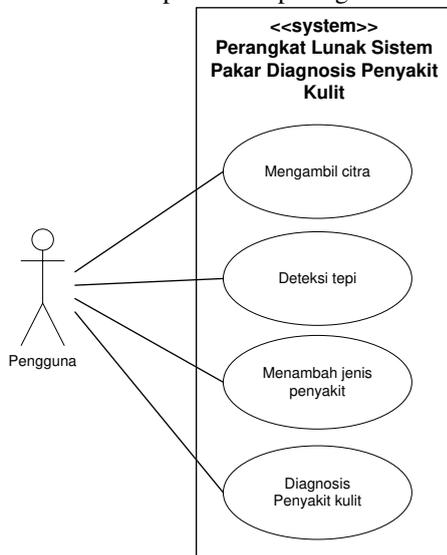
hasil input, akan dihasilkan sejumlah *keypoint* menggunakan algoritma SIFT. *Keypoint* tersebut akan dicocokkan dengan *keypoint* dari basis data citra. Hasil pencocokan *keypoint* yang terbesar akan menentukan jenis penyakit kulit yang

diderita oleh pasien tersebut. Proses pencocokan dilakukan dengan metode Brute Force. Flowchart proses pencocokan dapat dilihat pada gambar 1.



Gambar 1. Proses Pencocokan.

Diagram Use Case dapat dilihat pada gambar 2.



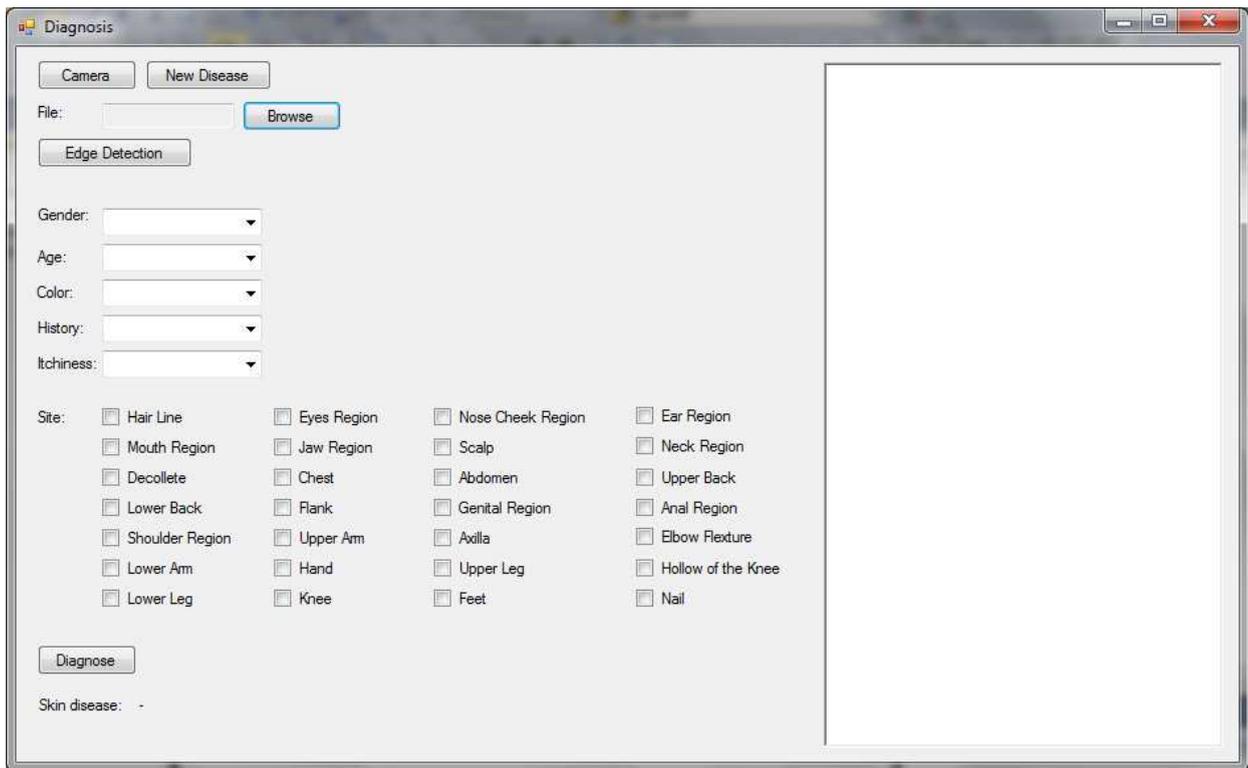
Gambar 2. Diagram Use Case

Aplikasi diimplementasikan menggunakan kelas CVCapture untuk mengambil citra dari kamera, kelas iplimage untuk menyimpan citra digital, kelas mat untuk menyimpan matriks citra, kelas SIFT untuk mendapatkan *keypoint* menggunakan algoritma SIFT, kelas FLANN untuk pencocokan *keypoint* menggunakan algoritma FLANN, kelas BFMatcher untuk pencocokan *keypoint* menggunakan algoritma brute force, kelas driver, connection, statement, dan result set untuk memproses penyimpanan ke dalam basis data citra. Penyimpanan data dalam tabel untuk menyimpan data umur, warna, penyakit, lokasi, gender, dan sejarah (*history*) penyakit.

IV. IMPLEMENTASI

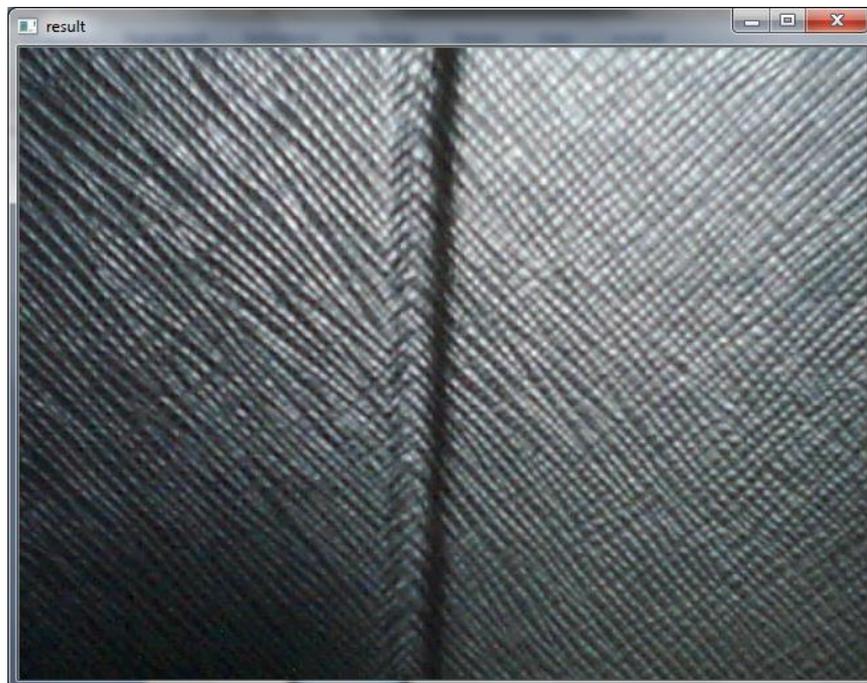
Pengkodean perangkat lunak dilakukan dengan menggunakan bahasa pemrograman C++, basis data MySQL 5.0, lingkungan pengembangan MS Visual Studio 2010, dan pustaka OpenCV 2.5 [14].

Menu utama dapat dilihat pada gambar 3.



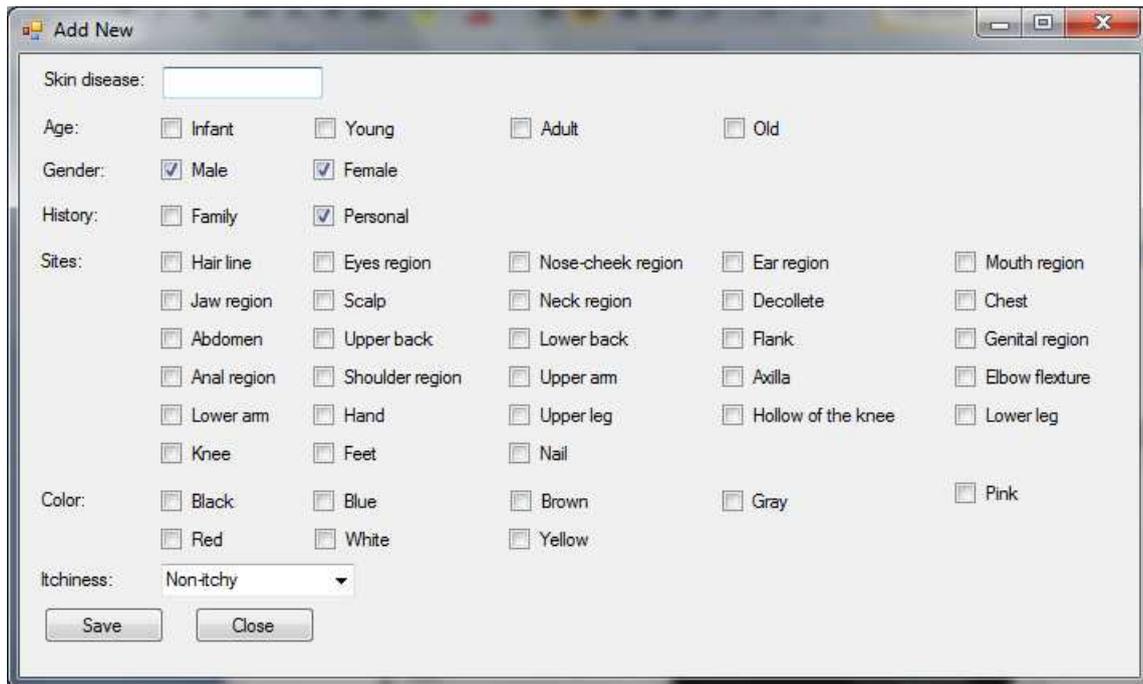
Gambar 3. Menu Utama.

Setelah mengklik tombol Camera, maka akan ditampilkan form capture seperti pada gambar 4.



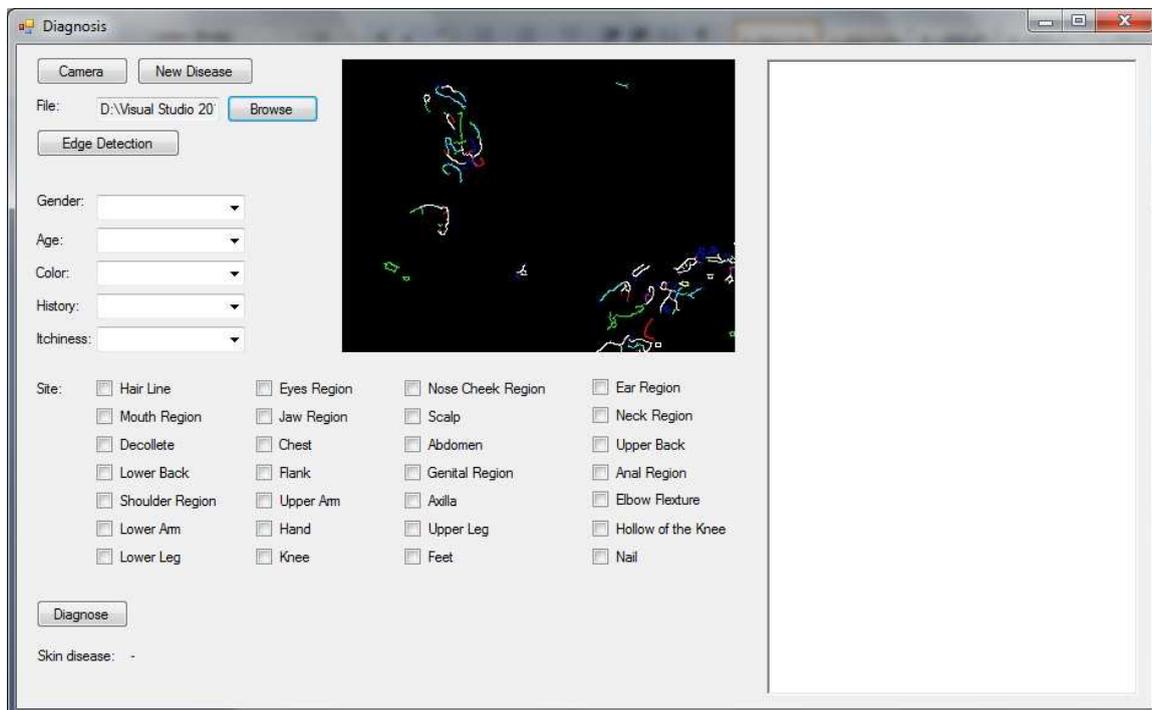
Gambar 4. Form Capture.

Jika menekan tombol Add new, maka akan ditampilkan form seperti pada gambar 5.



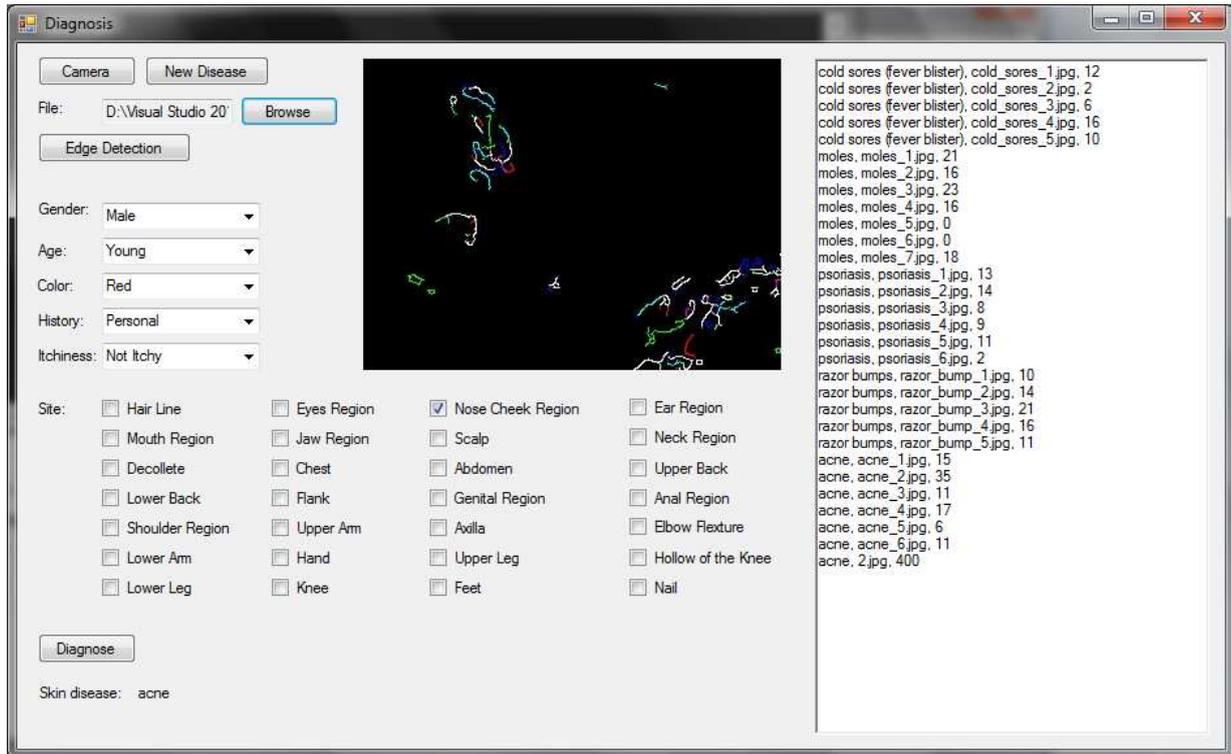
Gambar 5. Form Menambah Jenis Penyakit.

Setelah mengklik tombol Browse, maka akan ditampilkan file open dialog untuk memilih file citra penyakit. Setelah itu pada menu utama akan ditampilkan nama file yang dipilih dan gambar yang dipilih seperti pada gambar 6 berikut ini.



Gambar 6. Form Membuka File Gambar.

Pilih gender, age, color, history, dan itchiness. Setelah menekan tombol Diagnose, maka akan ditampilkan jenis penyakit seperti pada gambar 7.



Gambar 7. Form Hasil Akhir

V. PENGUJIAN DAN EKSPERIMEN

Pengujian bertujuan untuk menguji hasil diagnosis yang dihasilkan oleh perangkat lunak. Pengujian dilakukan dengan membandingkan jenis penyakit yang telah disimpan dalam basis data citra dengan jenis penyakit hasil diagnosis perangkat lunak. Hasil deteksi tepi yang dipakai pada pengujian adalah operator Canny. Teknik pencocokan yang dipakai adalah Brute Force. Jumlah maksimum keypoint yang akan dihasilkan adalah 400.

TABEL II
KRITERIA PENGUJIAN

No.	Kriteria Pengujian (Gender, Age, Color, History, Itchiness, Site)	Kandidat Jenis Penyakit
1.	Male, young, red, personal, non itchy, nose/cheek region	Acne, cold sores, psoriasis, razor bump
2.	Male, young, red, personal, itchy, nose/cheek region	Hives, scabies, shingles
3.	Male, old, brown, personal, non itchy, nose/cheek region	Age spot, moles
4.	Male, old, red, personal, itchy, feet	Athletes foot, hives, scabies
5.	Male, infant, red, family, itchy, nose	Eczema
6.	Female, young, brown, personal,	Melasma, moles,

No.	Kriteria Pengujian (Gender, Age, Color, History, Itchiness, Site)	Kandidat Jenis Penyakit
	non itchy, nose/cheek region	warts
7.	Male, old, red, personal, non itchy, nose/cheek region	Moles, pityriasis rosea, psoriasis, razor bump
8.	Male, young, brown, family, non itchy, nose/cheek region	Seborrheic keratoses
9.	Male, young, brown, personal, non itchy, neck region	Moles, skin tags

Kriteria pengujian dan kandidat jenis penyakit yang dipakai untuk pengujian dapat dilihat pada tabel II. Flowchart proses pencocokan yang dipakai pada pengujian dapat dilihat pada gambar 1.

TABEL III
HASIL PENGUJIAN DENGAN KRITERIA 1

No.	Citra	Jenis Penyakit Berdasarkan Basis Data Citra	Jenis Penyakit Hasil Diagnosis
1.	Acne_1.jpg	Acne	Acne
2.	Acne_2.jpg	Acne	Acne
3.	Acne_3.jpg	Acne	Acne
4.	Acne_4.jpg	Acne	Acne

No.	Citra	Jenis Penyakit Berdasarkan Basis Data Citra	Jenis Penyakit Hasil Diagnosis
5.	Acne_5.jpg	Acne	Acne
6.	Acne_6.jpg	Acne	Acne
7.	Cold_sores_1.jpg	Cold sores	Cold sores
8.	Cold_sores_2.jpg	Cold sores	Cold sores
9.	Cold_sores_3.jpg	Cold sores	Cold sores
10.	Cold_sores_4.jpg	Cold sores	Cold sores
11.	Cold_sores_5.jpg	Cold sores	Cold sores
12.	Moles_1.jpg	Moles	Moles
13.	Moles_2.jpg	Moles	Moles
14.	Moles_3.jpg	Moles	Moles
15.	Moles_4.jpg	Moles	Moles
16.	Moles_5.jpg	Moles	Moles
17.	Moles_6.jpg	Moles	Moles
18.	Moles_7.jpg	Moles	Moles
19.	Psoriasis_1.jpg	Psoriasis	Psoriasis
20.	Psoriasis_2.jpg	Psoriasis	Psoriasis
21.	Psoriasis_3.jpg	Psoriasis	Psoriasis
22.	Psoriasis_4.jpg	Psoriasis	Psoriasis
23.	Psoriasis_5.jpg	Psoriasis	Psoriasis
24.	Psoriasis_6.jpg	Psoriasis	Psoriasis
25.	Razor_bump_1.jpg	Razor bump	Razor bump
26.	Razor_bump_2.jpg	Razor bump	Razor bump
27.	Razor_bump_3.jpg	Razor bump	Razor bump
28.	Razor_bump_4.jpg	Razor bump	Razor bump
29.	Razor_bump_5.jpg	Razor bump	Razor bump

TABEL IV
HASIL PENGUJIAN DENGAN KRITERIA 2

No.	Citra	Jenis Penyakit Berdasarkan Basis Data Citra	Jenis Penyakit Hasil Diagnosis
1.	Hives_1.jpg	Hives	Hives
2.	Hives_2.jpg	Hives	Hives
3.	Hives_5.jpg	Hives	Hives
4.	Scabies_1.jpg	Scabies	Scabies
5.	Scabies_2.jpg	Scabies	Scabies
6.	Scabies_3.jpg	Scabies	Scabies
7.	Scabies_4.jpg	Scabies	Scabies
8.	Scabies_5.jpg	Scabies	Scabies
9.	Shingles_1.jpg	Shingles	Shingles
10.	Shingles_2.jpg	Shingles	Shingles
11.	Shingles_3.jpg	Shingles	Shingles
12.	Shingles_4.jpg	Shingles	Shingles
13.	Shingles_5.jpg	Shingles	Shingles

TABEL V
HASIL PENGUJIAN DENGAN KRITERIA 3

No.	Citra	Jenis Penyakit Berdasarkan Basis Data Citra	Jenis Penyakit Hasil Diagnosis
1.	Age_spot_1.jpg	Age spot	Age spot
2.	Age_spot_2.jpg	Age spot	Age spot
3.	Age_spot_5.jpg	Age spot	Age spot

No.	Citra	Jenis Penyakit Berdasarkan Basis Data Citra	Jenis Penyakit Hasil Diagnosis
4.	Moles_1.jpg	Moles	Moles
5.	Moles_2.jpg	Moles	Moles
6.	Moles_3.jpg	Moles	Moles
7.	Moles_4.jpg	Moles	Moles
8.	Moles_5.jpg	Moles	Moles
9.	Moles_6.jpg	Moles	Moles
10.	Moles_7.jpg	Moles	Moles

TABEL VI
HASIL PENGUJIAN DENGAN KRITERIA 4

No.	Citra	Jenis Penyakit Berdasarkan Basis Data Citra	Jenis Penyakit Hasil Diagnosis
1.	Athletes_foot_1.jpg	Athletes foot	Athletes foot
2.	Athletes_foot_2.jpg	Athletes foot	Athletes foot
3.	Athletes_foot_3.jpg	Athletes foot	Athletes foot
4.	Athletes_foot_4.jpg	Athletes foot	Athletes foot
5.	Athletes_foot_5.jpg	Athletes foot	Athletes foot
6.	Hives_1.jpg	Hives	Hives
7.	Hives_2.jpg	Hives	Hives
8.	Hives_5.jpg	Hives	Hives
9.	Scabies_1.jpg	Scabies	Scabies
10.	Scabies_2.jpg	Scabies	Scabies
11.	Scabies_3.jpg	Scabies	Scabies
12.	Scabies_4.jpg	Scabies	Scabies
13.	Scabies_5.jpg	Scabies	Scabies

TABEL VII
HASIL PENGUJIAN DENGAN KRITERIA 5

No.	Citra	Jenis Penyakit Berdasarkan Basis Data Citra	Jenis Penyakit Hasil Diagnosis
1.	Eczema_1.jpg	Eczema	Eczema
2.	Eczema_2.jpg	Eczema	Eczema
3.	Eczema_3.jpg	Eczema	Eczema
4.	Eczema_4.jpg	Eczema	Eczema
5.	Eczema_5.jpg	Eczema	Eczema

TABEL VIII
HASIL PENGUJIAN DENGAN KRITERIA 6

No.	Citra	Jenis Penyakit Berdasarkan Basis Data Citra	Jenis Penyakit Hasil Diagnosis
1.	Melasma_1.jpg	Melasma	Melasma
2.	Melasma_2.jpg	Melasma	Melasma
3.	Melasma_3.jpg	Melasma	Melasma
4.	Melasma_4.jpg	Melasma	Melasma
5.	Melasma_5.jpg	Melasma	Melasma

No.	Citra	Jenis Penyakit Berdasarkan Basis Data Citra	Jenis Penyakit Hasil Diagnosis
6.	Moles_1.jpg	Moles	Moles
7.	Moles_2.jpg	Moles	Moles
8.	Moles_3.jpg	Moles	Moles
9.	Moles_4.jpg	Moles	Moles
10.	Moles_5.jpg	Moles	Moles
11.	Moles_6.jpg	Moles	Moles
12.	Moles_7.jpg	Moles	Moles
13.	Warts_1.jpg	Warts	Warts
14.	Warts_2.jpg	Warts	Warts
15.	Warts_3.jpg	Warts	Warts
16.	Warts_4.jpg	Warts	Warts
17.	Warts_5.jpg	Warts	Warts
18.	Warts_6.jpg	Warts	Warts

TABEL IX
HASIL PENGUJIAN DENGAN KRITERIA 7

No.	Citra	Jenis Penyakit Berdasarkan Basis Data Citra	Jenis Penyakit Hasil Diagnosis
1.	Moles_1.jpg	Moles	Moles
2.	Moles_2.jpg	Moles	Moles
3.	Moles_3.jpg	Moles	Moles
4.	Moles_4.jpg	Moles	Moles
5.	Moles_5.jpg	Moles	Moles
6.	Moles_6.jpg	Moles	Moles
7.	Moles_7.jpg	Moles	Moles
8.	Pityriasis_rosea_1.jpg	Pityriasis rosea	Pityriasis rosea
9.	Pityriasis_rosea_2.jpg	Pityriasis rosea	Pityriasis rosea
10.	Pityriasis_rosea_6.jpg	Pityriasis rosea	Pityriasis rosea
11.	Psoriasis_1.jpg	Psoriasis	Psoriasis
12.	Psoriasis_2.jpg	Psoriasis	Psoriasis
13.	Psoriasis_3.jpg	Psoriasis	Psoriasis
14.	Psoriasis_4.jpg	Psoriasis	Psoriasis
15.	Psoriasis_5.jpg	Psoriasis	Psoriasis
16.	Psoriasis_6.jpg	Psoriasis	Psoriasis
17.	Razor_bump_1.jpg	Razor bump	Razor bump
18.	Razor_bump_2.jpg	Razor bump	Razor bump
19.	Razor_bump_3.jpg	Razor bump	Razor bump
20.	Razor_bump_4.jpg	Razor bump	Razor bump
21.	Razor_bump_5.jpg	Razor bump	Razor bump

TABEL X
HASIL PENGUJIAN DENGAN KRITERIA 8

No.	Citra	Jenis Penyakit Berdasarkan Basis Data Citra	Jenis Penyakit Hasil Diagnosis
1.	Seborrheic_keratosis_1.jpg	Seborrheic keratosis	Seborrheic keratosis
2.	Seborrheic_keratosis_2.jpg	Seborrheic	Seborrheic

No.	Citra	Jenis Penyakit Berdasarkan Basis Data Citra	Jenis Penyakit Hasil Diagnosis
		keratosis	keratosis
3.	Seborrheic_keratosis_3.jpg	Seborrheic keratosis	Seborrheic keratosis
4.	Seborrheic_keratosis_4.jpg	Seborrheic keratosis	Seborrheic keratosis
5.	Seborrheic_keratosis_5.jpg	Seborrheic keratosis	Seborrheic keratosis
6.	Seborrheic_keratosis_6.jpg	Seborrheic keratosis	Seborrheic keratosis

TABEL XI
HASIL PENGUJIAN DENGAN KRITERIA 9

No.	Citra	Jenis Penyakit Berdasarkan Basis Data Citra	Jenis Penyakit Hasil Diagnosis
1.	Moles_1.jpg	Moles	Moles
2.	Moles_2.jpg	Moles	Moles
3.	Moles_3.jpg	Moles	Moles
4.	Moles_4.jpg	Moles	Moles
5.	Moles_5.jpg	Moles	Moles
6.	Moles_6.jpg	Moles	Moles
7.	Moles_7.jpg	Moles	Moles
8.	Skin_tags_1.jpg	Skin tags	Skin tags
9.	Skin_tags_2.jpg	Skin tags	Skin tags
10.	Skin_tags_3.jpg	Skin tags	Skin tags
11.	Skin_tags_4.jpg	Skin tags	Skin tags
12.	Skin_tags_5.jpg	Skin tags	Skin tags
13.	Skin_tags_6.jpg	Skin tags	Skin tags

Tabel III sampai dengan XI masing-masing merupakan pengujian untuk 9 kriteria yang ditampilkan pada tabel II. Pada tabel-tabel tersebut, kolom “Jenis Penyakit Berdasarkan Basis Data Citra” merupakan jenis penyakit yang sebenarnya sesuai dengan citra yang diuji, dan kolom “Jenis Penyakit Hasil Diagnosis” merupakan jenis penyakit berdasarkan hasil diagnosis menggunakan aplikasi yang dibuat. Berdasarkan hasil pengujian pada tabel-tabel tersebut didapatkan bahwa perangkat lunak dapat berfungsi dengan baik, terlihat dari kesesuaian hasil diagnosis program dengan jenis penyakit yang sebenarnya.

Eksperimen bertujuan menguji kinerja algoritma SIFT dan algoritma pencocokan. Eksperimen dilakukan untuk citra dengan jumlah *keypoint* yang dihasilkan sama dengan 400, pada kasus ini adalah citra *acne_2.jpg*, *cold_sores_1.jpg*, *moles_6.jpg*, *psoriasis_1.jpg*, dan *razor_bump_3.jpg*. Eksperimen dilakukan dengan kriteria pengujian 1. Ada tiga eksperimen yang dilakukan, yaitu rotasi ke kanan, pembesaran, dan perubahan jenis pencocokan.

Jenis pencocokan menggunakan algoritma FLANN dengan menggunakan kasus eksperimen 2 pembesaran. FLANN (*Fast Library for Approximate Nearest Neighbors*) adalah library yang terdiri dari kumpulan algoritma pencarian tetangga terdekat tercepat (*fast nearest neighbor search*) dalam sekumpulan besar dataset dan fitur dengan dimensi yang besar (*high dimensional features*) [15].

TABEL XII
HASIL EKSPERIMEN ROTASI KE KANAN

No.	Citra	Besar Rotasi	Jenis Penyakit Berdasarkan Basis Data Citra	Jenis Penyakit Hasil Diagnosis
1.	Acne_2.jpg	1°	Acne	Acne
2.	Cold_sores_1.jpg	1°	Cold sores	Moles
3.	Moles_6.jpg	1°	Moles	Moles
4.	Psoriasis_1.jpg	1°	Psoriasis	Psoriasis
5.	Razor_bump_3.jpg	1°	Razor bump	Razor bump
6.	Acne_2.jpg	3°	Acne	Acne
7.	Cold_sores_1.jpg	3°	Cold sores	Moles
8.	Moles_6.jpg	3°	Moles	Moles
9.	Psoriasis_1.jpg	3°	Psoriasis	Psoriasis
10.	Razor_bump_3.jpg	3°	Razor bump	Razor bump
11.	Acne_2.jpg	7°	Acne	Acne
12.	Cold_sores_1.jpg	7°	Cold sores	Moles
13.	Moles_6.jpg	7°	Moles	Moles
14.	Psoriasis_1.jpg	7°	Psoriasis	Psoriasis
15.	Razor_bump_3.jpg	7°	Razor bump	Razor bump
16.	Acne_2.jpg	15°	Acne	Acne
17.	Cold_sores_1.jpg	15°	Cold sores	Razor bump
18.	Moles_6.jpg	15°	Moles	Moles
19.	Psoriasis_1.jpg	15°	Psoriasis	Psoriasis
20.	Razor_bump_3.jpg	15°	Razor bump	Razor bump

TABEL XIII
HASIL EKSPERIMEN PEMBESARAN

No.	Citra	Besar Pembesaran	Jenis Penyakit Berdasarkan Basis Data Citra	Jenis Penyakit Hasil Diagnosis
1.	Acne_2.jpg	1%	Acne	Acne
2.	Cold_sores_1.jpg	1%	Cold sores	Cold sores
3.	Moles_6.jpg	1%	Moles	Moles
4.	Psoriasis_1.jpg	1%	Psoriasis	Psoriasis
5.	Razor_bump_3.jpg	1%	Razor bump	Razor bump
6.	Acne_2.jpg	2%	Acne	Moles
7.	Cold_sores_1.jpg	2%	Cold sores	Cold

No.	Citra	Besar Pembesaran	Jenis Penyakit Berdasarkan Basis Data Citra	Jenis Penyakit Hasil Diagnosis
				sores
8.	Moles_6.jpg	2%	Moles	Moles
9.	Psoriasis_1.jpg	2%	Psoriasis	Psoriasis
10.	Razor_bump_3.jpg	2%	Razor bump	Razor bump
11.	Acne_2.jpg	4%	Acne	Moles
12.	Cold_sores_1.jpg	4%	Cold sores	Cold sores
13.	Moles_6.jpg	4%	Moles	Moles
14.	Psoriasis_1.jpg	4%	Psoriasis	Psoriasis
15.	Razor_bump_3.jpg	4%	Razor bump	Razor bump
16.	Acne_2.jpg	8%	Acne	Moles
17.	Cold_sores_1.jpg	8%	Cold sores	Moles
18.	Moles_6.jpg	8%	Moles	Moles
19.	Psoriasis_1.jpg	8%	Psoriasis	Psoriasis
20.	Razor_bump_3.jpg	8%	Razor bump	Razor bump

TABEL XIV
HASIL EKSPERIMEN JENIS PENCOCOKAN

No.	Citra	Besar Pembesaran	Jenis Penyakit Berdasarkan Basis Data Citra	Jenis Penyakit Hasil Diagnosis dengan Jenis Pencocokan Brute Force	Jenis Penyakit Hasil Diagnosis dengan Jenis Pencocokan FLANN
1.	Acne_2.jpg	1%	Acne	Acne	Acne
2.	Cold_sores_1.jpg	1%	Cold sores	Cold sores	Cold sores
3.	Moles_6.jpg	1%	Moles	Moles	Moles
4.	Psoriasis_1.jpg	1%	Psoriasis	Psoriasis	Psoriasis
5.	Razor_bump_3.jpg	1%	Razor bump	Razor bump	Razor bump
6.	Acne_2.jpg	2%	Acne	Moles	Moles
7.	Cold_sores_1.jpg	2%	Cold sores	Cold sores	Cold sores
8.	Moles_6.jpg	2%	Moles	Moles	Moles
9.	Psoriasis_1.jpg	2%	Psoriasis	Psoriasis	Psoriasis
10.	Razor_bump_3.jpg	2%	Razor bump	Razor bump	Razor bump
11.	Acne_2.jpg	4%	Acne	Moles	Moles
12.	Cold_sores_1.jpg	4%	Cold sores	Cold sores	Cold sores
13.	Moles_6.jpg	4%	Moles	Moles	Moles
14.	Psoriasis_1.jpg	4%	Psoriasis	Psoriasis	Psoriasis

No.	Citra	Besar Pembesaran	Jenis Penyakit Berdasarkan Basis Data Citra	Jenis Penyakit Hasil Diagnosis dg Jenis Pencocokan Brute Force	Jenis Penyakit Hasil Diagnosis dg Jenis Pencocokan FLANN
15	Razor_bump_3.jpg	4%	Razor bump	Razor bump	Razor bump
16	Acne_2.jpg	8%	Acne	Moles	Moles
17	Cold_sores_1.jpg	8%	Cold sores	Moles	Moles
18	Moles_6.jpg	8%	Moles	Moles	Moles
19	Psoriasis_1.jpg	8%	Psoriasis	Psoriasis	Psoriasis
20	Razor_bump_3.jpg	8%	Razor bump	Razor bump	Razor bump

Dari hasil eksperimen pada tabel XII didapatkan bahwa rotasi mempengaruhi jenis penyakit hasil diagnosis. Dari hasil eksperimen pada tabel XIII didapatkan bahwa pembesaran 1% tidak mempengaruhi jenis penyakit hasil diagnosis. Dari hasil eksperimen pada tabel XIV didapatkan jenis pencocokan tidak mempengaruhi jenis penyakit hasil diagnosis.

VI. KESIMPULAN DAN SARAN

Kesimpulan dari penelitian ini adalah sebagai berikut:

1. Prosedur yang perlu dilakukan untuk mendapatkan citra digital (*image acquisition*) adalah :
 - a. Pengguna menekan tombol Camera
 - b. Sistem menampilkan form hasil *capture* dari kamera
 - c. Pengguna menekan tombol 's'
 - d. Sistem menyimpan citra
2. Prosedur manipulasi (*pre-processing*) yang harus dilakukan pada citra digital adalah :
 - a. Pengurangan noise (*noise reduction*) dengan menggunakan filter Gaussian
 - b. Pemecahan (*split*) citra warna ke dalam masing-masing *color channel* (merah, hijau, dan biru) 8 bit
3. Basis data citra untuk penyakit kulit berhasil dibangun. Basis data ini berisi pengetahuan tentang penyakit kulit dan citra hasil deteksi tepi.
4. Penelitian ini menghasilkan perangkat lunak sistem pakar diagnosis penyakit kulit. Perangkat lunak menerima input berupa citra digital hasil deteksi tepi dari penyakit kulit dan informasi mengenai penderita penyakit kulit seperti jenis kelamin, usia, warna penyakit kulit, sejarah

penyakit dalam keluarga, tingkat kegalatan, dan lokasi terjadinya penyakit kulit. Perangkat lunak menghasilkan output berupa jenis penyakit kulit. Algoritma *Forward Chaining* dipakai untuk inferensi informasi mengenai penderita penyakit kulit. Algoritma deteksi tepi Canny, algoritma SIFT, dan metode pencocokan *Brute Force* dipakai untuk mengenali jenis penyakit berdasarkan input citra digital.

5. Berdasarkan hasil eksperimen pertama, rotasi mempengaruhi jenis penyakit hasil diagnosis
6. Berdasarkan hasil eksperimen kedua, pembesaran 1% tidak mempengaruhi jenis penyakit hasil diagnosis
7. Berdasarkan hasil eksperimen ketiga, jenis pencocokan tidak mempengaruhi jenis penyakit hasil diagnosis

Saran untuk pengembangan pada tahap selanjutnya adalah:

1. Penggunaan kamera yang memiliki resolusi yang lebih baik akan menghasilkan citra yang memiliki ketajaman yang lebih baik.
2. Jenis penyakit yang lebih mementingkan perubahan warna dibandingkan bentuk/*feature* seperti Age / Liver Spot, Hives, Razor Bump, dan Rosacea dihilangkan dari basis data citra.
3. Penggunaan metode pencocokan yang mampu mendeteksi *key point interest*.

DAFTAR PUSTAKA

- [1] Forsyth, David A. & Ponce, Jean., *Computer Vision: A Modern Approach*, Prentice Hall, 2nd ed., 2011.
- [2] Szeliski, Richard. *Computer Vision : Algorithms and Applications*, Springer, 2011
- [3] Mitra, Anal Kumar & Parekh, DR. Ranjan. *Automated Detection of Skin Diseases Using Texture Features*, International Journal of Engineering Science and Technology, 2011
- [4] Bin, Li & Samiei Yeganeh, Mehdi., *Comparison for Image Edge Detection Algorithms*, IOSR Journal of Computer Engineering, 2012
- [5] Burger, Wilhelm & Burge, Mark J., *Principles of Digital Image Processing: Fundamental Techniques (Undergraduate Topics in Computer Science)*, Springer, 2011.
- [6] Burger, Wilhelm & Burge, Mark J., *Principles of Digital Image Processing: Core Algorithms (Undergraduate Topics in Computer Science)*, Springer, 2009.
- [7] Treiber, Marco Alexander, *An Introduction to Object Recognition: Selected Algorithms for a Wide Variety of Applications (Advances in Computer Vision and Pattern Recognition)*, Springer, 2010.
- [8] Nixon, Mark, *Feature Extraction and Image Processing for Computer Vision*, 3rd Edition, 2012
- [9] Garcia, Gloria Bueno, et.al., *Learning Image Processing with OpenCV*, Packt Publishing, 2015.
- [10] Baggio, Daniel Lélis, et. al., *Mastering OpenCV with Practical Computer Vision Projects*, Packt Publishing, 2012.
- [11] Lowe, David G., *Distinctive Image Features from Scale-Invariant Keypoints*, International Journal of Computer Vision, 2004
- [12] Bastos, Rafael, FIRST - Fast Invariant to Rotation and Scale Transform: Invariant Image Features for Augmented Reality and Computer Vision, VDM Verlag, 2009.

- [13] Razeghi, Orod, dkk. *Skin Lesion Image Recognition with Computer Vision and Human in the Loop*, 2012
- [14] Laganiere, Robert, *OpenCV 2 Computer Vision Application Programming Cookbook*, 2011
- [15] Muja, Marius & Lowe, David G. *Fast Approximate Nearest Neighbors With Automatic Algorithm Configuration*, 2009