



ITS
Institut
Teknologi
Sepuluh Nopember

TUGAS AKHIR - EC 184801

**PERBANDINGAN METODE SCALE INVARIANT FEATURE
TRANSFORM (SIFT) DAN SPEEDED-UP ROBUST
FEATURE (SURF) DALAM MENDETEKSI KEPALSUAN
CITRA DERMOSCOPIK YANG TERSERANG COPY-MOVE**

Muhammad Rizki
NRP 0721144000052

Dosen Pembimbing
Dr. Supeno Mardi Susiki Nugroho, S.T., M.T.
Dr. I Ketut Eddy Purnama, S.T., M.T.

DEPARTEMEN TEKNIK KOMPUTER
Fakultas Teknologi Elektro
Institut Teknologi Sepuluh Nopember
Surabaya 20190



TUGAS AKHIR - EC 184801

**PERBANDINGAN METODE SCALE INVARIANT FEATURE
TRANSFORM (SIFT) DAN SPEEDED-UP ROBUST FEATURE
(SURF) DALAM MENDETEKSI KEPALSUAN CITRA
DERMOSCOPIK YANG TERSERANG COPY-MOVE**

Muhammad Rizki
NRP 0721144000052

Dosen Pembimbing
Dr. Supeno Mardi Susiki Nugroho, S.T., M.T.
Dr. I Ketut Eddy Purnama, S.T., M.T.

Departemen Teknik Komputer
Fakultas Teknologi Elektro
Institut Teknologi Sepuluh Nopember
Surabaya 2019

LEMBAR PENGESAHAN

Perbandingan Metode *Scale Invariant Feature Transform (SIFT)* dan *Speeded-Up Robust Feature (SURF)* dalam Mendeteksi Kepalsuan Citra *Dermoscopic* yang Terserang Copy-Move.

Tugas Akhir ini disusun untuk memenuhi salah satu syarat memperoleh gelar Sarjana Teknik di Institut Teknologi Sepuluh Nopember Surabaya

Oleh : Muhammad Rizki (NRP: 0721144000052)

Tanggal Ujian : 6 Juli 2020

Periode Wisuda : September 2020

Disetujui oleh:

Dr. Supeno Mardi Susiki Nugroho, ST., MT. (Pembimbing I)
NIP. 19700313 199512 1 001

Dr. I Ketut Eddy Purnama, ST., MT. (Pembimbing II)
NIP. 19690730 199512 1 001

Ahmad Zaini, ST., M.Sc. (Penguji I)
NIP. 19750419 200212 1 003

Arief Kurniawan, ST., MT. (Penguji II)
NIP. 19740907 200212 1 001

Dr. Eko Mulyanto Yuniarno, ST., MT. (Penguji III)
NIP. 19680601 199512 1 009



Mengetahui
Kepala Departemen Teknik Komputer

Dr. Supeno Mardi Susiki Nugroho, ST., MT.
NIP. 19700313 199512 1 001

[Halaman ini sengaja dikosongkan].

ABSTRAK

Nama Mahasiswa : Muhammad Rizki
Judul Tugas Akhir : Perbandingan Metode Scale Invariant Feature Transform (SIFT) dan Speeded-Up Robust Feature (SURF) Dalam Mendeteksi Kepalsuan Citra *Dermoscopic* yang Terserang *Copy-Move*
Dosen Pembimbing : 1. Dr. Supeno Mardi Susiki Nugroho, S.T., M.T.
2. Dr. I Ketut Eddy Purnama, S.T., M.T.

Citra forensik atau Digital forensik merupakan area penting dalam penelitian untuk mengetahui apakah sebuah citra yang merupakan jejak digital adalah citra yang original atau hasil manipulasi. Suatu jejak digital sekarang ini merupakan hal yang penting dalam penilaian untuk suatu keputusan. Banyak sekali citra-citra yang ada sekarang dengan mudahnya di manipulasi atau dirusak dengan menggunakan beberapa software seperti *Photoshop*. Cukup sulit untuk membedakan atau menentukan keaslian dari citra. Untuk analisis forensik, teknik pendeteksian pemalsuan citra digunakan untuk mengidentifikasi keaslian citra. Dalam tugas akhir ini metode Scale Invariant Feature Transform (SIFT) dan Speeded-up Robust Feature (SURF) digunakan untuk mendeteksi citra yang telah di *Copy-Move*. Tujuannya adalah untuk membandingkan dan menentukan mana metode terbaik untuk mendeteksi citra *dermoscopic* palsu yang diserang *copy-move*. Dikhususkan untuk dataset yang digunakan adalah citra di bidang *dermoscopic* kulit karena belum adanya pengamanan citra dalam bidang tersebut. Diharapkan dengan adanya perbandingan metode deteksi citra ini dapat memudahkan untuk melakukan pengecekan kepalsuan suatu citra dan mengurangi atau menekan jumlah citra palsu yang beredar di dunia digital agar tidak disalah gunakan.

Kata Kunci: Kata kunci: Deteksi Copy-Move, Digital Forensik, Image Processing, SIFT, SURF

Halaman ini sengaja dikosongkan

ABSTRACT

Name : Muhammad Rizki
Title : *Comparison of Scale Invariant Feature Transform (SIFT) and Speeded-up Robust Feature (SURF) Method on Detecting Copy-Moved Dermoscopic Image Forgery*
Advisors : 1. Dr. Supeno Mardi Susiki Nugroho, S.T., M.T.
2. Dr. I Ketut Eddy Purnama, S.T., M.T.

Forensic imagery or digital forensics is an important area in research to find out whether an image that is a digital trace is an original or manipulated image. A digital footprint is now an important thing in judgment for a decision. Lots of images now are easily manipulated or forged by using some software such as Photoshop. It is quite difficult to distinguish or determine the authenticity of the image. For forensic analysis, an image forgery detection technique is used to identify the authenticity of the image. In this final project, the Scale Invariant Feature Transform (SIFT) and Speeded-up Robust Feature (SURF) methods are used to detect images that have been copy-moved. The aim is to compare and determine which is the best method for detecting fake dermoscopic images that are attacked by copy-move. The dataset used is an image in the dermo-scopic area of the skin because there is no image protection in that field. It is hoped that this comparison of image detection methods can make it easier to check the falsity of an image and reduce or reduce the number of fake images circulating in the digital world so that they are not misused.

Keywords: Copy-Move, Digital Forensics, Image Processing, SIFT, SURF

Halaman ini sengaja dikosongkan

KATA PENGANTAR

Puji dan syukur penulis panjatkan kehadiran Allah SWT atas rahmat dan hidayah-Nya, penulis dapat menyelesaikan tugas akhir ini dengan judul **Perbandingan Metode Scale Invariant Feature Transform (SIFT) dan Speeded-Up Robust Feature (SURF) dalam Mendeteksi Kepalsuan Citra Dermoscopic yang Terserang Copy-Move.**

Tugas akhir ini disusun untuk memenuhi persyaratan menyelesaikan pendidikan Sarjana pada Bidang Studi Telematika, Departemen Teknik Komputer, Institut Teknologi Sepuluh Nopember. Tugas akhir ini dapat selesai tidak lepas dari bantuan berbagai pihak. Pada kesempatan ini, penulis mengucapkan terima kasih kepada:

1. Bapak, Wisnoe Satrijono, Ibu, Nur Mardiati, dan adik Safira Afifah Putri serta keluarga yang telah memberi bantuan doa, materi, tenaga, dan waktu.
2. Bapak Dr. Supeno Mardi Susiki Nugroho, S.T., M.T. dan Bapak Dr. I Ketut Eddy Purnama, S.T., M.T. selaku dosen pembimbing yang meluangkan banyak waktu untuk berdiskusi, memberi semangat, arahan dan nasehat selama proses pengerjaan tugas akhir ini.
3. Bapak dan Ibu dosen Departemen Teknik Komputer ITS atas waktu dan tenaga mendidik dan memberi perhatian kepada penulis selama menjalani masa perkuliahan.
4. Teman-teman asisten Laboratorium Komputasi Multimedia B401, Laboratorium Telematika B201, dan Laboratorium Visi Komputer yang telah banyak membantu dalam proses pengerjaan tugas akhir ini.
5. Keluarga besar e-54 atas semangat dan motivasi selama masa pendidikan di Departemen Teknik Komputer ITS.
6. Adik-adik e-55, e-56, dan e-57 yang telah membantu dan memberi dukungan langsung maupun tidak langsung.
7. Semua pihak yang telah membantu penulis dalam pengerjaan tugas akhir ini dan tidak dapat penulis sebutkan satu-persatu.

Penulis menyadari bahwa tugas akhir ini masih memiliki banyak kekurangan, oleh karena itu saran dan masukan sangat diharapkan untuk perbaikan. Semoga tugas akhir ini bermanfaat bagi pembaca dan masyarakat pada umumnya.

Surabaya, Juli 2020

Penulis

Halaman ini sengaja dikosongkan

DAFTAR ISI

ABSTRAK.....	i
<i>ABSTRACT</i>	iii
KATA PENGANTAR	v
DAFTAR ISI.....	vii
DAFTAR GAMBAR	x
DAFTAR TABEL.....	xii
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Permasalahan	2
1.3 Tujuan.....	2
1.4 Batasan Masalah.....	2
1.5 Sistematika Penulisan	3
BAB 2 TINJAUAN PUSTAKA	5
2.1 Citra Digital.....	5
2.1.1 Pengolahan Citra Digital (<i>Image Processing</i>)	6
2.1.2 Citra <i>Dermoscopic</i>	7
2.2 Forensik Digital.....	8
2.2.1 Pemalsuan Citra (<i>Image Forgery</i>).....	9
2.2.2 <i>Copy-Move</i>	11
2.3 <i>Computer Vision</i>	12
2.4 Ekstraksi Fitur	12
2.5 <i>Scale Invariant Feature Transform (SIFT)</i>	13
2.5.1 Pencarian Nilai Ekstrem pada Skala Ruang	14
2.5.2 Pencarian <i>Keypoint</i>	16
2.5.3 Penentuan Orientasi.....	19
2.5.4 Deskriptor <i>Keypoint</i>	19
2.6 <i>Speeded-Up Robust Feature</i>	20
2.6.1 Pendeteksian Titik Perhatian (<i>Interest Point</i>).....	21
2.6.2 Representasi Ruang Skala	22
2.6.3 Deskripsi Fitur	23
2.6.4 Penyesuaian dan Pengenalan Fitur	24

2.7	Anaconda	26
2.8	OpenCV	27
2.9	Python	27
2.9.1	Numerical Python (Numpy)	28
2.10	Adobe Photoshop	29
BAB 3 DESAIN DAN IMPLEMENTASI		31
3.1	Desain Sistem	31
3.2	Alur Kerja	32
3.3	Dataset MNIST HAM10000	32
3.4	Dataset DERMNET	33
3.5	<i>Pre-Processing</i>	34
3.5.1	Citra <i>Dermoscopic</i> Asli	35
3.5.2	<i>Copy-Move</i> Citra	35
3.5.3	Citra <i>Dermoscopic</i> Termanipulasi	36
3.6	Pengolahan Citra	36
3.7	Data Keluaran	37
3.8	<i>Scale Invariant Feature Transform</i> (SIFT)	38
3.9	<i>Speeded-Up Robust Feature</i> (SURF)	40
3.10	Metodologi <i>Matching Keypoint</i>	41
BAB 4 PENGUJIAN DAN ANALISA		45
4.1	Spesifikasi Perangkat	45
4.2	Pengujian menggunakan citra <i>copy-move</i> MNIST HAM10000	45
4.3	Pengujian menggunakan citra <i>copy-move</i> Acne DERMNET..	47
4.4	Pengujian menggunakan citra <i>copy-move</i> Atopic Dermatitis DERMNET	47
4.5	Pengujian menggunakan citra <i>copy-move</i> Furuncle DERMNET	48
4.6	Pengujian menggunakan citra <i>copy-move</i> Psoriasis DERMNET	49
4.7	Pengujian menggunakan citra <i>copy-move</i> Ringworm DERMNET	50
4.8	Analisa perbandingan hasil pengujian.....	51

BAB 5 PENUTUP	55
5.1 Kesimpulan.....	55
5.2 Saran.....	55
DAFTAR PUSTAKA	57
BIOGRAFI PENULIS	59

DAFTAR GAMBAR

Gambar 2.1 Permodelan citra digital dalam bentuk matriks berukuran $N \times M$	6
Gambar 2.2 Porongan matriks angka dari citra seekor kucing	7
Gambar 2.3 Proses pengolahan citra.....	8
Gambar 2.4 Contoh citra <i>dermoscopic</i>	8
Gambar 2.5 Contoh jenis citra yang dimanipulasi dengan <i>splicing</i>	10
Gambar 2.6 Contoh jenis citra yang dimanipulasi dengan <i>retouching</i> ..	11
Gambar 2.7 Contoh <i>copy-move</i> pada citra	12
Gambar 2.8 Contoh ekstraksi fitur pada gambar sebuah motor	13
Gambar 2.9 Contoh hasil ekstraksi fitur menggunakan metode SIFT ..	14
Gambar 2.10 Ilustrasi pembentukan DoG (Difference of Gaussian)	16
Gambar 2.11 Ilustrasi proses pencarian nilai ekstrem	16
Gambar 2.12 Lokasi <i>keypoint</i> awal.....	17
Gambar 2.13 Lokasi <i>keypoint</i> akhir	18
Gambar 2.14 <i>keypoint descriptor</i>	19
Gambar 2.15 Contoh hasil ekstraksi fitur menggunakan metode SURF20	
Gambar 2.16 Representasi Ruang Skala	22
Gambar 2.17 <i>Sliding Orientation</i> setiap 60 derajat.....	23
Gambar 2.18 Deskripsi Fitur	23
Gambar 2.19 Penyesuaian Fitur.....	24
Gambar 2.20 Contoh tampilan dari <i>Anaconda Prompt</i>	25
Gambar 2.21 Tampilan utama dari Adobe Photoshop CS6	28
Gambar 3.1 Diagram alir sistem	29
Gambar 3.2 Sample Dataset HAM10000	31
Gambar 3.3 Sample dataset DERMNET	33
Gambar 3.4 Diagram alir tahap pre-processing	34
Gambar 3.5 Citra <i>dermoscopic</i> asli yang belum termanipulasi	35
Gambar 3.6 Citra <i>dermoscopic</i> yang sudah dimanipulasi <i>copy-move</i> ..	36
Gambar 3.7 Diagram Alir metodologi pengolahan citra.....	37
Gambar 3.8 Contoh hasil output citra menggunakan metode SIFT	38
Gambar 3.9 Contoh hasil output citra menggunakan metode SURF	38
Gambar 3.10 Diagram alir algoritma SIFT	39
Gambar 3.11 Diagram alir algoritma SURF	40
Gambar 3.12 Hasil dari ekstraksi <i>keypoint descriptor</i>	42
Gambar 3.13 Hasil dari <i>keypoint</i> yang berhasil <i>match</i>	43
Gambar 4.1 Grafik rata-rata prosentase kecocokkan pada seluruh pengujian citra <i>dermoscopic</i>	52

Gambar 4.2 Grafik rata-rata <i>running time</i> pada seluruh pengujian citra <i>dermoscopic</i> (dalam satuan sekon)	46
--	----

DAFTAR TABEL

Tabel 4.1 Spesifikasi Perangkat untuk Pengujian.....	45
Tabel 4.2 Hasil perbandingan metode SIFT dan SURF pada citra <i>dermo-</i> <i>scopic</i> dari MNIST HAM10000	46
Tabel 4.3 Hasil perbandingan metode SIFT dan SURF pada citra <i>dermo-</i> <i>scopic</i> berjenis Acne yang didapat dari DERMNET	47
Tabel 4.4 Hasil perbandingan metode SIFT dan SURF pada citra <i>dermo-</i> <i>scopic</i> berjenis Atopic Dermatitis yang didapat dari DERM- NET	48
Tabel 4.5 Hasil perbandingan metode SIFT dan SURF pada citra <i>dermo-</i> <i>scopic</i> jenis Furuncle yang didapat dari DERMNET	49
Tabel 4.6 Hasil perbandingan metode SIFT dan SURF pada citra <i>dermo-</i> <i>scopic</i> jenis Psoriasis yang didapat dari DERMNET	50
Tabel 4.7 Hasil perbandingan metode SIFT dan SURF pada citra <i>dermo-</i> <i>scopic</i> jenis Ringworm yang didapat dari DERMNET	51

NOMENKLATUR

D	: Jarak antar titik
x, y, σ	: Posisi pixel pada sebuah sumbu
l	: Citra masukan
∂	: Diferensial parsial
x bar	: Offset point
T	: Periode atau satuan waktu
H	: Matriks Hessian
Det	: Determinan
Tr	: Treshold keypoint
r	: Rasio dari threshold
m	: Besar gradien
L	: Citra gaussian yang dihaluskan
tan	: Skala citra
$G(x)$: Distribusi Gaussian 1D
π	: Nilai Pi
e	: Bilangan alami
v	: <i>Vector</i>
d	: <i>Respon gelombang Haar</i>
dx, dy	: Respon gelombang haar untuk sumbu x dan y
f, f_*	: Vektor f dan f_*
\in	: Elemen dari vektor
\forall	: Untuk semua elemen dari suatu vector
argmin	: Argumen minimum dari sebuah vector
CRelative	: Nilai jarak terdekat

Halaman ini sengaja dikosongkan

BAB I

PENDAHULUAN

Penelitian ini di latar belakang oleh berbagai kondisi yang menjadi acuan. Selain itu juga terdapat beberapa permasalahan yang akan dijawab sebagai luaran dari penelitian.

1.1 Latar Belakang

Saat ini citra digital digunakan secara luas di berbagai bidang dalam kehidupan kita sehari-hari. Bidang-bidang penting tersebut seperti laporan berita, ilmu forensik, layanan pengawasan, pemasaran online, dan diagnosis medis. Selain itu, mereka dapat digunakan sebagai bukti di persidangan pengadilan dan di pers untuk menyesuaikan arti sebuah citra untuk mempengaruhi sudut pandang seseorang. Pada tahun 2018 saja, menurut data dari Kominfo, terdapat sekitar 800.000 situs penyebar informasi palsu [1].

Orisinalitas gambar forensik ini telah menjadi bidang penting penelitian untuk bisa mendapatkan kembali keaslian pada suatu citra digital. Analisis forensik untuk gambar digital membantu memberikan informasi pendukung kepada penegak hukum, keamanan, dan badan intelijen [2]. Berbagai teknik dapat digunakan untuk memeriksa dan melegitimasi suatu konten citra digital.

Ada beberapa program pengeditan atau pemalsuan citra yang cukup mudah digunakan di zaman sekarang ini seperti contohnya *Photoshop*. Selain itu, seorang pemalsu yang pintar dapat menggunakan berbagai metode untuk mengolah citra palsu dengan cara *copy-move* citra dan mencocokkannya dengan citra yang sama [3], penyambungan citra menggunakan beragam bagian dari citra yang dimanipulasi[4] dan *re-touching* citra [5] dapat meninggalkan modifikasi yang baik. Dalam pemisahan citra ini, ada pengoptimalan area dari berbagai citra untuk membuat citra palsu.

Karena elemen yang mudah digunakan mirip dengan bagian yang disalin, seperti warna dan noise suatu gambar, perlu dan penting untuk membedakan area yang dimanipulasi dari citra sebenarnya. Selain itu, untuk menghilangkan jejak visual pemalsuan citra pemalsu menggunakan berbagai prosedur pasca pemrosesan seperti *blurring*, *edge smoothing*, dan *noise addition*.

Pada tugas akhir ini berkonsentrasi pada deteksi pemalsuan suatu citra yang di duplikasi dengan menggunakan metode Scale Invariant Feature Transform (SIFT) dan Speeded-up Robust Feature (SURF) untuk membandingkan metode pengecekan suatu citra yang sudah dipalsukan dengan cara copy-move, khususnya pada citra *dermoscopic*. Dengan adanya tugas akhir ini Penulis berharap dapat mengetahui metode terbaik dalam pengecekan suatu citra yang dipalsukan dan bisa memberikan saran yang mungkin bisa dipergunakan untuk penelitian selanjutnya dalam menghadapi era modern di dunia digital. Serta terciptanya sistem pendeteksian citra pada bidang dermatologi karena belum adanya sistem pendeteksian citra pada bidang tersebut.

1.2 Permasalahan

Dari latar belakang tersebut, maka masalah yang dapat diambil adalah tidak adanya pengamanan citra *dermoscopic* terkait copy-move pada sistem dermatologi. Oleh karena itu, diperlukan metode pengamanan citra *dermoscopic* dan dicari metode terbaik untuk mendeteksi citra *dermoscopic* yang di *copy-move*.

1.3 Tujuan

Tujuan penelitian tugas akhir ini dilakukan untuk membandingkan implementasi pendeteksian kepalsuan sebuah jejak digital atau citra yang menggunakan Copy-Move dengan berbasis metode Scale Invariant Feature Transform (SIFT) dan Speeded-up Robust Feature (SURF) pada citra *dermoscopic*. Diharapkan dengan adanya perbandingan metode deteksi citra ini dapat diketahui metode terbaik untuk melakukan pengecekan kepalsuan suatu citra *copy-move* dan mengurangi atau menekan jumlah citra palsu yang beredar di dunia digital agar tidak disalah gunakan.

1.4 Batasan Masalah

Batasan masalah dalam tugas akhir ini adalah:

1. Bahasa pemrograman yang digunakan untuk implementasi deteksi *copy-move* ini adalah *Python* versi 3.7.3

2. Dataset yang digunakan adalah citra digital berekstensi PNG dan JPG / JPEG.
3. Dataset merupakan beberapa citra *dermoscopic* (penyakit kulit) diambil dari DERMNET dan MNIST HAM10000 yang dirilis oleh *Harvard Dataserve* dan di publikasikan secara umum pada website *International Skin Image Collaboration (ISIC)*.
4. Ukuran citra dataset yang digunakan adalah 600 x 450 pixel.
5. Metode yang digunakan untuk mendeteksi *copy-move* adalah metode SIFT dan SURF.
6. Dataset yang digunakan untuk *copy-move* duplikasi menggunakan perangkat lunak *Adobe Photoshop CS6*.
7. Perangkat lunak yang digunakan adalah Spyder sebagai IDE, dan Anaconda sebagai kakas bantu.

1.5 Sistematika Penulisan

Laporan penelitian tugas akhir ini tersusun dalam sistematika dan terstruktur sehingga lebih mudah dipahami dan dipelajari oleh pembaca maupun seorang yang hendak melanjutkan penelitian ini. Alur sistematika penulisan laporan tugas akhir ini yaitu:

1. BAB I Pendahuluan

Bab ini berisi uraian tentang latar belakang, permasalahan, tujuan, batasan masalah dan sistematika penulisan.

2. BAB II Tinjauan Pustaka

Bab ini berisi beberapa subjudul yang terdiri atas uraian-uraian sistematis teori-teori yang berhubungan dengan permasalahan yang dibahas pada tugas akhir ini. Teori-teori yang digunakan sebagai dasar dalam tugas akhir, yaitu informasi terkait implementasi dari Algoritma SIFT dan SURF, *copy-move* dan teori-teori penunjang lainnya.

3. BAB III Desain dan Implementasi

Bab ini berisi penjelasan terkait sistem yang dibuat, digunakan desain, skenario dan alur kerja sistem untuk memudahkan ketika pengujian pada pelaksanaan tugas akhir ini.

4. BAB IV Pengujian dan Analisa

Bab ini menjelaskan tentang pengujian yang dilakukan terhadap sistem dan menganalisa sistem. Hasil pengujian dari berbagai skenario disajikan pada bab ini.

5. BAB V Penutup

Bab ini menjelaskan kesimpulan dari penelitian yang dilakukan disertai saran dan kritik yang membangun untuk pengembangan lebih lanjut.

BAB II TINJAUAN PUSTAKA

Demi mendukung penelitian dalam tugas akhir ini, dibutuhkan beberapa teori penunjang sebagai bahan acuan dan referensi. Dengan demikian penelitian ini menjadi lebih terarah. Teori-teori tersebut diantaranya ekstraksi fitur SIFT dan SURF, *image processing*, deteksi *copy-move* dan beberapa teori lain.

2.1 Citra Digital

Citra merupakan sebuah kombinasi antara titik, garis, bidang, dan warna untuk menciptakan suatu imitasi dari suatu objek. Citra dapat digolongkan menjadi dua jenis yaitu citra analog dan citra digital. Citra analog adalah citra yang dibentuk dari sinyal analog yang bersifat kontinyu, seperti contohnya adalah mata manusia. Sedangkan citra digital merupakan representatif dari citra yang diambil oleh mesin dengan bentuk pendekatan berdasarkan sampling dan kuantisasi. Sampling menyatakan besarnya kotak-kotak yang disusun dalam baris dan kolom. Dengan kata lain, sampling pada citra menyatakan besar kecilnya ukuran pixel (titik) pada citra, dan kuantisasi menyatakan besarnya nilai tingkat kecerahan yang dinyatakan dalam nilai tingkat keabuan (grayscale) sesuai dengan jumlah bit biner yang digunakan oleh mesin, dengan kata lain kuantisasi pada citra menyatakan jumlah warna yang ada pada citra.

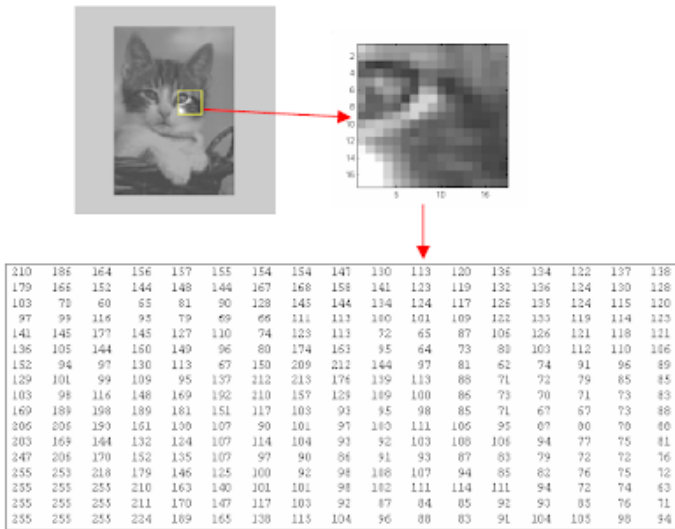
Citra digital dapat dimodelkan sebagai suatu matriks dimana indeks baris dan kolomnya menyatakan suatu titik pada citra tersebut dan elemen matrisknya menyatakan tingkat keabuan pada titik tersebut. Gambar 2.1 ini adalah pemodelan citra digital dalam bentuk matrix berukuran $N \times M$.

$$f(x,y) = \begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0,N-1) \\ f(1,0) & f(1,1) & \dots & f(1,N-1) \\ \vdots & \vdots & \vdots & \vdots \\ f(M-1,0) & f(M-1,1) & \dots & f(M-1,N-1) \end{bmatrix}$$

(Sumber: Basuki, 2005)

Gambar 2.1 Pemodelan citra digital dalam bentuk matriks berukuran $N \times M$.

Nilai pada suatu irisan antara baris dan kolom (pada posisi x,y) disebut dengan *picture element*, *image element*, dan *pixel*. Istilah pixel merupakan istilah yang sering digunakan dalam citra digital. Citra yang dilihat oleh komputer tidak sama seperti apa yang kita lihat. Pada umumnya, citra pada komputer direpresentasikan dalam bentuk matriks dan menggunakan model warna RGB. Gambar 2.2 adalah permodelan citra digital dalam bentuk matriks. [6]

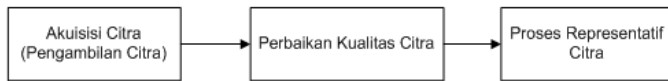


(Sumber: Basuki, 2005)

Gambar 2.2 Potongan matriks angka dari citra seekor kucing.

2.1.1 Pengolahan Citra Digital (*Image Processing*)

Image processing atau pengolahan citra adalah suatu bentuk pengolahan atau pemrosesan sinyal dengan input berupa gambar (image) dan ditransformasikan menjadi gambar lain sebagai keluarannya dengan teknik tertentu. Image processing dilakukan untuk memperbaiki kesalahan data sinyal gambar yang terjadi akibat transmisi dan selama akuisisi sinyal, serta untuk meningkatkan kualitas penampakan gambar agar lebih mudah diinterpretasi oleh sistem penglihatan manusia baik dengan melakukan manipulasi dan juga penganalisisan terhadap gambar.



(Sumber: Basuki, 2005)

Gambar 2.3 Proses pengolahan citra.

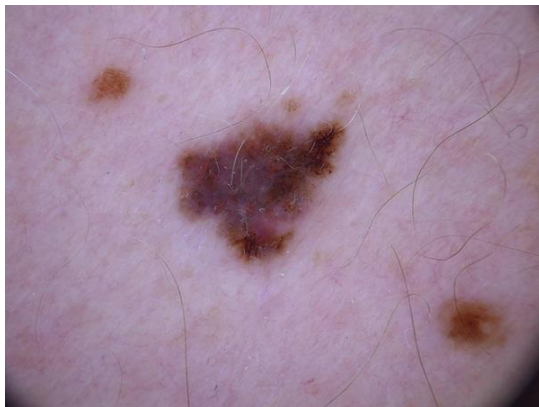
Dalam perkembangan lebih lanjut, image processing serta *computer vision* digunakan sebagai mata manusia, dengan perangkat input image capture seperti kamera dan scanner dijadikan sebagai mata dan mesin komputer (dengan program komputasinya) dijadikan sebagai otak yang mengolah informasi[6]. Ada beberapa Teknik-teknik pengolahan citra digital, diantaranya :

1. *Image Enhancement*, yaitu proses perbaikan citra dengan meningkatkan kualitas citra.
2. *Image Restoration*, yaitu proses memperbaiki model citra, biasanya ke bentuk semula atau lebih baik.
3. *Color Image Processing*, yaitu proses memperbaiki warna pada suatu citra untuk menjadi lebih baik dari sebelumnya.
4. *Image Compression*, yaitu proses yang digunakan untuk mengubah ukuran data pada citra.
5. *Segmentation*, yaitu proses untuk membedakan atau memisahkan objek-objek yang ada dalam suatu citra, seperti memisahkan objek dengan latar belakangnya.
6. *Object Recognition*, yaitu suatu proses yang dilakukan untuk mengenali objek-objek apa saja yang ada dalam suatu citra. [6]

2.1.2 Citra *Dermoscopic*

Citra *dermoscopic* merupakan sebuah citra yang digunakan untuk inspeksi atau mendiagnosis jaringan kulit abnormal (*lesion/ lesi*) tanpa terhalang oleh refleksi permukaan kulit. Citra ini dapat mendiagnosis kulit yang mempunyai pigmen jinak maupun ganas. Alat yang digunakan untuk mendiagnosis dinamakan Dermatoskop. Dermatoskop terdiri dari kaca pembesar, sumber cahaya (terpolarisasi atau non-polarisasi), piringan transparan, dan kadang-kadang media cair di antara instrumen dan kulit. Karena banyak sekali jenis penyakit yang dapat diderita oleh kulit, citra *dermoscopic* sangat diperlukan untuk meneliti jaringan lesi

kulit berpigmen. Dataset citra-citra dermoscopic ini bisa didapatkan dari MNIST HAM 10000 dan DERMNET. Gambar 2.4 adalah contoh citra *dermoscopic* yang didapat dari MNIST HAM10000.



Gambar 2.4 Contoh citra *dermoscopic* [7]

2.2 Forensik Digital

Forensik digital adalah salah satu cabang ilmu forensik, terutama untuk penyelidikan dan penemuan konten perangkat digital, dan seringkali dikaitkan dengan kejahatan komputer. Istilah forensik digital pada awalnya identik dengan forensik komputer tetapi kini telah diperluas untuk menyelidiki semua perangkat yang dapat menyimpan data digital. Forensik digital diperlukan karena biasanya data di perangkat target dikunci, dihapus, atau disembunyikan. Landasan forensik digital ialah praktik pengumpulan, analisis, dan pelaporan data digital. Investigasi forensik digital memiliki penerapan yang sangat beragam. Penggunaan paling umum adalah untuk mendukung atau menyanggah asumsi kriminal dalam pengadilan pidana atau perdata[2].

Forensik digital juga dapat dilakukan di sektor swasta; seperti penyelidikan internal perusahaan (in-house) atau penyelidikan intrusi (penyelidikan khusus mengeksplorasi sifat dan dampak intrusi jaringan yang tidak sah). Penguasaan ilmu forensik digital tidak hanya menuntut kemampuan teknis semata tetapi juga terkait dengan bidang lain, seperti bidang hukum[2]. Aspek teknis dari penyelidikan dapat dibagi menjadi beberapa subcabang, sesuai dengan jenis perangkat digital yang terlibat;

forensik komputer, forensik jaringan, analisis data forensik dan forensik peranti bergerak. Proses forensik umumnya meliputi penyitaan, forensic imaging (akuisisi) dan analisis media digital dan penyusunan laporan berdasarkan bukti yang dikumpulkan.

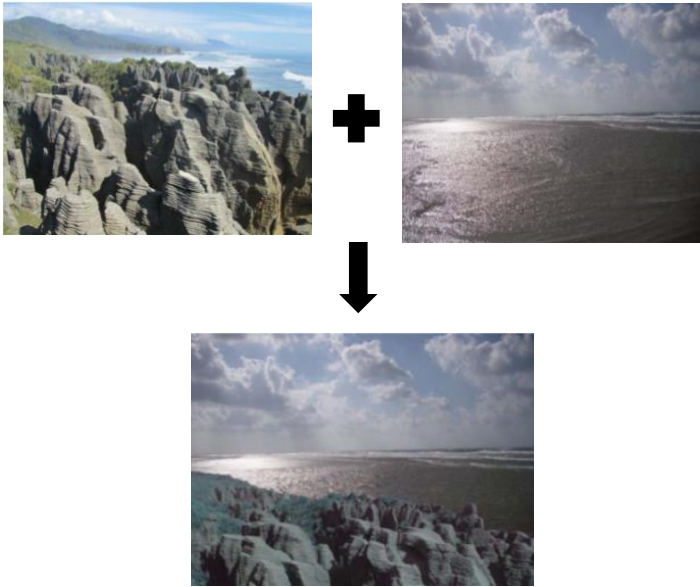
Selain mengidentifikasi bukti langsung sebuah kejahatan, forensik digital dapat digunakan untuk mengkonfirmasi hubungan antara tersangka dan kasus tertentu, mengkonfirmasi alibi-alibi atau pernyataan-pernyataannya, untuk memahami niat, mengidentifikasi sumber (misalnya, dalam kasus sengketa hak cipta), atau mengotentikasi dokumen-dokumen.

Ruang lingkup investigasi forensik digital lebih luas daripada bidang pengetahuan forensik lainnya (di mana sebagian besar ilmu forensik lain dirancang untuk menjawab pertanyaan yang relatif sederhana), sering melibatkan garis waktu atau hipotesis yang kompleks. Forensik citra digital merupakan cabang dari keamanan multimedia yang bertujuan untuk melakukan penelusuran terkait keaslian suatu citra dengan mengetahui informasi riwayatnya.

2.2.1 Pemalsuan Citra (*Image Forgery*)

Pemalsuan citra atau *image forgery* adalah sebuah manipulasi dari citra digital untuk mengganti atau menyembunyikan suatu informasi yang berguna atau penting dari citra digital tersebut[8]. Citra yang dimanipulasi ini terkadang sulit untuk ditemukan keasliannya. Kemajuan teknologi dalam bidang pengolahan citra menyebabkan bertambah canggihnya pula teknik-teknik yang dapat digunakan untuk memanipulasi suatu citra, bahkan tanpa meninggalkan bekas manipulasinya. Terdapat tiga jenis kategori pemanipulasian citra yang paling umum, yaitu *splicing*, *copy-move*, dan *retouching*.

Citra *splicing* adalah proses menggabungkan dua atau lebih citra untuk membuat suatu citra yang baru. Sebuah wilayah tertentu disalin dari satu citra dan disisipkan ke citra yang lain untuk membentuk citra baru yang berbeda. Gambar 2.5 adalah ilustrasi dari citra yang menggunakan Teknik *splicing* untuk manipulasi.



(Sumber: Ansari, 2014)

Gambar 2.5 Contoh jenis citra yang dimanipulasi dengan Teknik *splicing*.

Citra *Retouching* adalah proses mengubah piksel yang disalin sesuai dengan piksel yang ada di sekitarnya. Hal ini dapat meningkatkan atau mengurangi beberapa fitur dari citra asli tanpa mengubah arti citra yang sebenarnya. Beberapa contoh dari citra *retouching* ini adalah seperti yang ada di iklan-iklan makanan atau minuman dan biasanya terdapat pula pada majalah produk kecantikan. Gambar 2.6 merupakan contoh untuk citra yang dimanipulasi dengan cara *retouching*.

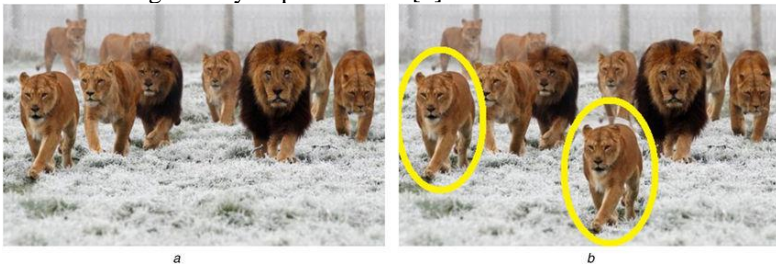


Gambar 2.6 Contoh jenis citra yang dimanipulasi dengan Teknik *retouching*.

Sedangkan untuk citra *copy-move* akan dijelaskan lebih detail pada sub-bab selanjutnya.

2.2.2 *Copy-move*

Copy-move adalah teknik memanipulasi citra yang umumnya digunakan untuk eksploitasi informasi karena kesederhanaan dan efek yang sangat visual. Juga dikenal sebagai *region duplication* atau *cloning*, *copy-move* menyalin wilayah citra dan memindahkan wilayah citra tadi ke wilayah lain dalam gambar yang sama. Teknik ini dapat digunakan untuk memberikan kesan yang palsu. Biasanya teknik *copy-move* ini dilakukan karena tidak meninggalkan jejak, atau sangat sedikit jejak. Karena alasan itu atribut dari citra yang sama, seperti iluminasi, proporsi, dan fokus sangat menyerupai citra asli[8].



Gambar 2.7 Contoh Copy-Move Citra. (A) Citra asli. (B) Citra yang telah dilakukan *copy-move* pada daerah singa yang digandakan

2.3 Computer Vision

Saat mata manusia melihat ke arah jendela di pagi hari, manusia dapat memahami dan mengenali dengan mudah apa yang sedang dilihatnya. Mulai dari cahaya matahari, rumput-rumput yang hijau, perbedaan bayangan pada benda yang membelakangi cahaya, dan segala keindahan objek yang ada di pagi hari. Kemudian saat manusia melihat sebuah potret dari sekumpulan orang, seperti foto keluarga misalnya, manusia dapat dengan mudah mengerti jumlah orang yang ada di dalam foto, mengerti latar belakang dari foto tersebut bahkan sampai memahami emosi dari setiap wajah yang ada di foto tersebut.

Lain halnya dengan komputer. Komputer tidak langsung mengenali objek apa saja yang ada pada suatu citra. Perlu adanya proses untuk mengenali objek tersebut. Inilah maksud dari *computer vision*. *Computer vision* adalah bidang ilmiah interdisipliner yang membahas bagaimana komputer dapat memperoleh pemahaman tingkat tinggi dari sebuah gambar atau video digital. Atau intinya, mengambil sebuah informasi dari suatu citra. Tugas dari Visi Komputer meliputi metode untuk memperoleh, memroses, menganalisis dan memahami gambar digital, dan ekstraksi data dimensi tinggi dari dunia nyata untuk menghasilkan informasi numerik atau simbolis, misalnya, dalam bentuk keputusan.

Pengertian dalam konteks ini berarti transformasi gambar visual (input retina) menjadi deskripsi mengenai dunia sekitar yang dapat berinteraksi dengan proses pemikiran lain dan memperoleh tindakan yang sesuai. Pemahaman gambar ini dapat dilihat sebagai penguraian informasi simbolik dari data gambar menggunakan model yang dibangun dengan bantuan geometri, fisika, statistik, dan teori pembelajaran[9].

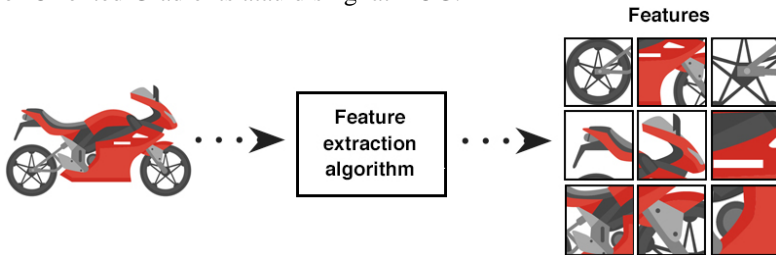
2.4 Ekstraksi Fitur

Dalam *computer vision*, Fitur adalah karakteristik atau hal-hal khusus yang mencirikan suatu objek. Fitur dibedakan menjadi dua jenis yaitu yang pertama adalah fitur alami, merupakan bagian dari gambar seperti contohnya kecerahan dan tepi objek. Sedangkan yang kedua adalah fitur buatan, merupakan fitur yang diperoleh dengan operasi tertentu pada gambar, misalnya histogram tingkat keabuan[10].

Feature Extraction atau ekstraksi fitur merupakan suatu pengambilan ciri (feature) dari suatu bentuk objek yang nantinya nilai yang didapatkan akan dianalisis untuk proses selanjutnya [11]. Ekstraksi fitur bertujuan

untuk mencari daerah fitur yang signifikan pada gambar tergantung pada karakteristik instansik dan aplikasinya. Wilayah tersebut dapat didefinisikan dalam lingkungan global atau local dan dibedakan oleh bentuk, tekstus, ukuran, intensitas, sifat statistik dan sebagainya. Ekstraksi fitur dapat dilakukan dengan cara menghitung jumlah titik atau pixel yang ditemui dalam setiap pengecekan, dimana pengecekan dilakukan dalam berbagai arah *tracing* pengecekan pada koordinat kartesian dari citra digital yang dianalisis, yaitu vertical, horizontal, diagonal kanan, dan diagonal kiri.

Ada beberapa metode untuk mengekstraksi fitur dari suatu citra, yaitu SIFT atau *Scale Invariant Feature Transform*, kemudian ada SURF atau *Speeded-Up Robust Feature*, lalu ada *Haar Cascades*, dan ada Histogram of Oriented Gradients atau disingkat HOG.



(Sumber: mc.ai)

Gambar 2.8 Contoh ekstraksi fitur pada gambar sebuah motor.

2.5 Scale Invariant Feature Transform (SIFT)

Scale-invariant feature transform (atau SIFT) adalah sebuah algoritma dalam Computer vision untuk mendeteksi dan mendeskripsikan fitur-fitur lokal dari citra[12]. Pendeteksian dan pendeskripsian fitur-fitur lokal dapat dipergunakan dalam aplikasi pengenalan obyek dan tracking. SIFT memiliki ketahanan yang kuat terhadap penskalaan, rotasi dan perubahan sudut pandang citra. SIFT juga tahan terhadap perubahan intensitas pencahayaan dan noise. SIFT mendeteksi fitur-fitur lokal yang relatif banyak dalam sebuah citra. Jika SIFT diaplikasikan untuk tracking, hal ini bisa berpengaruh terhadap waktu komputasinya. Proses yang berjalan pada metode SIFT terbagi menjadi empat tahapan, yaitu pendeteksian nilai ekstrem pada skala ruang, penentuan ciri atau titik kunci (keypoint), penentuan orientasi, dan pembangkitan keypoint descriptor yang akan dijelaskan pada bab selanjutnya.

Ada beberapa keuntungan ekstraksi fitur menggunakan metode SIFT, yang pertama adalah hasil ekstraksi fitur bersifat invariant terhadap ukuran, translasi, dan rotasi dua dimensi. Yang kedua, hasil ekstraksi fitur bersifat invariant sebagian terhadap perubahan iluminasi dan perubahan sudut pandang tiga dimensi. Yang ketiga, SIFT mampu meng-ekstrak banyak keypoint dari citra yang tipikal. Dan yang terakhir adalah hasil ekstraksi fitur dari SIFT benar-benar mencirikan secara khusus (*distinctive*).[12]



(Sumber: Lowe, 2004)

Gambar 2.9 Contoh hasil ekstraksi fitur menggunakan metode SIFT

Ada empat tahapan dalam penggunaan algoritma SIFT, yaitu:

1. Pencarian nilai ekstrem pada skala ruang.
2. Pencarian keypoint
3. Penentuan orientasi
4. *Descriptor Keypoint*

Masing-masing dari tahapan ini akan dijelaskan pada sub-bab selanjutnya. Beberapa contoh pengaplikasian algoritma SIFT biasanya untuk pengenalan objek, pendeteksian objek dan tracking. Biasanya diaplikasikan untuk peluru kendali (DSMAC), robot (sensor kamera robot), dan autonomous vehicle.

2.5.1 Pencarian Nilai Ekstrem pada Skala Ruang

Tahap pertama dari algoritma SIFT adalah Pendeteksian nilai ekstrem pada skala ruang. Tahap ini diperlukan untuk menentukan *keypoint* dari suatu citra. Nilai ekstrem ini merupakan kandidat titik kunci (*keypoint*) dari SIFT. Skala ruang dari suatu citra didefinisikan sebagai

fungsi $L(x,y,\sigma)$ yang merupakan hasil dari konvolusi antara fungsi Gaussian $G(x,y,\sigma)$ dengan citra masukan $I(x,y,\sigma)$. Setelah skala ruang dari citra didapatkan, langkah selanjutnya adalah menghitung Difference of Gaussian (DoG) dari citra tersebut menggunakan persamaan (1).

$$D(x,y,\sigma) = (G(x,y,k\sigma) - G(x,y,\sigma)) * I(x,y) \quad (1)$$

Keterangan:

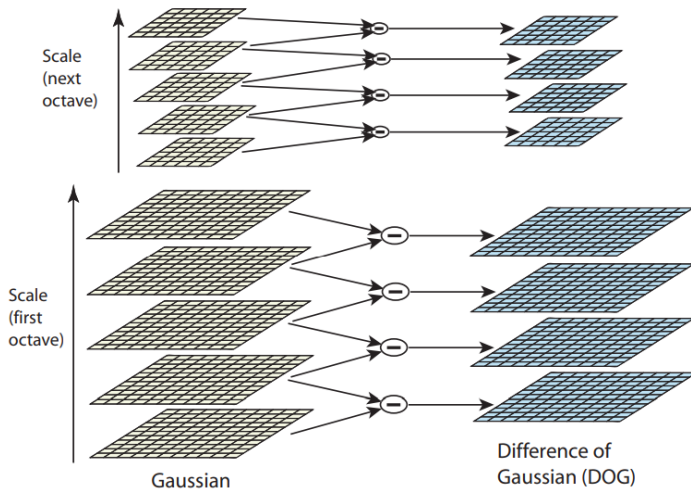
x,y, σ = titik citra

G = fungsi Gaussian

I = citra masukan

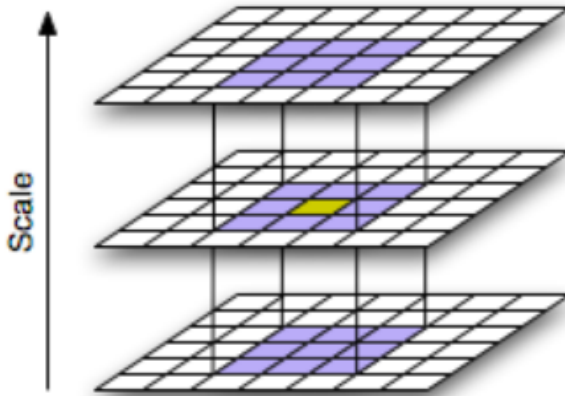
D = jarak titik

Ilustrasi dari pembentukan skala ruang ditunjukkan oleh gambar 2.10



Gambar 2.10 Ilustrasi pembentukan DoG (Difference of Gaussian)

Untuk mendeteksi nilai ekstrem pada citra DoG, nilai tiap piksel pada skala ruang DoG dibandingkan dengan delapan piksel yang berada di sekelilingnya dan sembilan piksel yang bersesuaian pada citra DoG sebelumnya dan setelahnya. Proses ini dapat dilihat pada gambar 2.11.



Gambar 2.11 Ilustrasi proses pencarian nilai ekstrem

2.5.2 Pencarian *Keypoint*

Tahap kedua dari metode SIFT adalah penentuan titik kunci atau *keypoint*. Lokasi *keypoint* didapat dengan persamaan (2).

$$\hat{x} = -\frac{\partial^2 D^{-1}}{\partial x^2} \frac{\partial D}{\partial x}. \quad (2)$$

Keterangan:

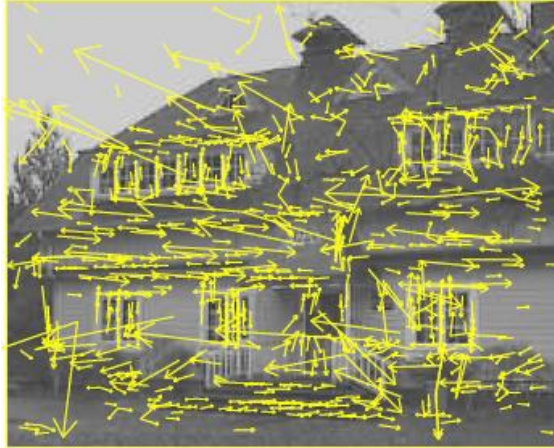
x bar = offset point

x = titik dimensional

D = jarak antar titik

∂ = diferensial parsial

Gambar 2.12 merupakan lokasi titik *keypoint* awal yang merupakan nilai ekstrem pada citra.



Gambar 2.12 Lokasi *keypoint* awal

Kemudian nilai *keypoint* didapatkan menggunakan persamaan (3)

$$D(\hat{x}) = D + \frac{1}{2} \frac{\partial D^T}{\partial x} \hat{x}. \quad (3)$$

Keterangan:

x = titik

D = jarak antar titik

x bar = offset point

T = periode atau waktu

Untuk menghilangkan *keypoint* dengan kontras rendah, nilai ekstrem $|D(\hat{x})|$ yang lebih rendah dari 0.03 dihilangkan. Untuk menghilangkan kandidat *keypoint* yang kurang jelas dan terletak di sepanjang pojokan, maka digunakan matriks *Hessian* H orde dua, seperti persamaan berikut (4).

$$H = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix} \quad (4)$$

Keterangan:

H = matriks Hessian

D_x, D_y = jarak titik

Kemudian nilai dan vektor eigen masuk dan rumusnya disederhanakan menjadi persamaan (5).

$$\frac{\text{Tr}(\mathbf{H})^2}{\text{Det}(\mathbf{H})} < \frac{(r + 1)^2}{r}. \quad (5)$$

Keterangan:

Det = determinan

H = matriks hessian

Tr = threshold

r = rasio threshold

Dengan r merupakan *threshold* dari *principal curvature* yang diizinkan. *Keypoint* yang mempunyai nilai *principal curvature* lebih besar dari $r = 10$ akan dihilangkan. Gambar 2.13 dibawah menunjukkan *keypoint* akhir setelah kandidat *keypoint* dengan kontras rendah, kurang jelas dan terletak sepanjang tepi dihilangkan.



Gambar 2.13 Lokasi *keypoint* akhir

2.5.3 Penentuan Orientasi

Tahap ketiga dari metode SIFT adalah penentuan orientasi yang bertujuan untuk mendapatkan sifat tak berubah atau disebut *Invariant* terhadap perlakuan rotasi. Penentuan orientasi dilakukan dengan persamaan (6) dan (7).

$$m(x, y) = \sqrt{L(x+1, y) - L(x-1, y) + L(x, y+1) - L(x, y-1)} \quad (6)$$

Keterangan:

m = besar gradien

L = *gaussian smoothed image*

x, y = titik

$$\theta(x, y) = \tan^{-1} \left(\frac{L(x, y+1) - L(x, y-1)}{L(x+1, y) - L(x-1, y)} \right) \quad (7)$$

Keterangan:

θ = orientasi piksel

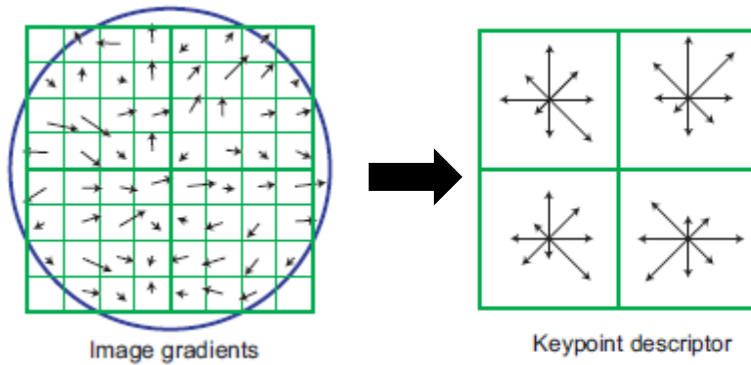
L = citra gaussian yang dihaluskan

x, y = titik citra

tan = skala citra

2.5.4 Deskriptor Keypoint

Tahap keempat dan terakhir dari metode SIFT adalah memunculkan *keypoint descriptor*. Pada tahap ini, masing-masing *keypoint* yang telah diberikan orientasi diberikan ciri khusus atau disebut *descriptor*. Proses ini bertujuan untuk mendapatkan *keypoint* yang tak berubah atau invariant terhadap perubahan intensitas cahaya atau perubahan sudut pandang 3D. *Descriptor* merupakan histogram orientasi pada piksel dengan ukuran 4x4. Histogram ini dihitung dari *magnitude* dan nilai orientasi dari sampel dalam wilayah 16x16 di sekitar *keypoint*. *Magnitude* dihitung dengan fungsi Gaussian dengan σ = satu setengah lebar *descriptor*. Hasil akhir dari tahapan ini adalah sebuah vektor dengan jumlah elemen sebanyak 4x4x8 yang hasilnya adalah 128 elemen.



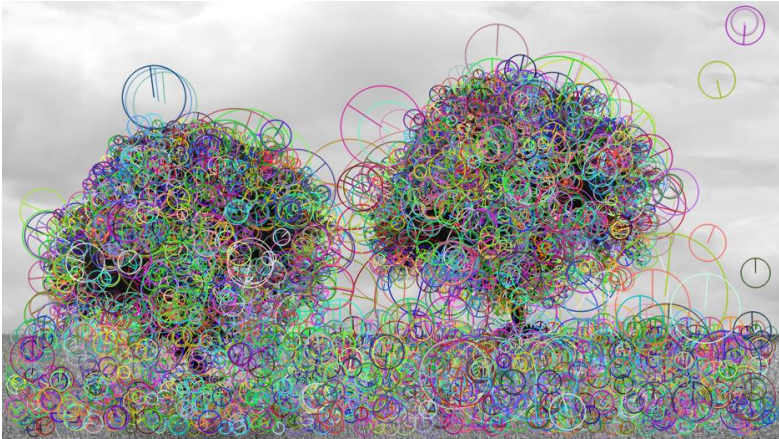
Gambar 2.14 *keypoint descriptor*

Deskriptor keypoint pada gambar 2.14 menunjukkan adanya delapan arah pada masing – masing histogram orientasi dengan panjang masing – masing anak panah sesuai dengan besar nilai dari histogram asal. Selanjutnya descriptor keypoint yang telah diperoleh akan dinormalisasi untuk mengatasi pengaruh perubahan cahaya. [12]

Keypoint ini yang kemudian menjadi fitur – fitur lokal pada suatu citra dan akan dicocokkan dengan keypoint yang terdapat pada citra lain untuk menyesuaikan dengan objek yang tersedia pada dataset. [12]

2.6 *Speeded-Up Robust Feature (SURF)*

Speeded-Up Robust Feature (atau disingkat SURF) adalah sebuah metode lanjutan dari SIFT yang telah dikembangkan dan berdasarkan proses *interest point detection* dan *interest point descriptor* dapat menyamai bahkan melebihi performa dari SIFT dalam waktu komputasi yang lebih singkat atau cepat[13]. Metode SURF ini sendiri dipublikasikan oleh peneliti dari ETH Zurich, Herbert Bay pada tahun 2006 dengan bantuan rekannya yang bernama Tinne Tuytelaars dari Katholieke Universiteit Leuven dan Luc Van Gool. Algoritma SURF menggunakan penggabungan algoritma citra integral (*integral image*) dan *blob detection* berdasarkan determinan dari matriks Hessian. Penjelasan lebih lanjut mengenai metode ini akan dijelaskan pada bab selanjutnya.



Gambar 2.15 Contoh hasil ekstraksi fitur menggunakan metode SURF pada citra *copy-move*.

Sama seperti SIFT, algoritma SURF juga mempunyai empat tahapan dalam penggunaannya, namun tahapannya berbeda. Keempat tahapan tersebut adalah:

1. Pendeteksian Titik Perhatian (*interest point*)
2. Representasi Ruang Skala
3. Deskripsi Fitur
4. Penyesuaian dan Pengenalan Fitur

2.6.1 Pendeteksian Titik Perhatian (*interest point*)

Deteksi titik perhatian (*interest point*) digunakan untuk memilih titik yang mengandung banyak informasi dan sekaligus stabil terhadap gangguan lokal atau global dalam citra digital. Dalam algoritma SURF, dipilih deteksi titik perhatian yang mempunyai sifat invariant terhadap skala, yaitu blob detection.

Blob merupakan luas pada citra digital yang memiliki sifat konstan atau bervariasi dalam kisaran tertentu. Untuk melakukan komputasi *blob detection* digunakan determinan matriks Hessian dari citra. Persamaan (8) menunjukkan perhitungan matriks Hessian ($H(x)$) pada sebuah gambar digital (I) dengan L_{xx} adalah konvolusi turunan kedua Gaussian.

$$H(x, \sigma) = \begin{bmatrix} L_{xx}(x, \sigma) & L_{xy}(x, \sigma) \\ L_{xy}(x, \sigma) & L_{yy}(x, \sigma) \end{bmatrix} \quad (8)$$

Keterangan:

H = matriks hessian

L = citra gaussian yang dihaluskan

x, y, σ = titik

Dengan fungsi Gaussian didefinisikan sebagai:

$$g(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{x^2}{2\sigma^2}} \quad (9)$$

Keterangan:

g(x) = fungsi gaussian

x, σ = titik

π = pi

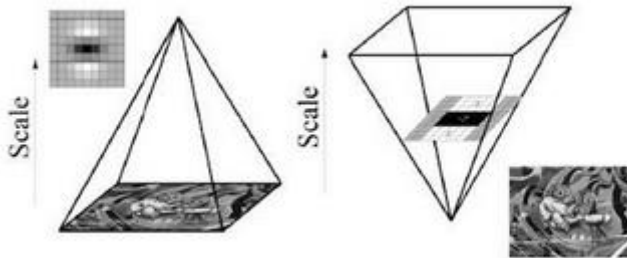
e = bilangan alami

Dalam algoritma SURF, determinan matriks Hessian dihitung dari wavelet Haar dengan menggunakan integral image secara optimal. Determinan dari matriks Hessian digunakan sebagai dasar algoritma SURF karena sifat invariant terhadap skala, kestabilan dan mudah untuk diulang. [13]

2.6.2 Representasi Ruang Skala

Dengan ukuran citra yang berbeda – beda, akan sangat sulit untuk membandingkan fitur – fitur yang terdapat pada citra. Oleh karena itu, diperlukan suatu proses yang menangani perbedaan ukuran dengan menggunakan metode perbandingan skala. Dalam metode ini digunakan scale space dimana citra diimplementasikan dalam sebuah image pyramid. Citra secara berulang mengalami proses smoothing dengan fungsi Gaussian secara beruntun dengan cara sub-sampling untuk mencapai tingkat tertinggi pada piramida. Dengan menggunakan integral image, perhitungan ini tidak perlu dilakukan secara iteratif dengan menggunakan filter yang sama, dapat dilakukan filter dengan ukuran sembarang ke dalam beberapa skala citra yang berbeda. Agar titik kunci

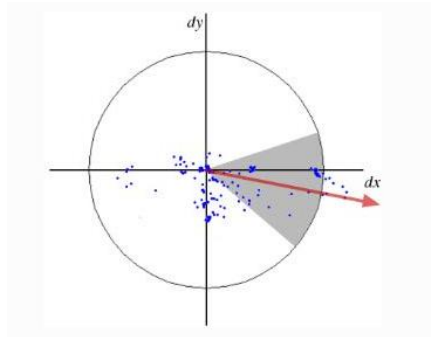
stabil di setiap level penyekalaan, akan dilakukan analisis dimana ketika level penyekalaan meningkat, ukuran citra akan bertambah, tidak mereduksi ukuran gambar. Gambar 2.16 memperlihatkan proses pelebaran filter ketika skala bertambah. [13]



Gambar 2.16 Representasi Ruang Skala.

2.6.3 Deskripsi Fitur

Untuk proses *interest point descriptor*, metode SURF menggunakan algoritma transformasi wavelet Haar tingkat 1 dalam arah vertical dan horizontal. Setelah mendapatkan respon dari transformasi ini, akan dilakukan sliding orientation sebesar 60 derajat untuk mencari arah orientasi yang paling dominan seperti terlihat pada gambar 2.17. Titik kunci, lokasi, dan orientasi dominan ini akan dimasukkan ke dalam fitur vector dengan total 64 dimensi yang akan dicari kedekatannya dengan perhitungan hasil kali titik.



Gambar 2.17 *Sliding Orientation* setiap 60 derajat.

Selanjutnya untuk deskripsi fitur dalam SURF digunakan hanya perhitungan gradient histogram dalam empat kelompok untuk mempercepat perhitungan, yang persamaannya:

$$v = (\sum d_x, \sum d_y, \sum |d_x|, \sum |d_y|) \quad (10)$$

Keterangan:

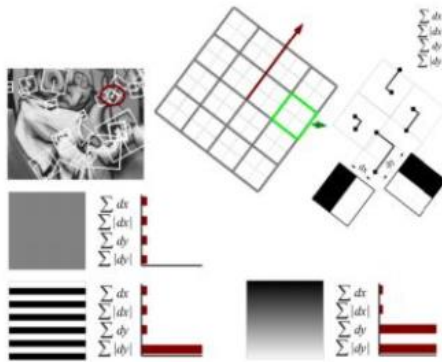
v = vector

d = respon gelombang haar

dx = respon untuk titik horizontal

dy = respon titik vertikal

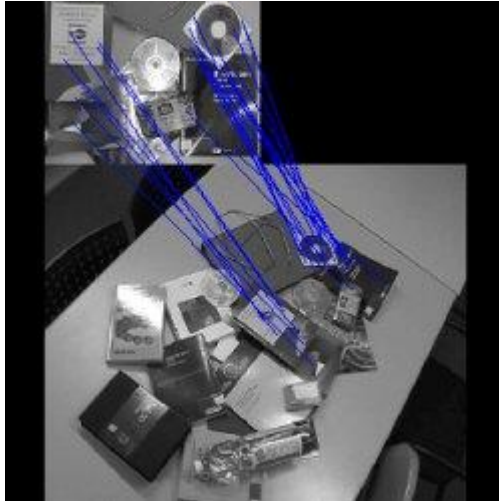
Dengan d_x adalah respon haar wavelet pada arah horizontal dan d_y pada arah vertikal. [13]



Gambar 2.18 Contoh deskripsi Fitur.

2.6.4 Penyesuaian dan Pengenalan Fitur

Dalam tahap ini dibandingkan fitur hasil perhitungan sebelumnya apabila terdapat perbedaan kontras yang dideteksi melalui tanda dari *trace* matriks Hessian. Proses *matching* dari fitur vector SURF akan dilakukan menggunakan proses hasil kali titik cosinus yang menghasilkan keluaran seperti metode tangga terdekat tetapi dengan proses waktu komputasi yang lebih cepat. Dengan kata lain, dengan cara ini biaya komputasi untuk algoritma SURF sangatlah minim.



(Sumber: <http://people.ee.ethz.ch>)

Gambar 2.19 Contoh pengaplikasian penyesuaian Fitur.

2.7 Anaconda

Anaconda adalah *open data science platform* yang dibangun menggunakan bahasa pemrograman *Python*. Versi Anaconda *open-source* menyediakan distribusi dengan performa tingkat tinggi dari bahasa pemrograman *Python* dan *R* yang berisikan lebih dari 100 paket untuk data science. Anaconda menyediakan CLI (*command line interface*) berupa perintah “**conda**” sebagai manajemen lingkungan untuk bahasa pemrograman *Python*. Perintah “conda” dapat digunakan untuk *create*, *export*, *list*, *remove* dan *update* lingkungan yang bisa digunakan untuk versi bahasa pemrograman *Python* atau paket yang dipasang didalamnya. [14]

```
Anaconda Prompt
(base) C:\Users\rwizk>conda
usage: conda-script.py [-h] [-V] command ...

conda is a tool for managing and deploying applications, environments and packages.

Options:
positional arguments:
  command
  clean                Remove unused packages and caches.
  config              Modify configuration values in .condarc. This is modeled
                    after the git config command. Writes to the user .condarc
                    file (C:\Users\rwizk\.condarc) by default.
  create              Create a new conda environment from a list of specified
                    packages.
  help                Displays a list of available conda commands and their help
                    strings.
```

Gambar 2.20 Contoh tampilan dari *Anaconda Prompt*.

2.8 OpenCV

OpenCV adalah kependekan dari Computer Vision, maksudnya adalah komputer yang digunakan untuk mengolah image (citra/gambar) yang ditangkap oleh alat perekam (seperti kamera atau webcam) yang dikonversi dari analog ke digital lalu diolah di dalam komputer. Tujuan pengolahan image ini ada banyak, di antaranya untuk memperbaiki kualitas gambar atau untuk indentifikasi citra. OpenCV menggunakan *library* gratis yang dapat digunakan di berbagai *platform*, seperti GNU/Linux maupun Windows. *OpenCV* mulanya ditulis dalam bahasa pemrograman C++, namun saat ini *OpenCV* dapat digunakan pada berbagai bahasa seperti *Python*, *Java*, atau *MATLAB*. *OpenCV* memiliki struktur ypong modular, yang berarti memiliki package yang mencakup beberapa library static dan bersama. Beberapa modul tersebut diantaranya adalah *core*, *img proc*, *video*, *HighGUI*, dan masih banyak lagi.

2.9 Python

Python adalah bahasa pemrograman yang mudah dipelajari, memiliki struktur data tingkat tinggi yang efisien dan pendekatan yang sederhana namun efektif untuk pemrograman berorientasi objek (*object-oriented programming*). Sintaksi python yang elegan dan pengetahuan yang dinamis, menjadikan bahasa Python ideal untuk *scripting* dan pengembangan aplikasi yang cepat di banyak area pada sebagian besar platform. Dengan menggunakan Python dapat mempelajari beberapa aspek *object-oriented programming* tanpa perlu memahami banyak hal detail yang rumit, hal ini memungkinkan untuk berkonsentrasi pada apa yang ingin dicapai tanpa terjebak pada detail.

Salah satu keunggulan Python dibanding Java, tidak membutuhkan pemahaman mendalam beberapa konsep *object-oriented programming* untuk menulis program. Python juga memiliki mode interaktif untuk memudahkan tanpa siklus edit-kompilasi-eksekusi seperti Java dan C++. Selain itu, mode non-interaktif pada Python juga memungkinkan untuk menjalankan *script* tanpa mengetik ulang setiap saat. Kombinasi mode interaktif dan non-interaktif tersedia dan berguna menghemat waktu karena dapat melakukan percobaan eksekusi selagi melakukan perbaikan *script*. Seperti Java, Python memiliki *small compact core* dan *library* yang besar serta dapat diperluas [15]. Jadi, yang diperlukan hanyalah memasang *library* dan mengimpor sesuai dengan kebutuhan.

2.9.1 Numerical Python (NumPy)

NumPy (kependekan dari Numerical Python) adalah salah satu library teratas yang dilengkapi dengan sumber daya yang berguna untuk membantu para data scientist mengubah Python menjadi alat analisis dan pemodelan ilmiah yang kuat. Library Open source terpopuler ini tersedia di bawah lisensi BSD (*The Berkeley Software Distribution*). Ini adalah pustaka Python dasar untuk melakukan tugas dalam komputasi ilmiah. NumPy adalah bagian dari ekosistem berbasis Python yang lebih besar dari tool open source yang disebut SciPy.

Perpustakaan memberdayakan Python dengan struktur data substansial untuk mudah melakukan perhitungan multi-dimensi (multi-dimensional arrays) dan perhitungan matriks [16]. Selain penggunaannya dalam menyelesaikan persamaan aljabar linier (linear algebra equations) dan perhitungan matematis lainnya, NumPy juga digunakan sebagai wadah multi-dimensi serbaguna untuk berbagai jenis data generik.

Lebih hebatnya, NumPy terintegrasi dengan bahasa pemrograman lain seperti C / C ++ dan Fortran. Fleksibilitas perpustakaan NumPy memungkinkannya untuk dengan mudah dan cepat bergabung dengan berbagai database dan tools. Ada beberapa perbedaan antara array pada NumPy dan standard *Python sequence* seperti:

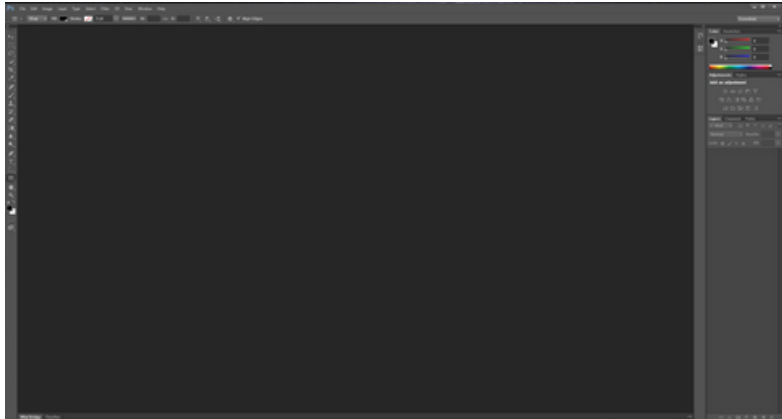
1. Array NumPy memiliki ukuran tetap tidak seperti Python yang tumbuh secara dinamis. Perubahan ndarray akan membuat array baru dan menghapus yang asli.
2. Elemen-elemen dalam array Numpy harus dari tipe data yang sama dengan demikian akan memiliki ukuran yang sama dalam memori. Selain itu, array dari objek (Python, NumPy) dapat

- memungkinkan dari berbagai elemen dengan ukuran yang berbeda.
3. Array NumPy memfasilitasi matematika tingkat lanjut dan jenis operasi lain dengan sejumlah besar data. Operasi-operasi semacam itu dijalankan lebih efisien dan dengan baris kode lebih sedikit daripada dilakukan menggunakan sequence bawaan Python.

2.10 Adobe Photoshop

Adobe Photoshop, atau biasa disebut Photoshop, adalah perangkat lunak editor citra atau gambar yang diciptakan oleh Adobe Systems yang dikhususkan untuk pengeditan foto/gambar dan penambahan efek. Perangkat lunak ini banyak digunakan oleh fotografer digital dan perusahaan iklan sehingga dianggap sebagai pemimpin pasar (market leader) untuk perangkat lunak pengolah gambar/foto, dan, bersama Adobe Acrobat, dianggap sebagai produk terbaik yang pernah diproduksi oleh Adobe Systems.

Ada banyak versi dari Adobe Photoshop ini, versi pertama kali adalah versi sebelum adanya brand CS (Creative Suite) dan masih minim fitur. Versi selanjutnya dari CS adalah CS2, CS3, CS4, CS5, dan CS6 dengan fitur dan tools yang lebih baik dan baru pada versi versi selanjutnya. Versi terbaru yang dikeluarkan oleh Adobe adalah Adobe Photoshop CC (singkatan dari Creative Cloud) sebagai seri lanjutan dari CS. Photoshop tersedia untuk Microsoft Windows, Mac OS X, dan Mac OS, sedangkan versi CS2 ke atas juga dapat digunakan oleh sistem operasi lain seperti Linux dengan bantuan perangkat lunak tertentu seperti CrossOver[17]. Saat ini Photoshop merupakan perangkat lunak terbaik jika di bandingkan dengan perangkat lunak lain sejenisnya. Dapat di katakan juga hampir tidak ada tandingannya. Format File Photoshop mampu untuk membaca dan menulis gambar berformat raster dan vektor seperti .png, .gif, .jpeg, dan lain-lain. Ada banyak sekali fitur dan tools yang terdapat pada adobe photoshop seperti *pen tool* yang dapat digunakan untuk menandai suatu citra secara *freehand*, kemudian ada *clone stamp tool* yaitu untuk menduplikasi bagian tertentu yang sudah di potong sebelumnya, *cropping*, *slicing*, dan masih banyak lagi fitur-fiturnya.



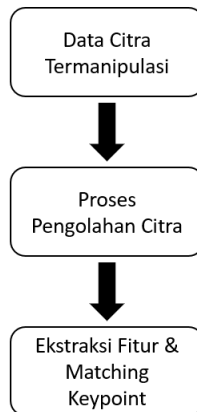
Gambar 2.21 Tampilan utama dari Adobe Photoshop CS6.

BAB III DESAIN DAN IMPLEMENTASI

Tugas akhir ini dilaksanakan sesuai dengan rancangan sistem berikut dengan implementasinya. Rancangan sistem merupakan konsep yang tersusun secara urut dan sistematis dari pembuatan dan perakitan infrastruktur sehingga memudahkan pengerjaan, kemudian diwujudkan dalam bentuk blok-blok alur yang harus dikerjakan. Sistem perangkat lunak yang dibuat pada tugas akhir ini adalah untuk mendeteksi *copy-move* pada citra digital *dermoscopic* dengan metode *Scale Invariant Feature Transform* (SIFT) dan *Speeded-up Robust Feature* (SURF).

3.1 Desain Sistem

Tugas akhir ini merupakan salah satu penerapan dari pengolahan citra (*image processing*) yang bertujuan untuk membandingkan metode pendeteksian mana yang terbaik yang dapat digunakan untuk pengecekan suatu citra *dermoscopic* yang termanipulasi dengan cara *copy-move*. Secara umum, proses yang dilalui dari citra yang termanipulasi meliputi beberapa proses hingga hasil akhirnya adalah hasil matching keypoint dimana hasil matching tersebut terdapat waktu komputasi dan prosentase kecocokkan keypoint. Pada gambar 3.1 ditampilkan diagram alir sistem.



Gambar 3.1 Diagram alir sistem

Data citra termanipulasi awalnya adalah citra penyakit kulit *dermoscopic* asli lalu kemudian melalui tahap *pre-processing* dan hasil akhirnya adalah citra *dermoscopic* tersebut menjadi termanipulasi dengan cara *copy-move*. Kemudian tahapan selanjutnya adalah proses pengolahan citra. Proses pengolahan citra ini menggunakan algoritma SIFT dan SURF untuk mengekstraksi fitur atau ciri dari citra yang sudah terserang *copy-move*. Pada tahap akhirnya, akan muncul hasil ekstraksi fitur berupa *keypoint*, yang kemudian *keypoint* tersebut akan dicocokkan (*matching*). Setelah *keypoint* yang diharapkan *match*, barulah bisa dihitung prosentase kecocokkan pada citra dan waktu komputasi yang dihabiskan oleh masing-masing algoritma dalam memproses citra termanipulasi tersebut.

3.2 Alur Kerja

Pengerjaan tugas akhir ini dibagi menjadi beberapa tahap berdasarkan metodologi penelitian, yaitu:

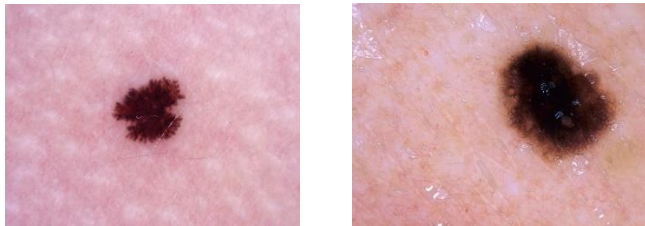
1. Mendapatkan dataset berupa citra *dermoscopic*.
2. Memanipulasi citra *dermoscopic*.
3. Pengolahan citra termanipulasi menggunakan algoritma SIFT & SURF.
4. Menganalisa hasil akhir berupa prosentase *matching* dan waktu komputasi.

3.3 Dataset MNIST HAM10000

Dataset merupakan kumpulan data yang akan digunakan untuk dilakukan proses penelitian. Data yang akan digunakan ini sendiri merupakan citra *dermoscopic* atau penyakit kulit asli yang kemudian nanti akan dilakukan tahap *pre-processing* berupa *copy-move*. Untuk menjalankan tugas akhir ini diperlukan dataset citra *dermoscopic* yang otentik dan beresolusi cukup tinggi agar mempermudah penelitian.

Dataset MNIST HAM10000 merupakan kumpulan citra *dermoscopic* lesi dari tujuh kelas jenis penyakit kulit. MNIST merupakan kepanjangan dari Modified National Institute of Standards and Technology. MNIST adalah bentuk database yang biasa digunakan untuk beberapa pengujian *image processing*. Sedangkan HAM10000 maksudnya adalah *Human Against Machine* dengan 10000 contoh citra

hasil uji. Dataset ini berasal dari citra *dermoscopic* pasien yang dirujuk kepusat rujukan tersier yang dikhususkan untuk deteksi dini melanoma pada kelompok beresiko tinggi. Citra *dermoscopic* diambil menggunakan sebuah kamera mikroskop yang khusus untuk pengambilan citra kulit terpolarisasi dan non-polarisasi. Lebih dari 95% lesi yang dianalisis selama praktik klinis akan masuk dalam salah satu dari tujuh kategori diagnosis. Dalam praktiknya, tugas dokter adalah untuk membedakan anatara lesi ganas dan jinak, juga untuk membuat diagnosis spesifik, misalnya melanoma dan karsinoma sel basal yang mempunyai metode pengobatan dan rentang waktu yang berbeda. Gambar 3.2 merupakan lesi citra *dermoscopic*.

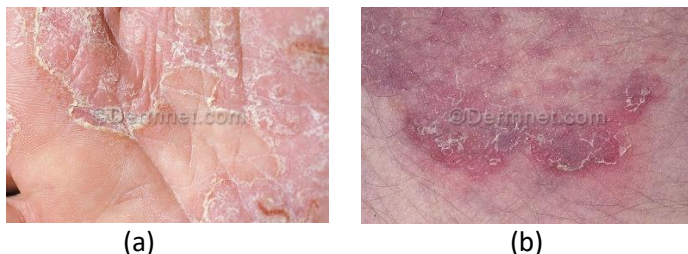


Gambar 3.2 Sample dataset HAM10000

Dataset citra *dermoscopic* MNIST HAM10000 yang digunakan diambil secara manual, dengan mengumpulkan gambar penyakit kulit yang ada pada website resmi. Setelah keseluruhan dataset terkumpul, langkah selanjutnya adalah melakukan pengecekan ulang terhadap gambar yang akan digunakan untuk proses selanjutnya. Ukuran dari dataset MNIST HAM10000 yang digunakan untuk penelitian ini adalah 600 x 450 pixel.

3.4 Dataset DERMNET

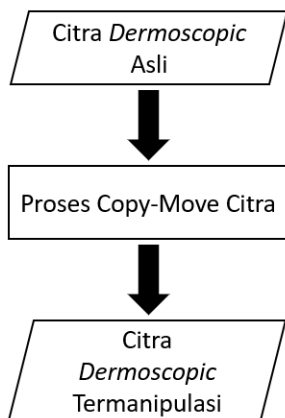
Pada tugas akhir ini juga, akan digunakan beberapa dataset yang diambil dari DERMNET. Dermnet merupakan dataset publik yang memiliki lebih dari 23000 gambar penyakit kulit yang terdiri lebih dari 600 kelas penyakit kulit. Dari dataset Dermnet ini diambil beberapa gambar dari 5 jenis penyakit kulit yang umum terjadi pada negara tropis. Contoh gambar dataset DERMNET bisa dilihat pada gambar 3.3.



Gambar 3.3 Sample dataset DERMNET. (a) penyakit kulit jenis *atopic dermatitis* dan (b) jenis *ringworm*

3.5 Pre-Processing

Tahap *pre-processing* merupakan tahap awal pada citra yang bertujuan untuk mendapatkan data masukan berupa citra agar dapat diolah lebih lanjut. Pada *pre-processing* akan memunculkan kualitas citra masukan yang diinginkan sehingga dapat terekstraksi dengan baik. Pada gambar 3.2 adalah gambar diagram alir untuk tahap *pre-processing* pada tugas akhir ini. Citra asli akan dilakukan penyerangan *copy-move* tanpa adanya proses *resize* dan *rotate*.

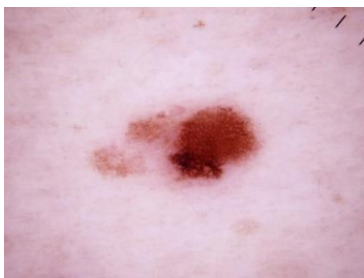


Gambar 3.4 Diagram alir tahap *pre-processing*

Dari diagram alir tersebut dapat dilihat bahwa pada tahap pre-processing perlu dilakukan beberapa tahapan untuk mendapatkan dataset masukan yaitu citra *dermoscopic* yang termanipulasi menggunakan *copy-move*.

3.5.1 Citra Dermoscopic Asli

Data masukan adalah data awal yang akan dimasukkan ke dalam perangkat lunak dan di uji. Dataset citra yang digunakan adalah dataset citra *dermoscopic* yang didapatkan dari dari MNIST HAM10000 yang merupakan kumpulan citra dari tujuh kelas jenis penyakit kulit dan juga DERMNET. Citra dataset ini yang nantinya akan dilakukan teknik manipulasi *copy-move* menggunakan Adobe Photoshop dengan salah satu fiturnya yaitu *stamp tool*. Gambar 3.5 adalah citra *dermoscopic* asli dari MNIST HAM10000.



Gambar 3.5 Citra *dermoscopic* asli yang belum termanipulasi.

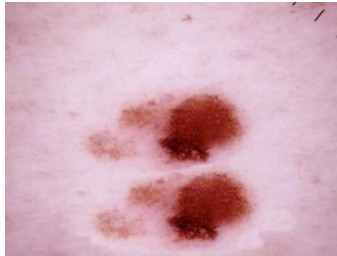
3.5.2 Copy-Move Citra

Dalam tahap ini, citra *dermoscopic* yang sudah didownload akan melalui proses manipulasi *copy-move* pada aplikasi Adobe Photoshop CS6. Citra asli akan dimanipulasi menggunakan *stamp tool* yang terdapat pada Photoshop. Seluruh data masukan yang akan diuji akan melalui tahap yang sama agar menjaga kualitas citra. Berikut adalah Langkah-langkah yang dilakukan untuk menduplikasi citra asli:

1. Citra asli di cap menggunakan *stamp tool* yang ada di photoshop.

2. Letakkan *stamp tool* yang sudah di cap citra asli pada bagian selain citra asli untuk mendapatkan citra yang palsu.
3. Arsir dan penuh seluruh bagian citra palsu hingga menyerupai citra yang asli.

Hasil akhir dalam tahapan ini adalah mendapatkan citra palsu yang sudah termanipulasi menggunakan teknik *copy-move*. Gambar 3.4 menunjukkan citra MNISTHAM 10000 yang sudah di *copy-move*.



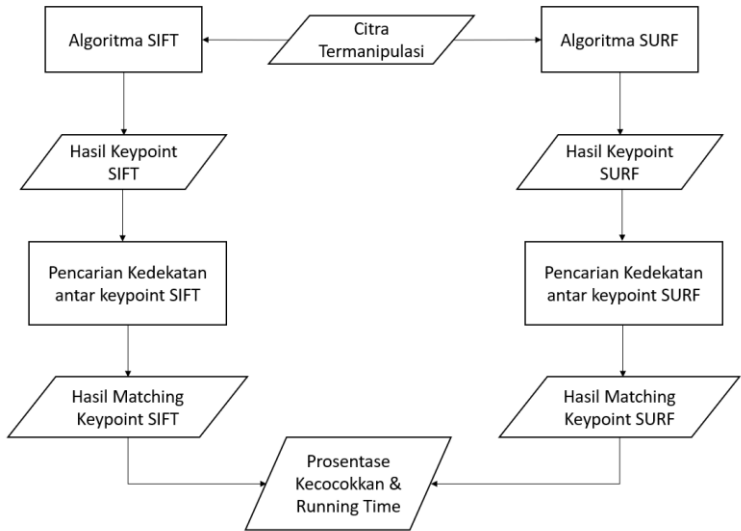
Gambar 3.6 Citra *dermoscopic* yang sudah dimanipulasi *copy-move*

3.5.3 Citra Dermoscopic Termanipulasi

Citra dermoscopic yang sudah termanipulasi inilah yang akan dijadikan dataset masukan yang kemudian akan dilakukan pengolahan citra.

3.6 Pengolahan Citra

Pengolahan citra untuk mendeteksi citra termanipulasi *copy-move* tersusun atas beberapa tahapan mulai dari ekstraksi fitur menggunakan algoritma SIFT dan SURF, memunculkan hasil keypoint pada masing-masing algoritma, kemudian mencari kedekatan pada tiap-tiap keypoint hingga keluar hasil output berupa prosentase kecocokkan dan running time pada masing-masing algoritma. Pada gambar 3.5 ditampilkan diagram alir dari metodologi pengolahan citra.



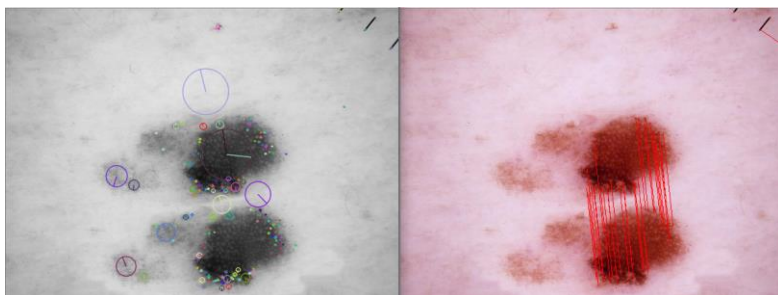
Gambar 3.7 Diagram Alir metodologi pengolahan citra.

Setelah citra *dermoscopic* berhasil di manipulasi, citra akan diinputkan ke dalam program untuk dilakukan pengolahan citra menggunakan algoritma SIFT dan SURF. Dalam prosesnya masing-masing dari algoritma SIFT dan SURF akan mengalami beberapa tahapan-tahapan sehingga memunculkan keypoint. Kemudian, dari hasil keypoint yang muncul, akan dicari kedekatannya menggunakan metode *matching* keypoint agar bisa terjadi kecocokkan antar keypoint. Setelah itu, hasil akhir yang diharapkan adalah muncul prosentase kecocokkan setelah terjadi proses *matching* serta muncul waktu komputasi atau *running time* dari masing-masing algoritma SIFT dan SURF.

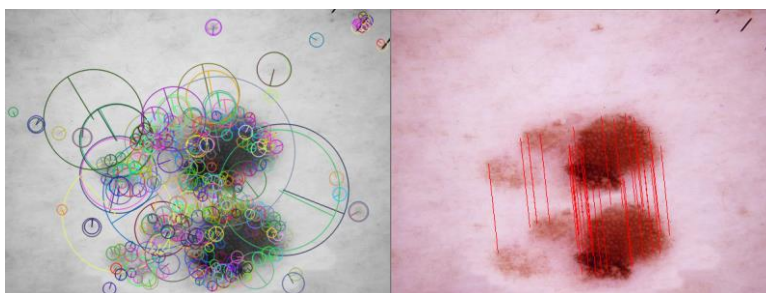
3.7 Data Keluaran

Data masukan berupa citra *dermoscopic* termanipulasi akan diproses dengan menggunakan algoritma *Scale Invariant Feature Transform (SIFT)* dan *Speeded-up Robust Feature (SURF)*. Hasil dari proses deteksi serangan copy-move pada masing-masing metode tersebut adalah dua buah output yakni yang pertama adalah citra masukan dengan daerah yang sudah menunjukkan *Keypoint*, kemudian yang kedua adalah citra

yang *keypoint*nya sudah melalui proses *matching*. Dengan adanya daerah *Keypoint* serta *threshold* yang ditentukan maka program akan otomatis mencocokkan *keypoint* citra asli dengan citra hasil duplikasi. Garis berwarna merah menunjukkan kecocokan atau kesamaan *keypoint* yang berarti citra tersebut mempunyai fitur yang sejenis dalam satu citra. Hasil proses akhir deteksinya adalah jumlah *keypoint*, jumlah *keypoint* yang cocok, prosentase kecocokkan *keypoint* serta waktu komputasi masing-masing algoritma dalam satuan sekon. Waktu komputasi diperlukan untuk membandingkan metode mana yang lebih cepat dalam menjalankan program. Gambar 3.6 dan 3.7 merupakan contoh hasil output citra menggunakan kedua metode.



Gambar 3.8 Contoh hasil output citra menggunakan metode SIFT.

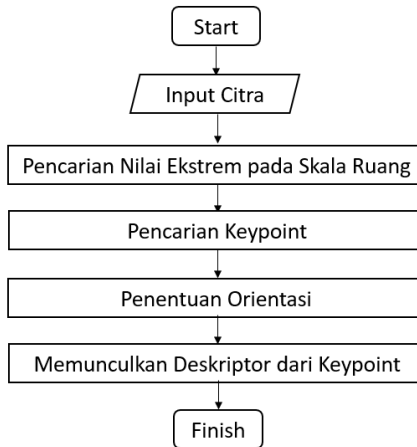


Gambar 3.9 Contoh hasil output citra menggunakan metode SURF.

3.8 Scale Invariant Feature Transform (SIFT)

Ekstraksi ciri pertama yang digunakan pada penelitian tugas akhir ini adalah *Scale Invariant Feature Transform* (SIFT). Ekstraksi ciri ini akan

mengubah suatu citra menjadi vector fitur local yang kemudian akan digunakan sebagai pendekatan dalam mendeteksi citra yang dimaksud. Secara garis besar, metode SIFT terdiri dari empat tahapan yaitu mencari nilai ekstrem pada skala ruang, mencari *keypoint*, penentuan orientasi dan *descriptor keypoint*. Gambar 3.8 merupakan diagram alir dari algoritma SIFT.



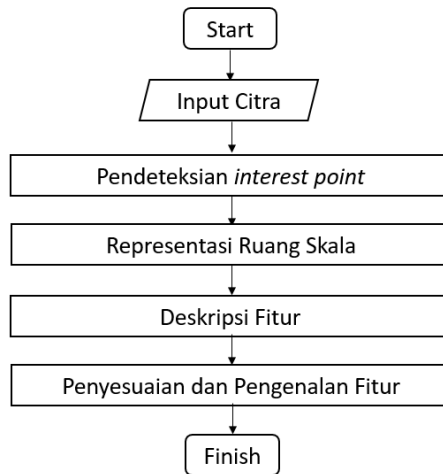
Gambar 3.10 Diagram alir untuk algoritma SIFT.

Pada tugas akhir ini langkah pertama untuk mendapatkan deskriptor adalah dengan melakukan pengujian pada citra termanipulasi. Citra diolah melalui tahap *pre-processing*. Kemudian mendeteksi titik menggunakan ekstraksi ciri SIFT dengan menggunakan perangkat lunak Spyder sebagai IDE, sehingga akan didapatkan output yang berisikan fitur setiap citra dan letak posisi fitur – fitur tersebut.

Kemudian langkah selanjutnya adalah memasukkan metode *matching keypoint* yang akan dijelaskan pada sub-bab selanjutnya guna untuk memasang *keypoint-keypoint* yang terdeteksi sebagai descriptor yang sudah muncul dari hasil pengolahan citra menggunakan metode SIFT ini. Hasil perbandingan kecepatan komputasi dan prosentase kecocokkan tiap *keypoint* nantinya akan dibandingkan dengan metode ekstraksi ciri yang menggunakan SURF.

3.9 Speeded-up Robust Feature

Selain daripada menggunakan algoritma SIFT untuk mendeteksi fitur dan *matching*, akan digunakan pula metode lainnya yang akan dijadikan pembanding yaitu menggunakan algoritma SURF atau *speeded-up robust feature*. Metode SURF menggunakan pendekatan *integral images* dan matriks Hessian untuk mencari titik kunci yang stabil dalam berbagai tingkat penyekalaan. Gambar 3.10 merupakan diagram alir dari algoritma SURF.



Gambar 3.11 Diagram alir algoritma SURF.

Tidak berbeda dengan metode ekstraksi ciri SIFT, untuk proses pengujian dilakukan pada citra termanipulasi. Citra yang telah diolah melalui tahap *pre-processing* ini, dengan menggunakan perangkat lunak berbahasa pemrograman python, maka akan didapatkan citra yang berisikan ciri dari hasil ekstraksi ciri SURF.

Kemudian langkah selanjutnya adalah memasukkan metode *matching keypoint* yang akan dijelaskan pada sub-bab selanjutnya guna untuk memasang *keypoint-keypoint* yang terdeteksi sebagai descriptor yang sudah muncul dari hasil pengolahan citra menggunakan metode SURF ini. Hasil perbandingan kecepatan komputasi dan prosentase

kecocokkan tiap *keypoint* nantinya akan dibandingkan dengan metode ekstraksi ciri yang menggunakan SIFT.

3.10 *Matching Keypoint*

Setelah fitur-fitur dari metode SIFT dan SURF berhasil dan muncul titik-titik *keypoint*, lalu citra masukan tadi diproses menggunakan *matching keypoint* yang tujuannya adalah untuk menemukan kecocokkan *keypoint* pada kedua citra. Jika citra yang diuji berhasil menemukan kecocokkan *keypoint* lebih dari 10, maka citra tersebut positif merupakan hasil *copy-move*. Jika citra yang diuji menghasilkan kecocokkan *keypoint* kurang dari 10 maka citra tersebut bukan merupakan hasil *copy-move* karena tidak memenuhi syarat *threshold*.

Ada dua langkah yang diperlukan untuk melakukan *matching keypoint*, yaitu *pairing keypoint* atau memasang *keypoint* dan yang kedua adalah memilih *keypoint* mana yang dapat diandalkan (*reliable*). Langkah pertama, *pairing keypoint*, menggunakan rumus berikut (11) untuk menemukan pasangan antar *keypoint descriptor*.

$$\forall f \in L, f_* = \operatorname{argmin}_{f_* \in L} \|f_* - f\|_2 \text{ such that } L_* = L - \{f\} \quad (11)$$

Keterangan:

f = vector f

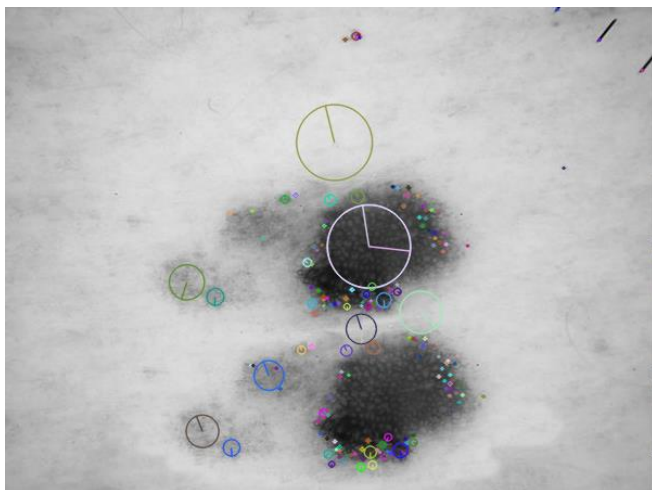
f_* = vector f_*

\in = elemen dari

\forall = untuk semua

L = *keypoint descriptor*

argmin = argument minimum dari sebuah vector



Gambar 3.12 Hasil dari ekstraksi *keypoint descriptor*.

Langkah berikutnya adalah memilih keypoint yang paling bisa diandalkan melalui nilai $C_Relative$ yang ditentukan dengan menggunakan rumus persamaan (12).

$$\forall f_ \in L_ - \{f_ \}, \|f_^* - f_ \| < C_relative \|f_ - f_ \| \quad (12)$$

Keterangan:

f = vector f

$f_$ = vector $f_$

\in = elemen dari

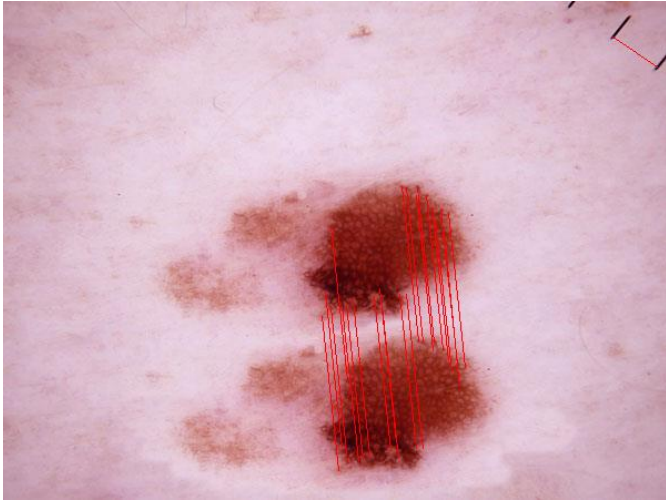
\forall = untuk semua

L = keypoint descriptor

argmin = argument minimum dari sebuah vector

$C_relative$ = nilai jarak terdekat

Kemudian dengan rumus tersebut dikalkulasikan apakah jarak antara dua *keypoint* sudah lebih kecil dari $C_Relative$ dari jarak terdekat yang sudah ditentukan. Dengan cara ini *threshold* relatif lebih efisien.



Gambar 3.13 Hasil dari keypoint yang berhasil *match*.

[Halaman ini sengaja dikosongkan].

BAB IV

PENGUJIAN DAN ANALISA

Pada bab ini dipaparkan mengenai pengujian terhadap sistem dan perangkat yang telah direalisasikan untuk mengetahui input dan output sesuai dengan yang diharapkan. Analisis hasil pengujian juga disampaikan pada bab ini. Data yang digunakan dalam pengujian pada tugas akhir ini diperoleh dengan dataset yang sudah disediakan oleh MNIST HAM10000 dan DERMNET kemudian dilanjutkan dengan metode-metode yang telah dirancang.

4.1 Spesifikasi Perangkat

Lingkungan pengujian pada uji coba permasalahan perbandingan metode deteksi SIFT dan SURF dalam mendeteksi kepalsuan citra yang terserang *copy-move* dengan menggunakan spesifikasi perangkat keras dan perangkat lunak seperti yang ditunjukkan pada Tabel 4.1.

Tabel 4.1 Spesifikasi Perangkat untuk Pengujian

Perangkat	Jenis Perangkat	Spesifikasi
Perangkat Keras	Prosesor	Intel® Core™ i7-7700HQ CPU @2.8GHz (4 Cores)
	Memori	16 GB 2800 MHz DDR4
Perangkat Lunak	Sistem Operasi	Windows 10 Pro
	Perangkat Pengembang	Spyder 3.3.3

4.2 Pengujian Menggunakan Citra Copy Move MNIST HAM10000

Pengujian menggunakan citra dermoskopik dari MNIST HAM10000 yang di *copy-move* ini bertujuan untuk mendapatkan kecocokan antara fitur citra asli dengan citra yang sudah di *copy-move* pada satu citra yang sama. Ketika citra asli menemukan kecocokkan pada

citra palsu maka akan dilakukan pencocokkan antar keypoint berupa garis merah.

Pada pengujian menggunakan citra yang sudah di copy-move ini terdapat hasil yang diharapkan. Keypoint yang muncul dari fitur-fitur yang di ekstraksi dari tiap citra asli dan copy-move saling mencocokkan satu sama lain sehingga dapat dideteksi bahwa citra yang diujikan positif dilakukan copy-move. Pada tabel 4.2 ditunjukkan semua hasil pengujian citra yang telah dilakukan copy-move.

Tabel 4.2 Hasil perbandingan metode SIFT dan SURF pada citra MNIST HAM10000.

Citra	<u>SIFT</u>		<u>SURF</u>	
	Prosentase Kecocokkan	Waktu Running	Prosentase Kecocokkan	Waktu Running
(1)	29%	0.06s	19,94%	0.04s
(2)	33%	0.035s	31,43%	0.031s
(3)	18,58%	0.8s	9,5%	0.19s
(4)	25,29%	0.05s	21,18%	0.03s
(5)	24,66%	0.06s	18,05%	0.03s
(6)	32,43%	0.05s	19,5%	0.03s
(7)	33,30%	0.04s	19,29%	0.03s
(8)	23,75%	0.04s	22,73%	0.03s
(9)	19,80%	0.06s	16,50%	0.04s
(10)	18%	0.08s	15,70%	0.03s

Dari hasil uji coba untuk menjalankan program, dapat ditemukan hasil seperti Tabel 4.2. Terlihat bahwa memang SIFT dapat menemukan kecocokkan lebih banyak daripada SURF, namun SIFT menggunakan Running Time lebih banyak daripada SURF dalam memproses algoritmanya.

4.3 Pengujian Menggunakan Citra Copy-Move Acne DERMNET

Pengujian menggunakan citra dermoscopic dengan jenis *Acne* yang di *copy-move* mempunyai hasil lumayan. Dengan algoritma SIFT, prosentase kecocokkan antara 20% - 45% dengan tiga citra pada citra ke 4, 5, dan 6 masih dibawah 20% untuk algoritma. Sedangkan untuk citra ke 1 dan 7 mempunyai prosentase 45% karea memiliki ketajaman citra yang lebih baik dari citra yang lainnya. Sementara algoritma SURF mempunyai prosentase kecocokkan selalu di bawah algoritma SIFT. Seluruh running time juga mempunyai trend yang sesuai dengan hasil yang diinginkan yaitu untuk metode SURF lebih cepat dari SIFT. Tabel 4.3 menunjukkan hasil dari pengujian citra copy-move dengan jenis acne.

Tabel 4.3 Hasil perbandingan metode SIFT dan SURF pada citra *dermoscopic* berjenis *Acne* yang didapat dari DERMNET.

<u>SIFT</u>			<u>SURF</u>	
Citra	Prosentase Kecocokkan	Waktu Running	Prosentase Kecocokkan	Waktu Running
(1)	45%	0.6s	36,5%	0.17s
(2)	21%	0.5s	17,5%	0.12s
(3)	24%	0.2s	14,3%	0.05s
(4)	18,7%	0.7s	15,3%	0.4s
(5)	19%	0.6s	15,4%	0.4s
(6)	17,3%	0.21s	12%	0.1s
(7)	45%	0.1s	33,1%	0.08s
(8)	23%	0.24s	17,6%	0.13s
(9)	25%	0.20s	19,5%	0.15s
(10)	24,3%	0.16s	18%	0.1s

4.4 Pengujian Menggunakan Citra Copy-Move Atopic Dermatitis DERMNET

Pengujian menggunakan citra dermoskopik dengan jenis *Atopic Dermatitis* yang di *copy-move* mempunyai hasil yang cukup signifikan dibandingkan dengan citra acne sebelumnya. Dengan algoritma SIFT,

prosentase kecocokkan paling kecil adalah 21,7% dan yang tertinggi sampai 75%. Pada citra ke 1 dan 6 mempunyai kecocokkan sampai 70% karena citra lebih tajam dibandingkan dengan ctra yang lain. Sementara algoritma SURF mempunyai prosentase kecocokkan selalu di bawah algoritma SIFT dengan angka kecocokkan tertinggi di 45,4% pada citra ke 4. Untuk running time masih menunjukkan SURF masih lebih cepat dari SIFT. Hasil dapat dilihat pada tabel 4.4.

Tabel 4.4 Hasil perbandingan metode SIFT dan SURF pada citra *dermoscopic* jenis Atopic Dermatitis yang didapat dari DERMNET.

	<u>SIFT</u>		<u>SURF</u>	
Citra	Prosentase Kecocokkan	Waktu Running	Prosentase Kecocokkan	Waktu Running
(1)	75%	0.6s	41,5%	0.2s
(2)	25,2%	0.4s	21,4%	0.13s
(3)	54%	0.09s	42,5%	0.06s
(4)	59%	0.12s	46,4%	0.08s
(5)	63,8%	0.19s	38,2%	0.19s
(6)	72,9%	0.14s	35%	0.08s
(7)	23,2%	0.5s	21%	0.24s
(8)	26,5%	0.1s	16,8%	0.06s
(9)	56,6%	0.2s	32,8%	0.09s
(10)	21,7%	0.48s	14,5%	0.21s

4.5 Pengujian Menggunakan Citra Copy-Move Furuncle DERMNET

Pengujian menggunakan citra dermoscopic dengan jenis *Furuncle* yang di *copy-move* mempunyai hasil yang berbeda. Dengan algoritma SIFT, prosentase kecocokkan paling kecil adalah 30% dan yang tertinggi sampai 76,4%. Pada citra ke 8 mempunyai kecocokkan sampai 76,4% karena citra lebih tajam dibandingkan dengan ctra yang lain. Citra *Furuncle* ini mempunyai rata-rata hasil kecocokkan terbaik dengan kecocokkan yang paling kecilnya adalah 30%. Sementara algoritma SURF mempunyai prosentase kecocokkan selalu di bawah

algoritma SIFT dengan angka kecocokkan tertinggi di 44,8% pada citra ke 4. Untuk running time masih menunjukkan SURF masih lebih cepat dari SIFT. Hasil dapat dilihat pada tabel 4.5.

Tabel 4.5 Hasil perbandingan metode SIFT dan SURF pada citra *dermoscopic* jenis Furuncle yang didapat dari DERMNET.

	<u>SIFT</u>		<u>SURF</u>	
Citra	Prosentase Kecocokkan	Waktu Running	Prosentase Kecocokkan	Waktu Running
(1)	47,16%	0.9s	24,56%	0.4s
(2)	41,3%	0.5s	26,3%	0.19s
(3)	34,3%	0.36s	23,6%	0.18s
(4)	59,3%	0.2s	44,8%	0.17s
(5)	38,1%	0.25s	27,9%	0.11s
(6)	36,6%	0.34s	17,8%	0.11s
(7)	36,9%	0.6s	22,7%	0.5s
(8)	76,4%	0.4s	28%	0.25s
(9)	30%	0.12s	19,4%	0.05s
(10)	51,2%	0.2s	44,2%	0.08s

4.6 Pengujian Menggunakan Citra Copy-Move Psoriasis DERMNET

Pengujian menggunakan citra *dermoscopic* dengan jenis *Psoriasis* yang di *copy-move* hasilnya tidak sebaik dengan pengujian citra sebelumnya. Dengan algoritma SIFT, prosentase kecocokkan pada citra ke 2, 3, 7, dan 9 masih dibawah 20%. Sedangkan prosentase yang tertinggi adalah 54,2%. Banyak dari citra *psoriasis* mempunyai tekstur yang tidak begitu tajam sehingga menghasilkan ekstraksi fitur yang minim dan sedikit terjadi pencocokkan. Sementara algoritma SURF tetap mempunyai prosentase kecocokkan yang selalu di bawah algoritma SIFT dengan angka kecocokkan tertinggi di 37,3% pada citra ke 1. Untuk running time masih menunjukkan SURF masih lebih cepat dari SIFT. Hasil dapat dilihat pada tabel 4.6.

Tabel 4.6 Hasil perbandingan metode SIFT dan SURF pada citra *dermoscopic* jenis Psoriasis yang didapat dari DERMNET.

	<u>SIFT</u>		<u>SURF</u>	
Citra	Prosentase Kecocokkan	Waktu Running	Prosentase Kecocokkan	Waktu Running
(1)	54,2%	0.19s	37,3%	0.11s
(2)	15,7%	0.6s	14,4%	0.2s
(3)	15,9%	0.52s	14%	0.37s
(4)	40,5%	0.12s	29,2%	0.09s
(5)	26,2%	0.26s	15,3%	0.07s
(6)	24,2%	0.14s	15,4%	0.06s
(7)	15,3%	0.37s	13,5%	0.24s
(8)	26,3%	0.15s	19,6%	0.08s
(9)	18,5%	0.19s	14,6%	0.1s
(10)	30,2%	0.38s	20,2%	0.15s

4.7 Pengujian Menggunakan Citra Copy-Move Ringworm DERMNET

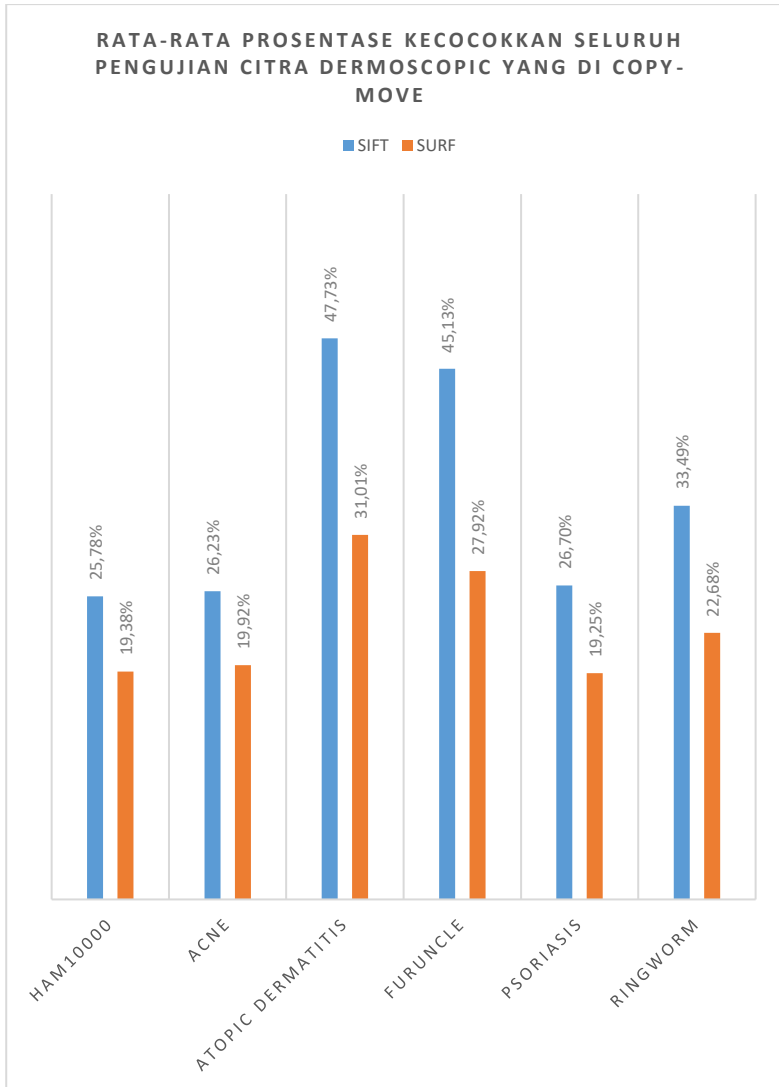
Pengujian menggunakan citra *dermoscopic* dengan jenis Ringworm yang di *copy-move*, pada algoritma SIFT, prosentase kecocokkan paling kecil adalah 15,9% pada citra ke-2 dan yang tertinggi adalah 62,4% pada citra ke-6. Ada dua citra yang prosentase kecocokkannya masih dibawah 20% karena mempunyai tekstur yang tidak begitu tajam sehingga ekstraksi fitur dan pencocokkannya masih kurang yaitu pada citra ke 1 dan 3. Sementara algoritma SURF mempunyai prosentase kecocokkan selalu di bawah algoritma SIFT dengan angka kecocokkan tertinggi di 40,4% pada citra ke 2. Untuk running time masih menunjukkan SURF masih lebih cepat dari SIFT. Hasil dapat dilihat pada tabel 4.7.

Tabel 4.7 Hasil perbandingan metode SIFT dan SURF pada citra *dermoscopic* jenis Ringworm yang didapat dari DERMNET.

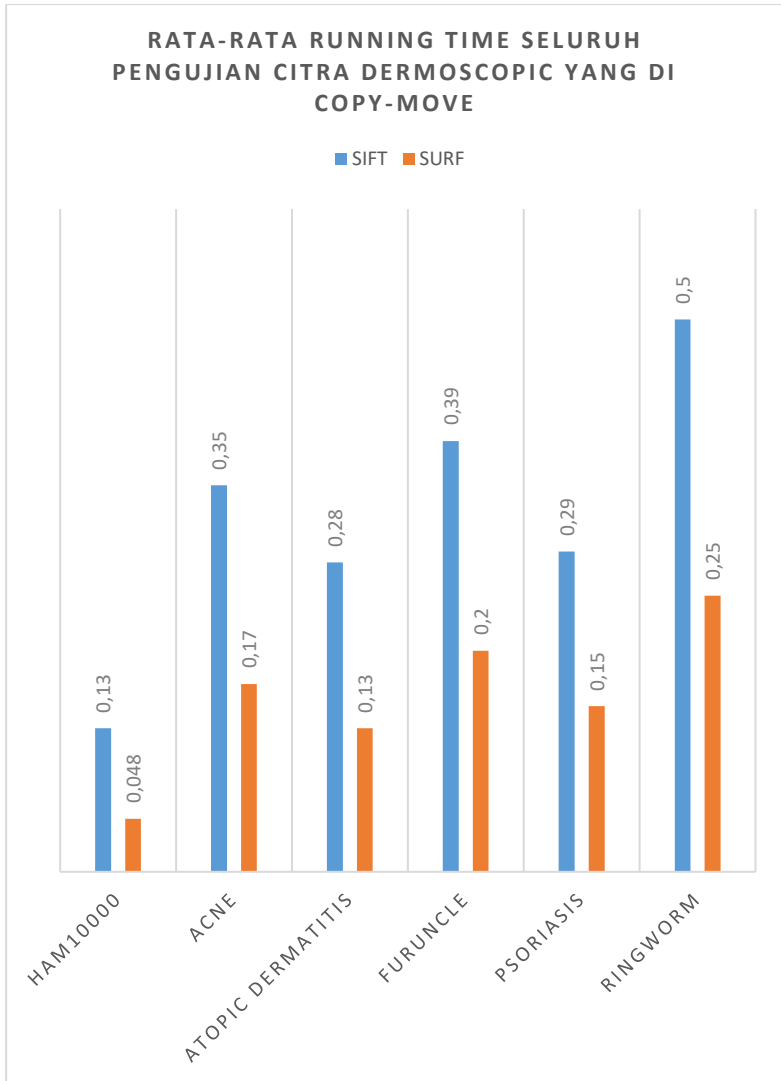
<u>SIFT</u>			<u>SURF</u>	
Citra	Prosentase Kecocokkan	Waktu Running	Prosentase Kecocokkan	Waktu Running
(1)	17,5%	0.68s	16,3%	0.3s
(2)	44,15%	0.13s	40,4%	0.08s
(3)	15,9%	0.6s	12,5%	0.4s
(4)	34,9%	0.8s	18%	0.4s
(5)	24,9%	0.6s	13,6%	0.29s
(6)	62,4%	0.8s	37,5%	0.3s
(7)	46,5%	0.34s	36,3%	0.14s
(8)	28,9%	0.3s	15,3%	0.21s
(9)	30,9%	0.39s	19,6%	0.2s
(10)	28,8%	0.24s	17,3%	0.12s

4.8 Analisa Perbandingan Hasil Pengujian

Setelah dilakukan pengujian terhadap citra *dermoscopic* dengan 60 dataset, dapat ditemukan bahwa performa dari algoritma SIFT dalam hal ketepatan pencocokkan citra *copy-move* mempunyai hasil selalu di atas dari algoritma SURF. Dengan citra *Atopic Dermatitis* yang didapat dari DERMNET mempunyai citra dengan rata-rata prosentase terbaik yaitu 47,73% dengan tiga citra menghasilkan prosentase kecocokkan diatas 60%. Sedangkan untuk *running time* nya, pada setiap pengujian citra didapatkan hasil algoritma SURF akan selalu lebih cepat dalam menjalankan programnya dari pada algoritma SIFT, namun dengan ketepatan pencocokkan yang selalu dibawah algoritma SIFT namun tidak begitu jauh. Pada gambar 4.1 diberikan grafik rata-rata prosentase kecocokkan para seluruh pengujian. Lalu pada gambar 4.2 adalah grafik rata-rata running time pada seluruh pengujian yang sudah dilakukan.



Gambar 4.1 Grafik rata-rata prosentase kecocokkan pada seluruh pengujian citra *dermoscopic*.



Gambar 4.2 Grafik rata-rata *running time* pada seluruh pengujian citra *dermoscopic* (dalam satuan sekon).

[Halaman ini sengaja dikosongkan].

BAB V

PENUTUP

Dalam bab ini dijelaskan beberapa kesimpulan dari tugas akhir ini beserta saran untuk kelengkapan dan perbaikan penelitian selanjutnya.

5.1 Kesimpulan

Dalam tugas akhir ini, telah diimplementasikan dan dibandingkan dua metode yaitu *Scale Invariant Feature Transform* (SIFT) dan *Speeded-up Robust Feature* (SURF) untuk pendeteksian kepalsuan citra *dermoscopic*. Setelah dilakukan beberapa pengujian data, didapatkan kesimpulan:

1. Hasil penelitian menunjukkan bahwa pada seluruh pengujian, algoritma SIFT akan selalu lebih baik dalam melakukan kecocokkan citra yang *tercopy-move*.
2. Algoritma SURF selalu lebih cepat pada *running time* saat menganalisa daripada menggunakan algoritma SURF. Perbedaan kecepatan *running time* ketika menggunakan algoritma SURF adalah hampir setengah dari Ketika menggunakan algoritma SIFT.
3. Citra *dermoscopic* dengan jenis penyakit kulit *Atopic Dermatitis* mempunyai rata-rata prosentase kecocokkan tertinggi dibandingkan dengan pengujian citra yang lain. Ini karena citra *atopic dermatitis* yang didapat mempunyai tekstur lebih baik dibandingkan citra yang lain.
4. Citra *dermoscopic* yang didapat dari MNIST HAM10000 mempunyai rata-rata *running time* tercepat dalam pengujian dibandingkan dengan citra yang lain.

5.2 Saran

Demi pengembangan penelitian lebih lanjut mengenai tugas akhir ini, disarankan beberapa hal yaitu:

1. Penggunaan Spesifikasi perangkat terbaru atau terbaik untuk memangkas waktu komputasi.
2. Memperbanyak variasi dataset penyakit kulit atau bahkan penyakit jenis lain untuk memperluas data pengujian.

DAFTAR PUSTAKA

- [1] A. Yuliani, Ada 800.000 Situs Penyebar Hoax di Indonesia, https://kominfo.go.id/content/detail/12008/ada-800000-situs-penyebar-hoax-diindonesia/0/sorotan_media, Diakses 20 April 2020.
- [2] J. Sammons, The basics of digital forensics. Amsterdam: Syngress Media, 2015.
- [3] Oommen RS, Jayamohan M, Sruthy S. A survey of copy-move forgery detection techniques for digital images. *Int J Innov Eng Technol* 2015;5(2):429-36.
- [4] Salloum Ronald, Ren Yuzhuo, Jay Kuo CC. Image splicing localization using a multi-task fully convolutional network (MFCN). *J Vis Commun Image Represent* 2018;51:201-9.
- [5] Nandini C, Meenakshi sundaram A. Image retouching and IT'S detection - a SURVEY. *Int J Res Eng Technol* 2015;04(14):30-4.
- [6] Basuki, Achmad. 2005. *Metode Numerik dan Algoritma Komputasi*. Yogyakarta: ANDI.
- [7] P. Tschandl, C. Rosendahl, and H. Kittler, The ham10000 dataset, "*a large collection of multi-source dermatoscopic images of common pigmented skin lesions*," *Scientific data*, vol. 5, p. 180161, 2018. (Dikutip pada halaman 5, 25, 26, 27, 28, 29).
- [8] Dilshad M, Ghrera S, Tyagi V. Pixel based image forgery detection: a review. *IETE J Edu*. 2014;55(1):40– 44.
- [9] David A. Forsyth; Jean Ponce (2003). *Computer Vision, A Modern Approach*. Prentice Hall.
- [10] Gualtieri et al, Feature Extraction Method, (1985).

- [11] Elgendy, Mohamed. 2019. *Deep Learning for Vision Systems*. Manning.
- [12] D. G. Lowe, “*Distinctive image features from scale-invariant keypoints*,” *Int.’l Journal of Computer Vision*, vol. 60, no. 2, halaman 91–110, Januari 2004.
- [13] H. Bay et al, “Speeded-Up Robust Features (SURF),” *Computer Vision and Image Understanding* 110 (2008), pp. 346-39.
- [14] “Anaconda Documentation,” Anaconda, [Online]. Available: <https://docs.anaconda.com/> [Diakses 21 November 2019].
- [15] Ninh, K., “Python Scripting Language”, Creative Commons Attribution License 4.0, 2017.
- [16] The NumPy Community, “NumPy User Guide”, 2016.
- [17] Adobe. "Photoshop CS6 Beta Now Available on Adobe Labs". Press Releases. Adobe.

BIOGRAFI PENULIS



Muhammad Rizki, lahir pada 25 Maret 1996 di Jakarta, DKI Jakarta. Penulis merupakan anak pertama dari dua bersaudara. Saat ini penulis tinggal di Surabaya, provinsi Jawa Timur. Pada tahun 2008 penulis menyelesaikan pendidikan sekolah dasar di SD Mutiara Bunda di Sidoarjo. Tahun 2011 penulis lulus dari SMP Al-Falah Tropodo di Waru. Untuk pendidikan di bangku menengah atas, tahun 2014 penulis lulus dari SMA Muhammadiyah 2 Sidoarjo. Penulis sedang menjalani studi S1 di Departemen Teknik Komputer, Fakultas Teknologi Elektro dan Informatika Cerdas, Institut Teknologi Sepuluh Nopember Surabaya. Selama masa perkuliahan, penulis aktif mengikuti organisasi dan event-event kampus lainnya seperti Petrolida, Electra 5, dan Electra 6. Bagi pembaca yang memiliki kritik, saran, atau pertanyaan mengenai tugas akhir ini dapat menghubungi penulis melalui surel (25rwizki@gmail.com)

