



**UNIVERSITAS INDONESIA**

**ANALISIS PENGENALAN OBJEK KAPAL PERANG  
BERDASARKAN PENDEKATAN FITUR DENGAN METODE  
*SCALE INVARIANT FEATURE TRANSFORM***

**SKRIPSI**

**CHANDRA HARTANTO**

**(0806455143)**

**FAKULTAS TEKNIK  
PROGRAM STUDI TEKNIK ELEKTRO  
DEPOK  
JUNI 2012**



**UNIVERSITAS INDONESIA**

**ANALISIS PENGENALAN OBJEK KAPAL PERANG  
BERDASARKAN PENDEKATAN FITUR DENGAN METODE  
*SCALE INVARIANT FEATURE TRANSFORM***

**SKRIPSI**

Diajukan sebagai salah satu syarat untuk memperoleh gelar  
Sarjana Teknik

**CHANDRA HARTANTO**  
**(0806455143)**

**FAKULTAS TEKNIK  
PROGRAM STUDI TEKNIK ELEKTRO  
DEPOK  
JUNI 2012**

## HALAMAN PERNYATAAN ORISINALITAS

Skripsi ini adalah hasil karya saya sendiri, dan semua sumber baik yang dikutip maupun dirujuk telah saya nyatakan dengan benar

Nama : Chandra Hartanto

NPM : 0806455143

Tanda Tangan : 

Tanggal : 21 Juni 2012

## HALAMAN PENGESAHAN

Skripsi ini diajukan oleh:

Nama : Chandra Hartanto  
NPM : 0806455143  
Program Studi : Teknik Elektro  
Judul Skripsi : ANALISIS PENGENALAN OBJEK KAPAL  
PERANG BERDASARKAN PENDEKATAN  
FITUR DENGAN METODE *SCALE*  
*INVARIANT FEATURE TRANSFORM*

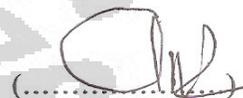
Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana Teknik pada Program Studi Teknik Elektro, Fakultas Teknik, Universitas Indonesia.

### DEWAN PENGUJI

Pembimbing : Prof. Dr. Ir. Dadang Gunawan, M.Eng



Penguji : Dr. Ir. Arman D. Diponegoro



Penguji : Dr. Muhammad Suryanegara, S.T., M.Sc



Ditetapkan di : Depok

Tanggal : 21 Juni 2012

## UCAPAN TERIMA KASIH

Puji syukur saya panjatkan kepada Tuhan Yang Maha Esa, karena atas berkat dan karunia-NYA saya dapat menyelesaikan skripsi ini. Skripsi ini ditulis sebagai salah satu persyaratan untuk mencapai gelar Sarjana Teknik pada Program Studi Teknik Elektro Fakultas Teknik Universitas Indonesia. Skripsi ini telah selesai dibuat tentunya dengan bantuan dari beberapa pihak. Tanpa bantuan dan bimbingan mereka sangat sulit bagi saya untuk dapat menyelesaikan skripsi ini tepat waktu. Oleh karena itu, perkenankanlah saya sebagai penulis untuk mengucapkan terima kasih kepada:

1. Bapak Prof. Dr. Ir. Dadang Gunawan, M.Eng, selaku pembimbing yang selalu memberikan bimbingan dan arahan sehingga skripsi ini dapat terselesaikan dengan baik.
2. Para peneliti sebelumnya yang penelitiannya telah menjadi sumber referensi bagi saya.
3. Bpk. Midi Wijaya dan Ibu Fie-Fie selaku kedua orang tua serta Sdr. Arwindy dan Sdr. Edward selaku adik-adik penulis yang selalu memberikan dukungan moril dan materi untuk penulis sehingga dapat menyelesaikan skripsi ini.
4. Sdri. Lietalia yang selalu memberikan semangat dan motivasi bagi penulis sehingga skripsi ini dapat diselesaikan dengan baik.
5. Seluruh Civitas Akademik Departemen Teknik Elektro Universitas Indonesia atas dukungannya dalam penyelesaian skripsi ini.

Akhir kata, tentunya penulis mendoakan semoga Tuhan YME berkenan membalas kebaikan semua pihak yang turut membantu sehingga skripsi ini dapat diselesaikan dengan baik. Semoga skripsi ini dapat berkontribusi bagi perkembangan ilmu pengetahuan

Depok, Juni 2012

Chandra Hartanto

**HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI  
TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS**

---

Sebagai sivitas akademik Universitas Indonesia, saya yang bertanda tangan dibawah ini:

Nama : Chandra Hartanto

NPM : 0806455143

Program Studi : Teknik Elektro

Departemen : Teknik Elektro

Fakultas : Teknik

Jenis Karya : Skripsi

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Indonesia **Hak Bebas Royalti Noneksklusif (*Non-exclusive Royalty-Free Right*)** atas karya ilmiah saya yang berjudul:

**ANALISIS PENGENALAN OBJEK KAPAL PERANG BERDASARKAN  
PENDEKATAN FITUR DENGAN METODE *SCALE INVARIANT*  
*FEATURE TRANSFORM***

berserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini Universitas Indonesia berhak menyimpan, mengalihmedia/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan memublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya,

Dibuat di : Depok

Pada tanggal : 21 Juni 2012

Yang Menyatakan



(CHANDRA HARTANTO)

v

Universitas Indonesia

v

Universitas Indonesia

## ABSTRAK

Nama : Chandra Hartanto  
Program Studi : Teknik Elektro  
Judul : Analisis Pengenalan Objek Kapal Perang Berdasarkan Pendekatan Fitur dengan Metode *Scale Invariant Feature Transform*.

Salah satu pendekatan dalam pengenalan objek dengan menggunakan komputer adalah pendekatan fitur. Pendekatan fitur dilakukan dengan mencari fitur-fitur penting atau *keypoint-keypoint* dari suatu gambar. *Scale invariant feature transform* (SIFT) merupakan salah satu metode yang cukup populer dalam pengenalan objek berdasarkan pendekatan fitur. Skripsi ini dibuat untuk merancang suatu program untuk mengenali objek kapal perang dengan metode SIFT dan melakukan simulasi serta analisis dari hasil program tersebut. Performa SIFT dalam mengenali objek akan diukur dari persentase kesuksesan mengenali gambar, parameter *recall vs 1-precision*, dan parameter F1 Score sesuai dengan nilai *threshold* yang divariasikan. Hasil simulasi menunjukkan bahwa nilai *threshold* yang paling optimal adalah senilai 0.7 dengan tingkat akurasi sebesar 83,33%.

Kata Kunci : Pengenalan Objek, SIFT, Kapal Perang

## ABSTRACT

Name : Chandra Hartanto  
Study Program : Electrical Engineering  
Title : Object Recognition Analysis for Battleship based on  
Feature Approaches with Scale Invariant Feature Transform Method.

Feature approach is one of computerized object recognition approaches. Feature approach is done by finding important features or *keypoints* from an image. Scale Invariant Feature Transform (SIFT) is a popular method in object recognition based on feature approach. This thesis is created to design an object recognition program on battleship applications with SIFT method, do the simulation, and analyze the program results. SIFT's performance on recognizing object will be measured from successful recognition percentage, recall vs 1-precision, and F1 Score parameter with varied threshold value. Simulation results show that the best threshold value is 0.7 with 83.33% accuracy.

Keywords : Object Recognition, SIFT, Battleship

## DAFTAR ISI

HALAMAN JUDUL .....	I
HALAMAN PERNYATAAN ORISINALITAS.....	II
HALAMAN PENGESAHAN.....	III
UCAPAN TERIMA KASIH.....	IV
HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI.....	V
ABSTRAK.....	VI
ABSTRACT .....	VII
DAFTAR ISI.....	VIII
DAFTAR GAMBAR.....	X
DAFTAR TABEL .....	XI
DAFTAR LAMPIRAN.....	XII
<b>BAB 1 PENDAHULUAN.....</b>	<b>1</b>
1.1    LATAR BELAKANG .....	1
1.1.1    Pengenalan Objek.....	1
1.1.2    Objek Kapal Perang.....	2
1.1.3    Pengenalan Kapal Perang.....	3
1.2    TUJUAN PENELITIAN.....	3
1.3    PEMBATASAN MASALAH .....	3
1.4    METODE PENELITIAN.....	4
1.5    SISTEMATIKA PENULISAN .....	4
<b>BAB 2 PENGENALAN OBJEK .....</b>	<b>6</b>
2.1    METODE PENGENALAN OBJEK .....	6
2.1.1 <i>Geometry Based Approaches</i> .....	6
2.1.2 <i>Appearance Based Approaches</i> .....	8
2.1.3 <i>Feature Based Approaches</i> .....	11
2.1.3.1    Pendeteksian Lokasi Potensial.....	12
2.1.3.2    Pemilihan Lokasi <i>Keypoint</i> .....	14
2.1.3.3    Pemberian Orientasi .....	14
2.1.3.4    Representasi <i>Keypoint</i> .....	15
<b>BAB 3 PERANCANGAN SISTEM PENGENALAN OBJEK</b>	
<b>BERDASARKAN PENDEKATAN FITUR DENGAN METODE SIFT .....</b>	<b>17</b>
3.1    METODE PENELITIAN.....	17
3.2    PERANGKAT SIMULASI.....	18
3.3    GAMBAR <i>TRAINING</i> .....	18

3.4	DIAGRAM BLOK PERANCANGAN SIMULASI .....	19
3.4.1	<i>Preprocessing</i> .....	20
3.4.2	<i>Keypoints Detection</i> .....	20
3.4.3	<i>Keypoints Deskriptor</i> .....	21
3.4.4	<i>Matching and Decide</i> .....	21
3.5	KRITERIA EVALUASI.....	21
<b>BAB 4 HASIL SIMULASI DAN ANALISIS .....</b>		<b>23</b>
4.1	GAMBAR PADA BASIS DATA .....	23
4.1.1	<i>Gambar Training</i> .....	23
4.1.2	<i>Gambar Testing</i> .....	24
4.2	HASIL DAN ANALISIS .....	24
4.2.1	<i>Preprocessing (Proses Pendahuluan)</i> .....	24
4.2.2	<i>Proses Deteksi Keypoints</i> .....	26
4.2.3	<i>Proses Matching</i> .....	26
4.2.4	<i>Tingkat Akurasi dan Waktu Proses</i> .....	30
4.2.5	<i>Nilai Threshold Optimal</i> .....	31
4.3	ANALISIS PERFORMA METODE SIFT DALAM PENGENALAN OBJEK KAPAL PERANG .....	31
<b>BAB 5 PENUTUP .....</b>		<b>33</b>
5.1	KESIMPULAN.....	33
5.2	SARAN .....	34
<b>DAFTAR REFERENSI.....</b>		<b>35</b>

## DAFTAR GAMBAR

Gambar 2.1. Ilustrasi <i>shape space method</i> untuk objek dua dimensi [2].....	7
Gambar 2.2. Contoh gambar yang akan dijadikan gambar <i>training</i> [5].....	10
Gambar 2.3. Beberapa gambar representasi dari <i>eigenfaces</i> [5].....	10
Gambar 2.4. Ilustrasi pembentukan gambar <i>Difference of Gaussian</i> [9].....	13
Gambar 2.5. Ilustrasi nearest <i>neighbor</i> untuk penentuan <i>keypoint</i> [9].....	13
Gambar 2.6. Ilustrasi pembentukan <i>keypoint</i> deskriptor [9].....	15
Gambar 3.1. Kapal perang jenis fregat Indonesia.....	18
Gambar 3.2. Kapal perang jenis korvet Indonesia.....	18
Gambar 3.3. Kapal perang fregat Amerika.....	19
Gambar 3.4. Kapal perang <i>landing</i> Australia.....	19
Gambar 3.5. Diagram blok perancangan simulasi.....	20
Gambar 4.1. Gambar-gambar yang akan digunakan sebagai gambar <i>training</i> .....	23
Gambar 4.2. Beberapa contoh gambar uji ( <i>testing</i> ).....	24
Gambar 4.3. Grafik nilai <i>recall</i> vs <i>1-precision</i> untuk setiap nilai T.....	27
Gambar 4.4. Grafik <i>f1 score</i> untuk setiap nilai T.....	29

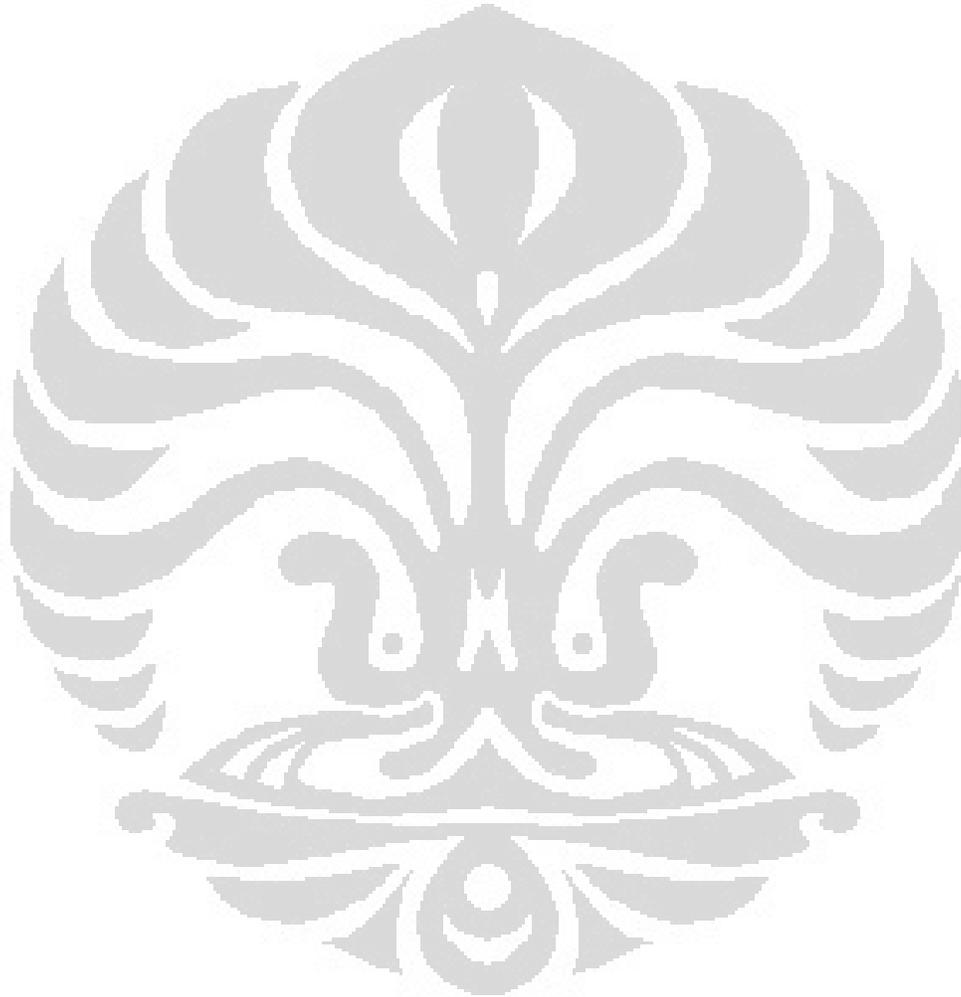
## DAFTAR TABEL

Tabel 4.1. Jumlah <i>keypoints</i> yang didapatkan pada gambar <i>training</i> .....	25
Tabel 4.2. Rata-rata jumlah <i>keypoints</i> yang didapatkan pada gambar <i>testing</i> .....	25
Tabel 4.3. Rata-rata nilai <i>recall</i> dan <i>1-precision</i> setiap jenis kapal untuk variasi nilai T .....	27
Tabel 4.4. <i>F1 score</i> setiap jenis kapal untuk variasi nilai T .....	28
Tabel 4.5. Tingkat akurasi dan waktu proses sistem pengenalan objek.....	30



## DAFTAR LAMPIRAN

Lampiran 1 : Gambar Testing .....	37
Lampiran 2 : Data Hasil Simulasi.....	41
Lampiran 3 : Kode Program.....	45



# BAB 1

## PENDAHULUAN

### 1.1 Latar Belakang

#### 1.1.1 Pengenalan Objek

Bagi manusia, mengenali objek merupakan salah satu kemampuan luar biasa yang secara alamiah muncul sejak masih anak-anak. Meskipun warna, pencahayaan, dan sudut pandangnya berubah, manusia mampu mengenali dan membedakan suatu objek dengan objek lain hanya dengan beberapa kali melihat saja. Hal ini menjadi tantangan tersendiri bagi para peneliti untuk menciptakan suatu sistem pengenalan objek yang sebaik sistem pengenalan objek manusia. Sistem yang diciptakan harus mampu mengenali objek meskipun pose gambar berubah, pencahayaan dalam ruangan berubah, dan kesulitan dalam generalisasi dari beberapa pencitraan objek yang diambil.

Sistem yang dibuat untuk mengenali objek biasanya adalah dengan menggunakan komputer, atau sistem pandangan robot (*robot vision*). Pengenalan objek adalah target yang paling penting dalam banyak aplikasi pengolahan citra dan *robot vision*. Pengenalan objek dengan menggunakan komputer telah diteliti lebih dari empat dekade dan tetap dianggap sebagai hal yang paling penting dan penuh tantangan tersendiri [1]. Berbagai peneliti dari mancanegara telah menciptakan metode-metode tersendiri dalam pengenalan objek untuk robot atau komputer. Zhang, Krim, dan Walter [2] menjelaskan bahwa banyak acuan yang dipakai peneliti untuk menciptakan metode pengenalan objek seperti warna, tekstur, gerak, dan bentuk. Pengenalan objek dengan bentuk merupakan metode yang paling banyak digunakan dalam dua puluh tahun awal penelitian mengenai pengenalan objek ini dilakukan.

Pada dasarnya pengenalan objek adalah suatu proses pemberian label kepada suatu citra visual dari objek agar diketahui identitas dari objek tersebut. Seperti telah dijelaskan sebelumnya bahwa suatu sistem pengenalan objek harus bersifat *robust* yang berarti bahwa sistem pengenalan objek harus tahan terhadap perubahan variasi tampilan objek akibat perubahan bentuk, warna,

pencapaian. Karena sudut pandang yang berbeda dari objek yang sama dapat mengubah representasi citra secara matematis.

Secara garis besar, metode pengenalan objek dibagi menjadi tiga jenis pendekatan. Metode-metode ini dibuat berdasarkan acuan yang dipakai untuk mengenali objek. Metode yang pertama adalah metode dengan pendekatan bentuk geometri. Metode ini mengenali objek dengan menentukan bentuk geometri dari objek apakah bentuk geometrinya segiempat, segilima, atau lingkaran untuk citra dua dimensi, atau kubus, prisma, limas, untuk citra tiga dimensi. Metode yang kedua adalah metode dengan pendekatan penampilan (*appearance*). Metode ini mengenali objek dengan tidak memperhatikan bentuk maupun warna dari objek, melainkan mengolah seluruh informasi citra yang terdapat dalam suatu gambar, seperti piksel dan warna. Metode yang ketiga adalah metode dengan pendekatan fitur (*feature*). Fitur yang dimaksud adalah titik-titik penting (*Interest Point*) atau wilayah-wilayah yang menjadi ciri khas dari citra suatu objek.

### 1.1.2 Objek Kapal Perang

Kapal perang merupakan salah satu alat pertahanan suatu negara. Indonesia, sebagai negara maritim memiliki resiko-resiko atau ancaman tersendiri dari sisi kelautan (*marine*). Sebagai salah satu bentuk optimalisasi pertahanan dalam negeri untuk laut adalah dengan membeli dan memiliki kapal-kapal perang dengan spesifikasi senjata, perlindungan, dan fasilitas yang baik.

Di suatu negara, terdapat beberapa tipe kapal perang yang dibeli oleh pemerintah negara tersebut atau kapal perang sebagai hibah dari negara sahabat. Kapal perang yang paling penting dalam jajaran kapal perang di suatu negara adalah kapal perang yang berada di garis depan wilayah pertahanan suatu negara. Dari beberapa kapal perang seperti kapal perang untuk persediaan senjata, kapal perang untuk mengangkut persediaan logistik, dan kapal selam, kapal tempur yang berada paling depan merupakan kapal terbaik yang menjadi ujung tombak wilayah pertahanan suatu negara.

Kapal tempur atau kapal yang berada di garis depan pertahanan suatu negara terdiri atas beberapa jenis kapal yang dibagi lagi menjadi kelas-kelas berdasarkan fungsi dan wilayah pertahanannya. Masing-masing dari kapal perang,

baik dari jenis maupun kelas, memiliki keunikan tersendiri. Namun demikian, bagi penjaga pelabuhan, atau bahkan bagi tentara angkatan laut sebagian besar tidak mengetahui jenis apa dan spesifikasi apa kapal yang dilihat. Untuk mengenali jenis kapal dari ciri-ciri fisik akan membutuhkan waktu yang cukup lama bagi mereka untuk membuka buku panduan kapal perang. Untuk itulah, dibutuhkan suatu sistem yang dapat mengenali kapal perang dari jenis dan kelasnya serta mampu menampilkan spesifikasi kapal secara jelas hanya dari gambar digital.

### 1.1.3 Pengenalan Kapal Perang

Untuk membangun suatu sistem yang mampu mengenali kapal perang, dibutuhkan suatu sistem yang mandiri, *robust*, dan akurat. Dengan kata lain, permasalahan yang memicu penelitian ini adalah :

1. Pengenalan objek merupakan suatu proses yang kompleks dan paling rumit dalam pengolahan citra, sehingga dibutuhkan suatu metode pengenalan objek yang akurat dan mudah diimplementasikan.
2. Dibutuhkan sistem pengenalan objek kapal perang yang mampu merespon secara cepat dan akurat yang dibangun dengan metode tertentu untuk keperluan pendidikan, keamanan, dan pertahanan.

## 1.2 Tujuan Penulisan

Tugas akhir ini memiliki tujuan untuk merancang suatu sistem pengenalan objek kapal perang berdasarkan pendekatan fitur dengan metode *Scale invariant feature transform*, melakukan simulasi menggunakan perangkat lunak komputer, dan menganalisis performa hasil simulasi tersebut dengan berbagai parameter kriteria evaluasi.

## 1.3 Pembatasan Masalah

Penelitian ini dilakukan dengan beberapa batasan masalah, antara lain:

1. Peneliti tidak menurunkan persamaan matematis dan menghitung persamaan matematis secara manual, persamaan matematis didapatkan dari studi literatur dan perhitungan matematis dilakukan oleh komputer.

2. Peneliti menggunakan citra gambar digital statis (bukan video) untuk pembuatan sistem pengenalan objek kapal perang.
3. Peneliti menggunakan gambar dengan ukuran yang sama antara satu gambar dengan yang lainnya untuk meminimalisasi kesalahan dan menjaga integritas sistem.
4. Peneliti tidak melakukan pengujian proses kerja dengan mengambil gambar objek kapal perang secara nyata.
5. Peneliti hanya mempertimbangkan objek kapal perang yang berada di garis depan pertahanan wilayah suatu negara.

#### **1.4 Metode Penelitian**

Metode yang dilakukan selama melakukan penelitian dan penulisan laporan adalah:

1. Studi Literatur

Peneliti membaca buku, jurnal, thesis, dan bahan-bahan lain yang berkaitan dengan ilmu pengenalan objek, pengolahan citra, pengolahan sinyal digital, dan pengenalan pola (*Pattern Recognition*)

2. Konsultasi dengan Dosen Pembimbing

Pertemuan yang digelar setiap minggu membuat peneliti dapat melakukan penelitian dan penulisan secara kontinu. Jika ada kesalahan dalam penelitian atau format penulisan, pembimbing akan memberikan arahan dan solusi untuk memperbaikinya.

3. Diskusi dengan orang yang memahami materi penelitian

Diskusi dilakukan dengan para pakar yang memahami materi penelitian.

#### **1.5 Sistematika Penulisan**

Laporan seminar ini tersusun atas 5 (lima) bab, yaitu:

1. Bab 1 : Pendahuluan

Pada bab ini penulis menjelaskan Latar Belakang, Tujuan Penulisan, Pembatasan Masalah, Metode Penelitian, dan Sistematika Penulisan.

## 2. Bab 2 : Pengenalan Objek

Pada bab ini penulis menjelaskan mengenai dasar teori dan metode-metode yang digunakan dalam pengenalan objek, pendekatan apa saja yang digunakan, serta metode SIFT yang menjadi fokus pada penelitian ini.

## 3. Bab 3: Pengenalan objek berdasarkan pendekatan fitur dengan menggunakan metode *scale invariant feature transform*.

Pada bab ini, akan dijelaskan tahapan-tahapan perancangan untuk simulasi, perangkat simulasi, gambar yang digunakan, dan kriteria evaluasi untuk menguji performa SIFT.

## 4. Bab 4 : Hasil Simulasi dan Analisis

Pada bab ini, akan disajikan hasil simulasi yang telah dilakukan, parameter evaluasi, tingkat akurasi, dan analisis performa SIFT berdasarkan hasil yang telah didapatkan.

## 5. Bab 5 : Penutup

Pada bab ini, akan disajikan kesimpulan dari penelitian yang telah dilakukan dan saran yang menjadi pertimbangan untuk penelitian selanjutnya.

## BAB 2 PENGENALAN OBJEK

### 2.1 Metode Pengenalan Objek

Pengenalan objek merupakan suatu tantangan dan juga tujuan utama dalam pengolahan citra. Tantangan yang dihadapi dalam pengenalan objek adalah bagaimana komputer atau robot secara spesifik dapat mengidentifikasi objek meskipun terdapat variasi dari tampilan objek tersebut. Variasi-variasi tersebut dapat berupa sudut pandang objek yang berubah-ubah, variasi cahaya atau iluminasi, dan perbedaan pose objek terhadap penangkap gambar.

Sebagai salah satu cabang ilmu pengetahuan yang telah dipelajari selama lebih dari 40 tahun, pengenalan objek telah dikembangkan dengan beberapa metode berdasarkan pendekatan apa yang digunakan untuk mengidentifikasi suatu objek yaitu berdasarkan bentuk geometri (*shape / geometry-based approaches*), tampilan (*appearance-based*), dan berdasarkan fitur [1].

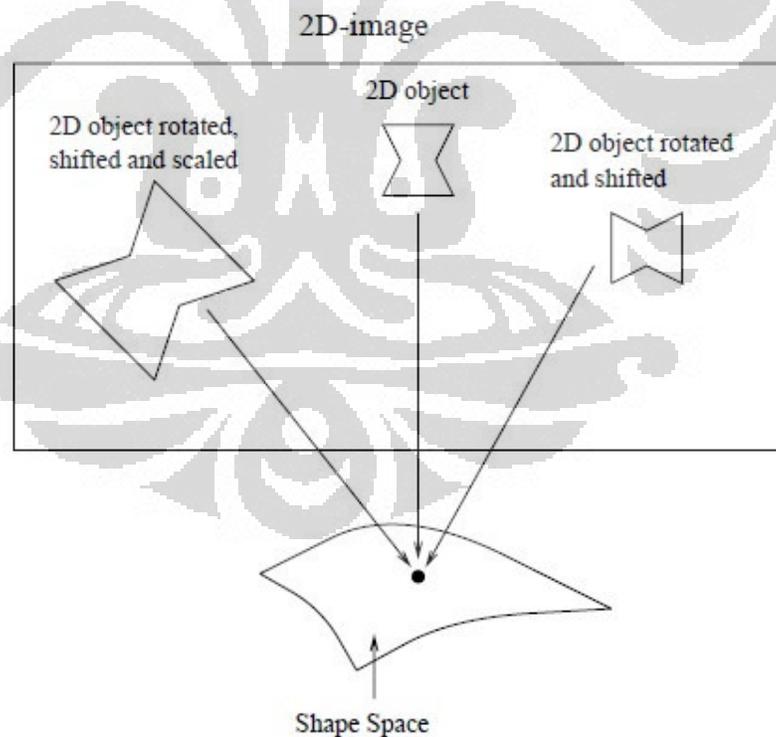
#### 2.1.1 Geometry Based Approaches

Menurut Chuang [3], bentuk geometri dari suatu objek merupakan salah satu acuan yang sering dipakai dibandingkan dengan warna, tekstur, atau pergerakan (*motion*). Metode dengan pendekatan ini merupakan metode yang paling awal muncul dalam studi dan pengembangan metode pengenalan objek [1]. Metode *Geometry Based Approaches* ini merupakan metode yang cukup sukses terutama dalam keadaan pencitraan yang dikendalikan dengan baik, contohnya adalah benda yang berada pada jalur perakitan dalam suatu pabrik.

Metode *geometry-based approaches* ini biasanya menggunakan bingkai atau tepian dari suatu objek untuk deskripsi dari suatu gambar [4]. Selain menggunakan tepi dari suatu objek, metode *geometry-based approaches* juga mengenali objek dengan ekstraksi bentuk dasar objek seperti garis lurus, lingkaran, dan bentuk dasar lainnya [1]. Metode *geometry-based approaches* yang mengenali objek dengan ekstraksi bentuk dasar seperti yang telah disebutkan di atas dapat disebut juga dengan *part decomposition method*.

Teknik *part decomposition* merupakan teknik yang menyederhanakan atau mengurai suatu objek menjadi bentuk dasar seperti kubus, silinder, prisma, dan lain sebagainya. Metode ini dapat digunakan pada objek yang sangat rumit, namun hanya dapat digunakan pada objek yang seragam. Apabila objek tidak seragam, misalnya seperti membedakan identitas mobil SUV dan Sedan maka akan ditemukan kesulitan karena bentuk dasar dari kedua identitas tersebut tidak lagi konsisten dan stabil.

Beberapa pengembangan dan perbaikan telah dilakukan untuk metode ini. salah satu pengembangan terkini adalah *shape space method* [2]. Dengan metode *shape space* ini, performa pengenalan objek dapat ditingkatkan. *Shape space method* merepresentasikan objek sebagai titik dalam sebuah permukaan multidimensi tinggi. Gambar 2.1 menunjukkan bagaimana sebuah objek direpresentasikan sebagai suatu titik dalam permukaan multidimensi tinggi meskipun telah ditranslasi, rotasi, atau dilatasi.



Gambar 2.1. Ilustrasi *Shape Space Method* untuk objek dua dimensi [2]

### 2.1.2 Appearance Based Approaches

Metode lain yang dikembangkan selain metode pengenalan objek berbasis bentuk geometri adalah pendekatan berbasis tampilan (*appearance*). Pada pendekatan ini, bentuk dari objek diabaikan, yang menjadi perhatian utama adalah mengolah dan membandingkan langsung vektor-vektor yang terdapat pada sekumpulan gambar *training* dengan gambar uji. *Appearance-based method* mengolah vektor-vektor yang merupakan representasi tiap piksel dari suatu gambar. Secara sederhana, metode ini membandingkan secara langsung seluruh elemen dari suatu gambar, mirip seperti korelasi [5].

Pendekatan *Appearance Based* ini biasanya digunakan pada pengenalan wajah (*face recognition*). Pada pengenalan wajah, sulit untuk mengimplementasikan pendekatan bentuk geometri karena pada wajah bentuk geometri semua serupa, sehingga pendekatan *Appearance Based* dianggap cocok untuk pengenalan wajah. Pengenalan wajah dengan pendekatan fitur juga sulit dilakukan karena pada wajah, titik-titik penting yang menjadi fitur biasanya serupa. Untuk dapat membedakan identitas suatu pribadi dari yang lain, perlu fitur yang menjadi ciri khas orang tersebut, sehingga metode dengan pendekatan fitur juga dirasa kurang optimal dalam pengenalan wajah.

Pengenalan wajah dengan pendekatan *Appearance Based* salah satunya adalah metode *Eigenfaces*. *Eigenfaces* merupakan metode yang mengolah eigenvector dari suatu set vektor yang ada pada suatu gambar yang merupakan representasi dari letak piksel dan level *greyscale* dari gambar tersebut. Jika terdapat beberapa gambar untuk beberapa orang yang terlebih dahulu di-*training* maka akan diperoleh sekelompok *eigenvector*.

Pengenalan objek dapat dilakukan dengan membandingkan eigenvector suatu gambar uji  $\mathbf{X}(r,c)$  yang memiliki jumlah baris  $r$  dan kolom  $c$  dengan setiap eigenvector yang terdapat pada kelompok *eigenvector* hasil *training*. Perbandingan yang dilakukan adalah dengan menemukan jarak *Euclidean* yang paling minimum atau dapat disebut juga dengan *finding nearest neighbor* antara titik pada gambar yang pertama ( $x_{1i}$ ) dengan titik pada gambar kedua ( $x_{2i}$ ). Perhitungan *Euclidean distance* dapat dilihat pada persamaan (2.1) [5].

$$d(x_1, x_2) = \left( \sum_{i=1}^{rc} (x_{1i}^2 - x_{2i}^2) \right)^{1/2} \quad (2.1)$$

Pengenalan objek dengan pendekatan *Appearance Based* biasanya berdasarkan pada konsep ruang citra (*Image Space*). Gambar dua dimensi biasanya direpresentasikan dalam suatu matriks  $I(x,y)$  dengan dimensi  $N$ . Matriks  $I$  ini disebut juga dengan ruang citra (*Image Space*) dimana koordinat  $x$  dan  $y$  menyatakan piksel dari gambar tersebut (lihat Gambar 2.2). Perhitungan dimensi ruang dari gambar tersebut dapat dilihat pada persamaan (2.2) [5]

$$N = rc \quad (2.2)$$

dimana :

$N$  = Jumlah dimensi ruang gambar

$r$  = Jumlah baris piksel pada gambar

$c$  = Jumlah kolom piksel pada gambar.

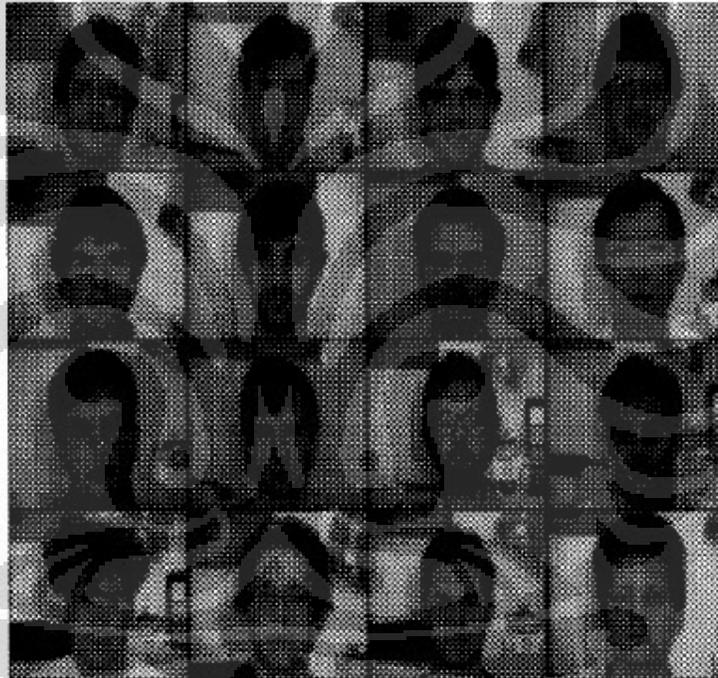
Selanjutnya matriks ini akan diubah dimensinya menjadi menjadi matriks vektor baris dengan dimensi  $1 \times N$  atau vektor kolom dengan dimensi  $N \times 1$ . Dengan representasi gambar seperti ini satu gambar akan dianggap sebagai satu vektor, dan gambar akan menjadi fitur-fitur dengan dimensi tinggi. Selanjutnya dari beberapa gambar, akan diperoleh beberapa vektor yang merupakan vektor *training*.

Untuk mengenali objek baru maka terdapat beberapa metode untuk membandingkan vektor uji dengan vektor *training* seperti *classifier*, *correlation*, dan lain sebagainya, namun yang paling sederhana adalah dengan metode *nearest neighbor*. Vektor yang diperoleh pada gambar uji akan dihitung jarak Euclidean dengan masing-masing vektor *training* dan dilihat mana yang paling dekat. Vektor *training* yang memiliki *Euclidean distance* paling dekat dengan vektor gambar uji maka objek yang diuji dapat dikatakan sama identitasnya dengan vektor *training*.

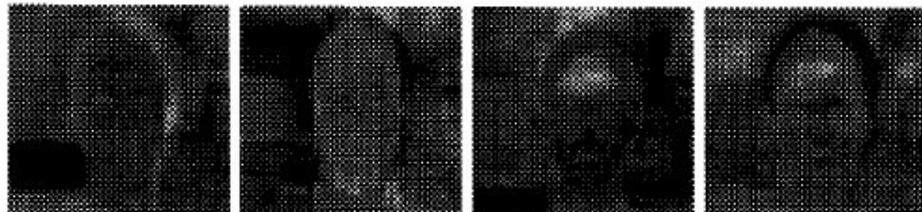
Beberapa riset mengenai metode *Appearance based* ini telah menghasilkan beberapa perkembangan dan metode tambahan atau pelengkap untuk menghasilkan metode *Appearance based* yang lebih efisien. *Classifier* atau metode menentukan *matching* antara gambar uji dengan gambar *training* telah dikembangkan seperti *Artificial Neural Network with Radial Basis Function* [6],

*Fisher Linear Discriminant* [7], *Support Vector Machines* [8], *Sparse Network of Winnows*, dan lain sebagainya.

Menurut Turk [5], *Appearance based approaches* tidak terbatas pada data mentah dari piksel gambar saja, terdapat metode yang digunakan untuk mengolah data terlebih dahulu dengan mengeliminasi elemen yang dirasa kurang merepresentasikan gambar tersebut, salah satunya adalah PCA (*Principal Component Analysis*) yang menjadi salah satu komponen *Eigenfaces*, contoh gambar *Eigenfaces* dapat dilihat pada Gambar 2.3. Semua pengembangan dan metode yang diteliti menjadikan *Appearance based approaches* semakin banyak dipakai pada pengenalan wajah karena dikenal dengan *robustness* yang baik.



Gambar 2.2. Contoh gambar yang akan dijadikan gambar *training* [5]



Gambar 2.3. Beberapa gambar representasi dari *Eigenfaces* [5]

### 2.1.3 Feature Based Approaches

Pengenalan objek dengan menggunakan pendekatan *Feature Based* merupakan salah satu metode pengenalan objek yang lebih terbaru dibandingkan metode-metode lainnya. Pendekatan *Feature based* merupakan metode yang mengenali objek dengan cara menemukan titik-titik penting (*interest point*) yang tetap (*invariant*) terhadap perubahan skala, iluminasi, atau bahkan transformasi objek melalui algoritma-algoritma tertentu [1].

Pengenalan objek dengan menggunakan pendekatan berbasis fitur ini telah banyak diteliti dan dikembangkan, salah satu di antaranya adalah dengan menggunakan Harris Corner yaitu memperoleh fitur-fitur dari titik-titik pojok sebuah objek. Metode Harris Corner ini tidak hanya memperoleh *keypoints* dari titik pojok dari sebuah objek, tetapi juga daerah-daerah yang memiliki gradien yang luas ke semua arah [9].

Salah satu metode terbaru mengenai pengenalan objek yang menggunakan pendekatan berbasis fitur adalah *Scale invariant feature transform* [9]. Metode SIFT ini merupakan metode dengan pendekatan fitur yang paling banyak digunakan dan paling sukses dengan karakteristik cepat, *invariant* terhadap skala dan rotasi, dan dapat dilakukan *training* dengan hanya satu gambar saja [10].

Metode *Scale invariant feature transform* merupakan metode yang mengadopsi sistem kerja primata dalam pengenalan objek, metode ini cocok digunakan untuk objek yang lebih kaya akan tekstur. Metode ini memiliki dua gol utama yaitu untuk mencari titik-titik penting (*interest point*) atau disebut juga titik titik kunci (*keypoints*) dan algoritma untuk pengenalan objek itu sendiri yaitu *matching* dengan *database* [1].

Pada dasarnya terdapat empat tahapan dalam metode SIFT, yaitu :

- mendeteksi lokasi potensial,
- pemilihan lokasi dan skala,
- memberikan orientasi, dan
- Representasi *keypoints* ke dalam bentuk sebuah vektor berdimensi tinggi.

Vektor-vektor yang merepresentasikan titik-titik kunci (*keypoints*) ini kemudian disimpan ke dalam *database* [9]. Untuk pengenalan objek, maka dari

gambar uji (*testing*) juga akan dicari titik-titik kuncinya, kemudian dengan algoritma tertentu, akan dilakukan *indexing* dan *matching* untuk mencari kesesuaian antara objek dalam gambar uji dengan yang ada di dalam *database* [11].

### 2.1.3.1 Pendeteksian Lokasi Potensial

Tahap pertama dalam pencarian *keypoints* adalah dengan mendeteksi lokasi *keypoints* yang potensial. *Keypoints* yang dicari haruslah yang *invariant* terhadap perubahan skala. Untuk mencari lokasi yang *invariant* diperlukan fitur yang stabil terhadap perubahan skala, menggunakan suatu fungsi yang bernama *Scale Space* [12]. Untuk memperoleh fungsi *Scale Space*  $\mathbf{L}(\mathbf{x}, \mathbf{y}, \sigma)$  maka dapat digunakan fungsi *Gaussian*  $\mathbf{G}(\mathbf{x}, \mathbf{y}, \sigma)$  dengan  $\sigma$  merupakan nilai varians fungsi Gaussian pada persamaan (2.4) [9] yang di konvolusi dengan gambar input  $\mathbf{I}(\mathbf{x}, \mathbf{y})$  seperti ditunjukkan pada persamaan (2.3) [9]. Untuk mendeteksi lokasi potensial (disebut juga *maxima* dan *minima*) maka harus dibentuk suatu gambar dengan skala ruang yang disebut *Difference of Gaussian Function*  $\mathbf{D}(\mathbf{x}, \mathbf{y}, \sigma)$ , yaitu dengan cara mengurangi *Scale Space* dengan nilai varians  $k\sigma$  dengan varians bernilai  $\sigma$  seperti ditunjukkan pada persamaan (2.5) (ilustrasi dapat dilihat pada Gambar 2.4).

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \quad (2.3)$$

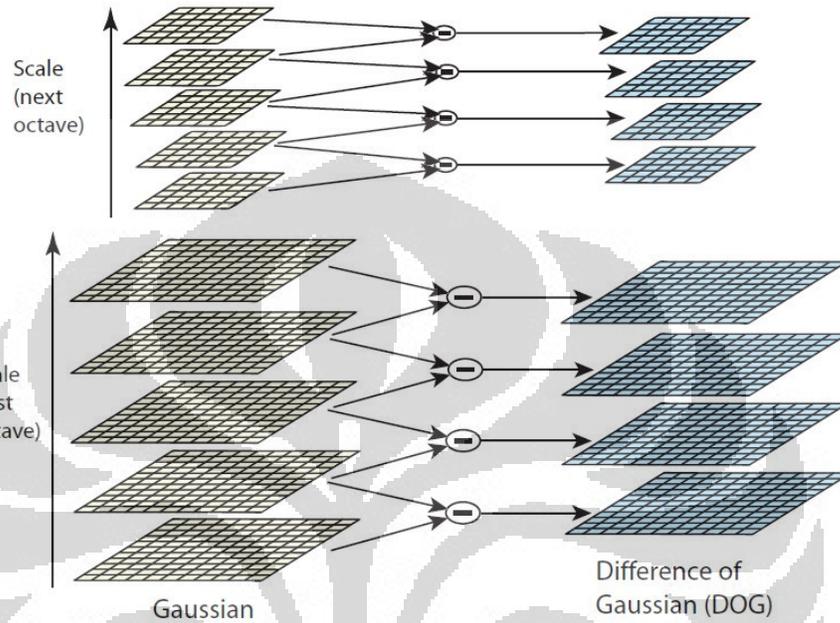
$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2} \quad (2.4)$$

$$D(x, y, \sigma) = L(x, y, k\sigma) - L(x, y, \sigma) \quad (2.5).$$

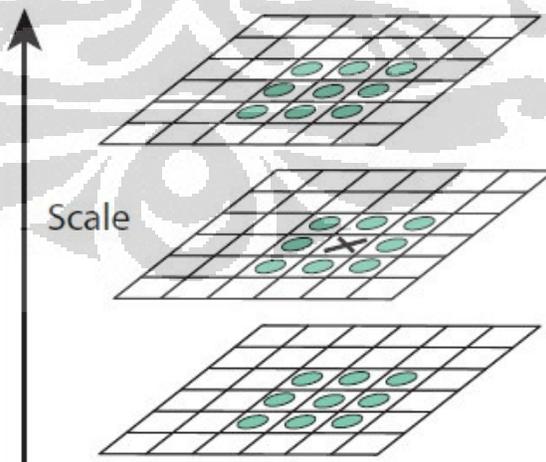
Kemudian dari persamaan (2.5) [9], akan dibentuk beberapa gambar dengan DoG sehingga beberapa gambar tersebut membentuk sebuah oktaf. Selanjutnya pembentukan oktaf baru akan dilakukan dengan *resample* gambar dengan hanya mengambil piksel genap, sehingga diperoleh gambar baru yang ditunjukkan pada gambar.

Selanjutnya untuk mencari *maxima* dan *minima* adalah dengan membandingkan suatu piksel di gambar DoG yang berada di tengah oktaf dengan piksel-piksel tetangganya seperti pada Gambar 2.5. Apabila nilai piksel ini lebih tinggi dari seluruh nilai piksel yang menjadi tetangganya, atau bahkan lebih kecil dari seluruh nilai piksel yang menjadi tetangganya, maka piksel ini dapat

dianggap sebagai lokasi yang tepat untuk *keypoint* karena piksel ini merupakan maxima atau minima dari gambar tersebut.



Gambar 2.4. Ilustrasi pembentukan gambar Difference of Gaussian [9]



Gambar 2.5. Ilustrasi nearest *neighbor* untuk penentuan *keypoint* [9]

### 2.1.3.2 Pemilihan Lokasi *Keypoint*

Setelah ditemukan lokasi-lokasi yang menjadi *keypoint*, maka harus ditentukan *keypoint* mana saja yang akan benar-benar dipilih untuk proses selanjutnya. Dalam SIFT, semakin banyak *keypoint*, bukan semakin baik. *Keypoint-keypoint* yang ditemukan, di antaranya banyak yang merupakan *keypoint* yang tidak merepresentasikan gambar yaitu *keypoint* yang sensitif terhadap *noise*. Salah satu caranya adalah dengan menentukan fungsi bobot nilai extrema yang berada dalam skala ruang  $\mathbf{D}(\mathbf{x}, \mathbf{y}, \sigma)$  dan berlokasi pada titik  $\hat{x}$ . Fungsi  $\mathbf{D}(\hat{x})$  yang merupakan fungsi bobot dari sebuah extrema dapat dilihat pada persamaan (2.6) [9].

$$D(\hat{x}) = D + \frac{1}{2} \frac{\delta D^T}{\delta x} \hat{x} \quad (2.6).$$

Apabila bobot dari extrema kurang dari suatu nilai maka statusnya sebagai *keypoint* akan ditolak. Pada SIFT, biasanya dipakai nilai 0.03 sebagai minimum bobot ekstrema.

### 2.1.3.3 Pemberian Orientasi

Orientasi merupakan suatu representasi *keypoint* yang paling nyata dalam suatu gambar. Orientasi ini merupakan besar dan arah suatu *keypoint*. Pemberian orientasi ini bertujuan untuk membentuk *keypoint* yang *invariant* terhadap rotasi. Orientasi yang diberikan kepada gambar Gaussian  $\mathbf{L}(\mathbf{x}, \mathbf{y}, \sigma)$  adalah berupa gradien  $\mathbf{m}(\mathbf{x}, \mathbf{y})$  dan sudut  $\theta(\mathbf{x}, \mathbf{y})$  seperti pada persamaan (2.7) [9] dan (2.8) [9]. Selanjutnya, akan dibentuk histogram sebanyak 36 bin yang menyatakan  $360^\circ$  arah vektor, maka setiap bin akan merepresentasikan  $10^\circ$  arah. Histogram ini memiliki bobot sebesar gradien *magnitude* pada persamaan (2.6). Puncak tertinggi dalam histogram mewakili orientasi sebenarnya dari *keypoint* tersebut, biasanya tiga nilai tertinggi yang masih berkisar 80% juga dianggap sebagai orientasi dominan.

$$m(x, y) = \sqrt{(L(x+1, y) - L(x-1, y))^2 + (L(x, y+1) - L(x, y-1))^2} \quad (2.7)$$

$$\theta(x, y) = \tan^{-1} \left( \frac{L(x, y+1) - L(x, y-1)}{L(x+1, y) - L(x-1, y)} \right) \quad (2.8)$$

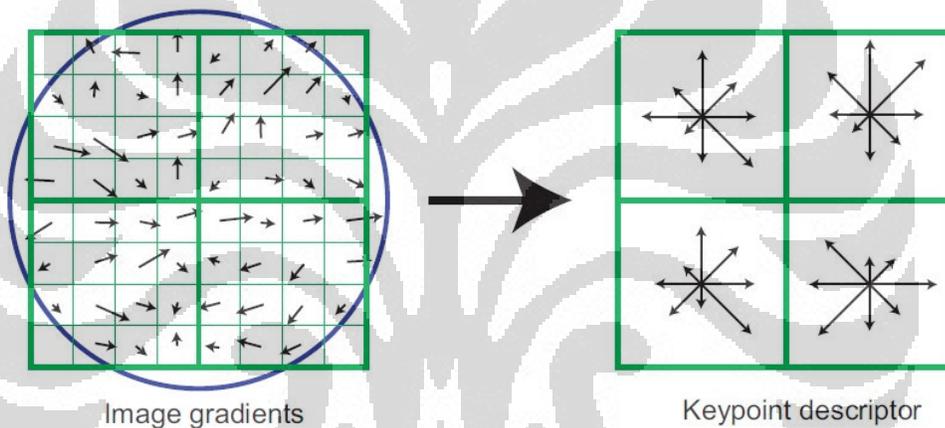
dimana :

$m(x,y)$  = besar *magnitude* orientasi

$\theta(x,y)$  = besar sudut orientasi.

#### 2.1.3.4 Representasi *Keypoint*

*Keypoint* yang telah dihitung lokasi, skala, dan orientasinya akan dibentuk menjadi sebuah vektor yang merupakan fitur dari gambar atau disebut juga dengan deskriptor. Vektor ini akan dibentuk dari gradien *magnitude* dan orientasi yang telah dibahas pada bagian sebelumnya. Gradien dan orientasi akan dihitung pada area di sekitar lokasi *keypoint*. Pada penelitian ini, gradien dan orientasi yang dihitung adalah sebanyak  $16 \times 16$  sampel. Dari  $16 \times 16$  sampel ini,

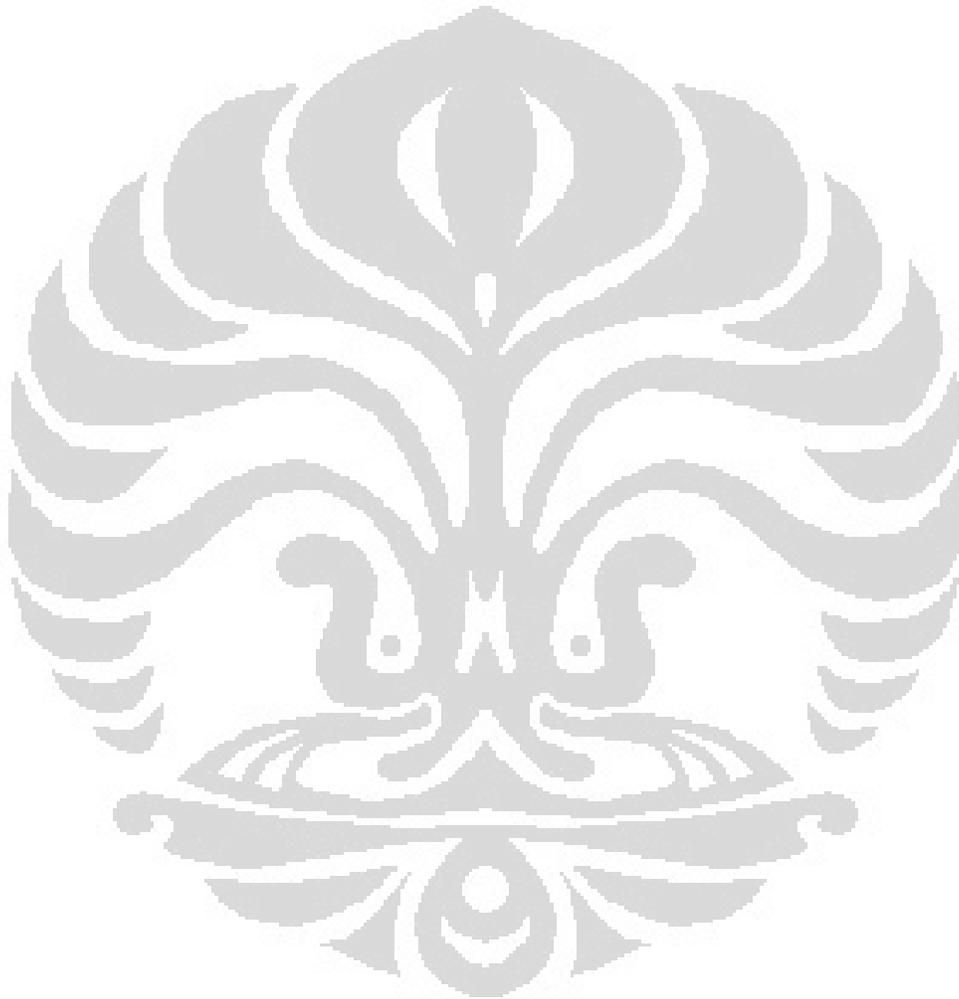


Gambar 2.6. Ilustrasi pembentukan *Keypoint* Deskriptor [9]

akan dibagi ke dalam wilayah  $4 \times 4$  yang disebut dengan bin, sehingga terdapat 16 ( $4 \times 4$ ) bin. Dari setiap bin, akan dibentuk histogram yang merupakan akumulasi dari gradien *magnitude* dan dibagi ke dalam 8 arah (utara-timur laut-timur-tenggara-selatan-barat daya-barat-barat laut). Sebagai totalnya, maka akan terdapat  $4 \times 4 \times 8 = 128$  elemen vektor, atau dengan kata lain vektor dengan 128 dimensi. Ilustrasi representasi *keypoint* ini dapat dilihat pada Gambar 2.6.

Setelah dilakukan pemberian orientasi, *keypoint-keypoint* tadi akan direpresentasikan dalam bentuk vektor berdimensi tinggi yang kemudian akan disimpan di dalam *database*. Sebuah gambar *testing* akan mengalami proses yang sama seperti di atas hingga diperoleh vektor berdimensi tinggi yang merupakan fitur dari gambar tersebut. Selanjutnya akan dilakukan matching vektor *testing* dengan vektor yang terdapat pada *database training*, Metode-metode tambahan

digunakan dalam mencocokkan fitur di dalam gambar *testing* dengan *database* untuk mencapai sistem matching yang lebih bersifat *robust* seperti *Best Bin First* [11], *Hough Transform* [13] dan *Least Square Method*. Peneliti sebelumnya mengungkapkan bahwa sebanyak tiga fitur yang cocok cukup untuk mendapatkan sistem pengenalan objek yang baik [14].



**BAB 3**  
**PERANCANGAN SISTEM PENGENALAN OBJEK BERDASARKAN**  
**PENDEKATAN FITUR DENGAN MENGGUNAKAN METODE *SCALE***  
***INVARIANT FEATURE TRANSFORM***

**3.1 Metode Penelitian**

Dalam penelitian mengenai pengenalan objek berdasarkan pendekatan fitur ini, akan digunakan metode *Scale Invariant Feature Transform* (SIFT). Seperti yang telah dijelaskan pada sub-bab 2.1.3 bahwa pengenalan objek dengan metode ini dilakukan melalui beberapa tahapan yang cukup kompleks dan memerlukan analisis matematis yang cukup sulit, namun dengan bantuan perangkat lunak komputer, kompleksitas dan kesulitan diharapkan akan berkurang.

Penelitian yang dilakukan akan terbagi menjadi dua bagian besar yaitu pra-penelitian dan penelitian utama. Pra-penelitian merupakan tahapan untuk mempersiapkan gambar-gambar yang menjadi gambar *training* untuk sebuah sistem pengenalan objek sampai penyimpanan dalam *database*. Penelitian utama, yang menjadi fokus dalam penelitian ini adalah bagaimana proses pengenalan objek dilakukan, yang berarti apabila ada sebuah gambar *testing* yang telah mengalami perubahan sudut pandang, rotasi, nilai kekontrasan, dan perubahan skala.

Penelitian sebelumnya yang dilakukan oleh Lowe [14] memiliki ketahanan terhadap beberapa faktor di atas, namun terdapat keterbatasan pada iluminasi dan perubahan geometri yang sebagian masih dapat dikenali. Penelitian ini diharapkan mampu memperbaiki masalah tersebut dengan menggunakan parameter yang berbeda dengan peneliti sebelumnya, seperti nilai varians untuk fungsi Gaussian, rasio antara *closest neighbor* dengan *second-closest neighbor* pada bagian matching gambar *testing*, atau bahkan mereduksi waktu kerja yang dibutuhkan untuk menyelesaikan tahapan-tahapan yang dilakukan dibandingkan dengan penelitian sebelumnya.

### 3.2 Perangkat Simulasi

Perancangan program dan simulasi akan dilakukan dengan bantuan perangkat lunak MATLAB dengan menggunakan SIFT *Toolboxes* yang dirancang oleh David G. Lowe [14] dengan beberapa tambahan sub-program dari penulis untuk proses *matching* antara gambar *training* dan gambar *testing*.

Perangkat keras yang digunakan adalah Notebook Toshiba L510 dengan prosesor Intel Pentium Dual Core T4500 2,3 GHz dan memori RAM 1 GB.

### 3.3 Gambar *Training*

Pada penelitian ini, akan digunakan gambar kapal perang dengan empat jenis kapal perang yang berbeda. Masing-masing jenis kapal memiliki satu gambar untuk di *training*, contoh gambar kapal yang akan digunakan dalam proses *training* dapat dilihat pada Gambar 3.1, Gambar 3.2, Gambar 3.3, dan Gambar 3.4.



Gambar 3.1. Kapal Perang Jenis Fregat Indonesia



Gambar 3.2. Kapal Perang Jenis Korvet Indonesia



Gambar 3.3. Kapal Perang Fregat Amerika



Gambar 3.4. Kapal Perang *Landing* Australia

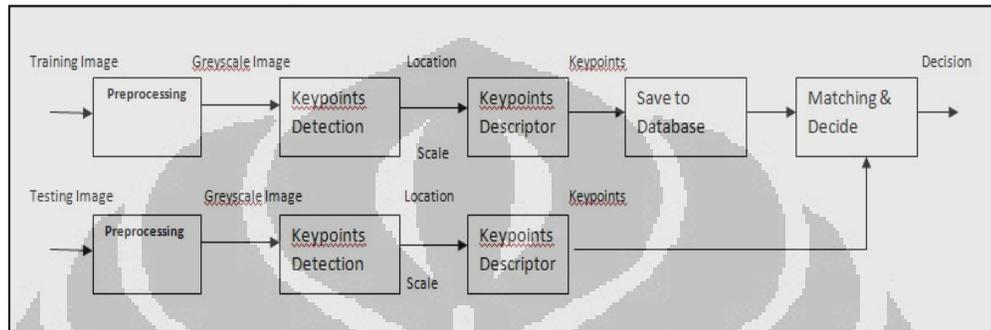
### 3.4 Diagram Blok Perancangan Simulasi

Pada penelitian ini, akan dibuat program simulasi pengenalan objek dengan menggunakan bantuan software MATLAB. Program simulasi akan dibuat dalam bentuk M-File. Persamaan-persamaan matematis dan fungsi khusus akan dikerjakan dengan bantuan *tools* yang ada dalam MATLAB sehingga tidak dibutuhkan penghitungan manual.

Pada simulasi pengenalan objek, akan dikenali empat jenis objek yaitu kapal perang jenis Fregat Indonesia, kapal perang jenis Korvet, Kapal Fregat Amerika, dan kapal *Landing* Australia. Masing-masing jenis kapal memiliki empat buah kapal yang berbeda, sehingga terdapat empat gambar dengan level iluminasi, sudut pandang, dan letak kapal yang berbeda. Satu dari masing-masing jenis akan di-*training* untuk *database*, dan tiga dari masing-masing jenis akan digunakan untuk *testing*.

Empat gambar untuk *training* akan diolah terlebih dahulu untuk disamakan ukuran pikselnya sehingga gambar seragam dan lebih mudah untuk

diolah. Keempat gambar untuk *training* ini selanjutnya akan dicari *keypoints*nya untuk kemudian disimpan ke dalam *database*. Sedangkan ke-duabelas gambar *testing*, akan dicari pula titik pentingnya untuk kemudian menjalani tahap *matching* dengan titik penting dari gambar *training* yang telah disimpan di dalam *database*. Pada Gambar 3.5, dapat dilihat diagram blok perancangan simulasi program pengenalan objek.



Gambar 3.5. Diagram Blok Perancangan Simulasi

### 3.4.1 Preprocessing

Pada simulasi pengenalan objek, akan dikenali empat jenis objek yaitu kapal perang jenis fregat Indonesia, kapal perang jenis korvet Indonesia, kapal perang fregat Amerika, dan kapal perang *landing* Australia. Masing-masing jenis kapal memiliki empat gambar dengan level iluminasi, sudut pandang, dan letak kapal yang berbeda. satu dari masing-masing jenis akan di-*training* untuk *database*, dan tiga dari masing-masing jenis akan digunakan untuk *testing*.

Seluruh gambar untuk *training* maupun *testing* akan diolah terlebih dahulu untuk disamakan ukuran pikselnya sehingga gambar seragam dan lebih mudah untuk diolah. Setelah disamakan ukuran file, akan dilakukan perubahan warna gambar dari RGB menjadi *grayscale* dengan menggunakan program MATLAB.

### 3.4.2 Keypoints Detection

Pada langkah ini, akan dilakukan deteksi *keypoint* dengan menentukan, lokasi, skala dan orientasi dari titik-titik yang berpotensi sebagai *keypoints*. Sesuai dengan tahapan teori pada sub-bab 2.1.3.1 sampai 2.1.3.4 baik untuk gambar *testing* dan gambar *training*.

### 3.4.3 Keypoints Deskriptor

Pada langkah ini, akan didapatkan *keypoint-keypoint* yang akan direpresentasikan dalam bentuk deskriptor. Kemudian untuk gambar *training*, deskriptor akan disimpan ke dalam *database*.

### 3.4.4 Matching and Decide

Pada langkah ini, deskriptor dari gambar *training* dan gambar *testing* akan dicocokkan. Pencocokan dilakukan penentuan Euclidean Distance antara *keypoints* dari gambar *training* dengan gambar *testing* seperti pada persamaan (2.1) kemudian dilanjutkan dengan metode membandingkan *closest neighbor* dengan *second closest neighbor*. Kemudian setelah dilihat perbandingan tersebut maka akan diambil keputusan apakah gambar *testing* cocok dengan gambar *training*, tidak cocok dengan gambar *training*, atau tidak dikenali berdasarkan jumlah *keypoints* yang *match*. Keputusan cocok akan ditentukan ketika nilai perbandingan lebih kecil daripada nilai *threshold*. Pada penelitian ini nilai *threshold* divariasikan dengan nilai 0.65-0.8 dengan kenaikan 0.05.

## 3.5 Kriteria Evaluasi

Kriteria yang digunakan pada penelitian kali ini sebagai parameter untuk mengevaluasi sistem adalah dengan menggunakan grafik *recall vs 1-precision* [15] dan *F1 Score* [16]. Grafik *recall vs 1-precision* sudah menjadi parameter yang digunakan untuk analisis dan evaluasi performa metode SIFT dalam pengenalan objek. *Recall* merupakan perbandingan antara jumlah *correct matches* dengan jumlah *correspondences* seperti ditunjukkan pada persamaan (3.1) [15]. *1-precision* merupakan perbandingan antara jumlah *false matches* dengan jumlah *total matches* seperti pada persamaan (3.2) [15]. *Correspondences* merupakan jumlah *keypoints* yang ada pada gambar *training*, *total matches* merupakan jumlah *keypoint* yang cocok dengan nilai perbandingan *neighbor* berada di bawah nilai *threshold*, *correct matches* merupakan jumlah *keypoint* yang cocok dan berada pada lokasi fisik yang sama, sedangkan *false matches* merupakan jumlah *keypoint* yang cocok namun tidak berada pada kondisi fisik yang sama.

Menurut Treen [16], pengujian performa SIFT dengan menggunakan *recall* dan *1-precision* terkadang menimbulkan kesulitan karena pada beberapa kasus pengenalan objek nilai *recall* dapat menjadi lebih baik daripada nilai *1-precision*. Pengujian performa metode SIFT dengan dua variabel seperti ini juga menimbulkan kesulitan dalam menentukan apakah metode SIFT dapat mengenali suatu objek dengan baik. Oleh karena itu dibuat suatu parameter pengujian yang memiliki suatu nilai skalar tanpa harus membandingkan dua variabel yaitu dengan *F1 Score*. *F1 Score* ini merepresentasikan kombinasi linear dari *recall* dan *1-precision* seperti ditunjukkan persamaan (3.3) [16].

$$Recall = \frac{\#correct\ matches}{\#correspondences} \quad (3.1)$$

$$1 - precision = \frac{\#false\ matches}{\#total\ matches} \quad (3.2)$$

$$F1\ Score = \frac{2 \times precision \times recall}{(precision + recall)} \quad (3.3).$$

Selain parameter pengujian yang telah dijelaskan di atas, pada penelitian kali ini juga akan dilihat berbagai parameter lain seperti tingkat akurasi dalam persentase dan waktu proses sehingga dapat dianalisis pula keterkaitan parameter-parameter tersebut dengan variasi nilai *threshold*.

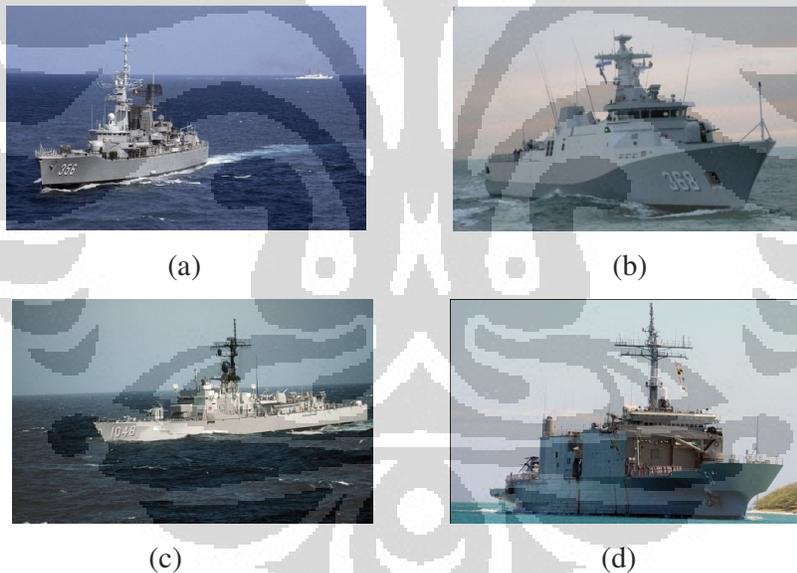
## BAB 4

### HASIL SIMULASI DAN ANALISIS

#### 4.1 Gambar pada Basis Data

##### 4.1.1 Gambar *Training*

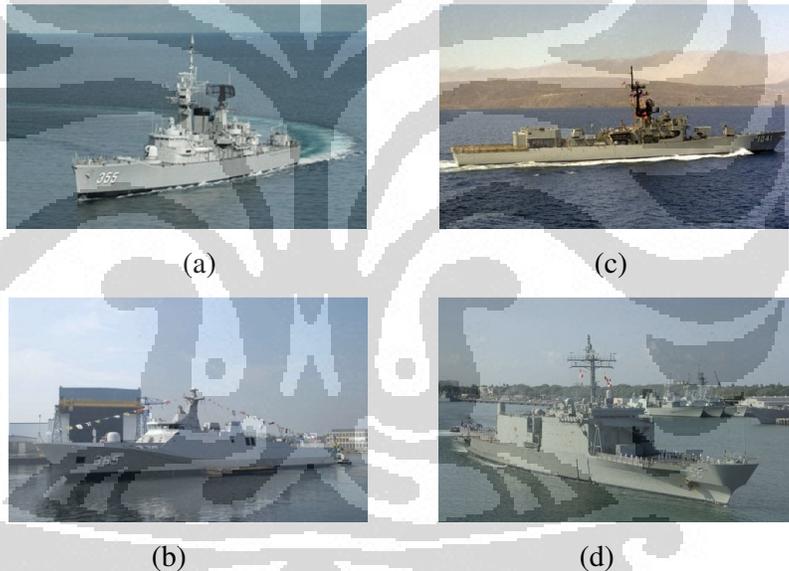
Pada penelitian ini digunakan gambar *training* yang didapatkan dari internet yaitu situs khusus angkatan laut Republik Indonesia, Amerika Serikat, serta Australia. Gambar *training* yang didapatkan berukuran 600 x 375 piksel. Ada empat tipe kapal yang akan dikenali dalam penelitian kali ini yaitu kapal fregat Republik Indonesia, kapal korvet Republik Indonesia, kapal fregat Amerika Serikat, dan kapal *landing* Australia. Terdapat empat gambar *training* masing-masing mewakili tipe kapal tersebut. Empat buah gambar kapal yang menjadi gambar *training* dapat dilihat pada Gambar 4.1.



Gambar 4.1. Gambar-Gambar yang akan digunakan sebagai Gambar *Training*,  
(a) Fregat Indonesia, (b) Korvet Indonesia, (c) Fregat Amerika, (d) *Landing*  
Australia.

#### 4.1.2 Gambar *Testing*

Sesuai dengan yang telah dijelaskan pada sub-bab 3.4 bahwa dua belas gambar lain yang terdiri atas empat jenis kapal yang berbeda sehingga masing-masing jenis kapal memiliki tiga gambar yang digunakan sebagai gambar uji (*testing*). Gambar-gambar *testing* ini juga diperoleh dari internet yaitu dari situs-situs resmi angkatan laut Republik Indonesia, angkatan laut Amerika Serikat, dan angkatan laut Australia. Gambar-gambar yang akan diuji tentunya memiliki tingkat iluminasi, orientasi, sudut pandang, dan nilai skala yang berbeda sehingga pengenalan kapal perang dengan metode SIFT dapat benar-benar diuji. Beberapa contoh gambar yang digunakan sebagai gambar uji dapat dilihat pada Gambar 4.2. Gambar uji lainnya dapat dilihat pada Lampiran 1.



Gambar 4.2. Beberapa Contoh Gambar Uji, (a) Fregat Indonesia, (b) Korvet Indonesia, (c) Fregat Amerika, (d) *Landing* Australia.

## 4.2 Hasil dan Analisis

### 4.2.1 Preprocessing (Proses Pendahuluan)

Pre-processing yang dimaksud adalah dengan menyeragamkan ukuran piksel dan mengubah gambar berwarna (RGB) menjadi hitam putih (*grayscale*). Gambar-gambar yang didapatkan dari internet pada awalnya sudah memiliki ukuran piksel dengan porsi yang mirip, namun terdapat beberapa gambar yang

kelebihan piksel pada bagian lebar (*width*) sehingga perlu di-*crop* dengan menggunakan perangkat sederhana yaitu *Paint*. Perlu diingat bahwa proses cropping tidak menghilangkan piksel yang mewakili kapal perang.

Setelah dilakukan proses penyeragaman ukuran gambar, selanjutnya dilakukan perubahan dari gambar berwarna menjadi gambar hitam putih dengan menggunakan perangkat lunak MATLAB. Fungsi yang digunakan pada perangkat lunak adalah *rgb2gray*. Dengan fungsi ini, maka gambar berwarna akan diubah ke dalam bentuk *greyscale* untuk selanjutnya dapat diolah. Proses perubahan gambar dari gambar berwarna menjadi gambar hitam putih ini dilakukan karena proses pencarian *keypoints* yang terdapat dalam *SIFT Toolboxes* yang dibuat oleh peneliti pertama dari metode ini hanya dapat mengolah matriks gambar hitam putih.

#### 4.2.2 Proses Deteksi *Keypoints*

Proses penentuan dan deteksi dengan menggunakan *Toolboxes* yang dibuat oleh peneliti pendahulu berdasarkan algoritma *SIFT* berhasil dilakukan. Jumlah *keypoints* yang didapatkan dari gambar *training* dapat dilihat pada Tabel 4.1, sedangkan rata-rata jumlah *keypoints* yang didapatkan dari gambar *testing* dapat dilihat pada Tabel 4.2.

Tabel 4.1. Jumlah *keypoints* yang didapatkan pada gambar *training*.

No.	Jenis Kapal	Jumlah <i>Keypoints</i>
1	Fregat Republik Indonesia	732
2	Korvet Republik Indonesia	749
3	Fregat Amerika Serikat	559
4	<i>Landing Ship</i> Australia	691

Tabel 4.2. Rata-rata jumlah *keypoints* yang didapatkan pada gambar *testing*.

No.	Jenis Kapal	Jumlah <i>Keypoints</i>
1	Fregat Republik Indonesia	1373
2	Korvet Republik Indonesia	591
3	Fregat Amerika Serikat	858
4	<i>Landing Ship</i> Australia	855

#### 4.2.3 Proses *Matching*

Seperti telah dijelaskan pada sub-bab 3.5, performa metode SIFT dalam pengenalan objek kapal perang akan dievaluasi dengan menggunakan dua parameter yaitu variabel *recall* vs *1-precision* dan *F1 Score*. Berdasarkan persamaan (3.1) dan (3.2), nilai *recall* merupakan perbandingan antara jumlah *correct matches* dengan *correspondences* yang artinya menunjukkan seberapa mirip gambar *testing* dengan gambar *testing*. Semakin besar nilai *recall* maka suatu gambar dapat dikatakan semakin mungkin mendekati citra aslinya. Namun variabel *recall* ini tidak dapat berdiri sendiri, terdapat variabel *1-precision* untuk menjadi variabel yang memiliki keterkaitan dengan *recall*. Nilai *1-precision* merupakan perbandingan antara jumlah *false matches* dengan total *matches* yang artinya menunjukkan tingkat akurasi dari sistem pengenalan objek. Semakin besar nilai *1-precision* menunjukkan *keypoints* yang cocok semakin kurang presisi.

Keterkaitan dua parameter *recall* vs *1-precision* terkadang menyulitkan peneliti untuk menentukan mana yang lebih penting. Untuk itulah dibuat suatu parameter yaitu *F1 Score*. *F1 Score* merupakan suatu parameter skalar yang dapat berdiri sendiri sehingga performa metode SIFT dalam pengenalan suatu objek dapat dinilai secara kuantitatif. Semakin besar nilai *F1 Score* akan menunjukkan performa yang semakin baik.

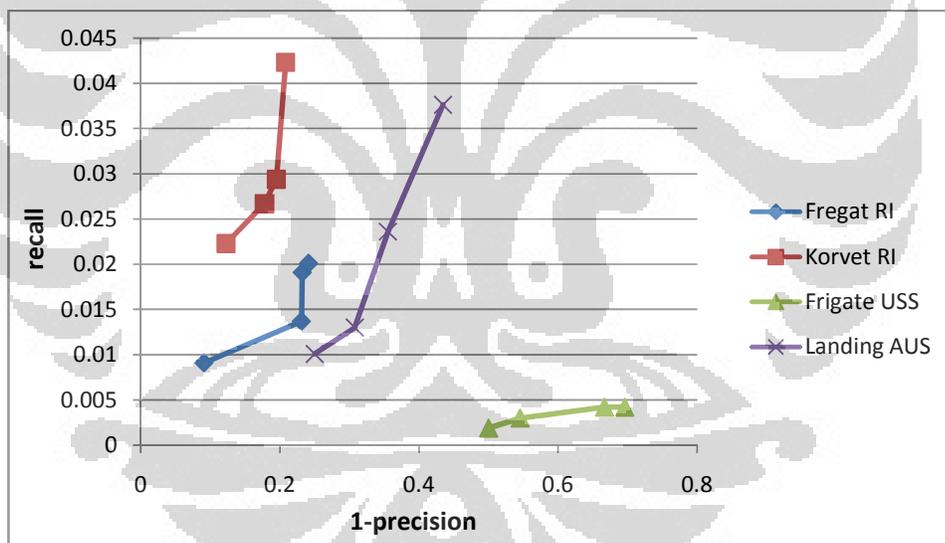
Nilai *recall*, *1-precision*, dan *F1 Score* secara lengkap dari masing-masing gambar dapat dilihat pada bagian Lampiran 2. Dari hasil pengenalan objek dengan metode SIFT yang dilakukan peneliti, terdapat tiga hasil keluaran dari program yaitu YA, SALAH, dan TIDAK. Hasil keluaran YA berarti gambar *testing* berhasil dikenali sesuai dengan tipe pada gambar *training*. Hasil keluaran SALAH berarti gambar *testing* berhasil dikenali namun tidak sesuai dengan tipe pada gambar *training*, sedangkan hasil keluaran TIDAK berarti gambar tidak berhasil dikenali karena memang tidak ada kecocokan *keypoints*.

Dari ketiga hasil keluaran program, persentase akurasi dari sistem pengenalan objek akan dihitung dengan membandingkan jumlah gambar yang berhasil dikenali dan sesuai dengan tipenya dengan keseluruhan jumlah gambar. Sedangkan nilai *recall* dan *1-precision* hanya dihitung dari gambar yang berhasil dikenali dan sesuai dengan tipenya. Nilai rata-rata *recall* dan *1-precision* masing-

masing jenis kapal sesuai dengan variasi nilai *threshold* dapat dilihat pada Tabel 4.3 dan plot grafik ini ditunjukkan pada Gambar 4.3.

Tabel 4.3. Rata-rata nilai *recall* dan *1-precision* setiap jenis kapal untuk variasi nilai T.

No	Threshold (T)	Fregat RI		Korvet RI		Fregat USS		Land-AUS	
		<i>1-precision</i>	<i>recall</i>	<i>1-precision</i>	<i>recall</i>	<i>1-precision</i>	<i>recall</i>	<i>1-precision</i>	<i>recall</i>
1	0.65	0.0909	0.0091	0.1228	0.0223	0.5	0.0019	0.25	0.0101
2	0.7	0.2308	0.0137	0.1781	0.0267	0.5454	0.0030	0.3077	0.0116
3	0.75	0.2321	0.0191	0.1951	0.0294	0.6667	0.0042	0.3553	0.0236
4	0.8	0.2413	0.020	0.2083	0.0423	0.6956	0.0042	0.4348	0.0376



Gambar 4.3. Grafik nilai *recall* vs *1-precision* untuk setiap nilai T.

Dari grafik yang ditunjukkan pada Gambar 4.3 serta nilai *recall* dan *1-precision* yang ada pada Tabel 4.3 dapat dilihat bahwa nilai *recall* dan *1-precision* akan sama-sama naik ketika nilai *threshold* (T) semakin besar. Hal ini dapat dilihat secara langsung dari Gambar 4.3 yaitu nilai *recall* akan semakin ke atas dan *1-precision* akan semakin ke kanan seiring dengan kenaikan nilai *threshold*. Hal ini dapat disebabkan karena metode perbandingan *neighbor* akan semakin

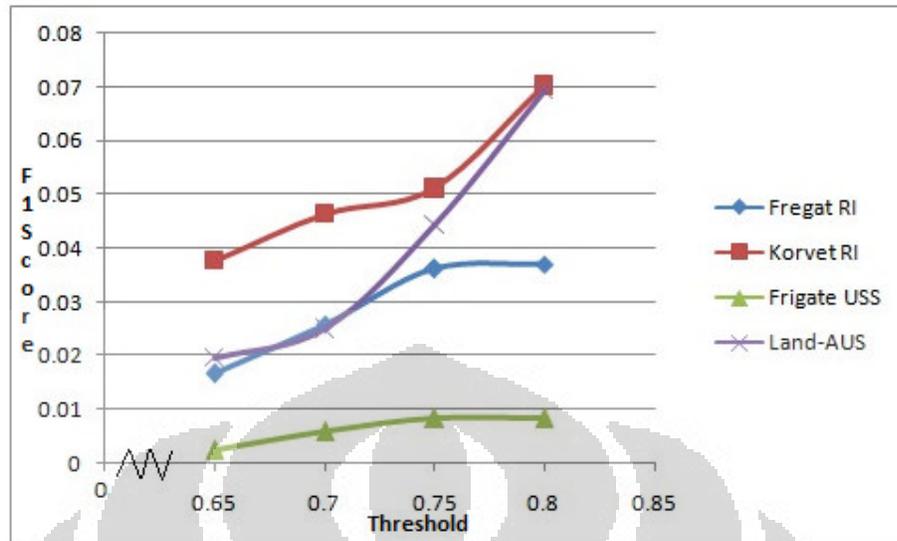
toleratif terhadap perbandingan *closest neighbor* dengan *second closest neighbor* sehingga semakin banyak *keypoints* yang akan dianggap cocok meskipun memiliki lokasi yang berbeda. Ketika semakin banyak *keypoints* yang dianggap cocok karena perbandingan tersebut semakin ditoleransi oleh nilai *threshold*, maka semakin banyak pula *keypoints* yang dianggap cocok namun memiliki lokasi berbeda (*false matches*). Hal ini tentunya akan mempengaruhi nilai *recall* dan *1-precision* karena ketika semakin banyak *matches* maka gambar *training* akan semakin mendekati gambar *testing* namun seiring semakin banyaknya *false matches* maka akurasi dari sistem pengenalan objek tersebut akan semakin turun sehingga nilai *1-precision* akan semakin meningkat.

Berdasarkan grafik pada Gambar 4.3 dapat diamati bahwa akurasi proses matching yang paling baik adalah pada jenis kapal perang korvet Republik Indonesia karena pada setiap *threshold*, nilai *1-precision*nya paling rendah dan di sisi lain nilai *recall*nya paling tinggi apabila dibandingkan dengan jenis kapal lain. Sebaliknya, kapal perang jenis fregat milik Amerika Serikat memiliki tingkat akurasi yang paling rendah dengan nilai *1-precision* yang paling tinggi dan nilai *recall* yang paling rendah.

Selain nilai *recall* dan *1-precision*, parameter skalar keduanya yang menjadi acuan baru untuk menguji performa SIFT pada sistem pengenalan objek kapal perang yaitu *F1 Score* dapat dilihat pada Tabel 4.4. dan grafik yang menunjukkan *F1 Score* ditunjukkan pada Gambar 4.4.

Tabel 4.4. *F1 Score* setiap jenis kapal untuk variasi nilai T.

No	<i>Threshold</i> (T)	Fregat RI	Korvet RI	Fregat USS	Land-AUS
		<i>F1 Score</i>	<i>F1 Score</i>	<i>F1 Score</i>	<i>F1 Score</i>
1	0.65	0.0166	0.0377	0.0024	0.0195
2	0.7	0.0258	0.0464	0.0059	0.0225
3	0.75	0.0361	0.0511	0.0083	0.0443
4	0.8	0.0370	0.0703	0.0083	0.0693



Gambar 4.4. Grafik *F1 Score* untuk setiap nilai *T*.

Dapat dilihat pada Tabel 4.4 bahwa nilai *F1 Score* akan semakin meningkat untuk setiap nilai *T*. Hal ini disebabkan karena semakin besar nilai *T* maka jumlah *matches* akan semakin banyak karena perbandingan *neighbor* akan semakin toleratif. Hal ini akan menyebabkan nilai *recall* dan *1-precision* semakin meningkat, oleh sebab itu *F1 Score* akan semakin meningkat. Nilai *F1 Score* yang semakin meningkat menunjukkan sistem pengenalan objek semakin baik dalam jumlah *matches* yang semakin banyak.

Dengan mengamati grafik pada Gambar 4.4, jenis kapal perang dengan nilai *F1 Score* tertinggi adalah jenis kapal korvet RI dan jenis kapal perang dengan nilai *F1 Score* terendah adalah jenis kapal fregat USS. Hal ini menunjukkan bahwa jenis kapal perang korvet RI memiliki hasil pengenalan yang lebih baik dengan tingkat akurasi yang lebih tinggi dibanding tiga jenis kapal perang lainnya. Hal ini mengacu kepada penelitian sebelumnya [16] yaitu bahwa *F1 Score* yang lebih tinggi akan semakin menunjukkan sistem pengenalan objek yang semakin baik. Hal ini sesuai dengan hasil keluaran grafik *recall* vs *1-precision* yaitu kapal perang jenis korvet RI memiliki tingkat akurasi yang paling tinggi dalam sistem pengenalan objek dengan metode SIFT.

#### 4.2.4 Tingkat Akurasi dan Waktu Proses

Tingkat akurasi dari sistem pengenalan objek dengan metode SIFT dihitung dengan membandingkan hasil keluaran bahwa kapal dikenali dan cocok dengan tipenya (YA) dengan seluruh jumlah gambar kapal. Pada penelitian kali ini, waktu proses juga dihitung untuk setiap tingkat akurasi dari masing-masing nilai *threshold* dalam persentase akan disajikan dalam Tabel 4.5.

Tabel 4.5. Tingkat Akurasi dan Waktu Proses Sistem Pengenalan Objek.

No	<i>Threshold</i> (T)	YA	TIDAK	SALAH	Tingkat Akurasi (%)	Waktu Proses (s)
1	0.65	8	4	0	66.67%	2.537
2	0.7	10	0	2	83.33%	2.448
3	0.75	9	0	3	75%	2.459
4	0.8	7	0	5	58.33%	2.468

Pada Tabel 4.5 nilai *threshold* menunjukkan tingkat akurasi yang paling tinggi adalah nilai *threshold* 0.7 yaitu dengan tingkat akurasi 83.33 %. Dengan tingkat akurasi sebesar 83.33 %, sistem pengenalan objek dengan metode SIFT dapat dikatakan baik untuk aplikasi pengenalan kapal perang. Dari Tabel 4.5 juga dapat dilihat waktu proses yang paling singkat adalah pada nilai *threshold* 0,7 meskipun selisih waktu proses dengan nilai *threshold* yang lain hanya sekitar 0.01 detik.

Dari pengamatan terhadap waktu proses dan tingkat akurasi, dapat dilihat bahwa semakin akurat proses pengenalan objek maka akan semakin cepat waktu proses yang diperlukan oleh sistem. Hal ini dapat disebabkan karena apabila nilai *threshold* merupakan nilai *threshold* nilai akurasinya tinggi, waktu yang dibutuhkan sistem untuk membandingkan dan menemukan setiap pasangan deskriptor akan lebih sedikit daripada waktu dengan nilai *threshold* yang kurang akurat karena sistem membutuhkan lebih banyak waktu untuk mencari pasangan deskriptor yang sesuai.

Waktu proses yang dibutuhkan oleh masing-masing proses untuk setiap nilai *threshold* tidak terlalu berbeda secara signifikan, apabila dilihat pada Tabel

4.5 waktu proses berkisar antara 2.448 sampai 2.537 detik sehingga waktu proses rata-rata sistem adalah 2.478 detik. Apabila sistem pengenalan objek digunakan untuk skala besar dengan gambar yang diolah jumlahnya semakin banyak dan gambar lebih kompleks sehingga *keypoints* yang didapat akan semakin banyak maka waktu proses dapat berjalan lebih lama. Namun dengan perangkat simulasi yang lebih mumpuni dan memiliki spesifikasi lebih tinggi, maka waktu proses dapat direduksi.

#### 4.2.5 Nilai *Threshold* Optimal

Apabila dilihat dari tingkat akurasi dan waktu prosesnya dari Tabel 4.5, maka nilai *threshold* yang paling optimal adalah senilai 0.7. Tingkat akurasi pada nilai *threshold* 0.7 adalah yang paling tinggi dengan persentase 83.33% dan waktu proses yang diperoleh adalah yang paling singkat yaitu 2.448 detik. Meskipun berdasarkan parameter *F1 Score* nilai *threshold* yang paling baik adalah 0.8, akan tetapi apabila dilihat nilai *1-precisionnya* yang paling tinggi, hal ini menunjukkan bahwa tingkat akurasinya semakin rendah. Oleh karena itu, penulis memilih nilai *threshold* 0.7 sebagai nilai *threshold* optimal.

### 4.3 Analisis Performa Metode SIFT dalam Pengenalan Objek Kapal Perang

Berdasarkan penjabaran hasil dan analisis seperti yang telah dijelaskan pada sub-bab sebelumnya, sistem pengenalan objek dengan menggunakan metode *Scale invariant feature transform* dapat memberikan performa yang baik dalam mengenali berbagai jenis kapal perang. Dengan tingkat akurasi yang mencapai 83.33%, sistem pengenalan objek dengan metode SIFT ini dapat dikatakan cukup berhasil dalam mengenali gambar kapal perang yang cukup bervariasi dari segi tingkat iluminasi, sudut pandang, dan skala. Sesuai dengan keunggulan SIFT yaitu cukup *robust* terhadap perubahan ketiga variabel tersebut, maka metode SIFT ini cocok digunakan untuk aplikasi pengenalan objek kapal perang yang memang dalam aplikasinya nanti memiliki variasi yang luas terhadap ketiga variabel tersebut oleh karena faktor cuaca dan posisi kapal. Selain itu, keunggulan lain dari SIFT yaitu dapat diandalkan meskipun hanya menggunakan satu gambar *training*

dapat menjadi alasan kuat metode ini cocok untuk pengenalan objek yaitu dari sisi efisiensi memori untuk basis data (*database*).

Meskipun dapat dikatakan baik dan cocok untuk aplikasi kapal perang, penggunaan metode SIFT pada penelitian kali ini juga memiliki beberapa kelemahan terutama pada bagian pencarian *keypoints*. Berdasarkan analisis penulis kelemahan-kelemahan ini dapat disebabkan oleh beberapa faktor yaitu:

1. Tidak adanya proses segmentasi yang dilakukan.

Dari hasil simulasi, dapat dilihat bahwa terdapat banyak *keypoints* yang didapatkan bukan pada daerah kapal perang, melainkan pada bagian lautan atau latar belakang dari gambar tersebut.

2. Ketidakteragaman ukuran gambar yang diperoleh.

Oleh karena keterbatasan sumber, gambar-gambar hanya diperoleh dari situs-situs resmi angkatan laut negara yang bersangkutan sehingga tidak diketahui gambar-gambar dari situs tersebut apakah telah mengalami kompresi atau pengolahan sebelumnya sehingga bobot piksel telah berubah. Hal ini dapat mengakibatkan kesalahan pada perolehan *keypoints*.

3. Penghitungan *correct matches* dan *false matches* dilakukan manual.

Penghitungan *correct matches* dan *false matches* dilakukan secara manual sehingga sangat mungkin terjadi *error parallax*. Hal ini dapat mengakibatkan data yang didapatkan menjadi kurang presisi.

Kelemahan-kelemahan di atas tentunya dapat diperbaiki pada penelitian selanjutnya sehingga didapatkan hasil yang lebih baik. Dengan melakukan sedikit penambahan atau perbaikan yang akan disebutkan pada bagian Saran pada bab V, tingkat akurasi sangat mungkin mencapai lebih dari 90% dan nilai parameter evaluasi akan menjadi lebih baik.

## BAB 5

### PENUTUP

#### 5.1 Kesimpulan

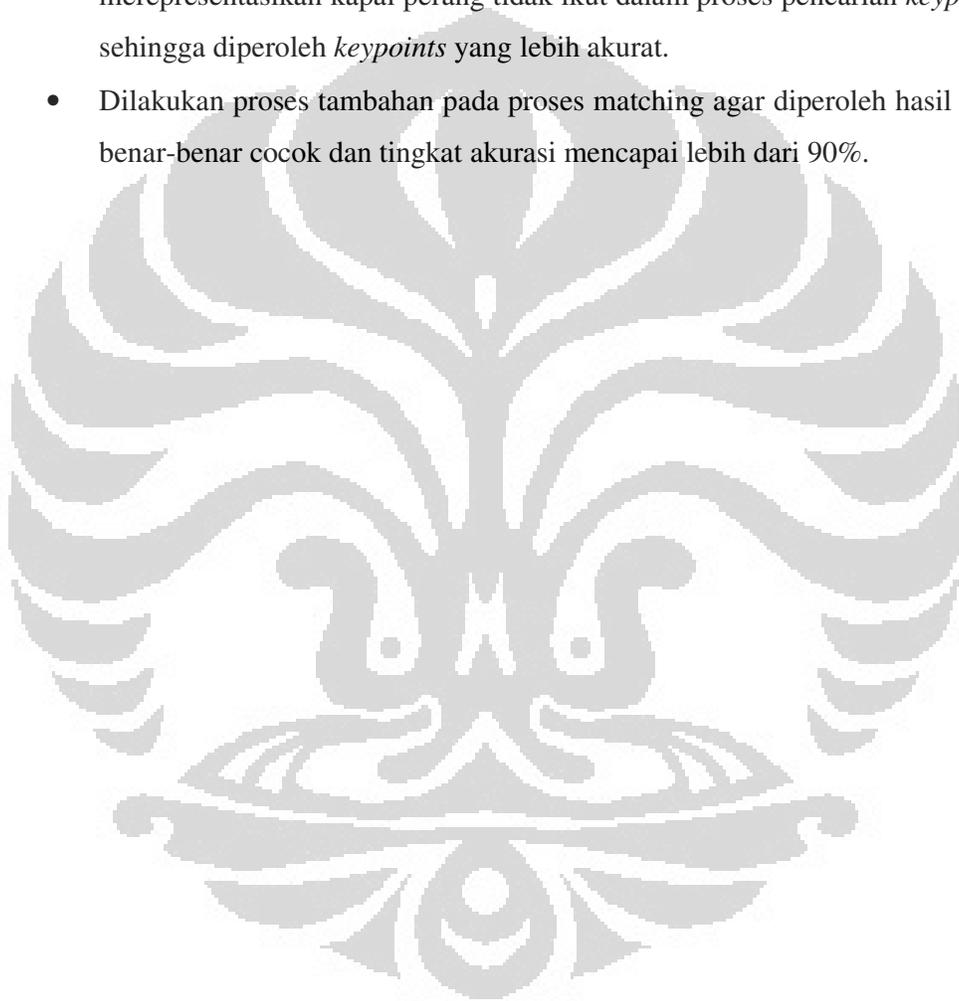
Berdasarkan hasil simulasi dan analisis, dapat disimpulkan bahwa:

- Semakin tinggi nilai *recall*, maka gambar *testing* akan dianggap semakin cocok dengan gambar *training*, semakin tinggi nilai *1-precision* maka proses *matching* akan memiliki tingkat akurasi semakin rendah. Sedangkan semakin tinggi *F1 Score* maka sistem pengenalan objek akan dikatakan semakin baik dan dapat dianggap semakin cocok dengan gambar *training*.
- Nilai *threshold* akan mempengaruhi akurasi dan tingkat kecocokan gambar *training* dan gambar *testing* yang secara kuantitatif ditunjukkan dengan perubahan ketiga parameter evaluasi. Nilai optimal *threshold* dalam hasil simulasi kali ini adalah 0.7 dengan tingkat akurasi sebesar 83.33%.
- Tingkat akurasi dan kecocokan pada proses *matching* akan mempengaruhi waktu proses. Sistem dengan tingkat akurasi dan kecocokan tinggi cenderung memiliki waktu proses yang lebih singkat meskipun tidak terlalu signifikan. Rata-rata waktu proses dalam hasil simulasi ini adalah sekitar 2,478 detik.
- Metode SIFT menunjukkan performa yang baik untuk sistem pengenalan objek kapal perang karena memiliki keunggulan dalam perubahan tingkat iluminasi, skala, dan sudut pandang (orientasi).

## 5.2 Saran

Beberapa saran dari penulis untuk penelitian selanjutnya adalah:

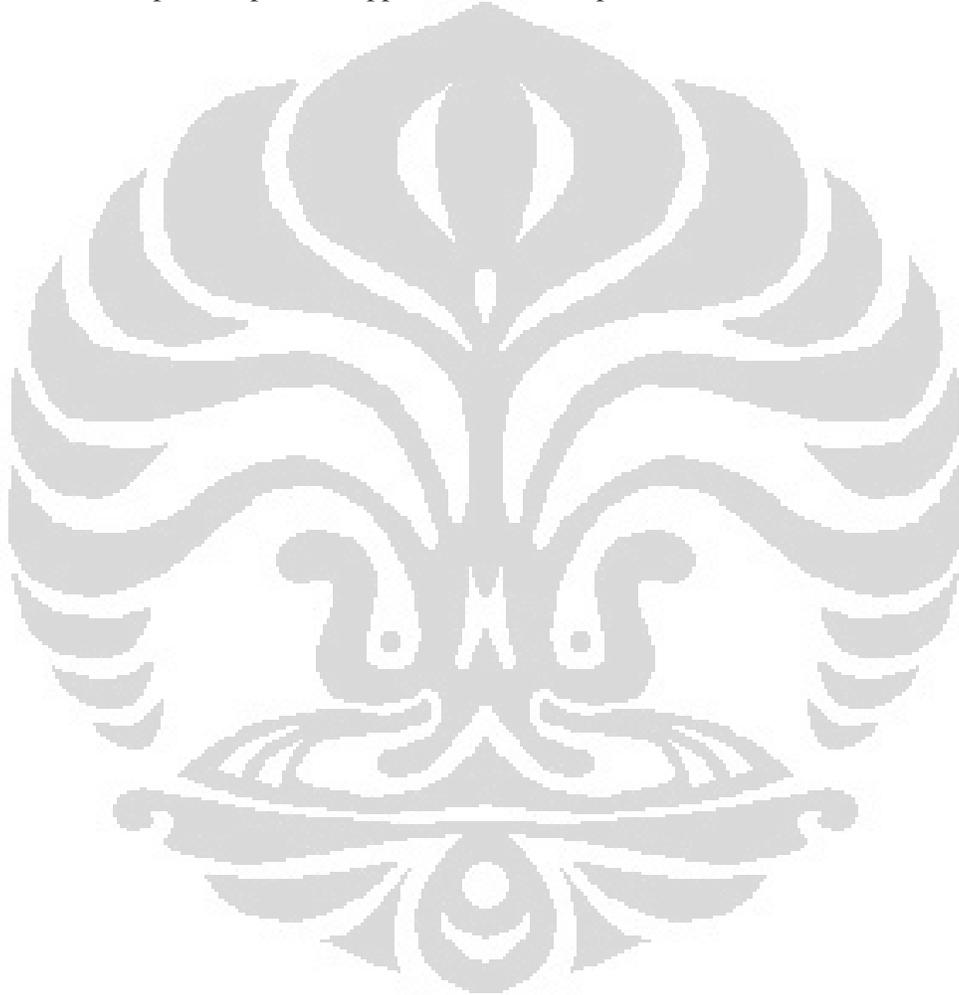
- Gambar yang diperoleh sebaiknya gambar yang berasal dari satu sumber dan memiliki ukuran piksel yang memang sudah default sama. Akan lebih baik jika menggunakan kamera atau devais penangkap gambar.
- Sebaiknya dilakukan proses segmentasi agar bagian-bagian yang tidak merepresentasikan kapal perang tidak ikut dalam proses pencarian *keypoints* sehingga diperoleh *keypoints* yang lebih akurat.
- Dilakukan proses tambahan pada proses matching agar diperoleh hasil yang benar-benar cocok dan tingkat akurasi mencapai lebih dari 90%.



## DAFTAR REFERENSI

1. Yang, Ming-Hsuan. (2007). *Object Recognition*. University of California at Merced.
2. Zhang, J., et al. (2003). *Object Representation and Recognition in Shape Spaces*. The Journal of The Pattern Recognition Society. 36, 1143-1154.
3. Chuang, Jen Hui, et al. (2001). *Shape matching and recognition using a physically based object model*. Journal of Computer and Graphics, 25, 211-222
4. Li, Wenjing. (2008). *3-D Object Recognition Using 2-D Views*. IEEE Transactions on Image Processing, Vol. 17, No. 11
5. Turk, Matthew. (2001). *A Random Walk Through Eigenspace*. IEICE Trans. Inf. & Syst., Vol.E84-D, No.12.
6. Roth,D., M.-H. Yang, and N. Ahuja.(2002). *Learning to recognize objects*. Neural Computation, 14(5):1071-1104.
7. Belhumeur, P., J. Hespanha, and D. Kriegman. (1997). *Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection*. IEEE Transactions on Pattern Analysis and Machine Intelligence, 19(7):711-720.
8. Pontil, M. and A. Verri. (1998). *Support vector machines for 3D object recognition*. IEEE Transactions on Pattern Analysis and Machine Intelligence, 20(6):637-646.
9. Lowe, David G. (2004). *Distinctive image features from scale-invariant keypoints*. International Journal of Computer Vision, 60(2):91-110.
10. Heisele, Bernd & Carlos Rocha. (2005). *Local Shape Features for Object Recognition*. Honda Research Institute, USA.
11. Beis, Jeffrey S. & David G. Lowe. (1997). *Shape Indexing Using Approximate Nearest-Neighbour Search in High-Dimensional Spaces*. Vancouver, Canada.
12. Witkin, A.P. 1983. Scale-space filtering. In International Joint Conference on Artificial Intelligence, Karlsruhe, Germany, pp. 1019-1022.
13. Hough, P.V.C. 1962. Method and means for recognizing complex patterns. U.S. Patent 3069654.

14. Lowe, D.G. (1999). Object *recognition from local scale-invariant features*. In International Conference on Computer Vision, Corfu, Greece, pp. 1150-1157.
15. K. Mikolajczyk K. and C. Schmid, 2005. A Performance Evaluation of Local Descriptors. IEEE transactions on pattern analysis and machine intelligence, vol. 27, no. 10.
16. Treen, G.; Whitehead, A.2009. Efficient SIFT Matching for *Keypoints* Descriptor Properties. Applications of Computer Vision, Ottawa, Kanada.



**LAMPIRAN 1 : GAMBAR TESTING**

Kapal Perang Jenis Fregat Republik Indonesia



## Kapal Perang Jenis Korvet Republik Indonesia



Kapal Perang Jenis Fregat Amerika Serikat (USS)



Kapal Perang Jenis *Landing* Australia



## LAMPIRAN 2 : DATA HASIL SIMULASI

### Jenis Kapal Perang Fregat Republik Indonesia

T = 0.65

No.	Cocok?	Total Matches	Correct Matches	False Matches	Correspondences	Waktu Proses	Recall	<i>1-precision</i>	F1 Score
1	YA	15	14	1	732	2.756	0.0191	0.0667	0.0297
2	TIDAK	0	0	0	732	1.915	0	0	0
3	YA	7	6	1	732	4.257	0.0082	0.1429	0.0155
	Mean	7.3333	6.6667	0.6667	732	2.976	0.0091	0.0909	0.0166

T = 0.7

No.	Cocok?	Total Matches	Correct Matches	False Matches	Correspondences	Waktu Proses	Recall	<i>1-precision</i>	F1 Score
1	YA	24	20	4	732	2.696	0.0273	0.1667	0.0469
2	SALAH	0	0	0	732	1.813	0	0	0
3	YA	11	8	3	732	4.856	0.0109	0.2727	0.0210
	Mean	11.6667	9.3333	2.3333	732	3.1217	0.0128	0.2	0.0239

T = 0.75

No.	Cocok?	Total Matches	Correct Matches	False Matches	Correspondences	Waktu Proses	Recall	<i>1-precision</i>	F1 Score
1	YA	34	29	5	732	2.774	0.0396	0.147058 824	0.0624 193
2	YA	9	4	5	732	2.081	0.0055	0.5556	0.0108
3	YA	22	14	8	732	4.367	0.0191	0.3636	0.0363
	Mean	21.6667	15.6667	6	732	3.074	0.0214	0.2769	0.0397

T = 0.8

No.	Cocok?	Total Matches	Correct Matches	False Matches	Correspondences	Waktu Proses	Recall	<i>1-precision</i>	F1 Score
1	YA	58	45	13	732	2.876	0.0615	0.2241	0.0965
2	SALAH	0	0	0	732	1.854	0	0	0
3	YA	44	28	16	732	4.676	0.0382	0.3636	0.0692
	Mean	34	24.3333	9.6667	732	3.1353	0.0332	0.2843	0.0595

### Jenis Kapal Perang Korvet Republik Indonesia

T = 0.65

No.	Cocok?	Total Matches	Correct Matches	False Matches	Correspondences	Waktu Proses	Recall	1-precision	F1 Score
1	YA	6	2	4	749	2.556	0.0027	0.6667	0.0053
2	YA	51	48	3	749	1.908	0.0641	0.0588	0.0613
3	TIDAK	0	0	0	749	2.394	0	0	0
	Mean	19	16.6667	2.3333	749	2.286	0.0223	0.1228	0.0377

T = 0.7

No.	Cocok?	Total Matches	Correct Matches	False Matches	Correspondences	Waktu Proses	Recall	1-precision	F1 Score
1	YA	8	2	6	749	1.831	0.0026	0.75	0.0053
2	YA	62	57	5	749	2.026	0.0761	0.0806	0.0783
3	YA	3	1	2	749	2.213	0.0013	0.6667	0.0027
	Mean	24.3333	20	4.3333	749	2.0233	0.0267	0.1781	0.0464

T = 0.75

No.	Cocok?	Total Matches	Correct Matches	False Matches	Correspondences	Waktu Proses	Recall	1-precision	F1 Score
1	YA	11	4	7	749	1.772	0.0053	0.6364	0.0106
2	YA	76	67	9	749	1.726	0.0893	0.1184	0.1019
3	YA	6	1	5	749	1.773	0.0013	0.8333	0.0027
	Mean	31	24	7	749	1.757	0.032	0.2258	0.0561

T = 0.8

No.	Cocok?	Total Matches	Correct Matches	False Matches	Correspondences	Waktu Proses	Recall	1-precision	F1 Score
1	SALAH	0	0	0	749	1.768	0	0	0
2	YA	98	85	13	749	1.811	0.1135	0.1327	0.1223
3	YA	14	2	12	749	1.792	0.0027	0.8571	0.0053
	Mean	37.3333	29	8.3333	749	1.7903	0.0387	0.2232	0.0660

## Jenis Kapal Perang Fregat Amerika Serikat (USS)

T = 0.65

No.	Cocok?	Total Matches	Correct Matches	False Matches	Correspondences	Waktu Proses	Recall	1-precision	F1 Score
1	YA	4	2	2	559	2.378	0.0036	0.5	0.0071
2	TIDAK	0	0	0	559	2.37	0	0	0
3	TIDAK	0	0	0	559	2.208	0	0	0
	Mean	1.3333	0.6667	0.6667	559	2.319	0.0012	0.5	0.0024

T = 0.7

No.	Cocok?	Total Matches	Correct Matches	False Matches	Correspondences	Waktu Proses	Recall	1-precision	F1 Score
1	YA	6	2	4	559	2.232	0.0036	0.6667	0.0071
2	SALAH	0	0	0	559	2.364	0	0	0
3	YA	5	3	2	559	1.96	0.0054	0.4	0.0106
	Mean	3.6667	1.6667	2	559	2.1853	0.0030	0.5454	0.0059

T = 0.75

No.	Cocok?	Total Matches	Correct Matches	False Matches	Correspondences	Waktu Proses	Recall	1-precision	F1 Score
1	YA	16	7	9	559	2.782	0.0125	0.5625	0.0245
2	SALAH	0	0	0	559	2.026	0	0	0
3	YA	5	0	5	559	1.955	0	1	0
	Mean	7	2.3333	4.6667	559	2.2543	0.0042	0.6667	0.0083

T = 0.8

No.	Cocok?	Total Matches	Correct Matches	False Matches	Correspondences	Waktu Proses	Recall	1-precision	F1 Score
1	SALAH	0	0	0	559	2.683	0	0	0
2	SALAH	0	0	0	559	2.032	0	0	0
3	YA	23	7	16	559	2.256	0.0125	0.6957	0
	Mean	7.6667	2.3333	5.3333	559	2.3237	0.0042	0.6956	0.0083

Jenis Kapal Perang *Landing* Australia (AUS)

T = 0.65

No.	Cocok?	Total Matches	Correct Matches	False Matches	Correspondences	Waktu Proses	Recall	1-precision	F1 Score
1	YA	4	2	2	691	2.666	0.0029	0.5	0.0058
2	YA	10	9	1	691	2.326	0.0130	0.1	0.0230
3	YA	14	10	4	691	2.707	0.0145	0.2857	0.0275
	Mean	9.3333	7	2.3333	691	2.566	0.0101	0.25	0.0195

T = 0.7

No.	Cocok?	Total Matches	Correct Matches	False Matches	Correspondences	Waktu Proses	Recall	1-precision	F1 Score
1	SALAH	0	0	0	691	2.223	0	0	0
2	YA	18	13	5	691	2.198	0.0188	0.2778	0.0352
3	YA	21	11	10	691	2.66	0.0159	0.4762	0.0308
	Mean	13	8	5	691	2.360	0.0116	0.3077	0.0225

T = 0.75

No.	Cocok?	Total Matches	Correct Matches	False Matches	Correspondences	Waktu Proses	Recall	1-precision	F1 Score
1	YA	15	10	5	691	2.294	0.0145	0.3333	0.0277
2	YA	30	20	10	691	2.175	0.0289	0.3333	0.0533
3	YA	31	19	12	691	2.698	0.0275	0.3871	0.0513
	Mean	25.3333	16.3333	9	691	2.389	0.0236	0.3553	0.0443

T = 0.8

No.	Cocok?	Total Matches	Correct Matches	False Matches	Correspondences	Waktu Proses	Recall	1-precision	F1 Score
1	YA	37	23	14	691	1.768	0.0333	0.3783	0.0612
2	YA	48	25	23	691	2.185	0.0362	0.4792	0.0673
3	YA	53	30	23	691	2.755	0.0434	0.4339	0.0789
	Mean	46	26	20	691	2.236	0.0376	0.4348	0.0693

### LAMPIRAN 3 : KODE PROGRAM

Berikut dilampirkan kode program simulasi yang digunakan pada penelitian kali ini:

- Program sift\_data.m

Program ini berperan untuk mencari *keypoints* yang terdapat pada setiap gambar.

```
% [image, descriptors, locs] = sift(imageFile)
%
% This function reads an image and returns its SIFT keypoints.
% Input parameters:
%   imageFile: the file name for the image.
%
% Returned:
%   image: the image array in double format
%   descriptors: a K-by-128 matrix, where each row gives an invariant
%       descriptor for one of the K keypoints. The descriptor is a vector
%       of 128 values normalized to unit length.
%   locs: K-by-4 matrix, in which each row has the 4 values for a
%       keypoint location (row, column, scale, orientation). The
%       orientation is in the range [-PI, PI] radians.
%
% Credits: Thanks for initial version of this program to D. Alvaro and
%         J.J. Guerrero, Universidad de Zaragoza (modified by D. Lowe)

function [image, descriptors, locs] = sift_data(imageFile)

% Load image
image = imread(imageFile);

% If you have the Image Processing Toolbox, you can uncomment the following
% lines to allow input of color images, which will be converted to grayscale.
if isrgb(image)
image = rgb2gray(image);
end

[rows, cols] = size(image);

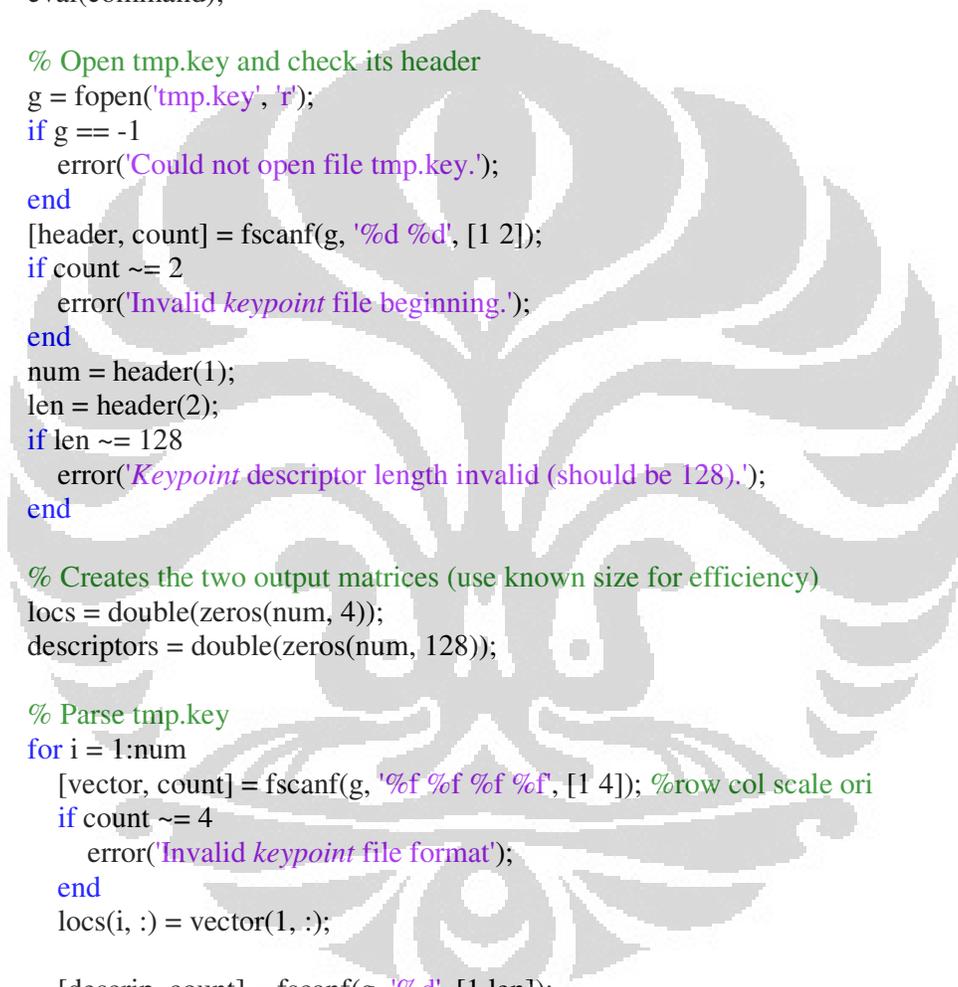
% Convert into PGM imagefile, readable by "keypoints" executable
f = fopen('tmp.pgm', 'w');
if f == -1
    error('Could not create file tmp.pgm.');
```

```

fclose(f);

% Call keypoints executable
if isunix
    command = './sift ';
else
    command = '!siftWin32 ';
end
command = [command ' <tmp.pgm >tmp.key'];
eval(command);

% Open tmp.key and check its header
g = fopen('tmp.key', 'r');
if g == -1
    error('Could not open file tmp.key.');
```



```

end
[header, count] = fscanf(g, '%d %d', [1 2]);
if count ~= 2
    error('Invalid keypoint file beginning.');
```

```

end
num = header(1);
len = header(2);
if len ~= 128
    error('Keypoint descriptor length invalid (should be 128).');
```

```

end

% Creates the two output matrices (use known size for efficiency)
locs = double(zeros(num, 4));
descriptors = double(zeros(num, 128));

% Parse tmp.key
for i = 1:num
    [vector, count] = fscanf(g, '%f %f %f %f', [1 4]); %row col scale ori
    if count ~= 4
        error('Invalid keypoint file format');
```

```

    end
    locs(i, :) = vector(1, :);

    [descrip, count] = fscanf(g, '%d', [1 len]);
    if (count ~= 128)
        error('Invalid keypoint file value.');
```

```

    end
    % Normalize each input vector to unit length
    descrip = descrip / sqrt(sum(descrip.^2));
    descriptors(i, :) = descrip(1, :);
end
fclose(g);

```

- `key_training.m`

Program ini berperan untuk mengambil *keypoints* dari gambar *training* kemudian disimpan ke dalam *database*.

*%Program untuk mencari dan mengambil keypoints dari gambar training*

```
clear all
clc;

x = 4;
for ctr = 1:x
    switch ctr
        case 1
            [imtrain1,desctrain1,loctrain1] = sift_data('training1.jpg');%mengambil
            keypoints dari gambar training 1
        case 2
            [imtrain2,desctrain2,loctrain2] = sift_data('training2.jpg');%mengambil
            keypoints dari gambar training 2
        case 3
            [imtrain3,desctrain3,loctrain3] = sift_data('training3.jpg');%mengambil
            keypoints dari gambar training 3
            [imtrain4,desctrain4,loctrain4] = sift_data('training4.jpg');%mengambil
            keypoints dari gambar training 4
        otherwise
            display('Tidak ada Training Image yang dimaksud');
    end
end
save training.mat; %penyimpanan keypoints ke dalam database
```

- `matching.m`

Program ini berperan untuk mengambil *keypoints* dari gambar *testing*, kemudian mencocokkan dengan basis data sehingga diperoleh jumlah matches, waktu proses, dan argument akhir apakah gambar dikenali sesuai dengan jenisnya.

```
function skripsimatch(image1,T)
tic;
load('training.mat');
x = 4 ; %jumlah image training
num2 = 0
y = 0
```

*% menemukan Keypoints untuk gambar testing*

```
[im1, des1, loc1] = sift_data(image1);
```

```

% distRatio: Nilai threshold perbandingan closest to second closest
% neighbor
distRatio = T;

for counter=1:x
    switch counter
        case 1
            des2=desctrain1;
        case 2
            des2=desctrain2;
        case 3
            des2=desctrain3;
        case 4
            des2=desctrain4;
        otherwise
            display('Tidak ditemukan Gambar Training');
    end

    des2t = des2'; % Transpose Matriks
    for i = 1 : size(des1,1)
        dotprods = des1(i,:) * des2t; % Perkalian dot
        [vals,indx] = sort(acos(dotprods)); % Penghitungan euclidean distance

        % Pengecekan apakah nilai perbandingan antara keypoints berada di bawah
        % threshold
        if (vals(1) < distRatio * vals(2))
            match(i) = indx(1);
        else
            match(i) = 0;
        end
    end
    num = sum(match > 0);
    if num>num2
        num2=num;
        y=counter;
        for i = 1:size(des1,1)
            match2(i)=match(i);
        end
    end
end
end
if num2 > 2
    switch y
        case 1
            im2=imtrain1;loc2=loctrain1;
        case 2
            im2=imtrain2;loc2=loctrain2;

```

```

    case 3
        im2=imtrain3;loc2=loctrain3;
    case 4
        im2=imtrain4;loc2=loctrain4;
    otherwise
        display('Tidak ditemukan gambar training')
    end
end

```

```

% membuat gambar training dan gambar testing dalam satu jendela
im3 = appendimages(im1,im2);

```

```

% Menarik garis dari keypoints yang match
figure('Position', [100 100 size(im3,2) size(im3,1)]);
colormap('gray');
imagesc(im3);
hold on;
cols1 = size(im1,2);
for i = 1: size(des1,1)
    if (match2(i) > 0)
        line([loc1(i,2) loc2(match2(i),2)+cols1], ...
            [loc1(i,1) loc2(match2(i),1)], 'Color', 'c');
    end
end
hold off;
fprintf('Found %d matches.\n', num2);
if num2>2
    switch y
        case 1
            display('Jenis Kapal : Fregat Republik Indonesia');
        case 2
            display('Jenis Kapal : Korvet Republik Indonesia');
        case 3
            display('Jenis Kapal : Fregat United States of America');
        otherwise
            case 3
                display('Jenis Kapal : Landing Australia');

                display('Tidak ada Gambar yang sesuai')
    end
else
    display ('Tidak ada Gambar yang sesuai')
end
fprintf('\nWaktu Proses: %.3f detik\n\n',toc);
else
display('tidak ada gambar yang sesuai');
fprintf('\nWaktu Proses: %.3f detik\n\n',toc);
end
end

```

end

- Program `appendimages.m`

Program ini berperan untuk menyajikan dua gambar yaitu gambar *training* dan *testing* dalam satu jendela.

```
% im = appendimages(image1, image2)
%
% Return a new image that appends the two images side-by-side.

function im = appendimages(image1, image2)

% Select the image with the fewest rows and fill in enough empty rows
% to make it the same height as the other image.
rows1 = size(image1,1);
rows2 = size(image2,1);

if (rows1 < rows2)
    image1(rows2,1) = 0;
else
    image2(rows1,1) = 0;
end

% Now append both images side-by-side.
im = [image1 image2];
```