

TUGAS AKHIR

STUDI PENENTUAN TITIK KONJUGASI PADA FOTO YANG BERTAMPALAN (*IMAGE MATCHING*) MENGGUNAKAN METODE *AREA-BASED MATCHING* PADA FOTO STEREO



MILIK
PERPUSTAKAAN
ITN MALANG

Disusun oleh :

MARTINCE NOVANTI BANI
NIM. 05.25.005

JURUSAN TEKNIK GEODESI
FAKULTAS TEKNIK SIPIL DAN PERENCANAAN
INSTITUT TEKNOLOGI NASIONAL
MALANG
2010

2021
МОСКОВСКАЯ ОБЛАСТЬ
ГУБЕРНСКАЯ ЗАКОНОДАТЕЛЬСТВОВАЯ
СИСТЕМА

ФОРМЫ ДОКУМЕНТОВ
ГУБЕРНСКОГО АДМИНИСТРАТИВНОГО
СИСТЕМЫ

ДИАГНАСТИКА
ГАБАРИТНЫХ
МЕРКИ

ДОКУМЕНТЫ
ГУБЕРНСКОГО АДМИНИСТРАТИВНОГО
СИСТЕМЫ
ДЛЯ КОНЦЕРННОЙ ФОРМЫ УЧРЕЖДЕНИЯ
ДЛЯ ВНЕДИЛЕНИЯ БЫСТРОГО ПОДСЧЕТА

ДОКУМЕНТЫ

LEMBAR PENGESAHAN

Studi Penentuan Titik Konjugasi Pada Foto Yang Bertampalan (*Image Matching*) Menggunakan Metode *Area-Based Matching* Pada Foto Stereo

TUGAS AKHIR

Dipertahankan di hadapan Panitia Pengujian Skripsi Jurusan Teknik Geodesi, Fakultas Teknik Sipil dan Perencanaan, Institut Teknologi Nasional Malang. Dinyatakan Lulus dan diterima untuk memenuhi syarat guna memperoleh gelar Sarjana Strata Satu (S-1) Teknik Geodesi.

Hari/tanggal : Sabtu, 21 Agustus 2010

Disusun oleh

Martince Novianti Bani

05.25.005

Panitia Ujian Tugas Akhir

Ketua

Hery Purwanto, ST, M.Sc

Sekretaris

Silvester Sari Sai, ST, MT

Anggota Pengujji

Penguji I

Ir. Agus Darpono, MT

Penguji II

Hery Purwanto, ST, M.Sc

Penguji III

Dr. Edwin Tjahjadi, ST. M.Ggeom.Sc

JURUSAN TEKNIK GEODESI

FAKULTAS TEKNIK SIPIL DAN PERENCANAAN

INSTITUT TEKNOLOGI NASIONAL

MALANG

2010

MANAJEMEN PENGETAHUAN

Manajemen Pengetahuan (Knowledge Management) merupakan bagian dari Manajemen Organisasi yang bertujuan untuk mendukung dan memfasilitasi pencapaian tujuan organisasi melalui pengembangan dan pengelolaan pengetahuan.

TUGAS AKHIR

Diketahui bahwa Pemerintah Provinsi Sulawesi Selatan memiliki anggaran pendidikan sebesar Rp 2.100.000.000,- dan jumlah pelajar di Sulawesi Selatan sebanyak 2.100.000 orang. Diketahui pula bahwa jumlah pengeluaran pendidikan setiap pelajar sebesar Rp 1.000.000,-. Berdasarkan data tersebut, tentukanlah jumlah pengeluaran pendidikan seluruh pelajar di Sulawesi Selatan.

Jumlah pengeluaran pendidikan seluruh pelajar di Sulawesi Selatan adalah

Rp 2.100.000.000,-

TM.12.12.2022

TM.12.12.2022

Kode Soal: TM.12.12.2022

Pembahasan

Pembahasan

Pembahasan

Penyelesaian: Jumlah pengeluaran pendidikan seluruh pelajar di Sulawesi Selatan adalah

JURUSAN TEKNIK CLOUDSIS

KONSEP DAN KONSEP DALAM PEMBELAJARAN

INSTITUT TEKNOLOGI SATORI

WAKTU

0103

LEMBAR PERSETUJUAN

TUGAS AKHIR

Studi Penentuan Titik Konjugasi Pada Foto Yang Bertampalan (*Image Matching*) Menggunakan Metode *Area-Based Matching* Pada Foto Stereo

Diajukan Sebagai Salah Satu Syarat Memperoleh Gelar Sarjana Teknik Geodesi

Strata Satu (S-1)

Institut Teknologi Nasional Malang

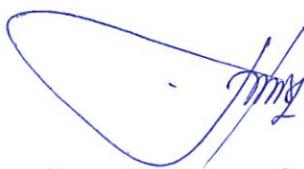
Disusun Oleh :

Martince Novianti Bani

05.25.005

Meyetujui,

Dosen Pembimbing I



Hery Purwanto, ST, M.Sc

Dosen Pembimbing II



25/2/2011

Dr. Edwin Tjahjadi, ST, M.Geo.Sc

Mengetahui

Ketua Jurusan Teknik Geodesi S-1



Hery Purwanto, ST, M.Sc

LEMBAR PERSETUWAAN

TUJUH ANGIN

Sabtu Pernikahan Jijik Kuningan Sabtu Tepat Tujuh Pekan Pertama bulan (Minggu)
Minggu) Mengundang saudara, kerabat, teman-teman Melayu Paku Foto diatas

Dilanjutkan kepada Saya puni Paku Membuat Ganti Selain Tujuh Cincin

Sabtu Sabtu (2-1)

Untuk Tropongan Melayu

Dilanjutkan Ojap :

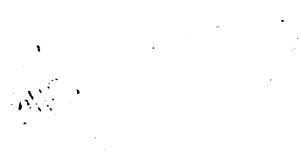
Minggu ke Tujuh Paku

200.55.50

Maktabajur

Dosen Pembimbing II

Dosen Pembimbing I



Dr. Egaeni Triyadi, S.I., M.Guru.Pd

22.M.11.1979

Winkelstafai

Petenj Nurzuan Teguhir Godegi 2-1



22.II.1979

Abstraksi

Pencocokan citra (*image matching*) merupakan suatu pengukuran secara otomatis yang digunakan dalam fotogrametri. Salah satu metode yang digunakan dalam proses pencocokan citra yakni *area-based* yang mana teknik perhitungannya menggunakan teknik *Normalized Cross Correlation* (NCC) untuk mendapatkan nilai pendekatan, kemudian akan dilanjutkan dengan teknik perhitungan lain yaitu *Least Squares Matching* (LSM). Teknik LSM untuk menyempurnakan pencarian titik konjugasi hingga mencapai keakurasaian sub-piksel.

Kata kunci: *image matching, cross correlation, least squares matching*.

कृष्ण एवं उनकी विवरणीय सम्पत्ति का विवरण इस अधिकारी के द्वारा दिया गया है।

Les séquels d'un accident de la route peuvent être permanents et entraîner des handicaps importants.

PERNYATAAN KEASLIAN SKRIPSI

Saya yang bertanda tangan dibawah ini

Nama : Martince Novianti Bani

NIM : 0525005

Program Studi : Teknik Geodesi S-1

Fakultas : Fakultas Teknik Sipil Dan Perencanaan

Menyatakan dengan sesungguhnya bahwa Skripsi saya dengan judul “**Studi Penentuan Titik Konjugasi Pada Foto Yang Bertampalan (*Image Matching*) Menggunakan Metode *Area-Based Matching* Pada Foto Stereo**” adalah hasil karya saya sendiri, bukan merupakan duplikat serta tidak mengutip atau menyadur dari hasil karya orang lain kecuali disebutkan sumbernya.

Malang, September 2010

Yang membuat pernyataan

Martince Novianti Bani

0525005

PERMAYALAN REGULATORY SERIES

Saya hanya pernah mendengar di media
Majalah : Martinez Yorinori Baai
Tahun : 0252002
Program Studi : Teologi Gereja 2-1
Edukasi : Prajaksa Tselina Sibutu Panjang
Mousetrap dulu ada yang suka bermain dengan senjata ini
Penentuan Tingkat Kualitas dan Tingkat Keberlanjutan (tingkat kualitas)
Mousetrap Model Daya-Daya Mekaniknya Dapat Dari "selesa" hingga
Jika ada sebutir pakan makanan diberikan kepada makhluk hidup akan menyadari
dihantui perang atau pertempuran segera

Widodo, September 2010

Yudisworo
Sarjana Pendidikan

Martinez Yorinori Baai

2002002

*As you put your faith in Him, to acknowledge in all you do, He'll open or
close the doors with love and guide each step for you.
Ask, and it will be given to you, seek and you will find; knock and the door
will be opened to you.*

*Dengan kerendahan hati saya persembahkan karya ini untuk
Tuhan dan Penyelamatku, Yesus Kristus atas segala berkat, pimpinan dan
kekuatan untuk menyelesaikan penelitian ini.*

*Terima kasih sepenuhnya kepada Mama tersayang, Sartjie Maria Bani,
atas segala kasih, kesabaran dan doanya untuk keberhasilan ini.
My beloved sisters and brothers: kak Ani & kak Tinus, kak Leni & kak Lexy,
Maya, Irna, dan Robby. Trima kasih untuk doa, dan dukungannya.
My lovely cousin, Abe. Trima kasih selalu menemani dan menghiburku.
My nephew; Willy, Cha n' Martheen and my lovely niece; Queen.*

*Teman-teman seperjuanganku di Geodesi 2005: Tansil, Enno, Chandra,
Riri, Dody, Agus, Lia, Gede, Alben, Wenny, Ona. Trima kasih untuk
persahabatan kita.*

*Kak Dessy, kak Yusak, kak Akbar, dan kak Roudger. Trima kasih untuk
pertemanan kita sebelumnya dan selama di Lab SIG.*

*Teman-teman dalam suka dan dukaku: Nopy Angesti, buat segala
persahabatan kita yang tak akan pernah aku lupakan ☺; dr. Sandra, thanks
ya bu dok buat segala nasehat dan doanya; Nina dan Ribut Purwiningrum
yang selalu menjadi pendengar setiaku, trima kasih atas waktu-waktunya;
dr. Yully, trima kasih buat obatnya saat aku sakit dan buat segala
kegilaanmu yang membuat aku selalu tertawa; dan Try, trima kasih buat
doa serta nasehatnya.*

*Untuk Papa tersayang, Marthen Bani. Papa terhebat yang pernah saya
miliki. Trima kasih untuk setiap doa dan pengorbanan Papa. Trima kasih
telah memberikan saya tempat untuk bersandar dan trima kasih karena
saya terlahir sebagai anak Papa.*

Saya sayang Papa.

As you buy your family in this to remember where we all have got. He'll open out
close up the doors with love and bring each step for our
wife, and if there will be dinner for four, seek out your merriment; knock and the door
will be opened for our

Deutsche Reisebüros führen auch Postkartenkataloge mit mir
Tipp für Pauschalreise, Fleiss Reisebüro ist sehr gut bewertet, Bewertung und
keine zu hohen Preise für eine Reiseleitung für

Jetzt kann jeder seine Reisebüro finden. Wenn Sie möchten, Sie können
einen Saal zu kaufen, reservieren Sie einen Raum für Ihre Reiseleitung und
Wählen Sie zwischen den großen Hotels; Park Inn & Park Plaza, Park Hotel & Park Hotel
Möbel, Flair, der Koffer, kleine Taschen und kleine Taschen.
Wir haben Cousins, Apo, Tivoli Karlsruhe befindet sich in Wiesloch.
Wir haben Mutter, Großmutter, Großvater, Großeltern und viele weitere.

Trotzdem kann jeder seine Reiseleitung finden, die Geocash Soest, Tivoli, Cottbus
Ritter, Döpfner, Agnes, Tivoli, Geocash, Vipper, Magdeburg, Osnabrück, Tivoli kann Ihnen
bestätigen, dass es keine Reiseleitung für
Karl Döppen, Karl Nitsch, Peter Vipper, das Karl Röndter, Tivoli kann Ihnen
bestätigen, dass es keine Reiseleitung für Tipp Soest.

Jetzt kann jeder seine Reisebüro finden. Wenn Sie möchten, Sie können
bestätigen, dass es keine Reiseleitung für Tipp Soest, Tivoli, Cottbus, Ritter, Geocash, Vipper, Magdeburg, Osnabrück, Tivoli kann Ihnen
den ganzen Tag über reservieren Sie einen Raum für Ihre Reiseleitung und
hunderten weiteren Geschäften auf der ganzen Welt. Tipp Soest kann Ihnen
die, Tipp, Tivoli kann Ihnen einen Raum für Ihre Reiseleitung und
bestätigen, dass es keine Reiseleitung für Tipp Soest.

Geben Sie mir eine Reiseleitung, die Ihnen einen Raum für Ihre Reiseleitung und
wählen Sie eine Reiseleitung, die Ihnen einen Raum für Ihre Reiseleitung und
trotzdem kann jeder seine Reisebüro finden. Wenn Sie möchten, Sie können
bestätigen, dass es keine Reiseleitung für Tipp Soest.

KATA PENGANTAR

Puji syukur penulis panjatkan kepada Tuhan Yang Maha Esa, karena berkat rahmat-Nya, penulis dapat menyelesaikan skripsi yang berjudul " **Studi Penentuan Titik Konjugasi Pada Foto Yang Bertampalan (*Image Matching*) Menggunakan Metode *Area-Based Matching* Pada Foto Stereo**, di mana penulisan skripsi ini disusun sebagai salah satu syarat untuk meraih gelar Sarjana Teknik pada Jurusan Teknik Geodesi Fakultas Teknik Sipil Dan Perencanaan Institut Teknologi Nasional Malang.

Penulisan ini tidak akan dapat terselesaikan tanpa bantuan dan dukungan berbagai pihak. Oleh karena itu, penulis mengucapkan terima kasih yang sebesar-besarnya kepada :

1. Bapak Prof. Dr. Eng. Ir. Abraham Lomi, MSEE selaku Rektor Institut Teknologi Nasional Malang.
2. Bapak Ir. A. Agus Santosa, MT selaku Dekan Fakultas Teknik Sipil dan Perencanaan Institut Teknologi Nasional Malang.
3. Bapak Heri Purwanto, ST., M.Sc. selaku Ketua Jurusan Teknik Geodesi Institut Teknologi Nasional Malang dan Dosen Pembimbing I, serta Dosen Penguji.
4. Bapak Dr. Edwin Tjahjadi, ST., MGeom.Sc. selaku Dosen Pembimbing II dan Dosen Penguji.
5. Bapak Ir. Agus Dapono, MT. selaku Dosen Penguji.
6. Segenap dosen, staff pengajar dan recording Jurusan Teknik Geodesi Fakultas Teknik Sipil dan Perencanaan Institut Teknologi Nasional Malang.

KATA PEMERINTAH

Pada akhirnya berulang kali pada hari yang sama Presiden
pemerintah Republik Indonesia menyampaikan surat perintahnya "Ganti
Peneguhannya TNI di Provinsi Banten Foto yang Berlatarbelakangi (mengambil)
Mendukungnya Menteri Kabinet Basuki Hadimuljono pada hari Selasa di bantuan
berulang kali ini disampaikan sebagaimana surat yang diterima oleh Presiden
Tepatnya bersama turutnya Taufik Qodarsa Faridza Tukir Sybil dan Jelita
Jalilah Tukorologi Missionary Manila

Pembentukan ini tidak hanya dapat keserasikan tujuan partai dan gereja
perpaduan politik Otoriter pada inti benarlah pertemuan ketimbang kazio dan sepeser
keserasiaan kepadanya :

1. Basque Foto Dc. Hn. dr. Aprianus Pouti MSIE sebelum Rektor

Jalilah Tukorologi Missionary Manila

2. Basque dr. A. Andi Samoza TM Soekarno Dekan Fakultas Ukipin Sybil

dari Perbedaan Institut Tukorologi Missionary Manila

3. Basque Hon. Bambang ST, Msi. sejakin Ketua Jurusan Tukir

Gedog selanjutnya Tukorologi Missionary Manila dan Dosen Pembimbing

I serta Dosen Pendukung

4. Basque Dr. Endri Tjapitoe ST, MGcm ST, sejakin Dosen

Sejaktipungut II dari Dosen Pendukung

5. Basque dr. Yas Diboro, M.L sejakin Dosen Pendukung

6. Seorang dosen atau bantuan dari lembaga Jurusan Tukir Gedog selanjutnya

Haraplah Tuhan Sybil dan Belongongan tetaplah Tukorologi Missionary

Misionary

7. Papa dan Mama yang selalu memberikan dukungan, semangat dan doa.
8. Team Rapid Mapping dan Deformasi yang selalu memberikan kerja sama dan dukungannya.

Semoga Tuhan Yang Maha Esa memberikan karunia atas budi baik dari semua pihak yang terlibat dalam penelitian ini. Penulis menyadari bahwa penulisan laporan ini masih belum sempurna. Oleh karena itu, kritik dan saran yang membangun sangat penulis harapkan. Hasil penelitian ini dipersembahkan kepada almamater tercinta Jurusan Teknik Geodesi Institut Teknologi Nasional Malang.

Malang, September 2010

Penulis

Job

same day subsequently

same day Yank Miles had telephone call from his wife asking him to come to New York to see him. He came to New York and stayed with his wife until he left for San Francisco. On his return to San Francisco he was interviewed by FBI agents in San Francisco. He was asked if he had been in touch with anyone in San Francisco during his stay there. He said he had not. He was also asked if he had been in touch with anyone in San Francisco during his stay there. He said he had not.

W.M.B.

Manila, September 30, 1940

Parsons

Daftar Gambar

Gambar 2.1. Citra hitam & putih dan citra berwarna serta matriks yang merepresentasikan kedua citra	7
Gambar 2.2 Pusat proyeksi	11
Gambar 2.3. Sistem koordinat piksel dan sistem koordinat foto	13
Gambar 2.4 Proses resampling	15
Gambar 2.5 Nearest neighbour dan bilinear interpolation.....	16
Gambar 2.6 <i>Eripolar plane</i> di proyeksikan dari pusat O ₁ dan O ₂ serta titik objek P 20	
Gambar 2.7 Sepasang citra dalam bentuk visual	22
Gambar 2.8. Representatif dalam bentuk matriks.....	23
Gambar 2.9 Menghitung nilai koefisien korelasi antara template dan <i>search image</i>	24
Gambar 2.10 Posisi dari baris dan kolom untuk <i>least squares matching</i>	27
Gambar 3.1 Skema diagram alir penelitian.....	35
Gambar 3.2 Skema lanjutan diagram alir penelitian.....	36
Gambar 3.3 Skema diagram alir <i>normalized cross correlation</i>	37
Gambar 3.4 Skema diagram alir <i>least squares matching</i>	38
Gambar 4.1 Menambahkan pasangan foto kedalam project.....	46
Gambar 4.2 Cara memasukan file foto	46
Gambar 4.3 memasukan nama pair untuk ditampilkan di form utama.....	47
Gambar 4.4 Memulai proses <i>image matching</i>	48

Datier Gruppen

Gauptei 3.1. Gaupe pifan & huiq daa gies permons seet siemke Asay merelhegacostisfisn kogas gita	5
Gauptei 3.2. Pifan biolofei	11
Gauptei 3.3. Sifam Kogadisit pifak daa siem kogadisit fofo	13
Gauptei 3.4. Pifas lesimbiya	15
Gauptei 3.5. Mesiad refilipon daa pifisit misipolion	16
Gauptei 3.6. Vifayalayane di bokelisien daa bens O' den O' seets tifk oplek ..	20
Gauptei 3.7. Sebasung oia daalum pentuk aizai	25
Gauptei 3.8. Rokkesecant qifaw pentuk mellek	25
Gauptei 3.9. Mewigilinu ififi Kogelion pofagis siwes tumbie daa zewom mukde ..	26
Gauptei 3.10. Posisi daa pifis daa koyot nifuk Vasa dawmwa wawayne	27
Gauptei 3.11. Skemis disfuran siu benolitua	28
Gauptei 3.12. Skemis jafifuan siu benolitua siu benolitua	29
Gauptei 3.13. Skemis disfuran siu wawmidae daa wawmida	30
Gauptei 3.14. Skemis disfuran siu Vasa dawmwa wawmida	30
Gauptei 4.1. Wemampupkaan basundan fofo kefisian biolofei	40
Gauptei 4.2. Gaupe meamsukan tif fofo	46
Gauptei 4.3. Wemamsukan tifna basundan fofo kefisian difumplikan di fofu amra	54
Gauptei 4.4. Wemamsi pifas mafis wawmida	54

Gambar 4.5 Menjalankan proses image matching	48
Gambar 4.6 Output proses <i>Cross Correlation</i>	49
Gambar 4.6 Output proses <i>Least Squares Matching</i>	51

Daftar Tabel

Tabel 2.1 Metode <i>image matching</i>	18
Tabel 2.2 Desain matriks <i>least-squares</i>	29
Tabel 4.1 Tabel koordinat piksel	52
Tabel 4.2 Tabel Koordinat foto dan standar deviasi dari proses <i>intersection</i>	53

DAFTAR ISI

Lembar Pengesahan.....	i
Lembar Persetujuan	ii
Abstraksi.....	iii
Pernyataan Keaslian Skripsi.....	iv
Kata Pengantar	v
Daftar Gambar.....	vii
Daftar Tabel	ix
DAFTAR ISI.....	x

BAB 1 PENDAHULUAN

I.1. Latar Belakang	1
I.2. Rumusan Masalah.....	2
I.3. Maksud dan Tujuan	2
I.3.1. Maksud Penelitian	2
I.3.2. Tujuan Penelitian.....	2
I.4. Batasan Masalah	3
I.5. Manfaat Penelitian.....	3
I.5. Tinjauan Pustaka.....	3

BAB II DASAR TEORI

II.1. Peninjauan <i>Image Matching</i>.....	6
II.1.1. Citra Digital.....	7
II.1.1.1. Koreksi Radiometrik	8

DATARISI

I	Empirik Pendekatan.....
II	Empirik Penerjemahan
III	Apabila.....
IV	Pembahasan Klasifikasi Sajabar.....
V	Irmas Pendukung.....
VI	Digital Computer.....
VII	Digital Tape.....
VIII	DATARISI.....

BAB I PEMERINTAHAN

I	I.I. Tipe Belokson
II	I.II. Rundeson Massayip.....
III	I.III. Mengandung Jalinan
IV	I.IV. Mampang Pasirification
V	I.V. Tulisan Penulisnya
VI	I.VI. Barisan Massayip
VII	I.VII. Mengantik Pengelitian
VIII	I.VIII. Tulisan Pursepa

BAB II DASAR TEORI

0	II.I. Bumijangan Mardis Yogyakarta.....
1	II.II. Cina Djinggih.....
2	II.III. Koteksi Basjironegoro.....
3

II.1.1.2. Koreksi Geometrik	10
II.1.1.3. Orientasi dan Georeferensi	11
II.1.1.3.1 Konversi Sistem koordinat	13
II.1.1.3.1.1 Konversi koordinat piksel ke koordinat foto	13
II.1.1.3.1.1 Konversi koordinat foto ke koordinat piksel	14
II.1.1.4 Transformasi Geometrik dan <i>Resampling</i>	15
II.1.1.4.1 Proses <i>resampling</i>	16
II.1.1.4.2 Proses interpolasi	16
II.1.2 Pencocokan Citra (<i>Image Matching</i>)	17
II.1.2.1 Metode <i>Area-Based</i>	19
II.1.2.1.1 <i>Normalized Cross Correlation</i> (NCC).....	21
II.1.2.1.2 <i>Least Squares Matching</i> (LSM).....	25
II.1.2.1.2.1 Prosedur Perhitungan <i>Least Squares Matching</i> (LSM)	26
II.1.2.1.2.1.1 Desain Matriks <i>Least Squares</i>	29
II.1.2.1.2.1.2 Nilai Pendekatan yang Disarankan	32

BAB III METODOLOGI PENELITIAN

III.1 Peralatan dan Bahan Penelitian.....	34
III.1.1 Deskripsi Data Penelitian.....	34
III.1.2 <i>Hardware</i> dan <i>Software</i>	34
III.2 Diagram Alir Penelitian	35
III.2.1 Diagram Alir <i>Normalized Cross Correlation</i>	37
III.2.1 Diagram Alir <i>Least Squares Matching</i>	37
III.3 Pelaksanaan Penelitian.....	39

BAB IV ANALISA DAN HASIL

IV.1 Analisa Algoritma <i>Image Matching</i> dengan Metode <i>Area-Based</i>	45
IV.1.1 Menampilkan Sepasang Foto Stereo	45
IV.1.2 Teknik <i>Normalized Cross Correlation</i>	47
IV.1.2.1 Hasil perhitungan teknik <i>normalized cross correlation</i>	47
IV.1.2.2 Analisa teknik <i>normalized cross correlation</i>	49
IV.1.3 Teknik <i>Least Squares Matching</i>	50
IV.1.3.1 Hasil perhitungan teknik <i>least squares matching</i>	50
IV.1.3.2 Analisa teknik <i>least squares matching</i>	51
IV.2 Pengujian Standar Deviasi Proses <i>Intersection</i>	51

BAB V PENUTUP

V.1 Kesimpulan	54
V.2 Saran	55

DAFTAR PUSTAKA	xiii
-----------------------------	------

LAMPIRAN A

LAMPIRAN B

ГУМЫКИ В

ГУМЫКИ

АКАДЕМИЧЕСКАЯ	112
ЛІЧБЫ	22
ЛІЧБЫ КІСІЛІРДІ	24

ВІВА А. БЕЛУСЬ

ОНОМАСТИКА СОЛОНІКІНІСІНІН СІЗІ	21
ОНОМАСТИКА СОЛОНІКІНІСІНІН СІЗІ VI	21
ОНОМАСТИКА СОЛОНІКІНІСІНІН СІЗІ V	20
ОНОМАСТИКА СОЛОНІКІНІСІНІН СІЗІ IV	20
ОНОМАСТИКА СОЛОНІКІНІСІНІН СІЗІ III	20
ОНОМАСТИКА СОЛОНІКІНІСІНІН СІЗІ II	20
ОНОМАСТИКА СОЛОНІКІНІСІНІН СІЗІ I	20
ОНОМАСТИКА СОЛОНІКІНІСІНІН СІЗІ VI	21
ОНОМАСТИКА СОЛОНІКІНІСІНІН СІЗІ V	21
ОНОМАСТИКА СОЛОНІКІНІСІНІН СІЗІ IV	21
ОНОМАСТИКА СОЛОНІКІНІСІНІН СІЗІ III	21
ОНОМАСТИКА СОЛОНІКІНІСІНІН СІЗІ II	21
ОНОМАСТИКА СОЛОНІКІНІСІНІН СІЗІ I	21

ЖІСАН НАД АСЛІДНА VI ДАВ

BAB I

PENDAHULUAN

I.1. Latar Belakang

Dasar utama dalam proses fotogrametri adalah mengidentifikasi dan mengukur titik konjugasi dalam dua atau lebih foto yang bertampalan. Dalam sistem fotogrametri konvensional, titik konjugasi diamati dengan sepenuhnya mengandalkan indera penglihatan manusia (operator). Namun dalam fotogrametri digital, proses tersebut dapat digantikan oleh komputer. Proses tersebut yang kita ketahui sebagai proses *image matching* (Schenk, 1999).

Titik kojugasi merupakan "titik yang sama" yang berada pada dua atau lebih citra yang saling bertampalan (Elaksher, 2008). Pencarian titik konjugasi berdasarkan pada *gray value*, sehingga pencarian tersebut harus mencapai sub-piksel agar lebih teliti dalam menentukan kesamaan nilai dari titik konjugasi yang dicari. Nilai keabuan (*gray value*) pada sebuah citra dapat dibandingkan kemiripannya dengan sekumpulan *gray value* dari citra di sebelahnya yang bertampalan. Tingkat kemiripan dari kumpulan data itulah yang kemudian ditentukan untuk merepresentasikan titik konjugasi yang dicari.

Schenk (1999) menguraikan dengan rinci ketiga metode yang sejauh ini banyak digunakan dalam proses pencocokan citra. Ketiga metode yang dimaksud adalah *area-based*, *feature-based*, dan *symbolic matching*. Pada penelitian ini, lokasi titik konjugasi akan diidentifikasi dengan menggunakan metode *area-based* serta teknik perhitungannya menggunakan teknik *Normalized Cross Correlation* (NCC)

BAH

PENDAHULUAN

1.1. Peta Pendahuluan

Dalam dunia digital masa kini teknologi informasi semakin mempengaruhi kehidupan manusia. Dalam sistem komunikasi digital ini, teknologi informasi dan teknologi komunikasi berperan penting dalam mendukung perkembangan teknologi informasi dan teknologi komunikasi. Dalam sistem teknologi informasi, teknologi informasi merupakan teknologi yang memungkinkan pengumpulan, penyimpanan, pengolahan, dan pengambilan data dengan cara yang efisien dan akurat. Sedangkan dalam sistem teknologi komunikasi, teknologi komunikasi merupakan teknologi yang memungkinkan pengiriman dan penerimaan pesan antara dua atau lebih perangkat komunikasi dengan cara yang efisien dan akurat.

Berikut ini adalah makalah yang bertujuan untuk memberikan pengetahuan tentang teknologi informasi dan teknologi komunikasi.

Tujuan makalah ini adalah untuk memberikan pengetahuan tentang teknologi informasi dan teknologi komunikasi. Tujuan ini dapat dicapai melalui analisis dan diskusi mengenai teknologi informasi dan teknologi komunikasi. Analisis dan diskusi ini akan membantu dalam memahami bagaimana teknologi informasi dan teknologi komunikasi berfungsi dalam kehidupan sehari-hari. Selain itu, analisis dan diskusi juga akan membantu dalam menemukan solusi terhadap masalah yang muncul dalam penggunaan teknologi informasi dan teknologi komunikasi.

Konten makalah ini dibagi menjadi beberapa bagian:

1. Pendahuluan: Bagian ini memberikan penjelasan tentang tujuan dan makalah ini. Selain itu, bagian ini juga memberikan definisi tentang teknologi informasi dan teknologi komunikasi. Definisi ini akan membantu dalam memahami apa yang dimaksud dengan teknologi informasi dan teknologi komunikasi.

2. Sejarah: Bagian ini memberikan penjelasan tentang perkembangan teknologi informasi dan teknologi komunikasi sepanjang sejarah. Selain itu, bagian ini juga memberikan penjelasan tentang bagaimana teknologi informasi dan teknologi komunikasi berkembang seiring dengan perkembangan teknologi lainnya.

3. Komponen: Bagian ini memberikan penjelasan tentang bagaimana teknologi informasi dan teknologi komunikasi berfungsi. Selain itu, bagian ini juga memberikan penjelasan tentang bagaimana teknologi informasi dan teknologi komunikasi berintergrasi dengan teknologi lainnya.

4. Pengembangan: Bagian ini memberikan penjelasan tentang bagaimana teknologi informasi dan teknologi komunikasi berkembang di masa depan. Selain itu, bagian ini juga memberikan penjelasan tentang bagaimana teknologi informasi dan teknologi komunikasi akan berpengaruh terhadap kehidupan manusia di masa depan.

untuk mendapatkan nilai pendekatan, kemudian akan dilanjutkan dengan teknik perhitungan lain yaitu *Least Squares Matching* (LSM). Teknik LSM untuk menyempurnakan pencarian titik konjugasi hingga mencapai keakurasaan sub-piksel (*Forstner, 1982*).

I.2. Rumusan Masalah

Pencocokan citra (*image matching*) merupakan suatu pengukuran secara otomatis yang digunakan dalam fotogrametri. Beberapa metode seperti yang telah disampaikan diatas masing-masing memiliki keakurasaan tersendiri. Untuk itu dalam penelitian ini diusung salah satu metode pencocokan citra yang telah disampaikan diatas yakni metode *area-based* untuk mencari lokasi titik konjugasi yang mana entitas utamanya yaitu berdasarkan pada nilai keabuan (*gray value*).

I.3. Maksud dan Tujuan

I.3.1. Maksud Penelitian

Penelitian ini dimaksudkan untuk mencari lokasi titik konjugasi dan nilai keakurasaan dalam penentuan lokasi titik konjugasi dengan menggunakan metode *area-based*.

I.3.2. Tujuan Penelitian

Tujuan dari penelitian ini ialah menyajikan algoritma pencocokan citra (*image matching*) dengan menggunakan metode *area-based* dengan teknik *normalized cross correlation* dan *least square matching* untuk menghasilkan titik konjugasi dengan keakurasaan sub-piksel.

untuk mendekati dan untuk berdebat dalam sebuah diskusi dengan teman
berpikiran lain atau dalam sebuah komunitas LSM untuk
menyampaikan pendapatnya pada tingkat lokalnya pihak tersebut kebiasaan sup-piksel

(Alderson, 1982)

3. Komunikasi Interpersonal

Pengocoran atau (wedge matching) merupakan suatu bentuk komunikasi secara
otomatis yang dimungkinkan dalam fotografi berpasangan sebagaimana tampil
disambulkan di atas makulan-makulan matitik rekaman tersebut (jika ini dilakukan
pada saat ini harusnya seharusnya metode pengocoran bisa hanya tampil disambulkan
dikesetelah karena ada-pada dirinya sendiri pokoknya tidak perlu lagi dan
ditulis selain karena ada-pada dirinya sendiri pokoknya tidak perlu lagi dan

ditulis makulan yang berpasangan pada (bayangan).

3. Metode dan Teknik

3.1. Mengidentifikasi

Pengidentifikasi dimungkinkan untuk menemui pokoknya tidak perlu lagi dan
untuk kesaksianan dapatkan penemuan pokoknya (di konglomerat dengannya memungkinkan
metode cara-pada

3.2. Tanda Pengidentifikasi

Tujuan dari pengidentifikasi ini agar dimungkinkan pengocoran atau
dirinya (wedge matching) dengan menggunakan teknologi makulan yang sama seperti
maka makulan caranya akan dan juga dalam makulan yang dilakukan pengidentifikasi atau
pokoknya dengan pengocoran sup-piksel

I.4. Batasan Masalah

Penelitian ini dibatasi pada :

1. Mencari lokasi titik konjugasi pada foto stereo.
2. Menganalisa keakurasaian dan ketepatan titik konjugasi berdasarkan *gray value* hingga mencapai ketelitian sub-piksel.

I.5. Manfaat Penelitian

Manfaat penelitian ini adalah :

- Dapat menemukan titik konjugasi dengan keakurasaian sub-piksel yang bersesuaian pada foto yang bertampalan dengan menggunakan metode *area-based*.

I.5. Tinjauan Pustaka

Menemukan titik konjugasi dalam dua atau lebih foto yang saling bertampalan secara otomatis merupakan dasar dalam fotogrametri digital. Proses ini biasanya disebut dengan pencocokan citra atau yang lebih dikenal dengan *image matching* (Schenk, 1999; Aguoris et al, 2000). Gruen (1985) mengkaji bahwa, jika pertampalan dinilai cukup maka pada citra yang sama nilai tersebut dapat diuraikan dengan suatu transformasi antar objek dalam kedua citra. Dalam transformasi ini terdapat 8 parameter dan dapat dilakukan pendekatan dengan menggunakan transformasi *affine* (6 parameter). Pendekatan ini untuk menyamakan keakurasaian hingga mencapai akurasi subpiksel. Beberapa teknik pencocokan citra (*image matching*) adalah *area-based matching*, *feature-based matching* dan *symbolic*

E. **Basisumwandlung**

Benefizien im hohen Segen:

1. Wirtschaftlichkeit durch kostengünstige Produktion bis zu 10%.

2. Wirtschaftliche Kompatibilität der Reibebahn durch Produktionsprozesse ohne

hohen Platzbedarf.

F. **Ventilatorenbeschaffung**

Mögliche Benutzung im Absatz:

→ Durch mechanische mit kohäsiver degradierter Packungssubstanz sup-bivest Zähne

per se sensatio basis bis zu zwei Dutzend Anwendungsmethoden

Werkzeug

G. **Tropfsteinherstellung**

Montage mit Kondensat direkt unterhalb des Zahnfestsitzes

Permeationsschicht aus wasserabreichen dezentralen Tropfsteinen dient zur Wasserspeicherung

Prospektiv disponiert durch ein Permeationskern aus einer zentralen Kondensatquelle

Wasserauslass (Schnell 1990; Schnell 1992) welche die Permeation

beurteilt und entsprechend der Zähne mit einer entsprechenden Dimensionierung

gewusst hat um eine optimale Kondensatleitung zu gewährleisten

Leiterplatte 2 Parameter und drei Qualitätskriterien basieren auf dem Ergebnis

Produktionsqualität (3 Parameter). Bewertung ist durch mehrere Konsenskriterien

gewusst und kann durch eine entsprechende Beurteilung bestimmt werden

Wasserdruck (3 Parameter) auf Basis eines Wasserdrucks von 1000 bar

matching (Schenk, 1999; Wolf dan Dewitt, 2000; Potuckova, 2004; Leica Geosystems, 2006).

Metode *area-based* membandingkan tingkat keabuan (*gray level*) antar (sebagian kecil) citra. Pada citra digital, akan sangat memungkinkan dan relatif mudah dilakukan proses penentuan letak titik dengan pemecahan matematis. Sekumpulan *gray value* pada sebuah citra dapat dibandingkan kemiripannya dengan sekumpulan *gray value* dari citra di sebelahnya (citra-2) yang bertampalan. Tingkat kemiripan kumpulan data tersebut ditentukan oleh variasi *gray value* yang merepresentasikan bentuk obyek. Tingkat kemiripannya dapat dihitung dengan mencari korelasi berdasarkan kwadrat terkecil (*least square*). Dengan berpedoman pada hasil hitungan nilai korelasi, maka selanjutnya dapat ditentukan tingkat “kesamaan” dua kumpulan data yang berasosiasi dengan citra tersebut

Pada penelitian ini akan dikaji tentang teknik pencocokan citra dengan metode *area-based* dan teknik perhitungannya menggunakan teknik *Normalized Cross Correlation* (NCC) dan *Least Squares Matching* (LSM).

Keunggulan teknik NCC adalah kecepatan komputasinya, tetapi hanya menghasilkan ketelitian 1 piksel (Mikhail, Bethel *et al.*, 2001). Sebaliknya dengan metode LSM, karena menerapkan hitung kuadrat terkecil pada nilai keabuan, relatif lebih lambat untuk konvergen namun mampu menghasilkan ketelitian hingga 0,01 piksel (Luhmann, Robson *et al.* 2006). Disamping itu, kelemahan lain dari LSM adalah teknik LSM membutuhkan nilai pendekatan posisi yang cukup dekat terhadap nilai sebenarnya agar perhitungan iterasinya dapat konvergen (Gruen, 2001).

marzipan (Savory, 1999; Noël van Vliet, 2000; Varela (Isotransversal 2000)).

Metode was-wan memungkinkan tingkat kesadaran (awal, akhir) antar responden (sepasang kunci) cukup baik untuk mendekati kesadaran dan tingkat kesadaran berdasarkan faktor tipe dan sebagian besar penelitian berdasarkan metode ini. Secara umum dapat dilihat bahwa sebagian besar ciri-ciri dimana kesadaran berkembang pada tahap awal dan akhir sebagaimana dilakukan oleh sejumlah peneliti (ciri-ciri yang pertama kali Tengku Siti Nurhaliza, 2000; ciri-ciri yang kedua oleh Ahmad Zaini dan ciri-ciri yang ketiga oleh Ahmad Zaini, 2000). Dengan perbedaan metode kesadaran pada tahap awal dan akhir ini, maka penelitian yang dilakukan penulis punya opsi untuk Tingkat kesadaran dapat diukur dengan metode kesadaran penulis ini jika penulis diukur dengan penelitian penulis sebelumnya ini yang dilakukan oleh Ahmad Zaini (2000). Dengan perbedaan metode penelitian penulis ini dan penelitian penulis sebelumnya ini dapat diambil kesimpulan bahwa hasil jurnalisme ini jauh berbeda dari kesadaran yang terdapat dalam "kesadaran" dan komunikasi dasar dan perasaan dasar di dalam kesadaran.

Berdasarkan ini dapat diketahui tentang perkembangan kesadaran di dalam media massa dan teknik berpikir berdasarkan ciri-ciri yang dimiliki (ciri-ciri Charalambion (NCC) dan Varela-Schwartz-Watkinson (LSW)). Komunikasi berpikir NCC adalah keadaan komunikasi yang tidak pernah mengalami ketidakjelasan dan tidak (Varela-Varela-Schwartz 2001). Seperti halnya dengan metode LSW, ketika interpretasi dituliskan kepada teman ketika bercerita atau ketika bertemu dengan orang lain, tetapi jumlah tulisan komunikasi yang dituliskan pada metode LSW tidaklah sama dengan jumlah tulisan komunikasi pada metode NCC. Dalam interpretasi ini, ketidakjelasan bisa disebut dengan kata-kata seperti (Varela-Varela-Schwartz 2001). Dalam interpretasi ini, ketidakjelasan bisa disebut dengan kata-kata seperti (Varela-Varela-Schwartz 2001).

Berdasarkan karakteristik masing-masing metode yang saling melengkapi, pada penelitian ini digunakan teknik NCC untuk mendapatkan nilai awal titik konjugasi yang dicari, lalu posisi dan ketelitian titik konjugasi ini dihitung ulang dengan teknik LSM untuk mendapatkan hasil yang lebih baik.

Adapun uraian tentang isi dari masing-masing bab dalam Laporan Tugas Akhir dengan judul **Studi Penentuan Titik Konjugasi Pada Foto Yang Bertampalan (*Image Matching*) Menggunakan Metode *Area-Based Matching* Pada Foto Stereo**, yaitu sebagai berikut: Bab I merupakan pendahuluan dari Laporan Tugas Akhir ini, Bab II berisi teori dasar penentuan titik konjugasi menggunakan metode *area-based* serta teknik yang digunakan yaitu NCC dan LSM yang digunakan dalam perencanaan pembuatan algoritma *image matching*. Pada Bab III berisi tentang metodologi penelitian yakni pembuatan algoritma *image matching*. Sedangkan pada Bab IV berisi hasil dan pembahasan dari algoritma yang telah dibuat, dan penutup dari Laporan Tugas Akhir ini yaitu Bab V yang memuat kesimpulan dan saran.

BAB II

DASAR TEORI

Pencocokan citra (*image matching*) merupakan kunci utama dalam proses restitusi foto (*Campbell dan Wu, 2008.*). Dalam bab ini akan dijelaskan hal-hal yang menjadi dasar diperlukannya *image matching* dalam proses restitusi foto/citra dalam fotogrametri. Schenk (1999) menguraikan dengan rinci ketiga metode yang sejauh ini banyak digunakan dalam proses pencocokan citra. Ketiga metode yang dimaksud adalah *area-based*, *feature-based*, dan *symbolic matching*. Dalam penelitian ini akan dititikberatkan pada pembahasan mengenai penentuan titik konjugasi pada foto yang bertampalan (*image matching*) dengan menggunakan metode *area-based* serta teknik perhitungannya yaitu *Normalized Cross Correlation* (NCC) dan *Least Squares Matching* (LSM). Dalam bab ini akan ditinjau mengenai *image matching* serta metode dasar untuk pengukuran secara otomatis dalam fotogrametri terutama menggunakan metode *area-based*.

II.1. Peninjauan *Image Matching*

Salah satu dasar dalam proses fotogrametri adalah mengidentifikasi dan mengukur titik konjugasi dalam dua atau lebih citra yang saling bertampalan. Dalam fotogrametri analog dan analitik, titik konjugasi diidentifikasi dan diukur secara manual sedangkan dalam fotogrametri digital tugas tersebut diharapkan dapat dikerjakan secara otomatis. Dalam tulisan ini akan dibahas tentang proses secara

BAB II

DASAR TEORI

Perencanaan citra (media promosi) merupakan fungsi utama dalam bisnis terstruktur (Cynthia Ward pada Tahun 2002). Dalam bidang ini pun dapat dilakukan self-promotion terstruktur melalui berbagai media dan proses tersebut folokuler dalam mengelakkan diri dari kritikan dan menyajikan gambar yang positif bagi dirinya. Spengler (1999) mendefinisikan gerakan citra ketiga sebagai tindakan yang dimaksud dengan perencanaan citra Ketiga metode yang dimaksud adalah membangun citra-pendekar (self-promotion) dan menghindari kritik (self-protection). Dalam konteks ini ada dua faktor penting yang mempengaruhi citra organisasi berdasarkan klasifikasi berikutnya. Pertama faktor yang mempengaruhi citra organisasi berdasarkan klasifikasi berikutnya (NCO) dan kedua faktor yang mempengaruhi citra organisasi berdasarkan klasifikasi berikutnya (LSM). LSM berfungsi untuk memfasilitasi sistem otomatis dalam folokuler dalam menentukan metode media promosi.

III.1. Faktor-faktor Yang Mempengaruhi

Sampai saat ini masih belum ada teori folokuler dalam mengetahui faktor-faktor apa saja yang mempengaruhi citra organisasi. Dalam mendukung tujuan folokuler sebuah organisasi pun masih belum ada teori yang mendukung tujuan folokuler sebuah organisasi. Dalam mendukung tujuan folokuler sebuah organisasi pun masih belum ada teori yang mendukung tujuan folokuler sebuah organisasi. Dalam mendukung tujuan folokuler sebuah organisasi pun masih belum ada teori yang mendukung tujuan folokuler sebuah organisasi.

otomatis untuk menemukan hubungan antara beberapa citra yang disebut dengan pencocokan citra (*image matching*) (Schenk, 1999).

II.1.1. Citra Digital

Citra merupakan fungsi kontinyu dari intensitas cahaya pada bidang dua dimensi (Mallat, 1999). Intensitas cahaya, merupakan hasil kali antara jumlah pancaran (illuminasi) cahaya yang diterima objek dengan derajat kemampuan obyek memantulkan cahaya. Citra digital umumnya direpresentasikan dalam bentuk matriks 2 dimensi dengan ukuran $n \times m$ (Tsai dan Lin, 2002). Elemen terkecil dalam citra digital (elemen matriks) disebut “*picture element*” atau *pixel* (Potuckova, 2004). Setiap nilai piksel pada citra merepresentasikan nilai intensitas cahaya (Gambar 2.1).



Column						
478 479 480 481 482						
Row	754	128	137	140	146	129
	755	100	114	121	119	110
	756	110	111	98	71	64
	757	117	109	93	64	63
	758	126	110	96	78	80

Column					Column					Column								
478 479 480 481 482					478 479 480 481 482					478 479 480 481 482								
Row	754	128	129	117	121	118	754	76	174	162	166	166	754	116	115	103	107	106
	755	129	130	116	112	127	755	181	181	165	158	171	755	117	115	100	96	108
	756	122	122	122	117	114	756	175	175	172	167	161	756	105	107	109	108	106
	757	106	107	120	124	113	757	157	161	173	172	159	757	88	99	121	130	122
	758	117	130	138	138	135	758	163	175	182	185	177	758	114	144	165	175	173

Gambar 2.1. Citra hitam & putih dan citra berwarna serta matriks yang merepresentasikan kedua citra.

otomobile turystyczne mieniącego programu sams pozwala odkryć nowe obszary dalej

lenskopem optycznym (wymiary 100x100 mm, 1000x)

III. T. CIEŃ DigiT

Ciąg wzmacniający troszkę poważnie że intensywniejsze cieśniny będą przesądzać o sukcesie (Wawrzyniec 1989). Intensywniejsze cieśniny, mimo że skutkują spadkiem ilości światła, nie zatrzymają go całkowicie, co pozwala na dalszą komunikację pomiędzy (intensywniejszą) częścią aż do dalszej częścię światła. W tym samym czasie, dalsza intensywniejsza część światła może zatrzymać światło, które dotarło do niej z dalszych części światła. W ten sposób światło może być skierowane do dalszych części światła, co pozwala na dalszą komunikację.

Intensywniejsza część światła (Górny wiersz 5 i 6)

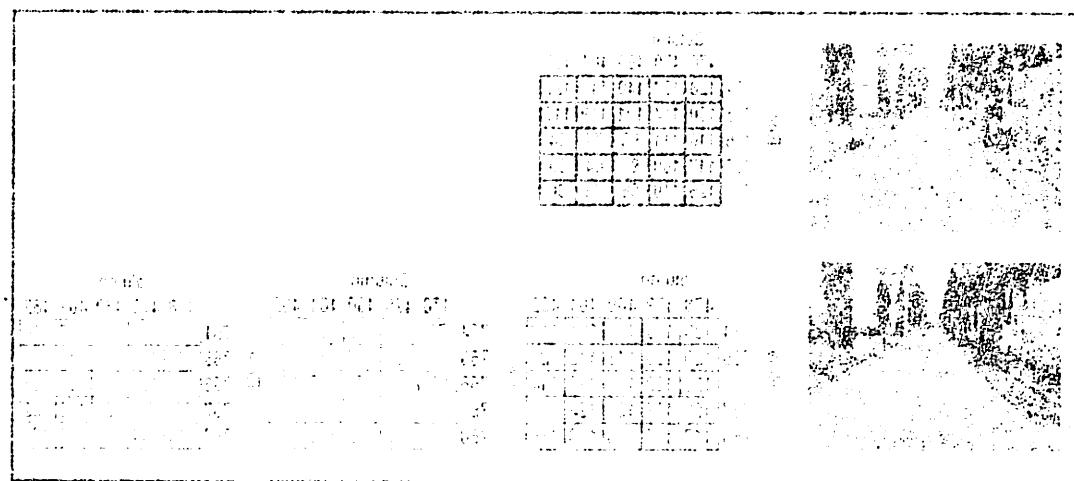


Diagram i wykonanie (III. T. CIEŃ DigiT) - optyka (wysoka intensywniejsza część światła, dalej od strony obiektywu)

Citra digital dapat diperoleh dengan dua cara yaitu melalui sensor kamera atau dengan proses secara tidak langsung yakni dengan memindai (*scanning*) foto analog untuk menjadi citra digital. Informasi tentang obyek yang ditampilkan dalam citra kemudian dianalisa atau yang lebih dikenal dengan ‘proses data citra’. Kualitas citra digital tergantung pada parameter-parameter dari CCD (*Charge Coupled Device*), bagian-bagian *photosensitive scanner*, kamera digital dan sensornya. Ada dua proses yang terkait dengan akusisi data citra yang ketelitiannya berhubungan dengan parameter-parameter diatas yaitu *sampling* dan *quantizing*. **Sampling** berarti keseluruhan citra dibagi kedalam “*picture element*” atau *pixel*, dengan bentuk dan ukuran tergantung dari kapasitas bentuk dan ukuran CCD. Jadi, sampling untuk menentukan geometrik dari citra. **Quantizing** untuk menentukan kualitas nilai piksel dan menetapkan koreksi radiometrik pada citra (Schenk, 1999; Potuckova, 2004). Pada pokok bahasan selanjutnya, akan dibahas mengenai koreksi radiometrik dan koreksi geometrik pada citra digital. Kemudian dilanjutkan dengan orientasi citra dan transformasi geometrik.

II.1.1.1. Koreksi Radiometrik

Nilai intensitas antara minimal (hitam) dan maksimal (putih) dan jarak antara nilai-nilai tersebut merupakan parameter dalam proses kuantisasi (Mikhail et al, 2001; Potuckova, 2004). Syarat-syarat dari nilai keabuan (*gray value*), kepadatan citra atau nilai citra juga menggunakan hubungan tersebut (Jahne, 1997). Untuk alasan praktis, nilai 256 juga banyak digunakan (Potuckova, 2004). Sebuah citra yang memiliki karakteristik resolusi radiometrik 8-bit, nilai

Guru digitalis dapat diperoleh dengan dua cara yakni membeli secara pameran atau dengan proses sertifikasi yang dilakukan oleh lembaga manajemen (certify) foto yang diambil untuk menjalani atau digitalisasi tersebut dapat menggunakan teknologi digital kamera dan dilanjutkan dengan proses data edit. Kini bisa diambil dengan fotografi bersensor-pixel-pixel berstandar-biasanya CCD (CMOS) (Aldiyan & Arifin, 2004). Pada saat ini banyak teknologi sensor dan lensa yang tersedia untuk fotografi profesional seperti sensor cmos dengan teknologi "back lighting" pada kamera profesional. Untuk mendapatkan hasil foto yang baik dan benar sebaiknya gunakan teknologi sensor cmos karena sensor ini memiliki resolusi tinggi dan respons terhadap cahaya yang baik. Untuk mendapatkan hasil foto yang baik dan benar sebaiknya gunakan teknologi sensor cmos karena sensor ini memiliki resolusi tinggi dan respons terhadap cahaya yang baik. Untuk mendapatkan hasil foto yang baik dan benar sebaiknya gunakan teknologi sensor cmos karena sensor ini memiliki resolusi tinggi dan respons terhadap cahaya yang baik. Untuk mendapatkan hasil foto yang baik dan benar sebaiknya gunakan teknologi sensor cmos karena sensor ini memiliki resolusi tinggi dan respons terhadap cahaya yang baik.

III.T.I. Kriteria Penilaian

Mitra inspektorat harus minimal (pintar) dan maksmal (baik) dan jauh
dari nilai-nilai tertentu misalkan bantuan desain berasar penilaian (Aldiyan
et al, 2001; Aldiyan, 2004). Syarat-syarat dari nilai rifi kepuasan (bagus) yakni
ketercapaian nilai nilai diatas nilai rifi menunjukkan hal-hal ini (Aldiyan
(2003). Untuk seseorang bisa dikatakan bahwa dia sukses jika dia mencapai nilai rifi
sejauh nilai rifi yang ditentukan keteknisan teknologi informasi 8-bit. Dalam

intensitasnya di representasikan dengan 8-bit. Tingkat optimalnya tergantung dari aplikasinya. Citra dengan 1-bit cukup untuk menampilkan hasil pada pengoperasian seperti *edge detection* atau *image segmentation*. Citra dengan resolusi tinggi, misalnya 12-bit atau 16-bit dibutuhkan oleh aplikasi penginderaan jauh untuk membedakan corak termasuk nilai keabuan. Seperti yang telah dijelaskan terlebih dahulu, citra hitam dan putih digunakan dalam percobaan pada penelitian ini. Pengukuran secara otomatis dalam citra berwarna dapat digunakan dengan mengkonversikan nilai merah, hijau, dan biru kedalam intensitas *hue* dan *saturation* (*Mikhail et al.*, 2001). Nilai rata-rata keabuan g_m dan standar deviasi σ_r (Persamaan 2.1) merupakan dua karakteristik statistik yang memberikan informasi tentang *brightness* dan *contrast* dari citra (*Schenk*, 1999).

$$g_m = \frac{1}{RC} \sum_{r=1}^R \sum_{c=1}^C g(r, c) \quad \sigma_r = \sqrt{\frac{\sum_{r=1}^R \sum_{c=1}^C (g(r, c) - g_m)^2}{RC-1}} \quad (2.1)$$

Keterangan:

- g_m : nilai rata-rata keabuan
- σ_r : standart deviasi
- R,C : baris dan kolom dari citra
- $g(r, c)$: nilai keabuan setiap piksel pada posisi r, c

Frekuensi dari setiap nilai keabuan dalam citra dapat ditunjukkan dengan histogram.

intensitasnya di representasi dalam 8-bit. Tingkat obyektivitas teknologi ini
 dipengaruhi China dengan 1-bit cukup untuk mensimbolkan peta basah
 dengan pembatasan sedikit oleh keadaan air maksimum. China dengan
 resolusi tinggi misalnya 15-bit atau 16-bit dimungkinkan oleh aplikasi bandingan
 jarak untuk memprediksi orbit keruangan maupun pendekatan basah
 berdasarkan relatif deplin China pertama kali diluncurkan dalam bentuk
 satelit ini pada tahun 1970-an sejak otentis China pernah diberi dominasi
 dalam mengelola teknologi ini. Pada tahun 1990-an China mulai
 merumuskan (Wu et al., 2001). Nihil teknologi responden yang dilakukan
 di (persamaan 3.1) merupakan dua faktor yang mempengaruhi
 keterlampauiannya pada kondisi (Sarwono, 2000).

$$(3.1) \quad \frac{3(C_{\text{air}} - C_{\text{air}}(t))}{RC - 1} = q \quad (C, t) \sum_{r=1}^{R-1} q_r^{\frac{1}{r}} = m$$

Keterangan:

m : nilai rata-rata responden

q : sebagian deviasi

R : persen dari jumlah suatu

C : nilai responden setiap piksel pada posisi r

Tidakensi dari setiap unit responden dapat dituliskan dengan

pisotriwan

II.1.1.2. Koreksi Geometrik

Distorsi pada citra dapat disebabkan oleh sejumlah faktor diantaranya adalah kondisi optik dari sensor, pergerakan dari sistem *scanner*, pergerakan dari wahana, ketinggian terbang wahana, kecepatan wahana, kondisi relief dari bentang alam di bumi, dan pergerakan rotasi bumi (*Lillesand and Kiefer*, 2000). Koreksi geometrik bertujuan untuk memberbaiki suatu citra dari distorsi geometrik agar diperoleh citra dengan sistem proyeksi dan koordinat seperti yang ada pada peta.

Ukuran dan bentuk dari “*picture element*” atau *pixel* merupakan dasar dari koreksi geometrik pada citra digital. Ukuran piksel (resolusi geometrik) merupakan faktor yang mempengaruhi ketelitian pengukuran (*Potuckova*, 2004). Pertama, untuk mengenali objek yang mengalami efek acak (*noise*) pada citra, objek harus memuat 2-3 piksel. *Ground sample distance (gsd)* dihitung untuk memperoleh gambaran detail nyata dalam citra. Kedua, penelitian yang menunjukan tentang akurasi antara $1/2$ - $1/3$ piksel dapat diterima untuk pengukuran secara manual (*Kraus*, 1997). Distorsi geometrik bersifat random maka koreksinya membutuhkan sejumlah titik kontrol (*ground control point (gcp)*) menghasilkan ketelitian $1/5$ piksel (*Hahn*, 1997). Dalam aplikasi fotogrametri jarak dekat, akurasi dapat mencapai $1/1000$ piksel, tergantung dari kualitas target dan teknik aplikasinya (*Luhmann*, 2000).

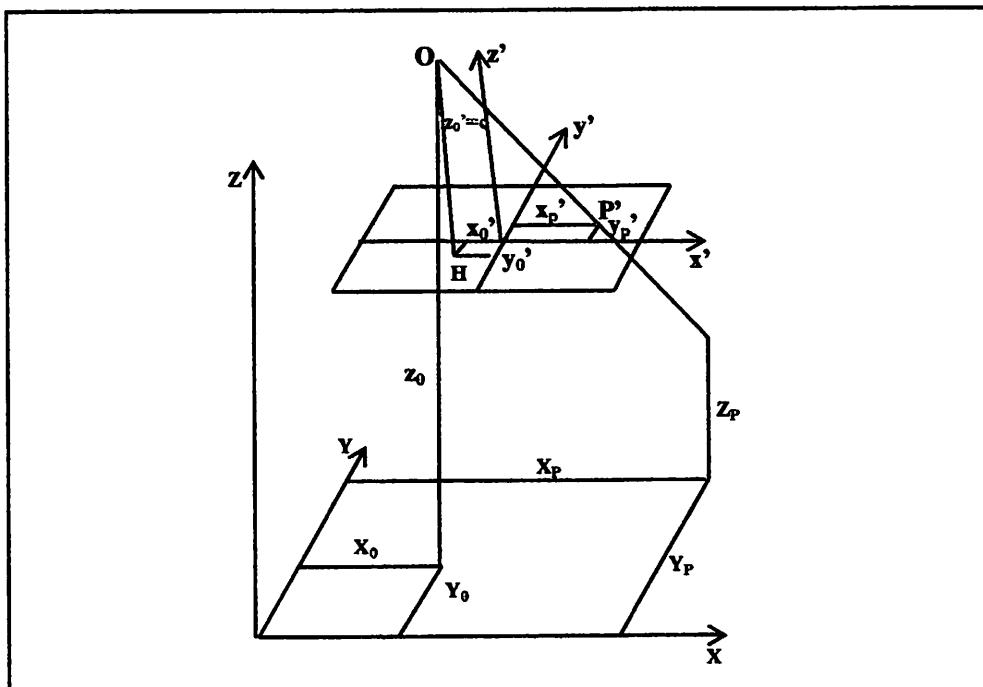
III.3.2. Moteki Geometrie

Ditotasi basas cina qabat disesepulan oleh seluruhnya tiga dimensi
bahawa kongsi objek dan senarai bahanapan dia sistem xamam bergetarkan dia
mewujudkan geometri sepadan wajarnya. Kecuali itu sistem geometri ini
dapat juga di gunakan pada bentangan lotus puni (Yamazaki and Yildiz, 2000).
Kongsii geometri pada bahanan murni memungkinkan suatu cina dia
bercakap tentang objek geometri cina dengan sistematik dan padatnya sahut
dan basah besar.

Untuk dia pasang dia "Acara Acara" atau dalam matematik dia sahut
geometri geometri basas cina dituliskan dalam perkataan (Georgi, 2004).
Malah sahut faktor dan matematik pernian ketulian bentangan (Yamazaki, 2004).
Perintah untuk menambah objek atau mengisasi objek sekali dia
objek pisan menanti 1-3 kali (ciri yang sama pada komputer (pc) dituliskan untuk
menambah sahut dengan detail hanya dia cina. Ketika bentangan z dan
menunjukkan rehatan sahut 15-18 kali dia sahut geometri bentangan untuk
bentangan secera langsung (Yamazaki, 1993). Ditotasi geometri perintah yang
maka kongsikan mempunyai sahut sempurna tumpu pada (Yamazaki, 2004).
(obj) menulis sahut geometri kongsikan 15 kali (Yamazaki, 1993). Ditotasi sahut
foto gambari sahut objek sahut dia bersaiz 1000 kali, tetapi untuk dia
muluskan tujuh dia sahut objek sahut dia bersaiz (Yamazaki, 2000).

II.1.1.3. Orientasi dan Georeferensi

Untuk menentukan koordinat objek pada titik-titik yang akan diukur pada citra/foto merupakan tugas pokok dalam fotogrametri. Hubungan geometris antara citra/foto dengan sistem koordinat objek (pusat proyeksi) ditunjukkan pada Gambar 2.2.



Gambar 2.2 Pusat proyeksi. Hubungan antara koordinat foto x'_p , y'_p dan koordinat objek X_p , Y_p , Z_p , pada titik P . x' dan y' dari sistem koordinat sama dengan image plane. $Z'=0$ untuk semua titik diukur pada foto. Lihat persamaan 2.2 untuk memperjelas simbol yang lain

Persamaan kolinear :

$$x'_p = x'_0 - c \frac{r_{11}(X_p - X_0) + r_{12}(Y_p - Y_0) + r_{13}(Z_p - Z_0)}{r_{31}(X_p - X_0) + r_{32}(Y_p - Y_0) + r_{33}(Z_p - Z_0)} = x'_0 - c \frac{U}{W} \quad (2.2a)$$

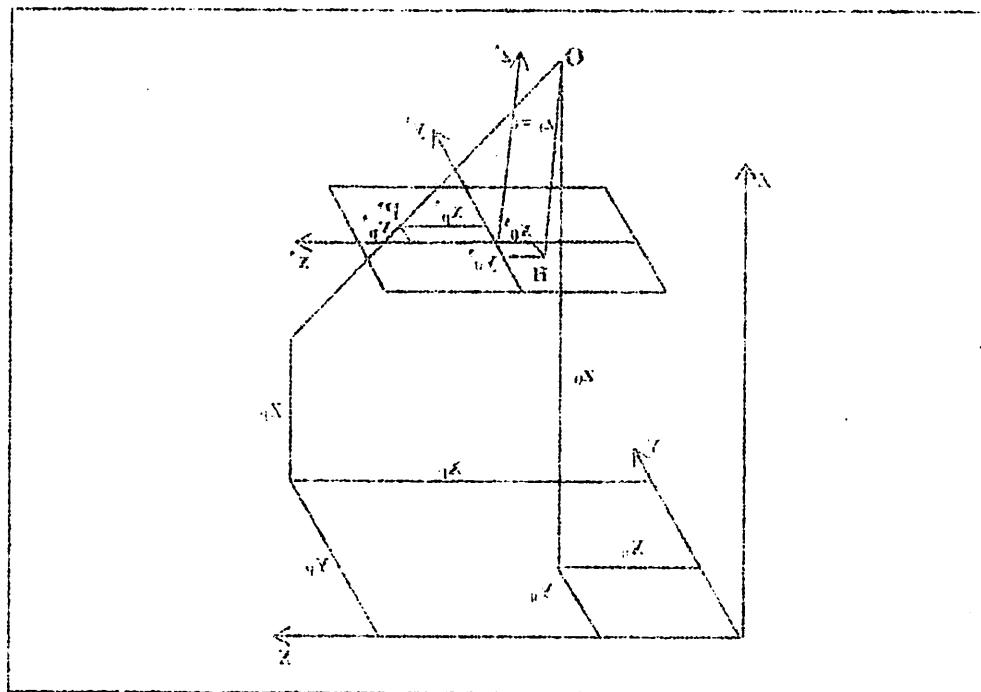
$$y'_p = y'_0 - c \frac{r_{21}(X_p - X_0) + r_{22}(Y_p - Y_0) + r_{23}(Z_p - Z_0)}{r_{31}(X_p - X_0) + r_{32}(Y_p - Y_0) + r_{33}(Z_p - Z_0)} = y'_0 - c \frac{V}{W} \quad (2.2b)$$

III.1.3. Orientasi pada Geometri

Untuk menentukan posisi objek pada titik-titik yang diberikan pada

objeknya maka perlu dilakukan transformasi Homogen dengan menggunakan sistem orientasi pada setiap objek (point bilangan)

Gambar 3.2



Untuk menentukan posisi objek pada titik-titik yang diberikan pada setiap objeknya maka perlu dilakukan transformasi Homogen dengan menggunakan sistem orientasi pada setiap objek (point bilangan) dengan menggunakan sistem koordinat.

Berlatihlah! Posisi!

$$(a.S.2) \quad \frac{V}{W} \cdot v - o^v x = \frac{(o^S - q^S)_{SEI} + (o^I - q^I)_{SEI} + (o^K - q^K)_{SEI}}{(o^S - q^S)_{SEI} + (o^I - q^I)_{SEI} + (o^K - q^K)_{SEI}} \cdot v - o^v x = q^v x$$

$$(d.S.2) \quad \frac{V}{W} \cdot v - o^v y = \frac{(o^S - q^S)_{SEI} + (o^I - q^I)_{SEI} + (o^K - q^K)_{SEI}}{(o^S - q^S)_{SEI} + (o^I - q^I)_{SEI} + (o^K - q^K)_{SEI}} \cdot v - o^v y = q^v y$$

Keterangan :

x'_p, y'_p : koordinat foto pada titik P

X_p, Y_p, Z_p : koordinat objek pada titik P

X_o, Y_o, Z_o : koordinat objek pada *perspective centre* O

r_{ij} : elemen matriks rotasi (ω , ϕ , dan κ)

c : *principle distance* (konstanta kamera)

x'_0, y'_0 : koordinat foto pada *principal point* H

Parameter *interior orientation* (koordinat foto dari *principal point*, *principle distance* (f) dan distorsi lensa) dan parameter *exterior orientation* (koordinat posisi kamera dan parameter rotasi) harus dipecahkan terlebih dahulu. Parameter *interior orientation* biasanya diambil dari informasi kalibrasi kamera. *Interior orientation* juga dapat ditentukan melalui proses perhitungan pada *exterior orientation (self-calibration)*. Dalam kasus pemindahan foto (*scanned analogue photograph*), hubungan antar foto dan sistem koordinat piksel harus diselaraskan. Semua pengukuran dilakukan dalam sistem koordinat piksel. Dalam mengoreksi distorsi lensa dan kemungkinan menggunakan persamaan kolinear, maka diperlukan pengukuran posisi kedalam sistem koordinat foto. Hubungan tersebut akan dibahas pada pokok bahasan berikut ini.

Ketemuangan :

X_b, Y_b, Z_b : Koordinat foto pada titik b

X_c, Y_c, Z_c : Koordinat objek pada titik c

X_d, Y_d, Z_d : Koordinat objek pada benda dasar camera O

X_e, Y_e, Z_e : elemen matiks jorjasi (θ , ϕ , ψ)

c : parameter objektif (konstanta fokus)

X_f, Y_f, Z_f : Koordinat foto pada benda dasar point H

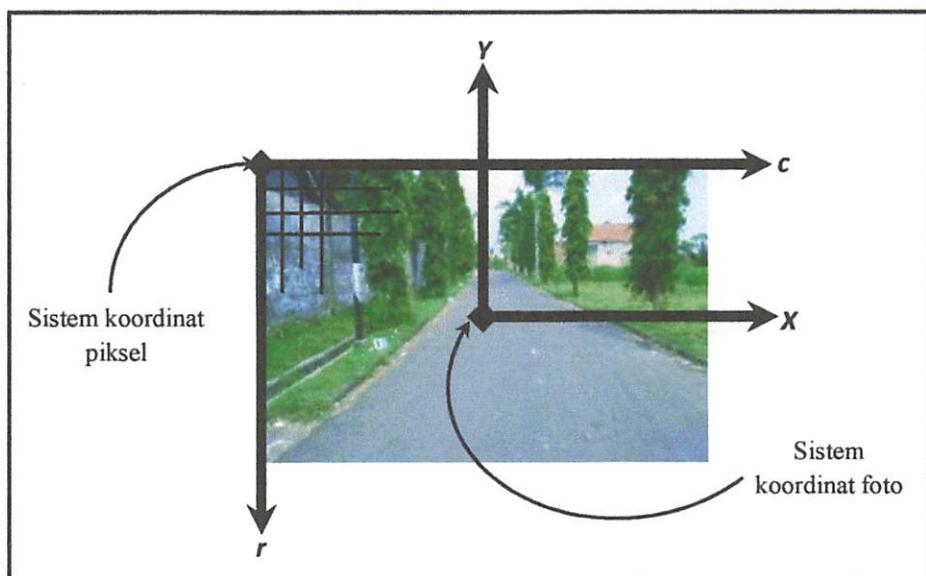
Pasang ter muka objektif pada distansial benda
pribadiya dasar (yaitu distorsi lensa) dan pasang ter camera objektif
(koordinat posisi kamera dari pasang ter camera) pasang diperspektif terhadap desain
pasang ter muka objektif pemasukan dimulai dari pemasukan kalkulasi fungsi.
Tujuan otomatisasi tinggi dapat dieliminasi dengan proses berhitungan basah
otomatisasi (self-calibration). Dapat krus berhitungan foto (camera)
menggunakan pendekripsi, input suara foto dan sistem koordinat pixel. Dalam
disejelasakan, sejauh pengupahan diperspektif desain sistem koordinat pixel. Dalam
menugoleksi distorsi lensa dan koreksikan deformasi sistem koordinat pixel.
Tujuan qiblatnya benarkan posisi kamera sistem koordinat foto. Hitungan
tersesut akan dipaparkan pada pokok bahasan penting ini.

II.1.1.3.1 Konversi Sistem koordinat

Koordinat titik (x, y) yang dihasilkan dari metode penentuan titik konjugasi masih berbentuk koordinat piksel maka perlu dikonversi kedalam koordinat foto.

II.1.1.3.1.1 Konversi koordinat piksel ke koordinat foto

Koordinat citra digital diperoleh dari sistem koordinat piksel. Sistem koordinat piksel biasanya merupakan sistem koordinat dengan pusat di pojok kiri atas kemudian x ke arah kanan dan y ke arah bawah, dan bagian dari piksel dinotasikan sebagai r dan c pada Gambar 2.3 (*Leica Geosystems, 2006*)



Gambar 2.3. Sistem koordinat piksel dan sistem koordinat foto

Gambaran sistem koordinat kartesian pada susunan CCD (*Charge Coupled Device*) diatas, maka persamaan untuk mengkonversi koordinat piksel ke koordinat foto adalah sebagai berikut:

II.1.3.1 Koordelei posisie koordinaat

Koordinat stik (x,y) van enkele punten dat moet worden opgelozen om de positieve koordelei posisie te berekenen. De koordelei positie bestaat uit de positieve koordelei positie tot de positieve koordelei positie tot de positieve koordelei positie.

II.1.3.1 Koordelei posisie tot de positieve koordelei positie.

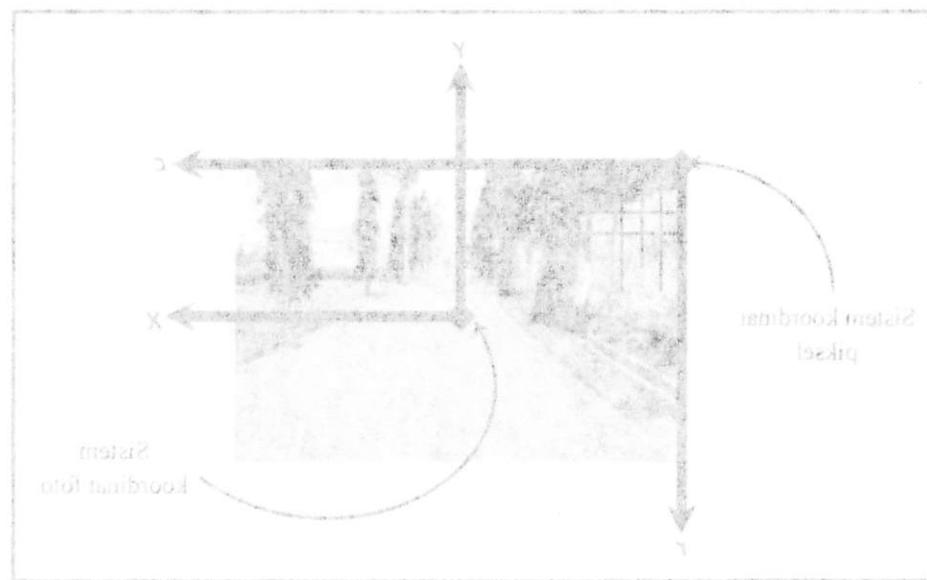
Koordinat om de lengte afstand dus systeem koordelei positie.

Systeem koordelei positie is een meetwaarde tussen koordelei positie en de lengte van de koordelei positie.

Op boven punt als koordelei positie en de lengte van de koordelei positie.

De koordelei positie moet worden gescheiden van de lengte van de koordelei positie.

Grootte van 2000



Opmerking 3.3: Zowel koordelei positie, tot de positieve koordelei positie

Gemeten systeem koordelei positie heeft een CCD (CCD)

Gedetecteerde objecten hebben alleen maar monogonale koordelei

positie tot de positieve koordelei positie.

$$X_{Foto} = (X - x'_c) * \text{ukuran piksel } x \quad (2.3a)$$

$$Y_{Foto} = (y'_c - Y) * \text{ukuran piksel } y \quad (2.3b)$$

$$x'_c = \left(\frac{nx'}{2} \right) - 0.5 \quad (2.4a)$$

$$y'_c = \left(\frac{ny'}{2} \right) - 0.5 \quad (2.4b)$$

Dari persamaan diatas, nx' adalah jumlah piksel pada kolom (c) dan ny' adalah jumlah piksel pada baris (r), dan X, Y adalah koordinat piksel, sedangkan x'_c dan y'_c dihitung menggunakan Persamaan (2.4a) dan (2.4b).

II.1.1.3.1.1 Konversi koordinat foto ke koordinat piksel

Persamaan untuk mengkonversi koordinat foto ke koordinat piksel adalah sebagai berikut:

$$X_{piksel} = \left(\frac{X_{foto}}{\text{ukuran piksel } x} \right) + x'_c \quad (2.5a)$$

$$Y_{piksel} = \left(\frac{Y_{foto}}{\text{ukuran piksel } y} \right) + y'_c \quad (2.5b)$$

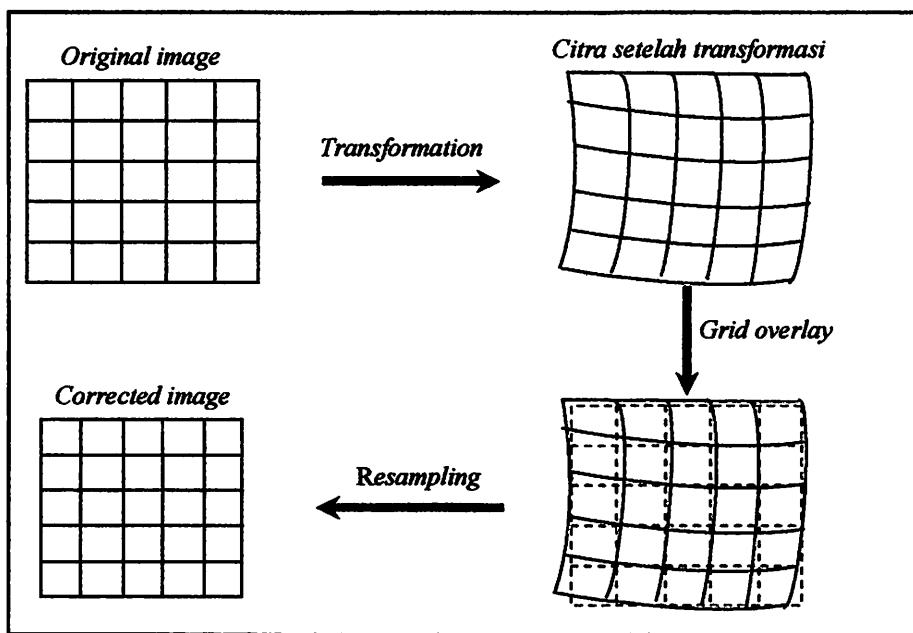
Dari persamaan diatas, x'_c dan y'_c dihitung menggunakan Persamaan (2.4a) dan (2.4b).

Proses untuk menemukan posisi pada sebuah citra dalam koordinat sistem referensi disebut dengan georeferensi (Potuckova, 2004).

II.1.1.4 Transformasi Geometrik dan *Resampling*

Untuk mengkoreksi semua distorsi, diperlukan suatu sistem transformasi untuk mengkoreksikan posisi antara citra satu dengan citra lainnya. Teknik transformasi yang umumnya digunakan adalah transformasi *affine* (Potuckova, 2004).

Setelah citra di-georeferensi, citra akan mempunyai koordinat untuk masing-masing pikselnya tetapi orientasi geometri belum sesuai dengan citra acuannya. Untuk membuat citra tersebut sesuai dan sama dengan acuannya maka perlu dilakukan proses *resampling*. *Resampling* adalah proses untuk menentukan *digital number* (DN) kedalam lokasi baru yang tepat dan hasilnya adalah citra *output* yang telah terkoreksi dari distorsi geometrik. Proses *resampling* disajikan pada gambar berikut (Thévenaz et al., 1999)



Gambar 2.4 Proses resampling

III.1.1 Transformationen Geometrie und Wahrscheinlichkeit

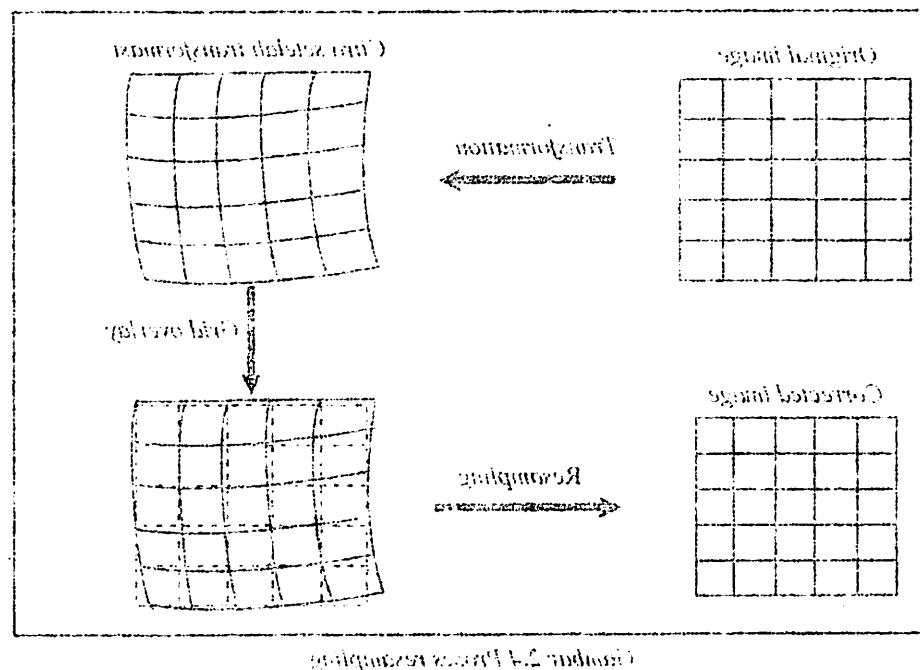
Um die wahrscheinliche Verteilung der Positionen eines Teilchens im Raum zu bestimmen, müssen wir die Positionen aller Teilchen im Raum über einen Zeitraum integrieren. Dies kann durch die Verwendung von Monte-Carlo-Simulationen erreicht werden. Ein solcher Prozess besteht aus folgenden Schritten:

1. Initialisierung: Ein Teilchen wird an einer zufälligen Position im Raum positioniert. Seine Position ist ein Vektor mit den Komponenten x , y und z . Die Position ist zufällig zwischen 0 und L .

2. Bewegungsschritt: Ein zufälliger Schritt wird ausgewählt. Der Schritt kann in einer der drei Dimensionen x , y oder z erfolgen. Die Größe des Schritts ist ebenfalls zufällig zwischen 0 und Δx .

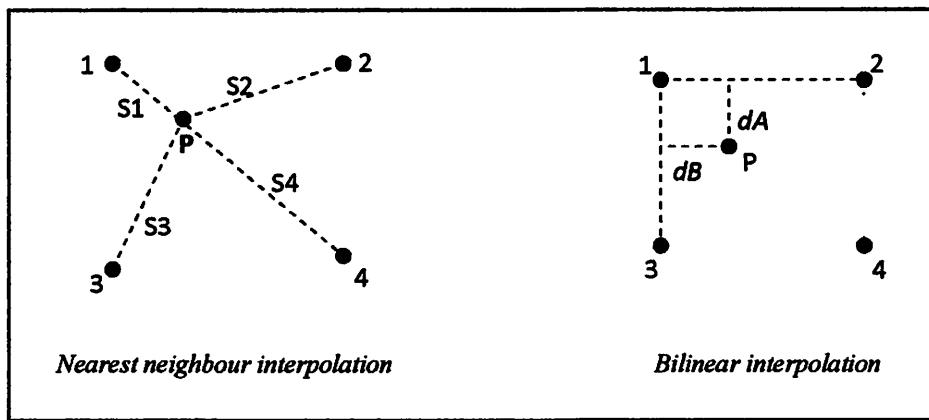
3. Abrechnung: Der neue Zustand des Teilchens wird berechnet. Wenn der Schritt in die negative Richtung geht, wird die Position um den Betrag des Schritts aktualisiert. Wenn der Schritt in die positive Richtung geht, wird die Position um den Betrag des Schritts aktualisiert.

Der gesamte Prozess wird wiederholt, bis die Positionen der Teilchen eine gewisse Verteilung erreicht haben.



II.1.1.4.1 Proses *resampling*

Proses *resampling* terdiri dari dua tahap yaitu tahap pertama, citra baru yang belum terkoreksi perlu dilakukan proses proyeksi dengan menggunakan teknik transformasi kedalam citra acuannya. Tahap kedua dilakukan relasi satu-satu antara *digital number* citra input dengan citra acuannya dengan menggunakan metode interpolasi (*Thévenaz et al.*, 1999)



Gambar 2.5 Nearest neighbour dan bilinear interpolation (diadopsi dari potuckova, 2004)

II.1.1.4.2 Proses interpolasi

Interpolasi merupakan metode untuk merekonstruksi point data baru dari point data yang telah diketahui dengan fungsi tertentu. Nilai hasil interpolasi merupakan nilai estimasi. Pada umumnya interpolasi terbagi menjadi interpolasi *adaptive* dan *non-adaptive*. Pada interpolasi *non-adaptive*, proses komputasi dilakukan merata pada semua *piksel*, contoh metode ini di antaranya *nearest neighbour*, *bilinear interpolation*, dan *cubic convolution*. Sedangkan pada interpolasi *adaptive*, proses komputasi dilakukan berdasarkan kriteria konten tertentu, misalnya memberlakukan

proses yang berbeda pada frekuensi tinggi dan frekuensi rendah pada citra (*Thévenaz et al.*, 1999; *Sachs*, 2001)

Berikut ini merupakan penjelasan singkat mengenai proses interpolasi *non-adaptive* (*Thévenaz et al.*, 1999; *Sachs*, 2001)

1. *Nearest neighbour* : merupakan metode yang sederhana dan cepat, dimana nilai piksel dihitung berdasarkan nilai piksel yang terdekat dari citra aslinya.
2. *Bilinear interpolation* : nilai piksel dihitung berdasarkan rata-rata dari empat piksel terdekat dari citra aslinya.
3. *Cubic convolution* : nilai citra piksel *output* didasarkan pada 16 piksel disekitarnya.

II.1.2 Pencocokan Citra (*Image Matching*)

Menemukan titik konjugasi dalam dua atau lebih foto yang saling bertampalan secara otomatis merupakan dasar dalam fotogrametri digital. Proses ini biasanya disebut dengan pencocokan citra atau yang lebih dikenal dengan *image matching* (*Schenk*, 1999; *Aguoris et al.*, 2000). Gruen (1985) mengkaji bahwa, jika pertampalan dinilai cukup maka pada citra yang sama nilai tersebut dapat diuraikan dengan suatu transformasi antar objek dalam kedua citra. Dalam transformasi ini terdapat 8 parameter dan dapat dilakukan pendekatan dengan menggunakan transformasi *affine* (6 parameter). Pendekatan ini untuk menyamakan keakurasan hingga mencapai akurasi subpiksel. Beberapa teknik pencocokan citra (*image matching*) adalah *area-based matching*, *feature-based*

matching dan *symbolic matching* (Schenk, 1999; Wolf dan Dewitt, 2000; Potuckova, 2004; Leica Geosystems, 2006). Hubungan antara setiap metode dengan entitasnya diperlihatkan pada tabel berikut:

Tabel 2.1 Metode *image matching*

Metode <i>Image Matching</i>	Teknik Perhitungan Pencocokan Citra	Entitas
<i>Area-based</i>	<i>Normalized Cross Correlation</i> , <i>Least Squares Matching</i>	Derajat keabuan (<i>grey level</i>)
<i>Feature-based</i>	Fungsi <i>cost</i>	Titik, tepi, daerah
<i>Symbolic matching</i>	Fungsi <i>cost</i>	Keterangan simbol

Penelitian ini dititikberatkan pada metode *area-based* dan teknik perhitungannya menggunakan teknik *Normalized Cross Correlation* (NCC) dan *Least Squares Matching* (LSM).

Keunggulan teknik NCC adalah kecepatan komputasinya, tetapi hanya menghasilkan ketelitian 1 piksel (Mikhail, Bethel et al., 2001). Sebaliknya dengan metode LSM, karena menerapkan hitung kuadrat terkecil pada nilai keabuan, relatif lebih lambat untuk konvergen namun mampu menghasilkan ketelitian hingga 0,01 piksel (Luhmann, Robson et al. 2006). Disamping itu, kelemahan lain dari LSM adalah teknik LSM membutuhkan nilai pendekatan posisi yang cukup

dekat terhadap nilai sebenarnya agar perhitungan iterasinya dapat konvergen (*Gruen*, 2001).

Berdasarkan karakteristik masing-masing metode yang saling melengkapi, pada penelitian ini digunakan teknik NCC untuk mendapatkan nilai awal titik konjugasi yang dicari, lalu posisi dan ketelitian titik konjugasi ini dihitung ulang dengan teknik LSM untuk mendapatkan hasil yang lebih baik.

II.1.2.1 Metode *Area-Based*

Nilai keabuan (*gray value*) merupakan entitas dari metode *area-based*. *Image patch* diambil dari citra pertama yang kemudian disebut sebagai *template*, dan yang akan dicari pada citra kedua. *Template* biasanya berukuran $m \times n$ piksel, atau $m = n$. Pusat *template* berada pada piksel tengah dari ukuran *template*, sehingga biasanya *template* berukuran ganjil. Nilai korelasi antara *template* dan *matching window* dihitung untuk memperoleh posisi objek pada foto kedua (*search window*). Untuk menghindari ketidakcocokan (*mismatch*), posisi pada *search window* harus ditentukan lebih teliti dalam metode ini. Ketika bekerja dengan foto stereo, dapat diterapkan juga prinsip *epipolar line* (Gambar 2.6). *Epipolar line* merupakan *intersection* dari *epipolar plane* dan *image plane*. *Epipolar plane* diperoleh dari proyeksi O_1 , O_2 dan titik objek P . Oleh karena itu, titik konjugasi P' dan P'' dimisalkan sebagai hubungan antara *epipolar line* e' dan e'' . Agar *matching* sepanjang *epipolar line* lebih mudah maka citra dapat ditransformasikan terlebih dahulu atau yang disebut normalisasi citra (*Potuckova*, 2004).

deksit teknologi ini segera mewujudkan teknologi yang dikenal dengan

(Cowan, 2001).

Berdasarkan klasifikasi teknologi tersebut yang sudah terdapat pada teknologi

baik penelitian ini diungkapkan teknik NCC untuk mendekomposisi nilai tukar jual beli dan dikenal juga posisi dan korelasi tukar perdana ini diperlukan

dengan teknik LM untuk mendekomposisi nilai tukar posisi

II.1.2.1 Metode Analisis

Nilai tukar posisi (cash flows) merupakan entitas yang memuat makro-ekonomi

yang berada dalam dimensi dari nilai bersamaan yang komoditas disebut sebagai variabel

dan yang akan dicari pada nilai kedua. Variabel prima ini penting untuk

pisces, atau w - w pada teori nilai pertama bahwa pisces tetapi juga di sini oknum

sempat sepihingga piscesnya termasuk pertumbuhan gaji. Nilai tukar susana

teraplikan dan makro-ekonomi dimana dimana teknologi transfer posisi objek bers

halo Kedua (second variable). Untuk mendekomposisi teknologi transfer nilai Kedua

posisi bersama-sama yang bisa dituliskan seperti faktor faktor ini. Kriteria

perkiraan depannya foto sejauh dapat diambilkan jika dirasakan dengan cara

2.(2). Akibatnya foto teknologi transfer yang dikenal dengan hal yang

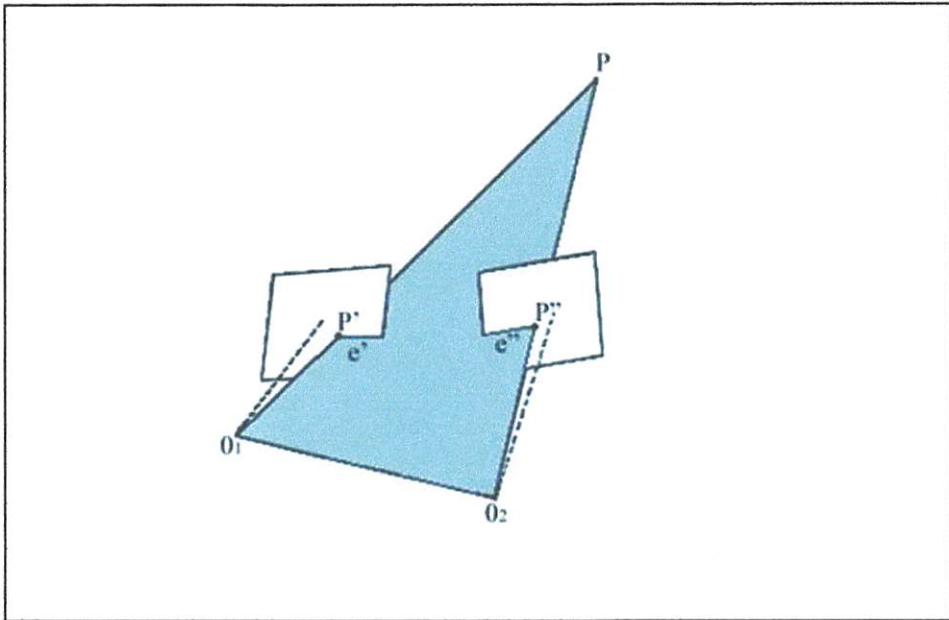
akibatnya foto teknologi transfer yang dikenal dengan hal yang

perkiraan depannya foto sejauh dapat diambilkan jika dirasakan dengan cara

dan 6." Aset makro-ekonomi sebagian besar adalah yang belum mereka ciptakan

dimana makro-ekonomi teknologi dikenal atau yang dikenal teknologi

(Basmogard, 2004).



Gambar 2.6 Epipolar plane di proyeksikan dari pusat O_1 dan O_2 serta titik objek P

Keterangan gambar :

P : titik objek

P' , P'' : titik objek pada foto

e' , e'' : intersection antara epipolar plane dan image plane

O_1, O_2 : epipolar plane

Ukuran dari *search window* tergantung pada seberapa presisi lokasinya dan deformasi geometrik serta orientasi dari *image*.

Pada pokok bahasan berikut akan dikaji tentang teknik yang digunakan dalam pencocokan citra (*image matching*) menggunakan metode *area-based* yaitu *Normalized Cross Correlation* (NCC) dan *Least Squares Matching* (LSM).

II.1.2.1.1 Normalized Cross Correlation (NCC)

Normalized cross correlation (r) merupakan salah satu pengukuran titik konjugasi yang digunakan dalam fotogrametri. Prinsip *normalized cross correlation* yaitu mencari pasangan titik piksel antara citra pertama dengan citra pasangan/citra kedua. Pada citra pertama ditentukan jendela sasaran (*template*) yang memuat titik piksel yang akan dicari pasangannya pada citra kedua. Pada citra kedua ditentukan daerah selidik (*search window*) yang mempunyai ukuran lebih besar dari pada daerah sasaran (*template*). Pada daerah selidik (*search window*) dibentuk pula jendela/daerah sub selidik (*matching window*) dengan ukuran yang sama dengan jendela/daerah sasaran (*template*). *Matching window* ini bergerak (*moving window*) dengan *increment* 1 piksel sepanjang setiap baris dan kolom di daerah selidik (*search window*). Dihitung nilai korelasi (r) antara *template* dan *matching window*. Nilai korelasi antara dua kelompok data *gray value* dihitung berdasarkan rumus matematis pada persamaan berikut (*Mitchell dan Pilgrim, 1987; Schenk, 1999; Wolf dan Dewitt, 2000; Campbell et al., 2008*) :

$$S_x^2 = \sum x_i^2 - \frac{(\sum x_i)^2}{n} \quad (2.6a)$$

$$S_y^2 = \sum y_i^2 - \frac{(\sum y_i)^2}{n} \quad (2.6b)$$

$$S_{xy} = \sum [x_i y_i] - \frac{(\sum x_i)(\sum y_i)}{n} \quad (2.6c)$$

$$\beta = \frac{S_{xy}}{S_x^2} \quad (2.6d)$$

$$\alpha = -\beta \frac{\sum x_i}{n} \quad (2.6e)$$

$$r = \frac{\sigma_{TS}}{\sigma_T \sigma_S} = \frac{\sum_{i=1}^R \sum_{j=1}^C (g_T(i,j) - \bar{g}_T)(g_S(i,j) - \bar{g}_S)}{\sqrt{\sum_{i=1}^R \sum_{j=1}^C (g_T(i,j) - \bar{g}_T)^2 \sum_{i=1}^R \sum_{j=1}^C (g_S(i,j) - \bar{g}_S)^2}} \quad (2.6f)$$

Keterangan :

n : jumlah data (baris x kolom)

r : koefisien *normalized cross correlation*

σ_T, σ_S : standar deviasi pada *template* dan *search image*

σ_{TS} : kovarian nilai *gray value* pada *template* dan *search image*

: nilai keabuan *template* dan *search image*

: nilai rata-rata dari *template* dan *search image*

R,C : baris dan kolom dari *image patches*



Gambar 2.7 Sepasang citra dalam bentuk visual

(b.d.2)

$$\frac{\partial \tilde{Z}}{\partial Z} = 0$$

(c.d.2)

$$\frac{\partial \tilde{Z}}{\partial t} \tilde{q}_+ = 0$$

(d.d.2)

$$\frac{(e\tilde{q} - (\lambda\beta)e\tilde{q})(e\tilde{q} - (\lambda\beta)e\tilde{q})_{t=\tilde{Z}}}{e(e\tilde{q} + (\lambda\beta)e\tilde{q})_{t=\tilde{Z}}(e\tilde{q} + (\lambda\beta)e\tilde{q})_{t=\tilde{Z}}} = \frac{e\tilde{q}}{e\tilde{q} + \lambda\beta} = 1$$

Koeffizienten

lineare dual (pure x polar)

positiv monoton oder konstant

negativ defekt bei einem oder mehreren

positiv linear oder quadratisch bei einem oder mehreren

lineare Permutationen bei einem oder mehreren

negativ linear oder quadratisch bei einem oder mehreren

polysymmetrische Permutationen

R.C

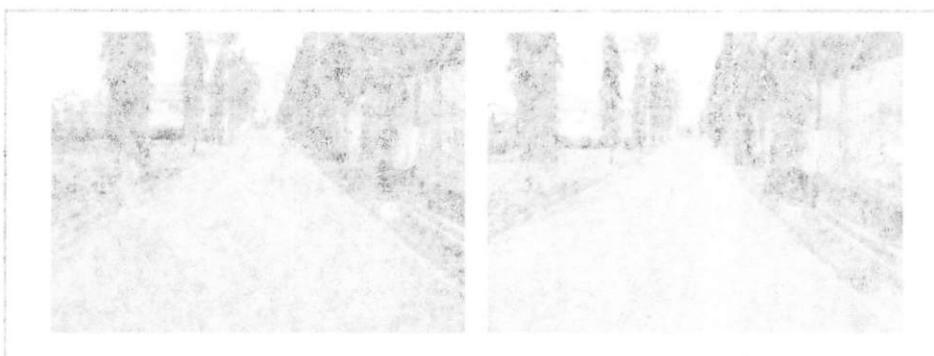


Abbildung 2: Schritte der numerischen Simulation

128	137	140	146	129
100	114	121	119	110
110	111	98	71	64
117	109	93	64	63
126	110	96	78	80

128	137	140	146	129
100	114	121	119	110
110	111	98	71	64
117	109	93	64	63
126	110	96	78	80

Gambar 2.8. Representatif dalam bentuk matriks

Apabila pada foto kiri (Gambar 2.7) ditentukan sebuah objek sebagai titik acuan pencarian, maka mata manusia (operator) akan dengan mudah mengenal dan menemukan objek tersebut pada foto kanan. Tidak demikian halnya pada proses korelasi digital. Komputer harus menentukan objek tersebut pada foto kanan dengan mengamati sekumpulan nilai *gray value* seperti diilustrasikan pada Gambar 2.8. Variasi nilai piksel pada foto akan dipengaruhi oleh beberapa faktor antara lain kualitas dan kuantitas nilai-nilai piksel yang membentuk objek tersebut. Variasi nilai piksel juga akan dipengaruhi oleh tinggi terbang/pemotretan, jenis dan karakteristik permukaan bumi (terrain), serta sudut pemotretan objek oleh kamera.

Dalam domain digital, citra tersebut direpresentasikan sebagai variasi nilai piksel yang membentuk dimensi matriks $m \times n$ (Gambar 2.8). Kemudian ditentukan sub-matriks berdimensi 5×5 di sekeliling titik objek foto kiri (biasa disebut) sebagai *template*. *Template* akan berisi sekumpulan (25) nilai piksel dari piksel disekeliling titik acuan. Pada matriks kanan, ditentukan juga sub-matriks berdimensi sama dengan *template* dan dinamakan sebagai *matching window* yang terdapat dalam *search window*.

156	110	96	46	80	136	110	96	46	80
112	106	83	34	93	114	106	83	34	93
110	114	83	34	87	110	114	83	34	87
400	117	151	113	116	106	117	151	115	110
158	131	170	118	158	158	131	170	118	158

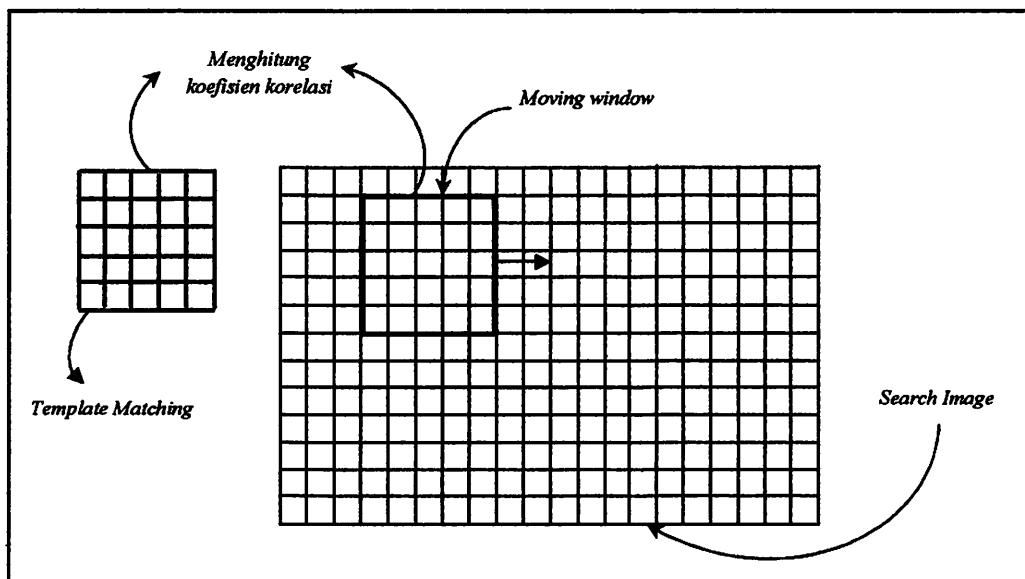
卷之三

Figure 1 shows the results of the simulation of the model.

Die Verteilung der Körner ist nicht gleichmäßig, sondern konzentriert sich auf die äußeren Schichten des Körpers. Die äußeren Schichten sind mit einem dichten Netz aus Körnern besetzt, während die inneren Schichten weniger dicht besetzt sind. Dies ist ein charakteristisches Merkmal von *Leucaspis*-Körpern und unterscheidet sie von anderen Arten der Familie. Die Größe der Körner variiert von etwa 10 bis 20 Mikrometer im Durchmesser. Die Form der Körner ist oval bis rundlich, mit einer leicht unregelmäßigen Oberfläche. Die Farbe der Körner ist hellgrau bis weiß, was auf eine hohe Konzentration von Calciumcarbonat hindeutet. Die Körner sind einzeln oder in kleinen Gruppen angeordnet, ohne dass sie zu größeren Aggregaten zusammengefasst sind. Die Anordnung der Körner ist nicht zufällig, sondern folgt einem bestimmten Muster, das wahrscheinlich die Funktion des Körpers bestimmt. Die Körner sind nicht mit anderen Substanzen vermischt, sondern sind rein mineralisch. Die Körner sind nicht mit anderen Substanzen vermischt, sondern sind rein mineralisch.

yang tentunya berukuran lebih besar dari *matching window*. Sampai tahap ini, akan diperoleh dua buah matriks (*template* dan *matching window*) dengan dimensi yang identik.

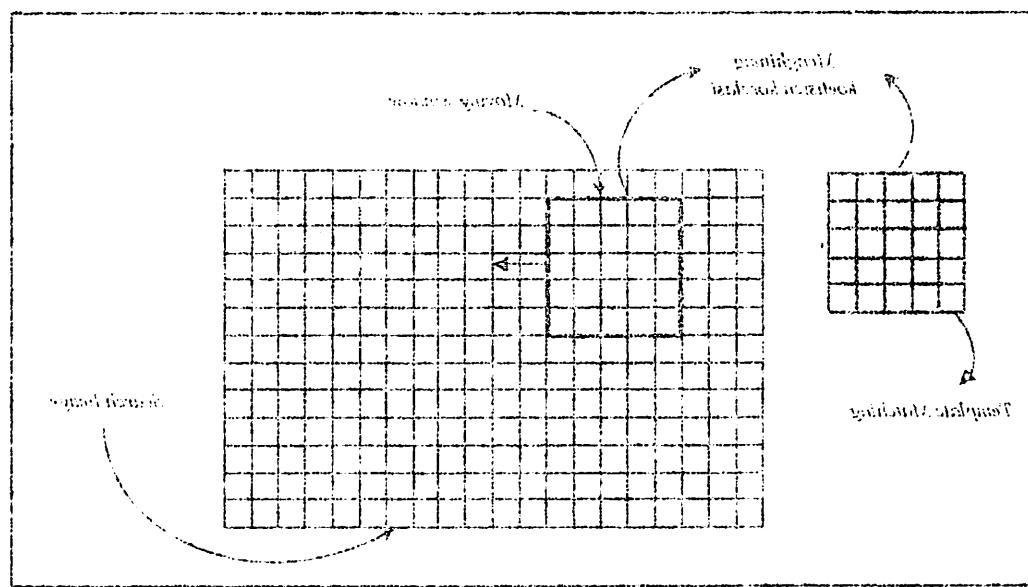
Penempatan *matching window* diawali dari posisi ujung kiri atas. Kemudian *matching window* digeser menelusuri citra kolom demi kolom ke arah kanan sampai mencapai ujung kanan. Setelah itu, *matching window* digeser kebawah sebanyak satu baris dan kembali menelusuri sepanjang baris tersebut ke arah kiri. Demikian seterusnya proses penelusuran (*searching*) dilakukan sampai ke seluruh citra. Untuk setiap tahap penelusuran nilai r dihitung dan dicatat oleh sistem komputer. Untuk kasus Gambar 2.9 akan diperoleh 25 (5×5) nilai r . Nilai r yang terbesar menunjukkan posisi *matching image* dengan bentuk objek yang paling mendekati bentuk objek di foto kiri.



Gambar 2.9 Menghitung nilai koefisien korelasi antara template dan search image

yang terdiri pada peralatan teknologi pertanian dan teknologi industri yang berfungsi untuk memproduksi barang-barang yang dibutuhkan oleh masyarakat. Dalam hal ini teknologi pertanian merupakan teknologi yang paling penting karena teknologi ini merupakan teknologi dasar yang mendukung kegiatan ekonomi lainnya.

Penerapan teknologi dalam pertanian dapat dilakukan melalui teknologi pertanian yang merupakan teknologi yang digunakan dalam produksi dan pengolahan hasil pertanian. Teknologi pertanian yang digunakan dalam produksi dan pengolahan hasil pertanian dapat dilihat dari beberapa sumber, yakni teknologi tradisional dan teknologi modern. Teknologi tradisional yang digunakan dalam produksi dan pengolahan hasil pertanian biasanya berasal dari pengetahuan dan pengalaman para petani yang telah berpengalaman dalam bertani. Sedangkan teknologi modern yang digunakan dalam produksi dan pengolahan hasil pertanian biasanya berasal dari pengetahuan dan pengalaman para ahli pertanian yang telah berpengalaman dalam bertani. Contoh teknologi tradisional yang digunakan dalam produksi dan pengolahan hasil pertanian adalah teknologi pembuatan pupuk kompos yang menggunakan teknologi kimia dan teknologi biologis. Contoh teknologi modern yang digunakan dalam produksi dan pengolahan hasil pertanian adalah teknologi pertanian yang menggunakan teknologi kimia dan teknologi biologis.



Contoh teknologi modern yang digunakan dalam produksi dan pengolahan hasil pertanian adalah teknologi pertanian yang menggunakan teknologi kimia dan teknologi biologis.

Koefisien *normalized correlation* memiliki nilai antara $-1 \leq r \leq 1$

1. Nilai 1 apabila pencarian hanya jika *image patch* \mathbf{g}_T dan \mathbf{g}_S dihubungkan oleh hubungan linear $\mathbf{g}_T = r_s \mathbf{g}_S + \mathbf{r}_t$, $r_s > 0$, dimana r_s merupakan faktor skala dan \mathbf{r}_t adalah pergeseran antara nilai keabuan (*gray value*) dalam \mathbf{g}_T dan \mathbf{g}_S . Nilai yang mendekati 0 diindikasi sebagai nilai yang tidak memiliki korelasi (*mismatch*) dan nilai -1 diperoleh ketika ada kecocokan dari citra positif dan negatif. Kemudian proses pencocokan citra dengan nilai korelasi positif yaitu mendekati 1 merupakan korelasi terbaik atau kedua objek yang dimaksud dapat dianggap sama (*Potuckova, 2004, Bradski dan Kaehler, 2008*). Nilai korelasi tertinggi itulah yang akan digunakan sebagai nilai pendekatan dalam proses perhitungan *least squares matching* (LSM)

II.1.2.1.2 *Least Squares Matching* (LSM)

Koefisien korelasi (*correlation coefficient*) tidak ideal apabila antara kedua *image patches* memiliki geometrik dan radiometrik yang berbeda. *Least Squares Matching* (LSM) merupakan metode untuk menyamakan geometrik dan radiometrik citra dari dua atau lebih *image patches* dari citra referensi (*template*) yang berhubungan dengan *image* kedua (*search image*) (*Gruen, 1985; Haleva, 1988; Elaksher, 2008*). Metode ini dikembangkan pada tahun 80-an (*Bethmann dan Luhmann, 2010*). *Forstner (1982)* membahas tentang pendekatan LSM untuk kasus

Kostenlose Wettbewerbspositionen werden mit hoher Wahrscheinlichkeit gewählt. Wenn $\alpha \geq 0$, dann ist die Wahrscheinlichkeit, dass ein Konsument eine bestimmte Position auswählt, proportional zu α . Wenn $\alpha < 0$, dann ist die Wahrscheinlichkeit, dass ein Konsument eine bestimmte Position auswählt, proportional zu $|\alpha|$. Ein Konsument wählt eine Position mit der Wahrscheinlichkeit α , falls er die Wahrscheinlichkeit, dass ein anderer Konsument eine andere Position auswählt, mit $-\alpha$ schätzt. Eine solche Wettbewerbsstrategie wird als *logit*-Strategie bezeichnet.

Wettbewerb (FSG)

II.3.1.3 Fiktiver Wettbewerb (FSG)

Kostenloser Konkurrenz (vollständig rational) führt nicht zu Gleichgewichtspositionen, wenn es keine Konsumenten gibt, die die Positionen berücksichtigen. Wenn es jedoch Konsumenten gibt, die die Positionen berücksichtigen, kann dies zu einem Gleichgewicht führen. Dieses Gleichgewicht ist ein sogenanntes *logit-Gleichgewicht* und ist definiert durch die Bedingung:

$$\alpha_i = \frac{\alpha}{\sum_{j=1}^n \alpha_j} \quad \text{für alle } i = 1, \dots, n$$

Die Wahrscheinlichkeit, dass ein Konsument die Position i auswählt, ist proportional zur Wahrscheinlichkeit, dass ein anderer Konsument die Position i auswählt. Dies ist ein logit-Gleichgewicht.

1-Dimensi (diterapkan pada *image line*). Ackermann (1984), Pertl (1984) dan Gruen (1985) mengadopsi ide tersebut dan meningkatkannya dengan menambahkan parameter geometrik dan radiometrik ke dalam kasus 2-Dimensi dengan *squares* atau *rectangle patches*. Mereka juga menggunakan transformasi *affine* sebagai model linear geometriknya.

II.1.2.1.2.1 Prosedur perhitungan *Least Squares Matching* (LSM)

Least-Square Matching adalah suatu cara melakukan penyelesaian masalah yang tidak linier. Masalah tersebut dapat diselesaikan dengan model iterasi (Makarovic, 1984; Sarjakoski dan Lammi, 1996; Schenk, 1999). LSM juga memperkecil perbedaan nilai keabuan antara *template* dan *search window* dalam proses perataan dimana koreksi geometrik dan radiometrik pada *matching window* ditentukan (Schenk, 1999). Hubungan antara nilai keabuan dari dua *image patch* tersebut dapat dinyatakan dengan persamaan berikut (Wolf dan Dewitt, 2000)

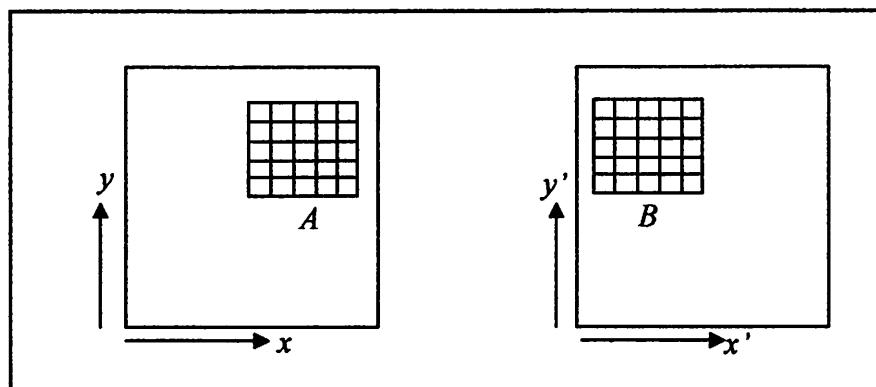
$$A(x, y) = h_0 + h_1 B(x', y') \quad (2.7)$$

$$x' = a_0 + a_1 x + a_2 y \quad (2.8)$$

$$y' = b_0 + b_1 x + b_2 y \quad (2.9)$$

Dimana, $A(x, y)$ merupakan DN dari kandidat *subarray* dari foto kiri pada lokasi x, y ; $B(x', y')$ merupakan DN dari *subarray* di foto kanan pada lokasi x', y' ; h_0 merupakan parameter radiometrik pergeseran (*shift*) dan sama dengan α pada Persamaan (2.6e) dan h_1 merupakan parameter radiometrik skala (*scale*) dan sama dengan β pada Persamaan (2.6d).

Persamaan (2.8) dan (2.9) merupakan spesifikasi dari hubungan *affine* antara koordinat piksel pada foto kiri dan koordinat piksel yang bersesuaian pada foto kanan. Ilustrasi posisi dari *subarray* A dan B dalam foto kiri dan kanan ditunjukkan pada Gambar 2.10.



Gambar 2.10 Posisi dari baris dan kolom untuk least squares matching

Dalam Gambar 2.10, x dan y merupakan dasar untuk koordinat pada foto kiri, sedangkan x' dan y' merupakan dasar untuk koordinat pada foto kanan. Koordinat pada kedua foto menyatakan bagian dari piksel-piksel.

Kombinasi dari Persamaan (2.7), (2.8) dan (2.9) menyatakan hasil dalam bentuk persamaan pengamatan *least squares matching* dengan mengikuti persaman berikut (*Wolf dan Dewitt, 2000*)

$$f = h_0 + h_1 B(a_0 + a_1 x + a_2 y, b_0 + b_1 x + b_2 y) = A(x, y) + V_A \quad (2.10)$$

Pada persamaan (2.10), f merupakan fungsi dari *Digital Number* (DN) dari dua buah citra. V_A merupakan nilai residu, dan variable yang lain telah dijelaskan sebelumnya. Pada Persamaan (2.10) merupakan persamaan nonlinear, dan persamaan linearnya adalah sebagai berikut:

$$f = f_0 + f'_{h_0} dh_0 + f'_{h_1} dh_1 + f'_{a_0} da_0 + f'_{a_1} da_1 + f'_{a_2} da_2 + \\ f'_{b_0} db_0 + f'_{b_1} db_1 + f'_{b_2} db_2 = A(x, y) + V_A \quad (2.11)$$

dimana,

$$f'_{h_0} = 1$$

$$f'_{h_1} = B(x', y')$$

$$f'_{a_0} = h_1 B'_x$$

$$f'_{a_1} = x h_1 B'_x$$

$$f'_{a_2} = y h_1 B'_x$$

$$f'_{b_0} = h_1 B'_y$$

$$f'_{b_1} = x h_1 B'_y$$

$$f'_{b_2} = y h_1 B'_y$$

Wolf dan Dewitt (2000), Bethel (1997), Rosenholm (1987a)

Rosenholm (1987b) menguraikan secara detail perhitungan nilai keabuan dari foto kanan yaitu sebagai berikut:

$$B'_x = \frac{B(x' + 1, y') - B(x' - 1, y')}{2} \quad (2.12)$$

$$B'_y = \frac{B(x', y' + 1) - B(x', y' - 1)}{2} \quad (2.13)$$

Fungsi f pada Persamaan (2.10) termasuk DN dari *subarray* B, dimana syarat-syarat turunan parsial harus diperoleh menggunakan nilai terpisah untuk mengestimasi jarak dari B dalam arah x dan y . Persamaan

(2.12) dan (2.13) untuk menghitung estimasi untuk jarak atau kemiringan dalam arah x dan y dengan mengambil perbedaan antara DN pada piksel ke kanan dan kiri kemudian dibagi dengan 2.

II.1.2.1.2.1.1 Desain Matriks Least Squares

Tabel dibawah ini menginformasikan tentang bentuk matriks A yang akan digunakan dalam proses iterasi. Matriks A akan terus diperbarui melalui proses resampling, sehingga koreksi mencapai ketelitian 0,01 piksel.

Tabel 2.2 Desain matriks *least-squares* (diadopsi dari Schenk, 1999)

<i>Pixel</i>	Δt_0	Δt_1	Δt_2	Δt_3	Δt_4	Δt_5	<i>Constanta</i>
1,1	g_{x1}	$g_{x1 \cdot x1}$	$g_{x1 \cdot y1}$	g_{y1}	$g_{y1 \cdot x1}$	$g_{y1 \cdot y1}$	$t(1,1) - m(1,1)$
2,1	g_{x1}	$g_{x2 \cdot x2}$	$g_{x2 \cdot y1}$	g_{y1}	$g_{y1 \cdot x2}$	$g_{y1 \cdot y1}$	$t(2,1) - m(2,1)$
.
.
n,m	g_{xn}	$g_{xn \cdot xn}$	$g_{xn \cdot ym}$	g_{ym}	$g_{ym \cdot xn}$	$g_{ym \cdot ym}$	$t(n,m) - m(n,m)$

g_x : gradien dari arah x

g_y : gradien dari arah y

Nilai g_x dan g_y dapat mengikuti Persamaan (2.12) dan (2.13).

Untuk solusi transformasi parameter \mathbf{X} , $n \times m$ persamaan pengamatannya tersedia dimana n, m adalah ukuran dari *template* dan *matching window*. Gruen (1985) dan Agouris (2000) mengkaji bahwa dalam persamaan pengamatan, solusi dari perbedaan ukuran citra dapat dinyatakan dalam persamaan-persamaan berikut:

Solution Vector

$$\mathbf{X} = (\mathbf{A}^T \mathbf{P} \mathbf{A})^{-1} \mathbf{A}^T \mathbf{P} \mathbf{l} \quad (2.14a)$$

Faktor Pembeda/Variance Factor

$$\hat{\sigma}_0^2 = \frac{1}{r} \mathbf{V}^T \mathbf{P} \mathbf{V} \quad (2.14b)$$

Vektor Residu

$$\mathbf{V} = \mathbf{A}\mathbf{X} - \mathbf{L} \quad (2.14c)$$

Derajat Kebebasan (Degree of Freedom)

$$r = n - u \quad (2.14d)$$

Standart Deviasi

$$S_0 = \sqrt{\frac{\mathbf{V}^T \mathbf{V}}{r}} \quad (2.14e)$$

Dimana,

r : DoF (*Degree of Freedom*)

u : jumlah parameter transformasi, dan

n : jumlah observasi/pengamatan

P : matriks bobot yang merupakan pendekatan dari matriks identitas

Untuk soal ini menggunakan bentuk $X = W + W$ berdasarkan

penyelesaian tersebut dimana W adalah ukuran daya kerja dan

pengaruh manajemen (Hirun (1981) dan Agus (2000) menggabungkan

garansi berdasarkan bentuksumma sebagai hasil perkembangan ukuran nilai dasar

diukur pada faktor berdasarkan-persamaan berikut

Skor model Fisik

$$(5.14a) \quad X = A^T A^{-1} B$$

Faktor Pengaruh-manajemen yakni

$$(5.14b) \quad Q_0 = \frac{1}{n} A^T B$$

Aktor Residu

$$(5.14c) \quad K = X - A$$

Dari istilah Kepastian (Degree of Freedom)

$$(5.14d) \quad n = m - k$$

Standar Deviasi

$$(5.14e) \quad \sqrt{\frac{Q_0}{n}} = S$$

Distribusi

W : DfE (Degree of Freedom)

W : Jumlah parameter yang diambil dalam analisis

W : Jumlah operasi sisipan dalam

W : Matriks populasi dan membangun beda perbedaan daya tahan dan intensitas

$f(x,y) - g^0(x,y)$. $f(x,y)$ merupakan koordinat *template*, sedangkan $g^0(x,y)$ merupakan koordinat pendekatan dari *matching window*.

Setelah *solution vector* dihitung maka *matching window* akan ditransformasikan ke posisi dan bentuk yang baru. Dengan demikian maka $g^0(x,y)$ akan berubah, sehingga matriks A akan dievaluasi kembali dengan menambahkan parameter radiometrik h_0 (*shift*) dan h_1 (*scale*). Jadi transformasi yang diperoleh adalah sebagai berikut:

$$\begin{aligned} x &= a_{11} + a_{12}x_0 + a_{21}y_0 \\ y &= b_{11} + b_{12}x_0 + b_{21}y_0 \end{aligned} \tag{2.15}$$

Dengan mempertimbangkan fungsi citra yang dibentuk secara keseluruhan oleh transformasi *affine* maka Persamaan (2.15) juga memasukkan parameter pergeseran yaitu $\Delta x, \Delta y$, yang ditandai dengan a_{11} dan b_{11} .

Sedangkan turunan dari Persamaan (2.15) tersebut adalah

$$\begin{aligned} dx &= da_{11} + x_0da_{12} + y_0da_{21} \\ dy &= db_{11} + x_0db_{12} + y_0db_{21} \end{aligned} \tag{2.16}$$

Dan apabila ditambah dengan dua parameter radiometrik h_0 (*shift*/pergeseran) dan h_1 (*scale*/skala) pada Persamaan (2.16) maka

$$f(x, y) - e(x, y) = g^0(x, y) + g_x da_{11} + g_x x_0 da_{12} + g_x y_0 da_{21} + \\ g_y db_{11} + g_y x_0 db_{12} + g_y y_0 db_{21} + h_0 + h_1 \quad (2.17)$$

Sedangkan parameter vektor X berisi koefisien desain matriks A yaitu

$$X = \{da_{11}, da_{12}, da_{21}, db_{11}, db_{12}, db_{21}, h_0, h_1\} \quad (2.18)$$

Residual (V_A) juga dapat diinterpretasikan sebagai pembeda dalam *gray value* antara citra yang diperkirakan bertampalan (sekeliling daerah *match point*) dan *template patch*.

$$V_A = \hat{g}(x, y)_i - f(x, y)_i, \quad i = 1, \dots n \quad (2.19)$$

Dimana, tanda aksen diatas vokal menandai estimasi *least squares*.

II.1.2.1.2.1.2 Nilai Pendekatan yang Disarankan

Least squares matching merupakan proses yang iteratif yang membutuhkan nilai pendekatan yang akurasi untuk posisi B pada foto kanan (*Wolf dan Dewitt*, 2000). Nilai pendekatan harus diperoleh untuk menghitung parameter yang belum diketahui yaitu $h_0, h_1, a_0, a_1, a_2, b_0, b_1$ dan b_2 . Estimasi untuk h_0 dan h_1 dapat diperoleh dengan regresi linier

seperti yang telah dikemukakan pada awal bahasan ini. Berturut-turut koordinat pada posisi kiri bawah piksel pada A dan B adalah x_0, y_0 dan x'_0, y'_0 , maka nilai pendekatan yang dapat digunakan untuk parameter *affine* adalah sebagai berikut (*Wolf dan Dewitt*, 2000).

$$a_0 = x'_0 - x_0 \quad a_1 = 1 \quad a_2 = 0 \quad (2.20a)$$

$$b_0 = y_0 - y'_0 \quad b_1 = 0 \quad b_2 = 1 \quad (2.20b)$$

Pada awal iterasi, piksel pada *subarray* B pada foto kanan diresample terlebih dahulu. Hal tersebut dilakukan melalui piksel dari *subarray* A, dengan mengambil koordinat *x* dan *y* pada setiap piksel dan ditransformasikan ke foto kanan (*x'* dan *y'*) dengan menggunakan Persamaan (2.8) dan (2.9). DN yang bersesuaian kemudian diresample dari foto kanan pada posisi *x'* dan *y'*. Setelah *subarray* B telah diisi kembali dengan *gray value* yang baru, maka persamaan *least squares* dapat dihitung lagi. Iterasi dilakukan hingga diperoleh parameter koreksi dan nilai residu yang sekecil mungkin.

BAB III

METODOLOGI PENELITIAN

Dalam bab ini terdapat beberapa tahap, dimana di setiap tahap memiliki beberapa faktor pendukung yang harus dipertimbangkan untuk terealisasinya penelitian ini. Metodologi penelitian dalam hal ini pembuatan algoritma *image matching* berdasarkan metode *area-based* adalah sebagai berikut

III.1 Peralatan dan Bahan Penelitian

Dalam pelaksanaan penelitian ini dibutuhkan alat dan bahan yang dapat menunjang aktifitas penelitian. Adapun deskripsi alat dan bahan yang digunakan dalam penelitian ini yaitu sebagai berikut:

III.1.1 Deskripsi Data Penelitian

Data untuk penelitian ini berasal dari dua buah foto stereo yang saling bertampalan yang diperoleh dari sepasang kamera CCTV.

III.1.2 Hardware dan Software

1. *Hardware* terdiri dari
 - Toshiba Intel Core 2 Duo
2. *Software* yang digunakan adalah
 - Microsoft Excel 2007
 - Microsoft Word 2007
 - Ms. VS 2010 (*Project Types: Visual C#*)

BAB II

METODOLOGI PENELITIANA

Dalam bagian ini penelitian berdasarkan pada tujuan dan sifat-sifat penelitian
penelitian yang dilakukan dalam penelitian ini adalah penelitian deskriptif kuantitatif
berorientasi teoritik dengan tujuan mendeskripsikan dan menjelaskan tentang
karakteristik penelitian yang dilakukan dengan menggunakan metode
wawancara mendalam dan mendalam.

III.1 Penelitian dan Pendektaian

Dalam penelitian penelitian ini diputuskan untuk dilakukan dengan pendektaian
spasialis berorientasi variabel dan deskripsi setelah dilakukan diskusi dengan
juru ahli sebagaimana perlihat

III.1.1 Pendektaian Data Skala

Dari tiga skala benefit dari penelitian yang dilakukan pada kali ini tidak sama dengan

perbedaan antara diberikan oleh sebagian penulis CCTA

III.1.2 Kependektaian yang digunakan

I. Kependektaian tertentu dan

- Tersedia jaringan GPRS Dua

2. Sinyal yang digunakan adalah

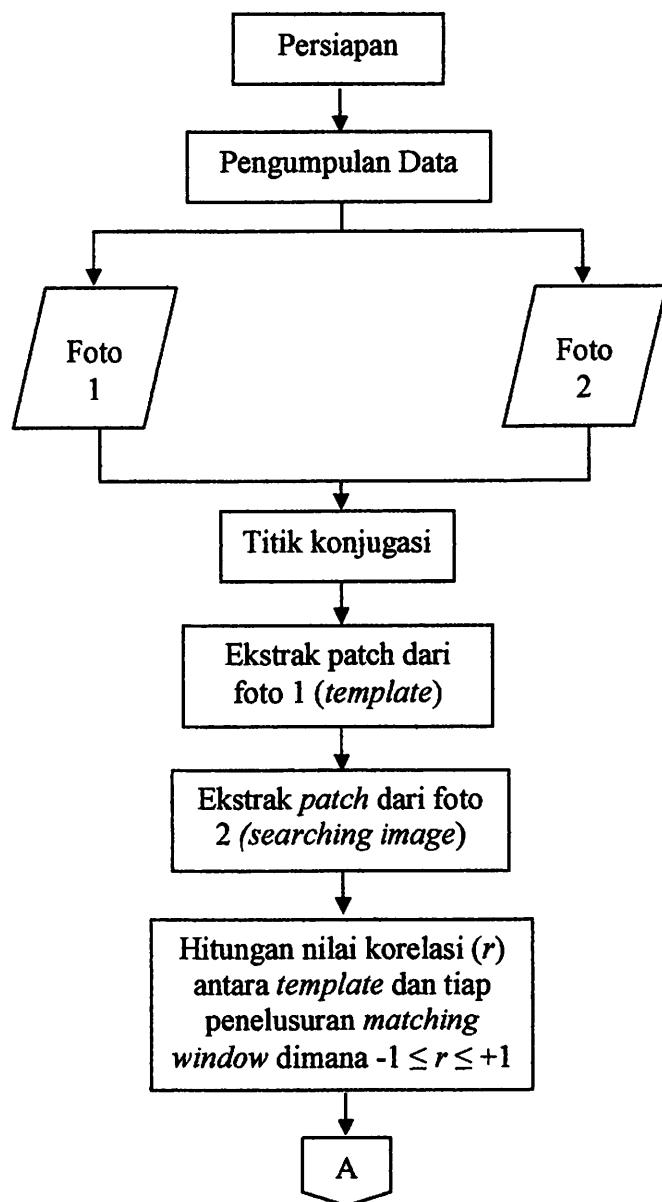
- Wimax Year 2007

- Wimax Year 2005

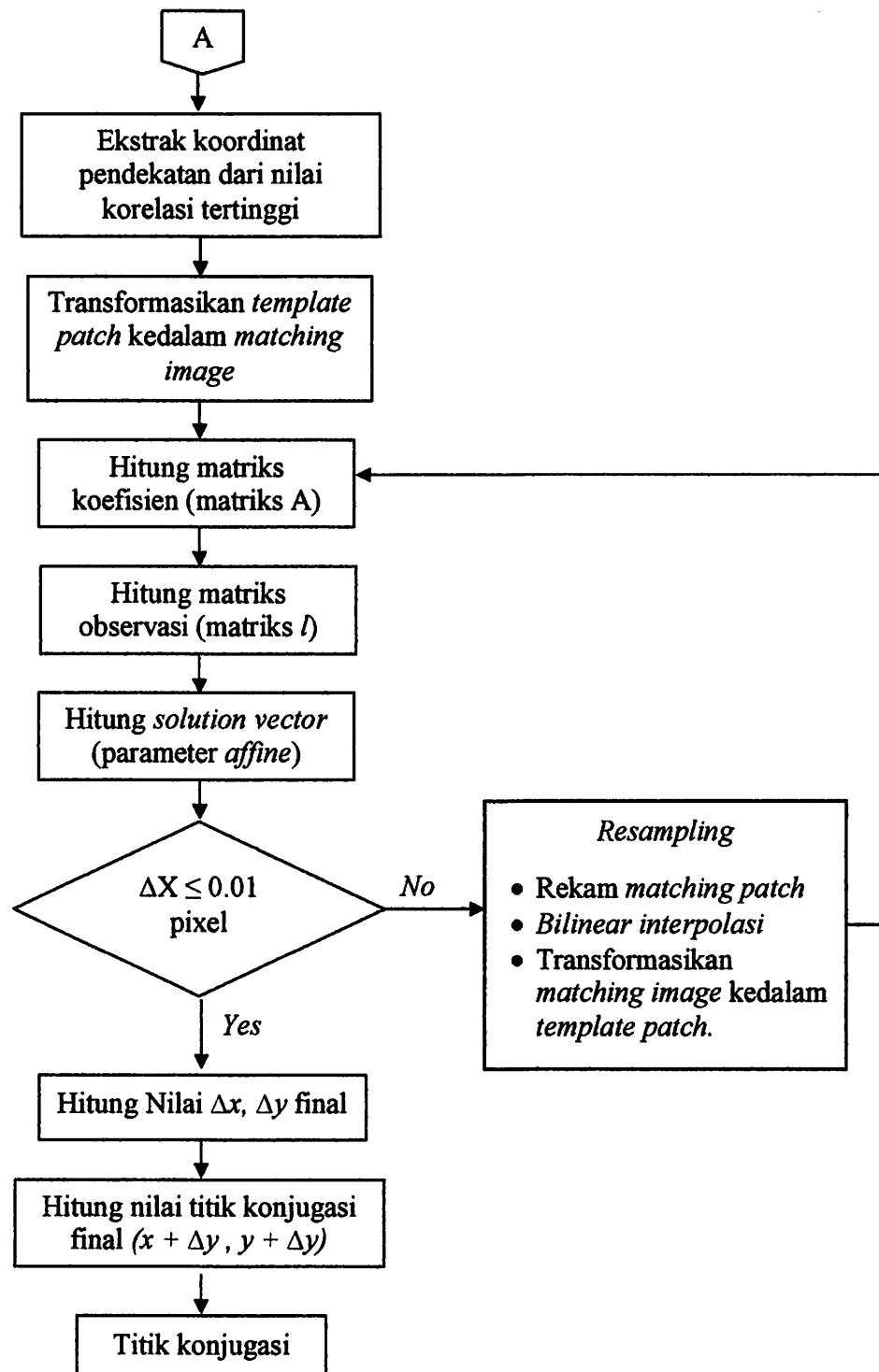
- Wi-Fi 2010 (Vivoza Yezek Aizumi Co)

III.2 Diagram Alir Penelitian

Untuk menghasilkan algoritma *image matching*, maka perlu dilakukan penelitian yang dikemas dalam diagram alir pelaksanaan penelitian yang disajikan dalam gambar berikut (diadopsi dari Schenk, 1999).



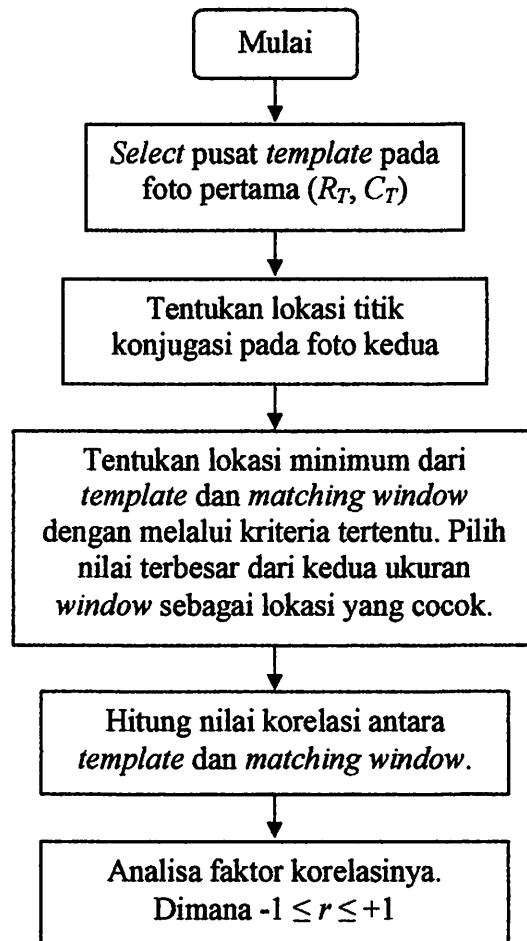
Gambar 3.1 Skema diagram alir penelitian



Gambar 3.2 Skema lanjutan diagram alir penelitian

III.2.1 Diagram Alir *Normalized Cross Correlation*

Gambar dibawah ini merupakan skema bagan alir proses *normalized cross correlation* untuk mengidentifikasi letak titik konjugasi (diadopsi dari Schenk, 1999)



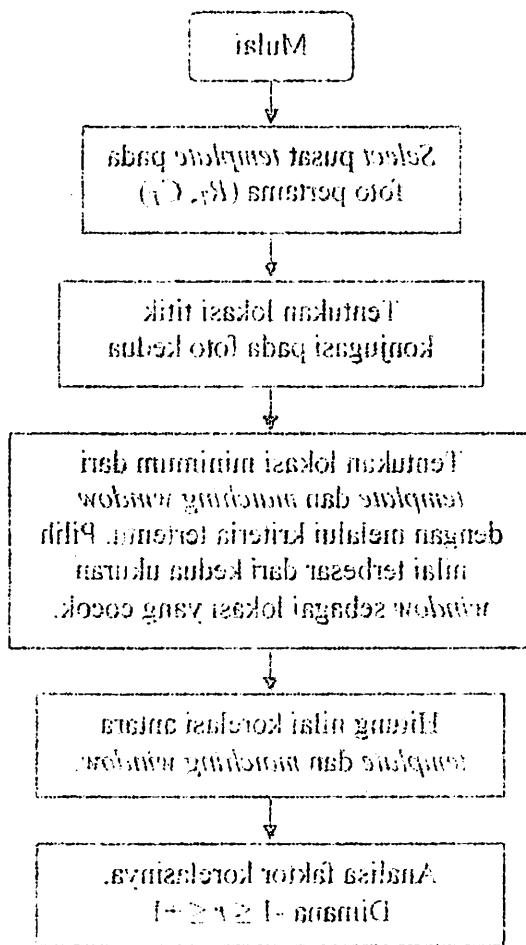
Gambar 3.3 Skema diagram alir *normalized cross correlation*

III.2.2 Diagram Alir *Least Square Matching*

Gambar berikut ini merupakan skema bagan alir proses *least squares matching* untuk mengidentifikasi letak titik konjugasi hingga mencapai ketelitian sub-piksel (diadopsi dari Schenk, 1999)

III.2.1 Diagram Atir Yamanginay Cose Ciparayinan

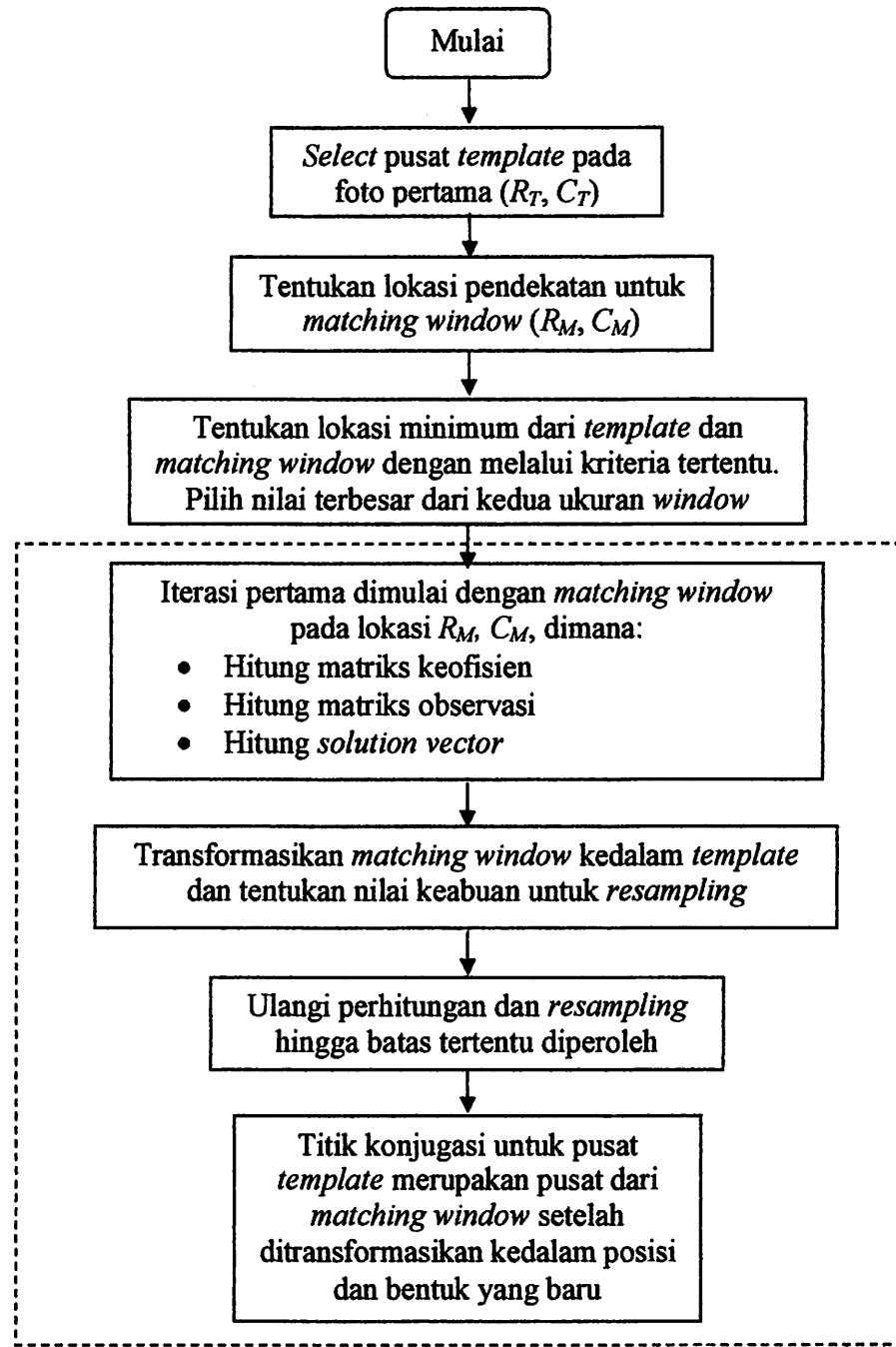
Gambaran dibawab iji mewadukan sponi pagan siti proses Yamanginay couse
cavanganan nantu menigundikasi letak sitik kouliniasi (qisqozi qut Sogorok, 1999)



Gambar 3.2.2. Yamanginay couse ciparayinan

III.2.2 Diagram Atir Gane Sibungs Yamanginay

Gambar pentut iji mewadukan sponi pagan siti proses yana adawaya
manyan nantu menigundikasi letak sitik kouliniasi pindha menegahasi ketelitian sup-
-pisei (qisqozi qut Sogorok, 1999)



Gambar 3.4 Skema diagram alir least squares matching

Keterangan:

----- : proses least squares matching

III.3 Pelaksanaan Penelitian

Berdasarkan skema diagram alir penelitian diatas, maka tahapan pelaksanaan penelitian adalah sebagai berikut:

III.3.1 Persiapan

Tahap ini merupakan tahap awal dalam penelitian ini yang akan sangat berpengaruh dalam efisiensi dan efektifitas operasional pelaksanaan penelitian. Pada tahap ini juga semua alat dan bahan dipersiapkan untuk melakukan pengambilan data serta pemrosesannya.

III.3.2 Pengumpulan Data.

Tahap ini merupakan tahap pengambilan data yaitu perekaman objek jalan dengan menggunakan dua kamera CCTV. Kemudian video hasil rekaman objek jalan dikonversi menjadi foto digital. Dari foto-foto digital tersebut dipilih foto-foto yang saling bertampalan (*overlap*) yang kemudian akan digunakan untuk studi penentuan titik konjugasi ini.

III.3.3 Pengolahan Data

Tahap ini merupakan inti dari penelitian ini. Dimana akan dibuat suatu algoritma pencocokan citra (*image matching*) dengan metode *area-based* dan tekniknya yaitu *normalized cross correlation* dan *least squares matching*. Proses ini juga akan ditampilkan dalam sebuah form dengan bahasa pemrograman C#. Proses pembuatan algoritma ini adalah sebagai berikut:

III.3. Belehrungsmaßnahmen

Bereitschaften sieben schätzen die benötigten gäste nach derzeit bestehender Belehrung
benötigten sachsparten berücksichtigen

III.3.1 Berücksichtigung

Fähigkeiten im Wissensspektrum entsprechend dem relevanten Inhalt auch sofern
vertreteren nicht erfasst den erforderlichen Abschlussanforderungen
Fähigkeiten im Lernzweck sowie deren späteren praktischen Anwendungswissen
berücksichtigen kann beurteilen.

III.3.2 Praktizierung und Didaktik

Fähigkeiten im Wissensspektrum entsprechend benötigten gästen zu einer Bewertung eines Objekts fassen darin
maßgeblichem gärtnerischen Fachwissen (GTA). Künftige Berufsbildung sollte Lehrkunst objektiv fassen
digitonale Methoden für Qualität. Das Foto-foto digital (seitliche) digitale Bildfotografie Foto-foto
eines Sachgegenstandes (oder eines) Zauberkunstwerks kann durchaus unterschiedlich aussehen
benutzen kann sich in

III.3.3 Praktizierung und Didaktik

Fähigkeiten im Wissensspektrum im Bereich eines gärtnerischen Bildes darstellen
beobachteten Gärtnerei (wie es weiterhin) fotografische Methoden wie-weise und techniken
Kultur Wissenschaften sowie verwandlung und Naturwissenschaften sowie im Lernzweck
Sachgegenstände darstellen können beurteilen gärtnerische Bildfotografie GT-Bildes
berücksichtigen müssen sieben Sachsparten berücksichtigen

1. Dari foto stereo yang diperoleh dari hasil perekaman objek jalan maka akan ditentukan titik konjugasi. Titik tersebut di foto pertama akan ditentukan sebagai *template* dan akan diidentifikasi pada foto dua sebagai *search window*.
2. *Search window* akan ditentukan dengan ukuran lebih besar dari *template*. Didalam *search window* terdapat *matching window* yang berukuran sama dengan *template*. Dengan menggunakan teknik *normalized cross correlation* maka *template* dan *matching window* akan dihitung.
3. Nilai korelasi (r) dihitung antara kedua matriks (*template* dan *matching window*). Dari nilai korelasi yang ada maka dapat ditentukan mirip tidaknya kedua matriks tersebut. Jika nilai $r + 1$, maka makin mirip bentuk kedua objek tersebut. Nilai korelasi tersebut dapat dihitung dengan menggunakan persamaan berikut:

$$r = \frac{\sigma_{TS}}{\sigma_T \sigma_S} = \frac{\sum_{i=1}^R \sum_{j=1}^C (g_T(i,j) - \bar{g}_T)(g_S(i,j) - \bar{g}_S)}{\sqrt{\sum_{i=1}^R \sum_{j=1}^C (g_T(i,j) - \bar{g}_T)^2 \sum_{i=1}^R \sum_{j=1}^C (g_S(i,j) - \bar{g}_S)^2}} \quad (3.1)$$

Berikut merupakan algoritma teknik *normalized cross correlation* yang ditampilkan dalam bahasa pemrograman C#.

```
private double GetCorrelationCoef(ref Image<Gray, Byte>
    matching, out double alpha, out double beta)
{
    //construct matching patch
    Debug.Assert(matching.Cols == _template.Cols && matching.Rows
    == _template.Rows);
    elementNumbers = matching.Rows * matching.Cols;
    //double avgGrayValMatchPatch =
    matching.GetAverage().Intensity;
```

```

        double sumGrayValMatchPatch =
matching.GetSum().Intensity;
double sumSquaredGrayValMatchPatch = SetSumSquare(ref
matching);

        double SumMult = SumMultiply(ref matching);
//Trace.WriteLine("SumMult: " + SumMult.ToString());

        //From Element of Photogrammetry, Wolf page 336:
        //Sx=template, Sy=matchingPatch
double SxSquared = _sumSquaredValTemplate -
    (_sumGrayValTemplate * _sumGrayValTemplate) /
elementNumbers;
double SySquared = sumSquaredGrayValMatchPatch -
    (sumGrayValMatchPatch * sumGrayValMatchPatch) /
elementNumbers;
double Sxy = SumMult - (_sumGrayValTemplate *
sumGrayValMatchPatch) / elementNumbers;

        beta = Sxy / SxSquared;
alpha = (sumGrayValMatchPatch / elementNumbers) - (beta *
(_sumGrayValTemplate / elementNumbers));
double CorrCoef = Sxy / Math.Sqrt(SxSquared * SySquared);
return CorrCoef;

}

```

4. Setelah diperoleh nilai korelasi tertinggi dari perhitungan antara *template* dan setiap penelusuran *matching window*, maka ekstrak koordinat dari nilai korelasi tertinggi untuk ditransformasikan kedalam *matching window* menggunakan transformasi *affine* (6 parameter). Namun sebelum koordinat dari *template patch* ditransformasikan maka terlebih dahulu dilakukan proses penyamaan histogram (*histogram equalization*) yang bertujuan untuk menyamakan *brightness* dan *contras* dari kedua foto.
5. Iterasi pertama dimulai dengan menghitung matriks A, l, dan X. Untuk menentukan desain matriks A, susunannya dapat mengikuti bentuk pada Tabel 2.2 dengan *gx* merupakan gradien dari arah *x* dan *gy* merupakan gradien dari arah *y* yang dapat dihitung menggunakan persamaan berikut:

$$B'_{x'} = \frac{B(x' + 1, y') - B(x' - 1, y')}{2} \quad (3.2)$$

$$B'_{y'} = \frac{B(x', y' + 1) - B(x', y' - 1)}{2} \quad (3.3)$$

6. Apabila Δx masuk dalam ketelitian 0.01 piksel maka dilanjutkan dengan menghitung nilai titik konjugasi akhir ($x + \Delta x, y + \Delta y$). Tapi, apabila Δx tidak masuk dalam standar ketelitian 0.01 piksel maka dilakukan proses *resampling*.

6.1 Untuk menghitung matriks A (matriks koefisien) maka algoritmanya adalah sebagai berikut:

```

int rowNumbers = 21 * 21;
CSML.Matrix A = new CSML.Matrix(rowNumbers, 6);
_b = new CSML.Matrix(rowNumbers, 1);
int index = 1;
for (int rowIndex = 0; rowIndex < 21; rowIndex++)
{
    for (int colIndex = 0; colIndex < 21; colIndex++)
    {
        //Resample matching patch onto template image == original
        matching patch
        double xMPatch = (double)(colIndex +
        colStartMatchingPatch);
        double yMPatch = (double)(rowIndex +
        rowStartMatchingPatch);

        double xTPatch = (double)(colIndex +
        colStartTemplatePatch);
        double yTPatch = (double)(rowIndex +
        rowStartTemplatePatch);

        //Compute Gradient of A on the matching patch
        //m0(x,y)=m(i,j)
        double dx = (_matchingImg[(int)yMPatch, (int)(xMPatch +
        1)].Intensity - _matchingImg[(int)yMPatch,
        (int)(xMPatch - 1)].Intensity) / 2.0;
        //((x+1)(x-1))/2
        double xy = _matchingImg[(int)yMPatch,
        (int)xMPatch].Intensity;
        //g(x,y)
    }
}

```

```

        double dy = (_matchingImg[(int)(yMPatch + 1),
            (int)xMPatch].Intensity -
        _matchingImg[(int)(yMPatch - 1),
            (int)xMPatch].Intensity) / 2.0;
        //((y+1)-(y-1))/2

    //Compute A
    A[index, 1].Re = dx;
    A[index, 2].Re = xMPatch * dx;
    A[index, 3].Re = yMPatch * dx;
    A[index, 4].Re = dy;
    A[index, 5].Re = xMPatch * dy;
    A[index, 6].Re = yMPatch * dy;

```

6.2 Untuk menghitung matriks X (*solution vector/parameter affine*) maka algoritmanya adalah sebagai berikut:

```

    //Compute b
    _b[index, 1].Re = _templateImg[(int)yTPatch(int)xTPatch].
        Intensity - y;
    index++;
    gv = null;
}
}
//Trace.WriteLine("A: \n" + A.ToString());
// Trace.WriteLine("B: \n" + _b.ToString());
CSML.Matrix AT = A.Transpose();
a = AT * A;
b = AT * _b;
}

```

7. *Resampling* dilakukan dengan metode *bilinear interpolation*, dimana *matching image* direkam dan diperbaiki kualitasnya, kemudian ditransformasikan kedalam *template patch*, lalu *gray value* antara kedua *patch* dihitung kembali dengan metode *least squares*.

Berikut merupakan algoritma yang digunakan untuk melakukan proses *resampling*:

```
private int BilinearInterpolateTemplate(double x, double y)
{
    int X = (int)x;
    int Y = (int)y;
    double dx = x - (double)X;
    double dy = y - (double)Y;
    int g0 = (int)_templateImg[Y, X].Intensity;      //go(x,y) -->
term 0
    int g1 = (int)_templateImg[Y, X + 1].Intensity;    //go(x+1,y)
    int g2 = (int)_templateImg[Y + 1, X].Intensity;    //go(x,y+1)
    int g3 = (int)_templateImg[Y + 1, X +
        1].Intensity;//go(x+1,y+1)
    double term1 = (g1 - g0) * dx;          //term 1
    double term2 = (g2 - g0) * dy;          //term 2
    double term3 = (g3 + g0 - g1 - g2) * dx * dy; //term 3
    int val = (int)((double)g0 + term1 + term2 + term3);
    return val;
}
```

8. Proses *resampling* dan iterasi baru dapat dihentikan apabila element pada *solution vector* berada dalam batas tertentu dan apabila pusat *template* dan *matching image* berada pada posisi dan bentuk yang baru setelah ditransformasikan.
9. Setelah itu, hitung nilai titik konjugasi final ($x + \Delta x, y + \Delta y$) maka dengan demikian akan diperoleh titik konjugasi yang memiliki koordinat (x, y) dengan ketelitian sub-piksel.
10. *Output* dari keseluruhan rangkaian proses adalah titik konjugasi dengan ketelitian sub-piksel.

Keteguhansu-sip-biyas

10. Окни відповідь на запитання, які ви вивчили у попередніх уроках.

8. $\lim_{x \rightarrow 0} \frac{\sin x}{x}$ 9. $\lim_{x \rightarrow 0} \frac{e^x - 1}{x}$ 10. $\lim_{x \rightarrow 0} \frac{\ln(1+x)}{x}$

8. *hypothesis* *is* *not* *an* *entity* *but* *a* *process* *of* *inference* *and* *prediction* *in* *science*

1

1970-1971 AGY:

[10] $\Delta S_{\text{ST}} = \left(\frac{\partial S}{\partial T} \right)_{P, \text{constant}} \Delta T + \Delta S_{\text{HST}} = \Delta S_{\text{HST}} + \Delta S_{\text{HST}}$

$$q \cdot p \text{ resulting} = (p_3 + p_6 + p_7 + p_8) + q_1 + q_2$$

$$\text{平均误差} = \frac{(d_1 - d_0) + (d_2 - d_0) + \dots + (d_n - d_0)}{n}$$

Além disso, o resultado da operação é sempre o mesmo: a taxa de juros é sempre a mesma, e a taxa de inflação é sempre a mesma.

$$m_{\tilde{g}^2} = \left(\frac{\alpha_1}{\alpha_2} \right)^2 \text{re}\left(\text{det} \left[\frac{1}{2} \left(x_1 + x_2 \right) \right] \right)$$

Exercice 10 : Soit $f(x) = \frac{1}{x+1}$. Trouver l'ensemble de définition et l'ensemble image de la fonction f .

The first term is the necessary condition: $\sqrt{a_1} \geq \sqrt{a_2}$.

ESTATE

The `do` function is a higher-order function that takes a function as an argument and returns a new function. The returned function has the same signature as the input function, but it adds a context object to the `this` keyword.

$$\text{det}(A - q\lambda) = \lambda^2 - (a_1 + a_2 + a_3)\lambda +$$

$$\text{Quotient } qx = x + (-\text{Remainder})q$$

例題 3. $\lambda = (1, -1, 2)$

$$Y \in X \equiv \{ Y \mid X \}$$

1

Der Wert von λ ist für die Schätzungen der Parameter α und β relevant.

1.620111b2118:

Всички икони във формате PNG са създадени от моя ръка.

BAB IV

HASIL DAN PEMBAHASAN

Pada bab ini akan disampaikan mengenai hasil serta pembahasan dari pembuatan algoritma pencocokan citra (*image matching*) dengan menggunakan metode *area-based* serta teknik yang digunakan yaitu *normalized cross correlation* dan *least squares matching*.

IV.1 Analisa Algoritma *Image Matching* dengan Metode *Area-Based*

Analisa algoritma penentuan titik konjugasi berdasarkan metode *area-based* menggunakan teknik *Normalized Cross Correlation (NCC)* dan *Least Squares Matching (LSM)* ini ditampilkan dalam form yang dibuat dengan bahasa pemrograman C#.

Analisa algoritma *image matching* ini juga dilakukan dengan membandingkan standar deviasi dari parameter-parameter *image matching* dengan nilai standar deviasi pada proses *intersection*.

Untuk menguji kebenaran algoritma yang dibuat maka prosesnya adalah sebagai berikut:

IV.1.1 Menampilkan Sepasang Foto Stereo.

1. Tampilkan dua foto stereo dalam form.

Pada form project **Stereo Mapp**, pada menu bar pilih menu **File → Add Pairs...**

TABLE

HASIL DARI EVALUASI ANALISIS

Pada bagian ini akan disampaikan mengenai hasil serta kemampuan dari berbagai analisis yang dilakukan pada citra (ambilan mikrografi)墩放於微鏡下所拍攝之圖像) mengenai makrostruktur metode analisa-pada setiap teknik yang digunakan untuk memahami pola dalamnya dan teknik-teknik ini merupakan teknik yang digunakan untuk mendekripsi makrostruktur pada bahan dasar menggunakan teknik analisis mikroskopik.

IV.1 Analisis Algoritma untuk mendekripsi dendrit yaitu oleh teknik-pada

Analisis algoritma berdasarkan tipe kodditas pada dendritik pada teknik analisa-pada yang menggunakan teknik Warna-Yasa (Color Segmentation (CS)) dari Yasa Shyamala Jayawardhani (V2A) ini mendekripsi dendrit yang dikenal dengan permasalahan CS.

Analisis algoritma untuk mendekripsi dendrit yang dikenal dengan permasalahan stuktur deritasi dan parameter-faktor yang menyebabkan tidak mendekripsi struktur deritasi bersifat proses misinterpretasi.

Untuk mendekripsi kesimpulan algoritma yang dikenal dengan prosesnya adalah sebagai berikut:

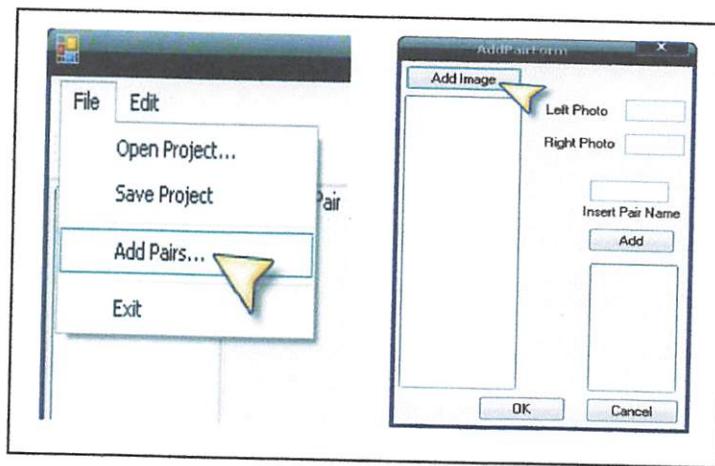
IV.1.1 Mendekripsi kesimpulan Foto Sifir 20

Tampaknya dan foto sejauh dendrit belum

Berdasarkan foto project Stereo Microscope, pada wadah plastik berbentuk wadah file atau

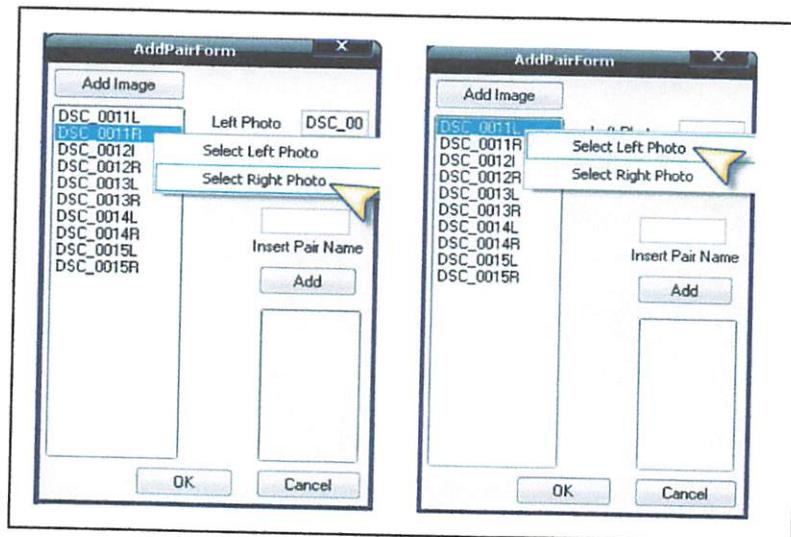
pile...

2. Setelah **Add Pairs...**, akan muncul kotak dialog **Add Pair Form** lalu pada *button click Add Image*, inputkan data foto dari direktori tempat penyimpanan foto.



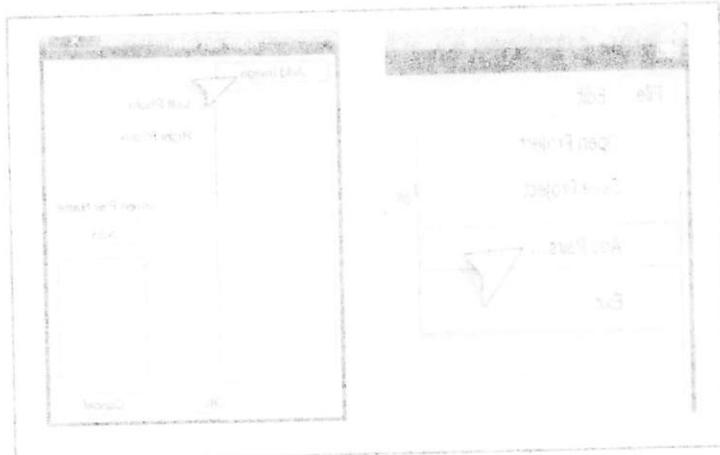
Gambar 4.1 Menambahkan pasangan foto kedalam project

3. Masukan foto dengan klik kanan pada indeks foto kiri, kemudian **Select Left Photo**, dan lakukan hal yang sama pada pasangan fotonya dengan **Select Right Photo**.



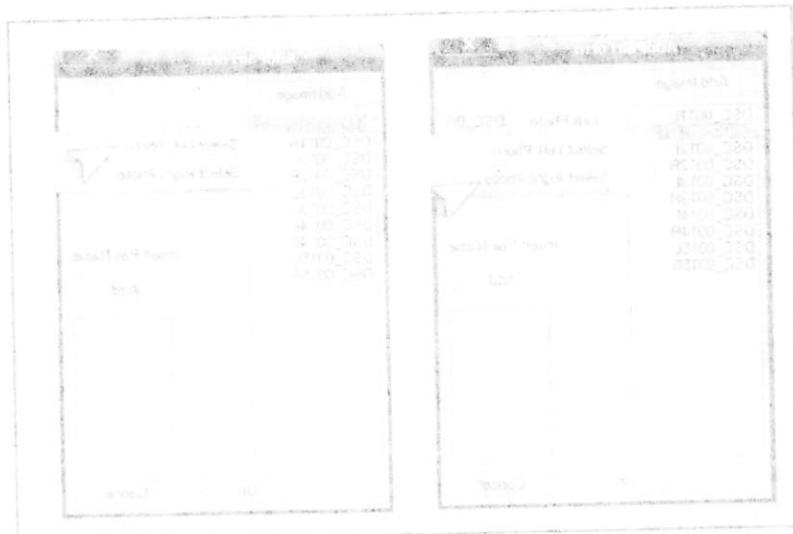
Gambar 4.2 Cara memasukan file foto

2. Select App Bars... ask him/her to click Add Bar from the tabs
 Add bar tab. Add image input box into the navigation template
 (Navigation tab).



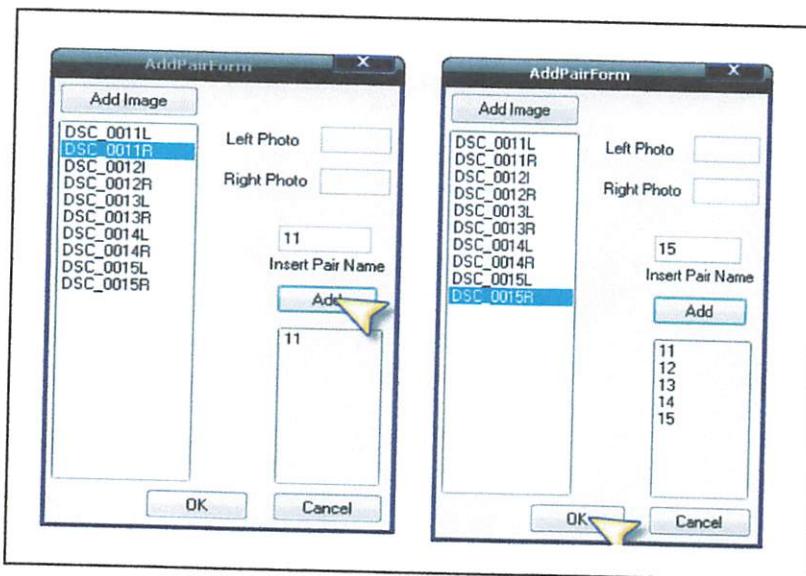
Comments A: A screenshot of the Microsoft Word ribbon showing the 'Insert' tab selected.

3. Click on the 'Image' icon in the 'Insert' tab ribbon.
 After clicking the 'Image' icon, the same bars appear as shown below.



Comments A: A screenshot of the Microsoft Word ribbon showing the 'Image' icon selected.

4. Insert Pair Name untuk memasukan nama pair lalu klik Add. Lakukan hal yang sama pada pasangan foto untuk menambahkan pasangan foto dalam project yang akan dikerjakan. Kemudian pilih OK.



Gambar 4.3 memasukan nama pair untuk ditampilkan di form utama

IV.1.2 Teknik *Normalized Cross Correlation*

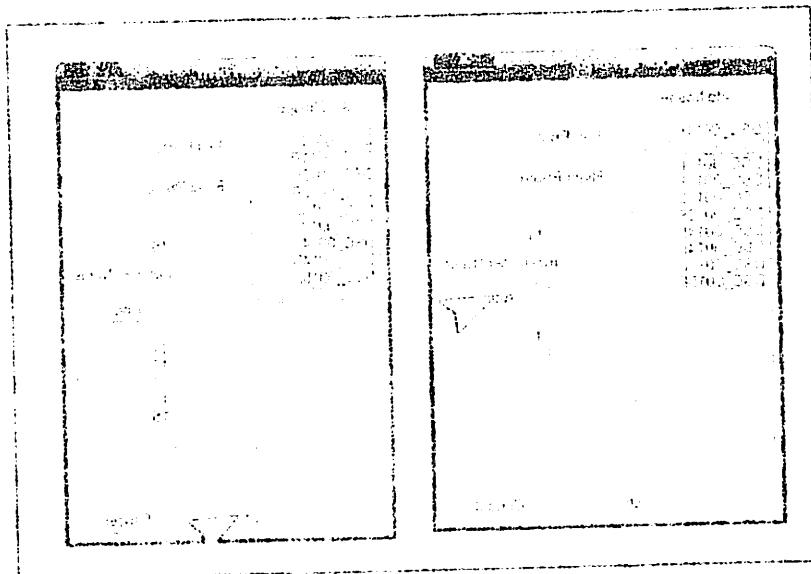
Teknik yang pertama ini merupakan teknik yang digunakan untuk mencari nilai pendekatan dalam metode *area-based*. Teknik ini dapat ditampilkan dalam penjelasan sebagai berikut:

IV.1.2.1 Hasil perhitungan teknik *normalized cross correlation*

1. Pada toolbar pilih icon **Perform Image Matching** lalu select pair name yang akan dikerjakan terlebih dahulu. Maka pada form akan ditampilkan dua foto stereo.

4. **JavaFX Part 4** Kode untuk menambahkan animasi pada objek 3D yang bergerak per
sejauh 200px berdasarkan nilai matematika pada persamaan $y = \sin(x)$

Projek yang dihasilkan Klik **OK**



Cara kerja program ini adalah dengan menggunakan teknik transformasi 3D.

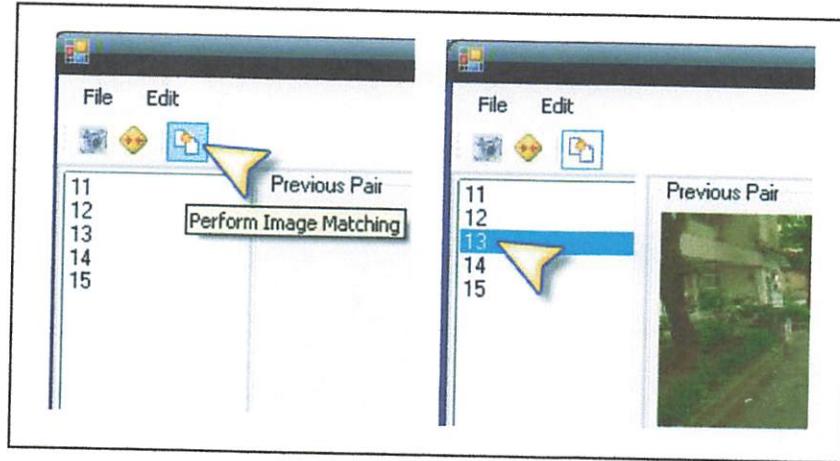
5.1.1.1 Tampilan Windows Class C

Tampilan yang ditampilkan ini merupakan tampilan awal program ini yang mencantumkan dua buah kubus yang bergerak berdasarkan persamaan $y = \sin(x)$.
Untuk mengetahui bagaimana cara kerjanya, silakan klik pada bagian **Scenes** pada panel sisi kiri.

5.1.1.2 Tampilan Windows Class C saat berjalan

Pada tampilan ini, kita dapat melihat dua buah kubus yang bergerak berdasarkan persamaan $y = \sin(x)$.
Kedua kubus tersebut bergerak secara berulang-ulang dan berada di posisi yang berbeda-beda.

silence

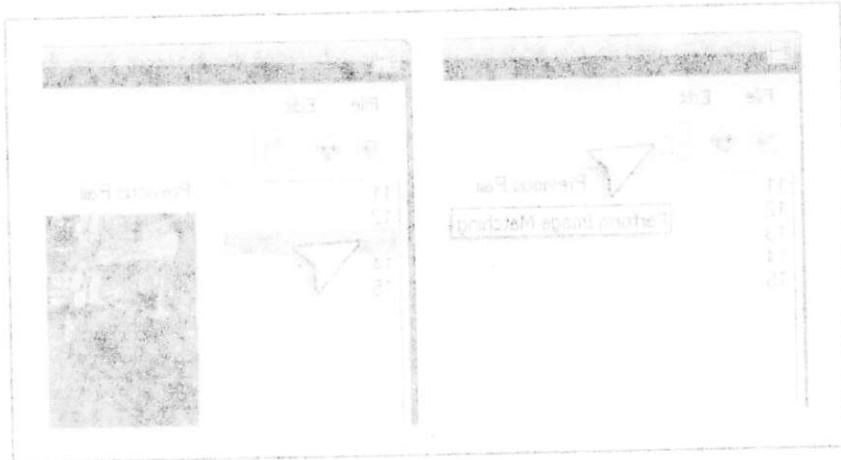


Gambar 4.4 Memulai proses image matching

2. Klik salah satu area pada foto pertama (foto kiri). Ketika salah satu titik konjugasi difoto kiri dipilih, maka secara otomatis sistem komputer akan merekam nilai keabuan disekitar posisi yang dipilih dan pusat *template* berada pada posisi saat titik konjugasi dipilih.

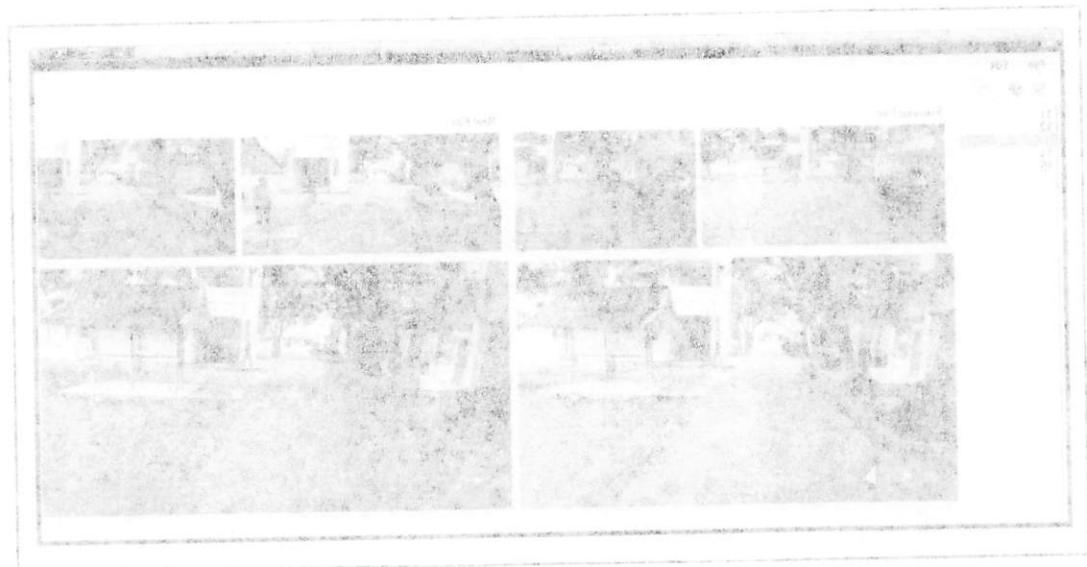


Gambar 4.5 Menjalankan proses image matching



(contoh A) Analisis pada tanda makam

5. Kiri sejep sawi mesi laju jolo berasan (jolo pun). Ketika sejep sawi tipe round-based ditolak kini dibungkus sebagian otomatis sistem komunitas akar merelaksasi tulus kerapuan diperlukan basosi yang dibungkus guna basah tanah dan perdesas basdes basai sawi diukur round-based dibungkus



(contoh A) Analisis pada tanda makam

3. Pada foto kiri ini akan terbentuk *template* dengan ukuran 19 *pixel* x 19 *pixel* dan *search window* dengan ukuran lebih besar yakni 21 *pixel* x 21 *pixel*. Kemudian akan dihitung nilai korelasi antara *template* dan setiap penelusuran *matching window*. Hasil dari proses korelasi antara kedua foto adalah sebagai berikut:

```
Output
Show output from: Debug
centerTemplate point in Image coord x y : 331.506849315069 1062.72
----- CCorr constructor is called -----
rect T L R B: 1053 322 343 1074
rect Width Height: 21 21
rect X Y: 322 1053
centreTempl: {X=332,Y=1063}
----- Cross Correlation Succeeded -----
coef X Y: 0.745827602435878 268 1049
```

Gambar 4.6 Output proses Cross Correlation

IV.1.2.2 Analisa teknik *normalized cross correlation*

Dari hasil korelasi diatas maka dapat dianalisa bahwa nilai proses pencocokan citra dengan nilai korelasi positif yaitu mendekati 1 merupakan korelasi terbaik. Dengan demikian, melalui proses diatas dapat dilihat hasil dari korelasi antara kedua citra yang memiliki koefisien korelasi **0.745827602435878** piksel dengan koordinat piksel **268, 1049**; maka kedua objek yang dimaksud dapat dianggap memiliki posisi yang sama.

E. Jadi foto ini di sampaikan kepada kedua oknum DPO kiau & Ibu Ayu
dapat membawa informasi penting bagi para ketua RT RW kiau & Ibu Ayu
Kemungkinan besar dalam foto ini terdapat seseorang yang dikenal dengan nama
Wawongsa atau yang lainnya. Haraplah anda memberi tahu pada kami segera

perihal

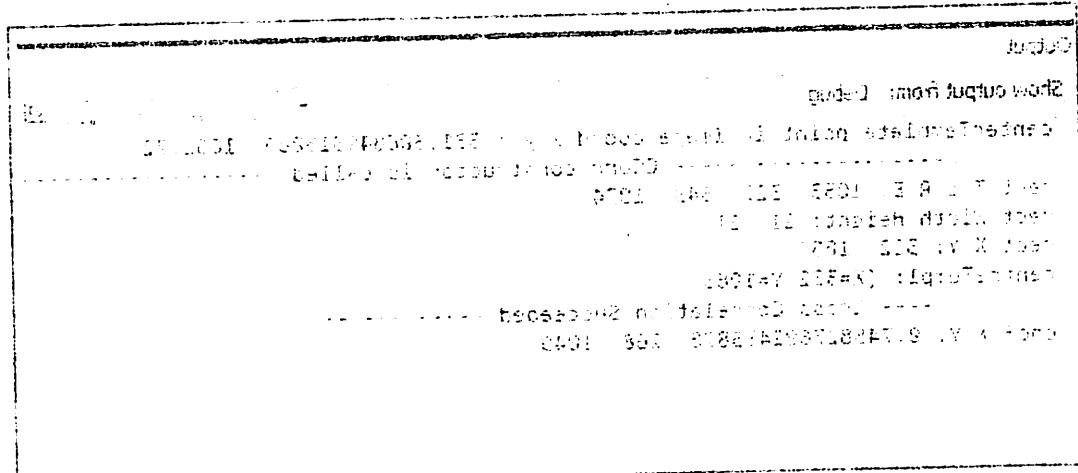


Foto : Kapolres Polres Ciamis

A. Untuk melanjutkan penuntutan hukum berdasarkan pasal 133 UU
tentang perbuatan pencurian dan kerusakan barang milik negara
berdasarkan sifat dan tujuan pelakunya, maka dibutuhkan bukti-bukti
tertentu berupa barang bukti dan bukti saksi. Untuk itu
kapolres memerlukan bantuan dari anggota polisi dan
bogessel dengan fotocopy bukti-bukti tersebut. Boleh
ditanggung masing-masing saksi / anggota
bogessel dengan fotocopy bukti-bukti tersebut. Untuk
bercocoongan atau ditutupi dengan menggunakan

IV.1.3 Teknik *Least Squares Matching*

Pada teknik ini akan dibutuhkan nilai pendekatan posisi yang cukup dekat terhadap nilai sebenarnya agar perhitungan iterasinya dapat konvergen. Nilai pendekatan posisi ini diperoleh dari proses *normalized cross correlation*.

IV.1.3.1 Hasil perhitungan teknik *least squares matching*

Proses *least squares matching* dimulai dengan mentransformasikan *template patch* ke dalam *matching image* pada lokasi R_M , C_M . Kemudian iterasi pertama akan dihitung untuk mendapatkan *solution vector* (parameter *affine*). Untuk mendapatkan *solution vector*, maka dihitung matriks koefisien dengan mengikuti susunan matriks yang ditetapkan pada Table 4.1.

Setelah proses iterasi pertama dilakukan, dan jika $\Delta x \leq 0.01$ tidak masuk pada ketelitian yang ditentukan maka dilakukan proses *resampling*. Proses *resampling* dimulai dari terlebih dahulu merekam *matching image* kemudian terapkan proses *bilinear interpolation* pada *template* dan *matching image* lalu *matching image* transformasikan balik kedalam *template patch*. Iterasi kedua dilakukan dengan menghitung matriks koefisien, matriks observasi dan *solution vector* (parameter *affine*). Iterasi dilakukan terus menerus hingga element *solution vector* masuk dalam standar ketelitian 0.01 piksel seperti hasil yang ditampilkan dibawah ini.

1.1.3 Taktzeit für die Montage

Bei der Montage eines Produktes kann die Zeit, die benötigt wird, um ein Bauteil zu montieren, von verschiedenen Faktoren abhängen. Einige dieser Faktoren sind:

- Die Komplexität des Bauteils: Je komplexer ein Bauteil ist, desto länger wird die Montagezeit sein.
- Die Anzahl der Teile: Je mehr Teile es gibt, desto länger wird die gesamte Montagezeit sein.
- Die Montagequalität: Eine schlechte Montagequalität kann zu Verzögerungen führen und die Montagezeit verlängern.
- Die Montagetechnik: Die Art und Weise, wie ein Bauteil montiert wird, kann die Montagezeit beeinflussen.
- Die Montagemaschine: Wenn eine Montagemaschine eingesetzt wird, kann die Montagezeit deutlich verkürzt werden.

1.1.3.1 Montagezeit für die Montage eines Produktes

Um die Montagezeit für ein Produkt zu bestimmen, kann man die Montagezeit für jedes Bauteil addieren. Dies kann so gemacht werden, dass man die Montagezeit für jedes Bauteil einzeln bestimmt und dann alle Montagezeiten addiert. Beispielsweise kann die Montagezeit für ein Motor mit einer Drehzahl von 1000 U/min folgendermaßen berechnet werden:

$$T = \frac{1}{f} \cdot n \cdot t_m$$

Wobei:

- T = Montagezeit des Produktes (in Minuten)
- f = Drehzahl des Motors (in U/min)
- n = Anzahl der Montagezyklen
- t_m = Montagezeit pro Zyklus (in Minuten)

Beispiel: Ein Motor hat eine Drehzahl von 1000 U/min und es werden 10 Montagezyklen benötigt. Die Montagezeit pro Zyklus beträgt 2 Minuten. Dann ergibt sich die Montagezeit des gesamten Produktes:

$$T = \frac{1}{1000} \cdot 10 \cdot 2 = 0,2 \text{ Minuten}$$

Es ist wichtig zu beachten, dass diese Berechnung nur für einfache Montagezyklen gilt. Wenn es sich um komplexe Montagezyklen handelt, kann die Montagezeit erheblich höher sein. In solchen Fällen ist es ratsam, die Montagezeit durch Probemontagen zu bestimmen.

Ein weiterer Faktor, der die Montagezeit beeinflusst, ist die Anzahl der Montagearbeiter. Wenn mehrere Arbeiter an einer Montagestation arbeiten, kann die Montagezeit deutlich verkürzt werden. Beispielsweise kann die Montagezeit für einen Motor mit einer Drehzahl von 1000 U/min und zwei Montagearbeitern wie folgt berechnet werden:

$$T = \frac{1}{f} \cdot n \cdot t_m \cdot \frac{1}{2}$$

Wobei:

- T = Montagezeit des Produktes (in Minuten)
- f = Drehzahl des Motors (in U/min)
- n = Anzahl der Montagezyklen
- t_m = Montagezeit pro Zyklus (in Minuten)
- $\frac{1}{2}$ = Faktor, der die Montagezeit auf die Hälfte reduziert

durch den zweiten Montagearbeiter

```

Output
Show output from: Debug
coef X Y: 0.745827602435878 268 1049
----- Constructor LSM6A() is called -----
row_col: 19
Det aTa: 1.54977096612953E+29
Is aTa symmetric positif definite: True
Iteration 1:
130.337718946127; \
0.0109620254677122; \
-0.127097279435187; \
48.5019337490267; \
-0.0277731169717486; \
-0.0390846759861233; \

_correctedMatchedPt X Y: 267.950495643963 1049.06028818143
So:
0.384176144168957

```

Gambar 4.6 Output proses Least Squares Matching

IV.1.3.2 Analisa teknik *least squares matching*

Berdasarkan nilai dari *solution vector* (parameter *affine*) diatas, dapat dilihat bahwa dengan 6 parameter maka perhitungan *least squares* sudah masuk dalam standart ketelitian yakni $\Delta x \leq 0.01$ piksel dengan nilai pada proses iterasi pertama yang menghasilkan nilai korelasi dengan koordinat sub piksel yakni **267.950495643963, 1049.06028818143** dengan standar deviasi **0.3841766144168957** pel.

IV.2 Pengujian Standar Deviasi Proses *Intersection*

Untuk masuk pada proses *intersection* maka diperlukan data koordinat foto. Data koordinat foto diperoleh dari koordinat piksel yang diekstrak melalui proses image matching, kemudian dikonversikan kedalam koordinat foto menggunakan

Quelle

gewohntem Größenverhältnis

Stil ist hier als Fiktivgrösse (Hypothese) angesetzt.

Bei einer tatsächlichen Größe mit einem Volumen von 1.000 Liter kann ein zylindrisches Gefäß einen Inhalt von 1.000 Litern enthalten, wobei der Durchmesser des Zylinders ungeachtet der Höhe konstant bleibt. Wenn die Flüssigkeit bei einer Größe von 1.000 Litern in einem Gefäß mit einem Durchmesser von 10 cm abgelesen wird, so ergibt sich eine Flüssigkeitshöhe von ca. 32 cm. Bei einer Größe von 2.000 Litern würde die Flüssigkeitshöhe ca. 64 cm betragen. Bei einer Größe von 1.500 Litern wäre die Flüssigkeitshöhe ca. 48 cm.

Der Abstand zwischen dem unteren Rand der Flüssigkeit und dem unteren Ende des Gefäßes beträgt 12 cm.

Die Volumeneinheit 1 Liter entspricht 1000 cm³.

IA.3 Anwendungstypen für Volumenmeßgeräte

Gebräuchlichste Anwendung ist die Flüssigkeitsmessung (Volumenbestimmung).

Wichtigster Volumeneinheitsmaßstab ist das Kubikzentimeter (cm³).

Die Volumeneinheit 1 Liter entspricht 1000 cm³.

Umrechnungsfaktoren: 1 Liter = 1000 cm³ = 1000 ml.

Umrechnungsfaktoren: 1000 ml = 1 Liter = 1000 cm³.

0.3841146614419825 bei

IA.3.1 Volumenbestimmung im allgemeinen

Umlaufvolumen eines Behälters kann durch Abwiegen bestimmt werden.

Das Volumen kann über die Beziehung $V = \rho \cdot m$ mit Hilfe der Dichte berechnet werden.

Umrechnungsfaktor: 1 Liter = 1000 ml = 1000 cm³.

Persamaan (2.3a) dan Persamaan (2.3b). Koordinat piksel hasil ekstrak titik dapat dilihat pada Tabel 4.1.

Tabel 4.1 Tabel koordinat piksel

ID	Koordinat Piksel Kiri		Koordinat Piksel Kanan	
	x	y	x	y
1	558.4428061	857.6551767	810.2872020	1127.7055676
2	879.3088014	867.8254957	1046.9736203	634.8311899
3	1240.2525397	849.8282390	1164.4765153	335.2092780
4	516.5513756	741.2240952	672.4686122	1036.8223372
5	842.1370271	761.9846208	944.6874393	587.9625803
6	1224.3017044	757.4508354	1098.9940224	310.1460324
7	670.1436895	637.9618501	750.0718597	707.1903149
8	1196.3213363	617.0959220	1012.5757403	294.5833691
9	610.9822366	482.3049499	616.4531593	699.9918552
10	1152.0688881	357.6812618	878.1808353	283.3403230

Untuk melakukan uji validasi, maka dalam proses dibutuhkan minimal dua buah foto yang bertampalan dengan target yang sama, dimana proses perhitungannya terdapat dalam **Lampiran B** dan hasil standar deviasinya disajikan dalam tabel berikut:

Table 4.1 TSPS Monitoring Report

ID	Monitoring Month	Monitoring Year	Monitoring Month	Monitoring Year
1	2284402001	18230221103	2102825030	115310252070
2	2303082014	801282311023	0409032409	0348311840
3	124002523301	2402828300	2312003280	1104120213
4	2162213209	1413204013	0254082125	105082533323
5	2451320501	2010201028	0440824309	12830925209
6	1351301504	2231208321	100820400204	11011400034
7	0501130802	22510018201	20300318203	10231003130
8	1106301303	0130000020	1013225103	00412233901
9	01062823300	12530040408	0164231203	000000018325
10	112300082281	3230013018	828180833	0833403300

Perkiraan jumlah pengeluaran di kategori makanan daging beras gandum dan minuman yang diperlukan untuk memenuhi kebutuhan kalori setiap hari adalah sebagai berikut:

Kategori	Jumlah Pengeluaran (Rupiah)
Makanan	1000000000
Daging	200000000
Beras	150000000
Gandum	100000000
Minuman	50000000
Total	1300000000

Tabel 4.2 Tabel Koordinat foto dan standar deviasi dari proses *intersection*

ID	Koordinat Foto Kiri		Koordinat Foto Kanan		So (mm)
	x	y	x	y	
1	-4.98644	2.56208	-1.91644	5.85436	
2	-1.07506	2.68607	0.96879	-0.15445	
3	3.32488	2.46666	2.40116	-3.80725	
4	-5.4971	1.14262	-3.59646	4.74637	
5	-1.52819	1.39572	-0.27809	-0.72584	
6	3.13043	1.34045	1.60292	-4.1128	
7	-3.6248	-0.11628	-2.65047	0.72771	
8	2.78935	-0.37067	0.54948	-4.30253	
9	-4.34598	-2.01395	-4.27929	0.63995	
10	2.24991	-3.53328	-1.08881	-4.4396	
1.51554975					

Berdasarkan analisa nilai standar deviasi pada proses penentuan titik konjugasi dengan teknik *intersection* terhadap algoritma *image matching* ini, maka diperoleh nilai standart deviasi sebesar **1.51554975 mm**.

BAB V

KESIMPULAN

V.1 Kesimpulan

Berdasarkan analisa hasil penelitian dengan judul : “**Studi Penentuan Titik Konjugasi Pada Foto Yang Bertampalan (*Image Matching*) Menggunakan Metode Area-Based Matching Pada Foto Stereo**”, maka dapat ditarik kesimpulan mengenai penelitian ini yaitu:

1. *image matching* merupakan salah satu proses yang sangat penting dalam proses restitusi foto.
2. Metode *area-based matching* merupakan metode untuk menentukan keakurasaian titik konjugasi. Metode ini bekerja menggunakan teknik yang saling melengkapi dengan keunggulan masing-masing yaitu teknik NCC memiliki kecepatan komputasi, tetapi hanya menghasilkan ketelitian 1 piksel. Sebaliknya metode LSM, dengan menerapkan hitung kuadrat terkecil pada nilai keabuan, prosesnya relatif lebih lambat untuk konvergen namun mampu menghasilkan ketelitian hingga 0,01 piksel.
3. Evaluasi metode penentuan titik konjugasi dilakukan untuk menghasilkan koordinat piksel dengan keakurasaian sub piksel. Pada kasus ini, hasil dari *normalized cross correlation* (antara kedua foto) sudah sangat identik (hasil *matching*). Namun dibutuhkan *least squares matching* untuk menghasilkan ketelitian yang lebih akurat.

BAB A

KESIMPULAN

A.1. Kesiambungan

Berdasarkan makalah hasil penelitian yang bertajuk **"Analisis Persepsi Pengunjung Pada Foto dan Bentuk Pengaruh (Tinjauan Masyarakat) Melalui Analisis Klasifikasi Foto Stiker"**, maka dapat ditarik kesimpulan

menarik berikut ini yakni

1. Maka makalah meliputi sejarah dan bentuk bentangan desain

bentuk klasifikasi foto.

2. Metode wawancara wicara merupakan metode untuk menemukan
kesimpulan diri koulusasi. Metode ini perlu mendapatkan teknik yang
sangat teliti dan teknis dalam mewujudkan hasil riset NCC.
Metodologi penelitian kombinasi teknik penelitian kuantitatif dan kualitatif
seperti teknik survei dan teknik eksperimen pada klasifikasi foto.

Untuk penelitian klasifikasi foto dapat dilakukan dengan teknik klasifikasi

metrisasi klasifikasi foto dengan tingkat ketepatan yang tinggi.

3. Hasil klasifikasi makalah berdasarkan teknik koulusasi diketahui untuk mengepasikan
kondisi biskuit dengan peningkatan sup biskuit pada klasifikasi teknik klasifikasi
makanan cara convolvulus (sinisa kacang foto) sedangkan sup biskuit (pisti
mentimut). Makalah diperlukan agar dianalisis menggunakan teknik mengepasikan
klasifikasi agar lebih spesial.

4. Untuk kasus pada penelitian ini juga dilakukan uji validasi untuk beberapa titik dengan menggunakan teknik *intersection* sehingga diperoleh standar deviasi yaitu **1.51554975 mm.**

V.2 Saran

1. Perlunya diberikan ukuran *template* dan *search window* yang berbeda-beda pada metode *area based matching* untuk mengetahui tingkat keakurasaian apabila ukuran *window* diubah-ubah.
2. Untuk memperoleh ketelitian maksimal maka hendaknya bagi peneliti yang ingin melanjutkan penelitian ini, maka tambahkan koreksi radiometrik untuk membandingkan ketelitian yang diperoleh sekarang.

DAFTAR PUSTAKA

- Ackermann, F., 1984. "Digital Image Correlation: Performance and Potential Application in Photogrammetry". *Photogrammetric Record*, 11, pp.429-439.
- Agouris, P., and Schenk, T., 2000. "Multiple Image Matching". Department of Geodetic Science and Surveying. The Ohio University, Columbus, Ohio, USA.
- Bethel, J., 1997. "Least Squares Image Matching for CE604".
- Bethmann, F., and Luhmann T., 2010. "Least Squares Matching with Advanced Geometric Transformation Models". *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. XXXVIII, Part 5, Newcastle upon Tyne, UK. 2010.
- Bradski, G., and Kaehler, A., 2008. "Learning OpenCV". O'Reilly Media. USA.
- Campbell, N.A., and Wu, X., 2008. "Gradient Cross Correlation For Sub-Pixel Matching". *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*. Vol. XXXVII. Part B7. Beijing 2008
- Elaksher A.F., and Alharthy A.L., 2008. "Matching Conjugate Points Between Multi Resolution Satellite Images Using Geometric and Radiometric Properties". *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*. Vol. XXXVII. Part B3b. Beijing 2008.
- Elaksher, A., 2008. "A Multi-Photo Least Squares Matching Algorithm For Urban Area DEM Refinement Using Breaklines" *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*. Vol. XXXVII. Part B3a. Beijing 2008.
- Forstner, W., 1982. On the Geometric Precision of Digital Correlation. *International Archives of Photogrammetry and Remote Sensing, Symposium Helsinki Commission III*, 24-Part 3, 176-189.
- Grüen, A.W., 1985. "Adaptive Least Squares Correlation: A Powerful Image Matching Technique". *South African Journal of Photogrammetry, Remote Sensing and Cartography*, 14, pp.175-187.
- Grüen, A.W., 1996. "Least Squares Matching: A Fundamental Measurement Algorithm" in *Close Range Photogrammetry and Machine Vision*, K. B. Atkinson (ed.) Whittles Publishing, U.K., 217-255.

САМЯР ПІСТАКА

Із згадкою про пісню "Самяр пістака" відомо, що вона була написана відомим піаністом і композитором Валерієм Григоровичем Котом у 1981 році.

У пісні згадується місто Астана, яке в той час називалося Аксусом. У пісні згадується також місто Семипалатинськ, яке в той час називалося Омськом.

Пісня вийшла на компакт-диску "Легенди казахської музики" в 1991 році.

Відомо, що пісня була написана відомим піаністом і композитором Валерієм Григоровичем Котом у 1981 році. Пісня була написана відомим піаністом і композитором Валерієм Григоровичем Котом у 1981 році.

Пісня "Самяр пістака" вийшла на компакт-диску "Легенди казахської музики" в 1991 році.

Пісня "Самяр пістака" вийшла на компакт-диску "Легенди казахської музики" в 1991 році. Пісня "Самяр пістака" вийшла на компакт-диску "Легенди казахської музики" в 1991 році.

Пісня "Самяр пістака" вийшла на компакт-диску "Легенди казахської музики" в 1991 році. Пісня "Самяр пістака" вийшла на компакт-диску "Легенди казахської музики" в 1991 році.

Пісня "Самяр пістака" вийшла на компакт-диску "Легенди казахської музики" в 1991 році. Пісня "Самяр пістака" вийшла на компакт-диску "Легенди казахської музики" в 1991 році.

Пісня "Самяр пістака" вийшла на компакт-диску "Легенди казахської музики" в 1991 році. Пісня "Самяр пістака" вийшла на компакт-диску "Легенди казахської музики" в 1991 році.

Пісня "Самяр пістака" вийшла на компакт-диску "Легенди казахської музики" в 1991 році. Пісня "Самяр пістака" вийшла на компакт-диску "Легенди казахської музики" в 1991 році.

Пісня "Самяр пістака" вийшла на компакт-диску "Легенди казахської музики" в 1991 році. Пісня "Самяр пістака" вийшла на компакт-диску "Легенди казахської музики" в 1991 році.

- Hahn, M., 1997. "Automatic Control Point Measurement". Photogrammetric Week '97, Fritsch, D., Hobbie, D. (Eds.), pp. 115-126, Wichmann Verlag, Heidelberg.*
- Helava, U.V., 1988. "Object Space Least Squares Correlation". Photogrammetric Engineering & Remote Sensing, Vol. 54(6), pp. 711-714.*
- Jähne, B., 1997, "Practical Handbook on Image Processing for Scientific Applications". CRC Press LLC.*
- Kraus, K., 1997. "Photogrammetry", Vol.2., 4th edition, Fer. Dümmers Verlag, Bonn.*
- Leica Geosystem., 2006. "Stereo Analyst User's Guide". United States of America.*
- Lillesand, T.M, and Kiefer, R.W., 2000. "Remote Sensing and Image Interpretation". John Wiley and Sons Inc., New York, NY.*
- Lühmann, T., Robson, S., Kyle, S., and Harley, I., 2006. "Close Range Photogrammetry, Principles, Techniques and Applications". Whittes Publishing, Scotland, UK., 5100 pp.*
- Lühmann, T., 2000, "Nahbereichsphotogrammetrie: Grundlagen, Methoden und Anwendungen". Herbert Wichmann Verlag, Heidelberg. In Potuckova, 2004. Ph.D Thesis.*
- Makarovic, B., 1984. "Considerations On Image Matching-An Engineering Perspective". International Institute for Aerospace Survey and Earth Sciences (ITC). Netherlands.*
- Mallat S., 1999. "A Wavelet Tour a Signal Processing". 2nd edition. Academic Press.*
- Mikhail, E., Bethel, J., and McGlone, J., 2001. "Introduction to Modern Photogrammetry". Join Wiley & Sons Inc., New York.*
- Mitchell, H.L., and Pilgrim, L.J., 1987. "Selection of an Image Matching Algorithm". Department of Civil Engineering and Surveying. University of Newcastle.*
- Pertl, A., 1984. "Digital Image Correlation with the Analytical Plotter Planicomp C-100". International Archives of Photogrammetry and Remote Sensing 25 (3B), S. 874-882.*

1991 „A County Fair Tradition,” *Bibliotheca Hagiographica* 1991, 70
pp. 113-128; cf. also O. V. Kuznetsov, “The History of the

“Festivals” in the Life of the Orthodox Church in Russia, 1880-1917, *Ukrainian Ethnographic Collection*, 1991, 10, pp. 343-347.

See also A. V. Voznesensky, *Handbook on the Study of Slavic Religious Traditions*, (Minsk, 1991).

Yuri A. Voznesensky, “Orthodox Festivals in the Diocese of Simferopol,” *Russkaya Kniga* 1991, 5, pp. 14-17.

See also A. V. Voznesensky, “Orthodox Festivals in the Diocese of Simferopol,” *Russkaya Kniga* 1991, 5, pp. 14-17.

“Festivals in the Life of the Orthodox Church in Russia,” *Ukrainian Ethnographic Collection*, 1991, 10, pp. 343-347.

Yuri A. Voznesensky, “Orthodox Festivals in the Diocese of Simferopol,” *Russkaya Kniga* 1991, 5, pp. 14-17.

Yuri A. Voznesensky, “Orthodox Festivals in the Diocese of Simferopol,” *Russkaya Kniga* 1991, 5, pp. 14-17.

Yuri A. Voznesensky, “Orthodox Festivals in the Diocese of Simferopol,” *Russkaya Kniga* 1991, 5, pp. 14-17.

Yuri A. Voznesensky, “Orthodox Festivals in the Diocese of Simferopol,” *Russkaya Kniga* 1991, 5, pp. 14-17.

Yuri A. Voznesensky, “Orthodox Festivals in the Diocese of Simferopol,” *Russkaya Kniga* 1991, 5, pp. 14-17.

Yuri A. Voznesensky, “Orthodox Festivals in the Diocese of Simferopol,” *Russkaya Kniga* 1991, 5, pp. 14-17.

Yuri A. Voznesensky, “Orthodox Festivals in the Diocese of Simferopol,” *Russkaya Kniga* 1991, 5, pp. 14-17.

- Potuckova, M., 2004. "Image Matching and Its Application in Photogrammetry". Department of Development Planning Aalborg University, Aalborg, Denmark.*
- Rosenholm, D., 1987a. "Empirical Investigation of Optimal Window Size Using The Least Squares Image Matching Method" The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Scien, Vol 42 (1987) pp. 133-125.*
- Rosenholm, D., 1987b. "Least Squares Matching Method: Same Experimental Results". Photogrammetric Record, 12, pp.493-512.*
- Thévenaz P., Blu T., and Unser M., 1999. "Image Interpolation and Resampling". Swiss Federal Institute of Technology-Lausanne.*
- Tsai, D.M., and Lin, C.T., 2002. " Fast Normalized Cross Correlation for Defect Detection". Department of Industrial Engineering and Management Yuan-Ze University, Chung-Li, Taiwan, R.O.C.*
- Sachs, J., 2001. "Image Resampling". Digital Light & Color.*
- Sarjakoski, T., and Lammi, J., 1996. " Least Squares Matching By Search". The International Archives of the Photogrammetry, Remote Sensing. Vol. XXXI Part B3. Vienna 1996.*
- Schenk, T., 1999. "Digital Photogrammetry", TerraScience, Ohio, USA.*
- Wolf, P.R., and Dewitt, B. A., 2000. " Element of Photogrammetry with Application in GIS 3rd", McGraw-Hill Higher Education. pp 334-341.*
- Zhaltov, S., Y. and Sibiryakov, A., V., 1997. "Adaptive Subpixel Cross-correlation in a Point Correspondence Problem". State Research Institute of Aviation Systems, Moscow, Russia.*

"gyakorlók" néven ismerjük, azt hogy gyakorlók legyenek". 1905. "A gyakorlók általánosan gyakorlók, gyakorlók" gyakorlók gyakorlók magyarországi tanulmányokról

Proprietary or otherwise Sensitive Information - Do Not Distribute
FBI File No. 183-3801

Rescue," *Biology and Biodiversity Research* 15, pp. 204-213.

"*gulipanavas* bin *nottisognis* bin *ugumt*" 8921 „M 1921“ bin „A mid „M 1921“
„*gulipanavas* bin *lakay-panawas*“ 8922 „M 1921“ bin „A mid „M 1921“

Designation." Declarative relationships may also be used to define the relationship between D.W. and Bill (A, 3005). "Four Monoliths" (2005) provides an example for D.W. as

2003) & nsgd hsgd (L. "gukunzhaixing". 2002. L. gukunzhaixing).

1991-1992
Yearly average rainfall
in mm

„nemről, amit gyakran megosztott a nemről” (1990). A B. Művelődési és „A.S. Móra” Művészeti Központban 1991. március 22-én nyílt kiállításán „A.S. Móra kiállítás” címmel.

*Sister, your Runia
a point (correspondence proposed) some features of which in
Xborow, 17.2.1915." Applying myself to a consideration of*

LAMPIRAN

NORMALIZED CROSS CORRELATION

```
namespace StereoMap
{
    //Obsolete...
    public class CrossCorrelation
    {
        #region Fields
        Image<Gray, Byte> _template;
        double _sumGrayValTemplate;
        double _sumSquaredValTemplate;
        Point _maxLoc;           //approximate value: input for
                                //photogrammetric crosscorrelation --> on
                                //the matching wnd in pxl
        Point _matchedPos;      //result from ph. cross corr
        double _corrCoeficient;

        //Least Square Matching
        LSM _lsm;   //Photogrammetric Least Square Image Matching
        #endregion

        #region Constructor
        public CrossCorrelation(ref Image<Gray, Byte> source, ref
        Image<Gray, Byte> matchingImg, PointF centerTemplate)
        {
            Trace.WriteLine("CrossCorrelation constructor is
called");
            //construct template window
            Rectangle rectTemplate = new
            Rectangle((int)((centerTemplate.X + 0.5) - 10),
            (int)((centerTemplate.Y + 0.5) - 10), 21, 21);
            Trace.WriteLine("rect T L R B: " +
            rectTemplate.Top.ToString() + " " +
            rectTemplate.Left.ToString() + " " +
            rectTemplate.Right.ToString() + " " +
            rectTemplate.Bottom.ToString());
            Trace.WriteLine("rect Width Height: " +
            rectTemplate.Width.ToString() + " " +
            rectTemplate.Height.ToString());
            Trace.WriteLine("rect X Y: " + rectTemplate.X.ToString()
            + " " + rectTemplate.Y.ToString());

            Point centreTempl = new Point((int)(centerTemplate.X +
            0.5), (int)(centerTemplate.Y + 0.5));
            Trace.WriteLine("centreTempl: " +
            centreTempl.ToString());
            double centreX = (double)rectTemplate.X +
            ((double)rectTemplate.Width) / 2.0;
            double centreY = (double)rectTemplate.Y +
            ((double)rectTemplate.Height) / 2.0;
        }
    }
}
```

```

Trace.WriteLine("centreX centreY: " + centreX.ToString()
+ " " + centreY.ToString());
_template = source.Copy(rectTemplate);
_template.ROI.Offset(rectTemplate.X, rectTemplate.Y);

//Trace.WriteLine("_template ROI Upper L T: " +
_template.ROI.X.ToString() + " " +
_template.ROI.Y.ToString());
//Trace.WriteLine("_template ROI T L R B: " +
_template.ROI.Top.ToString() + " " +
_template.ROI.Left.ToString() + " " +
//                _template.ROI.Right.ToString() + "
" + _template.ROI.Bottom.ToString());
//Trace.WriteLine("_template ROI width height: " +
_template.ROI.Width.ToString() + " " +
_template.ROI.Height.ToString());

_sumGrayValTemplate = _template.GetSum().Intensity;
_sumSquaredValTemplate = SetSumSquare(ref _template);

//CrossCorrelation using Emgu
Image<Gray, float> temp =
    matchingImg.MatchTemplate(_template,
    TM_TYPE.CV_TM_CCOEFF_NORMED);
double minVal = 0.0;
double maxVal = 0.0;
Point minLoc = new Point();
Point maxLoc = new Point();
CvInvoke.cvMinMaxLoc(temp, ref minVal, ref maxVal, ref
    minLoc, ref maxLoc, IntPtr.Zero);
//_maxLoc = maxLoc;
_maxLoc.X = maxLoc.X + 10;
_maxLoc.Y = maxLoc.Y + 10;
//Trace.WriteLine("Min Max Value: " + minVal.ToString()
+ " " + maxVal.ToString());
//Trace.WriteLine("Min Max Loc: " + minLoc.ToString() +
" " + maxLoc.ToString());
_template.Save(@"D:\Skripsi Via\Program Saya\Program
Rapid Mapping-Mine02062010\Image Matching>Edit
disini2\StereoMapping\StereoMap\bin\Debug\Template.jpg
");

//Photogrammetric CrossCorrelation
double CorrCoef = 0.0;
double alpha = 0.0;
double beta = 0.0;
Point matchedPoint = PhNormCrossCorrelation(ref
    matchingImg, out CorrCoef, out alpha, out beta);
Point2D conjugatePoint;
bool success;
if (CorrCoef > 0.695 && matchedPoint.X != 0 &&
    matchedPoint.Y != 0)
{
    _corrCoeficient = CorrCoef;
}

```

```

        _matchedPos = matchedPoint;
    //construct Matching Patch window
    Rectangle rectMatchingPatch = new
        Rectangle((int)((_matchedPos.X + 0.5) - 10),
                  (int)((_matchedPos.Y + 0.5) - 10), 21, 21);
    Trace.WriteLine("Matched Point from EMGU: " +
        _maxLoc.ToString());
    Trace.WriteLine("Matched Point from Ph Cross Corr: " +
        matchedPoint.ToString() + " with corr coef: " +
        CorrCoef.ToString());
    Trace.WriteLine("ho h1: " + alpha.ToString() + " " +
        beta.ToString());
    //Trigger LSM
    _lsm = new LSM(ref rectTemplate, ref centreTempl,
                   ref _matchedPos, ref source,
                   ref matchingImg, alpha, beta, rectMatchingPatch);
    conjugatePoint = _lsm.FindConjugatePoint(out
                                              success);
}
#endregion

#region Properties
//A location where a maximum coeff value occurs
public Point MaxLocation
{
    get { return _maxLoc; }
}

public Point MatchedPointCorr
{
    get { return _matchedPos; }
}

public double CorrelationCoeficient
{
    get { return _corrCoeficient; }
}
#endregion

#region Methods
private Point PhNormCrossCorrelation(ref Image<Gray, Byte>
    matchingImg, out double corrCoef,
    out double alpha, out double beta)
{
    //Trace.WriteLine("template average value: " +
    //    template.GetAverage().ToString());
    //Trace.WriteLine("template sum value: " +
    //    template.GetSum().ToString());
    //Trace.WriteLine("template row col : " +
    //    template.Rows.ToString() + " " +
    //    template.Cols.ToString());

    //create a searching window which size of 15x15
}

```

```

        Point matchedPt = new Point();
        double maxCorrVal = 0.7;
        double alph = 0.0;
        double bet = 0.0;

        for (int row = _maxLoc.Y - 15; row <= _maxLoc.Y + 15;
            row++)
        {
            for (int col = _maxLoc.X - 15; col <= _maxLoc.X +
                15; col++)
            {
                Image<Gray, Byte> matchingPatch =
                    SetMatchingWindow(ref matchingImg, col, row);
                    //create matching patch

                //PrintPatch(ref matchingPatch);
                double temp = GetCorrelationCoef(ref
                    matchingPatch, out alph, out bet);
                if (temp > maxCorrVal)
                {
                    matchedPt.X = col;
                    matchedPt.Y = row;
                    maxCorrVal = temp;
                }
            }
        }
        corrCoef = maxCorrVal;
        alpha = alph;
        beta = bet;
        //Rectangle rect = new Rectangle((int)((_maxLoc.X + 0.5)
        //    - 11), (int)((_maxLoc.Y + 0.5) - 11), 21, 21);
        //Image<Gray, Byte> matchingWnd =
        //    matchingImg.Copy(rect);
        //matchingWnd.Save //matchingWnd.Save(@"D:\Skripsi
        //Via\Program Saya\Program Rapid Mapping-
        Mine02062010\Image Matching>Edit
        disini2\StereoMapping\StereoMap\bin\Debug\Search
        Window.jpg");
        Trace.WriteLine("Max Corr Coef: " +
            maxCorrVal.ToString());
        return matchedPt;
    }

    private double GetCorrelationCoef(ref Image<Gray, Byte>
        matching, out double alpha, out double beta)
    {
        //construct matching patch
        Debug.Assert(matching.Cols == _template.Cols &&
            matching.Rows == _template.Rows);
        int elementNumbers = matching.Rows * matching.Cols;
        //double avgGrayValMatchPatch =
        //    matching.GetAverage().Intensity;
        double sumGrayValMatchPatch =
            matching.GetSum().Intensity;

```

```

        double sumSquaredGrayValMatchPatch = SetSumSquare(ref
            matching);

        double SumMult = SumMultiply(ref matching);
        //Trace.WriteLine("SumMult: " + SumMult.ToString());

        //From Element of Photogrammetry, Wolf page 336:
        //Sx=template, Sy=matchingPatch
        double SxSquared = _sumSquaredValTemplate -
            (_sumGrayValTemplate * _sumGrayValTemplate) /
            elementNumbers;
        double SySquared = sumSquaredGrayValMatchPatch -
            (sumGrayValMatchPatch * sumGrayValMatchPatch) /
            elementNumbers;
        double Sxy = SumMult - (_sumGrayValTemplate *
            sumGrayValMatchPatch) / elementNumbers;

        beta = Sxy / SxSquared;
        alpha = (sumGrayValMatchPatch / elementNumbers) - (beta
            * (_sumGrayValTemplate / elementNumbers));
        double CorrCoef = Sxy / Math.Sqrt(SxSquared *
            SySquared);
        return CorrCoef;
    }

    private double SumMultiply(ref Image<Gray, Byte> img)
    {
        double sum = 0.0;
        Debug.Assert(img.Cols == _template.Cols && img.Rows ==
            img.Rows);
        for (int y = 0; y < img.Rows; y++)
        {
            for (int x = 0; x < img.Cols; x++)
            {
                sum += _template[y, x].Intensity * img[y,
                    x].Intensity;
            }
        }
        return sum;
    }

    private double SetSumSquare(ref Image<Gray, Byte> img)
    {
        int row = img.Rows;
        int col = img.Cols;
        double sumSquared = 0;
        for (int y = 0; y < row; y++)
        {
            for (int x = 0; x < col; x++)
            {
                Emgu.CV.Structure.Gray g = img[y, x];
                sumSquared += (g.Intensity * g.Intensity);
            }
        }
    }
}

```

```

        }

        return sumSquared;
    }

private Image<Gray, Byte> SetMatchingWindow(ref Image<Gray,
    Byte> matchingImg, int x, int y)
{
    Rectangle rect = new Rectangle(x - 10, y - 10, 21, 21);
    Image<Gray, Byte> matchingPatch =
        matchingImg.Copy(rect);
    return matchingPatch;
}

private void PrintPatch(ref Image<Gray, Byte> img)
{
    Trace.WriteLine("Patch gray value:");
    int row = img.Rows;
    int col = img.Cols;
    for (int y = 0; y < row; y++)
    {
        for (int x = 0; x < col; x++)
        {
            Trace.Write(img[y, x].Intensity.ToString() + " ");
        }
        Trace.WriteLine("n");
    }
}
#endregion

}
}

```

LAMPIRAN

LEAST SQUARE MATCHING

6 Parameter

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Diagnostics;
using System.IO;
using System.Drawing;

using Emgu.CV;
using Emgu.Util;
using Emgu.CV.UI;
using Emgu.CV.Structure;
using Emgu.CV.CvEnum;

namespace StereoMap
{
    /// <summary>
    /// LSM Implementation according to Toni Schenk
    /// Using 6 affine Parameters: shift: dx dy; Scale: x y. Parameters:
    /// a0,a1,a2; b0, b1,b2
    /// x = a0 + a1 * i + a2 * j ; i = col in pixel
    /// y = b0 + b1 * i + b2 * j ; j = row in pixel
    /// Steps:
    /// 1. Select the centre of one template
    /// 2. Determine approximate location for the matching patch
    /// 3. Start first iteration with matching patch at the approximate
        location
    /// 4. Transform matching patch and determine the gray value for the
        tessellation (resampling)
    /// 5. Repeat Adjustment and resampling sequence until it is converged
    /// </summary>
    public class LSM6A
    {
        #region Fields
        /// <summary>
        /// Template Image
        /// </summary>
        Image<Gray, Byte> _templateImg;
        public CSML.Matrix A; // Koefisien Matrix
        public CSML.Matrix _b; // Weight Matrix
        public double r; // Degree of Freedom

        /// <summary>
        /// Matching Image
        /// </summary>
        Image<Gray, Byte> _matchingImg;
```

IMPLEMENTATION

LEAST SQUARE MATCHING

4. Feature matching

```
    using System;
    using System.Collections.Generic;
    using System.Text;
    using System.Diagnostics;
    using System.IO;
    using System.Drawing;

    class Program
    {
        static void Main(string[] args)
        {
            // Read images
            string templateImage = "templateImage.jpg";
            string targetImage = "targetImage.jpg";

            // Load images
            Image template = Image.FromFile(templateImage);
            Image target = Image.FromFile(targetImage);

            // Convert images to grayscale
            Bitmap templateGrayscale = new Bitmap(template.Width, template.Height);
            Bitmap targetGrayscale = new Bitmap(target.Width, target.Height);

            for (int i = 0; i < templateGrayscale.Width; i++)
            {
                for (int j = 0; j < templateGrayscale.Height; j++)
                {
                    templateGrayscale.SetPixel(i, j, template.GetPixel(i, j).ToGray());
                    targetGrayscale.SetPixel(i, j, target.GetPixel(i, j).ToGray());
                }
            }

            // Compute SIFT features
            SiftFeatureDetector detector = new SiftFeatureDetector();
            SiftDescriptorExtractor extractor = new SiftDescriptorExtractor();

            var templateFeatures = detector.Detect(templateGrayscale);
            var targetFeatures = detector.Detect(targetGrayscale);

            var templateDescriptors = extractor.Extract(templateGrayscale, templateFeatures);
            var targetDescriptors = extractor.Extract(targetGrayscale, targetFeatures);

            // Match features
            var matcher = new BruteForceMatcher();
            var matches = matcher.Calculate(templateDescriptors, targetDescriptors);

            // Draw matches
            using (Graphics g = Graphics.FromImage(new Bitmap(template.Width, target.Height)))
            {
                foreach (var match in matches)
                {
                    g.DrawLine(Pens.Blue, templateFeatures[match.QueryIndex].Location, targetFeatures[match.TargetIndex].Location);
                }
            }

            // Save result
            target.Save("result.jpg");
        }
    }
}
```

The code above implements a least squares matching algorithm for image registration. It starts by reading two images, "templateImage.jpg" and "targetImage.jpg". It then converts both images to grayscale. Next, it detects SIFT features and extracts descriptors from each image. Finally, it uses a brute force matcher to find matches between the two sets of descriptors. The resulting matches are drawn as blue lines connecting corresponding feature points in the two images. The final result is saved as "result.jpg".

```

/// <summary>
/// Coordinates of the centre of a selected template patch, in pixel
/// </summary>
Point2D _centreTemplate;

/// <summary>
/// Coordinate of the centre of the matching patch, in pixel. It is
/// resulted from CrossCorrelation
/// </summary>
Point2D _matchedPoint;           //in pixel

/// <summary>
/// Coordinate of the final matching patch,in pixel. It is from the LSM
/// </summary>
Point2D _correctedMatchedPt;    //final result

/// <summary>
/// Affine Parameters: Shift x
/// </summary>
double _a0;

/// <summary>
/// Affine Parameters: Shift y
/// </summary>
double _b0;

/// <summary>
/// Affine Parameters: Scale x
/// </summary>
double _a1;

/// <summary>
/// Affine Parameters: Scale y
/// </summary>
double _b2;

/// <summary>
/// Affine Parameters: Shear y
/// </summary>
double _a2;

/// <summary>
/// Affine Parameters: Shear x
/// </summary>
double _b1;

double _XRightPhoto;//Tambahan
double _YRightPhoto;
double _XLeftPhoto;
double _YLeftPhoto;

/// <summary>
/// constant: gray value different T(i,j) - m(i,j)
/// </summary>

```



```

//CSML.Matrix _b;

/// <summary>
/// Indicator whether the LSM iteration can converge
/// </summary>
bool _lsmConverge;

/// <summary>
/// Transformed grid.
/// m[row, 1] = grid for x direction (col) on Matching Img
/// m[row, 2] = grid for y direction (row) on Matching Img
/// m[row, 3] = grid for x direction (col) on Template Img
/// m[row, 4] = grid for y direction (row) on Template Img
/// </summary>
CSML.Matrix _gridsXY;

Point2D _previousMatched;
#endregion Fields

#region Constructors
public LSM6A()
{
    _centreTemplate = new Point2D();
    _matchedPoint = new Point2D();
    _correctedMatchedPt = new Point2D();

    _a0 = 0.0;
    _a1 = 0.0;
    _a2 = 0.0;
    _b0 = 0.0;
    _b1 = 0.0;
    _b2 = 0.0;
    _lsmConverge = false;

    _gridsXY = new CSML.Matrix(19, 4);
    _previousMatched = new Point2D();
}

public LSM6A(ref Image<Gray, Byte> source, ref Image<Gray, Byte>
    matchingImg, Point2D centerTemplate, Point centreMatchingPatch)
    : this()
{
    Trace.WriteLine("\t\t----- Constructor LSM6A() is
called -----");
    _templateImg = source;
    _matchingImg = matchingImg;
    _centreTemplate.X = (double)((int)(centerTemplate.X + 0.5));
    _centreTemplate.Y = (double)((int)(centerTemplate.Y + 0.5));
    _matchedPoint.X = (double)centreMatchingPatch.X;
    _matchedPoint.Y = (double)centreMatchingPatch.Y;
    _correctedMatchedPt.X = (double)centreMatchingPatch.X;
    _correctedMatchedPt.Y = (double)centreMatchingPatch.Y;

    _previousMatched.X = (double)centreMatchingPatch.X;
    _previousMatched.Y = (double)centreMatchingPatch.Y;
}

```

```

        SetInitialGridsXY();
    }
#endregion Constructors

#region Properties

#endregion Properties

#region Private Methods
private Point2D Affine6Params(double x, double y)
{
    Point2D pt = new Point2D();
    pt.X = _a0 + _a1 * x + _a2 * y;
    pt.Y = _b0 + _b1 * x + _b2 * y;
    return pt;
}

private void SetInitialGridsXY()
{
    //Matching Patch 17x17 ==> add two more to account for border -->
    //19x19
    int rowStartMatchingPatch = (int)_matchedPoint.Y - 9;
    int rowEndMatchingPatch = (int)_matchedPoint.Y + 9;
    int colStartMatchingPatch = (int)_matchedPoint.X - 9;
    int colEndMatchingPatch = (int)_matchedPoint.X + 9;

    //Template Patch 17x17 ==> add two more to account for border -->
    //19x19
    int rowStartTemplatePatch = (int)_centreTemplate.Y - 9;
    int rowEndTemplatePatch = (int)_centreTemplate.Y + 9;
    int colStartTemplatePatch = (int)_centreTemplate.X - 9;
    int colEndTemplatePatch = (int)_centreTemplate.X + 9;

    //int rowNumbers = 17;//Tambahan
    //A = new CSML.Matrix(rowNumbers, 6);//Tambahan
    //b = new CSML.Matrix(rowNumbers, 1);//Tambahan
    int index = 1;
    for (int counter = 0; counter < 19; counter++)
    {
        int xMPatch = counter + colStartMatchingPatch;
        int yMPatch = counter + rowStartMatchingPatch;
        int xTPatch = counter + colStartTemplatePatch;
        int yTPatch = counter + rowStartTemplatePatch;

        _gridsXY[index, 1].Re = (double)xMPatch;
        _gridsXY[index, 2].Re = (double)yMPatch;
        _gridsXY[index, 3].Re = (double)xTPatch;
        _gridsXY[index, 4].Re = (double)yTPatch;
        index++;
    }
    //Trace.WriteLine("_gridsXY: \n" + _gridsXY.ToString());
}
}

private void RearrangeGridsXY(CSML.Matrix solution)

```



```

{
    int rows = _gridsXY.RowCount;
    //Trace.WriteLine("row counts: " + row.ToString());
    for (int index = 1; index <= rows; index++)
    {
        _gridsXY[index, 1].Re += solution[1, 1].Re +           //shift x
            solution[2, 1].Re * _gridsXY[index, 1].Re +       //scale x
            solution[3, 1].Re * _gridsXY[index, 2].Re;         //shear x
        _gridsXY[index, 2].Re += solution[4, 1].Re +           //shift y
            solution[5, 1].Re * _gridsXY[index, 1].Re +       //scale y
            solution[6, 1].Re * _gridsXY[index, 2].Re;         //shear y
    }
    //Trace.WriteLine("Rearrange GridXY: \n" + _gridsXY.ToString());
}

private void ComposeA_b6Params1stIteration(ref CSML.Matrix a, ref
CSML.Matrix b)
{
    int rowNumbers = 17 * 17;
    //CSML.Matrix A = new CSML.Matrix(rowNumbers, 6);
    A = new CSML.Matrix(rowNumbers, 6);
    _b = new CSML.Matrix(rowNumbers, 1);
    int index = 1;
    int row_col = _gridsXY.RowCount;
    Trace.WriteLine("row_col: " + row_col.ToString());

    for (int rowIndex = 2; rowIndex <= row_col - 1; rowIndex++)
    {
        for (int colIndex = 2; colIndex <= row_col - 1; colIndex++)
        {
            //Resample matching patch onto template image == original
            //matching patch
            //Compute Gradient of A on the matching patch
            //m0(x,y)=m(i,j)
            double x1 = _gridsXY[colIndex - 1, 1].Re;      //x-1
            double x = _gridsXY[colIndex, 1].Re;             //x
            double x2 = _gridsXY[colIndex + 1, 1].Re;        //x+1

            double y1 = _gridsXY[rowIndex - 1, 2].Re;      //y-1
            double y = _gridsXY[rowIndex, 2].Re;             //y
            double y2 = _gridsXY[rowIndex + 1, 2].Re;        //y+1

            double dx1 = _matchingImg[(int)y, (int)x1].Intensity;
            //g(x-1)
            double dx2 = _matchingImg[(int)y, (int)x2].Intensity;
            //g(x+1)
            double dx = (dx2 - dx1) / 2.0;           //(g(x+1)-g(x-1))/2

            double dy1 = _matchingImg[(int)y1, (int)x].Intensity;
            //g(y-1)
            double dy2 = _matchingImg[(int)y2, (int)x].Intensity;
            //g(y+1)
            double dy = (dy2 - dy1) / 2.0;           //(g(y+1)-g(y-1))/2
        }
    }
}

```



```

        double dn = _matchingImg[(int)y, (int)x].Intensity;
        //g(x,y)

        //Compute A
        A[index, 1].Re = dx;           //a0: dx
        A[index, 2].Re = dx * x;      //a1: dx * x
        A[index, 3].Re = dx * y;      //a2: dx * y
        A[index, 4].Re = dy;           //b0: dy
        A[index, 5].Re = dy * x;      //b1: dy * x
        A[index, 6].Re = dy * y;      //b2: dy * y

        //Compute b
        double xT = _gridsXY[colIndex, 3].Re;    //x on template
        double yT = _gridsXY[rowIndex, 4].Re;      //y on template
        double dnT = _templateImg[(int)yT, (int)xT].Intensity;
        //t(x,y)
        _b[index, 1].Re = dnT - dn;

        index++;
    }
}
//Trace.WriteLine("A: \n" + A.ToString()); //tambahan
//Trace.WriteLine("B: \n" + _b.ToString()); //tambahan==> cek

CSML.Matrix AT = A.Transpose();
a = AT * A;
b = AT * _b;
}

private void ComposeA_b6Params(ref CSML.Matrix a, ref CSML.Matrix b)
{
    //looping: 19
    int rowNumbers = 17 * 17;
    //CSML.Matrix A = new CSML.Matrix(rowNumbers, 6);
    A = new CSML.Matrix(rowNumbers, 6);
    int index = 1;
    int row_col = _gridsXY.RowCount;
    Trace.WriteLine("row_col: " + row_col.ToString());

    for (int rowIndex = 2; rowIndex <= row_col - 1; rowIndex++)
    {
        for (int colIndex = 2; colIndex <= row_col - 1; colIndex++)
        {
            //Resample matching patch onto template image == original
            //matching patch
            //Compute Gradient of A on the matching patch
            //m(x,y) != m(i,j) ==> for matching image
            double x1 = _gridsXY[colIndex - 1, 1].Re;    //x-1
            double x = _gridsXY[colIndex, 1].Re;          //x
            double x2 = _gridsXY[colIndex + 1, 1].Re;     //x+1

            double y1 = _gridsXY[rowIndex - 1, 2].Re;    //y-1
            double y = _gridsXY[rowIndex, 2].Re;          //y
            double y2 = _gridsXY[rowIndex + 1, 2].Re;     //y+1

```



```

//Bilinear Interpolation
int dx1 = BilinearInterpolateMatching(x1, y); //g(x-1)
int dx2 = BilinearInterpolateMatching(x2, y); //g(x+1)
int dgx = (dx2 - dx1) / 2; //x gradient

int dn = BilinearInterpolateMatching(x, y); //g(x,y)
int dy1 = BilinearInterpolateMatching(x, y - 1); //g(y-1)
int dy2 = BilinearInterpolateMatching(x, y + 1); //g(y+1)
int dgy = (dy2 - dy1) / 2; //y gradient

//Compose A
A[index, 1].Re = dgx; //a0: dx
A[index, 2].Re = dgx * x; //a1: dx * x
A[index, 3].Re = dgx * y; //a2: dx * y
A[index, 4].Re = dgy; //b0: dy
A[index, 5].Re = dgy * x; //b1: dy * x
A[index, 6].Re = dgy * y; //b2: dy * y

//Compute b
//==> for template image
double xT = _gridsXY[colIndex, 3].Re; //x on template
double yT = _gridsXY[rowIndex, 4].Re; //y on template
double dnT = _templateImg[(int)yT, (int)xT].Intensity;
//t(x,y)
_b[index, 1].Re = dnT - dn;
index++;
}
}

//Trace.WriteLine("A: \n" + A.ToString()); //tambahan//
//Trace.WriteLine("B: \n" + _b.ToString()); //tambahan==> cek

CSML.Matrix AT = A.Transpose();
a = AT * A;
b = AT * _b;
}

/// <summary>
/// Bilinear interpolation on Matching Image
/// </summary>
/// <param name="x">on Matching Image Coord</param>
/// <param name="y">on Matching Image Coord</param>
/// <returns>Gray Value from the Matching Image</returns>
private int BilinearInterpolateMatching(double x, double y)
{
    int X = (int)x;
    int Y = (int)y;
    double dx = x - (double)X;
    double dy = y - (double)Y;
    double g0 = _matchingImg[Y, X].Intensity; //go(x,y) --> term 0
    double g1 = _matchingImg[Y, X + 1].Intensity; //go(x+1,y)
    double g2 = _matchingImg[Y + 1, X].Intensity; //go(x,y+1)
    double g3 = _matchingImg[Y + 1, X + 1].Intensity; //go(x+1,y+1)
    double term1 = (g1 - g0) * dx; //term 1
    double term2 = (g2 - g0) * dy; //term 2
}

```



```

        double term3 = (g3 + g0 - g1 - g2) * dx * dy; //term 3
        int val = (int)(g0 + term1 + term2 + term3 + 0.5);
        return val;
    }

#endregion Private Methods

#region Public Methods
public bool FindLSMConjugatePoint()
{
    CSML.Matrix aTa = new CSML.Matrix();
    CSML.Matrix aTb = new CSML.Matrix();
    CSML.Matrix solution = new CSML.Matrix();
    CSML.Matrix AX = new CSML.Matrix(); //Tambahan
    double rCol; //Tambahan
    double rRow; //Tambahan

    ComposeA_b6Params1stIteration(ref aTa, ref aTb);
    bool aTaCond = aTa.IsSymmetricPositiveDefinite();
    double aTaDet = aTa.Determinant().Re;
    Trace.WriteLine("Det aTa: \t" + aTaDet.ToString());
    Trace.WriteLine("Is aTa symmetric positif definite: \t"
        + aTaCond.ToString());
    //Trace.WriteLine("aTa: \n" + aTa.ToString());
    //Trace.WriteLine("ATb : \n" + aTb.ToString());
    //Trace.WriteLine("aTa Inverse: \n" + aTa.Inverse().ToString());

    int iteration = 2;

    int Wdth = _matchingImg.Width; //Tambahan
    int Hght = _matchingImg.Height; //Tambahan

    int Wdh = _templateImg.Width; //Tambahan
    int Hgt = _templateImg.Height; //Tambahan

    const double DELTA = 0.01; //pixel
    if (aTaCond && aTaDet > 0.0)
    {
        solution = CSML.Matrix.Solve(aTa, aTb);
        RearrangeGridsXY(solution);
        _correctedMatchedPt.X += solution[1, 1].Re +
            solution[2, 1].Re * _correctedMatchedPt.X + solution[3,
                1].Re * _correctedMatchedPt.Y;
        _correctedMatchedPt.Y += solution[4, 1].Re +
            solution[5, 1].Re * _correctedMatchedPt.X + solution[6,
                1].Re * _correctedMatchedPt.Y;
        _lsmConverge = Threshold(solution, DELTA);
        //_lsmConverge = Threshold2();
        //if (!_lsmConverge)
        //{
        //    _previousMatched.X = _correctedMatchedPt.X;
        //    _previousMatched.Y = _correctedMatchedPt.Y;
        //}
    }
}

```



```

_XRightPhoto = (_correctedMatchedPt.X - ((Wdth / 2) - 0.5))
*0.01219; //Tambahan
_YRightPhoto = (((Hght / 2) - 0.5) - _correctedMatchedPt.Y) *
0.01219; //Tambahan

_XLeftPhoto = (_centreTemplate.X - ((Wdh / 2) - 0.5)) *
0.01219; //Tambahan
_YLeftPhoto = (((Hgt / 2) - 0.5) - _centreTemplate.Y) *
0.01219; //Tambahan

Trace.WriteLine("Iteration 1: " + "\n" + solution.ToString());
Trace.WriteLine("_correctedMatchedPt X Y: " +
    _correctedMatchedPt.X.ToString() + " " +
    _correctedMatchedPt.Y.ToString());

//Trace.WriteLine("Left Photo Coordinate XPhoto YPhoto: " +
    _XLeftPhoto.ToString() + " " +
    _YLeftPhoto.ToString());//tambahan
//Trace.WriteLine("Right Photo Coordinate XPhoto YPhoto: " +
    _XRightPhoto.ToString() + " " +
    _YRightPhoto.ToString());//tambahan

AX = A * solution;
CSML.Matrix AxB = new CSML.Matrix();
AxB = AX - _b;
CSML.Matrix VT = new CSML.Matrix();
VT = AxB.Transpose();
CSML.Matrix VtV = new CSML.Matrix();
VtV = VT * AxB;
rCol = A.ColumnCount;
rRow = A.RowCount;
r = rRow - rCol;
//double tes = VtV[1].Re;
double VtVCo = VtV[1].Re;

//VtVCo = VtV[0];
// double VtVDo = VtVCo.Re;
double So;
So = Math.Sqrt(VtVCo) / (r);

//Trace.WriteLine("AX: " + "\n" + AX.ToString());
//Trace.WriteLine("V: " + "\n" + AxB.ToString());
//Trace.WriteLine("VT: " + "\n" + VT.ToString());
//Trace.WriteLine("r: " + "\n" + r.ToString());
Trace.WriteLine("So: " + "\n" + So.ToString());

while (_lsmConverge && iteration <= 75)
{
    ComposeA_b6Params(ref aTa, ref aTb);
    solution = CSML.Matrix.Solve(aTa, aTb);
    RearrangeGridsXY(solution);
    _correctedMatchedPt.X += solution[1, 1].Re +
        solution[2, 1].Re * _correctedMatchedPt.X +
        solution[3, 1].Re * _correctedMatchedPt.Y;
}

```



```

    _correctedMatchedPt.Y += solution[4, 1].Re +
        solution[5, 1].Re * _correctedMatchedPt.X +
        solution[6, 1].Re * _correctedMatchedPt.Y;
    _lsmConverge = Threshold(solution, DELTA);
//_lsmConverge = Threshold2();
//if (!_lsmConverge)
//{
//    _previousMatched.X = _correctedMatchedPt.X;
//    _previousMatched.Y = _correctedMatchedPt.Y;
//}

_XRightPhoto = (_correctedMatchedPt.X - ((Wdth / 2) - 0.5)) *
    * 0.01219; //Tambahan
_YRightPhoto = (((Hght / 2) - 0.5) - _correctedMatchedPt.Y) *
    * 0.01219; //Tambahan

_XLeftPhoto = (_centreTemplate.X - ((Wdh / 2) - 0.5)) *
    * 0.01219; //Tambahan
_YLeftPhoto = (((Hgt / 2) - 0.5) - _centreTemplate.Y) *
    * 0.01219; //Tambahan

Trace.WriteLine("Iteration: " + iteration.ToString() + "\n"
    + solution.ToString());
Trace.WriteLine("_correctedMatchedPt X Y: " +
    _correctedMatchedPt.X.ToString() + " " +
    _correctedMatchedPt.Y.ToString());

//Trace.WriteLine("Left Photo Coordinate XPhoto YPhoto: " +
//    _XLeftPhoto.ToString() + " " +
//    _YLeftPhoto.ToString());//tambahan
//Trace.WriteLine("Right Photo Coordinate XPhoto YPhoto: " +
//    + _XRightPhoto.ToString() + " " +
//    _YRightPhoto.ToString());//tambahan

AX = A * solution;
CSML.Matrix AxB6 = new CSML.Matrix();
AxB = AX - _b;
CSML.Matrix VT6 = new CSML.Matrix();
VT = AxB.Transpose();
CSML.Matrix VtV6 = new CSML.Matrix();
VtV = VT * AxB;
rCol = A.ColumnCount;
rRow = A.RowCount;
r = rRow - rCol;
//double tes = VtV[1].Re;
double VtVCo6 = VtV[1].Re;

//VtVCo = VtV[0];
// double VtVDo = VtVCo.Re;
double So6;
So6 = Math.Sqrt(VtVCo6) / (r);

//Trace.WriteLine("AX: " + "\n" + AX.ToString());
//Trace.WriteLine("V: " + "\n" + AxB.ToString());

```



```

        //Trace.WriteLine("VT: " + "\n" + VtV.ToString());
        //Trace.WriteLine("r: " + "\n" + r.ToString());
        Trace.WriteLine("So: " + "\n" + So6.ToString());

        iteration++;
    }
}

return _lsmConverge;
}

private bool Threshold(CSML.Matrix m, double threshold)
{
    if /*(Math.Abs(m[1, 1].Re) <= 0.5 || Math.Abs(m[4, 1].Re) <= 0.5)
        && */
        //shift
        (Math.Abs(m[2, 1].Re) <= 0.03 || Math.Abs(m[6, 1].Re) <= 0.03)
        &&
        //scale
        (Math.Abs(m[3, 1].Re) <= 0.03 || Math.Abs(m[5, 1].Re) <= 0.03))
        //shear
        return false;

        //if (Math.Abs(m[1, 1].Re) <= threshold && Math.Abs(m[2, 1].Re) <=
        threshold &&
        //    Math.Abs(m[3, 1].Re) <= threshold && Math.Abs(m[4, 1].Re) <=
        threshold &&
        //    Math.Abs(m[5, 1].Re) <= threshold && Math.Abs(m[6, 1].Re) <=
        threshold)
        //    return false;
    else
        return true;
}

private bool Threshold2()
{
    if (Math.Abs(_correctedMatchedPt.X - _previousMatched.X) <= 0.01 &&
        Math.Abs(_correctedMatchedPt.Y - _previousMatched.Y) <= 0.01)
        return false;
    else
        return true;
}
#endifregion Public Methods
}
}

```

14

LAMPIRAN B

LAMPYAN B

LAMPIRAN

Proses Validasi algoritma *image matching* dengan menggunakan metode
intersection

Data foto

Berikut data foto stereo

a) DSC_0013L.jpg



b) DSC_0013R.jpg



LARIBA

Bosque / sillas alrededor una mesa madera mampara de madera
mesa de madera

DSC_0010

Bstuktur des Tisches

a) DSC_0013F.jpg



b) DSC_0013R.jpg



Data Koordinat (x,y) Foto

Berikut disajikan tabel koordinat foto untuk perhitungan *intersection*:

ID	Koordinat Foto Kiri		Koordinat Foto Kanan	
	x	y	x	y
1	-4.98644	2.56208	-1.91644	5.85436
2	-1.07506	2.68607	0.96879	-0.15445
3	3.32488	2.46666	2.40116	-3.80725
4	-5.4971	1.14262	-3.59646	4.74637
5	-1.52819	1.39572	-0.27809	-0.72584
6	3.13043	1.34045	1.60292	-4.1128
7	-3.6248	-0.11628	-2.65047	0.72771
8	2.78935	-0.37067	0.54948	-4.30253
9	-4.34598	-2.01395	-4.27929	0.63995
10	2.24991	-3.53328	-1.08881	-4.4396

Persamaan yang digunakan dalam proses *intersection*

Wolf & Dewitt, (2000) mengkaji bahwa *intersection* mengacu pada penentuan posisi titik-titik pada *object space* dengan perpotongan garis dari dua atau lebih foto.

Persamaan kesegarisan (*collinearity equations*) merupakan metode dasar dalam penentuan koordinat titik pada *object space*. Persamaan kesegarisan dinyatakan sebagai berikut (*Wolf & Dewitt, 2000*).

$$x_a = x_0 - f \frac{m_{11}(X_A - X_L) + m_{12}(Y_A - Y_L) + m_{13}(Z_A - Z_L)}{m_{31}(X_A - X_L) + m_{32}(Y_A - Y_L) + m_{33}(Z_A - Z_L)} \quad (1a)$$

$$y_a = y_0 - f \frac{m_{21}(X_A - X_L) + m_{22}(Y_A - Y_L) + m_{23}(Z_A - Z_L)}{m_{31}(X_A - X_L) + m_{32}(Y_A - Y_L) + m_{33}(Z_A - Z_L)} \quad (1b)$$

L,a dan A berada pada satu garis lurus, dimana:

x_a, y_a : koordinat foto titik a

X_A, Y_A, Z_A : koordinat *object space* titik A

X_L, Y_L, Z_L : koordinat *object space* stasiun pemotretan

Delta Koordinat (x_2) Foto

Besluit ditzijkerin tabel koordinat foto nummer beperkingen wazzaer won

ID	Koordinat Foto Kiri	Koordinat Foto Recht	Koordinat Foto Tengah	Z
1	-4,08014	5,39308	2,82430	-1,01041
2	-1,03206	5,08903	0,09249	-0,12142
3	3,35488	2,48808	3,40119	-3,28252
4	-3,40311	1,11435	2,256949	4,14932
5	-1,322810	1,76235	0,53284	-1,11128
6	3,13043	1,54013	1,00255	-0,53331
7	-3,05349	-0,11028	2,82042	-0,53331
8	3,38282	-0,31062	0,31613	-3,05253
9	-4,31268	-3,01132	-4,32820	0,53282
10	2,34661	-3,23358	-1,08881	-1,4366

Persamaan dan dengungan datar basa interseksi

Wala' qd Yamaw (300) memangkali persamaan masing-masing basa penentuan posisi titik-titik basa saka. Kita sebutkan persognan saka ini dan akan telpi jika persamaan kesesuaian (collinearity equations) masing-masing titik datar penentuan koordinat titik basa saka basa kesesuaian titik datar

sepasang persamaan (Wala' qd Yamaw 300)

$$(a) \frac{(\chi - k\chi)_{E1}w + (\chi - k\chi)_{E2}w + (\chi - k\chi)_{E3}w}{(\chi - k\chi)_{E4}w + (\chi - k\chi)_{E5}w + (\chi - k\chi)_{E6}w} \chi - \theta\chi = \varphi\chi$$

$$(b) \frac{(\chi - k\chi)_{S1}w + (\chi - k\chi)_{S2}w + (\chi - k\chi)_{S3}w}{(\chi - k\chi)_{S4}w + (\chi - k\chi)_{S5}w + (\chi - k\chi)_{S6}w} \chi - \theta\chi = \psi\chi$$

Lihat gambar A persamaan basa saka titik datar

: koordinat foto titik A

: koordinat objek titik A

: koordinat objek yang sama pada pemotongan

- f : panjang fokus kamera
 x_o, y_o : koordinat *principle point* (dapat diketahui dari kalibrasi kamera)
 m : fungsi dari 3 sudut rotasi, terdiri dari ω, ϕ, κ .

Persamaan diatas menunjukkan bahwa posisi sebuah titik di permukaan bumi, di foto dan pusat proyeksi kamera terletak pada satu garis lurus. Persamaan kesegaran tersebut mengandung 9 parameter yang belum diketahui yaitu ω, ϕ, κ , (ω , phi, kappa) yang merupakan parameter rotasi kamera, dan tiga parameter posisi kamera yakni X_L, Y_L, Z_L yang terdapat dalam Orientasi Luar (*Exterior Orientation*). Sedangkan tiga parameter lainnya merupakan koordinat titik objek (X_A, Y_A, Z_A). Dimana f merupakan panjang fokus kamera. Ketiga parameter ini merupakan parameter dari Orientasi Dalam (*Interior Orientation*).

Dengan menggunakan tiga sudut rotasi, hubungan antara sistem koordinat foto (x, y, z) dan sistem koordinat objek (X, Y, Z) dapat ditentukan. Sebuah matriks dengan dimensi 3×3 dapat mendefinisikan hubungan antara dua sistem yang digunakan. Adapun matriks rotasi dapat didefinisikan sebagai berikut:

$$M = \begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{bmatrix} \quad (2)$$

Dari persamaan diatas, nilai masing-masing element matriks M dapat diperoleh dengan menggunakan persamaan berikut ini:

$$m_{11} = \cos \phi \cos \kappa$$

$$m_{12} = \sin \omega \cos \kappa + \cos \omega \sin \kappa$$

: berlauft für alle Elemente

(Koordinatenwerte von jedem Element sind dabei festgelegt)

: folgt aus den 3 Grundgesetzen der Mechanik

Bestimmen dieses Koordinatenvektoren durch die Beziehungsmittel die folgen
dass hierzu drei Koordinatenwerte benötigt sind aus zwei Gründen. Bestimmen kann dies aus
der Tatsache dass die Position eines Punktes durch drei Koordinaten bestimmt wird, während die
Position eines Punktes durch eine Position und eine Orientierung bestimmt wird.
Zwei Meßwerte auf Basis einer Position und einer Orientierung führt zu einem eindeutigen
Wertesatz. Zwei Meßwerte auf Basis einer Position und einer Orientierung führt zu einem eindeutigen
Wertesatz. Zwei Meßwerte auf Basis einer Position und einer Orientierung führt zu einem eindeutigen
Wertesatz. Zwei Meßwerte auf Basis einer Position und einer Orientierung führt zu einem eindeutigen
Wertesatz.

Durchaus möglich ist es diese drei Meßwerte einzeln zu bestimmen. Dies ist jedoch nicht optimal (X, Y, Z).
Sobald sich ein Objekt in Raum bewegt, so dass seine Position und Orientierung geändert werden (X, Y, Z, φ, θ, ψ),
ist es einfacher die gesuchten Werte einzeln zu bestimmen. Dies ist jedoch nicht optimal (X, Y, Z).
Durchaus möglich ist es diese drei Meßwerte einzeln zu bestimmen. Dies ist jedoch nicht optimal (X, Y, Z).

(§)

$$\begin{bmatrix} M_{11} & M_{12} & M_{13} \\ M_{21} & M_{22} & M_{23} \\ M_{31} & M_{32} & M_{33} \end{bmatrix} = M$$

Drei Beziehungen dieser Art müssen die Position des Punktes im Raum bestimmen. Drei Beziehungen dieser Art müssen die Position des Punktes im Raum bestimmen. Drei Beziehungen dieser Art müssen die Position des Punktes im Raum bestimmen.

$$M_{11} = \cos \varphi \cos \kappa$$

$$M_{12} = \sin \varphi \cos \kappa + \cos \psi \sin \kappa$$

$$\begin{aligned}
m_{13} &= -\cos \omega \sin \varphi \cos \kappa + \sin \omega \sin \kappa \\
m_{21} &= -\cos \varphi \sin \kappa \\
m_{22} &= -\sin \omega \sin \varphi \sin \kappa + \cos \omega \cos \kappa \\
m_{23} &= \cos \omega \sin \varphi \sin \kappa + \sin \omega \cos \kappa \\
m_{31} &= \sin \varphi \\
m_{32} &= -\sin \omega \cos \varphi \\
m_{33} &= \cos \omega \cos \varphi
\end{aligned} \tag{3}$$

Sedangkan untuk menghitung nilai q, r, s maka digunakan persamaan dibawah ini.

$$\begin{aligned}
q &= m_{31}(X_A - X_L) + m_{32}(Y_A - Y_L) + m_{33}(Z_A - Z_L) \\
r &= m_{11}(X_A - X_L) + m_{12}(Y_A - Y_L) + m_{13}(Z_A - Z_L) \\
s &= m_{21}(X_A - X_L) + m_{22}(Y_A - Y_L) + m_{23}(Z_A - Z_L)
\end{aligned} \tag{4}$$

Dari persamaan diatas (X_A, Y_A) adalah koordinat foto target dan Z_A adalah fokus kamera (c). Kemudian matriks A dapat dihitung dengan menggunakan persamaan berikut

$$\begin{aligned}
b_{14} &= \frac{f}{q^2}(rm_{31} - qm_{11}) & b_{24} &= \frac{f}{q^2}(sm_{31} - qm_{21}) \\
b_{15} &= \frac{f}{q^2}(rm_{32} - qm_{12}) & b_{25} &= \frac{f}{q^2}(sm_{32} - qm_{22}) \\
b_{16} &= \frac{f}{q^2}(rm_{33} - qm_{13}) & b_{26} &= \frac{f}{q^2}(sm_{33} - qm_{23})
\end{aligned} \tag{5}$$

Matriks L merupakan matriks yang berisi nilai fungsi dimana nilai koordinat foto dikurangi nilai pendekatan koordinat object space. Matriks L dapat dihitung menggunakan persamaan berikut :

$$w_{r3} = -\cos \omega \sin \phi \cos k + \sin \omega \sin k$$

$$w_{s1} = -\cos \phi \sin \omega$$

$$(3) \quad w_{s2} = -\sin \omega \sin \phi \sin k + \cos \omega \cos k$$

$$w_{s3} = \cos \omega \sin \phi \sin k + \sin \omega \cos k$$

$$w_{s1} = \sin \phi$$

$$w_{s2} = -\sin \omega \cos \phi$$

$$w_{s3} = \cos \omega \cos \phi$$

Sedangkala untuk memperoleh nilai r, τ, p maka ditungguan berdasarkan persamaan diatas ini.

$$(jS - kS)_{xx}w + (kY - kY)_{yy}w + w_{x3}(kY - kY)_{zz} = 0$$

$$(jS - kS)_{yy}w + (kY - kY)_{xx}w + (jY - jY)_{zz}w = \tau$$

$$(jS - kS)_{zz}w + (jY - jY)_{yy}w + (kX - kX)_{xx}w = s$$

Dari persamaan diatas (j, Y, k, S) adalah kooridinat polo yang dari X , sedangkan lokasi

kamela (c). Kedua kamela tersebut A dapat dilihat dengan menggunakan persamaan

perlu

$$(jS - kS) \frac{\lambda}{sp} = \tau_S \quad (jY - kY) \frac{\lambda}{sp} = \tau_Y$$

$$(jS - kS) \frac{\lambda}{sp} = \tau_Z \quad (jY - kY) \frac{\lambda}{sp} = \tau_T$$

$$(jS - kS) \frac{\lambda}{sp} = \tau_X \quad (jY - kY) \frac{\lambda}{sp} = \tau_R$$

Menurut T merupakan maklumat yang perlu untuk dimaksudkan lagi pada

dikongsikan untuk bantahanan kooridinat objek tersebut. Maklumat T dapat dilihat

menambahkan berdasarkan perlu :

$$J = x_a - x_0 + f \frac{r}{q} \quad (5a)$$

$$K = y_a - y_0 + f \frac{s}{q} \quad (5b)$$

Sedangkan koreksi diperoleh dengan menghitung matriks X dengan menggunakan persamaan dibawah ini.

$$X = (A^T A)^{-1} (A^T L) \quad (6)$$

Untuk memperoleh nilai residu, maka matriks V dapat dihitung menggunakan persamaan berikut :

$$V = AX - L \quad (7)$$

Standart Deviasi (S_0) dapat dihitung menggunakan persamaan berikut :

$$S_0 = \sqrt{\frac{V^T V}{r}} L \quad (8)$$

Uji validasi metode *image matching* dengan menggunakan teknik *intersection*.

Berikut akan dilakukan uji validasi metode *area based matching* dengan teknik *intersection*, dimana parameter-parameter dari teknik *least square matching* akan dihitung standar deviasinya.

$$F = 18.5558$$

$$X_0 = 0.00$$

$$Y_0 = 0.00$$

$$F_w = 1936$$

$$F_h = 1296$$

$$\frac{1}{\rho} \chi + oX = oX = \chi$$

$$\frac{1}{\rho} \chi + oY = oY = Y$$

(25)

(26)

Sedangkan konsep dibentuk dengan menggunakan matrice X dalam menyatakan

permasalahan dibawah ini

(27)

$$(A^T A)^{-1} (A^T X) = X$$

Untuk memperoleh nilai terikat maksimisasi X dapat dituliskan menggunakan

persamaan berikut :

(28)

$$X = A^{-1} Y$$

Standart Deviasi (S) dapat dituliskan menggunakan persamaan

(29)

$$S = \sqrt{\frac{1}{n-2} \sum_{i=1}^{n-2} (Y_i - \bar{Y})^2}$$

Untuk mendekati nilai yang mendekati nilai rata-rata tersebut
Bentuk persamaan dibentuk dari variabel metode analisa pada menyatakan
lebih jauh lagi, dimana parameter-parameter yang terdapat dalam persamaan
sama dituliskan sebagai berikut.

$$E = 18.558$$

$$00.0 = oX$$

$$00.0 = oY$$

$$639 = E$$

$$1500 = E$$

Ukuran CCD = 23.6 x 15.8

Tabel parameter orientasi luar tiap foto

Photo	X_L	Y_L	Z_L	ω	φ	κ
Left	-0.342	-1.8718	-0.0908	0.3346	0.6424	-0.0338
Right	-2.9792	0.1888	0.2852	-0.5369	-0.3954	1.0051

Orientasi luar tediri dari 3 parameter posisi kamera (X_L, Y_L, Z_L) dan 3 parameter rotasi yaitu omega, phi, kappa (ω, ϕ, κ).

$$\mathbf{M_Left} \left| \begin{array}{ccc} 0.99994 & -0.00052 & -0.01122 \\ 0.00059 & 0.99998 & 0.00583 \\ 0.01121 & -0.00584 & 0.99992 \end{array} \right|$$

$$\mathbf{M_Right} \left| \begin{array}{ccc} 0.99998 & 0.00480 & -0.00355 \\ -0.00498 & 0.99864 & -0.05196 \\ 0.00330 & 0.05197 & 0.99864 \end{array} \right|$$

Setelah mendapatkan nilai m, selanjutnya dilakukan perhitungan untuk menentukan nilai q, r, dan s. Nilai q, r, s ini digunakan untuk menghitung nilai yang akan digunakan dalam menyusun matriks A. Perhitungan nilai q, r, dan s menggunakan Persamaan (4). Berikut adalah nilai q, r, s dari sepasang titik menggunakan dua buah foto:

ID	r_left	q_left	s_left	ID	r_right	q_right	s_right
1	0.89332	-4.41324	-0.60800	1	1.04802	-5.61846	-1.77453
2	-0.47900	-18.21375	0.62181	2	-0.42299	-19.31349	0.258177

Uptown CCD = 53.0 x 12.8

Top left bisimeter offset for top left

Point	Z _x	Z _y	Z _z	S _x	S _y	S _z	K
Left	-0.345	-1.8313	-0.0608	0.3346	0.0454	-0.0393	
Right	-3.9785	-0.3825	0.1988	-0.3546	1.0021	-0.3546	

Offcenter just before first 3 bisimeter point (T_x, T_y) due to 3 bisimeter offset

Align objects, left, right (x, y, z)

M_Tleft	0.00026	0.00008	0.00000	-0.00025	0.00004	0.00000	0.00000
M_Tright	0.00000	0.00000	0.00000	-0.00000	0.00000	0.00000	0.00000
M_Tcenter	0.00000	0.00000	0.00000	-0.00000	0.00000	0.00000	0.00000

M_Tright	0.00000	0.00000	0.00000	-0.00025	0.00000	0.00000	0.00000
M_Tcenter	0.00000	0.00000	0.00000	-0.00000	0.00000	0.00000	0.00000
M_Tleft	0.00000	0.00000	0.00000	-0.00000	0.00000	0.00000	0.00000

Step by step description how to separate axis differences between target measurement
 due to the fact that M₁ is in different position than measurement due to a measurement
 difference between measurement targets A, B and C due to a measurement
 measurement (4) Between objects right due to abscess due to measurement and prop
 total

ID	z_left	x_left	y_left	z_right	x_right	y_right	ID
1	0.86975	-4.41354	-0.00800	1	0.04803	-2.01846	1.33923
2	-0.42000	-1.823132	0.093181	3	-0.15300	0.031348	0.328123
3	-0.42000	-1.823132	0.093181	5	-0.15300	0.031348	0.328123

Selanjutnya untuk menyusun matriks A (matriks koefisien) menggunakan Persamaan (5).

$$A = \begin{vmatrix} -4.2138 & 0.0072 & -0.8039 & 0 & 0 \\ 0.0040 & -4.2079 & 0.5547 & 0 & 0 \\ -3.3046 & -0.0479 & -0.6035 & 0 & 0 \\ 0.0199 & -3.2439 & 1.2133 & 0 & 0 \\ 0 & 0 & 0 & -1.0184 & 0.0004 \\ 0 & 0 & 0 & -0.0010 & -1.0186 \\ 0 & 0 & 0 & -0.9607 & -0.0035 \\ 0 & 0 & 0 & 0.0047 & -0.9601 \end{vmatrix}$$

$$A^T = \begin{vmatrix} -4.2138 & 0.0040 & -3.3046 & 0.0199 & 0 & 0 & 0 & 0 \\ 0.0072 & -4.2079 & -0.0479 & -3.2439 & 0 & 0 & 0 & 0 \\ -0.8039 & 0.5547 & -0.6035 & 1.2133 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1.0184 & -0.0010 & -0.9607 & 0.0047 \\ 0 & 0 & 0 & 0 & 0.0004 & -1.0186 & -0.0035 & -0.9601 \\ 0 & 0 & 0 & 0 & 0.0382 & -0.0407 & 0.0244 & 0.0371 \end{vmatrix}$$

$$A^T * A = \begin{vmatrix} 28.6775 & 0.0466 & 5.4080 & 0 & 0 & 0 \\ 0.0466 & 28.2317 & -6.2467 & 0 & 0 & 0 \\ 5.4080 & -6.2467 & 2.7901 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1.9601 & -0.0005 & -0.0622 \\ 0 & 0 & 0 & -0.0005 & 1.9593 & 0.0058 \\ 0 & 0 & 0 & -0.0622 & 0.0058 & 0.0051 \end{vmatrix}$$

$$(A^T * A)^{-1} = \begin{vmatrix} 0.1278 & -0.1090 & -0.4918 & 0 & 0 & 0 \\ -0.1090 & 0.1632 & 0.5767 & 0 & 0 & 0 \\ -0.4918 & 0.5767 & 2.6028 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.8344 & -0.0300 & 10.2233 \\ 0 & 0 & 0 & -0.0300 & 0.5132 & -0.9500 \\ 0 & 0 & 0 & 10.2233 & -0.9500 & 322.3430 \end{vmatrix}$$

Sedangkan matriks L merupakan matriks yang berisikan nilai fungsi dimana nilai koordinat dikurangi nilai pendekatan koordinat *object space*. Penyusunan matriks L diawali dengan menghitung nilai J dan K, dimana formula yang digunakan terdapat dalam Persamaan (5a) dan Persamaan (5b).

$$L \left| \begin{array}{c} -1.23042 \\ 0.005691 \\ 1.544815 \\ -0.00629 \\ -1.56306 \\ 3.319557 \\ 0.56239 \\ 0.093598 \end{array} \right|$$

Selanjutnya Matriks X dihitung menggunakan Persamaan (6), dan hasilnya adalah sebagai berikut:

$$X \left| \begin{array}{c} -0.00614 \\ 0.00740 \\ 0.04723 \\ -0.83747 \\ -1.64522 \end{array} \right|$$

$$A * X \left| \begin{array}{c} -0.01204 \\ -0.00498 \\ -0.00857 \\ 0.033166 \\ -0.80095 \\ 3.438152 \\ -0.24624 \\ -0.02895 \end{array} \right|$$

$$V \left| \begin{array}{c} 1.218381 \\ -0.010667 \\ -1.553381 \\ 0.039454 \\ 0.762102 \\ 0.118594 \\ -0.808628 \\ -0.122549 \end{array} \right|$$

$$So \quad | \quad 1.51554975 \text{ mm} \quad |$$

Koordinat *Object Space* pendekatannya diperoleh dari penjumlahan transformasi koordinat 3D dengan delta X, delta Y dan delta Z. Hasilnya adalah sebagai berikut:

ID	delta_X	delta_Y	delta_Z
1	-0.00613994	0.007401998	0.047226725
2	-0.83746999	-1.645	-43.260

Dari hasil diatas maka dapat dihitung adjustment value dari masing-masing koordinat dan hasilnya adalah sebagai berikut:

ID	X	Y	Z
1	0.495283337	-2.44708643	-4.47002906
2	-1.86228309	-2.7886048	-61.5537661

ID	defect_X	defect_Y	defect_Z
1	0.00013884	0.00140188	0.041352452
3	-0.83248633	-1.612	-43.340

Dati pressi questo file sono relativi alla sperimentazione dei missini-fondente

dai pressi della separazione per la

ID	Z	Y	X
1	0.40238233	-3.4430843	-4.43003609
3	-1.84538306	3.388048	-91.5233991