

SKRIPSI

PEMBUATAN PROGRAM PERHITUNGAN *BUNDLE ADJUSTMENT* DENGAN MENGGUNAKAN BAHASA PEMROGRAMAN C# 2008 DALAM *CLOSE RANGE PHOTOGRAMETRI*



**Diajukan untuk memenuhi persyaratan
dalam mencapai gelar sarjana S1 Teknik Geodesi**

Disusun Oleh :

AKBAR BAGUS RIYADI

04.25.008

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

2019

2019

REKAM JEKAT
REKAM JEKAT
REKAM JEKAT
REKAM JEKAT
REKAM JEKAT

01/12/2019

REKAM JEKAT

REKAM JEKAT

REKAM JEKAT

REKAM JEKAT

REKAM JEKAT

REKAM JEKAT



PERKUMPULAN PENGELOLA PENDIDIKAN UMUM DAN TEKNOLOGI NASIONAL MALANG
INSTITUT TEKNOLOGI NASIONAL MALANG

FAKULTAS TEKNOLOGI INDUSTRI
FAKULTAS TEKNIK SIPIL DAN PERENCANAAN
PROGRAM PASCASARJANA MAGISTER TEKNIK

PT. BNI (PERSERO) MALANG
BANK NIAGA MALANG

Kampus I : Jl. Bendungan Sigura-gura No. 2 Telp. (0341) 551431 (Hunting), Fax. (0341) 553015 Malang 65145
Kampus II : Jl. Raya Karanglo, Km 2 Telp. (0341) 417636 Fax. (0341) 417634 Malang

LEMBAR PENGESAHAN

**Pembuatan Program Perhitungan Bundle Adjustment Dengan Menggunakan
Bahasa Pemrograman C# 2008 Dalam Close Range Photogrametri**

SKRIPSI

Dipertahankan di hadapan Majelis Penguji Sidang Skripsi
Jenjang Starata Satu (S-1)

Pada hari : Jumat

Tanggal : 20 Agustus 2010

Dan diterima untuk memenuhi salah satu persyaratan guna memperoleh gelar Sarjana Teknik.

Disusun oleh

Akbar Bagus Riyadi

04.25.008

Panitia Ujian Tugas Akhir

Ketua

Hery Purwanto, ST, M.Sc

Sekretaris

Silvester Sari Sai, ST, MT

Anggota Penguji

Penguji I

Ir. Leo Patimena, M.Sc

Penguji II

Silvester Sari Sai, ST, MT

Penguji III

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

**LEMBAR PERSETUJUAN
SKRIPSI**

**Pembuatan Program Perhitungan Bundle Adjustment Dengan
Menggunakan Bahasa Pemrograman C# 2008 Dalam Close Range
Photogrametri**

Diajukan Sebagai Salah Satu Syarat Memperoleh Gelar Sarjana Teknik Geodesi
S-1

Institut Teknologi Nasional Malang

Disusun Oleh :

Akbar Bagus Riyadi

04.25.008

Meyetujui,

Dosen Pembimbing I



Hery Purwanto, ST, M.Sc

Dosen Pembimbing II



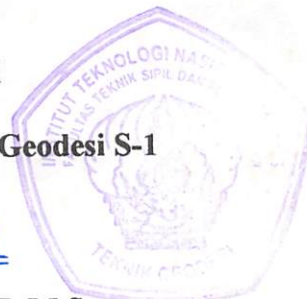
Dr. Edwin Tjahjadi, ST. M.Geom.Sc

Mengetahui

Ketua Jurusan Teknik Geodesi S-1



Hery Purwanto, ST, M.Sc



ABSTRAKSI

Akbar Bagus Riyadi, 2010, “Pembuatan Program Perhitungan Bundle Adjustment Dengan Menggunakan Bahasa Pemrograman C# 2008 Dalam Close Range Photogrametri”

Dosen Pembimbing I : Hery Purwanto, ST, M.Sc

Dosen Pembimbing II : Dr. Edwin Tjahjadi, ST. M.Geom.Sc.

Bundle adjustment merupakan salah satu bagian dari rangkaian proses *close range photogrammetry*. *Bundle adjustment* digunakan untuk mengevaluasi parameter *exterior orientation* (EO) dan koordinat titik objek yang dihasilkan pada tahap *relative orientation* dan *intersection*, sehingga diperoleh nilai akurasi dan presisi dari koordinat titik objek.

Program perhitungan proses *bundle adjustment* dalam metode *Close Range Photogrammetry* didesain menggunakan bahasa pemrograman Visual Studio C# 2008.

Berdasarkan hasil analisis, program perhitungan Bundle Adjustment ini memiliki kelebihan dan kekurangan. Kelebihan program ini adalah User interaktif dan fleksibel, sedangkan dari segi kekurangan data program ini masih menggunakan 2 foto untuk proses pengolahan datanya.

Kata Kunci : Program, *Bundle Adjustment*, *Close Range Photogrammetry*

PERNYATAAN KEASLIAN SKRIPSI

Saya yang bertanda tangan dibawah ini.

Nama : Akbar Bagus Riyadi

NIM : 0425008

Program Studi : Teknik Geodesi S-1

Fakultas : Fakultas Teknik Sipil Dan Perencanaan

Menyatakan dengan sesungguhnya bahwa Skripsi saya dengan judul :

“Pembuatan Program Perhitungan Bundle Adjustment Dengan Menggunakan Bahasa Pemrograman C# 2008 Dalam Close Range Photogrametri”

adalah hasil karya saya sendiri, bukan merupakan duplikat serta tidak mengutip atau menyadur dari hasil karya orang lain kecuali disebutkan sumbernya.

Malang, 29 September 2010

Yang membuat pernyataan

Akbar Bagus Riyadi

0425008

KATA PENGANTAR

Asslamulaikum,WR WB, Puji syukur penulis panjatkan kepada Allah SWT, karena berkat rahmat-Nya, penulis dapat menyelesaikan skripsi yang berjudul ” **Pembuatan Program Perhitungan Bundle Adjustment Dengan Menggunakan Bahasa Pemrograman C# 2008 Dalm Close Range Photogrametri**”, 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, peneliti ingin 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 Dan Dosen Pembimbing II Geodesi Institut Teknologi Nasional Malang.
4. Bapak Dr. Edwin Tjahjadi, ST., MGeom.Sc. selaku Dosen Pembimbing I dan Dosen Penguji.
5. Bapak Silvester Sari Sai, ST., MT. selaku Dosen Penguji.
6. Bapak Leo Patimena, ST., M.Sc. selaku Dosen Penguji.

7. Segenap dosen, staff pengajar dan rekording Jurusan Teknik Geodesi Fakultas Teknik Sipil dan Perencanaan Institut Teknologi Nasional Malang.
8. Papa, Mama dan kakak-kakakku serta seluruh anggota keluarga besar Ir.Mu'min MM
9. Team Deformasi,Rapid mapping dan keluarga besar 2004 yang selalu memberikan kerja sama dan dukungannya.
10. Semua pihak yang telah membantu peneliti yang tidak dapat disebutkan satu persatu.

Semoga Allah SWT memberikan karunia atas budi baik dari semua pihak yang terlibat dalam penelitian ini. Penulis menyadari bahwa penulisan laporan ini masih belum sempurna, baik dari segi materi, sistematika pembahasan, maupun susunan bahasanya Oleh karena itu, kritik dan saran yang membangun sangat penulis harapkan. Hasil penelitian ini dan dengan segala keterbatasannya dipersembahkan kepada dunia pendidikan, semoga ada manfaatnya untuk pengembangan sumber daya manusia di negara tercinta ini.

Malang, 29 September 2010

Penulis

LEMBAR PERSEMBAHAN

Assalamualaikum. WR, WB. Annyeonghaseyo, Konichiwa. Akhirnya selesai sudah skripsi yang rumit dan menyita segala perhatianku selama 2 tahun ini. Entah berapa hari aku tidak tidur, entah berapa kali ku melewatkan sarapan, makan siang dan makan malamku untuk menyelesaikan programku ini. Entah berapa kali aku harus berbohong kepada orang tua ku demi mendanai hidupku yang tidak karu-karuan hanya (hanya???) karena program ini. Kesabaran dan kerja keras ku hanya untuk menyelesaikan ini semua akhirnya terbayar sudah. Butuh 6 tahun untuk menyelesaikan gelar S-1 ku..huft!! tapi aku tidak menyesal, banyak hal dan pengalaman yang membuatku jadi lebih dewasa dan lebih mengenal kehidupan.

Puji Syukur yang sebesar-besarnya aku panjatkan kepada Allah SWT ats berkat rahmat dan nikmatnya aku bisa menyelesaikan semua kewajiban ku khususnya sebagai Mahasiswa, anak laki-laki dan sebagai manusia. Serta Nabi Muhammad SAW atas kerja keras serta tuntunan untuk membimbing manusia kejalan Allah sehingga aku tahu aku harus berdoa, beribadah, dan berserah diri kepada Allah SWT.

Ucapan terima kasihku tak henti-henti nya untuk Ibu dan Ayahku yang telah melahirkan aku dan memberikan kesempatan untuk melihat dunia ciptaan Allah yang indah ini. UMMMA..maafkan putramu yang selalu membuatmu marah dan menangis ini. Maafkan putramu yang nakal ini karena tidak pernah menuruti nasehatmu. Maafkan putramu yang selalu berbohong ini karena sering membohongimu. APPA..maafkan putramu yang boros ini selalu meminta uang yang terkira jumlahnya. Maafkan putramu yang selalu membebani pikiran mu ini. Aku berjanji akan membuat nama keluarga kita tetap menjadi keluarga yang dihormati dan akan menjaga ketuhanan keluarga kita. Aku berjanji akan menjaga ibu dan ayah hingga Allah memanggilku nanti APPA..CEPAT SEMBUH YA!!



Terima kasih yang sebesar-besarnya untuk semua jajaran Dosen Pengajar Teknik Geodesi ITN Malang. Khususnya Bapak Heri Purwanto, ST., M.Sc yang selalu sabar mengingatkan saya untuk segera menyelesaikan kewajiban sebagai mahasiswa, Bapak Dr. Edwin Tjahjadi, ST., M.GEOM.Sc yang telah banyak mengajari dan membimbing hingga program ini selesai. Bapak Silvester Sari Sai, ST., MT yang banyak membantu dan mengoreksi kesalahan selama program ini dibuat. Bapak Leo Patimena, ST., M.Sc selaku dosen penguji yang banyak memberi kritikan membangun. Dan juga Ibu Sulis selaku bagian administrasi. Sungguh hutang yang tak bisa terhitung untuk semua kebaikan beliau-beliau yang senantiasa memberikan ilmu, pengalaman serta kesabaran untuk membantu saya selama ini.



Terima kasih untuk semua anggota tim Deformasi, Tansil dan Dodik (both of them is the smartest friend i ever had), Lia (thanks for your kindness), Enno, Gedhe, Chandra, Alben (huft..!), Tim Rapid Mapping, Pia (thanks for everything), Riri, Yusak, Dodik, Ona, Dessy, Roger. THANKS GUYS...

Terima kasih juga untuk semua angkatan 2004 (Bagus, Nanang, Henry, Ary, Gigih, Arif, Justin, Elwyn, Nopy, Desy, Titin, Marwar, Acul, Tryana, Budi), Anto, Mas Ardi dan semua mahasiswa Geodesi yang tidak bisa saya sebutkan satu persatu.

Terima kasih pula untuk teman-teman kos ku yang selalu merepotkan atau pun kurepotkan..(he he); Bayu Prayoga (macul terus), Mas Adjie, Mamat, Gigih robot, Sampluk dan Semua yang pernah menderita di MOAT 8 tercinta. Teman teman mojokerto; Brian(Daob), Eka Mahendra, Supri, Sigit terima kasih banyak telah mau menjadi temanku hingga saat ini.

Ucapan terima kasih juga tidak lupa saya ucapkan kepada semua anggota dan Admin forum SAFE, SOSHIFIED (he he) dan SAYA. Semua pecinta dan penggemar musik dan film korea..Saranghamida...HWAITING..



DAFTAR ISI

Halaman Sampul Depan

Halaman Judul

Lembar Pengesahan

Lembar Persetujuan

Abstraksi

Pernyataan Keaslian Skripsi

Kata Pengantar

Daftar Isi

Daftar Gambar

Daftar Tabel

BAB 1 PENDAHULUAN

1.1 Latar Belakang.....	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah	2
1.4 Tujuan Penelitian	3
1.5 Manfaat Penelitian	3

BAB II DASAR TEORI

2.1 <i>Bundle Adjustment</i> dan Kesalahan Observasi	4
2.2 <i>Metode</i> Gabungan Kuadrat Terkecil (<i>Unified Least Square Method</i>)	6

2.2.1	<i>Adjustment of Indirect Observations</i>	6
2.2.2	<i>Adjustment of Observations Only</i>	7
2.3	<i>Conventional Bundle Adjustment</i>	8
2.3.1	Persamaan Kolinear.....	8
2.3.1.1	Matrik Rotasi.....	9
2.3.1.2	Transformasi Perspektif Pusat (<i>Central Perspective Transformation</i>)	13
2.3.1.3	Kondisi Kolinear.....	15
2.3.2	Persamaan Observasi.....	16
2.3.3	Struktur Matrik Bobot	17
2.3.4	Model Matematika	19
2.3.5	Linierisasi Persamaan Kolinier	19
2.3.6	Desain Matrik untuk <i>Bundle Adjustment</i>	20
2.3.7	Struktur dari Matrik Persamaan Normal	23
2.3.8	Penyelesaian Persamaan Normal <i>Bundle Adjustment</i>	26
2.4	Model Penilaian <i>Bundle Adjustment</i>	27
2.5	<i>Network Design</i>	27
2.5.1	Desain Target.....	28
2.5.2	Desain Pemotretan.....	29
2.5.3	Permasalahan Datum.....	31
2.6	Pembuatan Algoritma <i>Bundle Adjustment</i>	32
2.7	Pembuatan <i>Interface</i> Dalam C#.....	33

BAB III METODE PENELITIAN

3.1	Persiapan Penelitian	37
3.1.1	Materi Penelitian	37
3.1.2	Alat Penelitian	38
3.1.2.1	Perangkat Keras (<i>Hardware</i>).....	38
3.1.2.2	Perangkat Lunak (<i>Software</i>)	40
3.2	Diagram Alir Penelitian	41
3.3	Diagram Alir Program	43
3.3	Penjelasan Diagram Alir (<i>Flow Chart</i>) Program.....	44

BAB IV HASIL DAN ANALISA

4.1	Hasil	49
4.1.1	Langkah Kerja Program.....	49
4.1.1.1	<i>Project Setting</i>	53
4.1.1.2	<i>Add Image to Project</i>	54
4.1.1.3	<i>Processing Image Dengan Centroid</i>	55
4.1.1.4	<i>Processing Dengan Relative Orientation</i>	57
4.1.1.5	<i>Processing Intersection</i>	58
4.1.1.6	<i>Processing Resection</i>	61
4.1.1.7	<i>Bundle Adjustment</i>	62
4.2	Analisa Output Program.....	63

BAB V PENUTUP

5.1	Kesimpulan.....	64
5.2	Saran	68

DAFTAR PUSTAKA

LAMPIRAN

DAFTAR GAMBAR

Gambar 2.1	Proses <i>Bundle</i> Multi Foto	5
Gambar 2.2	Ilustrasi Proses <i>Bundle Adjustment</i>	6
Gambar 2.3	Rotasi Sudut Omega terhadap Sumbu $X\omega$	9
Gambar 2.4	Rotasi Sudut Phi terhadap Sumbu $Y\phi$	10
Gambar 2.5	Rotasi Sudut Kappa terhadap Sumbu $Z\kappa$	11
Gambar 2.6	Hubungan Keruangan antara Titik A Sistem Koordinat Sembarang dengan Titik a Sistem Koordinat Kamera.....	13
Gambar 2.7	Struktur Persamaan Normal	25
Gambar 2.8	Geometri Pemotretan	28
Gambar 2.9	Diagram Desain Target	29
Gambar 2.10	<i>Design Left-Right Measurements</i>	30
Gambar 2.11	Desain Jaringan Empat Stasiun	30
Gambar 2.12	Ilustrasi Penempatan Stasiun Pemotretan	30
Gambar 2.13	Tampilan Opening Program	34
Gambar 2.14	Tampilan Input Data	34
Gambar 2.15	Tampilan Ouput Data	35
Gambar 2.16	Tampilan Safe Final Data.....	35
Gambar 2.17	Tampilan Perubahan Parameter EO sebelum dan Sesudah Proses <i>Bundle Adjustment</i>	36
Gambar 2.18	Tampilan Perubahan Object Space Point sebelum dan Sesudah Proses <i>Bundle Adjustment</i>	36
Gambar 3.1	Kamera SLR Nikon D 60 tampak depan dan belakang	38
Gambar 3.2	Aksaesoris dari Kamera (Tipe Lensa, Baterai, Memori)	38

Gambar 3.3	Striker Retro Target	39
Gambar 3.4	Tampilan awal Program C#.....	40
Gambar 3.5	Diagram Alir Penelitian (1).....	41
Gambar 3.6	Diagram Alir Penelitian (2).....	42
Gambar 3.7	Diagram Alir Program	43
Gambar 3.8	Input Dari <i>Notepad</i>	44
Gambar 3.9	Susunan Matrix N dan t (Fraser, 1997).....	47
Gambar 4.1	Program <i>Overview</i>	50
Gambar 4.2	<i>Project Setting</i>	53
Gambar 4.3	<i>Add Image Open Dialog</i>	54
Gambar 4.4	<i>Show Image To Image Box</i>	55
Gambar 4.5	Proses <i>Centroid</i>	56
Gambar 4.6	<i>Save Centroid</i>	56
Gambar 4.7	Proses <i>Load Data</i> untuk <i>RO</i>	57
Gambar 4.8	<i>Save File RO</i>	58
Gambar 4.9	<i>Load file</i> untuk proses <i>Intersection</i>	59
Gambar 4.10	<i>Save File Intersection for Image 1</i>	60
Gambar 4.11	<i>Save File Intersection for Image 2</i>	60
Gambar 4.12	<i>Load file Intersection Image 1</i>	61
Gambar 4.13	<i>Load file Intersection Image 2</i>	61
Gambar 4.14	<i>Save File untul proses Bundle</i>	62
Gambar 5.1	Tampilan program keseluruhan	67

DAFTAR TABEL

Tabel 4.1	Parameter EO final	63
Tabel 4.2	Koordinat <i>Object Space Point Final</i>	63
Tabel 4.3	Presisi Object Space Point	63
Tabel 5.1	Parameter EO pendekatan.....	64
Tabel 5.2	Koordinat <i>Object Space Point</i> pendekatan	64
Tabel 5.3	Koordinat Foto dan Presisi setiap Titik pada Foto Satu	65
Tabel 5.4	Koordinat Foto dan Presisi setiap Titik pada Foto Dua.....	65
Tabel 5.5	Parameter EO final	66
Tabel 5.6	Koordinat <i>Object Space Point Final</i>	66
Tabel 5.7	Presisi <i>Object Space Poin finat</i>	67

BAB I

PENDAHULUAN

1.1 Latar Belakang

Dalam perkembangannya penggunaan metode *close range photogrammetry* dalam berbagai aplikasi teknik dan survei semakin meningkat (Linder, 2006). Istilah Close range Photogrametri sendiri digunakan untuk mendiskripsikan teknik fotogrametri saat jarak dari objek yang diukur kurang dari 100 meter dan kamera diletakan dekat dengan objek (Sahdan, 2006). Suatu metode penting didalam metode perhitungan dalam kasus *Close Range Photogrametri* adalah perhitungan *Bundle Adjustment*. Dimana metode ini bertujuan untuk mendapatkan akurasi dan ketelitian tinggi (Shirkhani et al., 2006).

Perhitungan *Bundle Adjustment* dalam *Close Range Photogrammetry* tidak terlepas dari dari proses *resection* dan *intersection*. Dimana proses *Resection* merupakan proses penentuan posisi dan orientasi luar dalam tiap foto. Proses *Intersection* merupakan teknik untuk menentukan koordinat titik-titik objek pada dua buah foto atau lebih yang saling bertampalan sehingga dapat diketahui posisi secara 3D (Mikhail et al., 2001). Karena parameter-parameter dalam proses *resection* dan *intersection* akan digunakan dalam perhitungan *bundle adjustment*.

Secara detail pengertian Bundle Adjustment sendiri adalah proses penentuan atau perhitungan parameter *interior orientation* (IO), *exterior orientation* (EO yaitu ω , φ , κ , X_L, Y_L, Z_L), dan koordinat objek yang dihitung secara bersamaan dengan menggunakan teknik hitung kuadrat terkecil (Tjahjadi, 2008a).

Tugas Akhir dengan topik : “ **Pembuatan Program Perhitungan *Bundle Adjustment* Di Dalam *Close Range Photogrametri* dengan menggunakan bahasa pemrograman *Visual Studio C# 2008*” memiliki tujuan untuk mendapatkan nilai output yang akurat dengan tingkat ketelitian yang sangat tinggi dengan Bahasa Pemrograman Visual Studio 2008 yang memiliki basis “*Windows*”.**

1.2 Rumusan Masalah

Adapun rumusan masalah yang akan dibahas dalam skripsi ini adalah bagaimana Proses Pembuatan Program Perhitungan *Bundle Adjustment* didalam *Close Range Photogrametri* dengan menggunakan bahasa pemrograman Visual Studio C# 2008?

1.3 Batasan Masalah

Batasan Permasalahan yang akan dibahas didalam penyusunan Tugas Akhir ini adalah :

1.3.1 Algoritma dan desain interface yang digunakan hanya untuk metode perhitungan bundle adjustment Didalam *Close Range Photogrametri*.

1.3.2 Output Program adalah parameter orientasi luar (*Eksterior Orientation* atau $\omega, \varphi, \kappa, X_L, Y_L, Z_L$), koordinat *Object Space* (X_i, Y_i, Z_i), akurasi dari titik-titik objek ($\delta x, \delta y, \delta z$) yang merupakan parameter akhir.

1.4 Tujuan Penelitian

Sesuai dengan rumusan masalah yang akan dibahas dalam tugas akhir ini, maksud dan tujuan dari penulisan skripsi ini adalah Membuat Program Perhitungan Bundle Adjustment Didalam Close Range Photogrametri yang memiliki tingkat akurasi tinggi.

1.5 Manfaat Penelitian

Adapun manfaat yang diharapkan dalam penelitian ini yaitu, Memudahkan user untuk melakukan perhitungan Bundle Adjustment didalam Close Range Photogrametri serta mempersingkat waktu dan biaya jika dibandingkan dengan menggunakan Software komersil yang lain.

BAB II

DASAR TEORI

Dalam penulisan ini, akan dibahas secara khusus mengenai program perhitungan *Bundle Adjustment* dalam *Photogrammetry*. *Photogrammetry* tidak terlepas dari dari proses *resection* dan *intersection*. Dimana proses *Resection* merupakan proses penentuan posisi dan orientasi luar dalam tiap foto. Proses *Intersection* merupakan teknik untuk menentukan koordinat titik-titik objek pada dua buah foto atau lebih yang saling bertampalan sehingga dapat diketahui posisi secara 3D (Mikhail et al, 2001). Dimana operasi ini akan dikombinasikan dalam proses triangulasi, atau *block adjustment*, dimana parameter orientasi foto dan koordinat objek akan dihitung secara serempak.

Penyesuaian dalam kuadrat terkecil membentuk dasar dari beberapa algoritma yang digunakan dalam proses perhitungan *close range photogrammetry*. *Bundle adjustment* merupakan aplikasi yang paling sering digunakan dalam teori penyesuaian kuadrat terkecil pada *close range photogrammetry*.

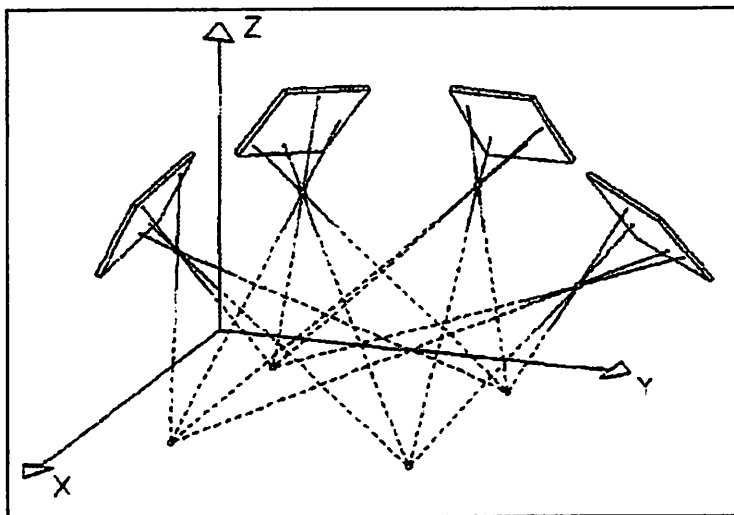
2.1 *Bundle Adjustment* dan Kesalahan Observasi

Bundle Adjustment merupakan proses penentuan atau perhitungan parameter *interior orientation* (IO), *exterior orientation* (EO), dan koordinat objek yang dihitung secara bersamaan dengan menggunakan teknik hitung kuadrat terkecil (Tjahjadi, 2008a). Parameter-parameter yang digunakan dalam proses *bundle adjustment* adalah parameter IO, EO, serta koordinat objek pendekatan dari objek yang diamati.

Mengenai alasan pemilihan *bundle adjustment* dalam perhitungan parameter-parameter *close range photogrammetry*, karena *bundle adjustment* (Triggs et al, 2000).

1. Fleksibilitas : *Bundle adjustment* dapat dengan baik memecahkan berbagai masalah antara lain berbagai bentuk 3D dan tipe kamera (titik, garis, kurva, surface, non-metrik, metrik, dsb), berbagai tipe skema (termasuk model dinamik dan artikulasi, skema constraints), sumber informasi (fitur 2D, tingkat intensitas, informasi 3D) dan *error models*.
2. Akurasi : *Bundle Adjustment* memberikan hasil yang presisi dan mudah dalam interpretasinya karena menggunakan model kesalahan statistik secara akurat.
3. Efisiensi : Algoritma *bundle* yang telah matang dan relatif efisien meskipun pada permasalahan yang sangat besar.

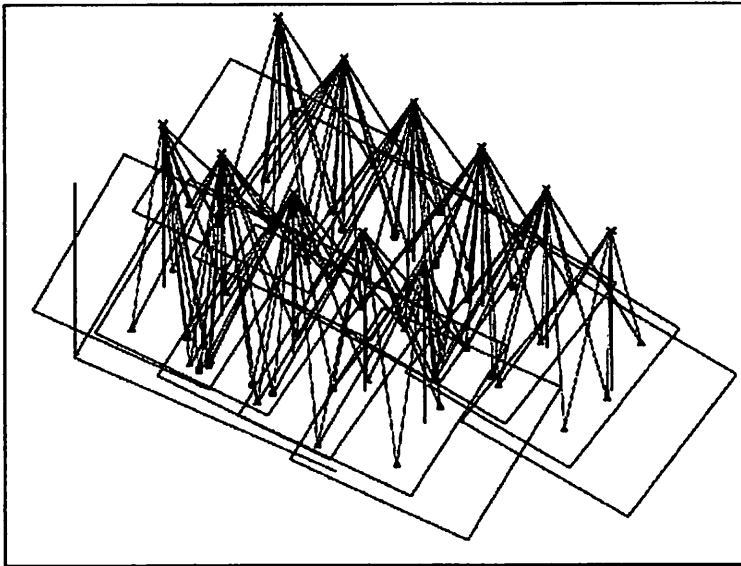
Bundle Adjustment merupakan suatu bentuk penyesuaian yang ditujukan untuk melakukan estimasi secara simultan dalam penentuan nilai parameter orientasi dari koordinat objek yang diinginkan. Proses tersebut meliputi penempatan foto-foto secara bebas dalam *object space* membentuk jaringan *photogrammetry*, seperti ditunjukkan pada Gambar (2.1) (Uffenkam, 1998) :



Gambar 2.1 Proses Bundle Multi Foto

Secara grafis proses dari *bundle adjustment* dapat dilihat pada gambar berikut

(Jacobsen, 2000) :



Gambar 2.2 Ilustrasi proses Bundle Adjustment

Dalam melakukan observasi maupun perhitungan, dapat terjadi tiga jenis kesalahan (*error*), yaitu *gross error*, *systematic error*, dan *random error*. Ukuran dari masing-masing kesalahan tersebut akan mempengaruhi keakuratan dan presisi dari parameter yang dihitung.

2.2 Metode Gabungan Kuadrat Terkecil (*Unified Least Square Method*)

Metode gabungan kuadrat terkecil (*Unified Least Square Method*) mengkombinasikan dua jenis persamaan fungsional yang berbeda yang umum digunakan dalam penyesuaian kuadrat terkecil, yaitu persamaan observasi (*observation equation*) dan persamaan kondisi (*condition equation*). Kedua tipe persamaan tersebut telah digunakan untuk merumuskan metode pemisahan kuadrat terkecil (King, 1993).

Mikhail menyatakan bahwa terdapat dua metode dalam penyesuaian, yaitu *Adjustment of Indirect Observations* dan *Adjustment of Observations Only*

2.2.1 *Adjustment of Indirect Observations*

Persamaan dengan menggunakan metode ini mengandung observasi, konstanta dan parameter. Parameter merupakan sebuah variabel *stochastic* yang nilainya perlu diestimasi selama proses kuadrat terkecil. Masing-masing persamaan mengandung hanya satu observasi. Model fungsional penyesuaian dari observasi tidak langsung (*Adjustment of Indirect Observations*) sebagai berikut :

$$l + v + B\Delta = d \dots\dots\dots (2.1)$$

atau

$$v + B\Delta = -l + d = f \dots\dots\dots (2.2)$$

dimana l dan v merupakan vektor observasi dan residu, d merupakan vektor konstanta, B merupakan matrik parameter koefisien, dan Δ merupakan vektor parameter (King, 1993).

2.2.2 *Adjustment of Observations Only*

Dalam metode ini, kombinasi observasi dan residunya sama dengan *physical constantnya*. Model fungsionalnya sebagai berikut:

$$A(l + v) = d \dots\dots\dots (2.3)$$

atau

$$Av = d - Al = f \dots\dots\dots (2.4)$$

Dimana A merupakan matrik koefisien observasi, l dan v adalah vektor observasi dan residunya, d adalah vektor nilai konstan (King, 1993).

Dengan pendekatan gabungan kuadrat terkecil diasumsikan bahwa semua variabel yang terkandung dalam fungsi matematika ini merupakan observasi. Maka persamaanya sebagai berikut :

$$AV + B\Delta = f \dots\dots\dots (2.5)$$

jika dikombinasikan dengan matrik bobot (W) akan menghasilkan persamaan sebagai berikut :

$$\begin{aligned} \Delta &= (B^T W B)^{-1} B^T W f \\ &= N^{-1} t \end{aligned} \dots\dots\dots (2.6)$$

2.3 Conventional Bundle Adjustment

Conventional Bundle Adjustment, yang sering disebut dengan BA, merupakan bangun dasar dari *Least Square Bundle adjustment* yang dikembangkan oleh Brown pada tahun 1958 (dalam King, 1993). Selain BA, terdapat beberapa model lain dalam *Bundle Adjustment*, seperti Model DLT, yang dikembangkan oleh Abdel-Aziz dan Karara serta Model Penyesuaian Observasi Tidak Langsung yang dikembangkan oleh Mikhail. Menurut hasil penelitian yang dilakukan oleh Ruther (1989), 80% dari program komputer yang menyediakan solusi untuk *photogrammetry* mengacu pada *bundle adjustment* (King, 1993). Hal ini menunjukkan bahwa *bundle adjustment* menjadi model perhitungan yang dominan diterapkan dalam ilmu *photogrammetry* dibandingkan dengan model lainnya.

Pada umumnya, *bundle adjustment* dipergunakan untuk menentukan parameter *interior orientation* (IO), *exterior orientation* (EO), dan koordinat objek yang dihitung secara bersamaan dengan menggunakan teknik hitung kuadrat terkecil (Fraser, 1997).

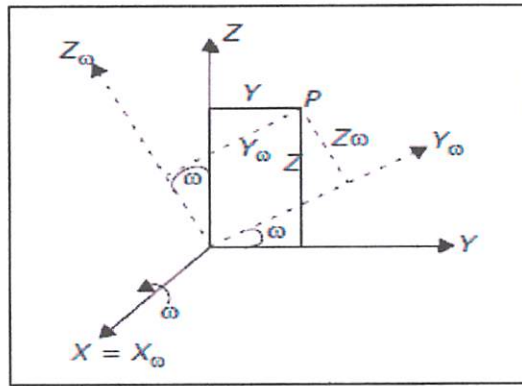
2.3.1 Persamaan Kolinear

Hubungan matematis antara titik konjugasi pada objek dengan titik konjugasi pada foto yang dijelaskan dengan transformasi projektif tiga dimensi disebut dengan persamaan kolinear. Disebut kolinear karena persamaan tersebut menunjukkan hubungan geometrik yang ideal antara sebuah titik objek, fotonya, dengan *imaging system perspective center*.

2.3.1.1 Matrik Rotasi

Rotasi matrik terdiri dari tiga parameter rotasi yaitu, omega (ω), phi (ϕ), dan kappa (κ). Dimana arah perputaran sumbu rotasi omega, phi dan kappa beserta sistem kaedah dan persamaannya dapat ditunjukkan pada gambar berikut :

1. Rotasi pertama yaitu omega (ω) dilakukan terhadap sumbu X_ω , dengan menggunakan sistem kaedah tangan kanan perputaran bernilai positif jika berlawanan arah jarum jam, seperti Gambar (2.3) (Geosystem, 2006):

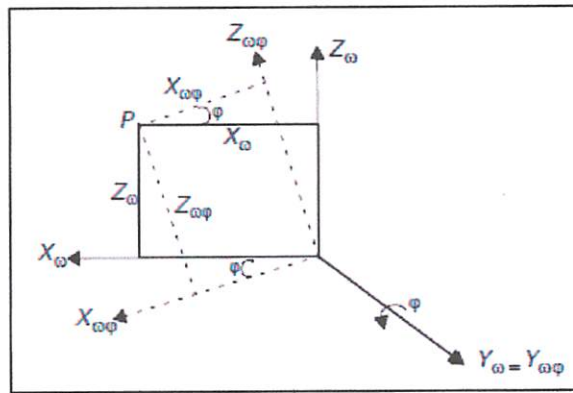


Gambar 2.3 Rotasi sudut omega terhadap sumbu X_ω

Dari Gambar (2.3) sistem rotasi ($X_\omega, Y_\omega, Z_\omega$) dapat diperoleh persamaan vektornya $[X_\omega Y_\omega Z_\omega]^T = R_\omega [XYZ]^T$, dimana (Cooper dan Robson, 2001) :

$$R_\omega = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \omega & \sin \omega \\ 0 & -\sin \omega & \cos \omega \end{bmatrix} \dots\dots\dots (2.7)$$

2. Rotasi kedua yaitu rotasi phi (φ) dilakukan terhadap sumbu Y_φ , dengan sistem kaedah tangan kanan perputaran bernilai positif jika berlawanan arah jarum jam seperti Gambar (2.4) (Geosystem, 2006).

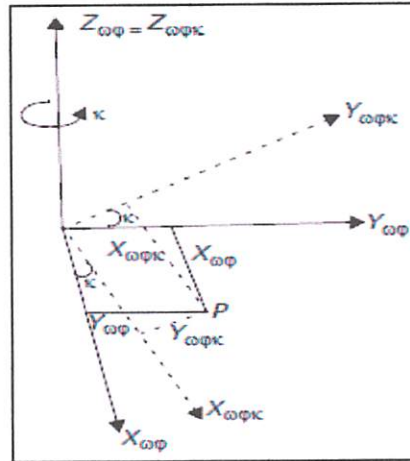


Gambar 2.4 Rotasi sudut phi terhadap sumbu Y_φ

Dari Gambar (2.4) sistem rotasi ($X_{\omega\varphi}, Y_{\omega\varphi}, Z_{\omega\varphi}$) dapat diperoleh vektornya $[X_{\omega\varphi} \ Y_{\omega\varphi} \ Z_{\omega\varphi}]^T = R_\varphi [X \ Y \ Z]^T$, dimana (Cooper dan Robson, 2001) :

$$R_\varphi = \begin{bmatrix} \cos \varphi & 0 & -\sin \varphi \\ 0 & 1 & 0 \\ \sin \varphi & 0 & \cos \varphi \end{bmatrix} \dots\dots\dots (2.8)$$

3. Rotasi ketiga yaitu rotasi kappa (κ) dilakukan terhadap sumbu $Z_{\omega\phi}$, dengan sistem kaedah tangan kanan perputaran bernilai positif jika berlawanan arah jarum jam seperti gambar dibawah ini (Geosystem, 2006).



Gambar 2.5 Rotasi sudut kappa terhadap sumbu $Z_{\omega\phi}$

Dari Gambar (2.5) sistem rotasi ($X_{\omega\phi\kappa}, Y_{\omega\phi\kappa}, Z_{\omega\phi\kappa}$) dapat diperoleh persamaan $[X_{\omega\phi\kappa} \ Y_{\omega\phi\kappa} \ Z_{\omega\phi\kappa}]^T = R_{\kappa} [X \ Y \ Z]^T$ dimana (Cooper dan Robson, 2001) :

$$R_{\kappa} = \begin{bmatrix} \cos \kappa & \sin \kappa & 0 \\ -\sin \kappa & \cos \kappa & 0 \\ 0 & 0 & 1 \end{bmatrix} \dots\dots\dots (2.9)$$

Dengan menggabungkan Persamaan (2.27, 2.28, 2.29) maka akan diperoleh persamaannya sebagai berikut (Mikhail et al., 2001):

$$R = R_{\omega} \times R_{\phi} \times R_{\kappa} \dots\dots\dots (2.10)$$

dimana R merupakan matrik 3x3, dengan nilai elemen masing-masing matrik sebagai berikut :

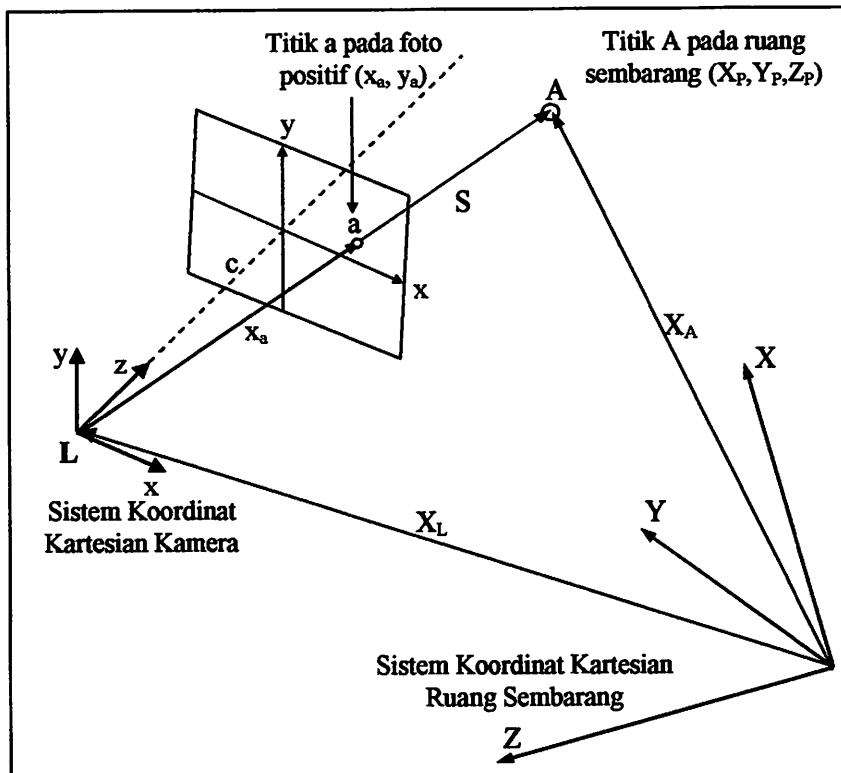
$$R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \dots\dots\dots (2.11)$$

dimana :

$$\begin{aligned} r_{11} &= \cos \varphi \cos \kappa \\ r_{12} &= \sin \omega \sin \varphi \cos \kappa + \cos \omega \sin \kappa \\ r_{13} &= -\cos \omega \sin \varphi \cos \kappa + \sin \omega \sin \kappa \\ r_{21} &= -\cos \varphi \sin \kappa \\ r_{22} &= -\sin \omega \sin \varphi \sin \kappa + \cos \omega \cos \kappa \dots\dots\dots (2.12) \\ r_{23} &= \cos \omega \sin \varphi \sin \kappa + \sin \omega \cos \kappa \\ r_{31} &= \sin \varphi \\ r_{32} &= -\sin \omega \cos \kappa \\ r_{33} &= \cos \omega \cos \varphi \end{aligned}$$

2.3.1.2 Transformasi Perspektif Pusat (*Central Perspective Transformation*)

Jika didalam kondisi yang ideal hubungan antara garis lurus Aa , sistem koordinat kamera dan sistem koordinat ruang dapat dijelaskan seperti Gambar (2.6), vektor S merupakan perpanjangan garis lurus Aa sebesar AL , dimana titik L berada pada bidang CCD/CMOS (Cooper dan Robson, 2001).



Gambar 2.6 Hubungan Keruangan antara titik A sistem koordinat sembarang dengan titik sistem koordinat kamera

Berdasarkan hubungan antara vektor S, X_J dan X_L pada gambar 2.6, vektor posisi titik A pada sistem koordinat ruang dapat ditulis sebagai berikut :

$$X_A = X_L + S \dots\dots\dots (2.13)$$

Didalam proses pemotretan berkas sinar dari A ke L merupakan garis lurus (Mikhail et al, 2001), maka vektor S pada sistem koordinat kamera haruslah dikonversikan ke dalam sistem koordinat ruang dan dikalikan faktor skala perbesaran (Cooper dan Robson, 2001) maka persamaannya menjadi :

$$X_A = X_L + \lambda R^T x_a \dots\dots\dots (2.14)$$

atau jika dinyatakan dengan notasi matrik akan setara dengan :

$$\begin{bmatrix} X_A \\ Y_A \\ Z_A \end{bmatrix} = \begin{bmatrix} X_L \\ Y_L \\ Z_L \end{bmatrix} + \lambda \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \begin{bmatrix} x \\ y \\ -c \end{bmatrix} \dots\dots\dots (2.15)$$

dimana λ adalah faktor skala, x adalah vektor komponen titik a di dalam sistem koordinat kamera; dan X adalah vektor komponen titik A didalam sistem koordinat ruang (X_A, Y_A, Z_A) adalah koordinat titik A dan sistem ruang (X_L, Y_L, Z_L) merupaka koordinat titik L (pusat perspektif kamera) pada sistem ruang. Sedangkan R merupakan matrik rotasi dengan sudut perputaran omega, phi, dan kappa ($\omega, \varphi, \text{ dan } \kappa$) (Mikhail et al., 2001).

Dari Persamaan (2.14) dan (2.15) adalah sistem koordinat dari sistem koordinat kamera ke sistem koordinat ruang. Jika arah transformasi dibalik menjadi sistem koordinat ruang ke sistem koordinat kamera, maka Persamaan (2.14) menjadi :

$$x_a = \lambda^{-1}R(X_A - XL) \dots \dots \dots (2.16)$$

atau jika dinyatakan dengan notasi matrik untuk titik sembarang akan setara dengan :

$$\begin{bmatrix} x - x_o \\ y - y_o \\ -c \end{bmatrix} = \lambda^{-1} \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \begin{bmatrix} X_A - X_L \\ Y_A - Y_L \\ Z_A - Z_L \end{bmatrix} \dots \dots \dots (2.17)$$

Dimana $(x, y, -c)$ adalah nilai koordinat pada sistem kamera dan (X_A, Y_A, Z_A) adalah koordinat sistem ruang, serta (x_o, y_o) adalah koordinat titik tengah foto (*principal point*), dimana *principal point* merupakan nilai titik tengah *pixel* pada bidang CCD/CMOS yang tidak selalu berimpit dengan titik tengah foto (Wolf dan Dewitt, 2000).

2.3.1.3 Kondisi Kolinear

Dari Persamaan (2.17), karena faktor skala tidak dibutuhkan didalam pembahasan selanjutnya maka faktor skala ini dieliminir dengan membagi elemen baris pertama dan kedua dibagi dengan elemen baris ketiga pada persamaan (2.15), maka persamaannya akan menjadi (Mikhail et al., 2001):

$$\begin{aligned} x_a &= x_o - c \left[\frac{r_{11}(X_A - X_L) + r_{12}(Y_A - Y_L) + r_{13}(Z_A - Z_L)}{r_{31}(X_A - X_L) + r_{32}(Y_A - Y_L) + r_{33}(Z_A - Z_L)} \right] \\ y_a &= y_o - c \left[\frac{r_{21}(X_A - X_L) + r_{22}(Y_A - Y_L) + r_{23}(Z_A - Z_L)}{r_{31}(X_A - X_L) + r_{32}(Y_A - Y_L) + r_{33}(Z_A - Z_L)} \right] \dots (2.18) \end{aligned}$$

Persamaan ini merupakan persamaan garis lurus (persamaan kolinier) antara titik AaL seperti Gambar (2.6). Didalam Persamaan (2.18), jika titik-titik koordinat pada foto digital diukur (ditentukan nilainya) dan disimbolkan dengan \underline{x} , maka yang menjadi parameter (*unknowns*) adalah

x_o dan y_o yang dikenal sebagai parameter orientasi dalam kamera; sedangkan parameter orientasi kamera (O) atau orientasi dan posisi kamera dikenal $\omega, \varphi, \kappa, X_L, Y_L, Z_L$; serta X, Y, Z yang merupakan titik-titik koordinat objek yang terekam pada saat pemotretan yang disimbolkan sebagai \underline{X} (Wolf dan Dewitt, 2000).

Jika penulisan Persamaan (2.18) disederhanakan menjadi (Fraser, 1997) :

$$f(\underline{x}, \underline{O}, \underline{X})_{i,j} = 0 \dots\dots\dots (2.19)$$

Keterangan :

- i, j = foto yang ke-i dan titik objek yang ke-j yang terekam oleh kamera.
- \underline{x} = vektor yang berisikan nilai-nilai koordinat kamera untuk titik-titik obyek pada foto ke-i.
- \underline{O} = parameter orientasi luar kamera ($\omega_i, \varphi_i, \kappa_i, X_{L_i}, Y_{L_i}, Z_{L_i}$) untuk foto ke-i
- \underline{X} = Koordinat ruang untuk titik yang ke-j (X_j, Y_j, Z_j)

2.3.2 Persamaan Observasi

Metode gabungan kuadrat terkecil mensyaratkan agar persamaan dituliskan untuk masing-masing observasi yang secara fisik dibuat dan untuk masing-masing parameter yang digunakan dalam model fungsional. Persamaan ini disebut dengan persamaan observasi. Persamaan observasi dikembangkan dalam tiga kelompok variabel, yaitu (King, 1993) :

1. *The plate coordinates*
2. The camera's exterior orientation parameters
3. The object point coordinates

2.3.3 Matrik Bobot Observasi

Salah satu keuntungan dari penerapan metode gabungan kuadrat terkecil adalah memungkinkan untuk menggabungkan parameter dan observasi yang digunakan dalam proses penyesuaian dengan mempertimbangkan bobot yang sesuai. Parameter atau observasi yang nilainya reliabel dapat digabungkan dengan bobot yang tinggi dengan sebaliknya nilai yang tidak reliabel digabungkan dengan bobot yang rendah.

Bobot dari sebuah observasi bernilai proporsional dengan varian observasi (σ^2). Varian utama untuk tiap bobot dilambangkan dengan (σ_0^2).

Bobot dari observasi dinyatakan sebagai berikut (King, 1993) :

$$W = \sigma_0^2 / \sigma^2 \dots\dots\dots (2.20)$$

Matrik bobot untuk *plate coordinates observation* dalam *i* pada foto *j* sebagai berikut (Fraser, 1997) :

$$W_{ij} = \begin{pmatrix} 1/\sigma_x^2 & 0 \\ 0 & 1/\sigma_y^2 \end{pmatrix}_{ij} \dots\dots\dots (2.21)$$

Pembahasan mengenai matrik bobot untuk kamera dan objek sebagai berikut (King, 1993).

Matrik bobot diagonal untuk kamera j sebagai berikut :

$$W_j = \begin{bmatrix} 1/\sigma_b^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1/\sigma_\phi^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1/\sigma_\kappa^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1/\sigma_{XL}^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1/\sigma_{YL}^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1/\sigma_{ZL}^2 \end{bmatrix} \dots\dots (2.22)$$

untuk keseluruhan n foto sebagai berikut :

$$W = \begin{bmatrix} W_1 & & & & & \\ & \ddots & & & & \\ & & W_j & & & \\ & & & \ddots & & \\ & & & & & W_n \end{bmatrix} \dots\dots\dots (2.23)$$

Matrik diagonal untuk titik objek i sebagai berikut :

$$W = \begin{bmatrix} 1/\sigma_x^2 & 0 & 0 \\ 0 & 1/\sigma_y^2 & 0 \\ 0 & 0 & 1/\sigma_z^2 \end{bmatrix} \dots\dots\dots (2.24)$$

dan untuk keseluruhan m titik objek sebagai berikut :

$$W = \begin{bmatrix} W_1 & & & & & \\ & \ddots & & & & \\ & & W_j & & & \\ & & & \ddots & & \\ & & & & & W_m \end{bmatrix} \dots\dots\dots (2.25)$$

2.3.4 Model Matematika

Untuk m titik didalam n foto dengan mengacu pada persamaan observasi digabungkan untuk membentuk persamaan pengamatan sebagai berikut (King, 1993) :

$$V + B\Delta = f \dots\dots\dots (2.26)$$

Dimana Δ merupakan matrik koreksi parameter yang dicari, V adalah matrik residu, B merupakan matrik koefisien dan f yaitu matrik observasi. Jika sekumpulan persamaan dikombinasikan dengan persamaan matrik bobotnya (W), solusi kuadrat terkecilnya sebagai berikut (King, 1993) :

$$\begin{aligned} \Delta &= (B^T W B)^{-1} B^T W f \dots\dots\dots (2.27) \\ &= N^{-1} t \end{aligned}$$

2.3.5 Linierisasi Persamaan Kolinier

Persamaan dasar dari *bundle adjustment* adalah persamaan kolinier, yang mendiskripsikan satuan dasar dari *photogrammetry*. Dengan menggunakan persamaan kolinier membuat bentuk persamaan dan proses penyelesaiannya akan lebih efisien (Mikhail et al., 2001). Bentuk linier dari persamaan kolinier untuk foto i dan titik j . Karena persamaan pengamatan garis lurus pada Persamaan (2.19) tidak linier, maka untuk menghitung parameter didalamnya harus dilinierkan terlebih dahulu dengan menggunakan deret Taylor sebagai berikut (Stewart, 1999) :

$$\left(\frac{df}{d\underline{x}} \cdot d\underline{x} \right)_{ij} + \left(\frac{df}{d\underline{Q}} \cdot d\underline{Q} \right)_{ij} + \left(\frac{df}{d\underline{X}} \cdot d\underline{X} \right)_{ij} + f(\underline{x}, \underline{Q}^0, \underline{X}^0)_{ij} = 0 \dots\dots\dots (2.28)$$

Bentuk linier dari persamaan pada Persamaan (2.28) ini setara dengan persamaan hitung kuadrat terkecil (*least square observation*) menurut Mikhail et al. (2001) yaitu :

$$\underset{(2,1)}{v} + \underset{(2,6)(6,1)}{B_1} \delta_1 + \underset{(2,3)(3,1)}{B_2} \delta_2 = \underset{(2,1)}{f} \dots\dots\dots (2.29)$$

dengan :

$$O = O^0 + \delta_1, \quad X = X^0 + \delta_2 \dots\dots\dots (2.30)$$

Keterangan :

$$v = \begin{bmatrix} v_x \\ v_y \end{bmatrix}_{ij}^T = \text{Residu dari koordinat foto}$$

$$f = f(\underline{x}, \underline{O}^0, \underline{X}^0)_{ij} = \text{Observasi}$$

$$\delta_1 = [d\omega_i, d\phi_i, d\kappa_i, dXL_i, dYL_i, dZL_i]^T = \text{Koreksi parameter EO 6m vektor)}$$

$$\delta_2 = [dX_j, dY_j, dZ_j]^T = \text{Koreksi Koordinat koreksi XYZ (3n vektor)}$$

2.3.6 Desain Matrik untuk *Bundle Adjustment*

Menurut Mikhail et al. (2001) dan Wolf dan Dewitt (2000), jika sebuah titik objek (objek ke-j) terekam pada sebuah foto (foto ke-i), maka dimensi-dimensi matrik didalam Persamaan (2.29) adalah sebagai berikut :

$$\underset{(2,2)}{B_{ij}} = \begin{pmatrix} \frac{\partial f_x}{\partial x} & \frac{\partial f_x}{\partial y} \\ \frac{\partial f_y}{\partial x} & \frac{\partial f_y}{\partial y} \end{pmatrix}_{ij} \dots\dots\dots (2.31)$$

$$\underset{(2,1)}{v_{ij}} = \begin{pmatrix} v_x \\ v_y \end{pmatrix}_{ij} \dots\dots\dots (2.32)$$

Susunan matrik B_i sebagai berikut (Fraser, 1997):

$$B_{1ij} = \begin{bmatrix} \left(\frac{\partial f_x}{\partial \omega}\right)_0 & \left(\frac{\partial f_x}{\partial \varphi}\right)_0 & \left(\frac{\partial f_x}{\partial \kappa}\right)_0 & \left(\frac{\partial f_x}{\partial X_L}\right)_0 & \left(\frac{\partial f_x}{\partial Y_L}\right)_0 & \left(\frac{\partial f_x}{\partial Z_L}\right)_0 \\ \left(\frac{\partial f_y}{\partial \omega}\right)_0 & \left(\frac{\partial f_y}{\partial \varphi}\right)_0 & \left(\frac{\partial f_y}{\partial \kappa}\right)_0 & \left(\frac{\partial f_y}{\partial X_L}\right)_0 & \left(\frac{\partial f_y}{\partial Y_L}\right)_0 & \left(\frac{\partial f_y}{\partial Z_L}\right)_0 \end{bmatrix}_{ij} \dots\dots\dots (2.33)$$

atau secara sederhana sebagai berikut (Wolf dan Dewitt, 2000).

$$B_{1ij} = \begin{bmatrix} (b_{11})_0 & (b_{12})_0 & (b_{13})_0 & (-b_{14})_0 & (-b_{15})_0 & (-b_{16})_0 \\ (b_{21})_0 & (a_{22})_0 & (b_{23})_0 & (-b_{24})_0 & (-b_{25})_0 & (-b_{26})_0 \end{bmatrix}_{ij} \dots\dots\dots (2.34)$$

Susunan matrik B_2 sebagai berikut (Fraser, 1997) :

$$B_{2ij} = \begin{bmatrix} \left(\frac{\partial f_x}{\partial X_i}\right)_0 & \left(\frac{\partial f_x}{\partial X_i}\right)_0 & \left(\frac{\partial f_x}{\partial X_i}\right)_0 \\ \left(\frac{\partial f_x}{\partial X_i}\right)_0 & \left(\frac{\partial f_x}{\partial X_i}\right)_0 & \left(\frac{\partial f_x}{\partial X_i}\right)_0 \end{bmatrix} \dots\dots\dots (2.35)$$

atau (Wolf dan Dewitt, 2000)

$$B_{2ij} = \begin{bmatrix} (b_{14})_0 & (b_{15})_0 & (b_{16})_0 \\ (b_{24})_0 & (b_{25})_0 & (b_{26})_0 \end{bmatrix}_{ij} \dots\dots\dots (2.36)$$

B_1 mempunyai dimensi 2x6 sedangkan B_2 mempunyai dimensi 2x3

dimana untuk tiap titik objek ke-j yang terekam pada foto ke-i.

Susunan matrik f sebagai berikut (Fraser, 1997) :

$$f_{ij} = \begin{bmatrix} J \\ K \end{bmatrix}_{ij} \dots\dots\dots (2.37)$$

Penjelasan elemen matrik B dan matrik f (Wolf and Dewitt, 2000):

Penjelasan penjabaran dari matrik B dapat dilihat pada Lampiran (1).

$$b_{11} = \frac{f}{q^2} [r(-m_{33}\Delta Y + m_{32}\Delta Z) - q(-m_{13}\Delta Y + m_{12}\Delta Z)]$$

$$b_{12} = \frac{f}{q^2} [r(\cos\phi\Delta X + \sin\omega\sin\phi\Delta Y - \cos\omega\sin\phi\Delta Z) - q(-\sin\phi\cos\kappa\Delta X + \sin\omega\cos\phi\cos\kappa\Delta Y - \cos\omega\cos\phi\cos\kappa\Delta Z)]$$

$$b_{13} = \frac{-f}{q} (m_{21}\Delta X + m_{22}\Delta Y + m_{23}\Delta Z)$$

$$b_{14} = \frac{f}{q^2} (rm_{31} - qm_{11})$$

$$b_{15} = \frac{f}{q^2} (rm_{32} - qm_{12})$$

$$b_{16} = \frac{f}{q^2} (rm_{33} - qm_{13})$$

$$b_{21} = \frac{f}{q^2} [s(-m_{33}\Delta Y + m_{32}\Delta Z) - q(-m_{23}\Delta Y + m_{22}\Delta Z)]$$

$$b_{22} = \frac{f}{q^2} [s(\cos\phi\Delta X + \sin\omega\sin\phi\Delta Y - \cos\omega\sin\phi\Delta Z) - q(-\sin\phi\cos\kappa\Delta X - \sin\omega\cos\phi\sin\kappa\Delta Y + \cos\omega\cos\phi\sin\kappa\Delta Z)]$$

$$b_{23} = \frac{f}{q} (m_{11}\Delta X + m_{12}\Delta Y + m_{13}\Delta Z)$$

$$b_{24} = \frac{f}{q^2} (sm_{31} - qm_{21})$$

$$b_{25} = \frac{f}{q^2} (sm_{32} - qm_{22})$$

$$b_{26} = \frac{f}{q^2} (sm_{33} - qm_{23})$$

$$\begin{aligned}
 J &= x_a - x_o + f \frac{r}{q} \\
 K &= y_a - y_o + f \frac{s}{q} \\
 &\dots\dots\dots (2.38)
 \end{aligned}$$

Untuk matrik δ_{1i} dan δ_{2j} struktur matriknya sebagai berikut (Mikhail et al. 2001) :

$$\delta_{1i}^{(6m,1)} = \begin{bmatrix} \delta\omega \\ \delta\varphi \\ \delta\kappa \\ \delta X_L \\ \delta Y_L \\ \delta Z_L \end{bmatrix}_i \dots\dots\dots (2.39)$$

$$\delta_{2j}^{(3n,1)} = \begin{bmatrix} \delta X \\ \delta Y \\ \delta Z \end{bmatrix}_j \dots\dots\dots (2.40)$$

Dimana δ_1 besar dimensinya $6m \times 1$ dan δ_2 besar dimensinya $3n \times 1$

2.3.7 Struktur dari Matrik Persamaan Normal

Jika kita sekarang akan membentuk persamaan normal yang berhubungan dengan sebuah foto untuk satu titik, maka akan diperoleh persamaan sebagai berikut (Mikhail et al., 2001) :

$$\begin{bmatrix} B_{1ij}^T W_{ij} B_{1ij} & B_{1ij}^T W_{ij} B_{2ij} \\ B_{2ij}^T W_{ij} B_{1ij} & B_{2ij}^T W_{ij} B_{2ij} \end{bmatrix} \begin{bmatrix} \delta_{1i} \\ \delta_{2j} \end{bmatrix} = \begin{bmatrix} B_{1ij}^T W_{ij} f_{ij} \\ B_{2ij}^T W_{ij} f_{ij} \end{bmatrix} \dots\dots\dots (2.41)$$

dimana W_{ij} merupakan matrik bobot 2×2 (*inverse* dari matrik *covarian* dari ketelitian pengukuran foto digital) yang berkaitan dengan koordinat foto

dari titik j pada foto i . Persamaan tersebut juga dapat ditulis sebagai berikut

:

$$\begin{bmatrix} \dot{N}_i & \bar{N}_{ij} \\ \bar{N}_{ij}^T & \dot{N}_j \end{bmatrix} \begin{bmatrix} \dot{\delta}_i \\ \dot{\delta}_j \end{bmatrix} = \begin{bmatrix} \dot{i}_i \\ \dot{i}_j \end{bmatrix} \quad \text{atau } N\delta = t \dots\dots\dots (2.42)$$

Perlu diperhatikan bahwa \dot{N} mengandung koefisien dari parameter foto, \dot{N} mengandung koefisien parameter titik (koordinat), \bar{N} mengacu pada keduanya.

Kita sekarang membuat asumsi penting bahwa kesalahan dari tiap-tiap titik pada koordinat foto tidak berhubungan dengan kesalahan pada koordinat foto lainnya, baik titik-titik pada foto yang sama maupun foto yang berbeda untuk titik yang sama, meskipun koordinat x dan y pada suatu foto mungkin saja memiliki hubungan.

Bundle adjustment pada umumnya diterapkan dengan menggunakan gabungan kuadrat terkecil, yang tidak mengubah bentuk dasar algoritma secara signifikan. Matrik bobot dari observasi parameter untuk setiap foto dan titik. W_i dan W_j ditambahkan pada masing-masing N_i dan N_j (Mikhail et al., 2001).

Dari pengembangan persamaan kolinier (2.28), maka akan dapat dibentuk persamaan yang elemen di dalamnya didefinisikan sebagai berikut

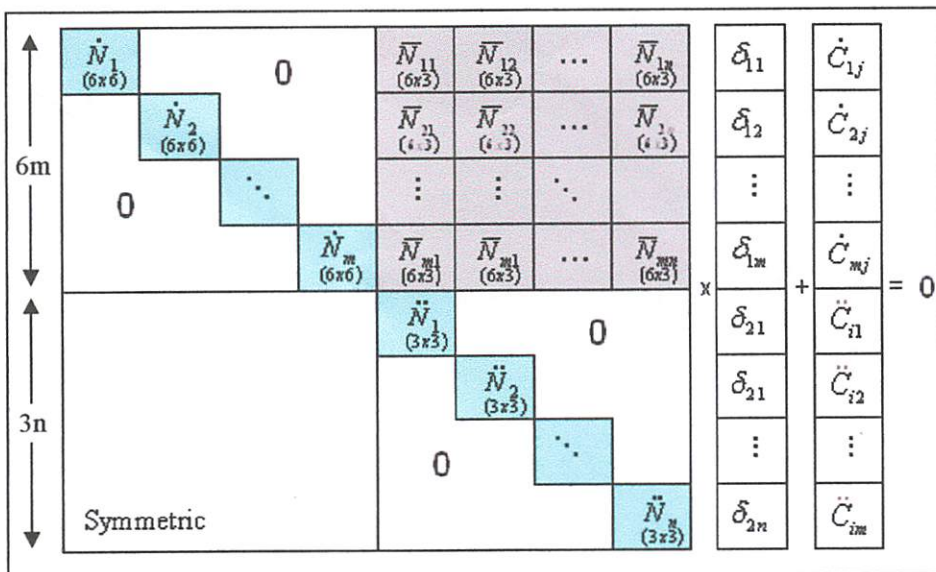
$$\begin{aligned} \dot{N}_{ij} &= \sum_{j=1}^n B_{1ij}^T W_{ij} B_{1ij} ; \quad \dot{i}_i = \sum_{j=1}^n B_{1ij}^T W_{ij} f_{ij} \\ \dot{N}_{ij} &= \sum_{i=1}^m B_{2ij}^T W_{ij} B_{2ij} ; \quad \dot{i}_j = \sum_{i=1}^m B_{2ij}^T W_{ij} f_{ij} \dots\dots\dots (2.43) \\ \bar{N}_{ij} &= B_{1ij}^T W_{ij} B_{2ij} \end{aligned}$$

Dimana \dot{N} dan \ddot{N} adalah sub matrik dari matrik blok-diagonal, dimana blok \dot{N} mengacu pada parameter EO dan \ddot{N} mengacu pada koordinat titik-titik obyek serta \bar{N} mengacu pada keduanya.

Penyusunan matrik N dapat disajikan sebagai berikut dengan m banyak foto dan n banyak titik obyek :

$$N_{(6m+3n) \times (6m+3n)} = \begin{pmatrix} \dot{N}_{1j} & 0 & \dots & 0 & \bar{N}_{11} & \bar{N}_{12} & \dots & \bar{N}_{1n} \\ 0 & \dot{N}_{2j} & \dots & 0 & \bar{N}_{21} & \bar{N}_{22} & \dots & \bar{N}_{2n} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \dot{N}_{mj} & \bar{N}_{m1} & \bar{N}_{m2} & \dots & \bar{N}_{mn} \\ \hline \bar{N}_{11}^T & \bar{N}_{21}^T & \dots & \bar{N}_{m1}^T & \ddot{N}_{i1} & 0 & \dots & 0 \\ \bar{N}_{12}^T & \bar{N}_{22}^T & \dots & \bar{N}_{m2}^T & 0 & \ddot{N}_{i1} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \bar{N}_{1n}^T & \bar{N}_{2n}^T & \dots & \bar{N}_{mn}^T & 0 & 0 & \dots & \ddot{N}_{in} \end{pmatrix} \quad (2.44)$$

Struktur persamaan normal *bundle adjustment* juga dapat disusun seperti Gambar (2.7) sebagai berikut :



Gambar 2.7 Struktur Persamaan Normal

dimana :

$$\underset{(6m+3n),1}{\delta} = \begin{bmatrix} \delta_{11} \\ \delta_{12} \\ \vdots \\ \delta_{1m} \\ \delta_{21} \\ \delta_{21} \\ \vdots \\ \delta_{2n} \end{bmatrix}; \quad \underset{(6m+3n),1}{C} = \begin{bmatrix} \sum_{j=1}^n \dot{C}_{1j} \\ \sum_{j=1}^n \dot{C}_{2j} \\ \vdots \\ \sum_{j=1}^n \dot{C}_{mj} \\ \sum_{i=1}^m \ddot{C}_{i1} \\ \sum_{i=1}^m \ddot{C}_{i2} \\ \vdots \\ \sum_{i=1}^m \ddot{C}_{im} \end{bmatrix} \dots\dots\dots (2.45 \text{ dan } 2.46)$$

Persamaan diatas adalah teknik *Bundle Adjustment Normal Equation* untuk mendapatkan nilai parameter dan koordinat titik objek didalam sistem kartesian 3D.

2.3.8 Penyelesaian Persamaan Normal *Bundle Adjustment*

Untuk penyelesaian persamaan normal diatas dapat diselesaikan sebagai berikut (Fraser, 1997) :

$$\begin{bmatrix} \delta_1 \\ \delta_2 \end{bmatrix} = - \begin{bmatrix} A_1^T P A_1 & A_1^T P A_2 \\ A_2^T P A_1 & A_2^T P A_2 \end{bmatrix}^{-1} \begin{bmatrix} A_1^T P w \\ A_2^T P w \end{bmatrix} \dots\dots\dots (2.47)$$

Penyelesaian dari teknik *Bundle Adjustment* bertujuan untuk mendapatkan nilai parameter EO dan koordinat titik objek didalam sistem kartesian 3D.

$$O = O^0 + \delta_1 \quad : \text{(Parameter orientasi luar (EO) yang terkoreksi)}$$

$$X = X^0 + \delta_2 \quad : \text{(Koordinat titik objek yang terkoreksi)}$$

Jika titik-titik objek ini hendak dihitung dengan tingkat keakuratan yang lebih tinggi lagi, maka kesalahan sistematis kamera harus dimodelkan

2.4 Model Penilaian *Bundle Adjustment*

Kinerja dari BA dinilai dengan beberapa kriteria, antara lain tingkat akurasi dan presisinya. Penilaian keakuratan *bundle adjustment* pada umumnya dilakukan dengan cara membandingkan koordinat titik objek yang dihitung dari *bundle adjustment* dengan nilai yang diperoleh dari hasil pengamatan. Indikator umum untuk penilaian keakuratan adalah *RMS error* dari koordinat titik objek. *RMS error* dari koordinat titik objek untuk masing-masing sumbu koordinat sebagai berikut (King, 1993) :

$$RMS_x = \sqrt{\frac{1}{n} \sum_{i=1}^n e_{x_i}^2} \dots\dots\dots (2.48)$$

$$RMS_y = \sqrt{\frac{1}{n} \sum_{i=1}^n e_{y_i}^2} \dots\dots\dots (2.49)$$

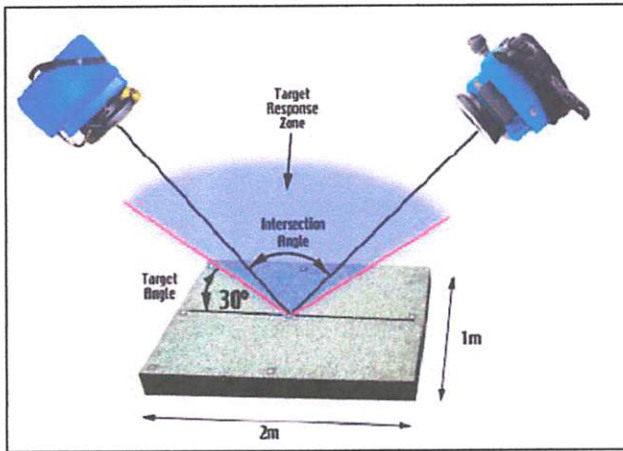
$$RMS_z = \sqrt{\frac{1}{n} \sum_{i=1}^n e_{z_i}^2} \dots\dots\dots (2.50)$$

2.5 *Network Design*

Pekerjaan *Close Range Photogrammetry* tidak jauh dari prose *bundle adjustment*, dengan titik-titik objek dan posisi kamera. Solusi ini mencangkup parameter orientasi luar kamera baik itu parameter tambahan untuk memodelkan efek residual dari kesalahan sistematis.

Untuk akurasi maksimum, geometri pemotretan harus di desain secara hati-hati dan hasilnya diuji sesuai prosedur untuk menjamin hasilnya sesuai dengan spesifikasi yang diinginkan (Mikhail et al, 2001).

Pada umumnya tingkat presisi merupakan sebuah fungsi dari seberapa besar tingkat arah dari sinar yang dipancarkan bias diukur dan kekuatan geometri *intersection* dari sinar yang dipantulkan, dimana keadaan tersebut ditentukan oleh dua faktor antara lain jumlah titik pada tiap-tiap objek dan sudut antara titik-titik tersebut.



Gambar 2.8 Geometri Pemotretan

Sehingga dalam proses perencanaan ini akan melibatkan beberapa tahap yang harus dilaksanakan untuk memaksimalkan tingkat akurasi yang dihasilkan pada proses *close range photogrammetry* antara lain :

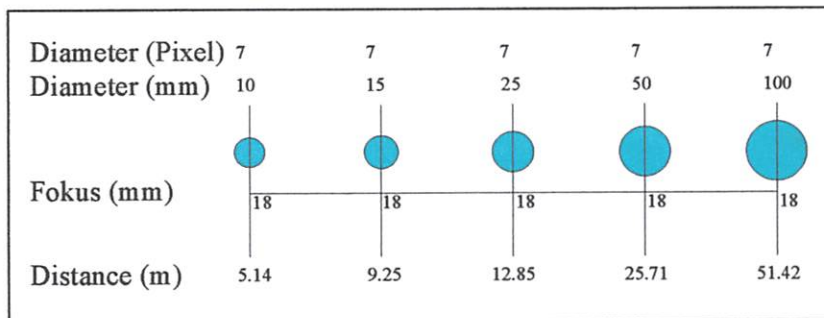
2.5.1 _Desain Target

Pada proses pendesainan target yang perlu di perhatikan antara adalah panjang fokus yang akan di gunakan pada saat melakukan pemotretan, jarak abjek terhadap posisi pengambilan gambar, dan jumlah piksel minimum yang akan terekam pada tiap-tiap target yang akan dipasang. Model persamaan yang digunakan dalam penentuan target ini merupakan persamaan perbandingan yang diberikan sebagai berikut (Manual Photomodeller):

$$r = \left(\frac{p \times fw \times d}{f \times pw} \right) \dots\dots\dots (2.51)$$

dimana f merupakan panjang fokus kamera, p merupakan jumlah piksel minimum target, d merupakan jarak kamera ke objek, f_w lebar CCD kamera, p_w merupakan jumlah horizontal peksel foto dan r merupakan diameter target.

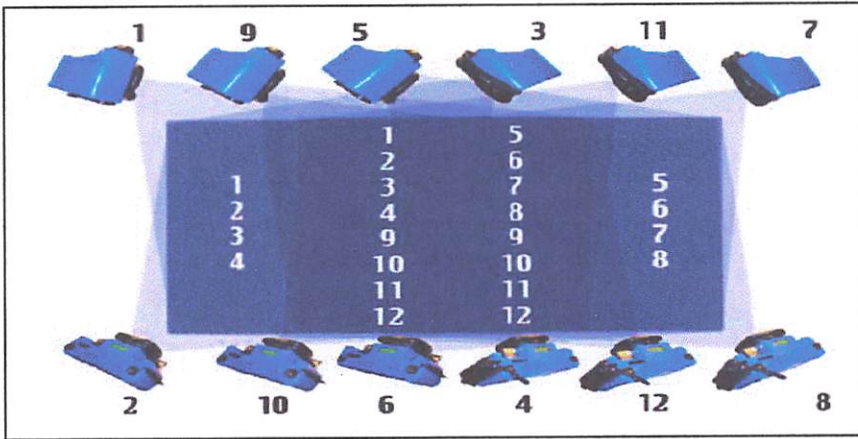
Untuk lebih jelasnya desain target akan diberikan sesuai dengan diagram sebagai berikut dengan jumlah minimum *pixel* sebesar 7 *pixel* dengan panjang fokus dari kamera Nikon D60 sebesar 18 mm (Manual Photomodeller).



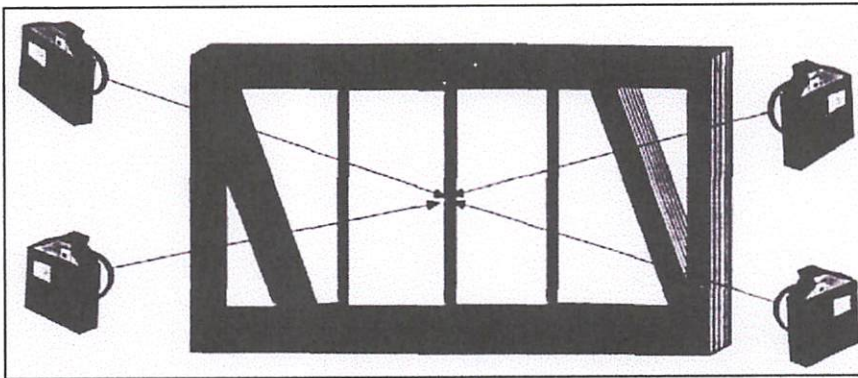
Gambar 2.9 Diagram Desain Target

2.5.2 Desain Pemotretan

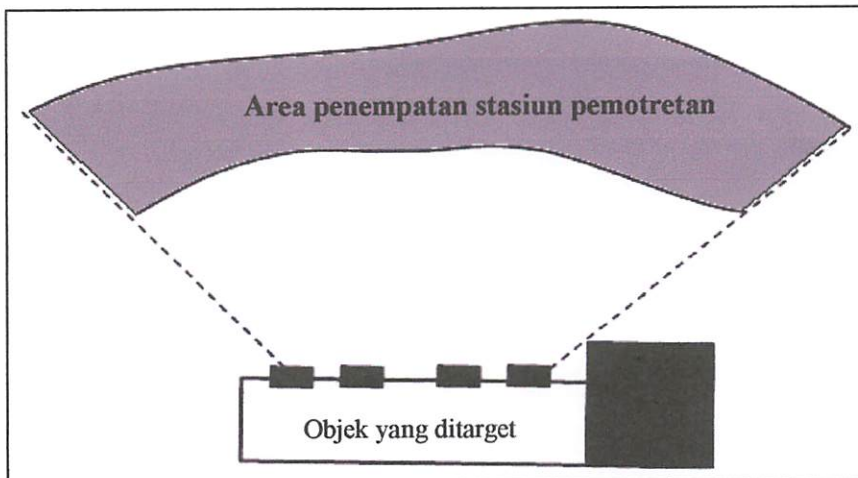
Dalam mendesain pemotretan ada beberapa hal yang perlu diperhatikan antara lain adalah jumlah pertampalan masing-masing objek, dan sudut yang terbentuk antara posisi kamera. Sehingga dalam kasus mendesai pemotretan seperti *Design Left-Right Measurement* dapat dilakukan seperti gambar dibawah ini (V-Stars, 2005) :



Gambar 2.10 Design Left-Right Measurements



Gambar 2.11 Design Jaringan Empat Station (Fraser, 1997)



Gambar 2.12 Ilustrasi penempatan stasiun pemotretan (Fraser, 1997)

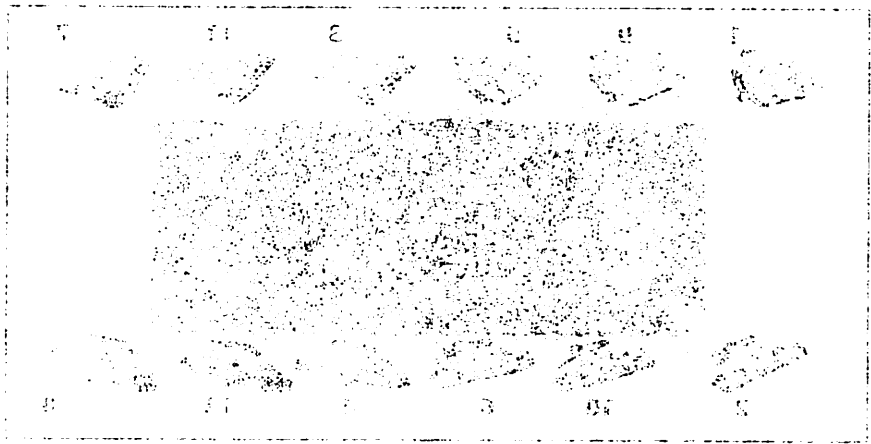


Figure 1. A circular diagram with a shaded area and six segments labeled A through F.

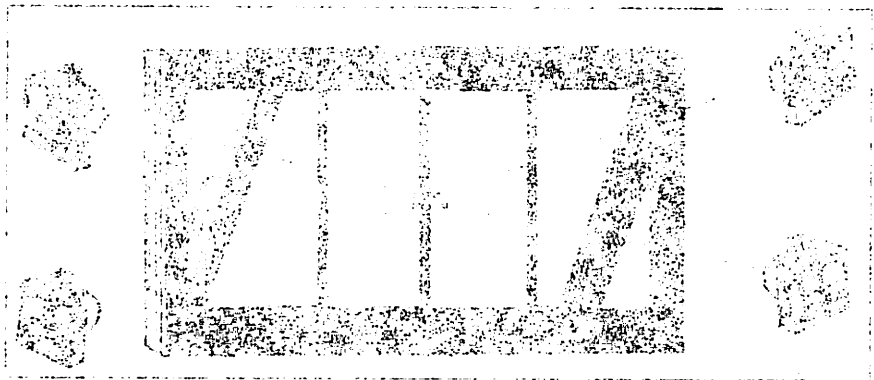


Figure 2. A diagram showing a central rectangular area and four surrounding circular segments labeled A through D.

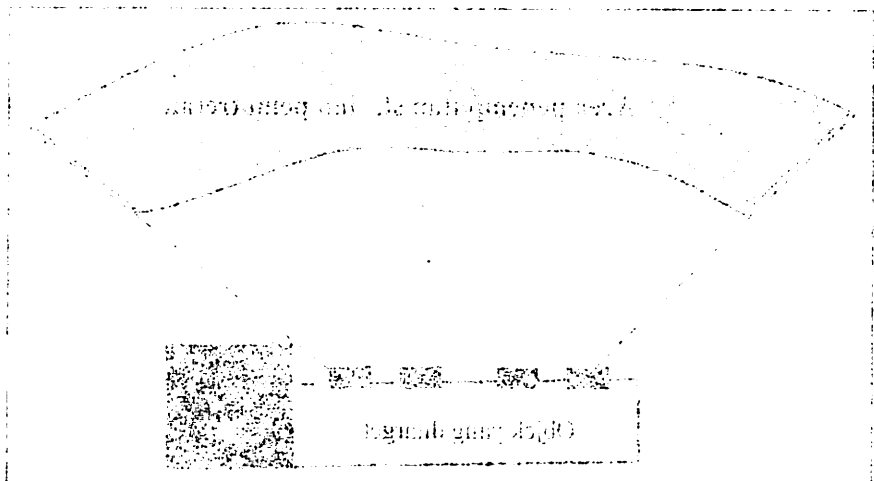


Figure 3. A diagram showing a curved shape and a smaller rectangular area containing the text 'Figure 3'.

2.5.3 Permasalahan Datum

Parameter datum sering disebut dengan *zero-variance computational base*. Disini muncul permasalahan dimana diketahui bahwa nilai C_2 akan berbeda sesuai dengan koordinat kontrol minimal yang dipilih, tetapi tidak diketahui koordinat kontrol minimal mana yang dapat menghasilkan C_2 terbaik.

$$\bar{\sigma}_e^2 = \frac{\sigma_0^2}{3n} \text{trace} Cx_2 = \text{a minimum} \dots\dots\dots (2.52)$$

$$\begin{bmatrix} C_1 & \cdot & \cdot \\ \cdot & C_2 & \cdot \\ \cdot & \cdot & C_3 \end{bmatrix} = \sigma_0^2 \begin{bmatrix} B_1^T W B_1 & B_1^T W B_2 & 0 \\ B_2^T W B_1 & B_2^T W B_2 & G \\ 0 & G^T & 0 \end{bmatrix}^{-1} \dots\dots\dots (2.53)$$

dimana transformasi matrik G sebagai berikut (Freser, 1997) :

$$G_j = \begin{bmatrix} 1 & 0 & 0 & 0 & Z_j & -Y_j & X_j \\ 0 & 1 & 0 & -Z_j & 0 & X_j & Y_j \\ 0 & 0 & 1 & Y_j & -X_j & 0 & Z_j \end{bmatrix} \dots\dots\dots (2.54)$$

dimana parameter matrik covariannya sebagai berikut :

$$C_\delta = \sigma_0^2 (A^T W A)^{-1} \text{ atau } \sigma_0^2 (A^T W A + \bar{P})^{-1} \dots\dots\dots (2.55)$$

Matrik covarian untuk koordinat titik oobjek sebagai berikut :

$$C_{xyz} = C_{\delta 2} = \begin{bmatrix} \sigma_{x_1^2} & \dots & \dots & \dots & \dots & \dots \\ \vdots & \sigma_{y_1^2} & & & & \vdots \\ \vdots & & \sigma_{z_1^2} & & & \vdots \\ \vdots & & & \ddots & & \vdots \\ \vdots & & & & \ddots & \vdots \\ \vdots & \dots & \dots & \dots & \dots & \sigma_{z_n^2} \end{bmatrix} \dots\dots\dots (2.56)$$

Presisi dari titik objek rata-rata dapat dicari menggunakan persamaan sebagai berikut :

$$\bar{\sigma}_c^2 = \text{trace } C_{xyz} / 3n \dots\dots\dots (2.57)$$

Matrik kovarian C_δ atau C_X diperoleh dari *bundle adjustment minimal constraint* tergantung dari pemilihan tujuh (7) parameter tetapnya. Dimana \bar{P} hanya berhubungan dengan dengan tujuh (7) parameter datum.

2.6. Pembuatan Algoritma untuk Program Perhitungan *Bundle Adjustment*

Definisi algoritma secara umum adalah urutan langkah logis tertentu untuk memecahkan suatu masalah. Yang ditekankan adalah urutan langkah logis, yang berarti algoritma harus mengikuti suatu urutan tertentu, tidak boleh melompat-lompat. (*Microsoft Press Computer and Internet Dictionary 1997, 1998*)

Dalam bidang komputer, algoritma sangat diperlukan dalam menyelesaikan berbagai masalah pemrograman, terutama dalam komputasi numeris. Tanpa algoritma yang dirancang baik maka proses pemrograman akan menjadi salah, rusak, atau lambat dan tidak efisien.

Berikut adalah Algoritma yang digunakan untuk menyusun program perhitungan *bundle adjustment* :

1. Input :
 - Parameter Orientasi Luar ($\omega_i, \phi_i, \kappa_i, XL_i, YL_i, ZL_i$) pendekatan
 - Koordinat *Object Space Point* (X, Y, Z) Pendekatan
 - Koordinat foto (x,y) dan ketelitian (σ_x, σ_y) serta Parameter *Interior Orientation* (IO)
2. Proses
 - Menghitung Matrik Rotasi (R)
 - Menghitung Parameter q, r, s

- Menghitung Matrik $B1, B2, f, w$
- Menghitung Matrik $\dot{N}_{ij}, \ddot{N}_{ij}, \bar{N}_{ij}, \dot{C}_{ij}, \ddot{C}_{ij}$
- Menyusun Matrik N dan t
- Menghitung matrik Koreksi ($\delta 1$ dan $\delta 2$)
- Menghitung Nilai Residu (v)
- Menghitung Nilai Akhir
- Menghitung Ketelitian Varian-Kovarian

3. Output

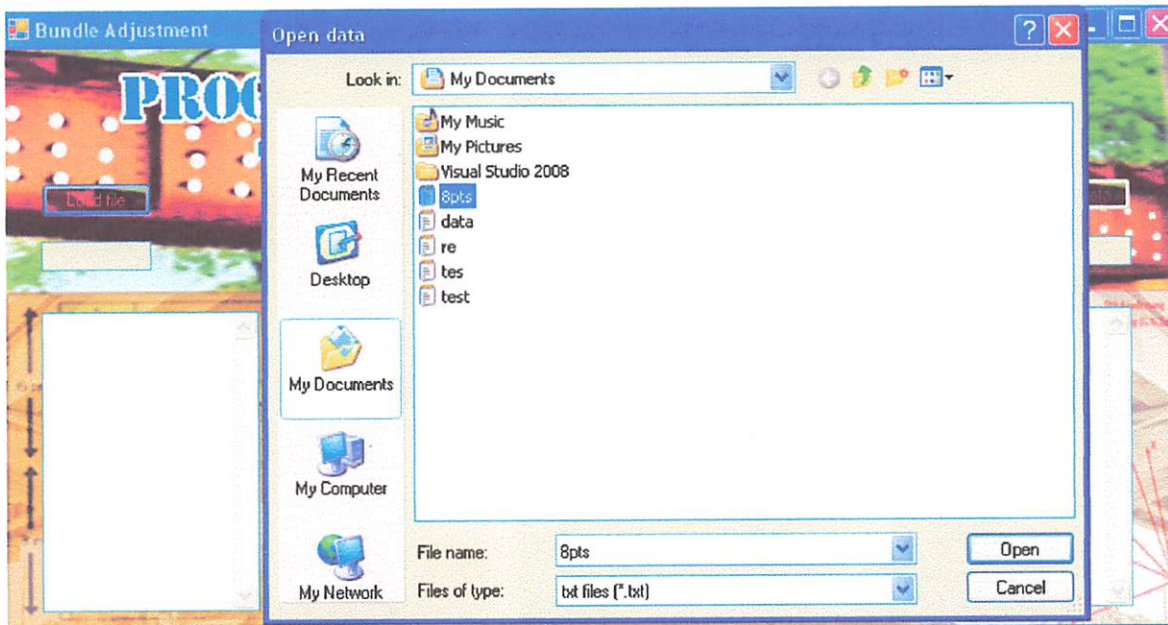
- Parameter Orientasi Luar ($\omega_i, \varphi_i, \kappa_i, XL_i, YL_i, ZL_i$) final
- Koordinat *Object Space Point* (X, Y, Z) final
- Ketelitian *Varian-Kovarian*

2.7. Pembuatan *Interface* didalam C# 2008

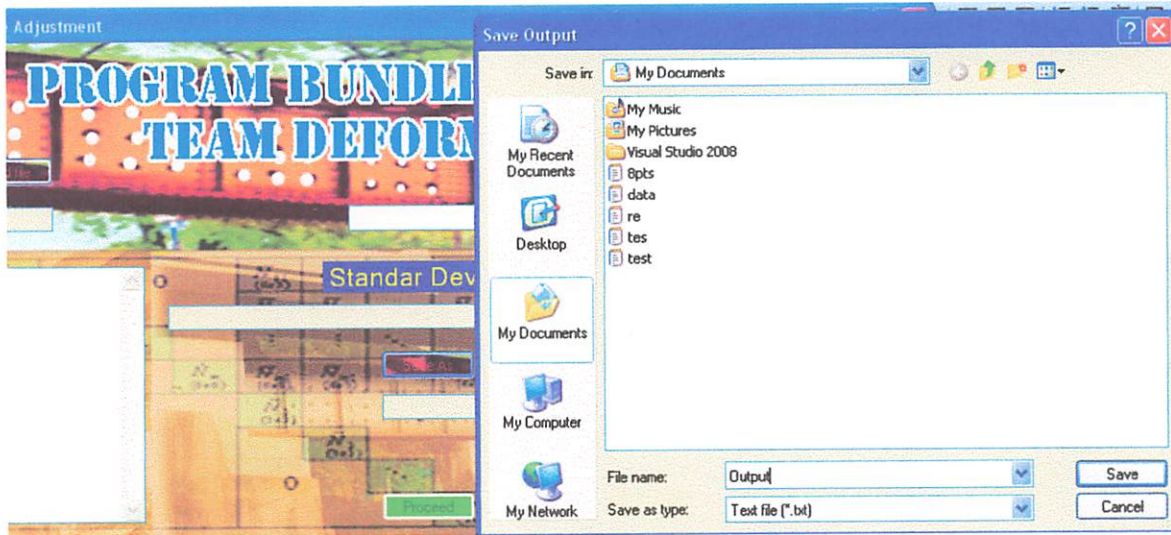
Pengertian *interface* adalah himpunan operasi-operasi yang menentukan perilaku sebuah objek, biasanya digunakan apabila kita ingin menentukan apa saja yang harus dilakukan oleh satu class tetapi tidak menentukan bagaimana cara untuk melakukannya. (*interface dan kelas abstrak, Husnan Wibowo, 2010*). Berikut adalah contoh *interface* dengan menggunakan C #2008:



Gambar 2.13 Tampilan Opening Program



Gambar 2.14 Tampilan Input Data



Gambar 2.15 Tampilan Output Data



Gambar 2.16 Tampilan Safe Final Data



Gambar 2.17 Tampilan perubahan Parameter EO sebelum dan sesudah proses BundleAdjustment

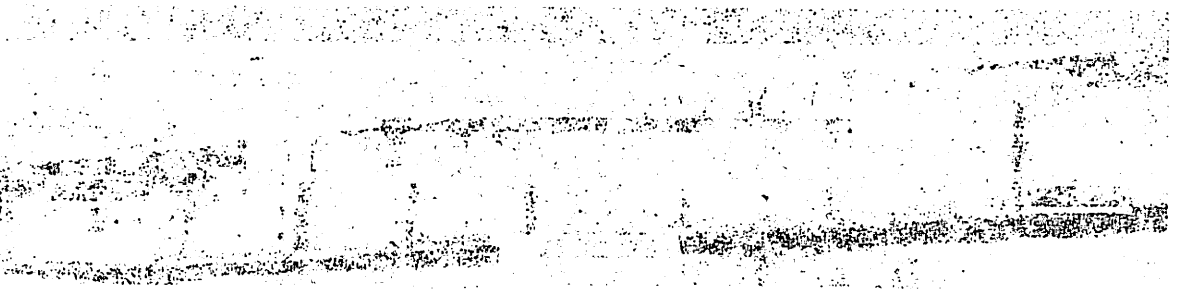


Gambar 2.18 Tampilan perubahan Object Space Point sebelum dan sesudah proses BundleAdjustment



Document 2: The Transatlantic Slave Trade (1482-1807)

Document 3: The Industrial Revolution (1750-1850)



Document 4: The American Civil War (1861-1865)

Document 5: The Great Depression (1929-1939)

BAB III

METODE PENELITIAN

3.1 Persiapan Penelitian

Sebelum melakukan sebuah penelitian, diperlukan suatu persiapan yang matang guna memperlancar proses penelitian sampai penyajian hasil dari batasan penelitian yang dibahas. Untuk memperoleh hasil yang maksimal maka ada beberapa hal yang harus dipersiapkan terlebih dahulu, sebagai berikut :

3.1.1 Materi Penelitian

Adapun materi yang digunakan sebagai bahan dalam penelitian ini meliputi data-data sebagai berikut :

1. Data foto, merupakan data dasar yang dibutuhkan dalam proses *Bundle Adjustment Close Range Photogrammetry* yang berupa data digital.
2. Data parameter *Interior Orientation* (IO) pendekatan yaitu (x, y, f) dan koordinat foto (x, y) serta ketelitian (σ_x, σ_y) dari proses *centroid*. Parameter interior mendefinisikan geometri internal sebuah kamera antara lain *perspektif center*, *principle point*, serta panjang fokus dari kamera.
3. Data parameter *Exterior Orientation* (EO) pendekatan yaitu $(X_L, Y_L, Z_L, \omega, \phi, \kappa)$ tiap foto dari proses *Relative Orientation* dan Proses *Resection*

4. Data parameter *Object space* pendekatan yaitu (X_i, Y_i, Z_i) dari proses *Intersection*.

3.1.2 Alat Penelitian

Untuk mendukung pelaksanaan kegiatan penelitian ini, maka diperlukan beberapa perangkat keras (*hardware*) dan perangkat lunak (*software*) antara lain :

3.1.2.1 Perangkat keras (*hardware*), terdiri dari :

- a. Kamera SLR Nikon D60

Spesifikasi kamera SLR Nikon D60 sebagai berikut :

- Tipe Lensa : *Single-lens reflex digital camera*
- LCD screen : *2.5-inch*
- Sensor Size : *23.6 x 15.8 mm*



Gambar 3.1 Kamera SLR Nikon D 60 tampak depan dan belakang



Gambar 3.2 Aksesoris dari Kamera (Tipe Lensa, Baterai, Memori)

4. Data parameter V&A di bawah ini akan diolah dengan menggunakan software Microsoft Excel.

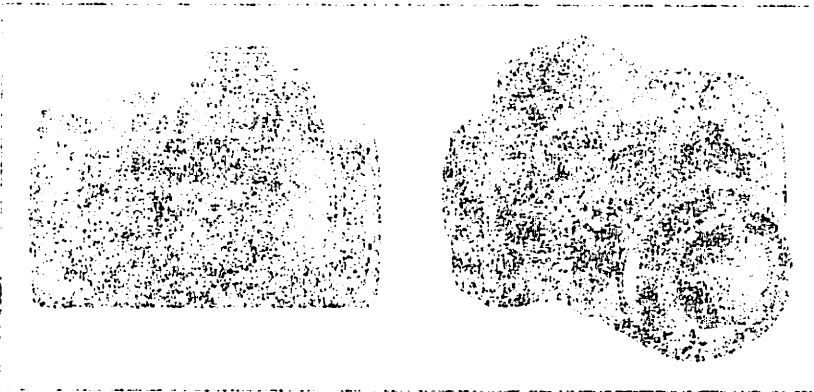
3.1.2. Alat Penelitian

Untuk mendukung pelaksanaan penelitian ini maka diperlukan beberapa perangkat keras (Hardware) dan perangkat lunak (software) antara lain:

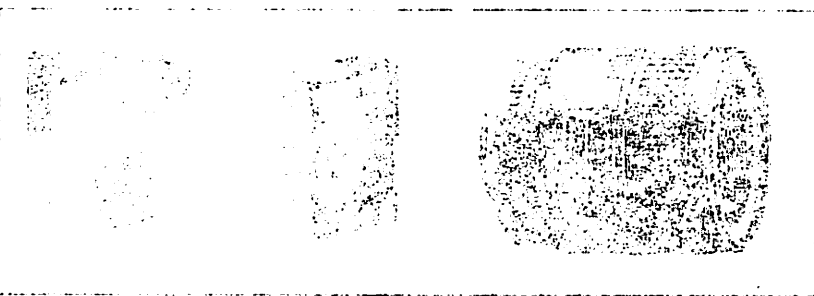
- 1. Perangkat keras (hardware) terdiri dari:
 - a. Kamera SLR Nikon D90

Spesifikasi kamera SLR Nikon D90 sebagai berikut:

- Tipe lensa : Zumbaxar 18-200 mm
- LCD screen : 3.0 inch
- Sensor size : 23.6 mm x 15.8 mm



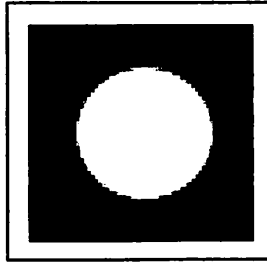
Gambar 3.1. Hasil pengolahan gambar dengan teknik



Gambar 3.2. Hasil pengolahan gambar dengan teknik

b. Striker *Retro Reflective Target*

Striker *retro reflective* berupa stiker berbentuk lingkaran berwarna hitam dengan background putih atau sebaliknya. Adapun contoh dari stiker *retro reflective target* adalah sebagai berikut :



Gambar 3.3 Striker Retro Target

Striker ini merupakan titik target yang dipasang pada objek yang digunakan untuk menentukan objek pemotretan. *Retro reflective target* sudah sering digunakan dalam aplikasi survey deformasi, dengan alasan karena *retro reflective target* mampu menghasilkan cahaya 2000 kali lebih terang dibandingkan dengan stiker biasa. sehingga mempermudah untuk menentukan koordinat nilai tengah dari target (Clarke dan Wang, 1999).

c. PC

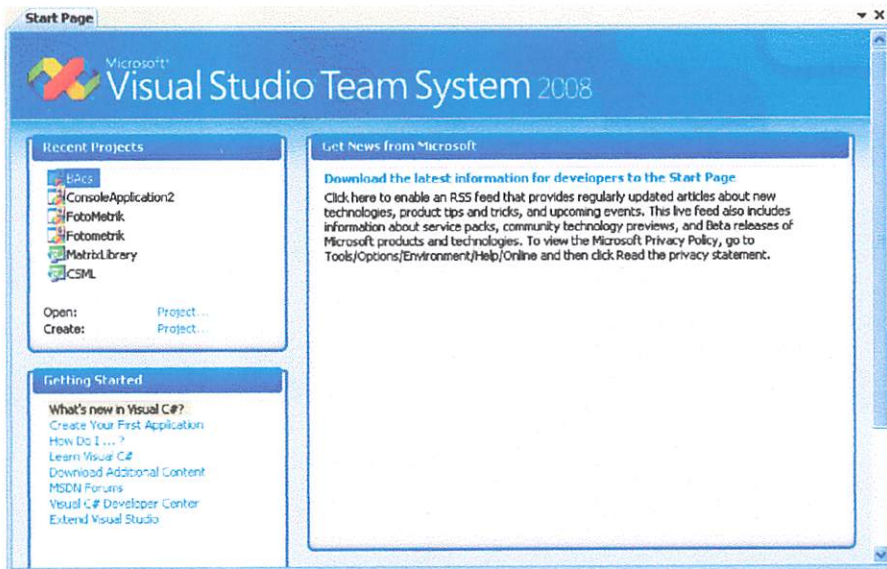
Spesifikasinya sebagai berikut :

- | | |
|-----------------------------|--------|
| - Intel® Pentium ® 4 2.0 GB | 1 unit |
| - RAM 1,25 GB DDR1 | 1 unit |
| - Hard Disk 80 GB | 1 unit |
| - Samsung Studio Works 14" | 1 unit |

3.1.2.2 Perangkat lunak (*software*), terdiri dari :

a. Microsoft Visual Studio 2008 (Bahasa Pemograman C#)

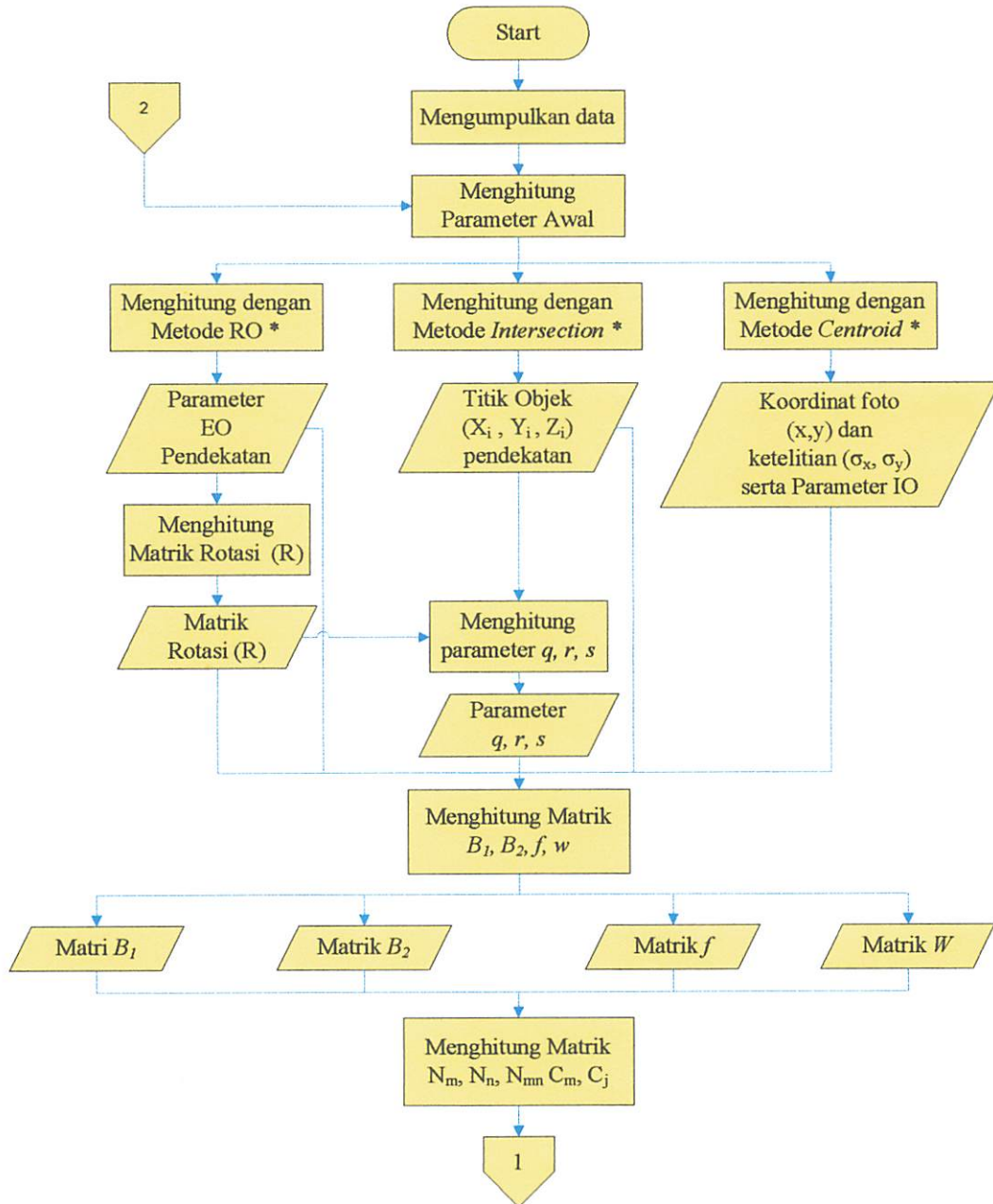
Microsoft Visual Studio 2008 merupakan salah satu perangkat lunak yang berorientasi Object (*Object-oriented programming* disingkat OOP). Konsep pemograman orientasi objek menekankan pada : Kelas, objek, method, event serta action.



Gambar 3.4 Tampilan awal Program C#

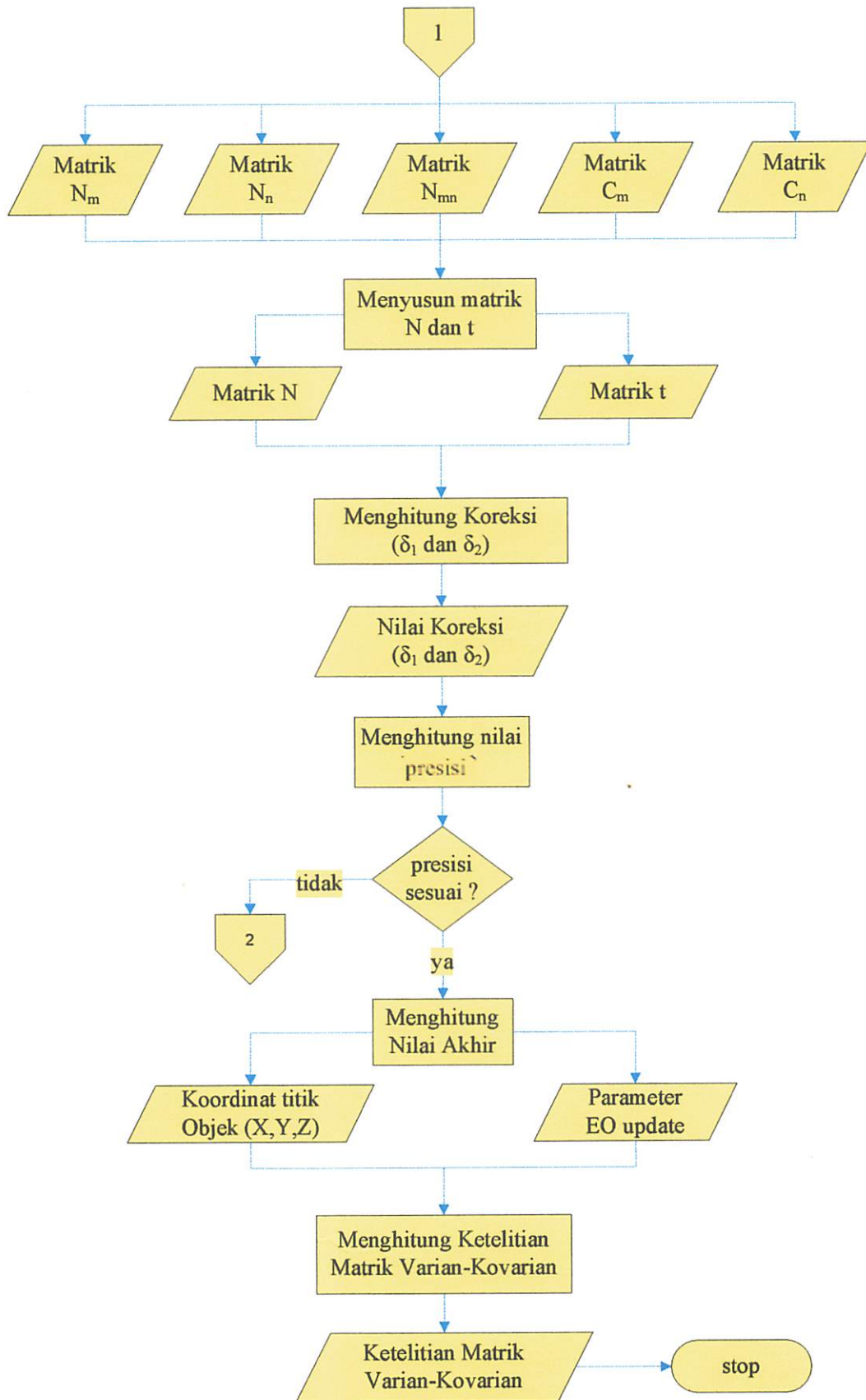
3.2 Diagram Alir Penelitian

Dalam proses penelitian haruslah dibuat suatu kerangka pekerjaan yang sistematis agar mudah dipahami dan mempermudah dalam penelitian. Adapun langkah atau alur penelitian yang akan dilakukan sebagai berikut :



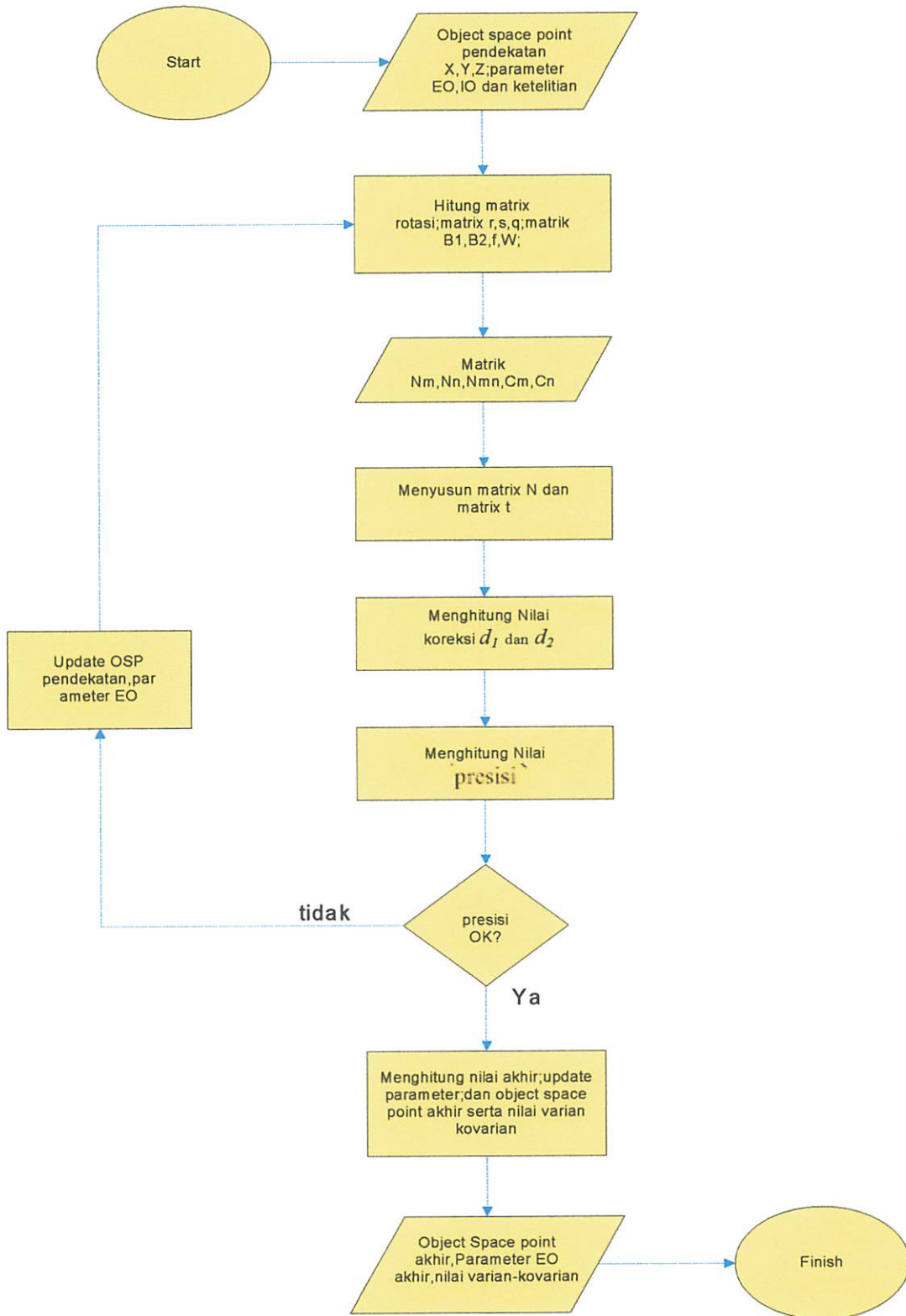
ket : * Proses telah dilakukan oleh peneliti lain pada tahap sebelumnya.

Gambar 3.5 Diagram Alir Penelitian (1)



Gambar 3.6 Diagram Alir Penelitian (2)

3.3 Diagram Alir Program



Gambar 3.7 Diagram Alir program

3.3 Penjelasan Diagram Alir (*FlowChart*) Program

1. Dari Program sebelumnya akan didapatkan *Output* koordinat foto (x,y) dan ketelitian (σ_x, σ_y) serta Parameter *Interior Orientation* (IO) dengan menggunakan metode *centroid*, Parameter *Exterior Orientation* (EO) pendekatan yaitu $\omega, \varphi, \kappa, XL, YL, ZL,$ diperoleh melalui proses *Relative Orientation* (RO), kemudian akan didapatkan juga koordinat titik-titik objek (X_i, Y_i, Z_i) pendekatan yang diperoleh menggunakan proses interseksi. Data awal tersebut akan disimpan didalam bentuk file “.txt” yang akan digunakan sebagai *input* awal untuk proses lebih lanjut didalam program ini. Berikut adalah bentuk format penulisan data awal didalam file “.txt” yang dapat dibaca dengan fasilitas “*notepad*” didalam “*windows*”.

File	Edit	Format	View	Help					
2	24	0	0						
1	0.0000000000000000	0.0000000000000000	0.0000000000000000	0.0000000000000000	0.0000000000000000	0.0000000000000000	0.0000000000000000	0.0000000000000000	0.000
2	-60.721492498851000	52.882672731586000	95.494547602322600	0.6480947787697900	0.4562751110935020				-0.60
8									
1	0.02929502575192260	0.04579306820004850	-0.52915865223955700	1.32896177715303000	2.07739149341532000				-9.04
2	0.11138421123157200	0.10159631754105700	-0.60470412206608100	4.42254319326426000	4.03391196464765000				-6.63
3	0.14835407247379500	0.12786028729578200	-0.64035562959913600	5.55408249255395000	4.78683578631020000				-5.26
4	0.21844079628691800	0.17936189090497900	-0.71180918344822600	7.36735810733454000	6.04934290188659000				-2.01
5	0.12635628874384600	0.22080243827207500	-0.78000527802147200	3.88850232242012000	6.79489851216393000				1.679
6	0.07774990844718800	0.24171439050729400	-0.81438727898585600	2.29096335592785000	7.12230820988226000				3.219
7	-0.01564561825798700	0.28169972009095300	-0.87997376792929000	-0.42672236295650600	7.68314605528816000				5.693
8	-0.10139146217175400	0.22238233945341600	-0.79838784129994000	-3.04788648123487000	6.68494281044383000				2.182
8									
1	0.024562546422411500	0.024522750908847800	0.040198401636953000	0.040062738250742800	0.029856463250862400				
2	0.030701479934893900	0.030454672116186200	0.030345956320900600	0.029856463250862400	0.027156914962175700				
3	0.027031115029389700	0.026989981808316500	0.027547147238551600	0.027156914962175700	0.025696341487272500				
4	0.023285362597697200	0.023500028654633800	0.025892493297080900	0.025696341487272500	0.0248414396296500				
5	0.025823505104790200	0.025551780247607600	0.028352944155996800	0.0248414396296500	0.031207347756492700				
6	0.04488402748833700	0.044806140342306800	0.031181034426962900	0.031207347756492700	0.036794875520373800				
7	0.048166307412915600	0.047030661890723000	0.036794875520373800	0.036794875520373800	0.042398152698081800				
8	0.045870506341846300	0.045570643003169900	0.042732448390881800	0.042398152698081800					

Gambar 3.8 Input dari notepad

Dengan fasilitas “*string reader*” dari *C#* maka data input tersebut dapat dikenali dan dibaca oleh program,yang selanjutnya akan diberikan variable

pada setiap data yang terbaca. Dibawah ini adalah proses pembacaan data dan pemvariabelan setiap data.

```
private void ReadFile(ref string file)
{
    // buka file untuk pembacaan data.
    string[] readText = File.ReadAllLines(file);
    System.Diagnostics.Trace.WriteLine("readtext: " + readText.Count());
    for (int i = 0; i < readText.Count(); i++)
    {
        if (i == 0)
        {
            String[] firstlineText = readText[0].Split(new char[] { ' ', '\t' });
            if (firstlineText.Length == 4)
            {
                // ekstrak nomor pair, panjang focus, X0, Y0
                ImageNumber = int.Parse(firstlineText[0]);
                Focal = Double.Parse(firstlineText[1]);
                Xo = Double.Parse(firstlineText[2]);
                Yo = Double.Parse(firstlineText[3]);
            }
        }
        else
        {
            String[] textc = readText[i].Split(new char[] { ' ', '\t' });
            if (textc.Length == 7)
            {
                //Extracy data parameter camera pada tiap image
                PhotoPairCams pairc = new PhotoPairCams();
                pairc.Omega = Double.Parse(textc[1]);
                pairc.Phi = Double.Parse(textc[2]);
                pairc.Kappa = Double.Parse(textc[3]);
                pairc.XL = Double.Parse(textc[4]);
                pairc.YL = Double.Parse(textc[5]);
                pairc.ZL = Double.Parse(textc[6]);
                PhotoPairListCams.Add(pairc);
            }
        }
        int im = _imageNumber + 1;
        if (i == im)
        {
            String[] pointNumberline = readText[im].Split(new char[] { ' ', '\t' });
            if (pointNumberline.Length == 1)
            {
                //extract the number image
                PointNumber = int.Parse(pointNumberline[0]);
            }
        }
        else //from the second line and beyond
        {
            String[] textp = readText[i].Split(new char[] { ' ', '\t' });
            if (textp.Length == 8)
            {
                // extract for each line: label, X,Y,Z, x1,y1,x2,y2
                PhotoPairPoint pairp = new PhotoPairPoint();
                pairp.Labelp = textp[0];
                pairp.Xc = Double.Parse(textp[1]);
            }
        }
    }
}
```

```

        pairp.Yc = Double.Parse(textp[2]);
        pairp.Zc = Double.Parse(textp[3]);
        pairp.X1 = Double.Parse(textp[4]);
        pairp.Y1 = Double.Parse(textp[5]);
        pairp.X2 = Double.Parse(textp[6]);
        pairp.Y2 = Double.Parse(textp[7]);
        PhotoPairListPoint.Add(pairp);

        //System.Diagnostics.Trace.WriteLine("X:" + pair.Xc.ToString());
        //System.Diagnostics.Trace.WriteLine("Y:" + pair.Yc.ToString());
    }
}
int ip = _imageNumber + 1 + _pointNumber + 1;
if (i == ip)
{
    String[] residuNumberline = readText[ip].Split(new char[] { ' ', '\t' });
    if (residuNumberline.Length == 1)
    {
        //extract the number residu
        ResiduNumber = int.Parse(residuNumberline[0]);
    }
}
else //
{
    String[] textr = readText[i].Split(new char[] { ' ', '\t' });
    if (textr.Length == 5)
    {
        PhotoPairResidu pairr = new PhotoPairResidu();
        pairr.Labelr = textr[0];
        pairr.Xresidu1 = Double.Parse(textr[1]);
        pairr.Yresidu1 = Double.Parse(textr[2]);
        pairr.Xresidu2 = Double.Parse(textr[3]);
        pairr.Xresidu2 = Double.Parse(textr[4]);
        PhotoPairListResidu.Add(pairr);

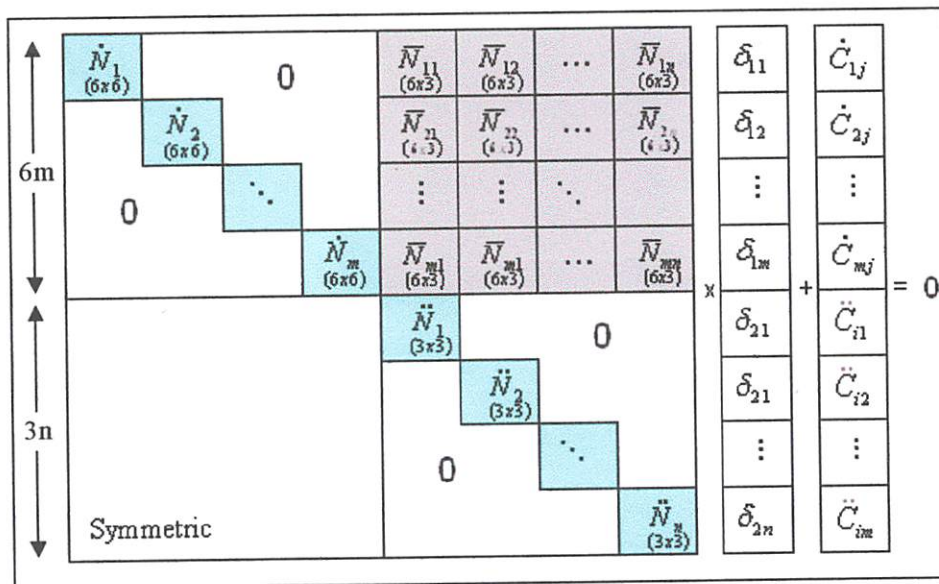
        //System.Diagnostics.Trace.WriteLine("X:" + pairr.Xc.ToString());
        //System.Diagnostics.Trace.WriteLine("Y:" + pairr.Yc.ToString());
    }
}
}
}
}

```

2. Setelah setiap data divariabelkan maka akan dimulai proses perhitungan untuk mendapatkan matrix rotasi dari parameter EO pendekatan, matrix r , s , q dari koordinat *object space* pendekatan yang kemudian akan diteruskan secara berurutan hingga mendapatkan matrix. Dari perhitungan tersebut akan didapatkan matrix $B_1, B_2, \delta_1, \delta_2$. Kemudian akan dilanjutkan dengan perhitungan untuk mendapatkan elemen-elemen matrix

$$\dot{N}_{ij}, \ddot{N}_{ij}, \bar{N}_{ij}, \dot{C}_{ij}, \ddot{C}_{ij}$$

3. Setelah matrik $\dot{N}_{ij}, \ddot{N}_{ij}, \bar{N}_{ij}, \dot{C}_{ij}, \ddot{C}_{ij}$ didapatkan maka langkah selanjutnya adalah menyusun matrik N dan matrix t. Berikut adalah susunan matriknya



Gambar 3.9 Susunan Matrix N dan t (Fraser, 1997):

4. Setelah matrik N dan t tersusun maka proses selanjutnya adalah menghitung matrix koreksi δ_1 dan δ_2
5. Menghitung nilai ketelitian dapat dilakukan jika matrik koreksi telah didapat. Didalam proses ini user akan melakukan verifikasi terhadap nilai ketelitian yang didapatkan. Jika nilai residu tidak sesuai maka perlu dilakukan proses iterasi hingga mendapatkan nilai ketelitian yang sesuai. Dalam program ini telah disediakan fasilitas iterasi hingga nilai ketelitian yang sesuai didapatkan. Perancang telah membuat hasil *output* sesuai dengan format input data dengan melakukan update pada setiap parameter *EO* dan *object space point* pendekatan (X,Y,Z) dengan menambahkan nilai koreksi yang didapatkan pada proses sebelumnya sehingga proses iterasi dapat dilakukan dengan menjalankan program dari awal lagi dengan mengganti input dengan *output* hasil proses sebelumnya.

6. Jika ketelitian telah sesuai maka akan dilanjutkan dengan menghitung nilai akhir dengan cara mengupdate data *input* dengan nilai koreksi yang telah didapatkan dalam proses ini, selanjutnya adalah menghitung nilai Standar deviasi. Proses selesai.

BAB IV

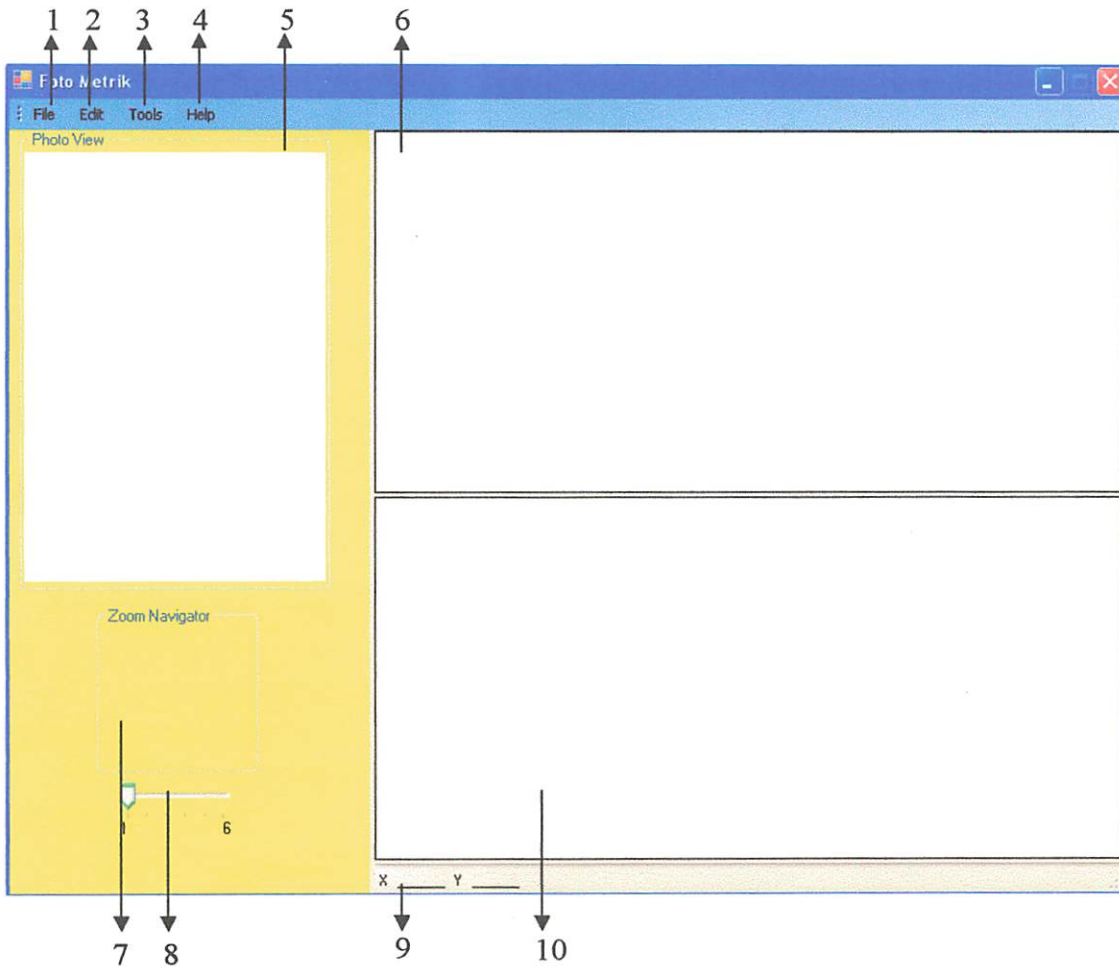
HASIL DAN ANALISA

4.1. HASIL

Didalam tema tugas akhir ini, Dihilangkan suatu program pengolahan dan perhitungan deformasi . Berikut pembahasan Hasil dan penjelasannya

4.1.1 Langkah Kerja Di dalam Program

Didalam pembuatan program perhitungan bundle adjustment banyak terdapat point-point penting yang harus diperhatikan baik didalam algoritma maupun didalam listing program itu sendiri. Algoritma yang tepat serta mudah dipahami akan sangat membantu terhadap ketepatan, ketelitian serta kestabilan suatu program. Demikian juga dengan dengan Listing program,dengan perintah-perintah yang tepat, efisien serta flesibel, kinerja dan kestabilan suatu program dapat terjaga didalam menghadapi kemungkinan-kemungkinan yang terjadi ketika program tersebut dijalankan. Oleh karena itu perlu dibahas secara jelas langkah-langkah didalam menjalankan program ini serta perintah-perintah didalam interface apa saja yang diberikan.



Gambar 4.1 Program Overview

Keterangan :

1. *Toolbar Menu*
2. *Toolbar Edit*
3. *Toolbar Tools*
4. *Toolbar Help*
5. *Image Preview*

6. *Imagebox Photo 1*
7. *Zoom Navigator*
8. *Trackbar Zoom*
9. *Statusbar*
10. *Imagebox Photo 2*

1. Toolbar Menu

- *Sub Menu Open Image*

Digunakan untuk memulai penambahan data foto ke dalam program dan akan secara otomatis mengaktifkan *project* baru.

- *Sub Menu New Project*

Digunakan untuk mengakhiri aktif *project* dan memulai *project* baru.

- *Sub Menu Open Project*

Digunakan untuk membuka *project* yang telah disimpan sebelumnya.

- *Sub Menu Save As*

Digunakan untuk menyimpan *project* dengan nama baru.

- *Sub Menu Save*

Digunakan untuk menyimpan *project* dengan nama yang sudah ada sebelumnya

- *Exit*

Digunakan untuk keluar dari Program fotometrik

2. Toolbar Tools

- *Sub Menu Save Centroid*

Digunakan untuk menyimpan hasil ekstraksi *centroid* dari *image* kedalam file untuk diproses kedalam *Relative Orientation*.

- *Sub Menu Relative Orientation*

Digunakan untuk melakukan perhitungan *Relative Orientation* dari data *Centroid* yang sudah disimpan di dalam file sebelumnya untuk mendapatkan

matrik rotasi dan koordinat posisi kamera dan menyimpannya kedalam file untuk dilakukan proses *Intersection*

- *Sub Menu Intersection*

Digunakan untuk melakukan perhitungan *Intersection* dari data yang telah disimpan sebelumnya untuk mendapatkan posisi 3 dimensi titik yang ada didalam 2 foto.

- *Sub Menu Save Intersection Image 1*

Digunakan untuk menyimpan titik koordinat 3 dimensi hasil *intersection* didalam *Image 1* kedalam file.

- *Sub Menu Save Intersection Image 2*

Digunakan untuk menyimpan titik koordinat 3 dimensi hasil *intersection* didalam *Image 2* kedalam file.

- *Sub Menu Resection Image 1*

Digunakan untuk melakukan perhitungan *Resection* dari posisi 3 dimensi titik yang ada di dalam *Image 1* untuk didapatkan parameter *Ekterior Orientatation* dan ketelitiannya dalam *Image 1* .

- *Sub Menu Resection Image 2*

Digunakan untuk melakukan perhitungan *Resection* dari posisi 3 dimensi titik yang ada di dalam *Image 2* untuk didapatkan parameter *Ekterior Orientatation* dan ketelitiannya dalam *Image 2*.

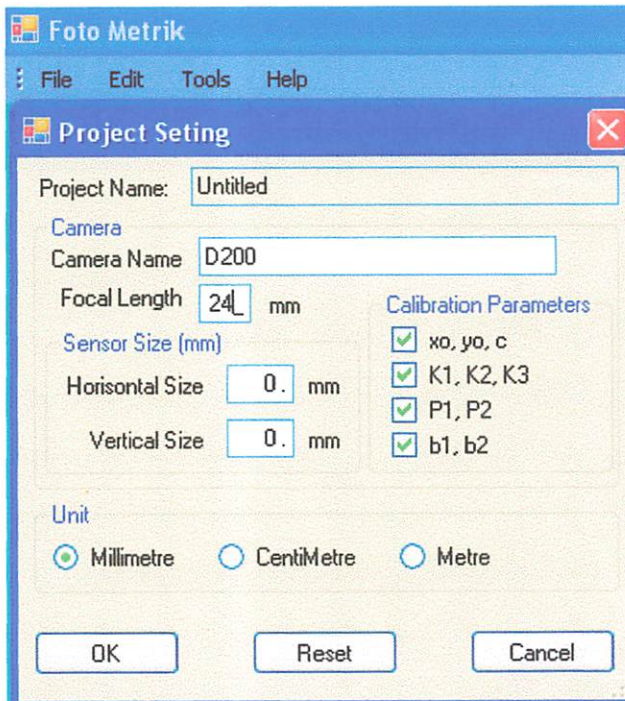
- *Save Everything for Bundle*

Menyimpan semua data dari *Centroid*, *Intersection*, dan *Resection* ke dalam file untuk digunakan dalam perhitungan *Bundle*.

- *Bundle Adjustment*

Digunakan untuk melakukan proses perhitungan Bundle Adjustment untuk mendapatkan nilai akurasi yang sekecil-kecilnya.

4.1.1.1 Project Setting



Gambar 4.2 Project Setting

Seperti gambar yang dtunjukkan diatas, pada kolom *project name* adalah nama project yang akan dibuat. Pada kolom Camera Name adalah nama kamera yang digunakan sedangkan pada kolom Focal length diisi dengan nilai panjang fokus kamera yang digunakan. Pada bagian Sensor size biarkan apa adanya. Setelah semuanya selesai tekan **OK**.