CASE#4

RESULT:

Norm delta\_x:

2.97498566732953e-12

Main\_Coordinates =

4×3 table

X\_Main Y\_Main Z\_Main

\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_

474.825 6397.219 617.924

208.221 1223.701 670.953

3739.286 5918.668 650.891

3785.639 715.638 700.419

Computed\_Coordinates =

4×3 table

X\_Computed Y\_Computed Z\_Computed

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

470.701854042659 6400.40753013371 623.028219712471

201.773546822054 1222.30470904238 676.760339546093

3736.61412944566 5917.68836364468 646.144239119921

3783.9972189921 719.878579581782 691.839169524631

Exterior\_Parameters =

14×8 table

Ran Image\_Num X0 Y0 Z0 Omega Phi Kappa

\_\_\_ \_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1 1 1179.16244346467 6354.73753726993 2191.72960452785 -0.000172477271076342 0.000136686105689272 4.7257735616698

1 2 1164.17306224184 5487.7253758323 2190.85203258125 0.000127578113995061 0.000107008747999924 4.72121132486602

1 3 1157.75987656451 4612.86628477231 2190.94878833743 -0.000184145868411098 -6.19262697784502e-07 4.71558686638035

1 4 1146.37910629844 3742.66213548604 2189.50296422378 -0.000222899172148439 0.000201658394301929 4.7167001495219

1 5 1143.03282278492 2904.99860946764 2188.94685603542 -0.000160215977006118 0.000131034104065461 4.71078602339411

1 6 1143.08870886211 2031.3470020988 2188.79995247428 -0.000161679882197645 0.000379352102071108 4.7199511158066

1 7 1134.03704031547 1190.63039179775 2189.46913154029 -0.000128807965000812 1.59865328211945e-05 4.71470816173286

2 1 2764.274508592 5940.95912849857 2181.68306113235 -7.82543849982756e-05 -8.25775173039776e-05 4.72122873432146

2 2 2787.89555603381 5056.87045558949 2179.33265232647 -0.000121285336962043 0.000258018698499446 4.72426699352682

2 3 2804.26608603203 4199.21260152619 2176.89374768545 -0.00020242984952412 -1.45034541445218e-05 4.72227005278618

2 4 2821.63747591815 3339.36193046527 2174.92603848363 -0.000138430315400835 -9.76414911060074e-05 4.72651557362799

2 5 2840.26868255655 2475.59073517619 2174.74623607153 -0.000212929468395147 1.98042697936924e-05 4.73030745353843

2 6 2856.10265709824 1609.4213688406 2174.86927213259 1.42392937977451e-05 0.000181873528968801 4.73200413219544

2 7 2869.42245207879 736.188125320849 2174.64215333397 -6.75240115456505e-05 0.000202285463983088 4.73454396564901

Elapsed time is 0.604316 seconds.

CODE:

**%% Project 4 --> MohammadJavadSoltani\_SamaneKarami**

**close all; clear; clc; format long g; tic;**

**%% First We Load Our Parameters Coumputed In Last Project And Our Data**

**load('parameters1.mat'); load('parameters2.mat');**

**load('data.mat'); load('data\_control\_points.mat');**

**%% These Variables Are Primary Values**

**data\_control\_points(4, 4) = 641.272969739521;**

**data\_control\_points(10, 4) = 657.615318655257;**

**data\_control\_points(6, 2) = 1.817848254013564e+03;**

**data\_control\_points(6, 3) = 3.830066680064776e+03;**

**compelete\_control\_point = [data\_control\_points(1 : 3, :);data\_control\_points(5, :); data\_control\_points(7 : 9, :); data\_control\_points(11, :)];**

**plot(compelete\_control\_point(:, 2), compelete\_control\_point(:, 3), '\*');**

**text(compelete\_control\_point(:, 2), compelete\_control\_point(:, 3), int2str(compelete\_control\_point(:, 1)));**

**C = 153.692 / 1000; % mm -> m**

**% So 1109 - 2189 - 1709 - 2789 Are Our Full Control points.**

**%% 4 FULL Cp.**

**block\_control\_IDs = [1109; 2189; 1709; 2789];**

**block\_control\_points = zeros(4, 4);%4 CP / ID + XYZ**

**for i = 1 : length(data\_control\_points)**

**for j = 1 : length(block\_control\_IDs)**

**if data\_control\_points(i, 1) == block\_control\_IDs(j)**

**block\_control\_points(j, :) = data\_control\_points(i, :);**

**end**

**end**

**end%block\_control\_points Are satisfied!**

**block\_control\_points = sortrows(block\_control\_points, 1);**

**block\_control\_points(:, 5) = [1:4];% Numbering them**

**block\_control\_points\_copy = block\_control\_points;%Creat a Copy From it.**

**height\_control\_point = [data\_control\_points(6, :)];**

**plan\_control\_point = [data\_control\_points(4, :); data\_control\_points(10, :)];**

**%% Now We Will Create Our Main Matrix**

**result\_images(:, 8) = result\_images(:, 6);%Use from prj1 result**

**result\_images(:, 6 : 7) = zeros(14, 2);**

**result\_images\_copy = result\_images;%Creat a Copy From it.**

**len\_data = length(data);**

**syms X0 Y0 Z0 Omega Phi Kappa X Y Z x y**

**R = [cos(Kappa), -sin(Kappa), 0; sin(Kappa), cos(Kappa), 0; 0, 0, 1] \* ...**

**[cos(Phi), 0, sin(Phi); 0, 1, 0; -sin(Phi), 0, cos(Phi)] \* ...**

**[1, 0, 0; 0, cos(Omega), -sin(Omega); 0, sin(Omega), cos(Omega)];**

**temp = R \* [(X - X0); (Y - Y0); (Z - Z0)];**

**m = temp(1);**

**n = temp(2);**

**q = temp(3);**

**r = m / q;**

**s = n / q;**

**clear temp;**

**Fx = x + C \* r;**

**Fy = y + C \* s;**

**F = [jacobian(Fx, [X0, Y0, Z0, Omega, Phi, Kappa]); jacobian(Fy, [X0, Y0, Z0, Omega, Phi, Kappa])];**

**Fg = [jacobian(Fx, [X, Y, Z]); jacobian(Fy, [X, Y, Z])];**

**Fg\_H = [jacobian(Fx, [X, Y]); jacobian(Fy, [X, Y])];**

**Fg\_Plan = [jacobian(Fx, Z); jacobian(Fy, Z)];**

**%% P Matrix**

**VARIANCE = [eye(328) \* 7 \* 10 ^ -6, zeros(328, 96); zeros(12, 328), eye(12) \* 0.15, zeros(12, 84)];**

**for i=1:8%sigma for Z is diffrent.**

**VARIANCE(len\_data \* 2 + 3\*i,len\_data \* 2 + 3\*i) = (20 / 100) .^ 2;**

**if i<2**

**VARIANCE(len\_data \* 2 + 3\*8 + 2\*2 +1) = (20 / 100) .^ 2;**

**end**

**end**

**%Now for EOPs:**

**for i = 1 : 14 % 0.2 ->XY / 0.25 -> Z / Omega,phi,kappa ->5''**

**VARIANCE(6 \* i - 5 + 340 : 6 \* i + 340, 6 \* i - 5 + 340 : 6 \* i + 340) = [0.2, 0, 0, 0, 0, 0; 0, 0.2, 0, 0, 0, 0; 0, 0, 0.25, 0, 0, 0; 0, 0, 0, 5 / 3600 / 180 \* pi, 0, 0; 0, 0, 0, 0, 5 / 3600 / 180 \* pi, 0; 0, 0, 0, 0, 0, 5 / 3600 / 180 \* pi];**

**end**

**VARIANCE = VARIANCE .^ 2;**

**P = inv(VARIANCE);**

**AEO = [];**

**Ag = [];**

**w = zeros(328 + 12 + 6 \* 14, 1);**

**delta\_x = 1;**

**%% Loop**

**while norm(delta\_x) > 10e-12**

**i\_temp = 0;**

**for i = 1 : len\_data**

**xcheck = false;**

**for k = 1 : length(block\_control\_points)**

**if data(i, 3) == block\_control\_points(k) || data(i, 6) == 0%Tie or Full Controll**

**xcheck = true;**

**end**

**end**

**if ~xcheck**

**continue;**

**else**

**i\_temp = i\_temp + 1;**

**end**

**%-------------------------------------------------------------------------------------------------------------------**

**image\_num = data(i, 2);**

**if data(i, 1) == 2**

**image\_num = image\_num + 7;**

**end**

**x = data(i, 4);**

**y = data(i, 5);**

**X0 = result\_images(image\_num, 3);**

**Y0 = result\_images(image\_num, 4);**

**Z0 = result\_images(image\_num, 5);**

**Omega = result\_images(image\_num, 6);**

**Phi = result\_images(image\_num, 7);**

**Kappa = result\_images(image\_num, 8);**

**if data(i, 6) == 1**

**% num = data(i, 7) - 100;**

**for i\_m = 1 : size(block\_control\_points, 1)%which Full CP?**

**if data(i, 3) == block\_control\_points(i\_m, 1)**

**num = block\_control\_points(i\_m, 5);**

**end**

**end**

**X = block\_control\_points\_copy(num, 2);**

**Y = block\_control\_points\_copy(num, 3);**

**Z = block\_control\_points\_copy(num, 4);**

**Ag(2 \* i\_temp - 1 : 2 \* i\_temp, num \* 3 - 2 + 150 : num \* 3 + 150) = [ (2768668935719305\*cos(Kappa)\*cos(Phi))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))) + (2768668935719305\*sin(Phi)\*((Z - Z0)\*(sin(Kappa)\*sin(Omega) + cos(Kappa)\*cos(Omega)\*sin(Phi)) - (Y - Y0)\*(cos(Omega)\*sin(Kappa) - cos(Kappa)\*sin(Omega)\*sin(Phi)) + cos(Kappa)\*cos(Phi)\*(X - X0)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))^2), - (2768668935719305\*(cos(Omega)\*sin(Kappa) - cos(Kappa)\*sin(Omega)\*sin(Phi)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))) - (2768668935719305\*cos(Phi)\*sin(Omega)\*((Z - Z0)\*(sin(Kappa)\*sin(Omega) + cos(Kappa)\*cos(Omega)\*sin(Phi)) - (Y - Y0)\*(cos(Omega)\*sin(Kappa) - cos(Kappa)\*sin(Omega)\*sin(Phi)) + cos(Kappa)\*cos(Phi)\*(X - X0)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))^2), (2768668935719305\*(sin(Kappa)\*sin(Omega) + cos(Kappa)\*cos(Omega)\*sin(Phi)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))) - (2768668935719305\*cos(Omega)\*cos(Phi)\*((Z - Z0)\*(sin(Kappa)\*sin(Omega) + cos(Kappa)\*cos(Omega)\*sin(Phi)) - (Y - Y0)\*(cos(Omega)\*sin(Kappa) - cos(Kappa)\*sin(Omega)\*sin(Phi)) + cos(Kappa)\*cos(Phi)\*(X - X0)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))^2);**

**(2768668935719305\*cos(Phi)\*sin(Kappa))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))) + (2768668935719305\*sin(Phi)\*((Y - Y0)\*(cos(Kappa)\*cos(Omega) + sin(Kappa)\*sin(Omega)\*sin(Phi)) - (Z - Z0)\*(cos(Kappa)\*sin(Omega) - cos(Omega)\*sin(Kappa)\*sin(Phi)) + cos(Phi)\*sin(Kappa)\*(X - X0)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))^2), (2768668935719305\*(cos(Kappa)\*cos(Omega) + sin(Kappa)\*sin(Omega)\*sin(Phi)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))) - (2768668935719305\*cos(Phi)\*sin(Omega)\*((Y - Y0)\*(cos(Kappa)\*cos(Omega) + sin(Kappa)\*sin(Omega)\*sin(Phi)) - (Z - Z0)\*(cos(Kappa)\*sin(Omega) - cos(Omega)\*sin(Kappa)\*sin(Phi)) + cos(Phi)\*sin(Kappa)\*(X - X0)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))^2), - (2768668935719305\*(cos(Kappa)\*sin(Omega) - cos(Omega)\*sin(Kappa)\*sin(Phi)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))) - (2768668935719305\*cos(Omega)\*cos(Phi)\*((Y - Y0)\*(cos(Kappa)\*cos(Omega) + sin(Kappa)\*sin(Omega)\*sin(Phi)) - (Z - Z0)\*(cos(Kappa)\*sin(Omega) - cos(Omega)\*sin(Kappa)\*sin(Phi)) + cos(Phi)\*sin(Kappa)\*(X - X0)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))^2)];**

**% those 4 Full CPs are done here.**

**% elseif data(i, 6) == 2**

**% num = data(i, 7) - 1000;**

**% X = plan\_control\_point(num, 2);**

**% Y = plan\_control\_point(num, 3);**

**% Z = plan\_control\_point(num, 4);**

**% Ag(2 \* i\_temp - 1 : 2 \* i\_temp, 50 \* 3 + 2 + num) = [(2768668935719305\*(sin(Kappa)\*sin(Omega) + cos(Kappa)\*cos(Omega)\*sin(Phi)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))) - (2768668935719305\*cos(Omega)\*cos(Phi)\*((Z - Z0)\*(sin(Kappa)\*sin(Omega) + cos(Kappa)\*cos(Omega)\*sin(Phi)) - (Y - Y0)\*(cos(Omega)\*sin(Kappa) - cos(Kappa)\*sin(Omega)\*sin(Phi)) + cos(Kappa)\*cos(Phi)\*(X - X0)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))^2);**

**% - (2768668935719305\*(cos(Kappa)\*sin(Omega) - cos(Omega)\*sin(Kappa)\*sin(Phi)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))) - (2768668935719305\*cos(Omega)\*cos(Phi)\*((Y - Y0)\*(cos(Kappa)\*cos(Omega) + sin(Kappa)\*sin(Omega)\*sin(Phi)) - (Z - Z0)\*(cos(Kappa)\*sin(Omega) - cos(Omega)\*sin(Kappa)\*sin(Phi)) + cos(Phi)\*sin(Kappa)\*(X - X0)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))^2)];**

**% elseif data(i, 6) == 3**

**% num = data(i, 7) - 10000;**

**% X = height\_control\_point(num, 2);**

**% Y = height\_control\_point(num, 3);**

**% Z = height\_control\_point(num, 4);**

**% Ag(2 \* i\_temp - 1 : 2 \* i\_temp, 50 \* 3 + 1 : 50 \* 3 + 2) = [ (2768668935719305\*cos(Kappa)\*cos(Phi))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))) + (2768668935719305\*sin(Phi)\*((Z - Z0)\*(sin(Kappa)\*sin(Omega) + cos(Kappa)\*cos(Omega)\*sin(Phi)) - (Y - Y0)\*(cos(Omega)\*sin(Kappa) - cos(Kappa)\*sin(Omega)\*sin(Phi)) + cos(Kappa)\*cos(Phi)\*(X - X0)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))^2), - (2768668935719305\*(cos(Omega)\*sin(Kappa) - cos(Kappa)\*sin(Omega)\*sin(Phi)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))) - (2768668935719305\*cos(Phi)\*sin(Omega)\*((Z - Z0)\*(sin(Kappa)\*sin(Omega) + cos(Kappa)\*cos(Omega)\*sin(Phi)) - (Y - Y0)\*(cos(Omega)\*sin(Kappa) - cos(Kappa)\*sin(Omega)\*sin(Phi)) + cos(Kappa)\*cos(Phi)\*(X - X0)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))^2);**

**% (2768668935719305\*cos(Phi)\*sin(Kappa))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))) + (2768668935719305\*sin(Phi)\*((Y - Y0)\*(cos(Kappa)\*cos(Omega) + sin(Kappa)\*sin(Omega)\*sin(Phi)) - (Z - Z0)\*(cos(Kappa)\*sin(Omega) - cos(Omega)\*sin(Kappa)\*sin(Phi)) + cos(Phi)\*sin(Kappa)\*(X - X0)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))^2), (2768668935719305\*(cos(Kappa)\*cos(Omega) + sin(Kappa)\*sin(Omega)\*sin(Phi)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))) - (2768668935719305\*cos(Phi)\*sin(Omega)\*((Y - Y0)\*(cos(Kappa)\*cos(Omega) + sin(Kappa)\*sin(Omega)\*sin(Phi)) - (Z - Z0)\*(cos(Kappa)\*sin(Omega) - cos(Omega)\*sin(Kappa)\*sin(Phi)) + cos(Phi)\*sin(Kappa)\*(X - X0)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))^2)];**

**%**

**elseif data(i, 6) == 0**

**num = data(i, 7);**

**X = result\_coordinates(num, 2);**

**Y = result\_coordinates(num, 3);**

**Z = result\_coordinates(num, 4);**

**Ag(2 \* i\_temp - 1 : 2 \* i\_temp, num \* 3 - 2 : num \* 3) = [ (2768668935719305\*cos(Kappa)\*cos(Phi))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))) + (2768668935719305\*sin(Phi)\*((Z - Z0)\*(sin(Kappa)\*sin(Omega) + cos(Kappa)\*cos(Omega)\*sin(Phi)) - (Y - Y0)\*(cos(Omega)\*sin(Kappa) - cos(Kappa)\*sin(Omega)\*sin(Phi)) + cos(Kappa)\*cos(Phi)\*(X - X0)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))^2), - (2768668935719305\*(cos(Omega)\*sin(Kappa) - cos(Kappa)\*sin(Omega)\*sin(Phi)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))) - (2768668935719305\*cos(Phi)\*sin(Omega)\*((Z - Z0)\*(sin(Kappa)\*sin(Omega) + cos(Kappa)\*cos(Omega)\*sin(Phi)) - (Y - Y0)\*(cos(Omega)\*sin(Kappa) - cos(Kappa)\*sin(Omega)\*sin(Phi)) + cos(Kappa)\*cos(Phi)\*(X - X0)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))^2), (2768668935719305\*(sin(Kappa)\*sin(Omega) + cos(Kappa)\*cos(Omega)\*sin(Phi)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))) - (2768668935719305\*cos(Omega)\*cos(Phi)\*((Z - Z0)\*(sin(Kappa)\*sin(Omega) + cos(Kappa)\*cos(Omega)\*sin(Phi)) - (Y - Y0)\*(cos(Omega)\*sin(Kappa) - cos(Kappa)\*sin(Omega)\*sin(Phi)) + cos(Kappa)\*cos(Phi)\*(X - X0)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))^2);**

**(2768668935719305\*cos(Phi)\*sin(Kappa))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))) + (2768668935719305\*sin(Phi)\*((Y - Y0)\*(cos(Kappa)\*cos(Omega) + sin(Kappa)\*sin(Omega)\*sin(Phi)) - (Z - Z0)\*(cos(Kappa)\*sin(Omega) - cos(Omega)\*sin(Kappa)\*sin(Phi)) + cos(Phi)\*sin(Kappa)\*(X - X0)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))^2), (2768668935719305\*(cos(Kappa)\*cos(Omega) + sin(Kappa)\*sin(Omega)\*sin(Phi)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))) - (2768668935719305\*cos(Phi)\*sin(Omega)\*((Y - Y0)\*(cos(Kappa)\*cos(Omega) + sin(Kappa)\*sin(Omega)\*sin(Phi)) - (Z - Z0)\*(cos(Kappa)\*sin(Omega) - cos(Omega)\*sin(Kappa)\*sin(Phi)) + cos(Phi)\*sin(Kappa)\*(X - X0)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))^2), - (2768668935719305\*(cos(Kappa)\*sin(Omega) - cos(Omega)\*sin(Kappa)\*sin(Phi)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))) - (2768668935719305\*cos(Omega)\*cos(Phi)\*((Y - Y0)\*(cos(Kappa)\*cos(Omega) + sin(Kappa)\*sin(Omega)\*sin(Phi)) - (Z - Z0)\*(cos(Kappa)\*sin(Omega) - cos(Omega)\*sin(Kappa)\*sin(Phi)) + cos(Phi)\*sin(Kappa)\*(X - X0)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))^2)];**

**end**

**AEO(2 \* i\_temp - 1 : 2 \* i\_temp, 6 \* image\_num - 5 : 6 \* image\_num) = [ - (2768668935719305\*cos(Kappa)\*cos(Phi))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))) - (2768668935719305\*sin(Phi)\*((Z - Z0)\*(sin(Kappa)\*sin(Omega) + cos(Kappa)\*cos(Omega)\*sin(Phi)) - (Y - Y0)\*(cos(Omega)\*sin(Kappa) - cos(Kappa)\*sin(Omega)\*sin(Phi)) + cos(Kappa)\*cos(Phi)\*(X - X0)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))^2), (2768668935719305\*(cos(Omega)\*sin(Kappa) - cos(Kappa)\*sin(Omega)\*sin(Phi)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))) + (2768668935719305\*cos(Phi)\*sin(Omega)\*((Z - Z0)\*(sin(Kappa)\*sin(Omega) + cos(Kappa)\*cos(Omega)\*sin(Phi)) - (Y - Y0)\*(cos(Omega)\*sin(Kappa) - cos(Kappa)\*sin(Omega)\*sin(Phi)) + cos(Kappa)\*cos(Phi)\*(X - X0)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))^2), (2768668935719305\*cos(Omega)\*cos(Phi)\*((Z - Z0)\*(sin(Kappa)\*sin(Omega) + cos(Kappa)\*cos(Omega)\*sin(Phi)) - (Y - Y0)\*(cos(Omega)\*sin(Kappa) - cos(Kappa)\*sin(Omega)\*sin(Phi)) + cos(Kappa)\*cos(Phi)\*(X - X0)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))^2) - (2768668935719305\*(sin(Kappa)\*sin(Omega) + cos(Kappa)\*cos(Omega)\*sin(Phi)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))), (2768668935719305\*((Y - Y0)\*(sin(Kappa)\*sin(Omega) + cos(Kappa)\*cos(Omega)\*sin(Phi)) + (Z - Z0)\*(cos(Omega)\*sin(Kappa) - cos(Kappa)\*sin(Omega)\*sin(Phi))))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))) - (2768668935719305\*(cos(Omega)\*cos(Phi)\*(Y - Y0) - cos(Phi)\*sin(Omega)\*(Z - Z0))\*((Z - Z0)\*(sin(Kappa)\*sin(Omega) + cos(Kappa)\*cos(Omega)\*sin(Phi)) - (Y - Y0)\*(cos(Omega)\*sin(Kappa) - cos(Kappa)\*sin(Omega)\*sin(Phi)) + cos(Kappa)\*cos(Phi)\*(X - X0)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))^2), (2768668935719305\*(cos(Kappa)\*cos(Omega)\*cos(Phi)\*(Z - Z0) - cos(Kappa)\*sin(Phi)\*(X - X0) + cos(Kappa)\*cos(Phi)\*sin(Omega)\*(Y - Y0)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))) + (2768668935719305\*(cos(Phi)\*(X - X0) + cos(Omega)\*sin(Phi)\*(Z - Z0) + sin(Omega)\*sin(Phi)\*(Y - Y0))\*((Z - Z0)\*(sin(Kappa)\*sin(Omega) + cos(Kappa)\*cos(Omega)\*sin(Phi)) - (Y - Y0)\*(cos(Omega)\*sin(Kappa) - cos(Kappa)\*sin(Omega)\*sin(Phi)) + cos(Kappa)\*cos(Phi)\*(X - X0)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))^2), -(2768668935719305\*((Y - Y0)\*(cos(Kappa)\*cos(Omega) + sin(Kappa)\*sin(Omega)\*sin(Phi)) - (Z - Z0)\*(cos(Kappa)\*sin(Omega) - cos(Omega)\*sin(Kappa)\*sin(Phi)) + cos(Phi)\*sin(Kappa)\*(X - X0)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0)));**

**- (2768668935719305\*cos(Phi)\*sin(Kappa))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))) - (2768668935719305\*sin(Phi)\*((Y - Y0)\*(cos(Kappa)\*cos(Omega) + sin(Kappa)\*sin(Omega)\*sin(Phi)) - (Z - Z0)\*(cos(Kappa)\*sin(Omega) - cos(Omega)\*sin(Kappa)\*sin(Phi)) + cos(Phi)\*sin(Kappa)\*(X - X0)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))^2), (2768668935719305\*cos(Phi)\*sin(Omega)\*((Y - Y0)\*(cos(Kappa)\*cos(Omega) + sin(Kappa)\*sin(Omega)\*sin(Phi)) - (Z - Z0)\*(cos(Kappa)\*sin(Omega) - cos(Omega)\*sin(Kappa)\*sin(Phi)) + cos(Phi)\*sin(Kappa)\*(X - X0)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))^2) - (2768668935719305\*(cos(Kappa)\*cos(Omega) + sin(Kappa)\*sin(Omega)\*sin(Phi)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))), (2768668935719305\*(cos(Kappa)\*sin(Omega) - cos(Omega)\*sin(Kappa)\*sin(Phi)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))) + (2768668935719305\*cos(Omega)\*cos(Phi)\*((Y - Y0)\*(cos(Kappa)\*cos(Omega) + sin(Kappa)\*sin(Omega)\*sin(Phi)) - (Z - Z0)\*(cos(Kappa)\*sin(Omega) - cos(Omega)\*sin(Kappa)\*sin(Phi)) + cos(Phi)\*sin(Kappa)\*(X - X0)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))^2), - (2768668935719305\*((Y - Y0)\*(cos(Kappa)\*sin(Omega) - cos(Omega)\*sin(Kappa)\*sin(Phi)) + (Z - Z0)\*(cos(Kappa)\*cos(Omega) + sin(Kappa)\*sin(Omega)\*sin(Phi))))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))) - (2768668935719305\*(cos(Omega)\*cos(Phi)\*(Y - Y0) - cos(Phi)\*sin(Omega)\*(Z - Z0))\*((Y - Y0)\*(cos(Kappa)\*cos(Omega) + sin(Kappa)\*sin(Omega)\*sin(Phi)) - (Z - Z0)\*(cos(Kappa)\*sin(Omega) - cos(Omega)\*sin(Kappa)\*sin(Phi)) + cos(Phi)\*sin(Kappa)\*(X - X0)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))^2), (2768668935719305\*(cos(Phi)\*sin(Kappa)\*sin(Omega)\*(Y - Y0) - sin(Kappa)\*sin(Phi)\*(X - X0) + cos(Omega)\*cos(Phi)\*sin(Kappa)\*(Z - Z0)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))) + (2768668935719305\*(cos(Phi)\*(X - X0) + cos(Omega)\*sin(Phi)\*(Z - Z0) + sin(Omega)\*sin(Phi)\*(Y - Y0))\*((Y - Y0)\*(cos(Kappa)\*cos(Omega) + sin(Kappa)\*sin(Omega)\*sin(Phi)) - (Z - Z0)\*(cos(Kappa)\*sin(Omega) - cos(Omega)\*sin(Kappa)\*sin(Phi)) + cos(Phi)\*sin(Kappa)\*(X - X0)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0))^2), (2768668935719305\*((Z - Z0)\*(sin(Kappa)\*sin(Omega) + cos(Kappa)\*cos(Omega)\*sin(Phi)) - (Y - Y0)\*(cos(Omega)\*sin(Kappa) - cos(Kappa)\*sin(Omega)\*sin(Phi)) + cos(Kappa)\*cos(Phi)\*(X - X0)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0)))];**

**w(2 \* i\_temp - 1, 1) = x + (2768668935719305\*((Z - Z0)\*(sin(Kappa)\*sin(Omega) + cos(Kappa)\*cos(Omega)\*sin(Phi)) - (Y - Y0)\*(cos(Omega)\*sin(Kappa) - cos(Kappa)\*sin(Omega)\*sin(Phi)) + cos(Kappa)\*cos(Phi)\*(X - X0)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0)));**

**w(2 \* i\_temp, 1) = y + (2768668935719305\*((Y - Y0)\*(cos(Kappa)\*cos(Omega) + sin(Kappa)\*sin(Omega)\*sin(Phi)) - (Z - Z0)\*(cos(Kappa)\*sin(Omega) - cos(Omega)\*sin(Kappa)\*sin(Phi)) + cos(Phi)\*sin(Kappa)\*(X - X0)))/(18014398509481984\*(cos(Omega)\*cos(Phi)\*(Z - Z0) - sin(Phi)\*(X - X0) + cos(Phi)\*sin(Omega)\*(Y - Y0)));**

**end**

**A = [AEO, Ag];**

**%-I3\*3 For 4 Full CPs.**

**for i = 1 : 4**

**A(328 + 3 \* i - 2 : 328 + 3 \* i, 14 \* 6 + 50 \* 3 + 3 \* i - 2 : 14 \* 6 + 50 \* 3 + 3 \* i) = -eye(3);**

**end**

**%-I6\*6 For EOPs**

**for i = 1 : 14**

**A(340 + 6 \* i - 5 : 340 + 6 \* i, 6 \* i - 5 : 6 \* i) = -eye(6);**

**end**

**delta\_x = -inv(A' \* P \* A) \* A' \* P \* w;**

**for i = 1 : 14%EOPs**

**result\_images(i, 3 : 8) = delta\_x(i \* 6 - 5 : i \* 6, 1)' + result\_images(i, 3 : 8);**

**end**

**for i = 1 : 50%Tie**

**result\_coordinates(i, 2 : 4) = result\_coordinates(i, 2 : 4) + delta\_x(14 \* 6 + 3 \* i - 2 : 14 \* 6 + 3 \* i)';**

**end**

**for i = 1 : 4%Full**

**block\_control\_points\_copy(i, 2 : 4) = delta\_x(14 \* 6 + 50 \* 3 + 3 \* i - 2 : 3 \* i + 14 \* 6 + 50 \* 3, 1)' + block\_control\_points\_copy(i, 2 : 4);**

**end**

**%Update W**

**for i = 1 : 4**

**w(328 + 3 \* i - 2 : 328 + 3 \* i, 1) = [block\_control\_points(i, 2) - block\_control\_points\_copy(i, 2); block\_control\_points(i, 3) - block\_control\_points\_copy(i, 3); block\_control\_points(i, 4) - block\_control\_points\_copy(i, 4)];**

**end**

**for i = 1 : 14**

**w(340 + 6 \* i - 5 : 340 + 6 \* i, 1) = -(result\_images(i, 3 : 8)' - result\_images\_copy(i, 3 : 8)');**

**end**

**norm(delta\_x);**

**end**

**disp('Norm delta\_x: ');**

**disp(norm(delta\_x));**

**%There are 4 GCPs:**

**for i = 1 : 4**

**X\_Computed(i, 1) = block\_control\_points\_copy(i, 2);**

**Y\_Computed(i, 1) = block\_control\_points\_copy(i, 3);**

**Z\_Computed(i, 1) = block\_control\_points\_copy(i, 4);**

**end**

**figure();**

**plot(X\_Computed, Y\_Computed, '\*r', 'MarkerSize', 10);%Computed Control Points.**

**hold on**

**plot(data\_control\_points(:, 2), data\_control\_points(:, 3), '\*g');%Origin Control Points.**

**text(data\_control\_points(:, 2), data\_control\_points(:, 3), int2str(data\_control\_points(:, 1)));**

**legend('Control Points Computed', 'Control Points');**

**for i = 1 : 4**

**X\_Main(i, 1) = block\_control\_points(i, 2);**

**Y\_Main(i, 1) = block\_control\_points(i, 3);**

**Z\_Main(i, 1) = block\_control\_points(i, 4);**

**end**

**Main\_Coordinates = table(X\_Main, Y\_Main, Z\_Main)**

**Computed\_Coordinates = table(X\_Computed, Y\_Computed, Z\_Computed)**

**Ran = result\_images(:, 1);**

**Image\_Num = result\_images(:, 2);**

**X0 = result\_images(:, 3);**

**Y0 = result\_images(:, 4);**

**Z0 = result\_images(:, 5);**

**Omega = result\_images(:, 6);**

**Phi = result\_images(:, 7);**

**Kappa = result\_images(:, 8);**

**Exterior\_Parameters = table(Ran, Image\_Num, X0, Y0, Z0, Omega, Phi, Kappa)**

**%% PLOT**

**% data\_control\_points(4, 4) = 0;**

**% data\_control\_points(10, 4) = 0;**

**% data\_control\_points(6, 2) = 0;**

**% data\_control\_points(6, 3) = 0;**

**%**

**% pair\_points\_matrix = zeros(20, 5);**

**% temp = 1;**

**% for i = 1 : 11**

**% if data\_control\_points(i, 2) == 0**

**% continue;**

**% end**

**% temp2 = 1;**

**% for j = 1 : len\_data**

**% if data(j, 3) == data\_control\_points(i, 1)**

**% if temp2 == 1**

**% pair\_points\_matrix(2 \* temp - 1, :) = data(j, 1 : 5);**

**% elseif temp2 == 2**

**% pair\_points\_matrix(2 \* temp, :) = data(j, 1 : 5);**

**% end**

**% temp2 = temp2 + 1;**

**% end**

**% end**

**% temp = temp + 1;**

**% end**

**% clear temp temp2;**

**%**

**% for i = 1 : 10**

**% temp\_image\_number1 = pair\_points\_matrix(2 \* i - 1, 2);**

**% temp\_image\_number2 = pair\_points\_matrix(2 \* i, 2);**

**% if pair\_points\_matrix(2 \* i - 1, 1) == 2**

**% temp\_image\_number1 = temp\_image\_number1 + 7;**

**% end**

**% if pair\_points\_matrix(2 \* i, 1) == 2**

**% temp\_image\_number2 = temp\_image\_number2 + 7;**

**% end**

**%**

**% omegal = result\_images(temp\_image\_number1, 6);**

**% omegar = result\_images(temp\_image\_number2, 6);**

**% phil = result\_images(temp\_image\_number1, 7);**

**% phir = result\_images(temp\_image\_number2, 7);**

**% kappal = result\_images(temp\_image\_number1, 8);**

**% kappar = result\_images(temp\_image\_number2, 8);**

**% Ml = [cos(kappal), -sin(kappal), 0;**

**% sin(kappal), cos(kappal), 0;**

**% 0, 0, 1;] \* [cos(phil), 0, sin(phil); 0, 1, 0; -sin(phil), 0, cos(phil)] \* ...**

**% [1, 0, 0; 0, cos(omegal), -sin(omegal); 0, sin(omegal), cos(omegal)];**

**% Mr = [cos(kappar), -sin(kappar), 0;**

**% sin(kappar), cos(kappar), 0;**

**% 0, 0, 1;] \* [cos(phir), 0, sin(phir); 0, 1, 0; -sin(phir), 0, cos(phir)] \* ...**

**% [1, 0, 0; 0, cos(omegar), -sin(omegar); 0, sin(omegar), cos(omegar)];**

**% L = Ml' \* [pair\_points\_matrix(2 \* i - 1, 4); pair\_points\_matrix(2 \* i - 1, 5); -C];**

**% R = Mr' \* [pair\_points\_matrix(2 \* i, 4); pair\_points\_matrix(2 \* i, 5); -C];**

**%**

**% K = ((result\_images(temp\_image\_number2, 3) - result\_images(temp\_image\_number1, 3)) \* L(2) - ...**

**% (result\_images(temp\_image\_number2, 4) - result\_images(temp\_image\_number1, 4)) \* L(1)) / ...**

**% (R(2)\*L(1) - L(2)\*R(1));**

**% X(i) = K \* R(1) + result\_images(temp\_image\_number2, 3);**

**% Y(i) = K \* R(2) + result\_images(temp\_image\_number2, 4);**

**% Z(i) = K \* R(3) + result\_images(temp\_image\_number2, 5);**

**% end**

**% figure();**

**% plot(X, Y, '\*r', 'MarkerSize', 10);%Computed Control Points.**

**% hold on**

**% plot(data\_control\_points(:, 2), data\_control\_points(:, 3), '\*g');%Origin Control Points.**

**% text(data\_control\_points(:, 2), data\_control\_points(:, 3), int2str(data\_control\_points(:, 1)));**

**% legend('Control Points Computed', 'Control Points');**

**%**

**% clear temp\_image\_number1 temp\_image\_number2 kappal kappar Ml Mr L R K;**

**% clear temp\_image\_number1 temp\_image\_number2 kappal kappar Ml Mr L R K;**

**toc;**



