HACETTEPE UNIVERSITY
DEPARTMENT OF
GEOMATICS ENGINEERING

HACETTEPE UNIVERSITY
GEOMATICS ENGINEERING

# GMT431- PHOTOGRAMMETRIC IMAGE ANALYSIS

## Dr. Ali Özgün OK

## T.A. Dr. Beste TAVUS

## ASSIGNMENT -2

### Content




**ABDULSAMET TOPTAŞ – 21905024**

## 1-Explanations for Question 1

First of all, in this assignment, the Matlab programming language has been used. This choice is made because I believe that Matlab is more functional and easily applicable in computer vision and image processing due to its extensive features.

In this section, images captured from the same scene at different angles (Florence 1, 2, and 3 images) have been utilized, and the Fundamental matrix, which defines the relationship between these images, has been employed.

Firstly, in this section, I briefly imported the second image and, using the Fundamental matrix, plotted the five points I manually selected on the third image. I visualized these points with different line colors and drew the epipolar lines.

```
1     % 1.Question -- Abdulsamet TOPTAŞ (21905024)
2
3     % Loading images - images must be in the same file path
4 -   im2 = imread('florence2.jpg');
5 -   im3 = imread('florence3.jpg');
6
7     % Fundamental Matrix for Florence2 and Florence3
8 -   Fmatrix23 = [3.03994528999160e-08, 2.65672654114295e-07, -0.000870550254997210;
9                  4.67606901933558e-08, -1.11709498607089e-07, -0.00169128012255720;
10                 -1.38310618285550e-06, 0.00140690091935593, 0.999997201170569];
11
```

*Figure 1*

The code in lines 4-5 of Figure 1 loads the images provided for Assignment 2. Additionally, the code in lines 8-10 represents the Fundamental matrix, which establishes the relationship between two images necessary for drawing epipolar lines. This matrix was provided to us in Assignment 2.

```
% Number of points to select
numPoints = 5;

% Starting figure for manual point selection
figure;
imshow(im2);
title('Select 5 Points in Florence 2');

% Getting manually selected points
points2 = ginput(numPoints);

% Displaying selected points in Florence2
hold on;
plot(points2(:,1), points2(:,2), 'ro', 'MarkerSize', 10, 'LineWidth', 3);
hold off;
```

*Figure 2*

In Figure 2, the "ginput" command was applied to manually select five points in the Florence2 image, and the selected points were graphically displayed. The variable "numPoints" is a parameter that specifies the number of points you want to select. By clicking on a figure window with the mouse, you can select points equal to "numPoints". The selected points are stored in a matrix named "points2". Each row of this matrix contains the x and y coordinates of a point. These selected points are later used for calculating the epipolar lines.

```
% Compute epipolar lines
lines3 = Fmatrix23 * [points2, ones(numPoints, 1)]';

% Displaying corresponding epipolar lines in Florence3
figure;
imshow(im3);
title('Corresponding Epipolar Lines in Florence 3');

hold on;
for i = 1:numPoints
    % Compute the endpoints of the line segment
    x = [1 size(im3, 2)];
    y = (-lines3(1,i)*x - lines3(3,i))/lines3(2,i);

    % Plot the epipolar line
    plot(x, y, 'LineWidth', 3, 'Color', rand(1,3));
end
hold off;
```

*Figure 3*

In Figure 3, the corresponding epipolar lines for the selected five points in the Florence3 image are calculated and visualized. Each line is displayed as a randomly colored line. The variable "lines3" contains the calculated epipolar lines using the coordinates of the selected five points in the Florence2 image. These lines represent the epipolar geometry between the points in the Florence2 and Florence3 images. Each column of the matrix "lines3" represents the epipolar line for a point.

The "for loop" seen in Figure 3 is used to draw the epipolar lines of each point in the Florence3 image. During iterations, the two ends (x and y coordinates) of each line are used to determine the position of the epipolar line in the Florence3 image. With this information, the "plot" function is used to draw each line. The lines are drawn in random colors with a thickness of 3 ('LineWidth', 3), making each epipolar line clearly visualized in the Florence3 image.

In Figure 4, the result of the Matlab code I created is displayed. On the left image, five points have been manually selected for Florence 2, and automatically, Florence 3 has opened. The corresponding epipolar lines for the selected points in Florence 2 are projected as lines of different colors in Florence 3.

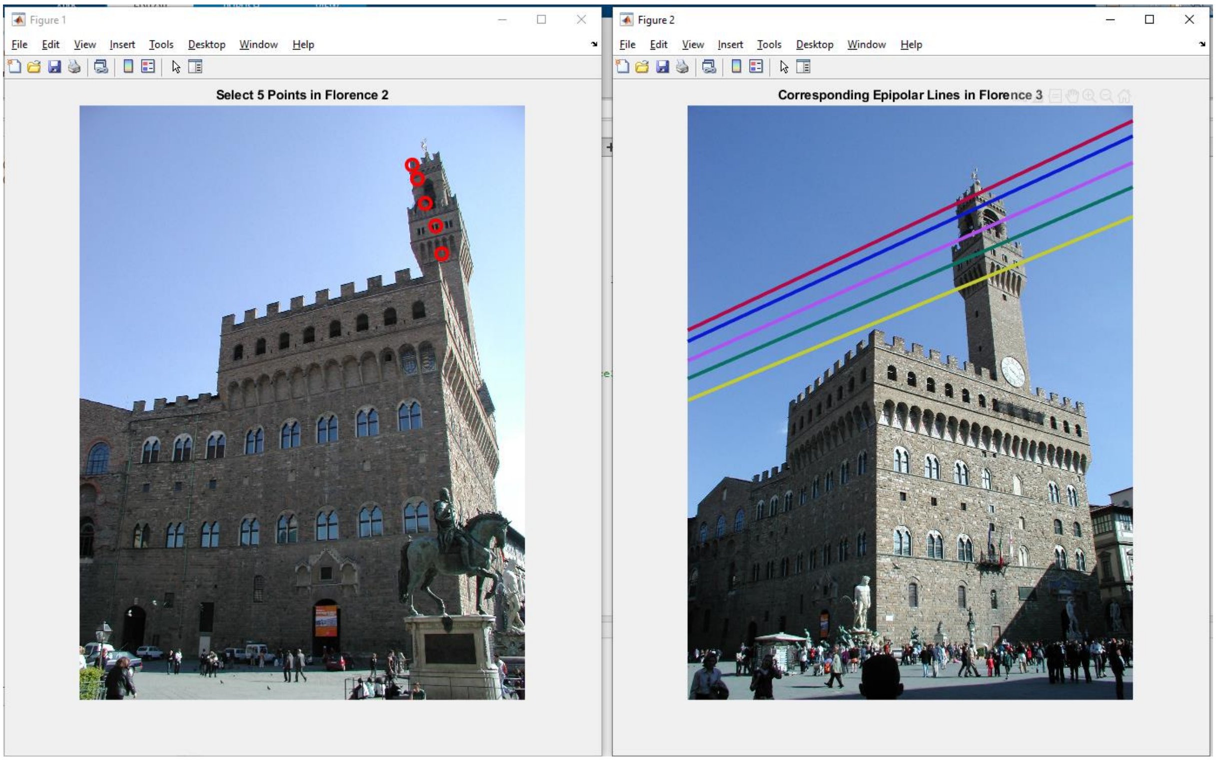


*Figure 4*

## 2-Explanations for Question 2

In this section, the first two images from the provided set (Florence 1 and 2) are loaded. Five points are manually selected in these two images to be projected onto the third image using epipolar geometry. Subsequently, the points matched in the first two images are projected onto the third image as epipolar lines with different colors and drawn.

```
% 2.Question -- Abdulsamet TOPTAŞ (21905024)

% Loading images - images must be in the same file path
im1 = imread('florence1.jpg');
im2 = imread('florence2.jpg');
im3 = imread('florence3.jpg');

% Fundamental Matrix for [Florence1 and Florence3] and [Florence2 and Florence3]
Fmatrix13 = [6.04444985855117e-08, 2.56726410274219e-07, -0.000602529673152695;
             2.45555247713476e-07, -8.38811736871429e-08, -0.000750892330636890;
            -0.000444464396704832, 0.000390321707113558, 0.999999361609429];

Fmatrix23 = [3.03994528999160e-08, 2.65672654114295e-07, -0.000870550254997210;
             4.67606901933558e-08, -1.11709498607089e-07, -0.00169128012255720;
            -1.38310618285550e-06, 0.00140690091935593, 0.999997201170569];

```

*Figure 5*

The lines seen in Figure 5, specifically lines 4-6, load the three images required for this section's solution. Lines 9-15 define the Fundamental matrix needed for epipolar geometry between the first and third images, and then the matrix required for epipolar geometry between the second and third images. These images and matrices were provided to us in Assignment 2.

```
% Number of points to select
numPoints = 5;

% Starting figure for manual point selection
figure;
imshow(im1);
title('Select 5 Points in Florence 1');

% Getting manually selected points
points1 = ginput(numPoints);

% Starting figure for manual point selection
figure;
imshow(im2);
title('Select 5 Points in Florence 2''(same location as im1)');

% Getting manually selected points
points2 = ginput(numPoints);
```

*Figure 6*

In Figure 6, the "ginput" command was applied to manually select five points in the Florence1 and Florence 2 images, and the selected points were graphically displayed. The variable "numPoints" is a parameter that specifies the number of points you want to select. By clicking on a figure window with the mouse, you can select points equal to "numPoints". The selected points are stored in matrices named "points1" and “points2”. Each row of this matrices contains the x and y coordinates of a point. These selected points are later used for calculating the epipolar lines.

```
% Match selected points
matchedPoints = [points1, points2];

% Calculate epipolar lines
lines3_from_im1 = Fmatrix13 * [matchedPoints(:, 1:2), ones(numPoints, 1)]';
lines3_from_im2 = Fmatrix23 * [matchedPoints(:, 3:4), ones(numPoints, 1)]';
```

*Figure 7*

In Figure 7, the points that match between the selected points are saved in the "matchedPoints" matrix. Then, using these matching points, the epipolar lines in the Florence3 image corresponding to points in the Florence1 and Florence2 images are calculated.

matchedPoints(:, 1:2) and matchedPoints(:, 3:4): These expressions select columns within the "matchedPoints" matrix. These columns contain the x and y coordinates of the points manually selected by the user. The first row contains the x and y coordinates of the points selected from Florence1, and the second row contains the x and y coordinates of the points selected from Florence2.

ones(numPoints, 1): This creates a column vector with the size of the selected point count (numPoints) and sets all values to 1. This represents homogeneous coordinates.

[matchedPoints(:, 1:2), ones(numPoints, 1)] and [matchedPoints(:, 3:4), ones(numPoints, 1)]: These two expressions create two matrices representing the homogeneous coordinates of the selected points.

Fmatrix13 * [matchedPoints(:, 1:2), ones(numPoints, 1)]': This calculates the epipolar lines representing vectors represented by the homogeneous coordinates of the points selected from Florence1 and Florence3 images.

Fmatrix23 * [matchedPoints(:, 3:4), ones(numPoints, 1)]': This calculates the epipolar lines representing vectors represented by the homogeneous coordinates of the points selected from Florence2 and Florence3 images.

```
% Show corresponding epipolar lines in Florence 3
figure;
imshow(im3);
title('Corresponding Epipolar Lines in Florence 3');

hold on;
for i = 1:numPoints
    % Calculate endpoints of line
    x = [1 size(im3, 2)];

    % Drawing epipolar lines of selected points from Florence 1
    y1 = (-lines3_from_im1(1,i)*x - lines3_from_im1(3,i))/lines3_from_im1(2,i);

    % Drawing epipolar lines of selected points from Florence 2
    y2 = (-lines3_from_im2(1,i)*x - lines3_from_im2(3,i))/lines3_from_im2(2,i);

    % Show two lines on the same chart
    plot(x, y1, 'LineWidth', 3, 'Color', rand(1,3));
    plot(x, y2, 'LineWidth', 3, 'Color', rand(1,3));
end
hold off;
```

*Figure 8*

In Figure 8, the calculated epipolar lines are visualized in the Florence3 image. Each line corresponds to the points selected from Florence1 and Florence2 images. Within the for loop, both lines are drawn on the same plot, each highlighted with different colors.

Figure 9 displays the result of the Matlab code you created. Florence 1 and Florence 2 images are loaded sequentially, and manually, five points are selected in the same location. Afterwards, the Florence 3 image opens automatically, and the selected points are projected using epipolar geometry.

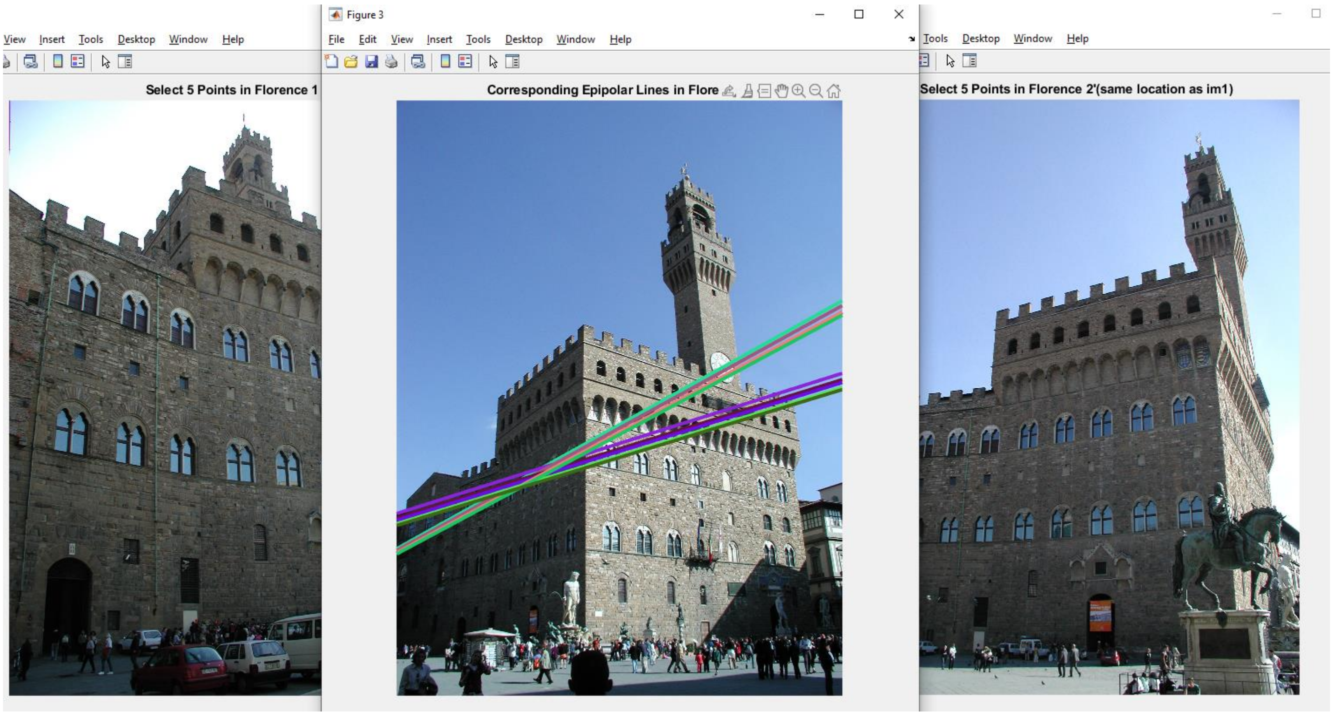


*Figure 9*

Finally, the Matlab code is executed, and the same point locations are selected from Florence 1 and Florence 2 images. Subsequently, as you can observe in Figure 9, you can see that a total of 10 points intersect in Florence 3.

## 3- Can you compute an accuracy for the projected points on im3 in Question 2? Comment on possible reasons for projection errors?

**Accuracy can be computed.** Matching accuracy depends on the accuracy of the provided Fundamental matrix. The accuracy of the Fundamental matrix can be calculated through various methods.

One of these methods is to calculate outliers using the RANSAC (Random Sample Consensus) algorithm and determine the accuracy of the matrix. This way, the precision and accuracy of the projected points can be calculated.

Another method is the 8-Point Algorithm. According to my research, for solving the system of equations in the formation of the Fundamental matrix from the selected points, the Singular Value Decomposition (SVD) method can be employed. The algorithm utilizes at least 8 matched pairs of point coordinates, solves a linear system of equations using these points, and obtains the fundamental matrix. The accuracy of the resulting matrix is high, allowing for the calculation of the precision and accuracy of the projected points.

Additionally, within the 8-Point Algorithm, the linear system of equations obtained using at least 8 matched pairs of points is typically solved using the least squares method. This solution provides the best estimate for the fundamental matrix.

Another method is the Robust LMedS (Least Median of Squares) Method. In the Robust LMedS method, the squares of distances to the epipolar line are calculated, and the least median value of these distance values is sought. In the LMedS method, erroneous point correspondences are eliminated (outliers removing), and the F matrix is calculated more effectively.

According to my research, for matching accuracy, we can use the Reprojection method. Reprojection errors can be measured by determining the 3D positions of matched points using triangulation, evaluating the accuracy of epipolar geometry. Smaller reprojection errors indicate more accurate matches and a more accurate depiction of epipolar geometry.

**Possible reasons for projection errors ;**

It may be due to the sensitivity of the fundamental matrix values. This is because the precision or accuracy of epipolar lines is dependent on the accuracy of the Fundamental matrix. The more accurately the Fundamental matrix is constructed, the more accurately the epipolar lines will represent the true locations.

Another possible source of error is selecting a small number of points for matching. The reliability of matching will decrease with fewer matched points, and selecting more corresponding points increases the accuracy in projection, reducing potential errors.

Another possible source of error is incorrect matching. In this assignment, manual matching has been performed, and it is crucial that the regions selected for manual matching in both images (im1 and im2) correspond to the same location. Mismatching, where corresponding points are not in the same position, can lead to projection errors.

Additionally, during manual matching, it is important for the distances between the points we select to be small, as this indicates the accuracy of the epipolar lines. If the distances between manually selected points are large, it can lead to projection errors. Keeping the distances between points small helps prevent potential errors in projection.

Projection errors can also occur when the correct matching algorithm is not selected. Additionally, low image resolution can contribute to projection errors because a low-resolution image makes accurate point matching more challenging.

Finally, improper calibration of the camera can also increase projection errors.