

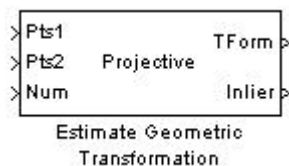
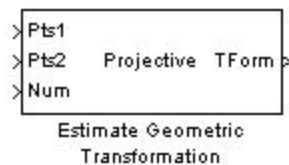
# Estimate Geometric Transformation

Estimate geometric transformation from matching point pairs

## Library

Geometric Transformations

## Description



Use the Estimate Geometric Transformation block to find the transformation matrix which maps the greatest number of point pairs between two images. A *point pair* refers to a point in the input image and its related point on the image created using the transformation matrix. You can select to use the RANdom SAMple Consensus (RANSAC) or the Least Median Squares algorithm to exclude outliers and to calculate the transformation matrix. You can also use all input points to calculate the transformation matrix.

Port	Input/Output	Supported Data Types	Complex Values Supported
<b>Pts1/Pts2</b>	2xN Matrix, (where <i>N</i> is the maximum number of points) coordinates of the input points	<ul style="list-style-type: none"> <li>Double</li> <li>Single</li> <li>8, 16, 32-bit signed integer</li> <li>8, 16, 32-bit unsigned integer</li> </ul>	No
<b>Num</b>	Scalar value that represents the number of valid points in Pts1 and Pts 2	<ul style="list-style-type: none"> <li>8, 16, 32-bit signed integer</li> <li>8, 16, 32-bit unsigned integer</li> </ul>	No
<b>TForm</b>	2x3 or 3x3, the transformation matrix	<ul style="list-style-type: none"> <li>Double</li> <li>Single</li> </ul>	No
<b>Inlier</b>	1xN, indicates which points have been used to calculate TForm	Boolean	No

Ports **Pts1** and **Pts2** are the points on two images that have the same data type. When **Pts 1** and **Pts 2** are single or double, the output transformation matrix will also have single or double data type. When **Pts1** and **Pts2** images are built-in integers, the option is available to set the transformation matrix data type to either **Single** or **Double**. The **TForm** output provides the transformation matrix. The **Inlier** output port provides the **Inlier** points on which the transformation matrix is based. This output appears when you select the **Output Boolean signal indication which point pairs are inliers** checkbox.

## RANSAC and Least Median Squares Algorithms

The RANSAC algorithm relies on a distance threshold. A pair of points,  $p_i^a$  (image *a*, Pts1) and  $p_i^b$  (image *b*, Pts 2) is

an inlier only when the distance between  $p_i^b$  and the projection of  $p_i^a$  based on the transformation matrix falls within the specified threshold. The distance metric used in the RANSAC algorithm is as follows:

$$d = \sum_{i=1}^{Num} \min(D(p_i^b, \psi(p_i^a : H)), t)$$

The Least Median Squares algorithm assumes at least 50% of the point pairs can be mapped by a transformation matrix. The algorithm does not need to explicitly specify the distance threshold. Instead, it uses the median distance between all input point pairs. The distance metric used in the Least Median of Squares algorithm is as follows:

$$d = \text{median}(D(p_1^b, \psi(p_1^a : H)), D(p_2^b, \psi(p_2^a : H)), \dots, D(p_{Num}^b, \psi(p_N^a : H)))$$

For both equations:

$p_i^a$  is a point in image  $a$  (Pts1)

$p_i^b$  is a point in image  $b$  (Pts2)

$\psi(p_i^a : H)$  is the projection of a point on image  $a$  based on transformation matrix  $H$

$D(p_i^b, p_j^b)$  is the distance between two point pairs on image  $b$

$t$  is the threshold

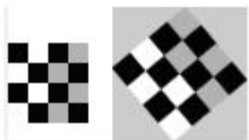
$Num$  is the number of points

The smaller the distance metric, the better the transformation matrix and therefore the more accurate the projection image.

## Transformations

The Estimate Geometric Transformation block supports `Nonreflective similarity`, `affine`, and `projective` transformation types, which are described in this section.

**Nonreflective similarity** transformation supports translation, rotation, and isotropic scaling. It has four degrees of freedom and requires two pairs of points.



The transformation matrix is:  $H = \begin{bmatrix} h_1 & h_2 & h_3 \\ -h_2 & h_1 & h_4 \end{bmatrix}$

The projection of a point  $[x \ y]^T$  by  $H$  is:  $[\hat{x} \ \hat{y}]^T = H[x \ y \ 1]^T$

**affine** transformation supports nonisotropic scaling in addition to all transformations that the nonreflective similarity transformation supports. It has six degrees of freedom that can be determined from three pairs of noncollinear points.



The transformation matrix is:  $H = \begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \end{bmatrix}$

The projection of a point  $[x \ y]^T$  by  $H$  is:  $[\hat{x} \ \hat{y}]^T = H[x \ y \ 1]^T$

**Projective** transformation supports tilting in addition to all transformations that the affine transformation supports.



$$h = \begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix}$$

The transformation matrix is :

The projection of a point  $[x \ y]^T$  by  $H$  is represented by homogeneous coordinates as:  $[\hat{u} \ \hat{v} \ \hat{w}]^T = H[x \ y \ 1]^T$

### Distance Measurement

For computational simplicity and efficiency, this block uses algebraic distance. The algebraic distance for a pair of points,  $[x^a \ y^a]^T$  on image  $a$ , and  $[x^b \ y^b]^T$  on image  $b$ , according to transformation  $H$ , is defined as follows;

For projective transformation:

$$D(P_i^b, \Psi(P_i^a : H)) = ((u^a - w^a x^b)^2 + (v^a - w^a y^b)^2)^{\frac{1}{2}}, \text{ where } [\hat{u}^a \ \hat{v}^a \ \hat{w}^a]^T = H[x^a \ y^a \ 1]^T$$

For Nonreflective similarity or affine transformation:  $D(P_i^b, \Psi(P_i^a : H)) = ((x^a - x^b)^2 + (y^a - y^b)^2)^{\frac{1}{2}},$

where  $[\hat{x}^a \ \hat{y}^a]^T = H[x^a \ y^a \ 1]^T$

### Algorithm

The block performs a comparison and repeats it M number of times between successive transformation matrices. If you select the **Find and exclude outliers** option, the RANSAC and Least Median Squares (LMS) algorithms become available. These algorithms calculate and compare a distance metric. The transformation matrix that produces the smaller distance metric becomes the new transformation matrix that the next comparison uses. A final transformation matrix is resolved when either:

- M number of random samplings is performed
- The RANSAC algorithm, when enough number of inlier point pairs can be mapped, (dynamically updating M)

The Estimate Geometric Transformation algorithm follows these steps:

1. A transformation matrix  $H$  is initialized to zeros
2. Set `count = 0` (Randomly sampling).
3. While `count < M`, where M is total number of random samplings to perform, perform the following;
  - a. Increment the count; `count = count + 1`.
  - b. Randomly select pair of points from images  $a$  and  $b$ , (2 pairs for Nonreflective similarity, 3 pairs for affine, or 4 pairs for projective).
  - c. Calculate a transformation matrix  $H$ , from the selected points.
  - d. If  $H$  has a distance metric less than that of  $H$ , then replace  $H$  with  $H$ .  
(Optional for RANSAC algorithm only)
    - i. Update M dynamically.
    - ii. Exit out of sampling loop if enough number of point pairs can be mapped by  $H$ .
4. Use all point pairs in images  $a$  and  $b$  that can be mapped by  $H$  to calculate a refined transformation matrix  $H$
5. Iterative Refinement, (Optional for RANSAC and LMS algorithms)
  - a. Denote all point pairs that can be mapped by  $H$  as inliers.
  - b. Use inlier point pairs to calculate a transformation matrix  $H$ .
  - c. If  $H$  has a distance metric less than that of  $H$ , then replace  $H$  with  $H$ , otherwise exit the loop.

### Number of Random Samplings

The number of random samplings can be specified by the user for the RANSAC and Least Median Squares

algorithms. You can use an additional option with the RANSAC algorithm, which calculates this number based on an accuracy requirement. The **Desired Confidence** level drives the accuracy.

The calculated number of random samplings,  $M$  used with the RANSAC algorithm, is as follows:

$$M = \frac{\log(1-p)}{\log(1-q^s)}$$

where

- $p$  is the probability of independent point pairs belonging to the largest group that can be mapped by the same transformation. The probability is dynamically calculated based on the number of inliers found versus the total number of points. As the probability increases, the number of samplings,  $M$ , decreases.
- $q$  is the probability of finding the largest group that can be mapped by the same transformation.
- $s$  is equal to the value 2, 3, or 4 for Nonreflective similarity, affine, and projective transformation, respectively.

### Iterative Refinement of Transformation Matrix

The transformation matrix calculated from all inliers can be used to calculate a refined transformation matrix. The refined transformation matrix is then used to find a new set of inliers. This procedure can be repeated until the transformation matrix cannot be further improved. This iterative refinement is optional.

### Dialog Box

**Function Block Parameters: Estimate Geometric Transformation**

Estimate Geometric Transformation

Find the transformation matrix that maps the largest number of points from Pts1 to Pts2.

The Pts1 and Pts2 inputs specify the location of points in two images. Each column in the Pts1 and Pts2 arrays has the format [row; column]. The points in the arrays must be ordered to form corresponding location pairs.

When the Find and exclude outliers option is selected, the block randomly selects matching point pairs until it computes a transformation matrix that fits a specified number of inliers, or until a user-specified stopping criterion is reached. It then refines this transformation matrix by using all inliers found in the first step.

Parameters

Transformation type: **Projective**

☒ Find and exclude outliers

Method: **Random Sample Consensus (RANSAC)**

Algebraic distance threshold for determining inliers: **1.5**

Determine number of random samplings using: **Specified value**

Number of random samplings: **100**

☐ Stop sampling earlier when a specified percentage of point pairs are determined to be inliers

☐ Perform additional iterative refinement of the transformation matrix

Outputs

☐ Output Boolean signal indicating which point pairs are inliers

When Pts1 and Pts2 are built-in integers, set transformation matrix data type to: **Single**

OK Cancel Help Apply

### Transformation Type

Specify transformation type, either `Nonreflective similarity`, `affine`, or `projective` transformation. If you select `projective` transformation, you can also specify a scalar algebraic distance threshold for determining inliers. If you select either `affine` or `projective` transformation, you can specify the distance threshold for determining inliers in pixels. See [Transformations](#) for a more detailed discussion.

**Find and exclude outliers**

When selected, the block finds and excludes outliers from the input points and uses only the inlier points to calculate the transformation matrix. When this option is not selected, all input points are used to calculate the transformation matrix.

**Method**

Select either the **RANdom SAMple Consensus (RANSAC)** or the **Least Median of Squares** algorithm to find outliers. See [RANSAC and Least Median Squares Algorithms](#) for a more detailed discussion.

**Algebraic distance threshold for determining inliers**

Specify a scalar threshold value for determining inliers. The threshold controls the upper limit used to find the algebraic distance in the RANSAC algorithm. This parameter appears when the **Method** parameter is **Random Sample Consensus (RANSAC)** and the **Transformation type** parameter is **projective**.

**Distance threshold for determining inliers (in pixels)**

Specify the upper limit distance a point can differ from the projection location of its associating point. This parameter appears when the **Method** parameter is set to **Random Sample Consensus (RANSAC)** and the value of the **Transformation type** parameter is set to **Nonreflective similarity** or **affine**.

**Determine number of random samplings using**

Select **Specified value** to enter a positive integer value for number of random samplings, or select **Desired confidence** to set the number of random samplings as a percentage and a maximum number. This parameter appears when you select **Find and exclude outliers** parameter, and the value of the **Method** parameter is **Random Sample Consensus (RANSAC)**.

**Number of random samplings**

Specify the number of random samplings for the algorithm to perform. This parameter appears when the value of the **Determine number of random samplings using** parameter is **Specified value**.

**Desired confidence (in %)**

Specify a percent by entering a number between 0 and 100. The **Desired confidence** is the probability to find the largest group of points that can be mapped by a transformation matrix. This parameter is visible when the **Determine number of random samplings using** is **Desired confidence**.

**Maximum number of random samplings**

Specify an integer number for the maximum number of random samplings. This parameter appears when the **Method** parameter is set to **Random Sample Consensus (RANSAC)** and the value of **Determine number of random samplings using** parameter is **Desired confidence**.

**Stop sampling earlier when a specified percentage of point pairs are determined to be inlier**

Specify to stop random sampling when a percentage of input points have been found as inliers. This parameter appears when the **Method** parameter is **Random Sample Consensus (RANSAC)**.

**Perform additional iterative refinement of the transformation matrix**

Specify whether to perform refinement on the transformation matrix. This parameter appears when you select **Find and exclude outliers** parameter.

**Output Boolean signal indicating which point pairs are inliers**

Select this option to output the inlier point pairs that were used to calculate the transformation matrix. This parameter appears when you select **Find and exclude outliers** parameter. This parameter is not used when the data type of points is signed or double.

**When Pts1 and Pts2 are built-in integers, set transformation matrix data type to**

Specify transformation matrix data type as **Single** or **Double** when the input points are built-in integers. This parameter is not used when the data type of points is signed or double.

Parameter Name	Default Value	Visibility	Tunability
<b>Transformation type</b>	projective	Always	No
<b>Find and exclude outliers</b>	Checked	Always	No

Parameter Name	Default Value	Visibility	Tunability
<b>Method</b>	RANSAC	Visible when <b>Find and exclude outliers</b> parameter is selected	No
<b>Algebraic distance threshold for determining inliers</b>	1.5	Visible when <b>Method</b> parameter is Random Sample Consensus (RANSAC) and the <b>Transformation type</b> parameter is projective	Yes
<b>Distance threshold for determining inliers (in pixels)</b>	1.5	Visible when <b>Method</b> parameter is Random Sample Consensus (RANSAC) and the <b>Transformation type</b> parameter is Nonreflective similarity or affine	Yes
<b>Determine number of random samplings</b>	Specified value	Visible when <b>Find and exclude outliers</b> parameter is selected	No
<b>Number of random samplings</b>	100	Visible when <b>Determine number of random samplings using</b> parameter is Specified value	Yes
<b>Maximum number of random samplings</b>	200	Visible when the <b>Method</b> parameter is set to Random Sample Consensus (RANSAC) and <b>Determine number of random samplings using</b> parameter is Desired confidence	Yes
<b>Desired confidence (in%)</b>	99	Visible when the <b>Determine number of random samplings using</b> is Desired confidence	Yes
<b>Stop sampling earlier when a specified percentage of point pairs are determined to be inliers</b>	Unchecked	Visible when the <b>Method</b> parameter is Random Sample Consensus (RANSAC)	No
<b>Inlier percentage</b>	75	Visible when <b>Stop sampling earlier when a specified percentage of point pairs are determined to be inliers</b> parameter is checked	Yes
<b>Perform additional iterative refinement of the transformation matrix</b>	Unchecked	Visible when <b>Find and exclude outliers</b> parameter is selected	No
<b>Output Boolean signal indicating which input points are inliers</b>	Unchecked	Visible when <b>Find and exclude outliers</b> parameter is selected	No

Parameter Name	Default Value	Visibility	Tunability
When Pts1 and Pts2 are built-in integers, set transformation matrix data type to	Single	Always	No

## Examples

### Calculate transformation matrix from largest group of point pairs

Examples of input data and application of the Estimate Geometric Transformation block appear in the following figures. Figures (a) and (b) show the point pairs. The points are denoted by stars or circles, and the numbers following them show how they are paired. Some point pairs can be mapped by the same transformation matrix. Other point pairs require a different transformation matrix. One matrix exists that maps the largest number of point pairs, the block calculates and returns this matrix. The block finds the point pairs in the largest group and uses them to calculate the transformation matrix. The point pairs connected by the magenta lines are the largest group.

The transformation matrix can then be used to stitch the images as shown in Figure (e).





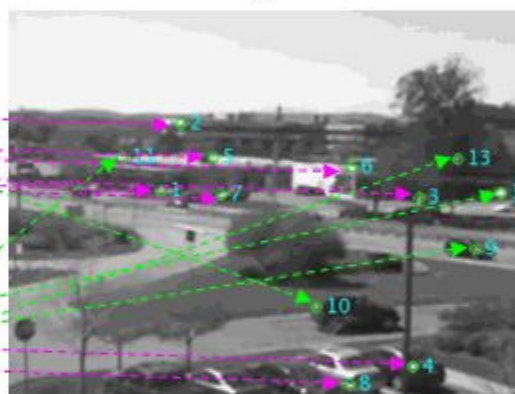
(a)



(b)



(c)



(d)



(e)

## Video Mosaicking

To see an example of the Estimate Geometric Transformation block used in a model with other blocks, see the **Video Mosaicking Demo**, [vipmosaicking](#).

## References

R. Hartley and A. Ziserman, "Multiple View Geometry in Computer Vision," Second edition, Cambridge University Press, 2003

## See Also

[cp2tform](#)

Image Processing Toolbox software

[vipmosaicking](#)

Video and Image Processing Blockset demo

[Provide feedback about this page](#)



 Erosion

Find Local Maxima 

© 1984-2009 The MathWorks, Inc. • [Terms of Use](#) • [Patents](#) • [Trademarks](#) • [Acknowledgments](#)