Image Alignment

Computer Vision

Adduru U G Sankararao, IIIT Sri City

Image alignment & Stitching

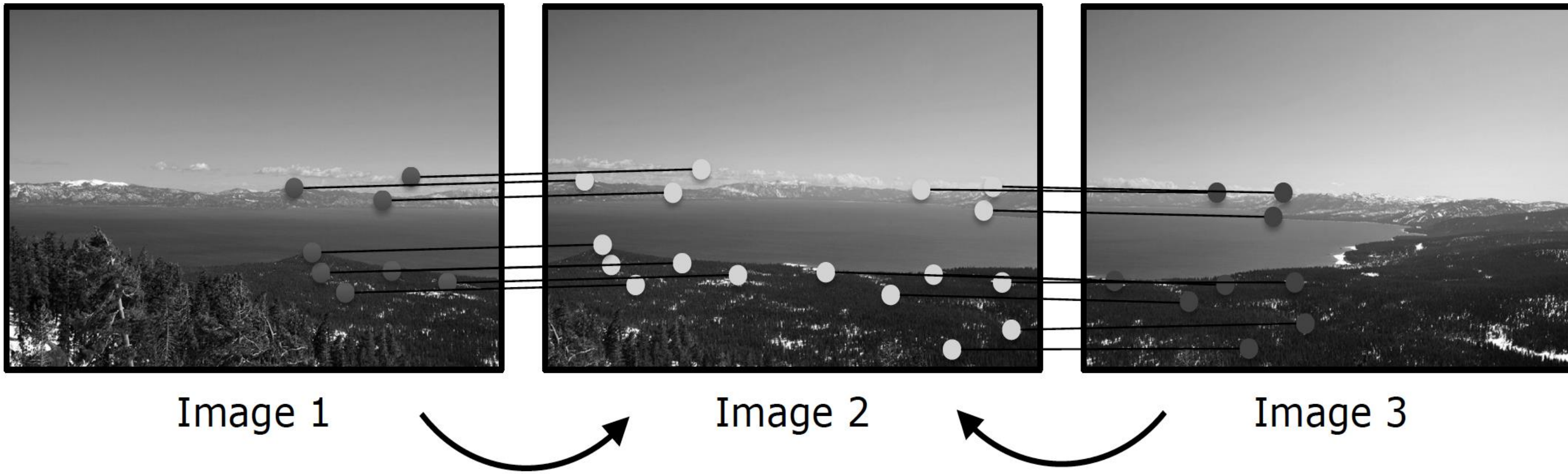
Image 1

Image 2

Image 3

How would you align these images?

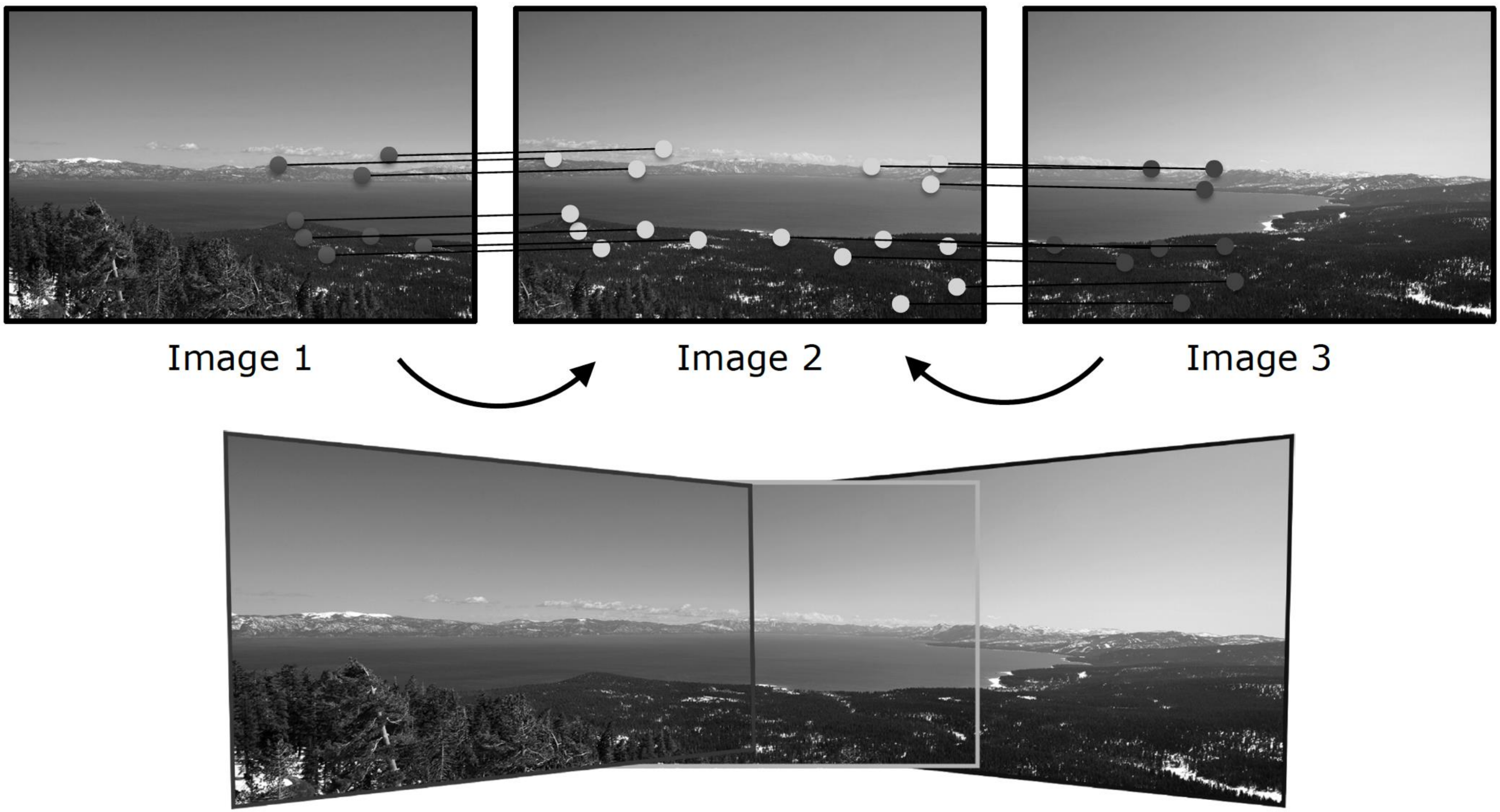
Image alignment and Stitching



Find corresponding points (using SIFT, etc.)

Find geometric relationship between the images

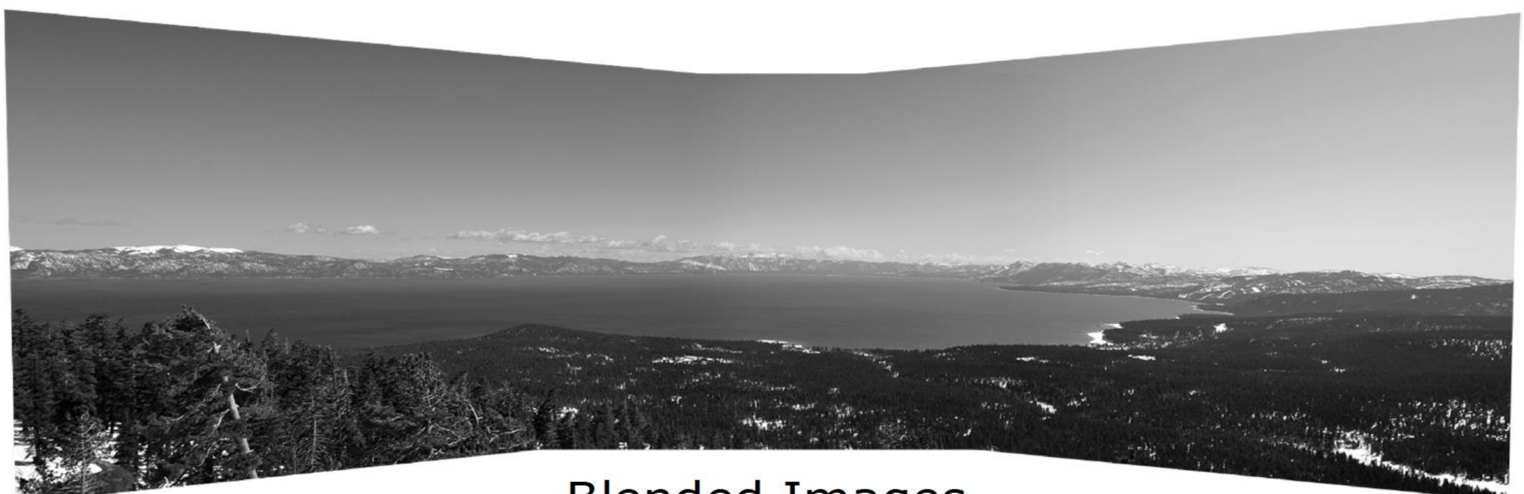
Image alignment and Stitching



Warp images so that corresponding points alig

Image alignment and Stitching

Overlaid Aligned Images



Blended Images

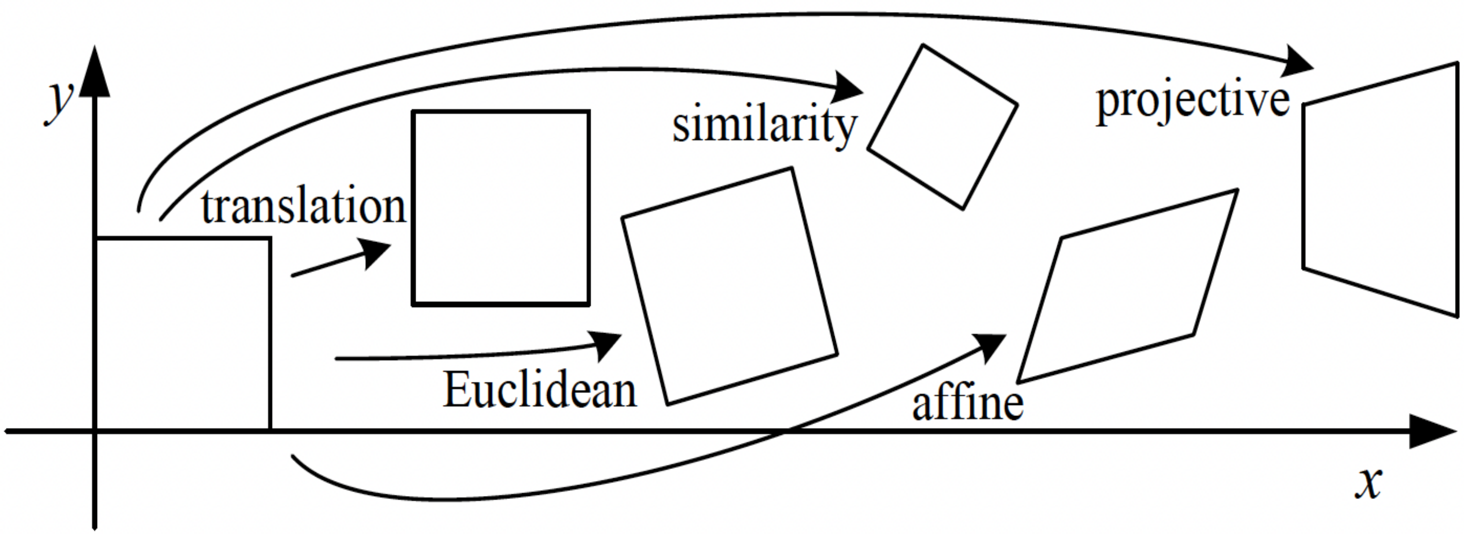
Blend images to remove hard seams

This Lecture

- Image/Geometric Transformations
- Least Square Fit
- Dealing with outliers: RANSAC
- Wrapping and Blending Images

Why Image Matching is Difficult ?

- should allow for a geometric transformation
- fitting the model to data (correspondences) is sensitive to outliers: should find a subset of inliers first
- finding inliers to a transformation requires finding the transformation in the first place
- correspondences have gross errors
- inliers are typically less than 50%



Image/geometric Transformations

- two images f, f' are equal at points $\mathbf{x}, \mathbf{x}'$

$$f(\mathbf{x}) = f'(\mathbf{x}')$$

- $\mathbf{x}$ is mapped to $\mathbf{x}'$

$$\mathbf{x}' = T(\mathbf{x})$$

- T is a bijection of $\mathbb{R}^2$ to itself:

$$T : \mathbb{R}^2 \to \mathbb{R}^2$$

Image/geometric Transformations

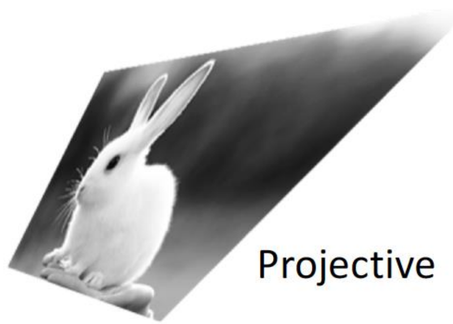
Translation

Rotation

Scaling and Aspect

$$\boxed{g(x,y) = f\big(T(x,y)\big)}$$

Affine



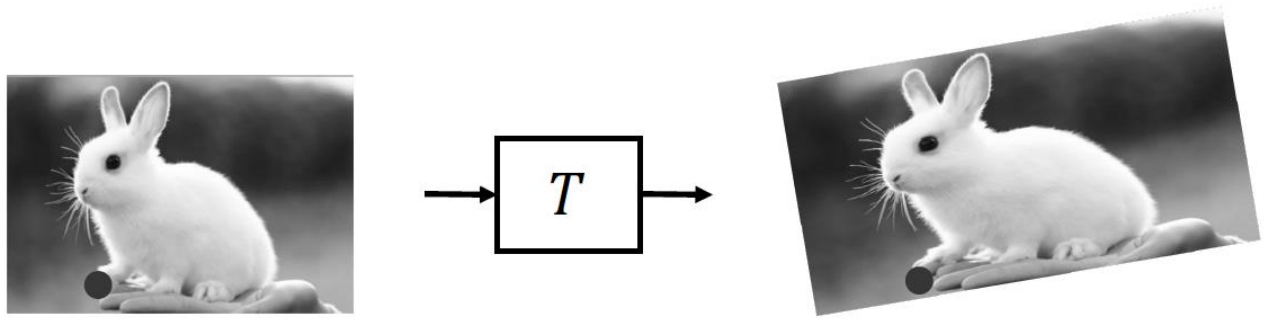
Projective

Barrel

Transformation T is the same over entire domain

Often can be described by just a few parameters

2x2 Linear Transformations

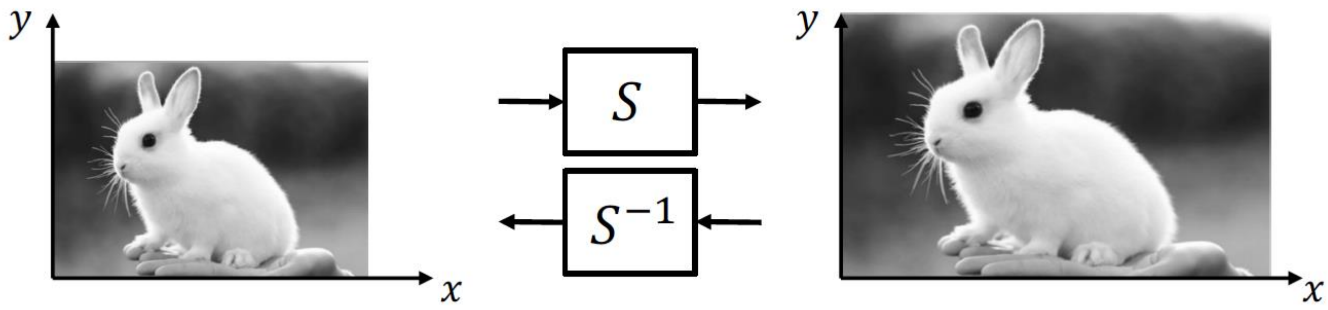


$\mathbf{p}_1 = (x_1, y_1)$ $\qquad$ $\mathbf{p}_2 = (x_2, y_2)$

T can be represented by a matrix.

$$\mathbf{p}_2 = T\mathbf{p}_1 \qquad \begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = T \begin{bmatrix} x_1 \\ y_1 \end{bmatrix} \qquad \begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \end{bmatrix}$$

2x2 Linear Transformations (Scaling)



Forward:

$$x_2 = a x_1 \qquad y_2 = b y_1$$

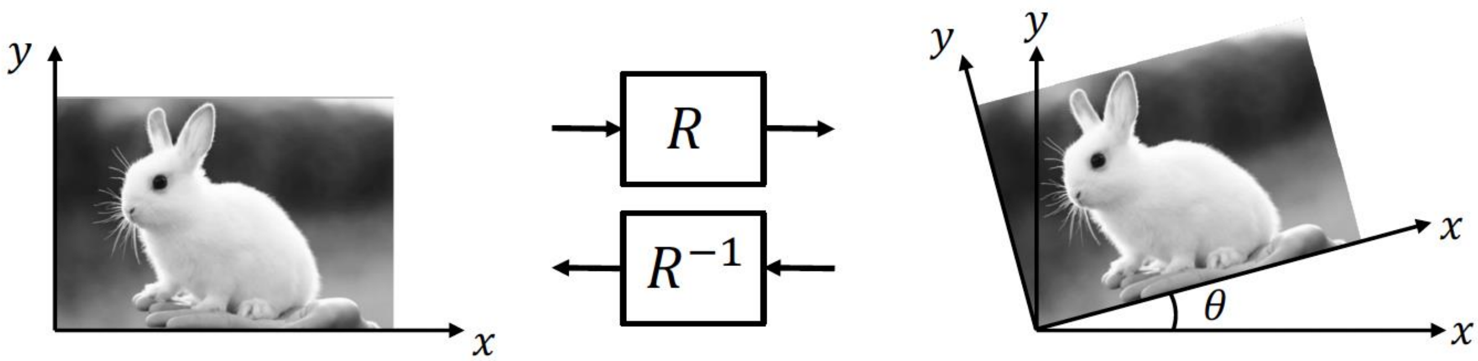
$$\begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = S \begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = \begin{bmatrix} a & 0 \\ 0 & b \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \end{bmatrix}$$

Inverse:

$$x_1 = \frac{1}{a} x_2 \qquad y_1 = \frac{1}{b} y_2$$

$$\begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = S^{-1} \begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = \begin{bmatrix} 1/a & 0 \\ 0 & 1/b \end{bmatrix} \begin{bmatrix} x_2 \\ y_2 \end{bmatrix}$$

Image Rotation



Forward:

$$x_2 = x_1 cos\theta - y_1 sin\theta$$

$$y_2 = x_1 sin\theta + y_1 cos\theta$$

$$\begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = R \begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = \begin{bmatrix} cos\theta & -sin\theta \\ sin\theta & cos\theta \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \end{bmatrix}$$

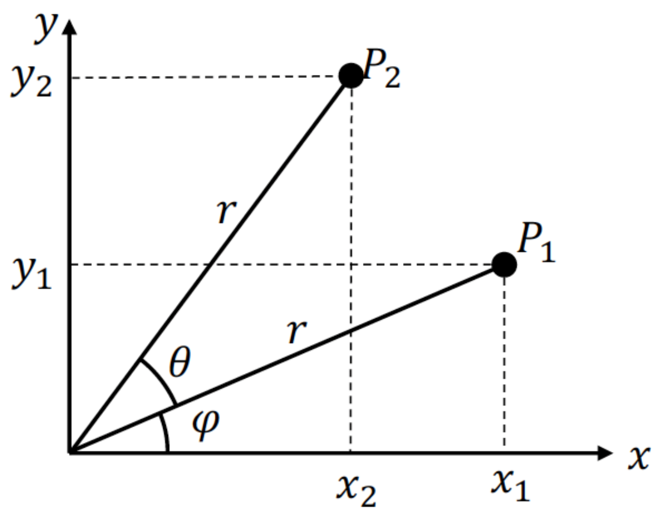
Inverse:

$$x_1 = x_2 cos\theta + y_2 sin\theta$$

$$y_1 = -x_2 sin\theta + y_2 cos\theta$$

$$\begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = R^{-1} \begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = \begin{bmatrix} cos\theta & sin\theta \\ -sin\theta & cos\theta \end{bmatrix} \begin{bmatrix} x_2 \\ y_2 \end{bmatrix}$$

Image Rotation

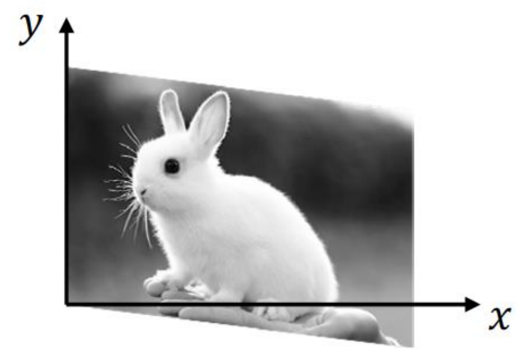


$$x_1 = r\cos(\varphi)$$
$$y_1 = r\sin(\varphi)$$

$$x_2 = r\cos(\varphi + \theta)$$
$$x_2 = r\cos\varphi\cos\theta - r\sin\varphi\sin\theta$$
$$x_2 = x_1\cos\theta - y_1\sin\theta$$

$$y_2 = r\sin(\varphi + \theta)$$
$$y_2 = r\cos\varphi\sin\theta + r\sin\varphi\cos\theta$$
$$y_2 = x_1\sin\theta + y_1\cos\theta$$

Image Skew



Horizontal Skew:

$$x_2 = x_1 + m_x y_1$$

$$y_2 = y_1$$

$$\begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = S_x \begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = \begin{bmatrix} 1 & m_x \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \end{bmatrix}$$

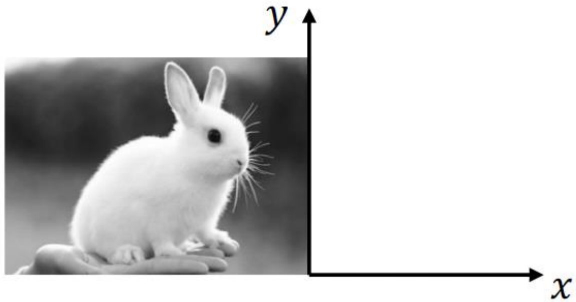
Vertical Skew:

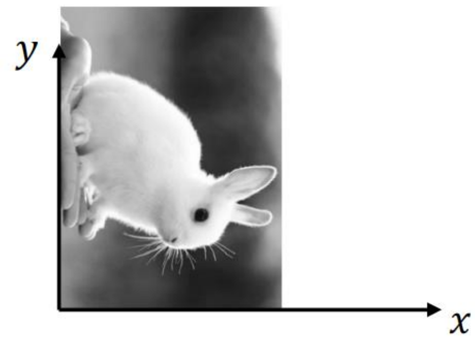
$$x_2 = x_1$$

$$y_2 = m_y x_1 + y_1$$

$$\begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = S_x \begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ m_y & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \end{bmatrix}$$

Image Transformations: Mirror





Mirror about Y-axis:

$$x_2 = -x_1$$

$$y_2 = y_1$$

$$M_y = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}$$

Mirror about line $y = x$:

$$x_2 = y_1$$

$$y_2 = x_1$$

$$M_{xy} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

2x2 Image Transformations

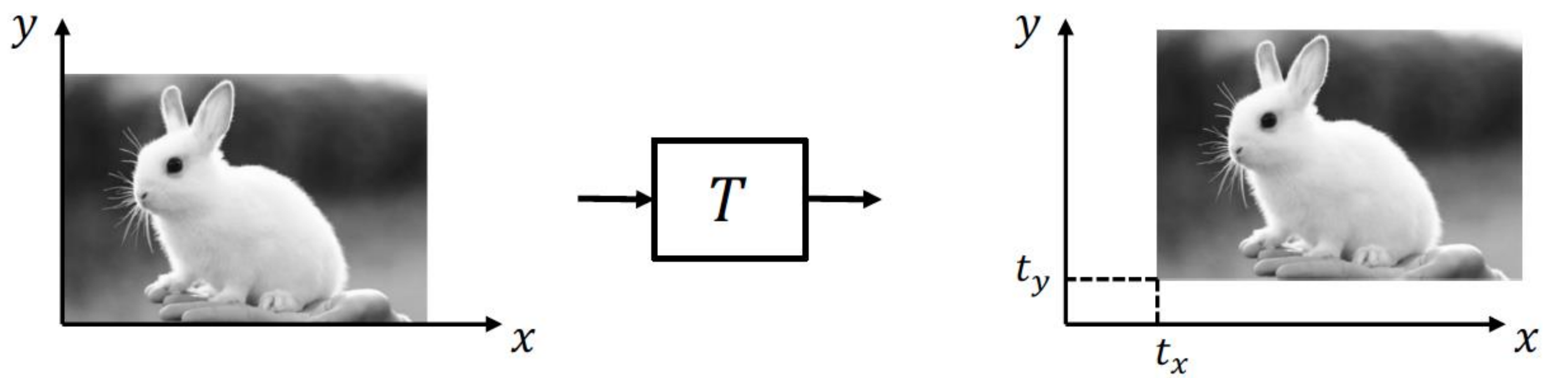
Any transformation of the form:

$$\begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \end{bmatrix}$$

- Origin maps to the origin
- Lines map to lines
- Parallel lines remain parallel
- Closed under composition

$$\left.\begin{aligned} \mathbf{p}_2 &= T_{21}\mathbf{p}_1 \\ \mathbf{p}_3 &= T_{32}\mathbf{p}_2 \\ \mathbf{p}_3 &= T_{31}\mathbf{p}_1 \end{aligned}\right\} \mathbf{p}_3 = T_{32}\mathbf{p}_2 = T_{32}T_{21}\mathbf{p}_1 \Rightarrow T_{31} = T_{32}T_{21}$$

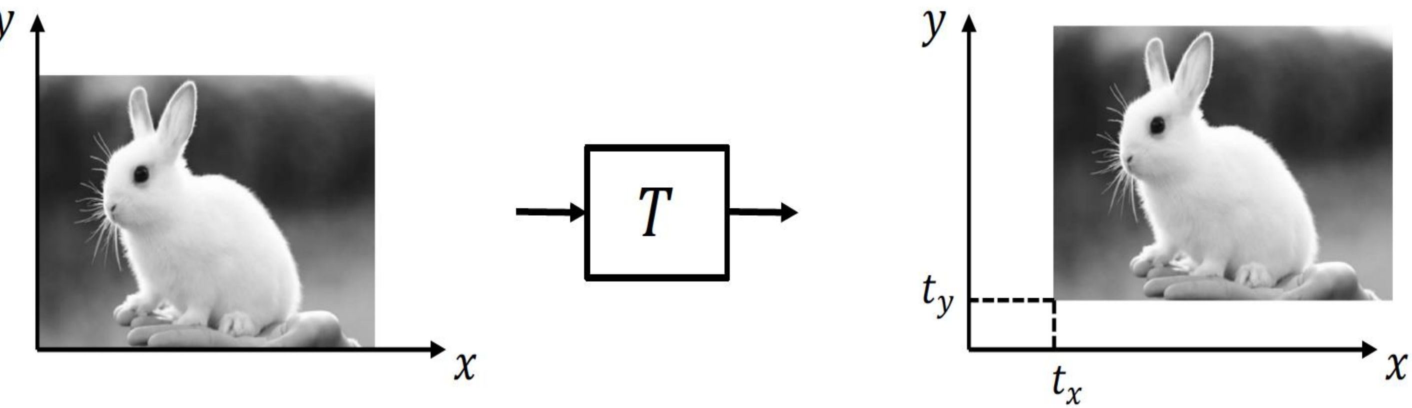
Image Translation



$$x_2 = x_1 + t_x \qquad y_2 = y_1 + t_y$$

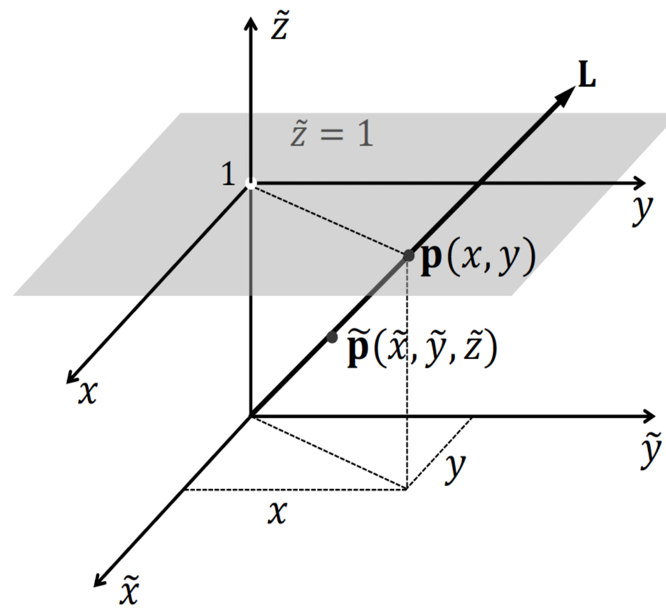
Can translation be expressed as a 2x2 matrix? No.

3x3 Transformations: Image Translation



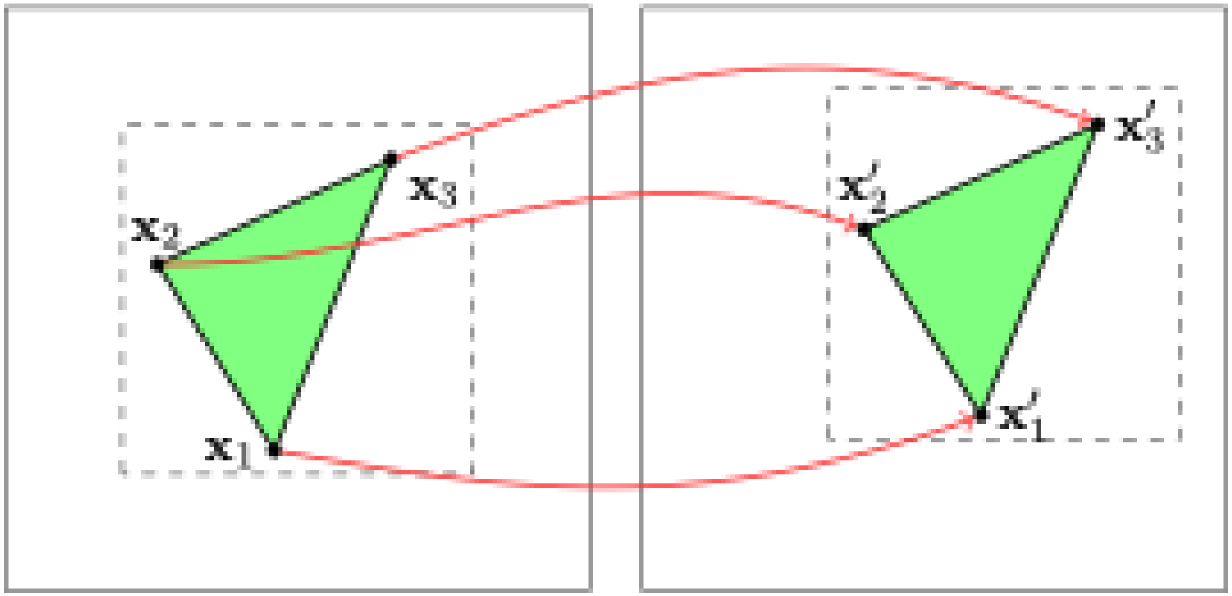
$$x_2 = x_1 + t_x \qquad y_2 = y_1 + t_y$$

$$\begin{bmatrix} x_2 \\ y_2 \\ 1 \end{bmatrix} \equiv \begin{bmatrix} \tilde{x}_2 \\ \tilde{y}_2 \\ \tilde{z}_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix}$$



The third coordinate $\tilde{z} \neq 0$ is fictitious

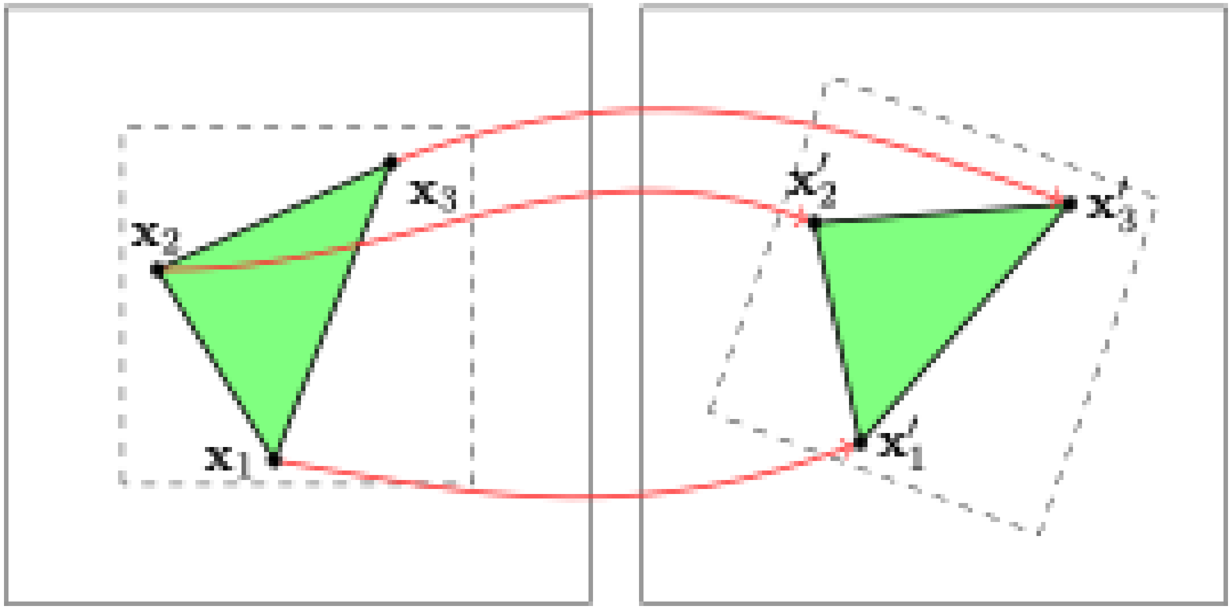
Image/geometric transformations



- translation: 2 degrees of freedom

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

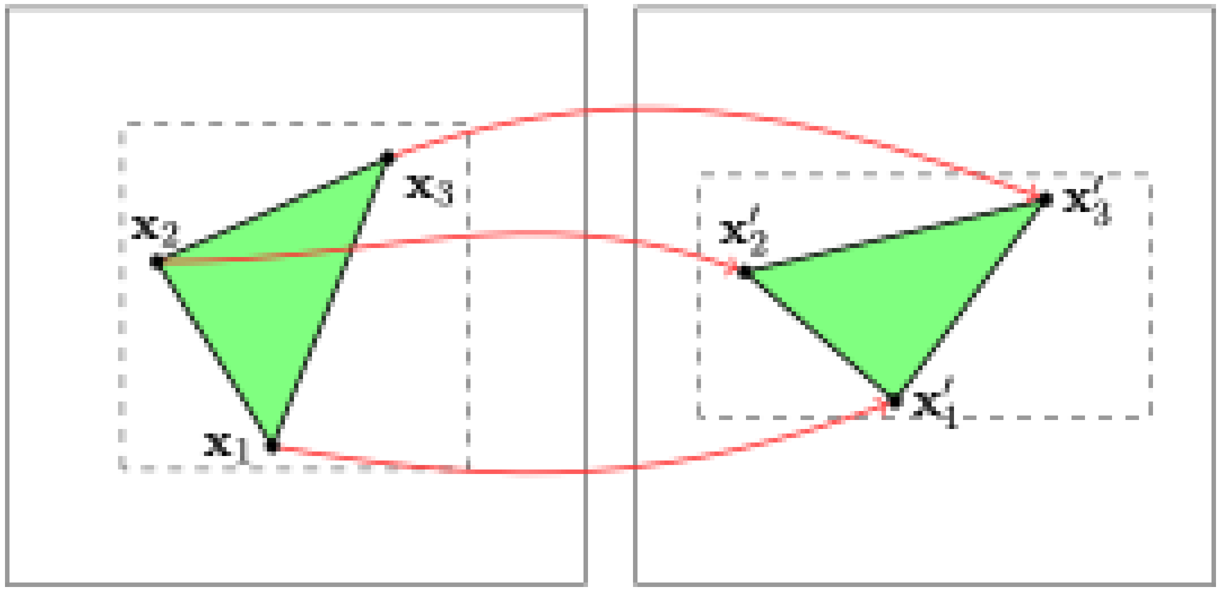
geometric transformations



- rotation: 1 degree of freedom

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

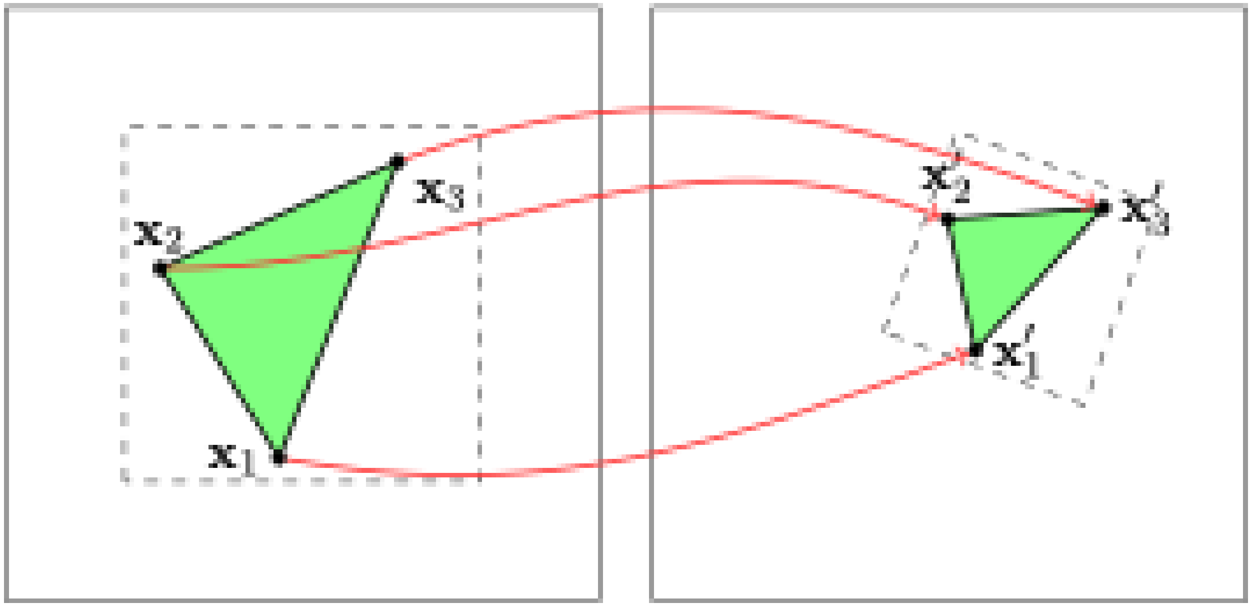
geometric transformations



- scale: 2 degrees of freedom

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

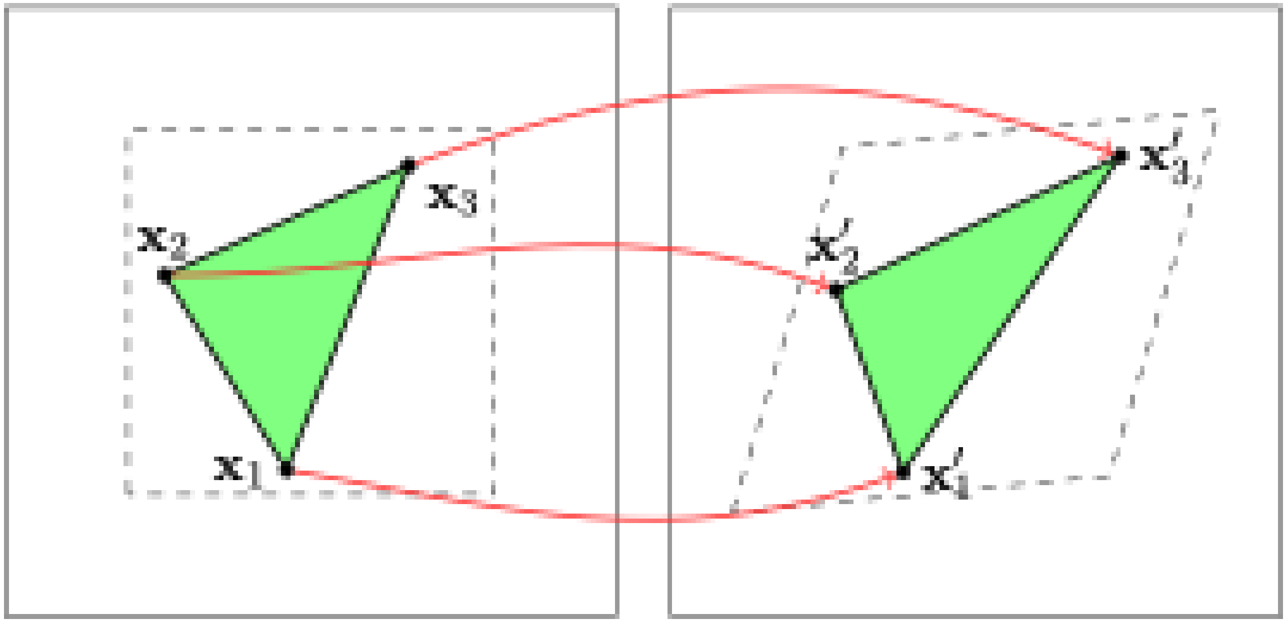
geometric transformations



- similarity: 4 degrees of freedom

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} r\cos\theta & -r\sin\theta & t_x \\ r\sin\theta & r\cos\theta & t_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

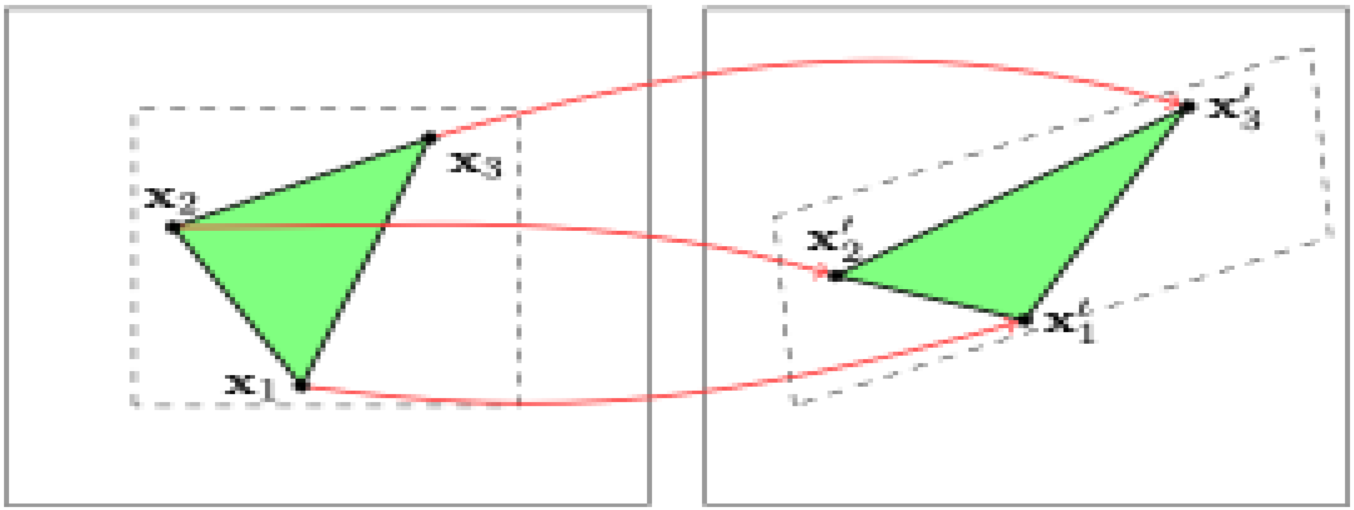
geometric transformations



- shear: 2 degrees of freedom

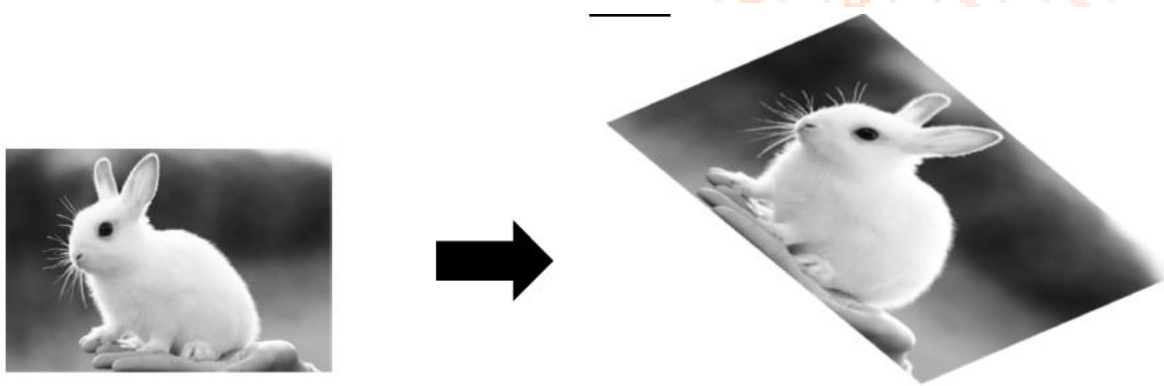
$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & b_x & 0 \\ b_y & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

geometric transformations



affine: 6 degrees of freedom

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$



Affine Transformation

Affine transformations are combinations of

- Linear transformations, and
- Translations

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

or

Properties of affine transformations:

- Lines map to lines
- Parallel lines remain parallel
- Ratios are preserved

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Image Alignment

Alignment: Find the parameters of the transformation (model) that best align matched points

- In all cases, the problem is transformed to a linear system (why?)

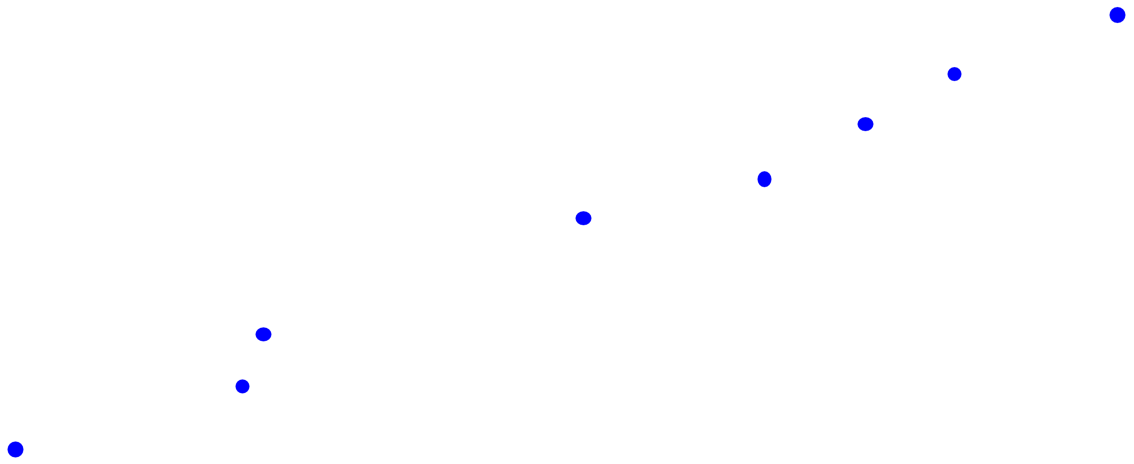
$$A\mathbf{x}=\mathbf{b}$$

where **x, b** contain coordinates of known point correspondences from images I, I′ respectively, and **A** contains our model parameters

- We need n=⌈d/2⌉ correspondences, where *d* are the degrees of freedom of our model

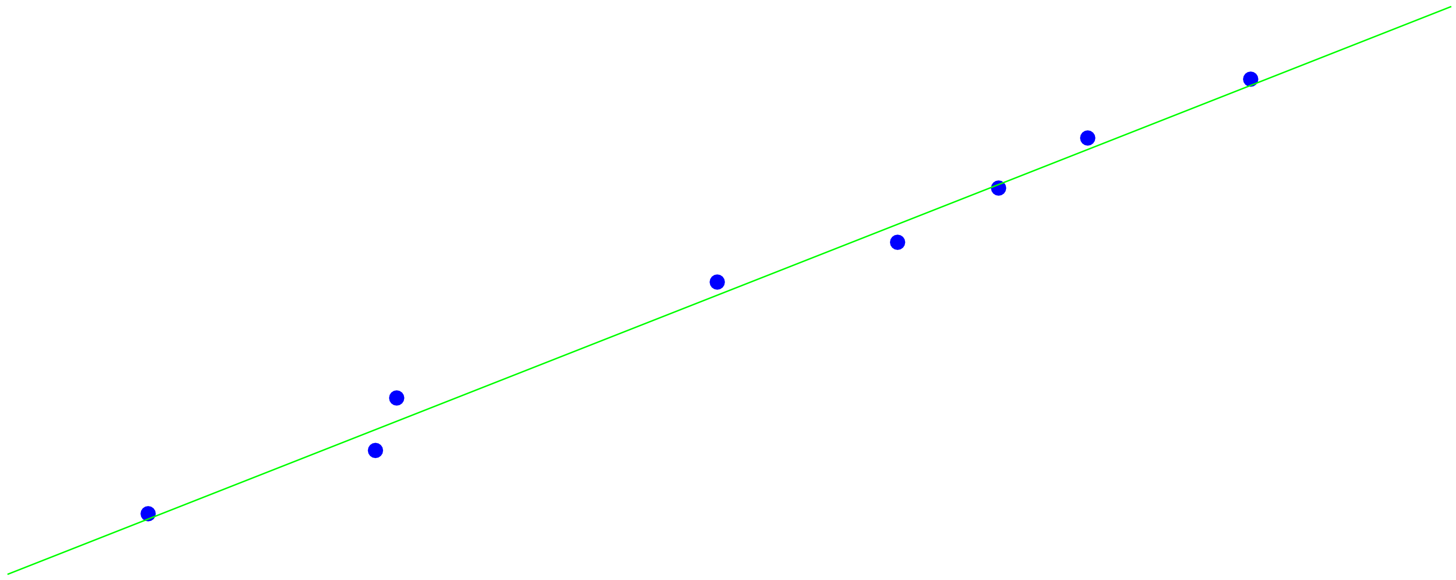
- Let's take the simplest model as an example: *fit a line to two points*

Least squares line fitting



- clean data, no outliers : least squares fit ok

Least squares line fitting

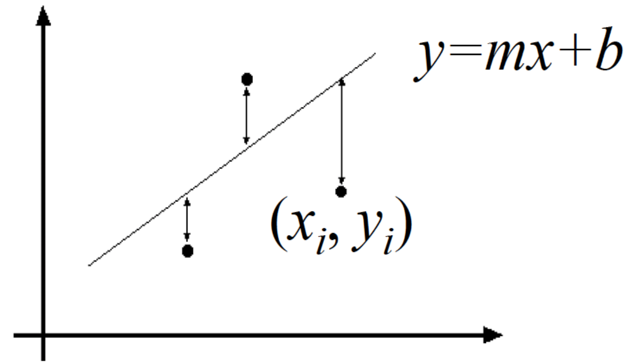


- clean data, no outliers : least squares fit ok

Least squares line fitting

- Data: $(x_1, y_1), \ldots, (x_n, y_n)$
- Line equation: $y_i = m\,x_i + b$
- Find (m, b) to minimize

$$\boxed{E = \sum_{i=1}^{n} (y_i - mx_i - b)^2}$$



$$E = \sum_{i=1}^{n} \left(\begin{bmatrix} x_i & 1 \end{bmatrix} \begin{bmatrix} m \\ b \end{bmatrix} - y_i \right)^2 = \left\| \begin{bmatrix} x_1 & 1 \\ \vdots & \vdots \\ x_n & 1 \end{bmatrix} \begin{bmatrix} m \\ b \end{bmatrix} - \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} \right\|^2 = \|\mathbf{Ap} - \mathbf{y}\|^2$$

$$= \mathbf{y}^T \mathbf{y} - 2(\mathbf{Ap})^T \mathbf{y} + (\mathbf{Ap})^T (\mathbf{Ap})$$

$$\frac{dE}{dp} = 2\mathbf{A}^T \mathbf{Ap} - 2\mathbf{A}^T \mathbf{y} = 0$$

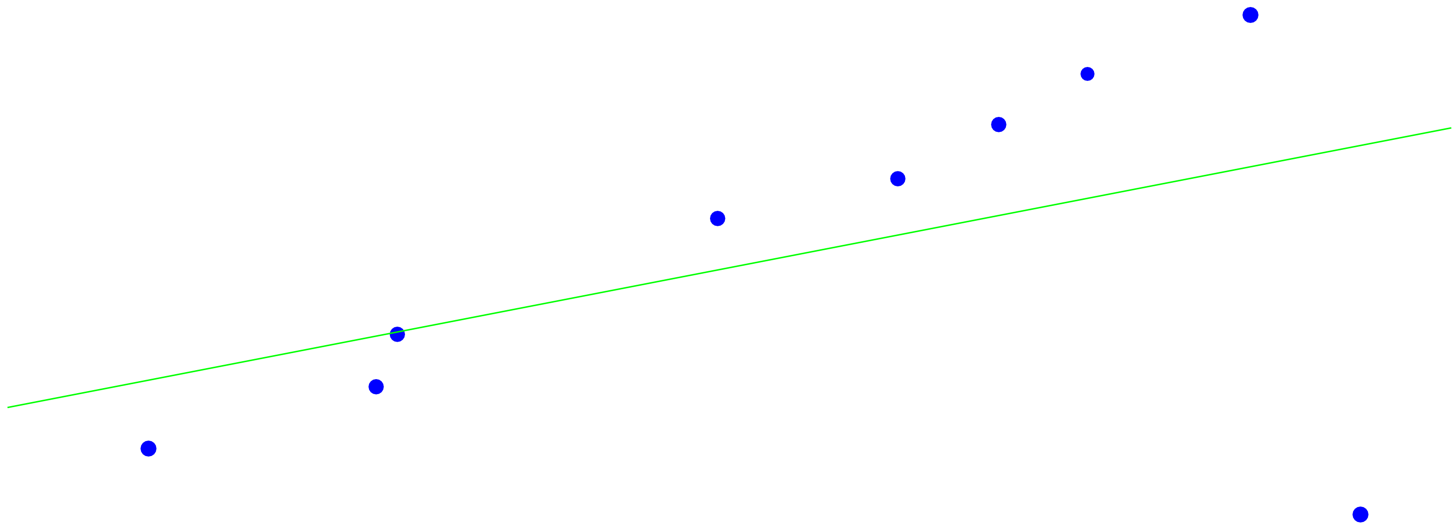
$$\mathbf{A}^T \mathbf{Ap} = \mathbf{A}^T \mathbf{y} \Rightarrow \mathbf{p} = \left(\mathbf{A}^T \mathbf{A}\right)^{-1} \mathbf{A}^T \mathbf{y}$$

Modified from S. Lazebnik

Least squares and gross outliers

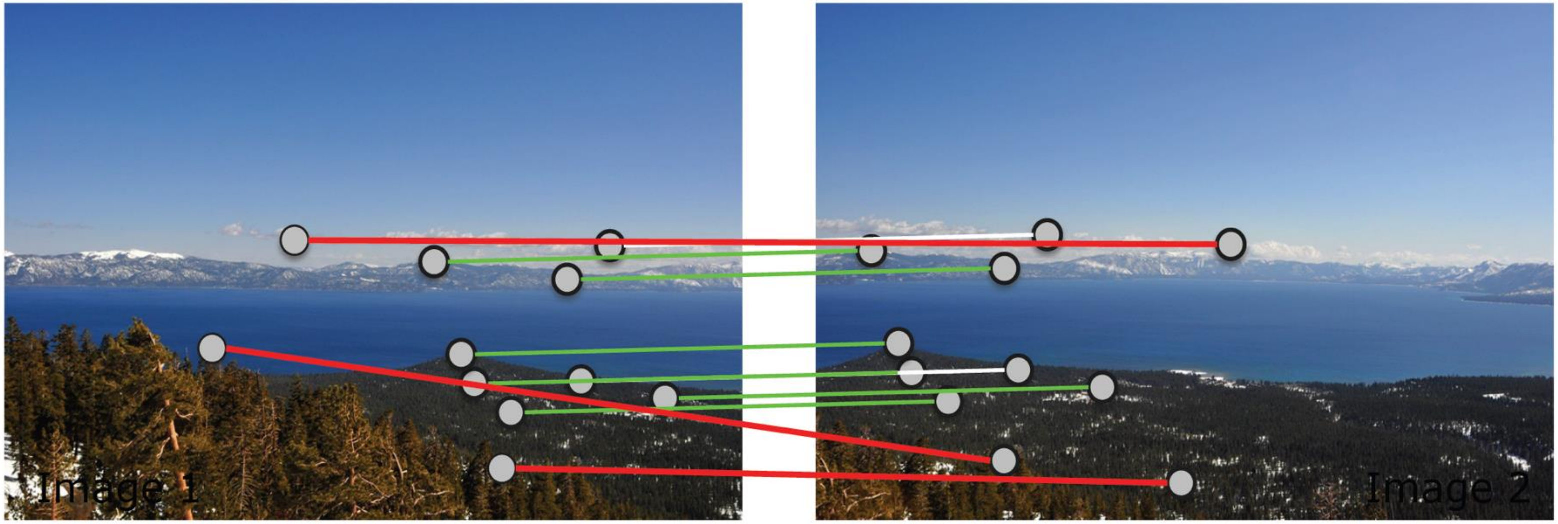
- one gross outlier : least squares fit fails

Least squares and gross outliers



- one gross outlier : least squares fit fails

RANSAC



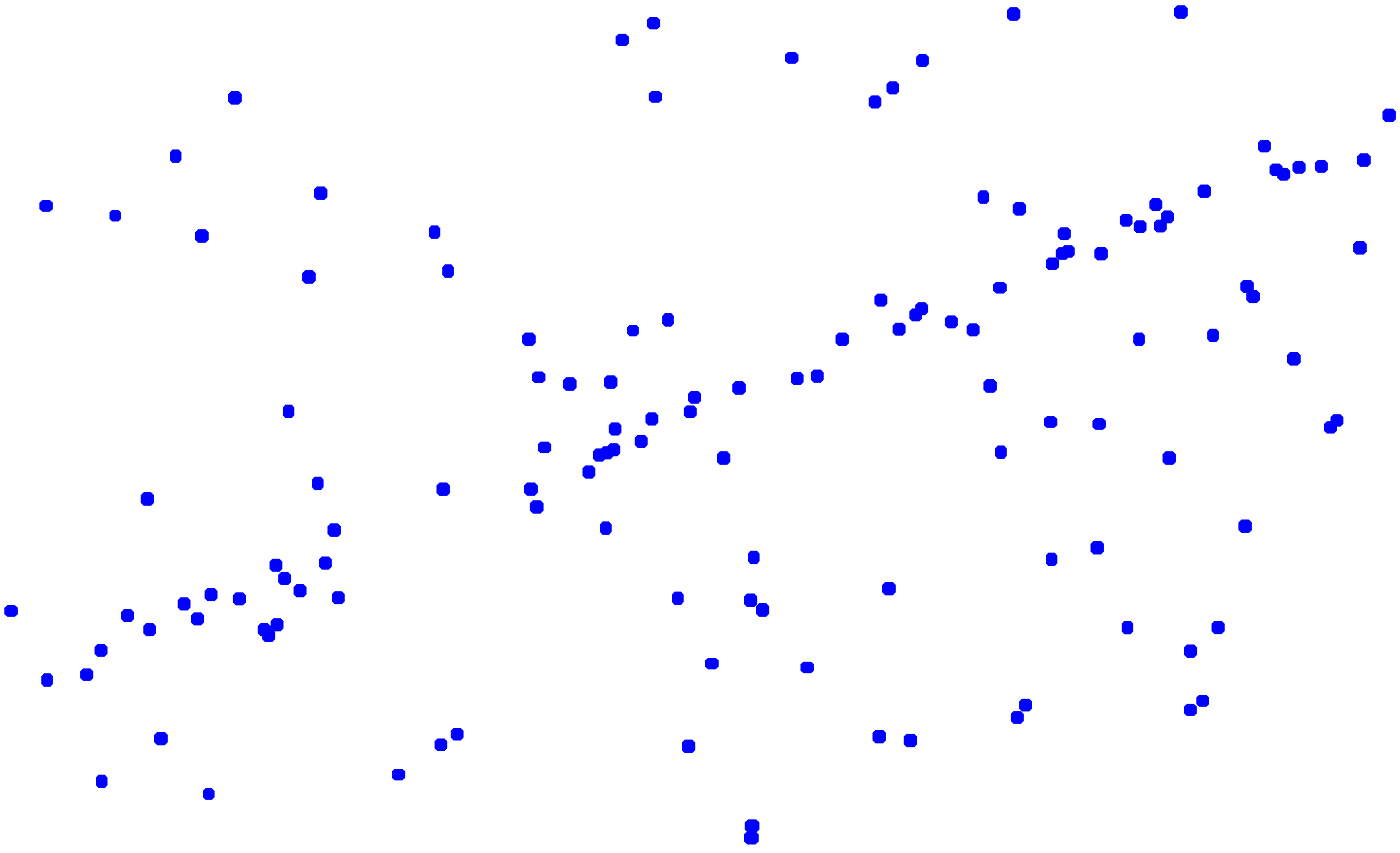
Outliers!

We need to robustly compute transformation in the presence of wrong matches.

If number of outliers < 50%, then RANSAC to the rescue!

Fischler and Bolles. CACM 1981. Random Sample Consensus: A Paradigm for Model Fitting With Applications to Image Analysis and Automated Cartography.

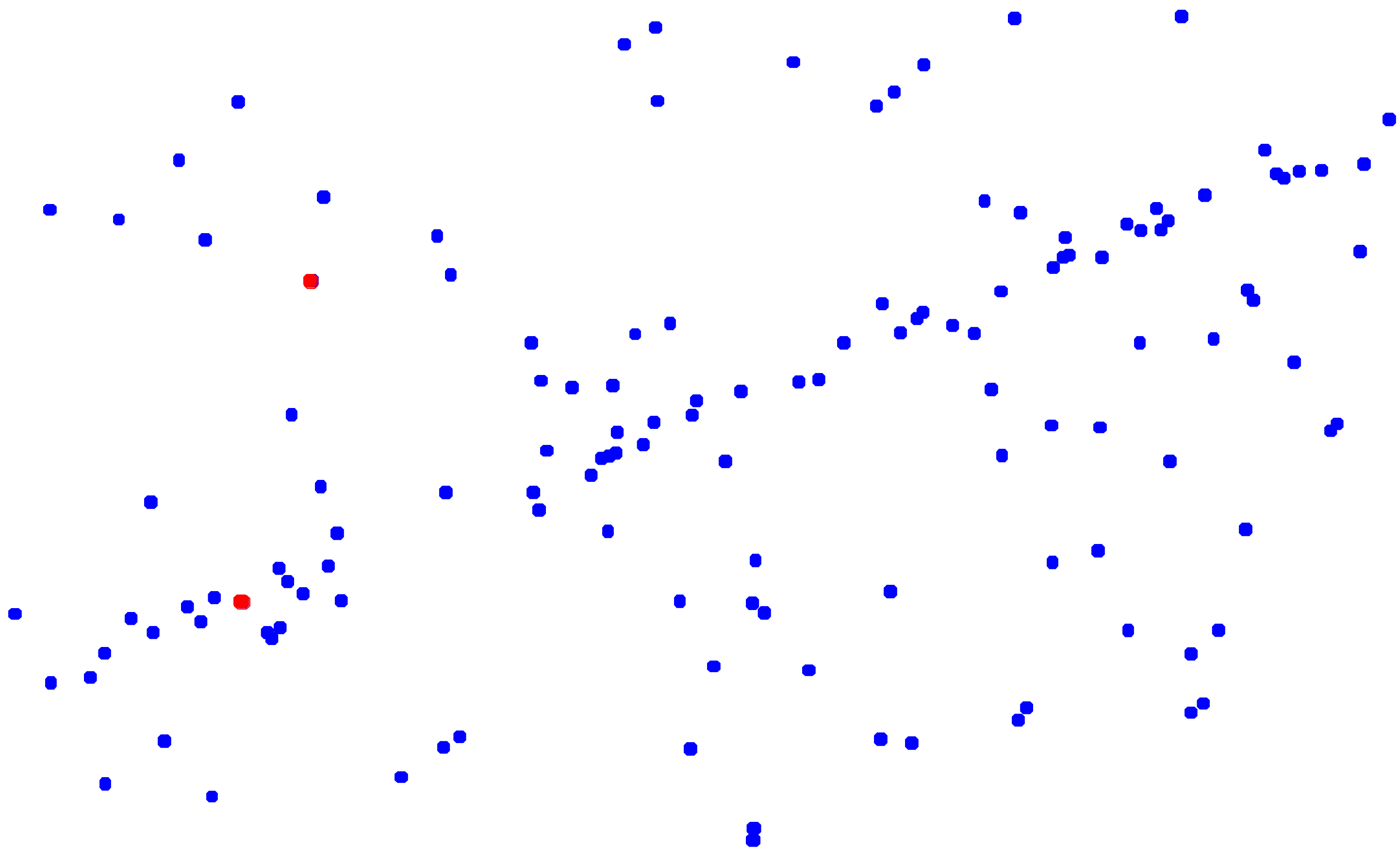
RANdom SAmple Consensus (RANSAC)



- data with outliers - pick two points at random - draw line through them - set margin on either side - count inlier points

Fischler and Bolles. CACM 1981. Random Sample Consensus: A Paradigm for Model Fitting With Applications to Image Analysis and Automated Cartography.

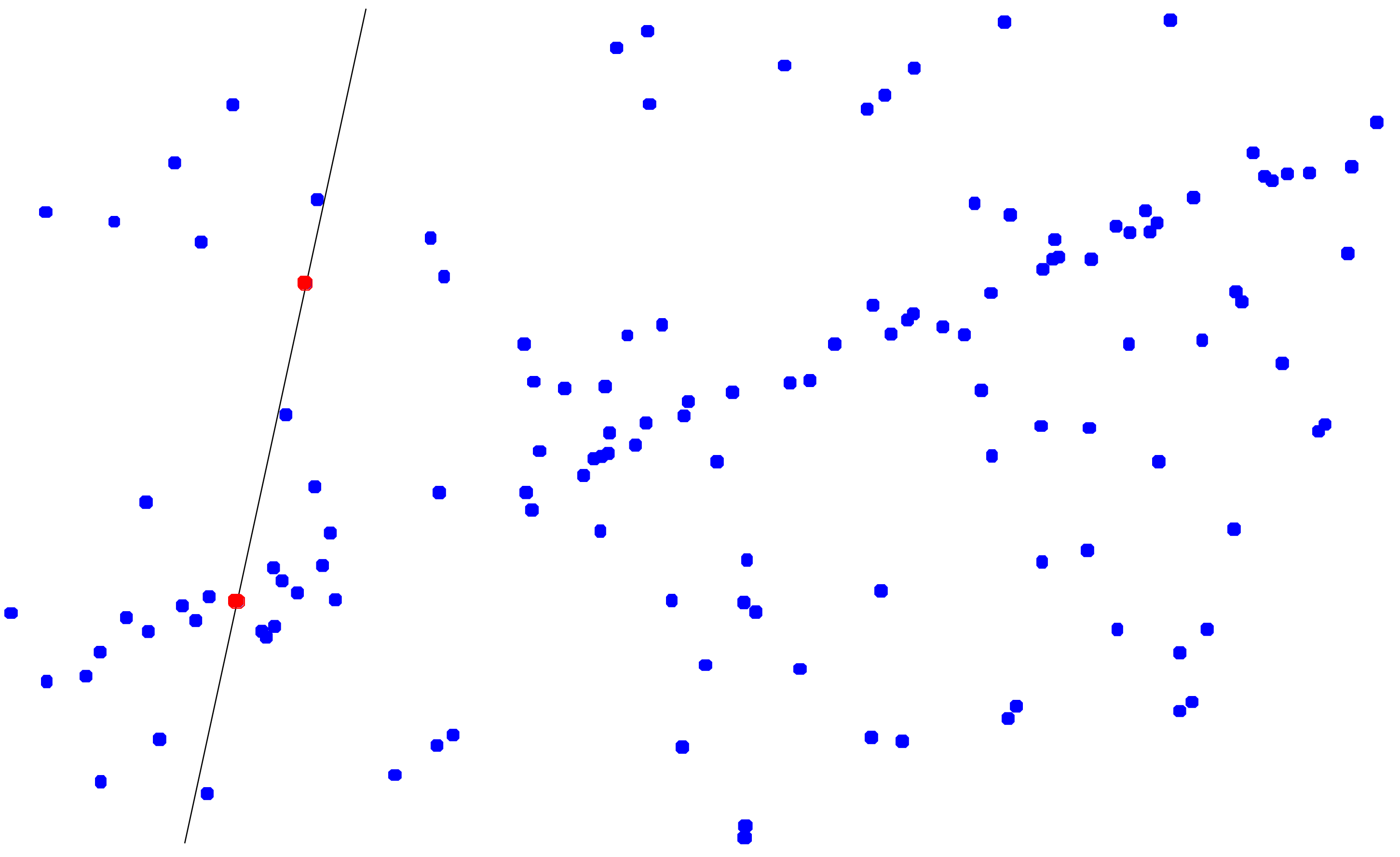
random sample consensus (RANSAC)



- data with outliers - pick two points at random - draw line through them - set margin on either side - count inlier points

Fischler and Bolles. CACM 1981. Random Sample Consensus: A Paradigm for Model Fitting With Applications to Image Analysis and Automated Cartography.

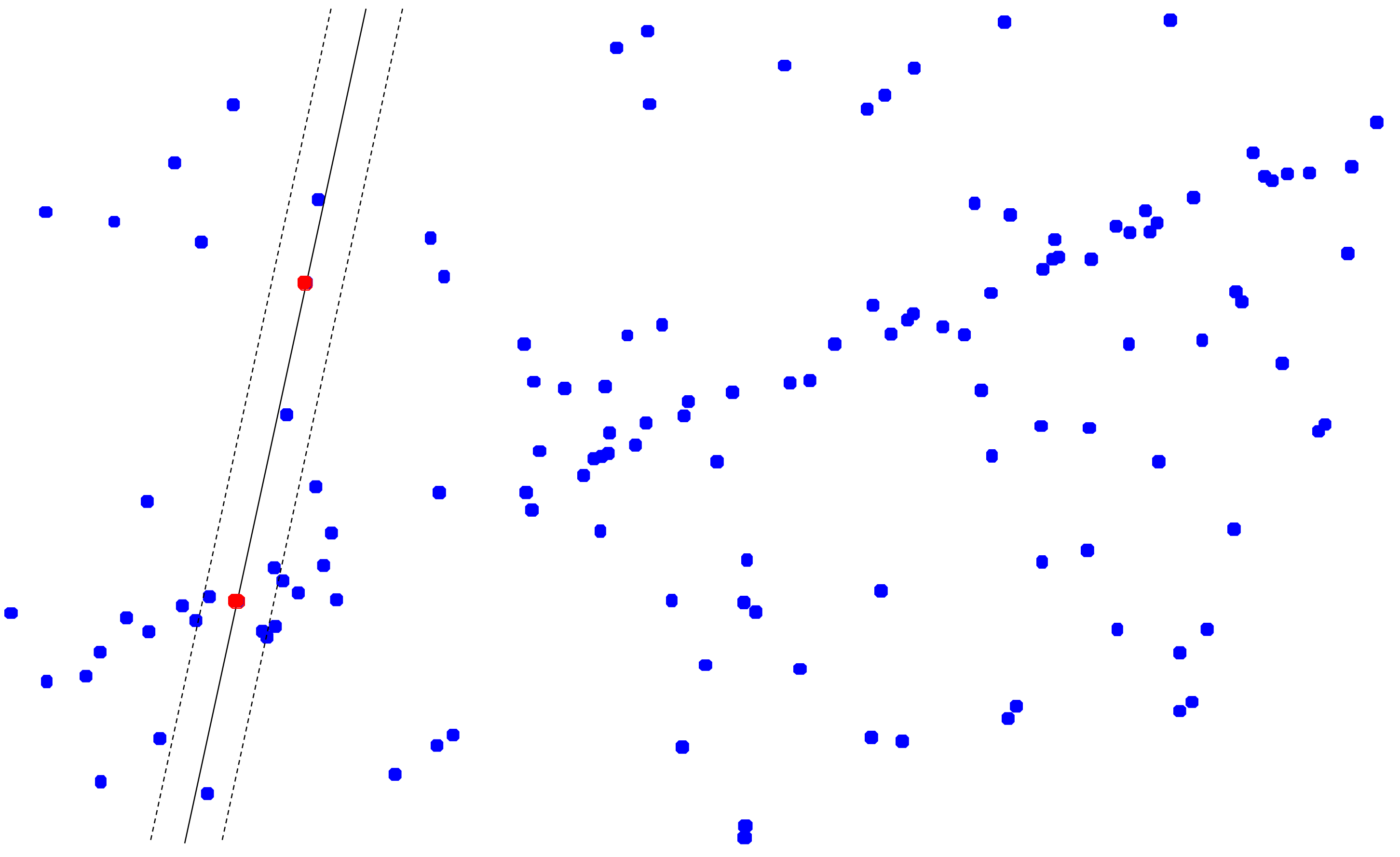
random sample consensus (RANSAC)



- data with outliers - pick two points at random - draw line through them - set margin on either side - count inlier points

Fischler and Bolles. CACM 1981. Random Sample Consensus: A Paradigm for Model Fitting With Applications to Image Analysis and Automated Cartography.

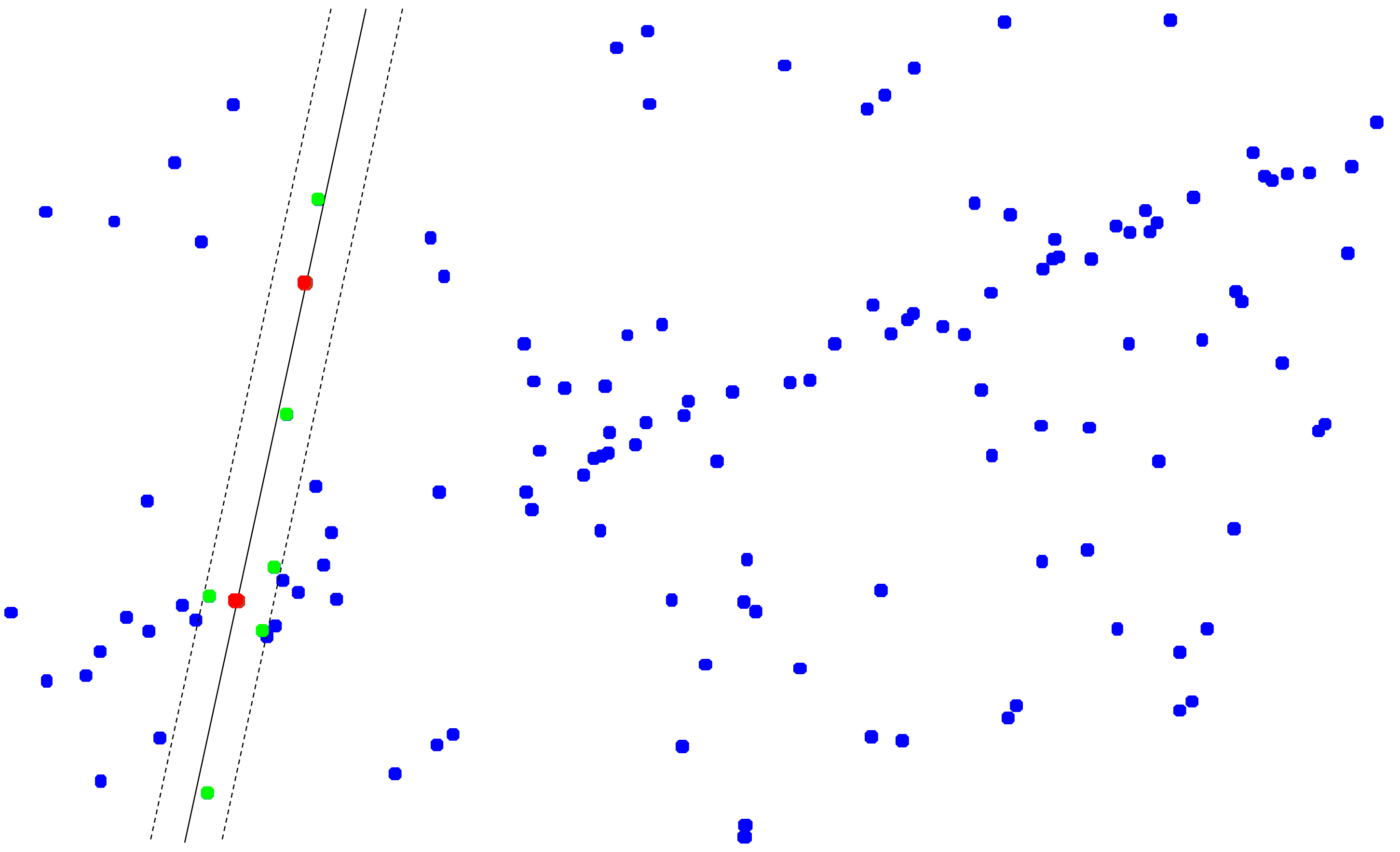
random sample consensus (RANSAC)



- data with outliers - pick two points at random - draw line through them - set margin on either side - count inlier points

Fischler and Bolles. CACM 1981. Random Sample Consensus: A Paradigm for Model Fitting With Applications to Image Analysis and Automated Cartography.

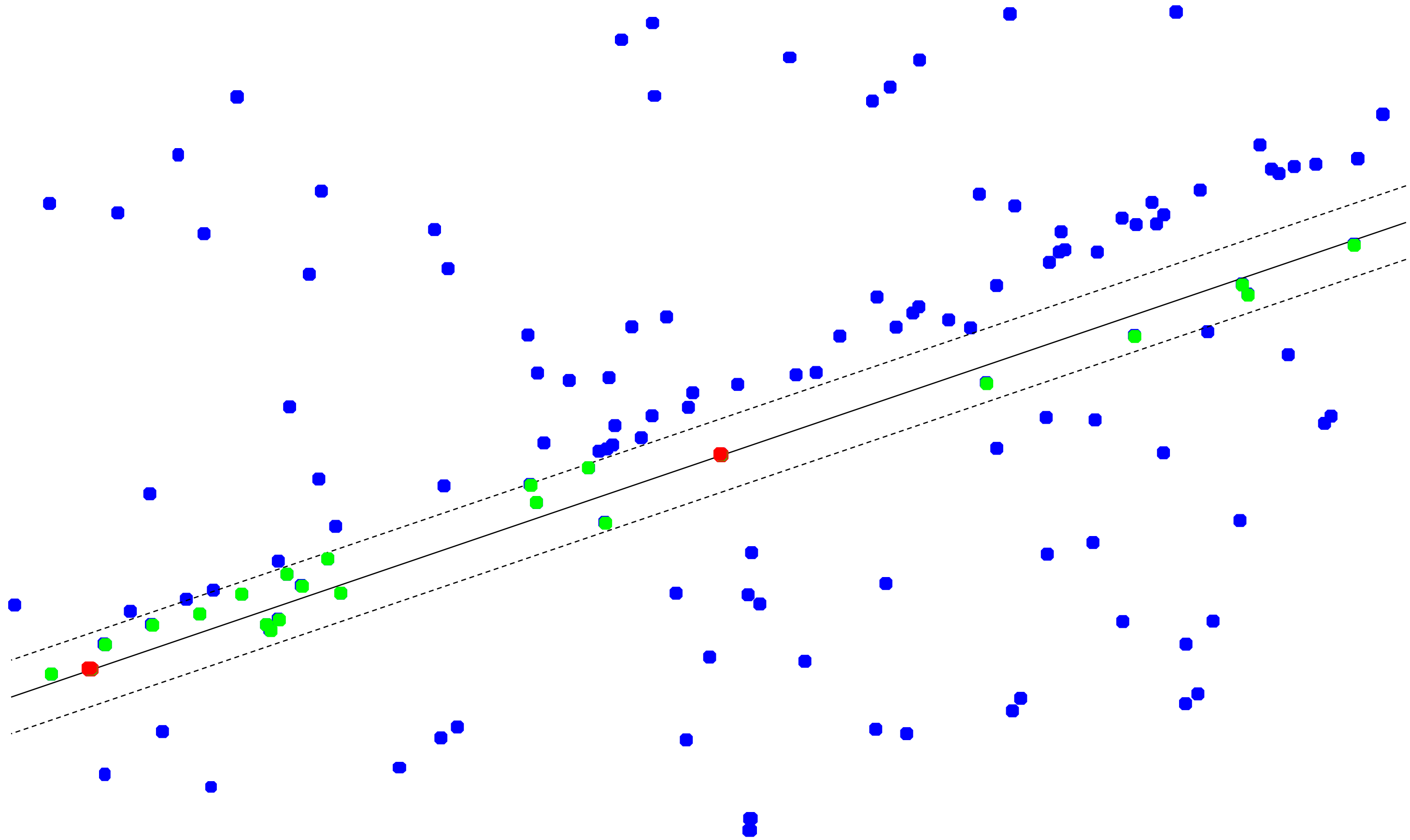
random sample consensus (RANSAC)



- data with outliers - pick two points at random - draw line through them - set margin on either side - count inlier points

Fischler and Bolles. CACM 1981. Random Sample Consensus: A Paradigm for Model Fitting With Applications to Image Analysis and Automated Cartography.

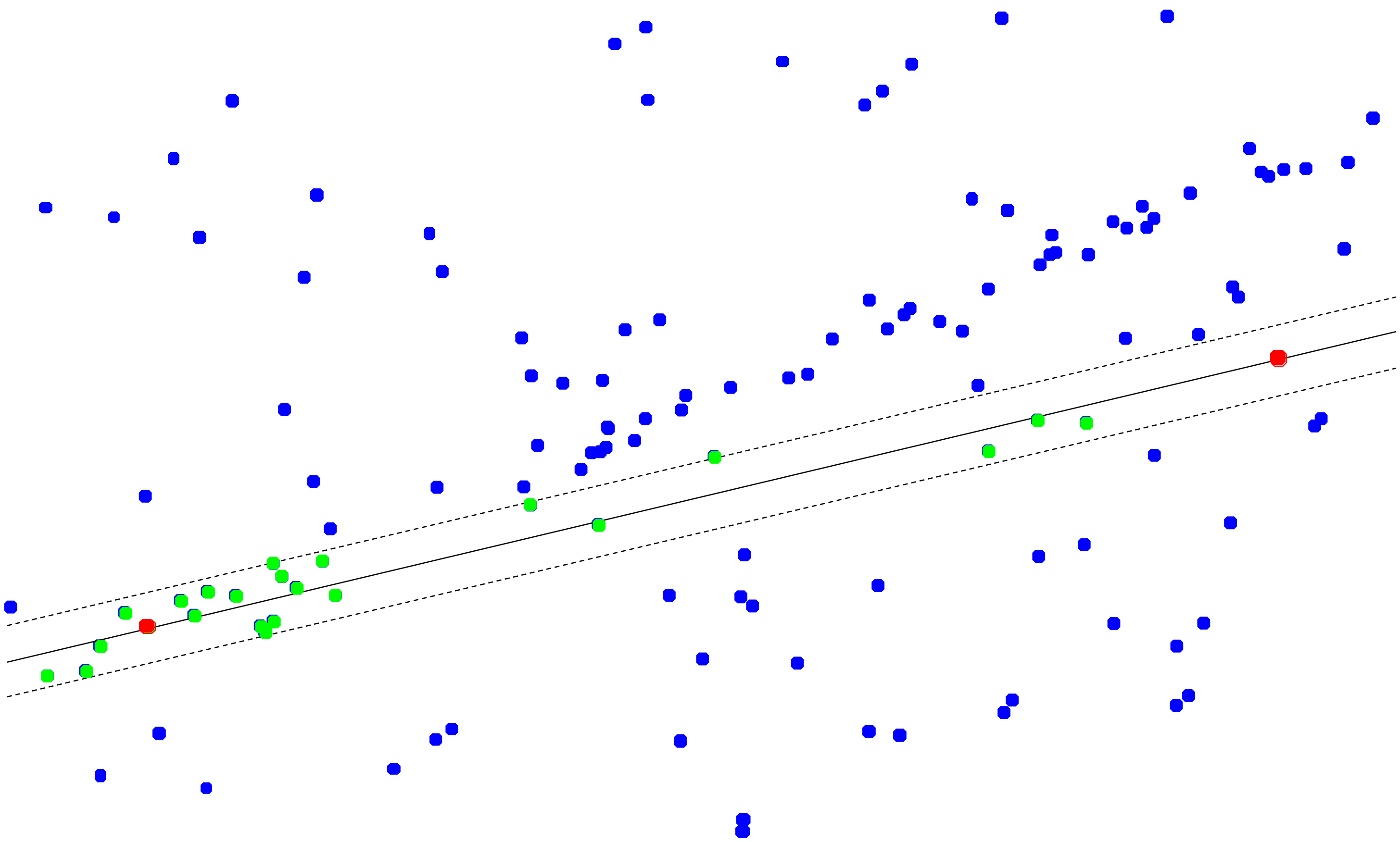
random sample consensus (RANSAC)



- repeat: pick two points at random, draw line through them, count inlier points at fixed distance to line, keep best hypothesis so far

Fischler and Bolles. CACM 1981. Random Sample Consensus: A Paradigm for Model Fitting With Applications to Image Analysis and Automated Cartography.

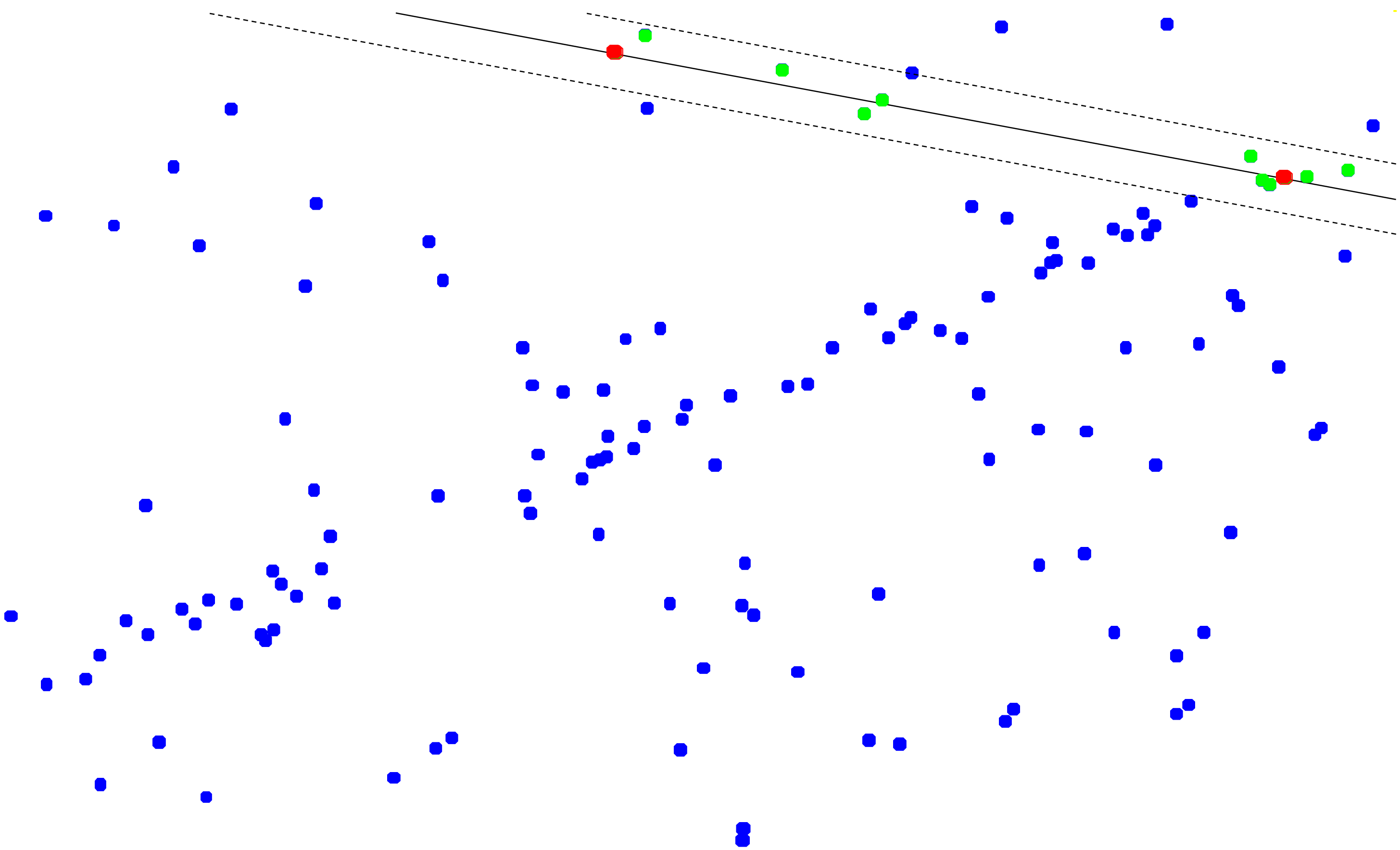
random sample consensus (RANSAC)



- repeat: pick two points at random, draw line through them, count inlier points at fixed distance to line, keep best hypothesis so far

Fischler and Bolles. CACM 1981. Random Sample Consensus: A Paradigm for Model Fitting With Applications to Image Analysis and Automated Cartography.

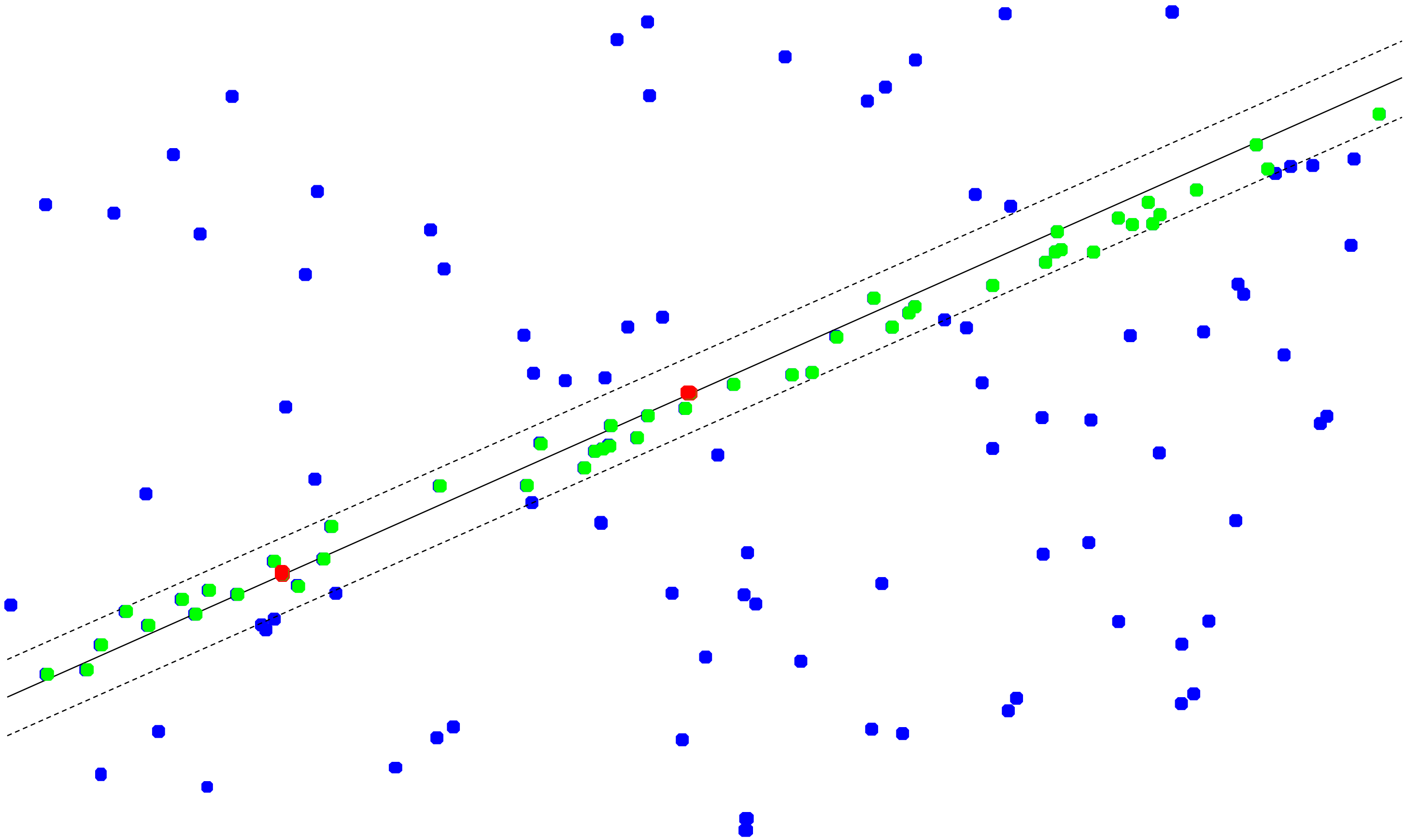
random sample consensus (RANSAC)



- repeat: pick two points at random, draw line through them, count inlier points at fixed distance to line, keep best hypothesis so far

Fischler and Bolles. CACM 1981. Random Sample Consensus: A Paradigm for Model Fitting With Applications to Image Analysis and Automated Cartography.

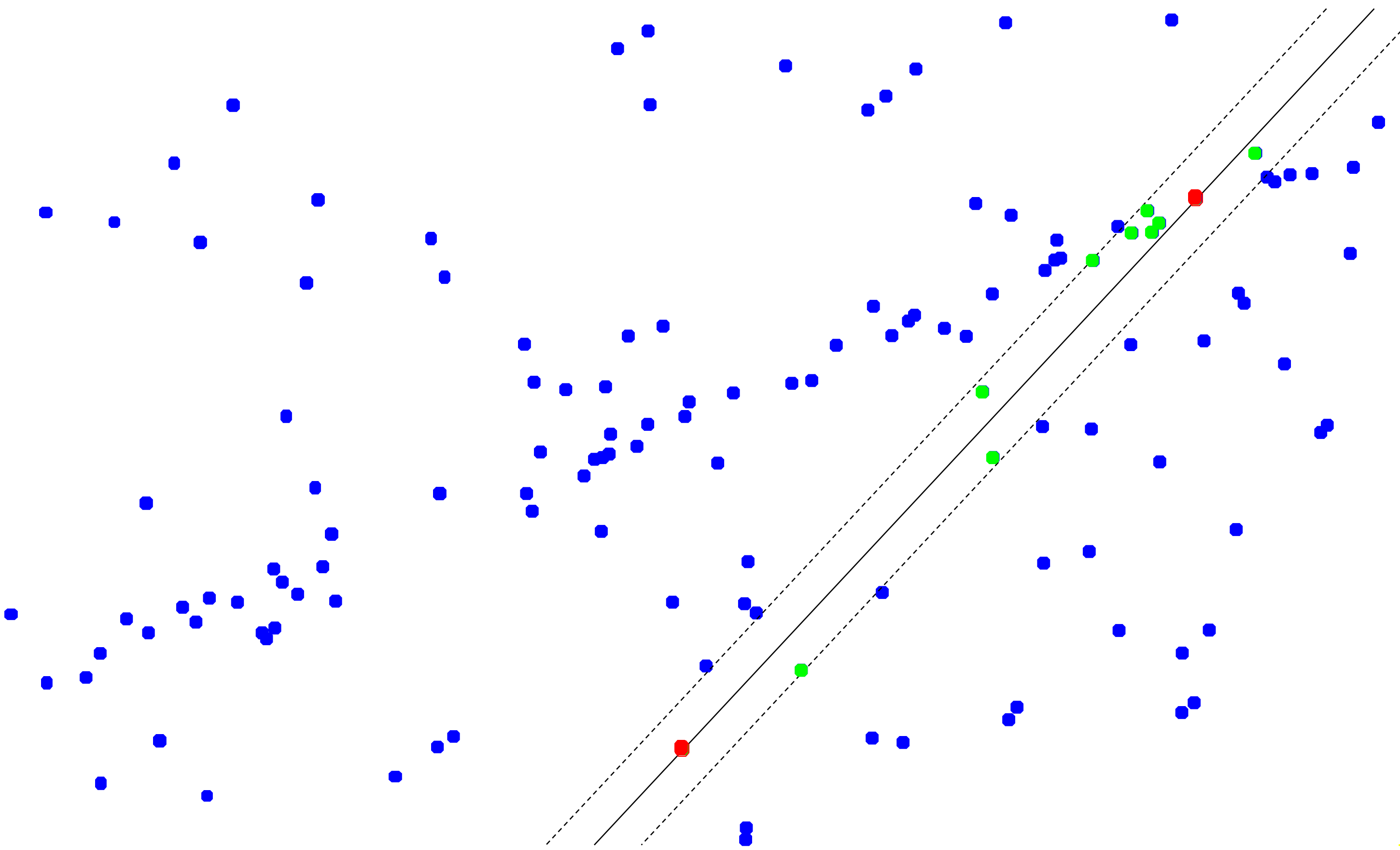
random sample consensus (RANSAC)



- repeat: pick two points at random, draw line through them, count inlier points at fixed distance to line, keep best hypothesis so far

Fischler and Bolles. CACM 1981. Random Sample Consensus: A Paradigm for Model Fitting With Applications to Image Analysis and Automated Cartography.

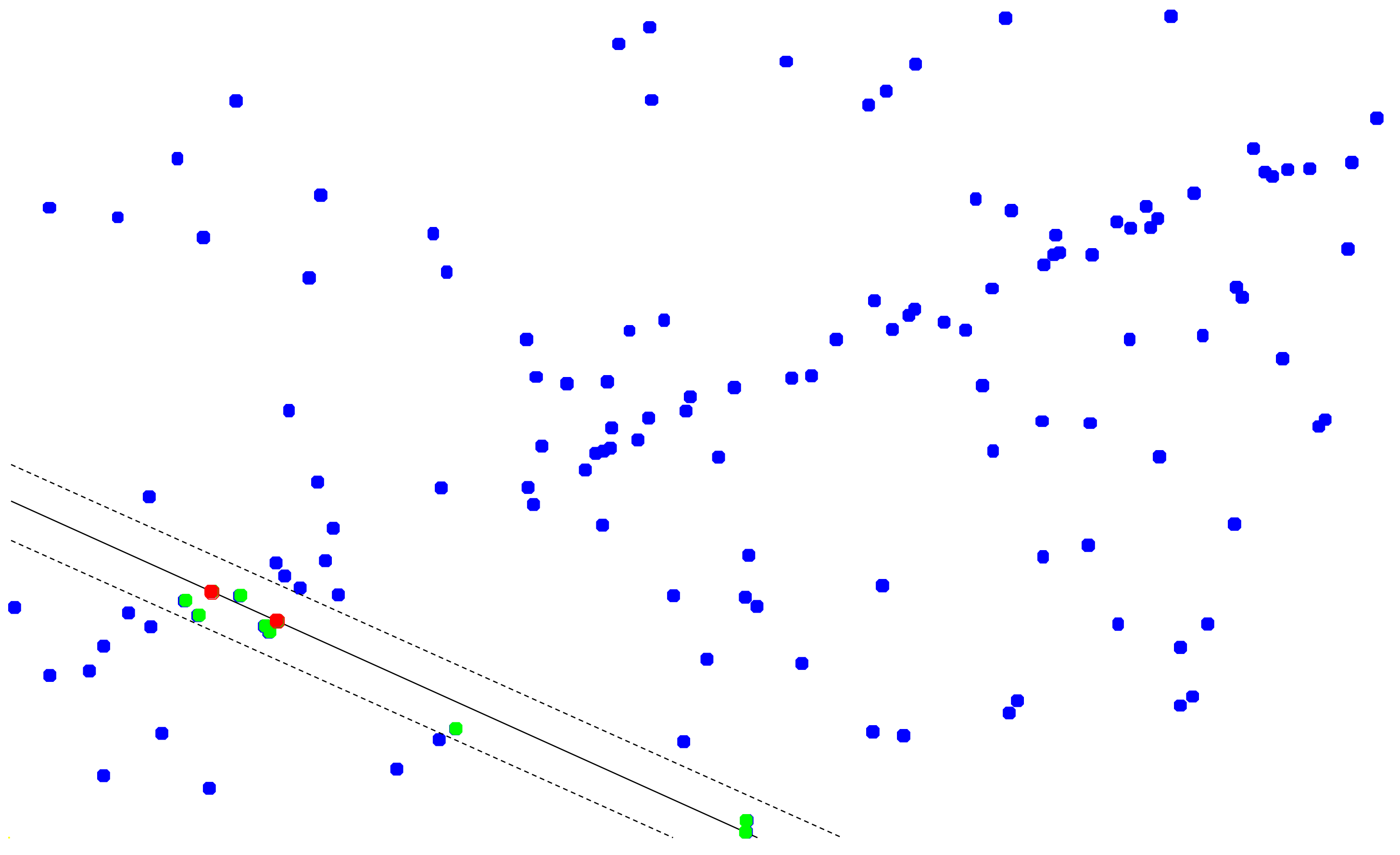
random sample consensus (RANSAC)



- repeat: pick two points at random, draw line through them, count inlier points at fixed distance to line, keep best hypothesis so far

Fischler and Bolles. CACM 1981. Random Sample Consensus: A Paradigm for Model Fitting With Applications to Image Analysis and Automated Cartography.

random sample consensus (RANSAC)



- repeat: pick two points at random, draw line through them, count inlier points at fixed distance to line, keep best hypothesis so far

Fischler and Bolles. CACM 1981. Random Sample Consensus: A Paradigm for Model Fitting With Applications to Image Analysis and Automated Cartography.

random sample consensus (RANSAC)

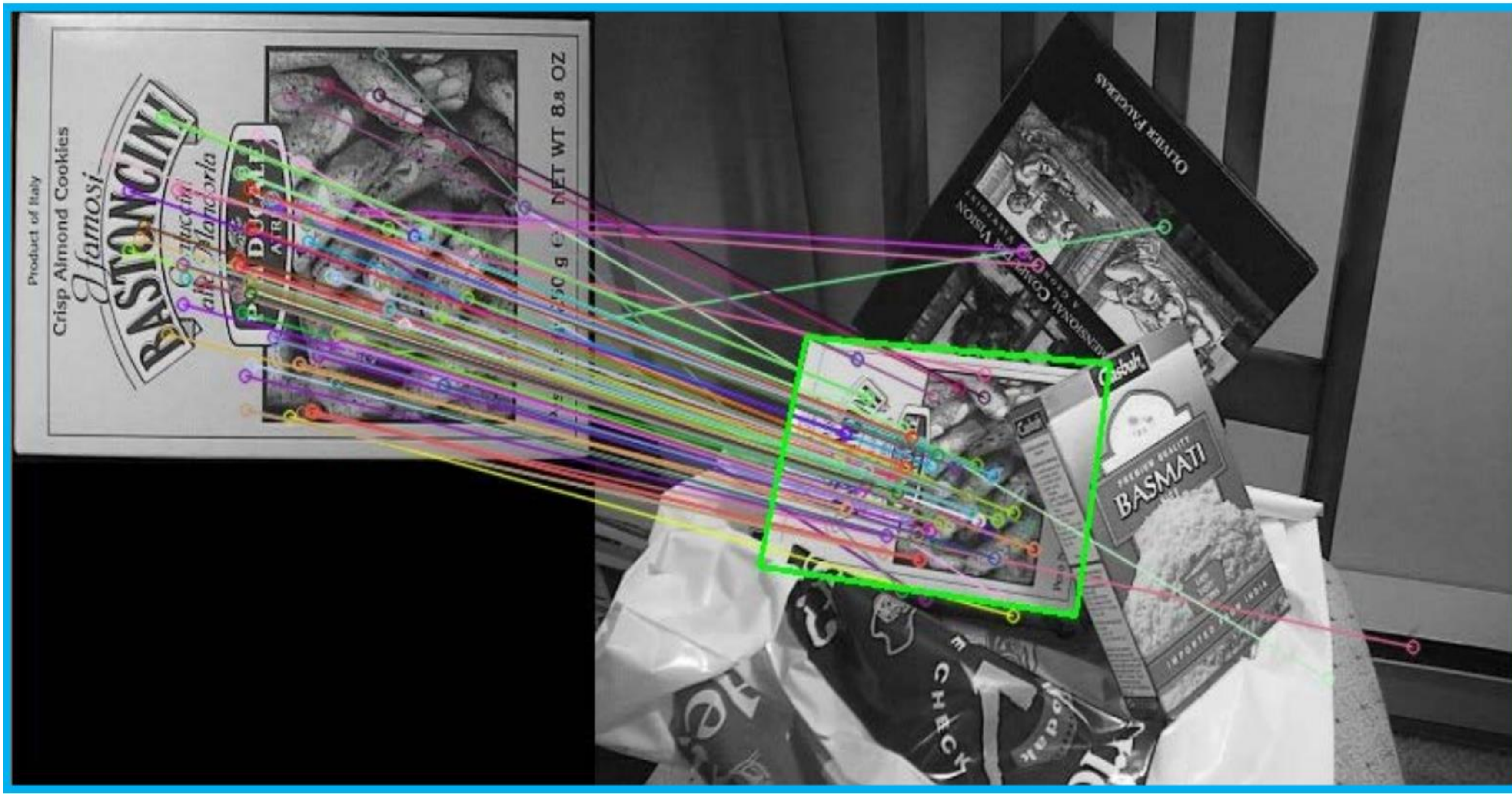
[Fischler and Bolles 1981]

- X: data (tentative correspondences)
- n: minimum number of samples to fit a model
- $s(x;\theta)$: score of sample x given model parameters θ
- repeat
  - hypothesis
    - draw n samples $H \subset X$ at random
    - fit model to H, compute parameters θ
  - verification
    - are data consistent with hypothesis? compute score $S = \sum_{x \in X} s(x;\theta)$
    - if $S^* > S$, store solution $\theta^* := \theta$, $S^* := S$

Fischler and Bolles. CACM 1981. Random Sample Consensus: A Paradigm for Model Fitting With Applications to Image Analysis and Automated Cartography.

RANSAC Applications

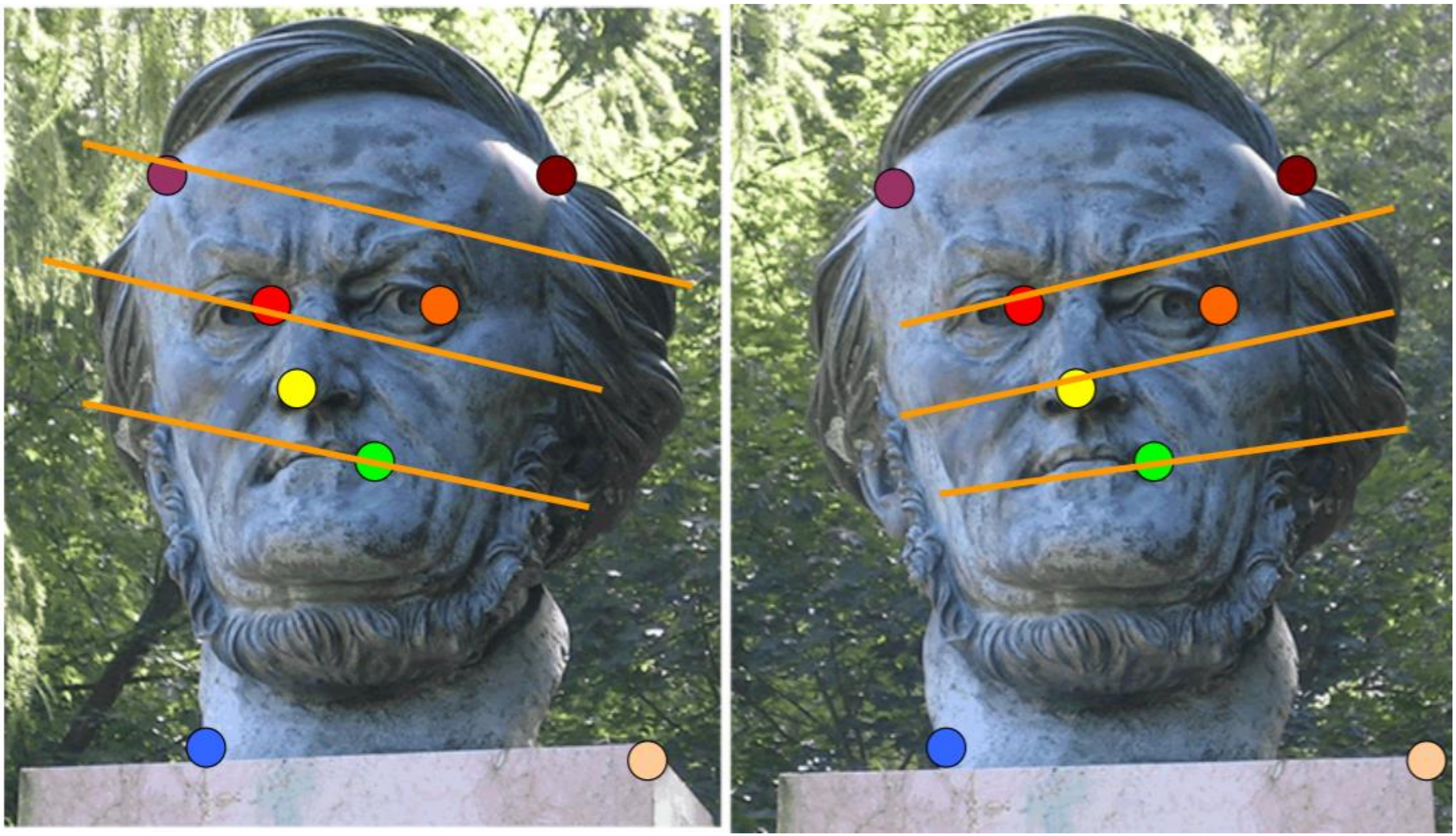
Rotation



Credit: Aaron Bobick, Washington University in St. Louis

RANSAC Applications

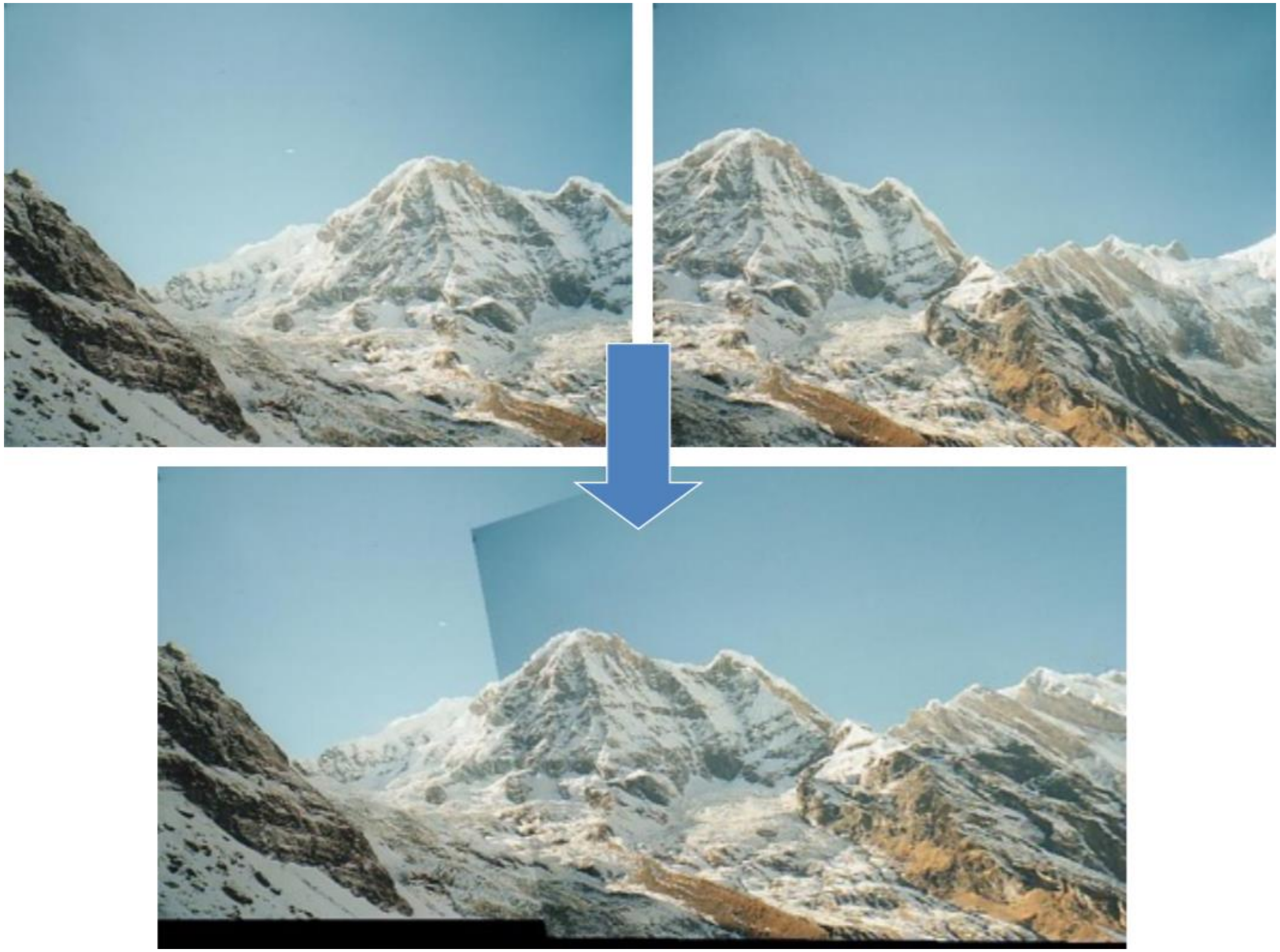
Estimating transformation matrix (fundamental matrix) relating two views



Credit: Derek Hoeim, UIUC

RANSAC Applications

Computing a homography (e.g., image stitching)



Credit: Ali Farhadi, Univ of Washington

RANSAC Conclusion

Pros:

- Robust to outliers
- Optimization parameters are easier to choose

Limitations :

- inlier ratio w (number of inliers in data / number of points in data) unknown
- too expensive when minimum number of samples is large (e.g. $n > 6$) and inlier ratio is small (e.g. $w < 10\%$)
- Computational time grows quickly with fraction of outliers and number of parameters
- Not as good for getting multiple fits (though one solution is to remove inliers after each fit and repeat)

Fischler and Bolles. CACM 1981. Random Sample Consensus: A Paradigm for Model Fitting With Applications to Image Analysis and Automated Cartography.

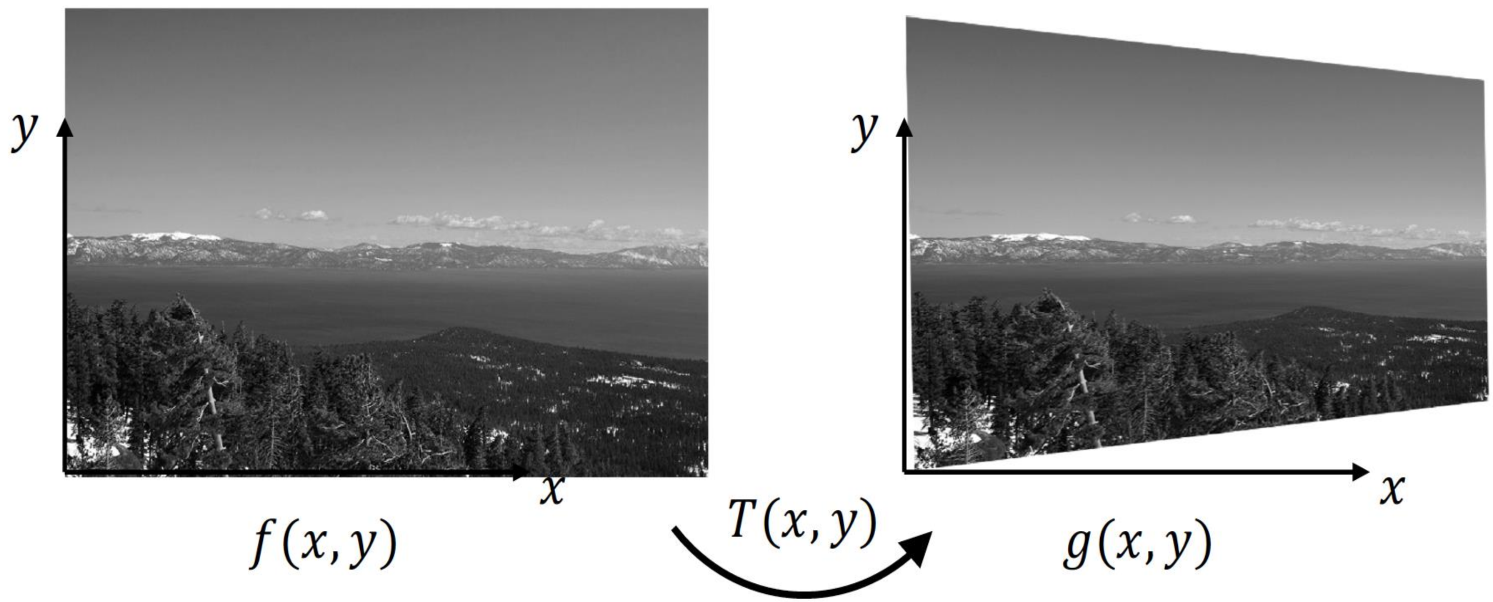
Image Alignment and Blending

Fischler and Bolles. CACM 1981. Random Sample Consensus: A Paradigm for Model Fitting With Applications to Image Analysis and Automated Cartography.

Warping

Given a transformation T and a image $f(x,y)$, compute the transformed image $g(x,y)$

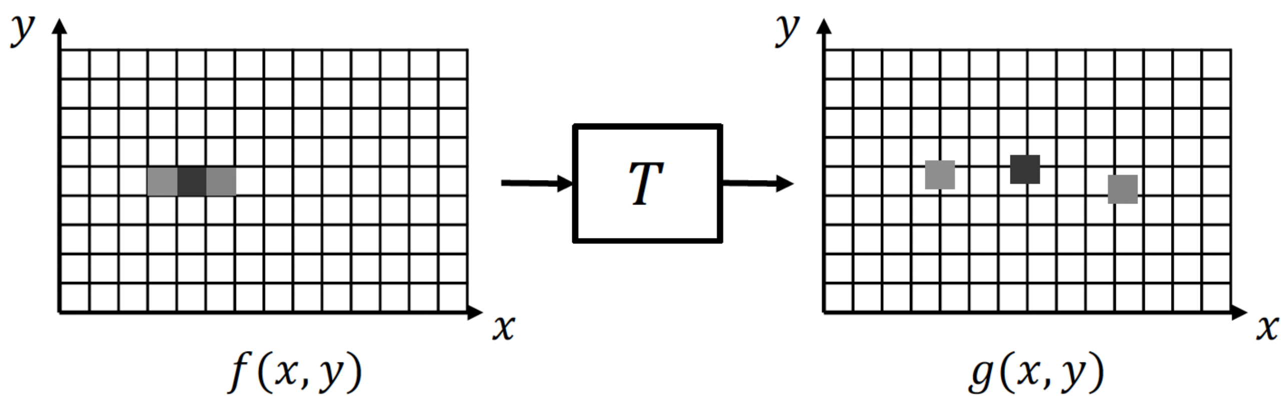
$$\boxed{g(x,y) = f\big(T(x,y)\big)}$$



Forward Warping

Send each pixel (x, y) in $f(x, y)$ to its corresponding location $T(x, y)$ in $g(x, y)$

$$\boxed{g(x, y) = f\big(T(x, y)\big)}$$



What if pixel lands in between pixels?

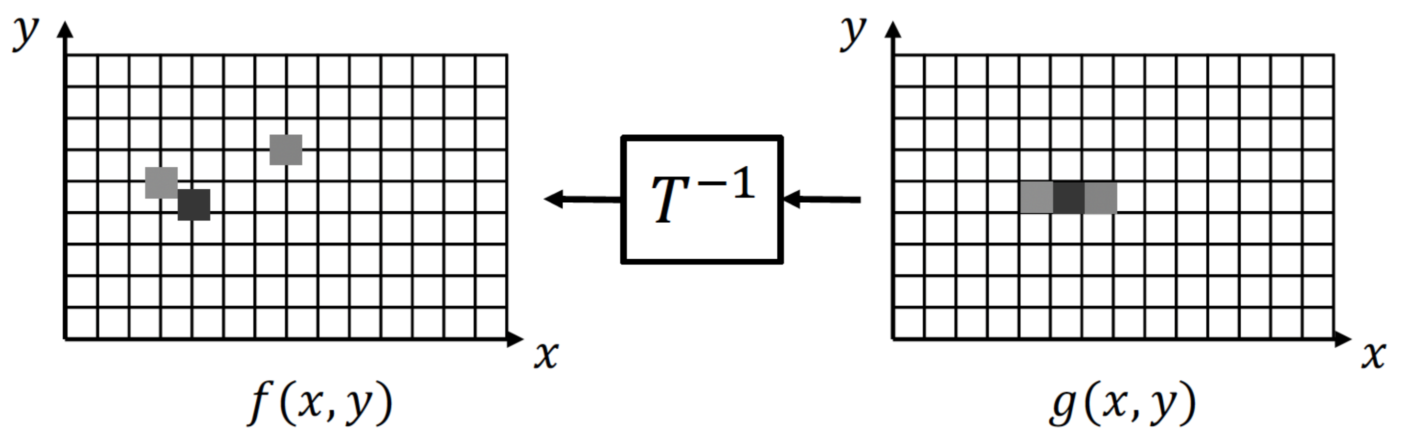
What if not all pixels in $g(x, y)$ are filled?

Can result in holes!

Backward Warping

Get each pixel (x, y) in $g(x, y)$ from its corresponding location $T^{-1}(x, y)$ in $f(x, y)$

$$\boxed{g(x, y) = f\big(T(x, y)\big)}$$



What if pixel lands between pixels?
Use Nearest Neighbor or Interpolate

Image Alignment Process

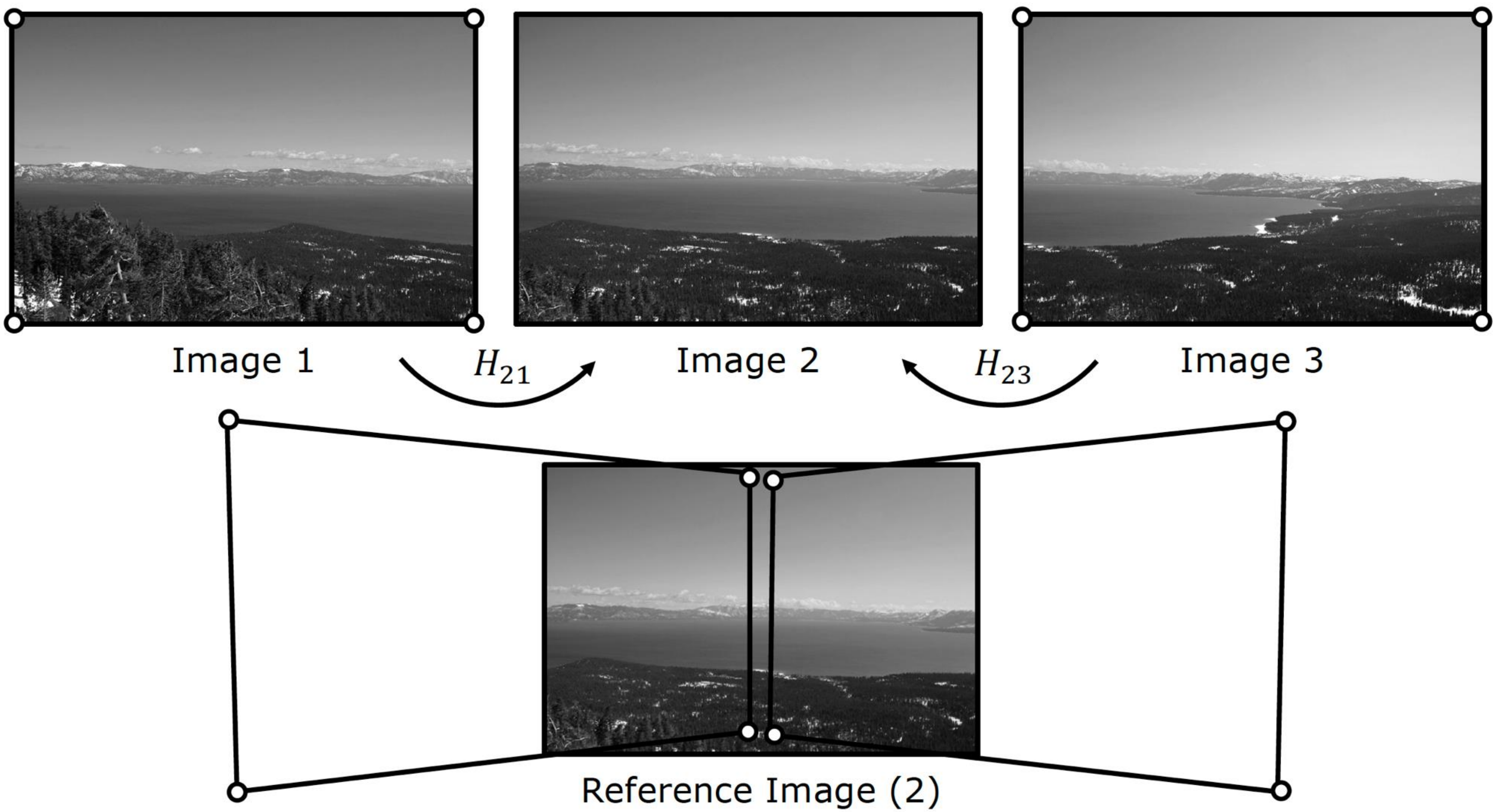
Image 1

Image 2

Image 3

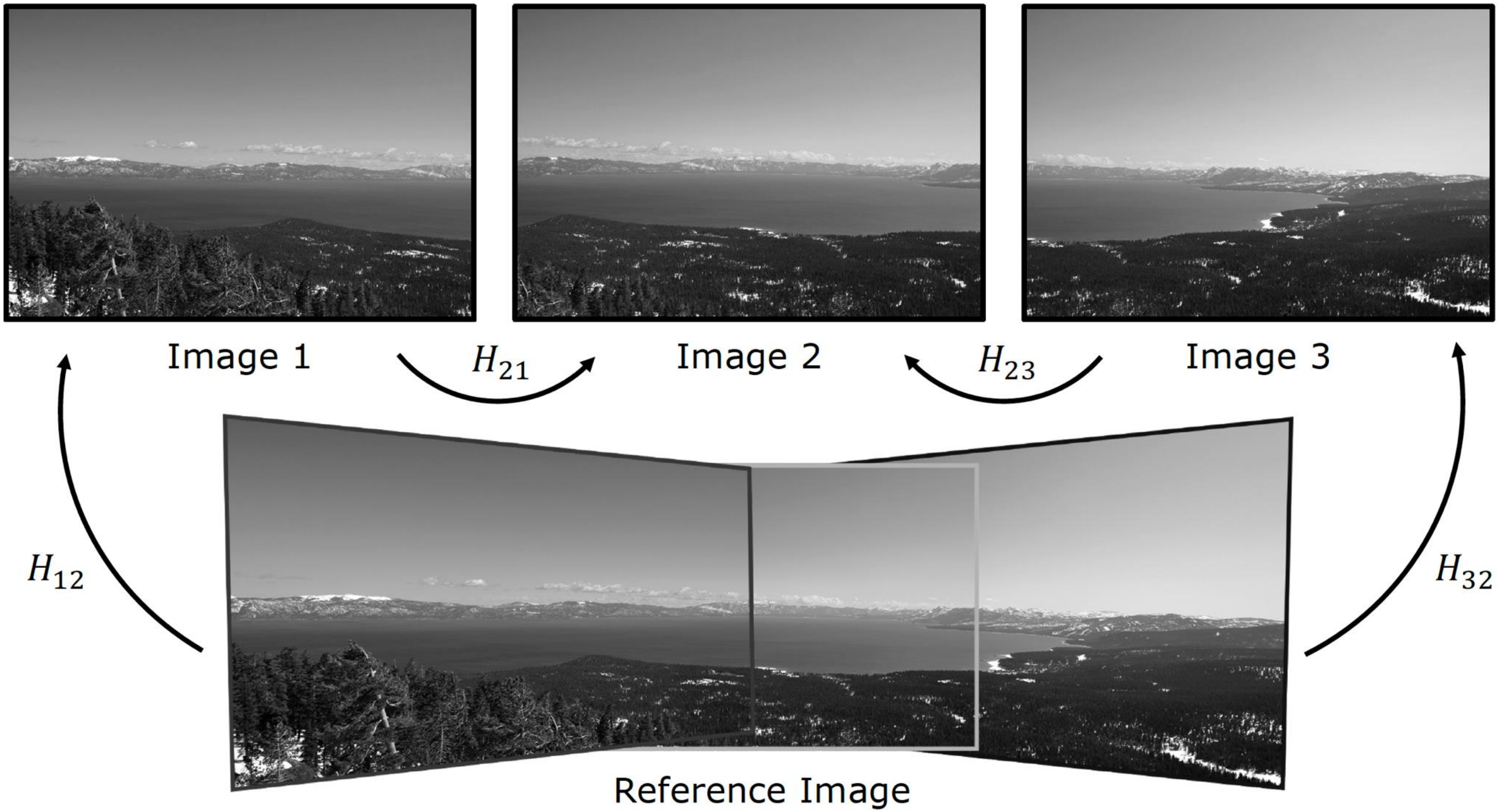
Reference Image
(Image 2)

Image Alignment Process



Compute the bounds of Image 1 and Image 3 in reference image space

Image Alignment Process



Fill each pixel within bounds, by computing its location in captured image

Fischler and Bolles. CACM 1981. Random Sample Consensus: A Paradigm for Model Fitting With Applications to Image Analysis and Automated Cartography.

Blending Images

Overlaid Aligned Images

Hard seams due to vignetting, exposure differences, etc.

Blending Images: Averaging

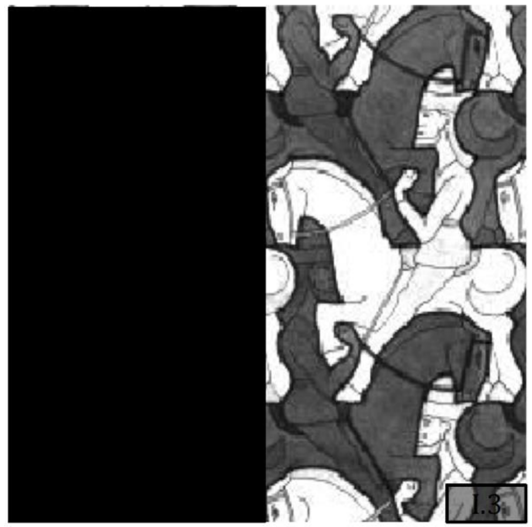
Averaged Images

Seams still visible

Blending Images

Say we want to blend images I_1 and I_2 at the center

+



=

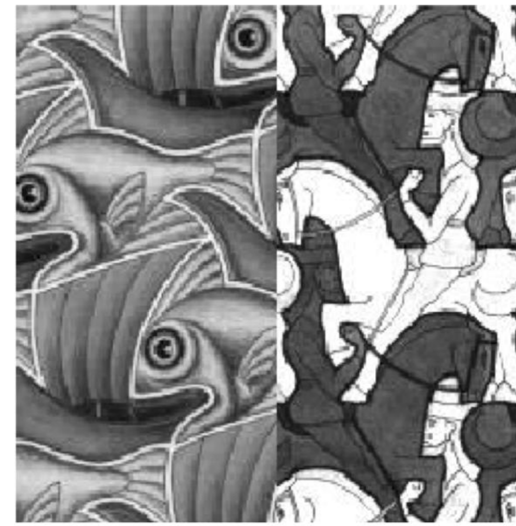
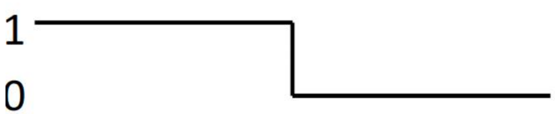


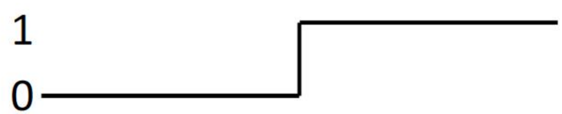
Image I_1

Image I_2

Hard overlay



Weight w_1



Weight w_2

Blending Images

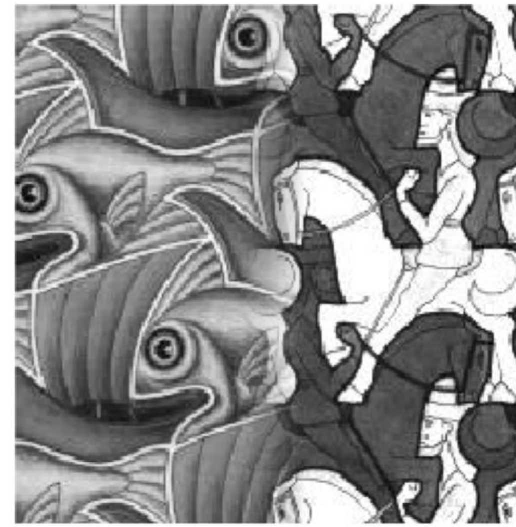
Say we want to blend images I_1 and I_2 at the center

Image I_1

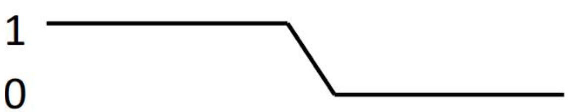
+

Image I_2

=



Blended Image I_{blend}



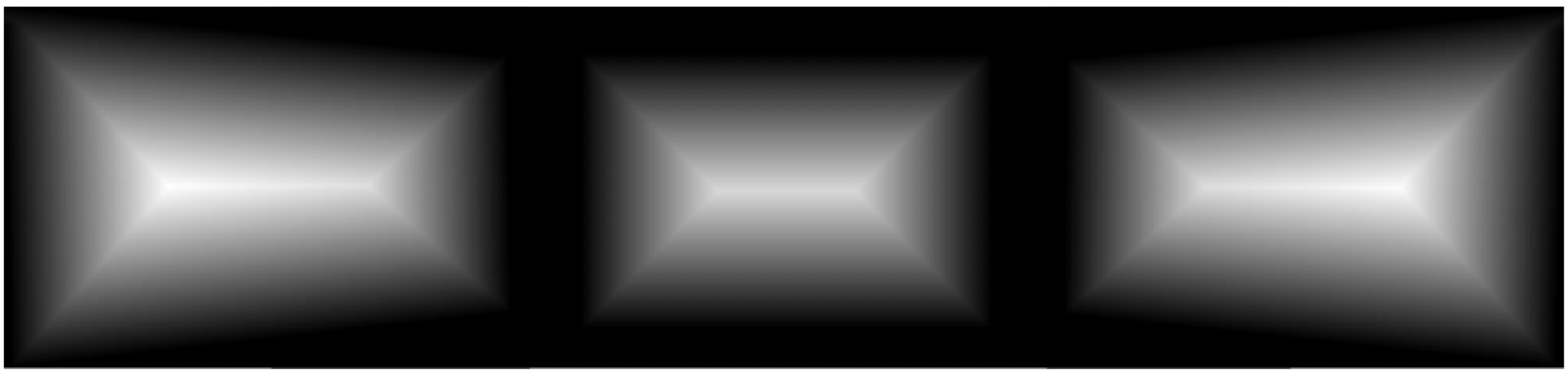
Weight w_1

Weight w_2

$$I_{blend} = \frac{w_1 I_1 + w_2 I_2}{w_1 + w_2}$$

Blending Images: With Weighting functions

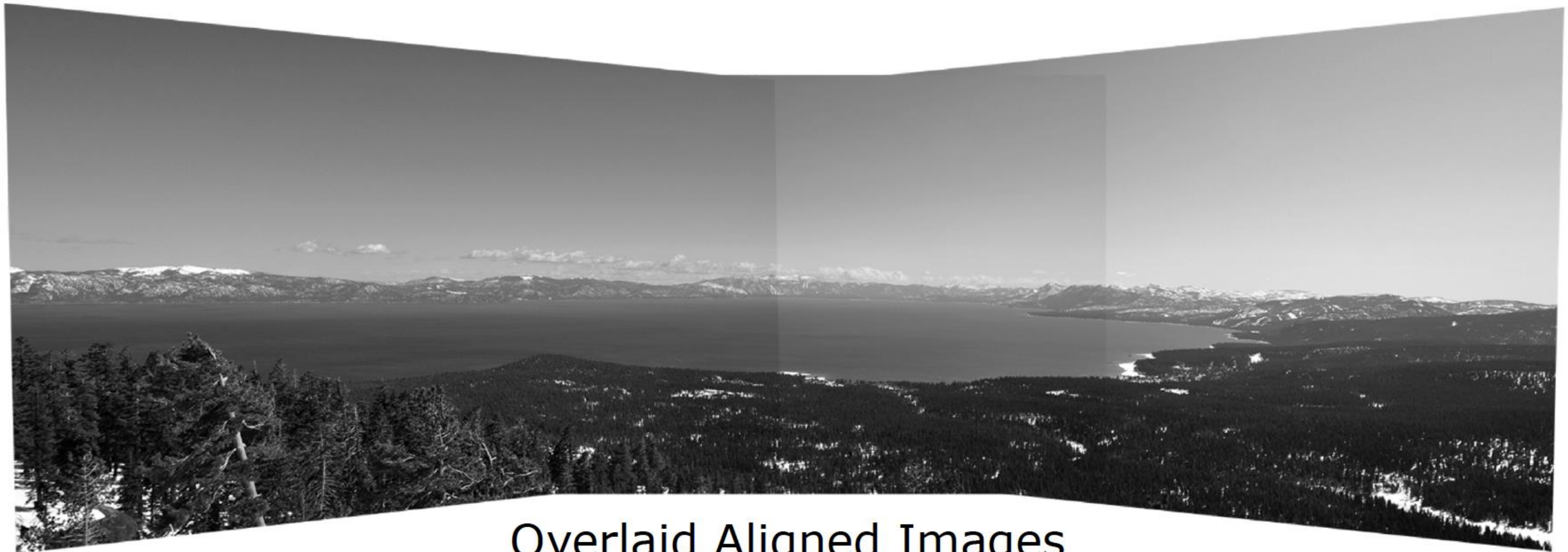
Image 1 Image 2 Image 3



Weight w_1 Weight w_2 Weight w_3

Pixels closer to the edge get a lower weight.

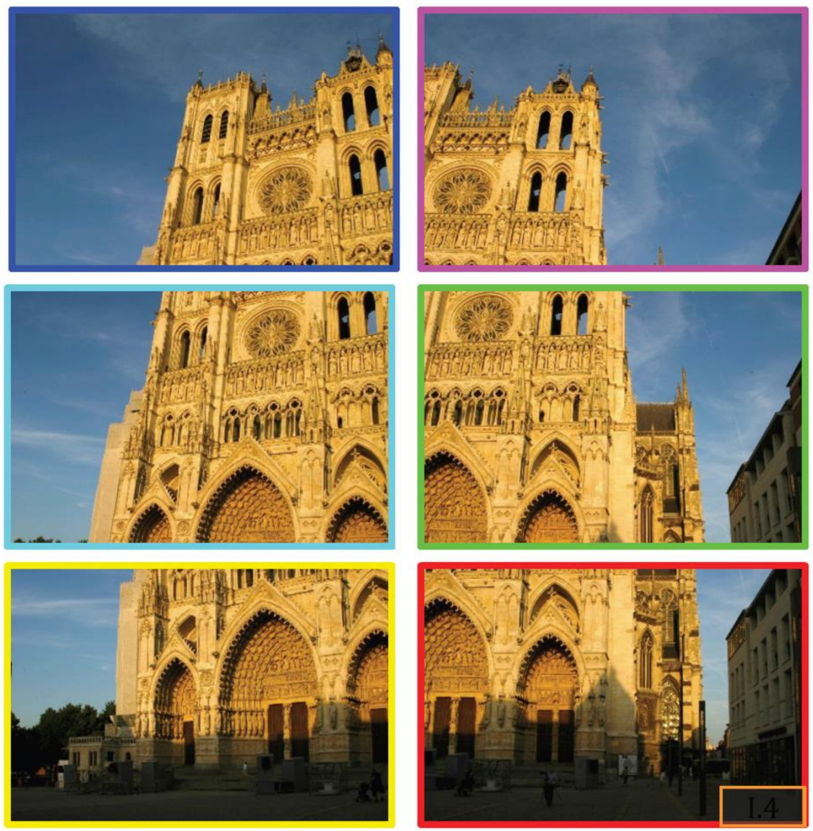
Weighted Blending



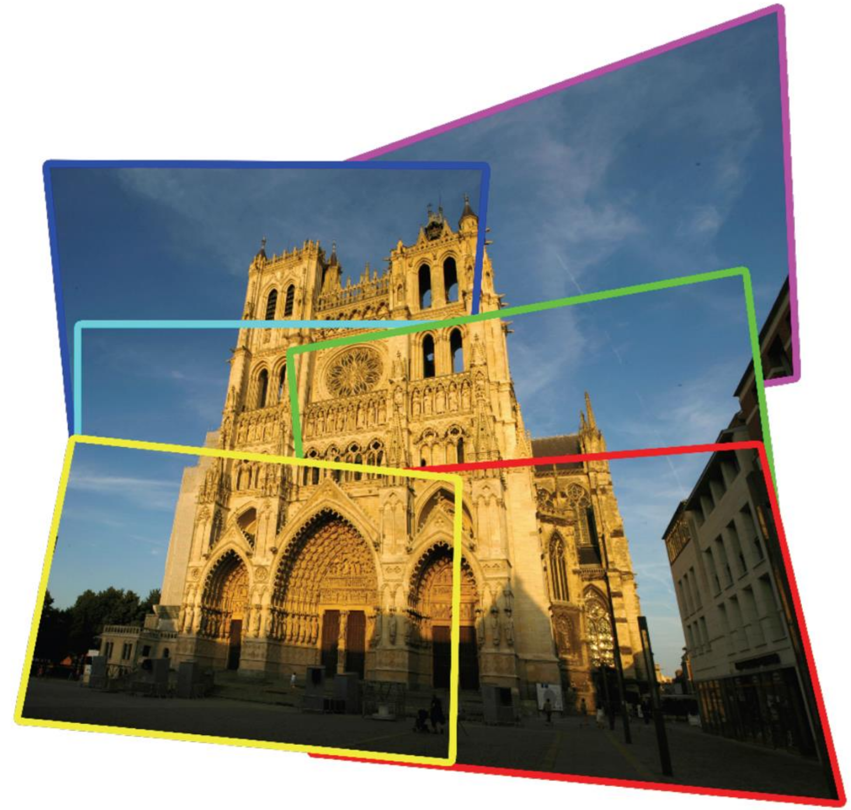
Overlaid Aligned Images

Blended Image

Example:



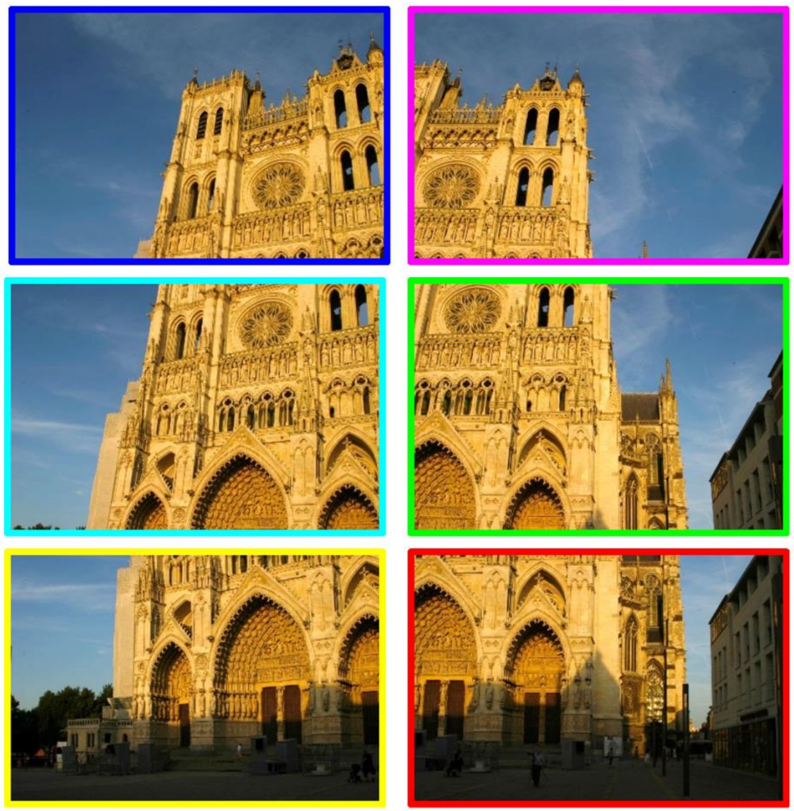
Source Images



Aligned Images

Pictures taken of Notre-Dame in Paris

Example:



Source Images

Blended Image

Pictures taken of Notre-Dame in Paris

Acknowledgements

- Thanks to the following researchers for making their teaching/research material online
  - Forsyth
  - Steve Seitz
  - Noah Snavely
  - J.B. Huang
  - Derek Hoiem
  - D. Lowe
  - A. Bobick
  - S. Lazebnik
  - K. Grauman
  - R. Zaleski
  - Leibe