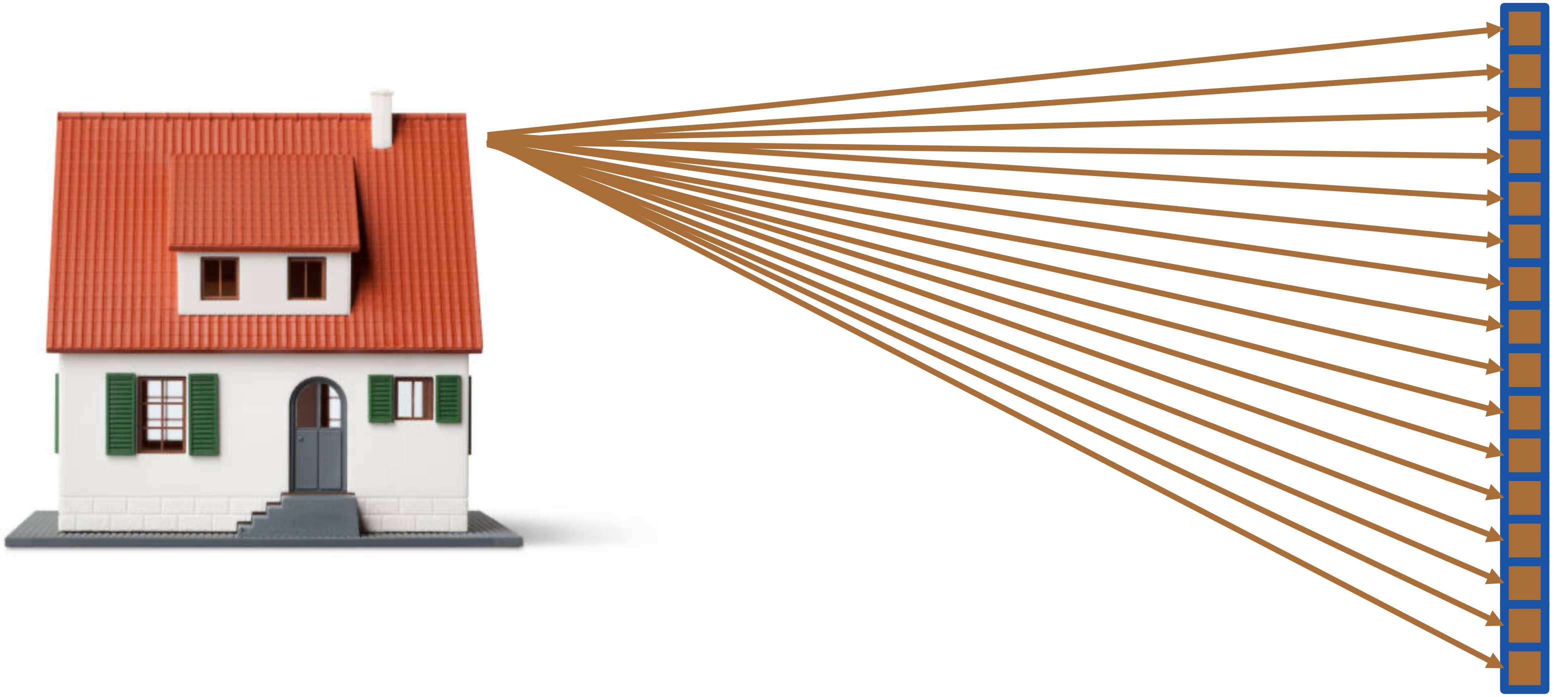
Projections and transformations

Mårten Björkman

Pinhole cameras

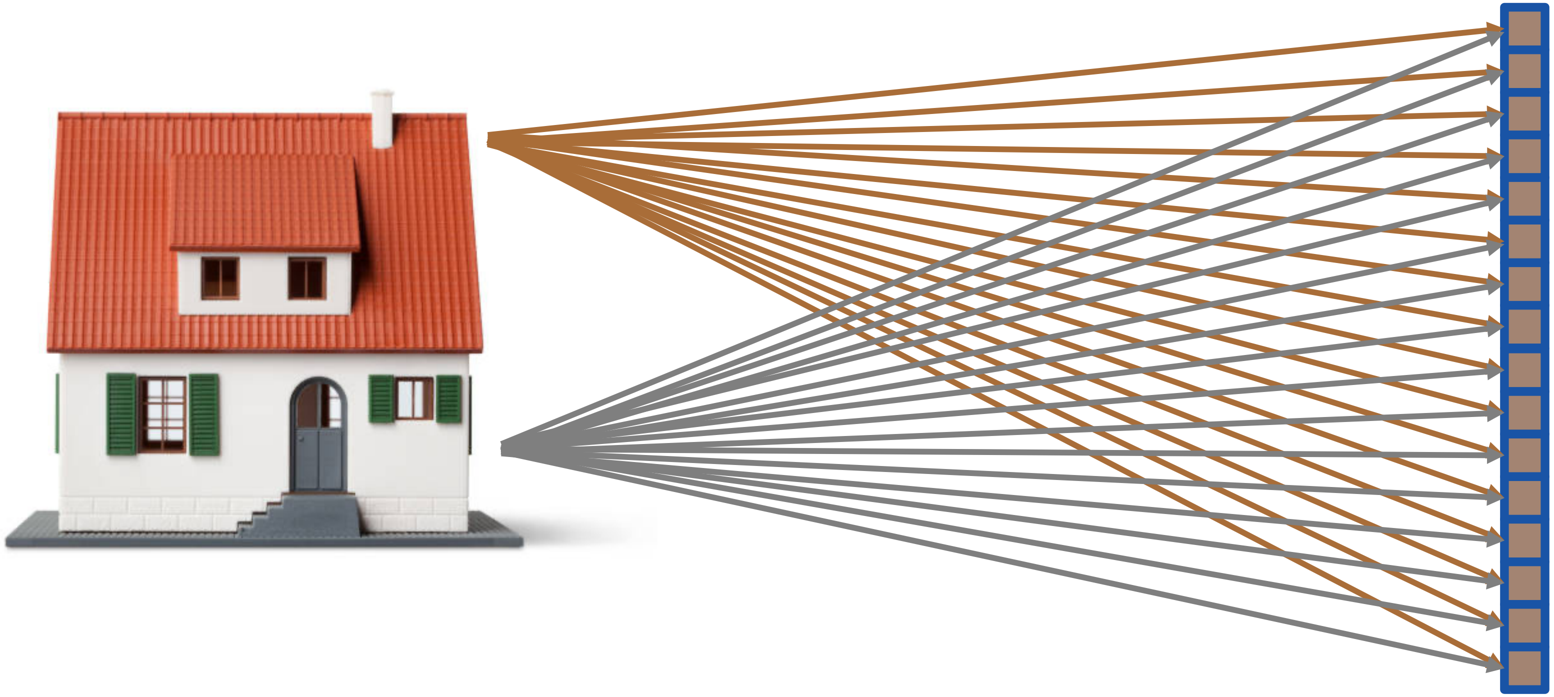
Assume we have a scene and a camera with a camera chip inside.

Pinhole cameras



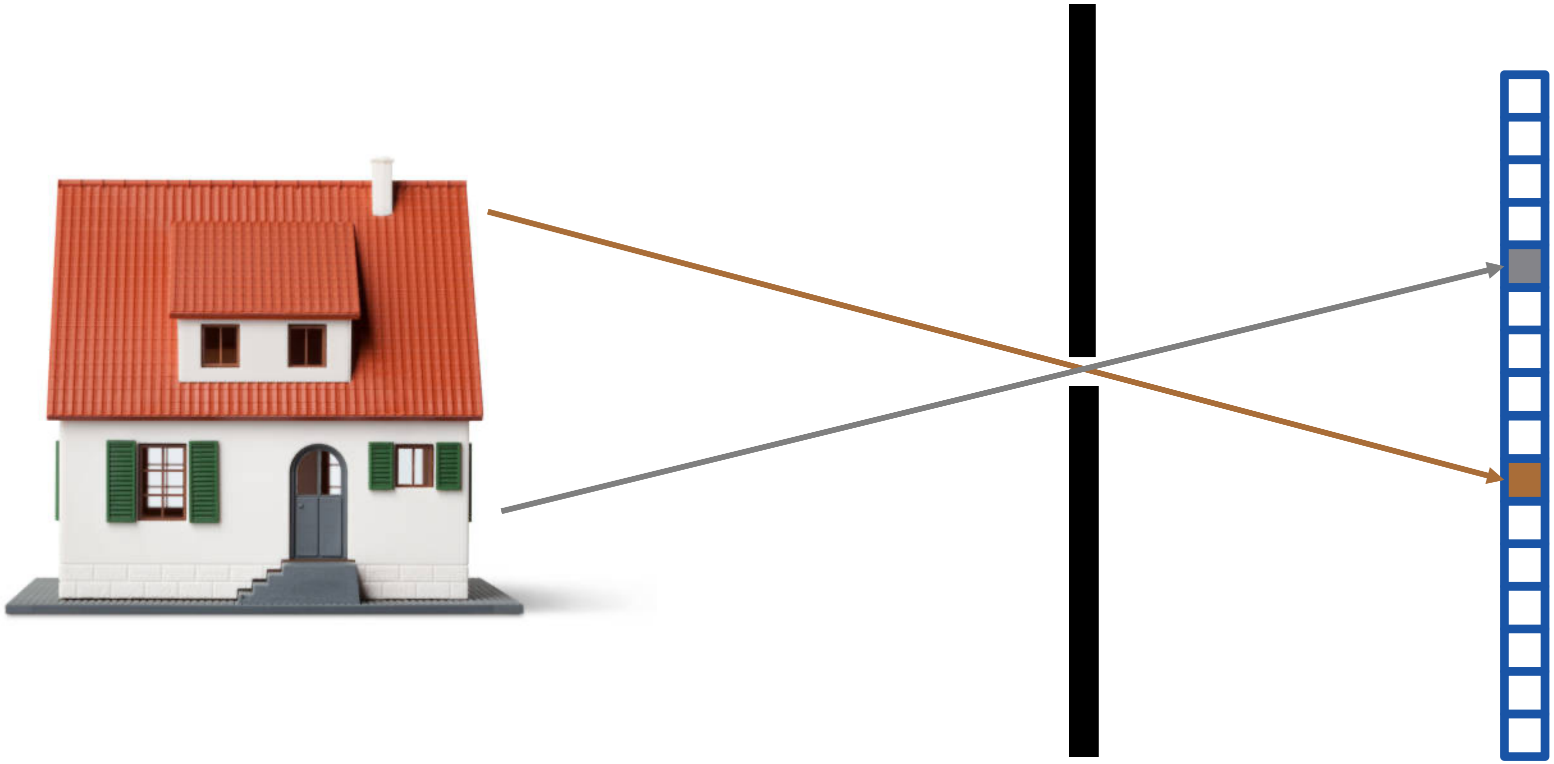
If we just let light into the camera, light from the same scene point will reach all sensors.

Pinhole cameras



If we just let light into the camera, light from the same point will spread to all sensors.

Pinhole cameras

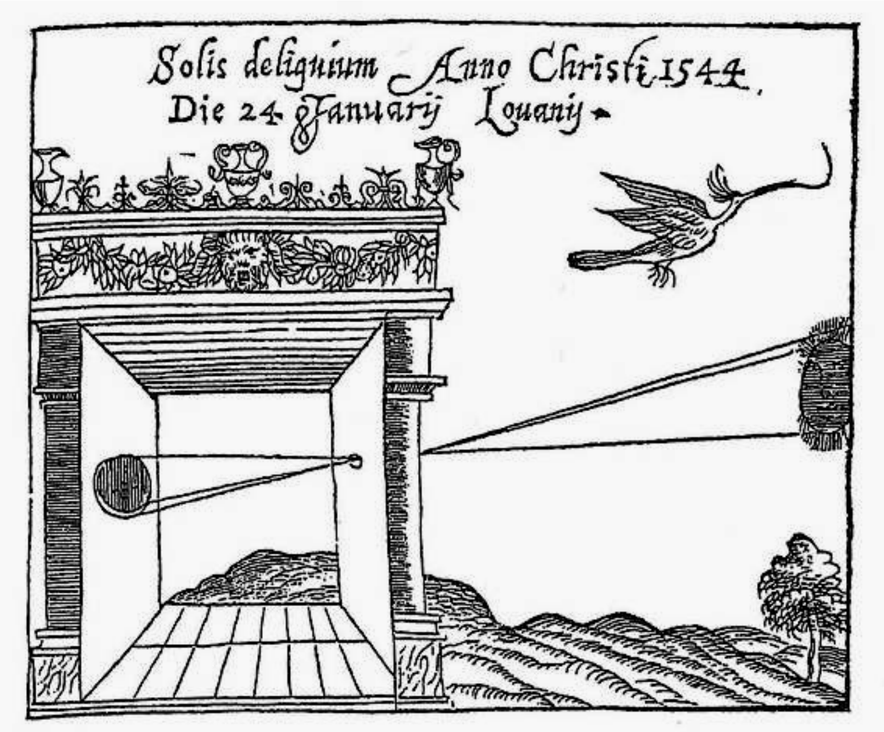


Instead we place a tiny little hole (pinhole) infront of the camera ship.

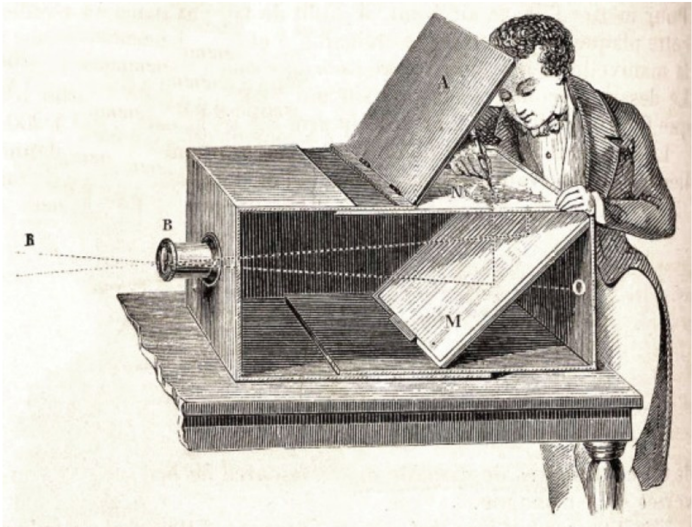
Pinhole cameras

On the camera chip we get a rescaled and rotated image of the scene.

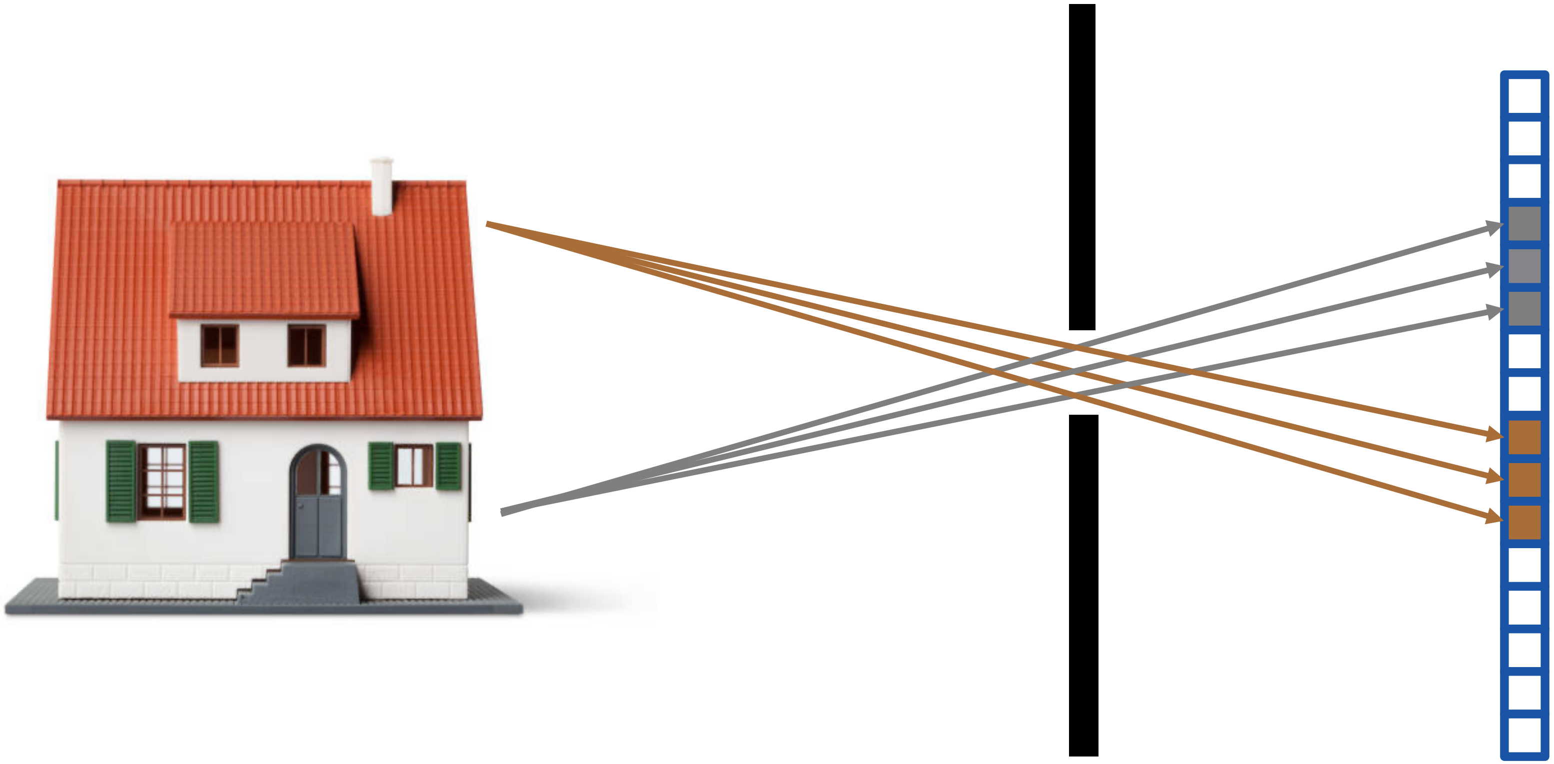
Pinhole camera – Camera Oscura



- Basic principle known since
  - Chinese philosopher Mozi (470-390 BC)
  - Greek philosopher Aristotle (384-322 BC)
- Used as drawing aid for artists
  - Leonardo da Vinci (1452-1519)



Pinhole cameras

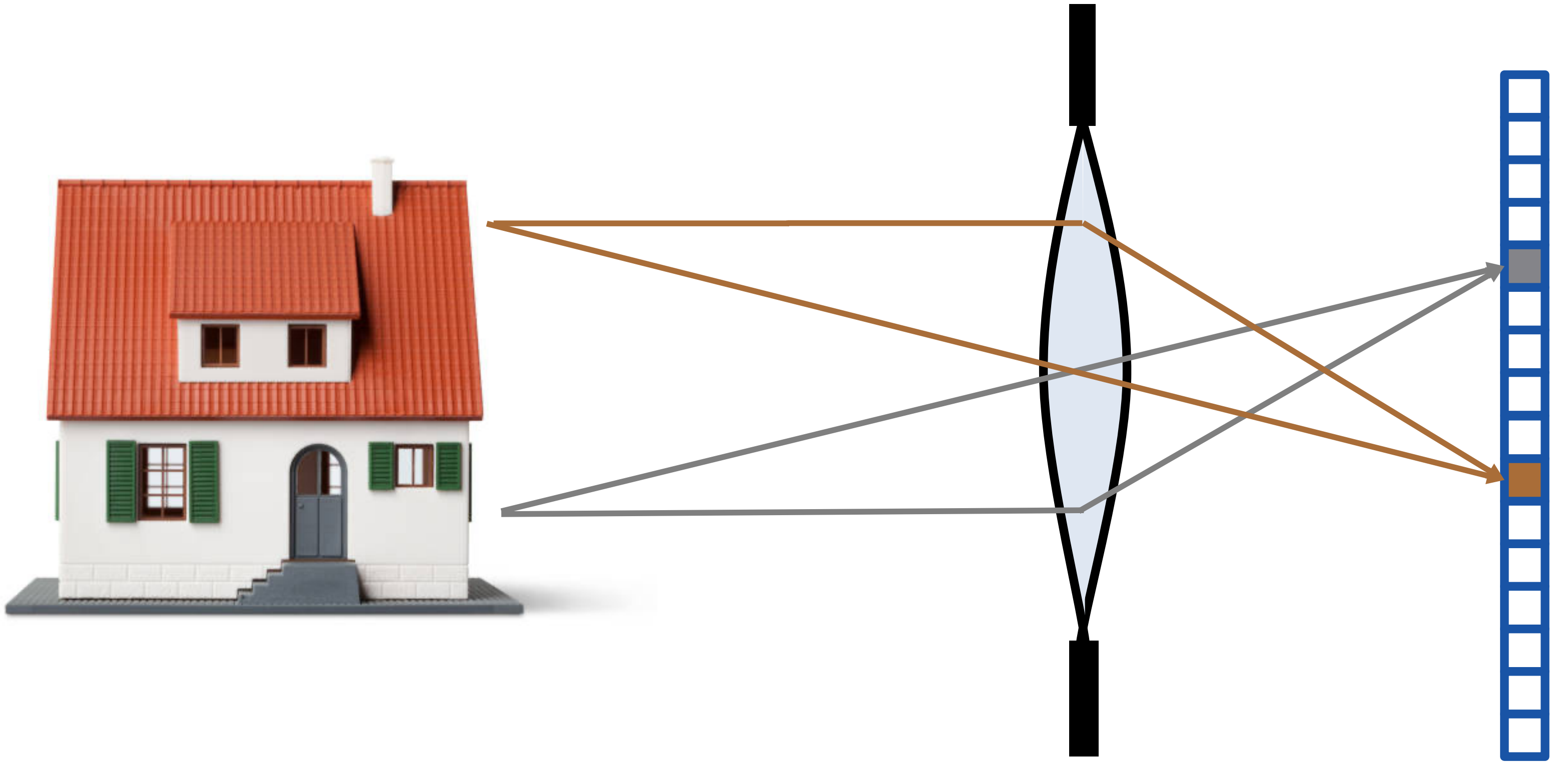


If the pinhole is too large, light will spread to multiple sensors.

Pinhole cameras

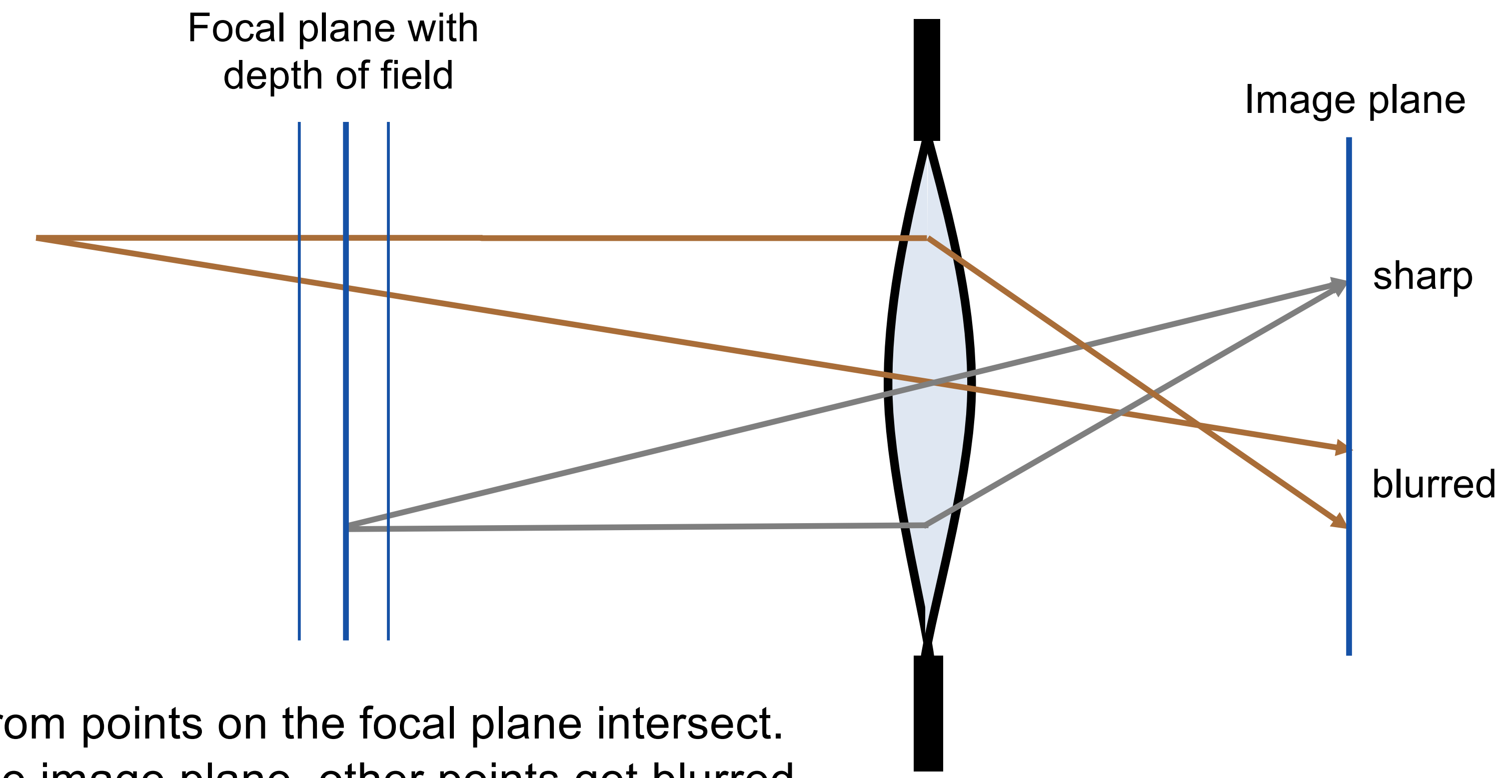
On the camera chip we get a blurred version of the image instead.

Pinhole cameras - lenses



With a lens we can have a larger hole, but without blurring the image

Pinhole cameras - lenses



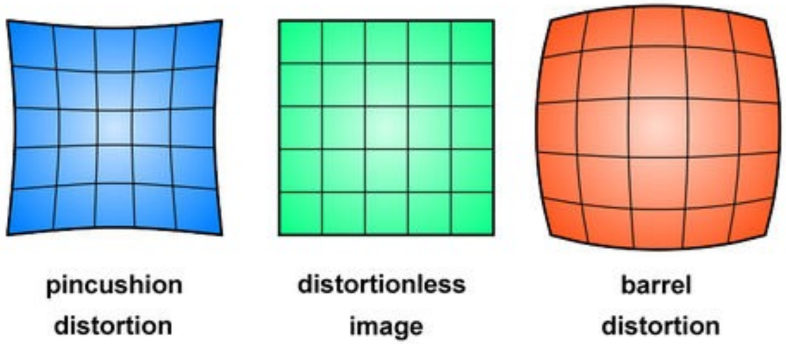
Problem: only light from points on the focal plane intersect.
the same point on the image plane, other points get blurred.

Depth of field: range of distances for which the image is sharp enough.

Lenses - depth of field

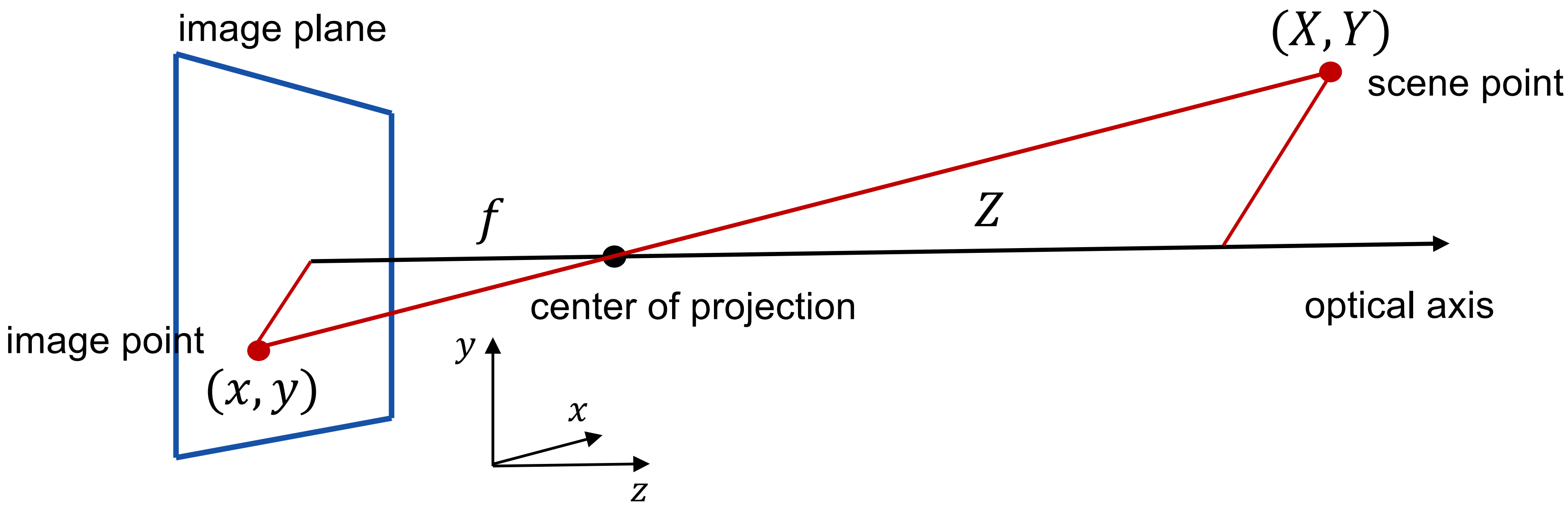
- The fact that the background gets blurred is not just effective in photography.
- It also seems natural, given that focal blur is a strong cue for human depth perception.
- In fact, people often miss this when trying to create fake images.

Lenses – lens distortion



- Problem: the thickness of lenses often leads to lens distortion, straight lines in reality become bent in the image.
- There are methods for correcting for this, but we will not cover those here.

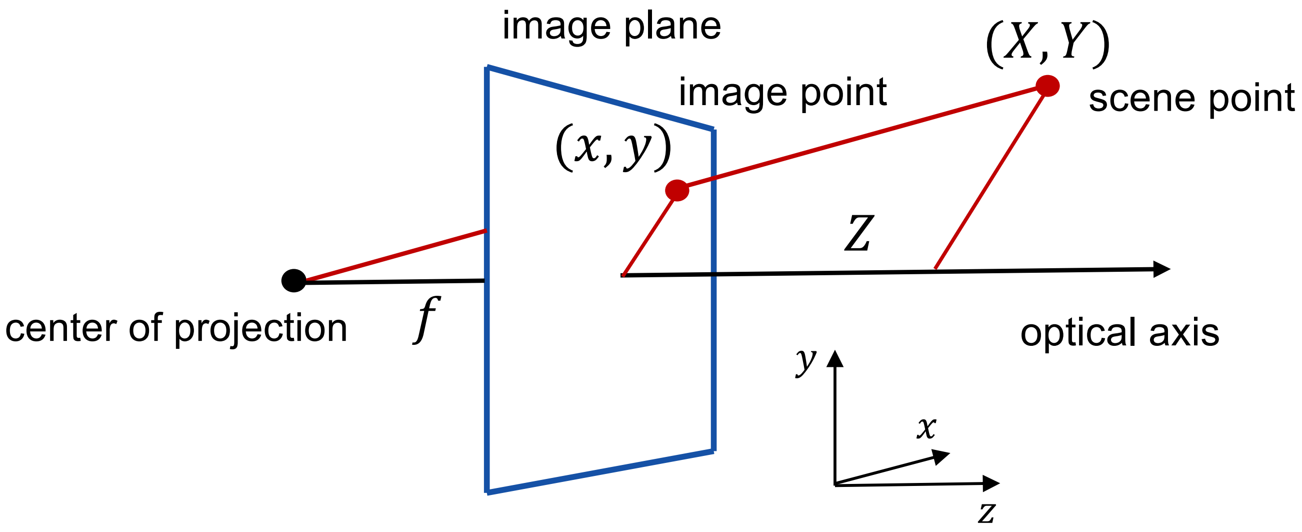
Perspective projection



- A scene point is projected to an image point with all light passing through the pinhole (centre of projection).
- We have two triangles of equal shape.

$$\frac{x}{f} = \frac{X}{Z} \Rightarrow x = f\,\frac{X}{Z}$$
$$\frac{y}{f} = \frac{Y}{Z} \Rightarrow y = f\,\frac{Y}{Z}$$

Perspective projection



- An equivalent model that is easier to draw.
- As if you are looking through a window.

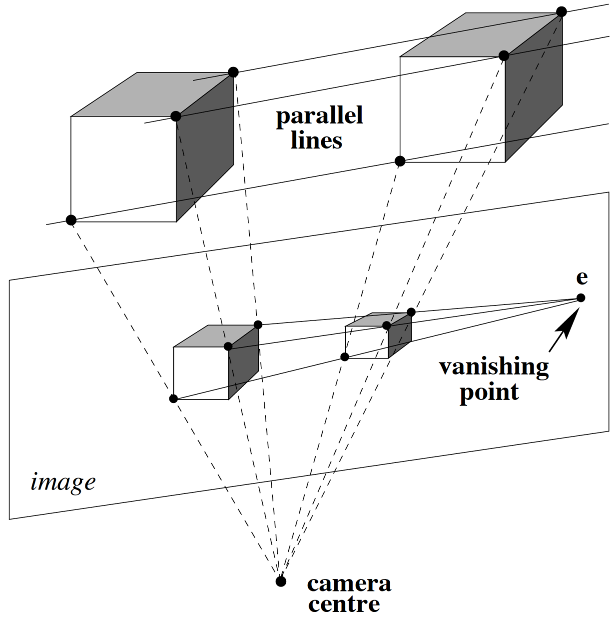
$$\frac{x}{f} = \frac{X}{Z} \Rightarrow x = f\,\frac{X}{Z}$$
$$\frac{y}{f} = \frac{Y}{Z} \Rightarrow y = f\,\frac{Y}{Z}$$

Perspective transformation

- A perspective transformation has three components:
  - Rotation - from world to camera coordinate system
  - Translation - from world to camera coordinate system
  - Perspective projection - from camera to image coordinates
- Basic properties which are preserved:
  - lines project to lines
  - tangencies
  - intersections

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \xrightarrow[\text{Rotation}]{} \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \xrightarrow[\text{Translation}]{} \begin{pmatrix} X' + t_x \\ Y' + t_y \\ Z' + t_z \end{pmatrix} \xrightarrow[\text{Projection}]{} \begin{pmatrix} x \\ y \end{pmatrix}$$

Perspective transformation - vanishing lines



- Parallel lines are usually not preserved, but intersect in a vanishing point in the image.
- Sets of parallel lines on the same plane lead to collinear vanishing points, which can be called a horizon of the plane.

Homogeneous coordinates

- Two problems with Euclidean coordinates:
  - Projections include a division, which makes them non-linear.
  - If you try to intersect two lines that are parallel, you get a division by zero.
- Solution: Instead use homogeneous coordinates
  - Extend all vectors with one extra dimension
  - Benefits: all we need can be expressed as matrix operations without divisions

Euclidean | Homogeneous

$$\begin{pmatrix} x \\ y \end{pmatrix} = \frac{f}{Z} \begin{pmatrix} X \\ Y \end{pmatrix} \quad \Rightarrow \quad \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \simeq \begin{pmatrix} fX \\ fY \\ Z \end{pmatrix}$$

Homogeneous coordinates

- Going from Euclidean to homogeneous coordinates
  - Add an extra dimension and set it to 1.

$$\begin{pmatrix} x \\ y \end{pmatrix} \Rightarrow \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \simeq \begin{pmatrix} kx \\ ky \\ k \end{pmatrix}$$

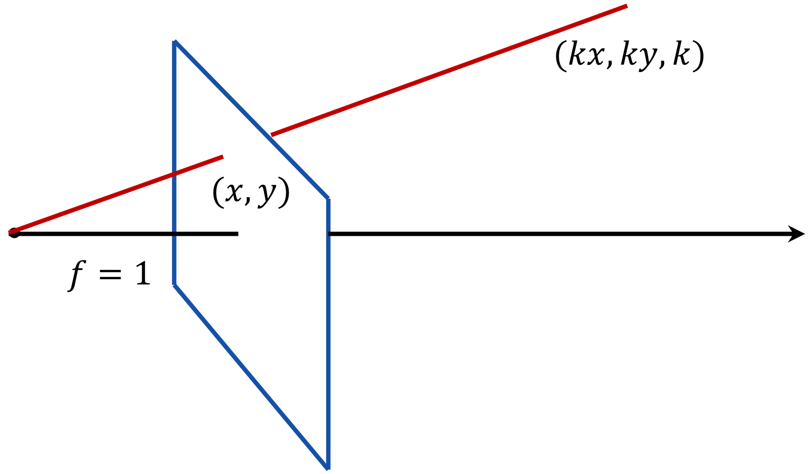
- Going from homogeneous to Euclidean coordinates
  - Divide the first coordinates by the last coordinate.

$$\begin{pmatrix} x' \\ y' \\ w' \end{pmatrix} \Rightarrow \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} x'/w' \\ y'/w' \end{pmatrix}$$

- Note: the scale of vectors and matrices does not matter.
  - If you rescale a vector, we get a new equivalent one.

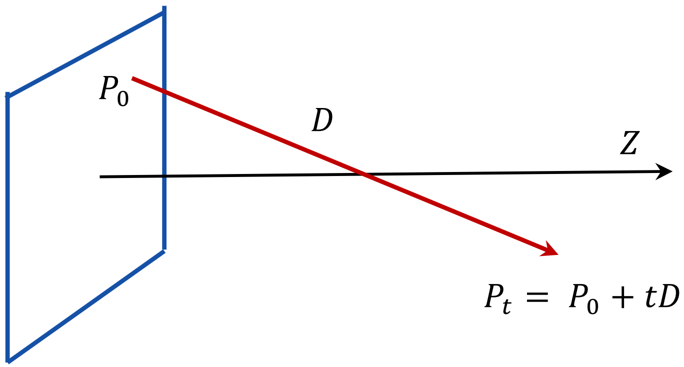
$$\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \simeq \begin{pmatrix} k_1 x \\ k_1 y \\ k_1 \end{pmatrix} \simeq \begin{pmatrix} k_2 x \\ k_2 y \\ k_2 \end{pmatrix}$$

Homogeneous coordinates



- It is like having a point (x, y) in 2D, but thinking of it as a line (kx, ky, k) in 3D.
- Similarly, a 3D point (X, Y, Z) can be written as $(X, Y, Z, 1)$ in homogeneous coordinates, which is equivalent to (cX, cY, cZ, c) for all c.

A vanishing line



In homogeneous coordinates:

$$P_t = \begin{pmatrix} P_x + tD_x \\ P_y + tD_y \\ P_z + tD_z \\ 1 \end{pmatrix} \simeq \begin{pmatrix} P_x/t + D_x \\ P_y/t + D_y \\ P_z/t + D_z \\ 1/t \end{pmatrix} \rightarrow \begin{pmatrix} D_x \\ D_y \\ D_z \\ 0 \end{pmatrix}$$

as $t \rightarrow \infty$

- We follow a line out into the infinity and look at the homogeneous coordinates.
  - At infinity, the starting point P_0 does not matter, only the direction D does.
  - Two parallel lines, with different starting points, will 'intersect' at infinity.
  - If the last homogeneous coordinate is 0, you have a direction, not a point.

Perspective projection

$$\begin{pmatrix} x \\ y \end{pmatrix} = \frac{f}{Z} \begin{pmatrix} X \\ Y \end{pmatrix} \quad \Rightarrow \quad \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \simeq \underbrace{\begin{pmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}}_{P} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} = \begin{pmatrix} fX \\ fY \\ Z \end{pmatrix}$$

- The projection be expressed as a (simple) matrix multiplication.
- The matrix P is called a projection matrix, since the dimensionality decreases.
- Note: the division is delayed until you go back to Euclidean coordinates,

Homogeneous coordinates – transformations

- Almost everything we need can be expressed as matrix multiplications

Translation:

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \rightarrow \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} + \begin{pmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{pmatrix} \qquad \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \rightarrow \begin{pmatrix} 1 & 0 & 0 & \Delta X \\ 0 & 1 & 0 & \Delta Y \\ 0 & 0 & 1 & \Delta Z \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

Scaling:

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \rightarrow \begin{pmatrix} S_X X \\ S_Y Y \\ S_Z Z \end{pmatrix} \qquad \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \rightarrow \begin{pmatrix} S_X & 0 & 0 & 0 \\ 0 & S_Y & 0 & 0 \\ 0 & 0 & S_Z & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

Homogeneous coordinates – transformations

Common case: Rigid body transformations (Euclidean)

$$\begin{pmatrix} X' \\ Y' \\ Z' \end{pmatrix} \rightarrow R \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} + \begin{pmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{pmatrix}$$

where R is a rotation matrix ($R^{-1} = R^T$) is written

$$\begin{pmatrix} X' \\ Y' \\ Z' \\ 1 \end{pmatrix} = \begin{pmatrix} & & & \Delta X \\ & R & & \Delta Y \\ & & & \Delta Z \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

Perspective transformation

Consider world coordinates $(X', Y', Z', 1)$ expressed in a coordinate system not aligned with the camera coordinate system.

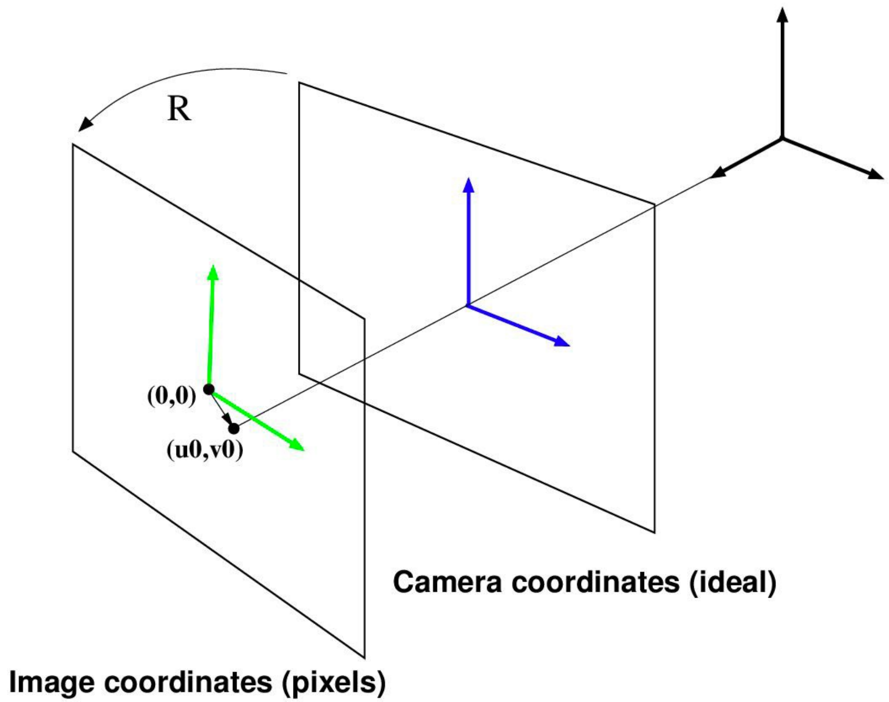
$$\begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} = \begin{pmatrix} & & & \Delta X \\ & R & & \Delta Y \\ & & & \Delta Z \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} X' \\ Y' \\ Z' \\ 1 \end{pmatrix} = A \begin{pmatrix} X' \\ Y' \\ Z' \\ 1 \end{pmatrix}$$

Perspective projection (more general later)

$$\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \simeq \begin{pmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} = PA \begin{pmatrix} X' \\ Y' \\ Z' \\ 1 \end{pmatrix} = M \begin{pmatrix} X' \\ Y' \\ Z' \\ 1 \end{pmatrix}$$

The matrix M is also a 3×4 projection matrix, but include a rigid transformation.

Intrinsic camera parameters



- Due to imperfect placement of the camera chip relative to lens system, there is always a small relative rotation and shift of centre position.

Intrinsic camera parameters

A more general projection matrix allows image coordinates with an offset origin, non-square pixels and skewed coordinate axes.

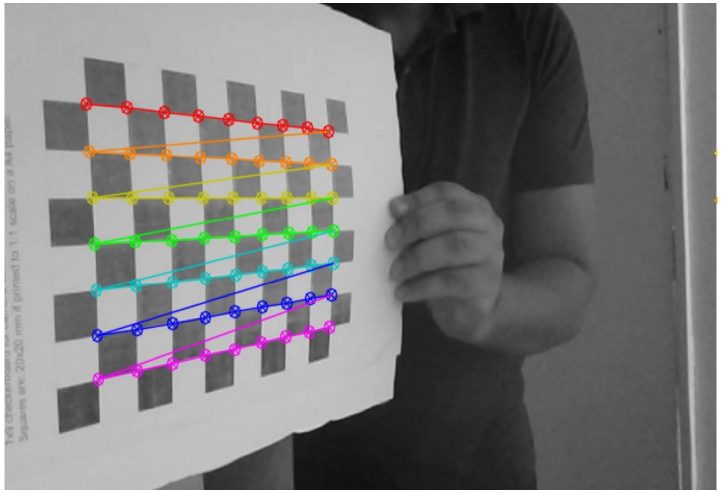
camera matrix

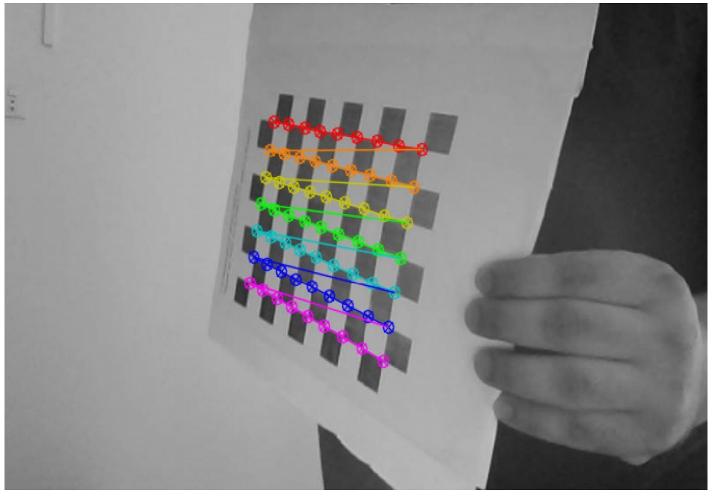
projection matrix

$$K = \begin{pmatrix} f_u & \gamma & u_0 \\ 0 & f_v & v_0 \\ 0 & 0 & 1 \end{pmatrix}, \qquad P = \begin{pmatrix} K & 0 \end{pmatrix} = \begin{pmatrix} f_u & \gamma & u_0 & 0 \\ 0 & f_v & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

- Five variables known as the camera's <u>intrinsic parameters</u>:
  - Focal lengths (f_u, f_v) are often assumed to be equal, $f = f_u = f_v$.
  - Skew γ captures small rotations and is often close to zero.
  - Principal point (u_0, v_0) is often close to image centre.
- The relative rotation R and translation t between world and camera coordinate systems are known as the <u>extrinsic parameters</u>.

Camera calibration

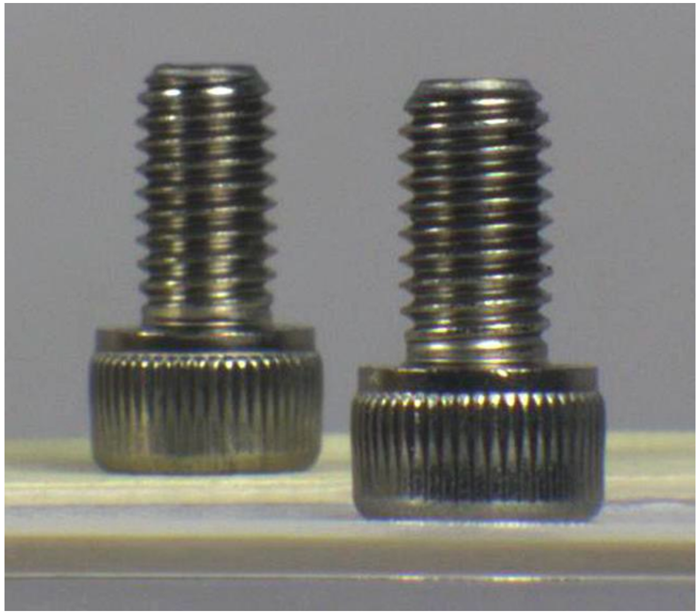




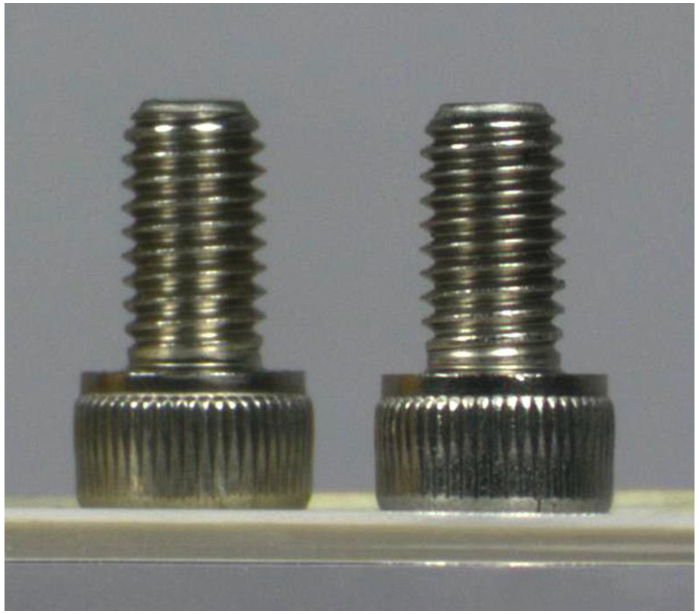
- The camera matrix K, with the five intrinsic parameters, can be estimated by
  - Detect image features from a known object in different orientations.
  - Find feature positions (x, y) for which you know the corresponding (X, Y, Z).
  - Solve for the intrinsic parameters, rotation R and translation T using a large system of equations.

Weak perspective

- Increase the focal length f (zoom in), while moving further away. Then the perspective effect will gradually disappear.



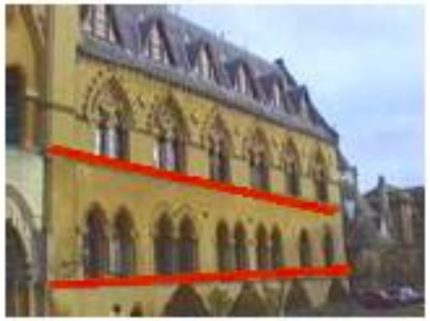
Perspective camera

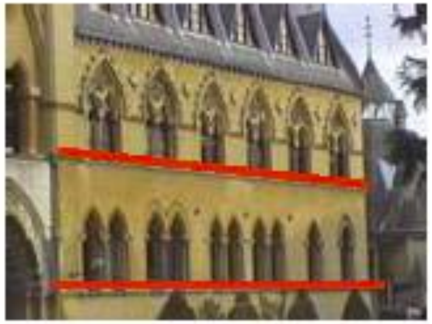


Weak perspective camera

Weak perspective

$$\begin{pmatrix} x \\ y \end{pmatrix} = \frac{f Z_0}{Z + Z_0} \begin{pmatrix} X \\ Y \end{pmatrix} \rightarrow f \begin{pmatrix} X \\ Y \end{pmatrix} \quad \Rightarrow \quad \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \simeq \begin{pmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$





- The effect of the depth Z will disappear, when Z_0 is large enough.
- The world is flattened, since everything appears to be on the same distance.
- Parallel lines in the world, become parallel also in the projection.

Akira Kurusawa used telescopic lenses to ’flatten’ his movies to create more action, when you lose the sense of depth and distances.

Two projection models

When combined with a rigid transformation (rotation and translation)

- Perspective (11 degrees of freedom): – under normal condition

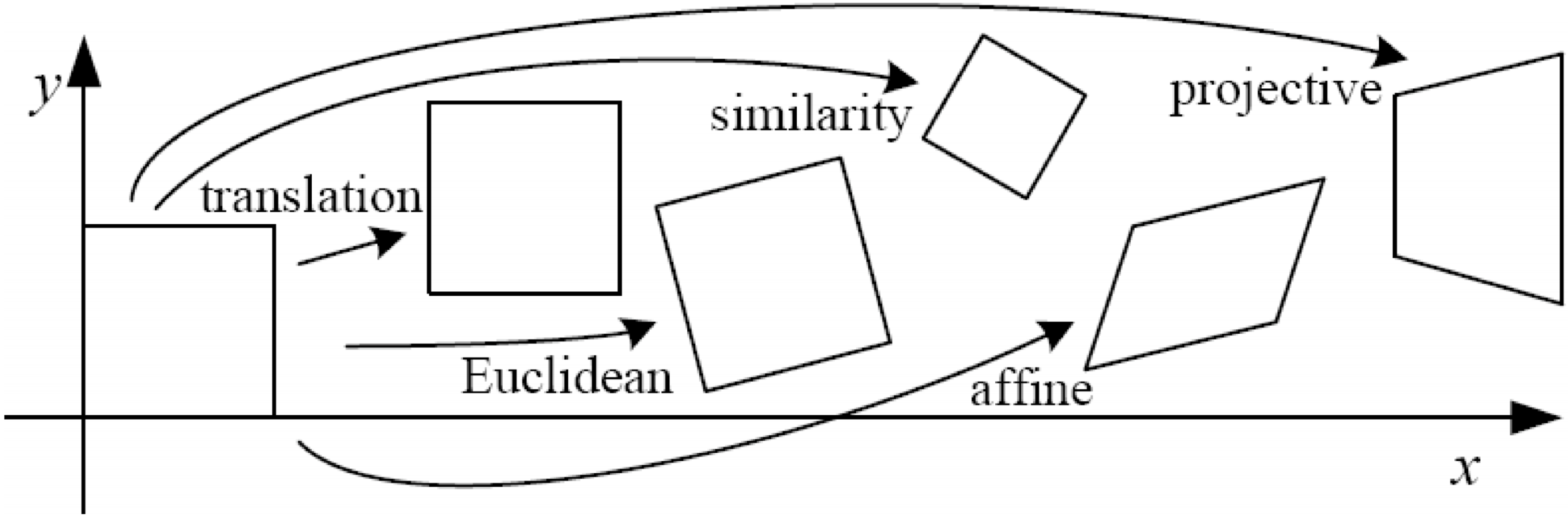
$$M = \begin{pmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \end{pmatrix}$$

- Affine (8 degrees of freedom): – when the perspective is too weak

$$M = \begin{pmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

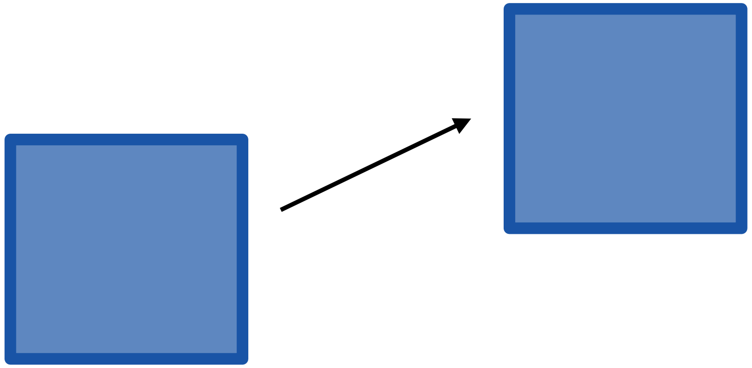
Note: The choice of model does not change the image (like computer graphics). Rather, select the model based on how the images look. Parallel lines?

Image transformations



Transformations from 2D to 2D, not from 3D to 2D.

Image transformations - Translation

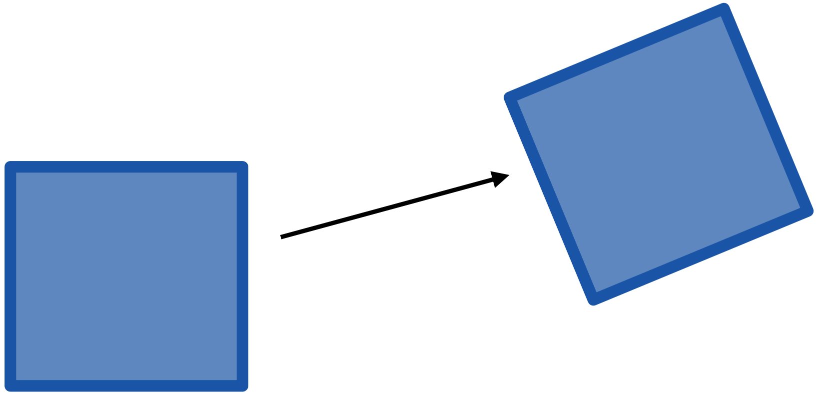


$$\begin{pmatrix} x \\ y \end{pmatrix} \rightarrow \begin{pmatrix} x + t_x \\ y + t_y \end{pmatrix}$$

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

- Two degrees of freedom
  - Translation x-wise and y-wise (2 dof)

Image transformations - Euclidean

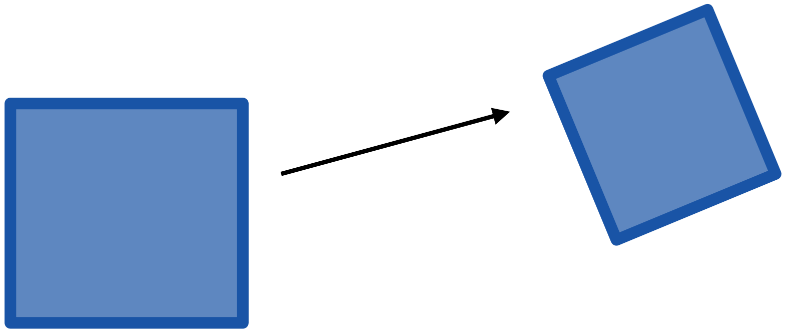


$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} \simeq \begin{pmatrix} c & -s & t_x \\ s & c & t_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

$$c^2 + s^2 = 1$$

- Three degrees of freedom
  - Translation x-wise and y-wise (2 dof)
  - Rotation (1 dof)

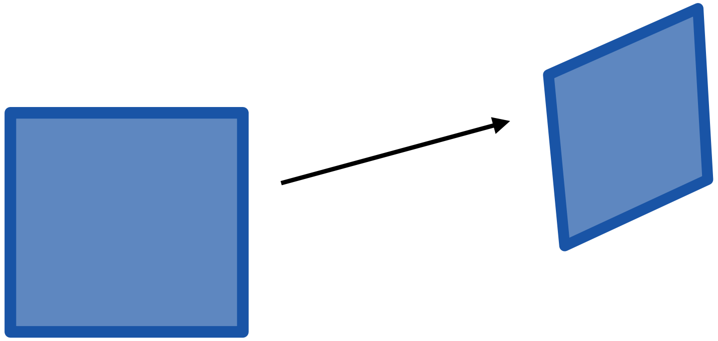
Image transformations - Similarity



$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} \simeq \begin{pmatrix} c & -s & t_x \\ s & c & t_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

- Four degrees of freedom
  - Translation x-wise and y-wise (2 dof)
  - Rotation (1 dof)
  - Scaling (1 dof)

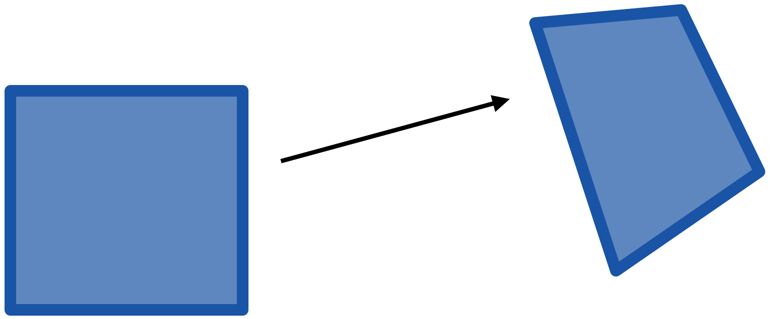
Image transformations - Affine



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

- Six degrees of freedom
  - Translation x-wise and y-wise (2 dof)
  - Rotation (1 dof)
  - Scaling (1 dof)
  - Shearing x-wise and y-wise (2 dof)

Image transformations - Projective



Homography

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} \simeq \begin{pmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

- Eight degrees of freedom
  - Translation x-wise and y-wise (2 dof)
  - Rotation (1 dof)
  - Scaling (1 dof)
  - Shearing x-wise and y-wise (2 dof)
  - Foreshortening x-wise and y-wise (2 dof)

Matching planes with homographies

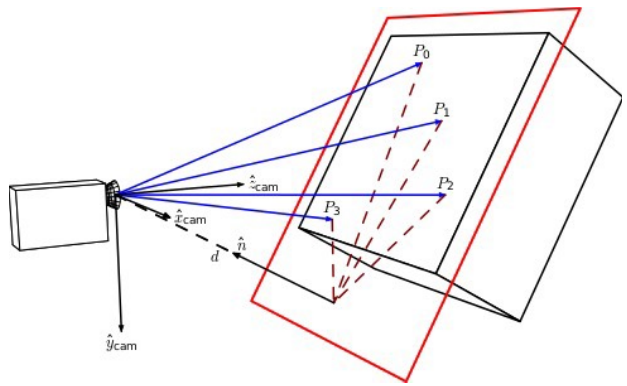
- Projections are normally not invertible.

$$\begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} \simeq \begin{bmatrix} p_{00} & p_{01} & p_{02} & p_{03} \\ p_{10} & p_{11} & p_{12} & p_{13} \\ p_{20} & p_{21} & p_{22} & p_{23} \end{bmatrix} \begin{bmatrix} X_i \\ Y_i \\ Z_i \\ 1 \end{bmatrix}$$

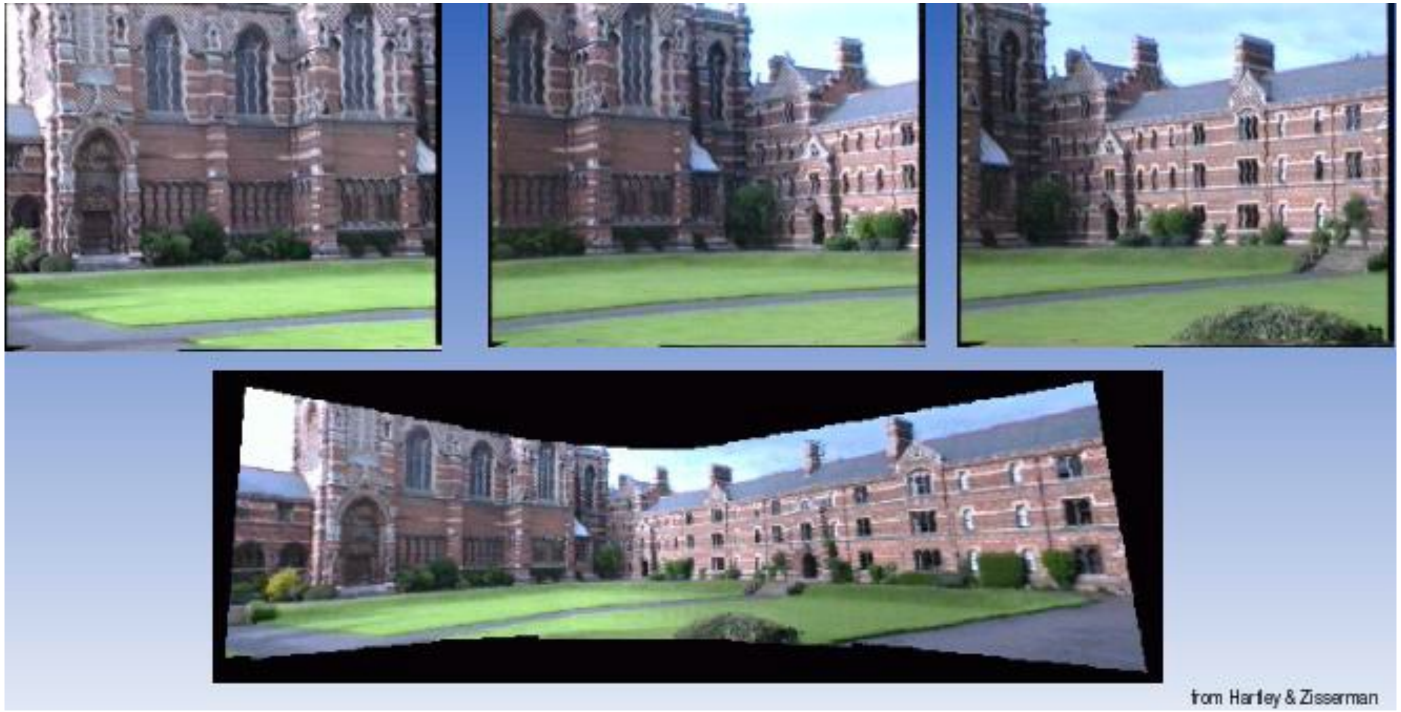
- However, for a plane defined by $Z_i = 0$, it can (usually) be inverted.

$$\begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} \simeq \begin{bmatrix} p_{00} & p_{01} & p_{03} \\ p_{10} & p_{11} & p_{13} \\ p_{20} & p_{21} & p_{23} \end{bmatrix} \begin{bmatrix} X_i \\ Y_i \\ 1 \end{bmatrix}$$

- If the world is flat, we can find a homography to map the image from one view to another.



Example: Mosaicing – creating a panorama



- Since the world is not a plane, there will be artefacts in the stitching.
- However, if the world is far away, the world looks almost flat.

Example: Image stitching video

Image stitching can be used to combine images into a larger mosaic of higher resolution.

Image warping

- Resample image $f(x, y)$ to get a new image $g(u, v)$, using a coordinate transformation: $u = u(x, y), v = v(x, y)$.
- Examples of transformations:

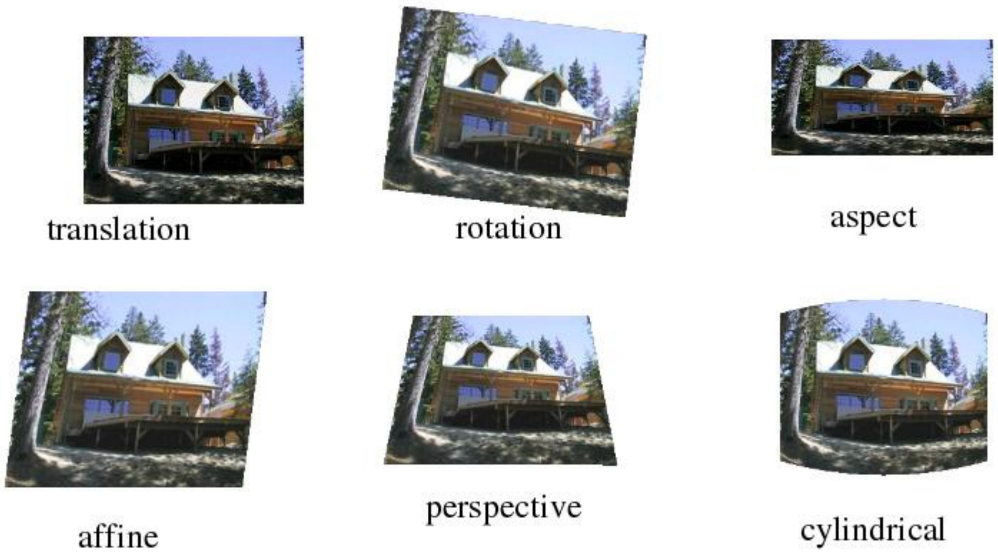
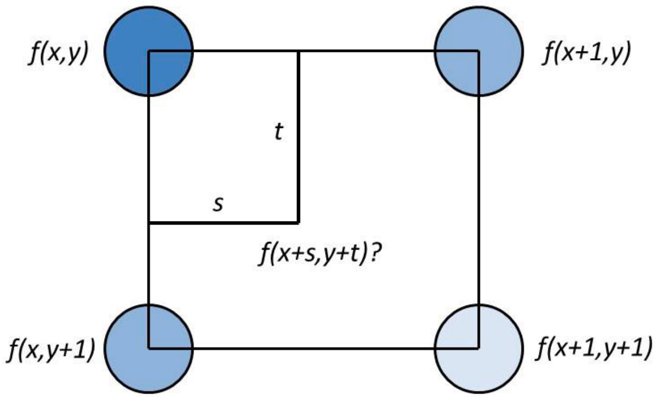


Image warping

- For each grid point in (u, v) domain compute corresponding (x, y) values.
  Note: transformation is inverted to avoid holes in result.
- Create $g(u, v)$ by sampling from $f(x, y)$ either by:
  - Nearest neighbour look-up (noisy result)
  - Bilinear interpolation (blurry result)



$$\begin{aligned} f(x+s, y+t) &= (1-t) \cdot ((1-s) \cdot f(x,y) + s \cdot f(x+1,y)) + \\ &+ \quad t \cdot ((1-s) \cdot f(x,y+1) + s \cdot f(x+1,y+1)) \end{aligned}$$

Nearest Neighbour vs Bilinear Interpolation

Nearest Neighbour

Noisy!

Bilinear Interpolation

Blurry!

Summary of good questions

- What is a pinhole camera model?
- What is the difference between intrinsic and extrinsic camera parameters?
- How does a 3D point get projected to a pixel with a perspective projection?
- What are homogeneous coordinates and what are they good for?
- How is a perspective projection expressed in homogeneous coordinates?
- What is a vanishing point and how do you find it?
- What is an affine camera model?
- When to choose to a perspective model vs an affine one?

Recommended readings

- Gonzalez and Woods: Chapters 2.4 - 2.5
- Szeliski: Chapters 2.1 and 3.6.1