ECE 634: Digital Video Systems
Motion models: 1/19/17

Professor Amy Reibman
MSEE 356
reibman@purdue.edu

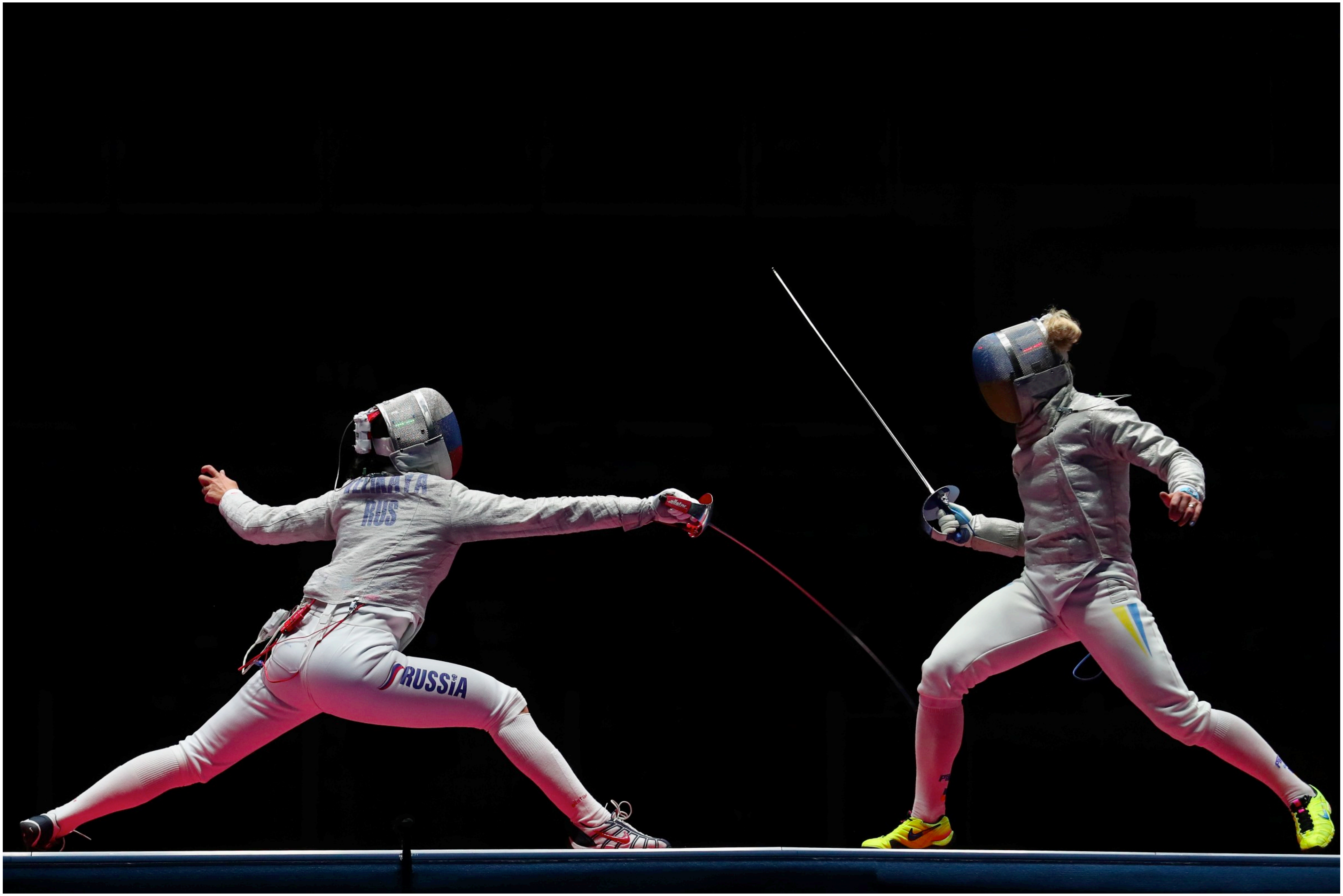
http://engineering.purdue.edu/~reibman/ece634/index.html

Outline

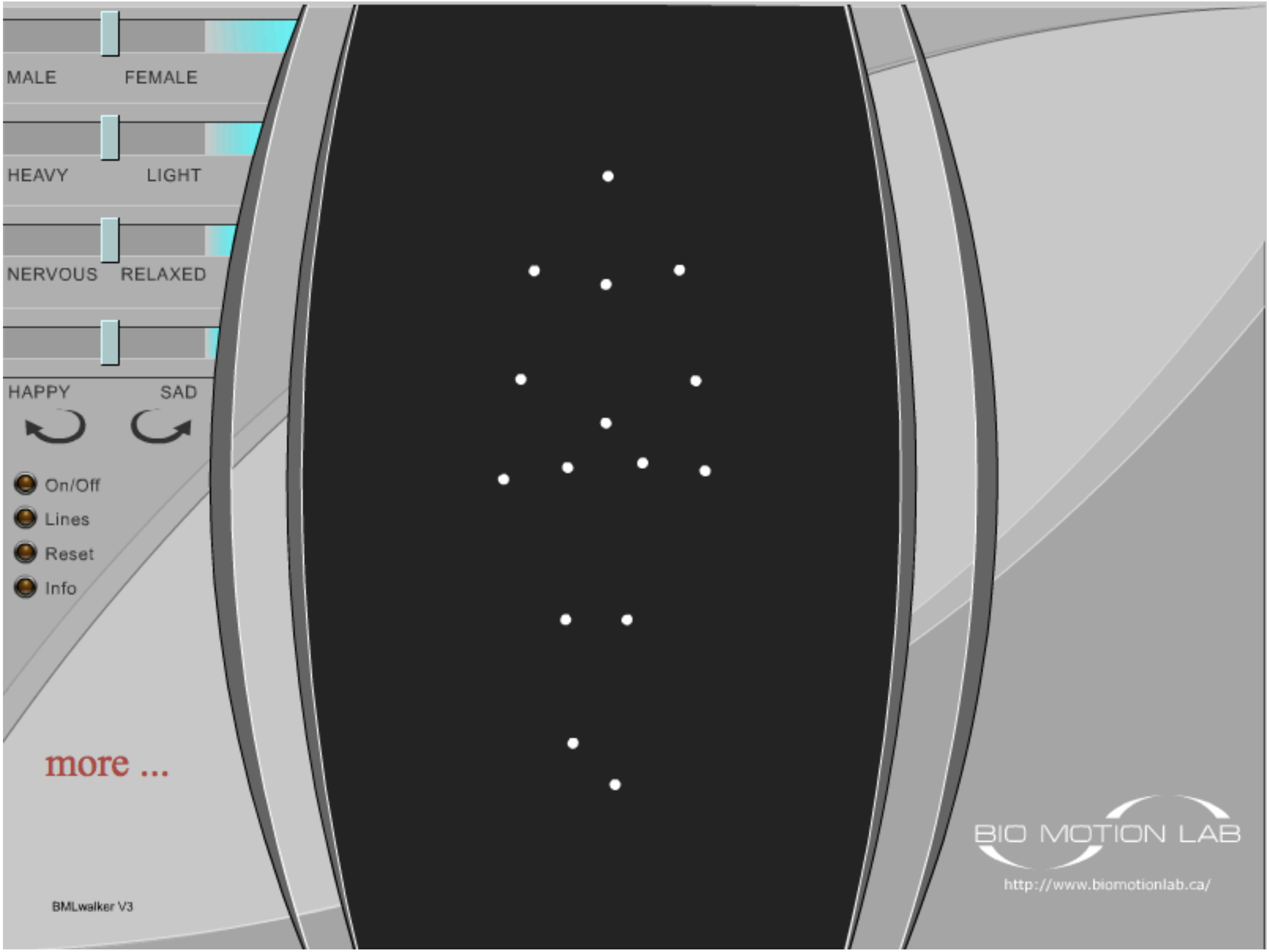
- Today: Motion models for motion estimation
  - Camera model and camera motion
  - Object motion
  - Models to represent motion
- Next: Estimating motion

Why study motion?

- Motion creates spatio-temporal information
  - Understand 3D motion
  - Better object recognition, identification, segmentation
  - Sometimes the information IS the motion (ex: action recognition)
  - Improve video quality through motion stabilization
- Objects themselves don't change much during motion
  - Remove temporal redundancies for compression
  - Motion-compensated temporal filtering (along motion trajectories) to remove noise
  - Motion-compensated frame interpolation


RUS
RUSSIA

Henri
Cartier-Bresson
1932
Derriere la Gare Saint-Lazare



https://www.biomotionlab.ca/Demos/BMLwalker.html

Reading resources

- J. Konrad, "Motion Detection and Estimation", Chapter 3 in A. Bovik (ed.), *The Essential Guide to Video Processing*, Elsevier, 2009.
- A. M. Tekalp, Digital video processing, Prentice Hall, 1995
  - Chapter 5: 5.1,5.2
  - Chapter 6: 6.1, 6.3, 6.4
- Y. Wang, J. Ostermann, and Y.-Q. Zhang, *Video Processing and Communications*, Prentice Hall, 2002.
  - Chapter 5.1, 5.3.2, 5.5: Video Modeling
  - Chapter 6.1-6.4, 6.7, 6.9, skip Sec. 6.4.5, 6.4.6: Two-dimensional motion estimation
  - Appendix A and B: Gradients and steepest descent

Motion detection

- Is an image region moving or stationary?
- We will postpone this discussion until later

Motion estimation

- Need accurate models for the motion
  - Understand the **camera** and its motion
  - Understand **objects** and their motion
- All we can see from the camera is the 2D projection of the 3D world
  - This mapping is not unique! Many 3D-worlds can produce the same 2D image
  - All our interpretations of the world must take place through this projection

Motion estimation applications

- Motion for **interpreting** 3D world requires an inverse mapping
- Motion for **compression** need not approximate physical motion; it is solely to reduce bit-rate
- **Processing** with motion is best if the true motion of image points can be obtained

Different applications/purposes of motion require different approaches to motion estimation

To estimate 2D motion we need..

- A motion model
- An estimation criterion
- A search strategy

- Today's class is about motion models

Summary of this class

- What causes 2D motion?
  - Camera motion
  - Object motion projected to 2D
- Camera model: 3D → 2D projection
  - Perspective projection vs. orthographic projection
- Models corresponding to typical camera motion and object motion
  - Piece-wise projective mapping is a good model for projected rigid object motion
  - Can be approximated by affine or bilinear functions
  - Affine functions can also characterize some global camera motions
- Ways to represent motion:
  - Pixel-based, block-based, region-based, global, etc.

2D Motion

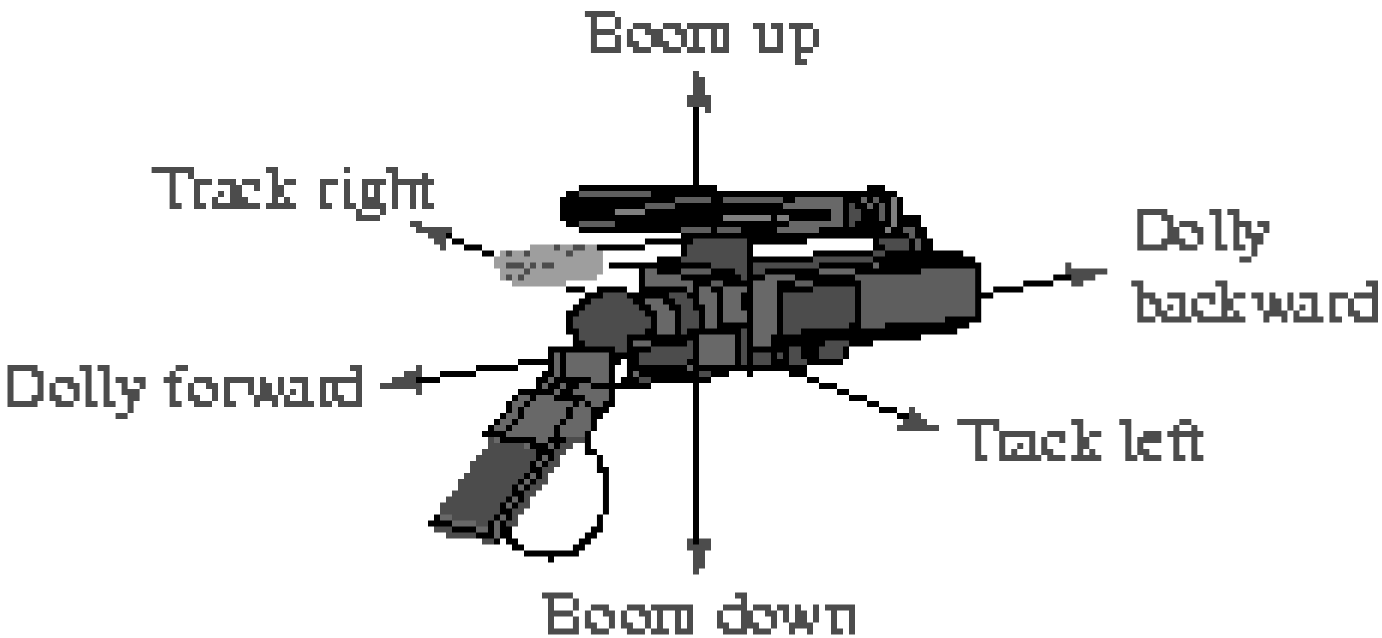
2D (or apparent) motion that is created by moving objects depends on 3 things:

1. An image formation (or camera) model
   – Perspective, orthographic, ..
2. Motion model of a 3D object (rigid body with 3D translation and rotation, 3D affine motion)
3. Surface model of 3D object (planar, parabolic..)

Camera models: outline

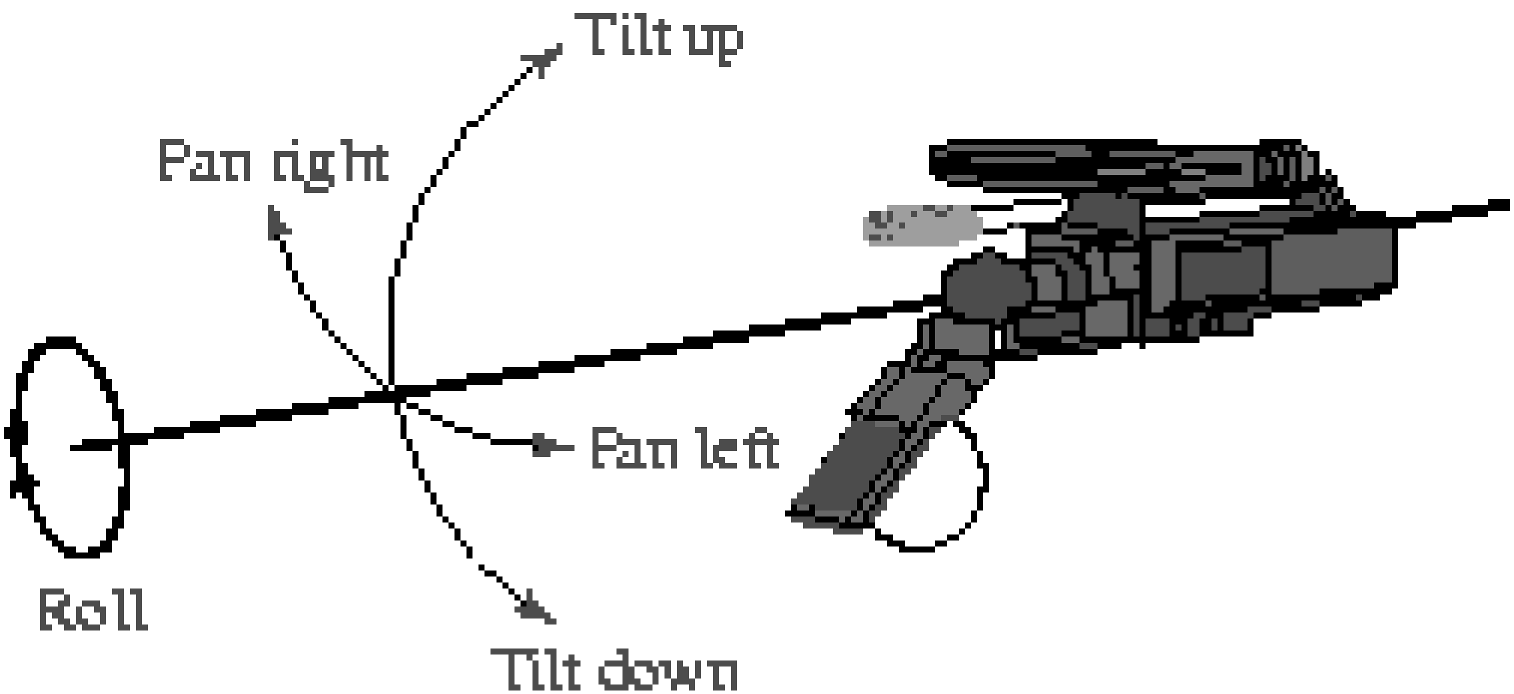
- Orient the camera in 3D space
- Basic camera model (pinhole)
- Simpler orthographic model
- More complex camera models exist
  - Example: CAHV

Translational camera motions in 3D



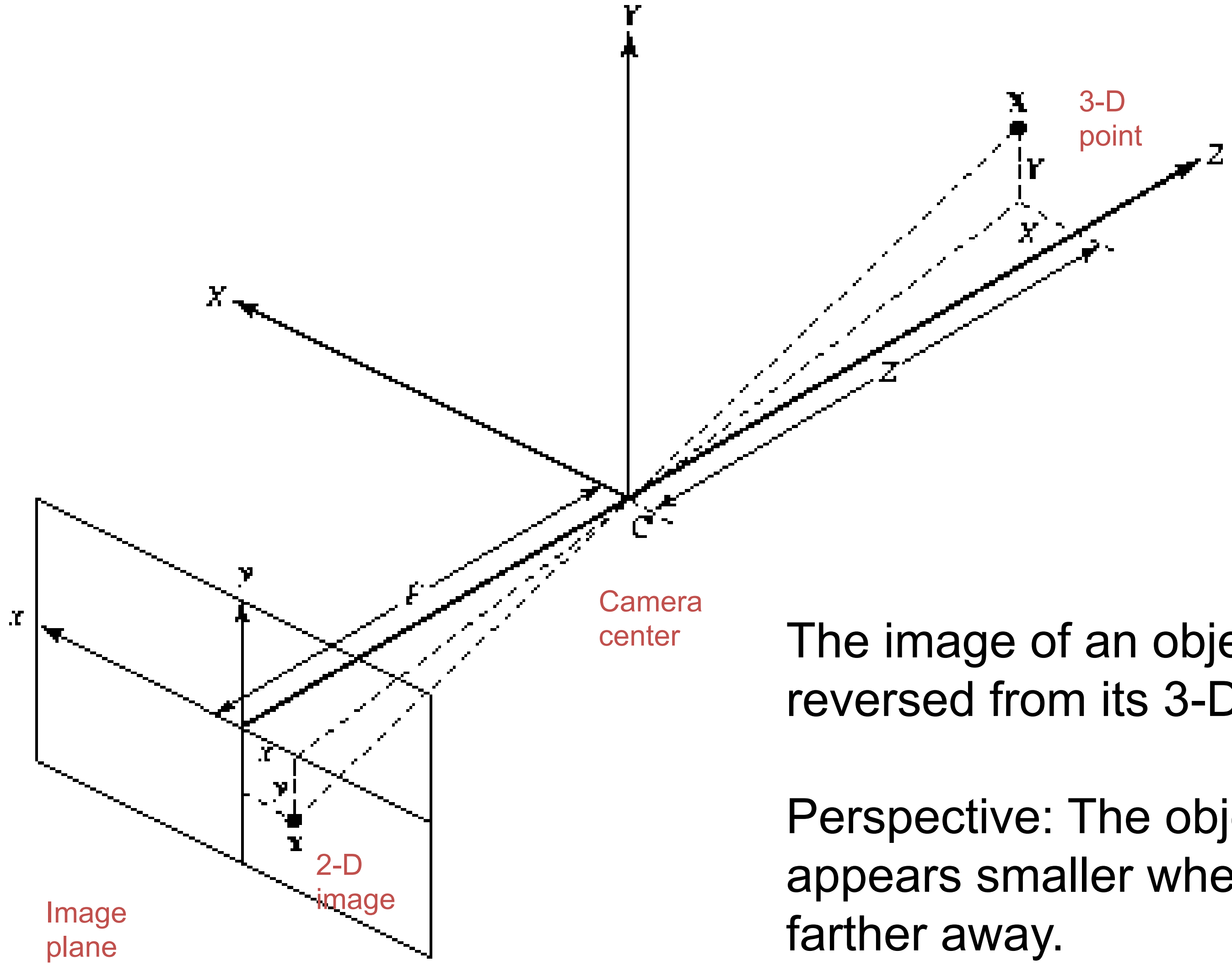
- Track (x), Dolly(z), Boom (y)

Rotational camera motions in 3D



- Pan (y), tilt (x), roll (z)
  - Stationary tripod mount

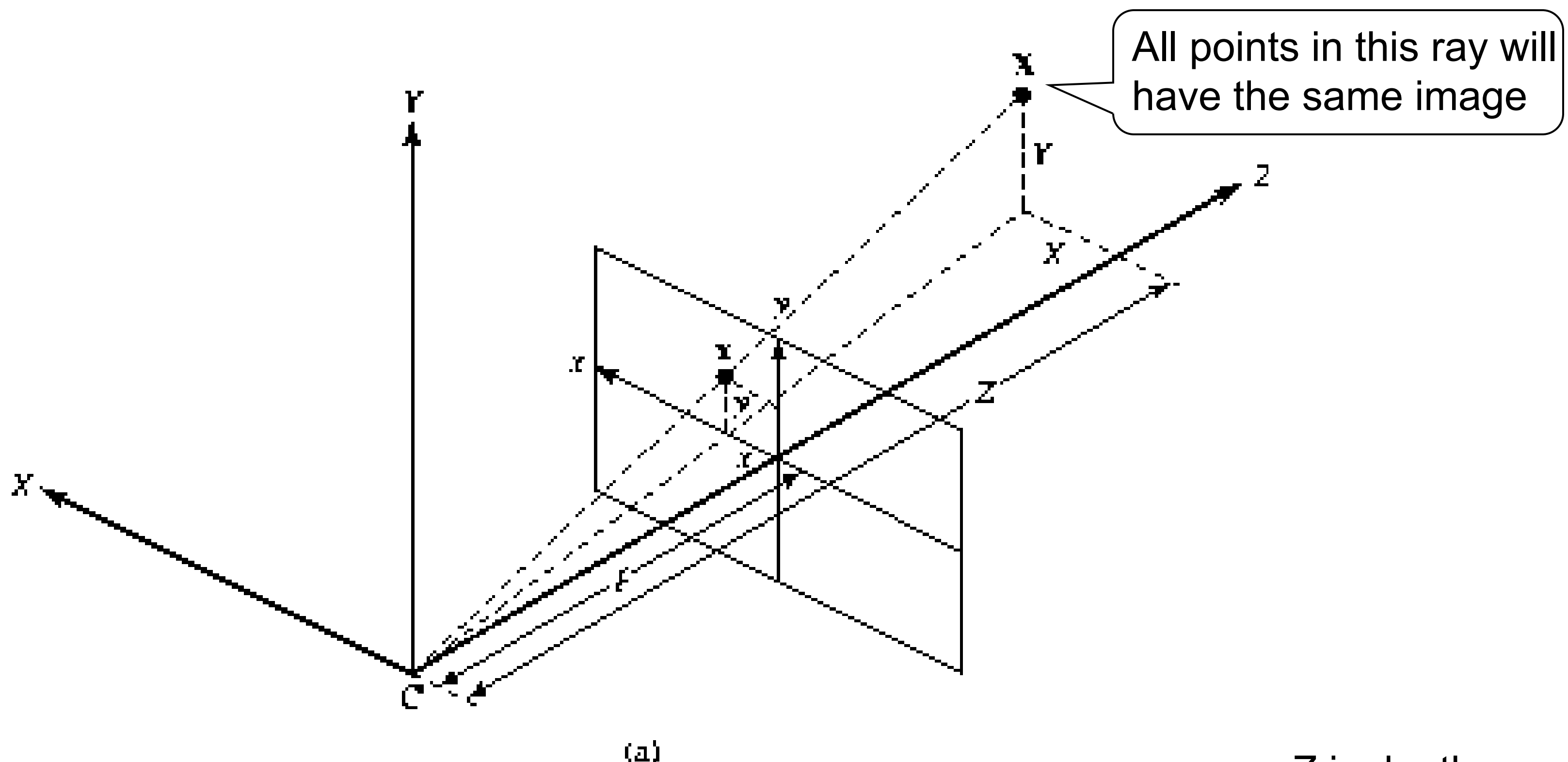
Pinhole Camera



The image of an object is reversed from its 3-D position.

Perspective: The object appears smaller when it is farther away.

Pinhole Camera Model: Perspective Projection



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

x, y are inversely related to Z

Z is depth; Distance from camera center to object

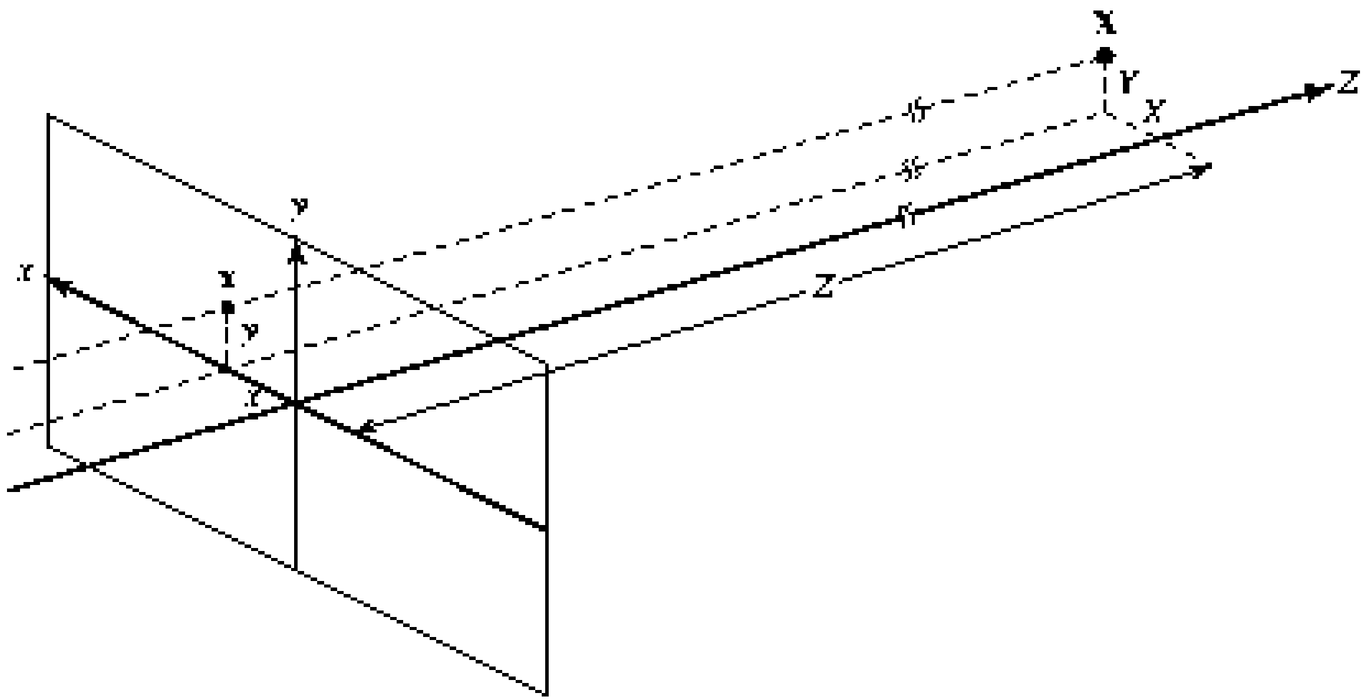
Attributes of perspective projection

- Straight lines in 3D space are projected into straight lines in 2D space
- Parallel lines in 3D may not be parallel in 2D
  - Vanishing points
- A nonlinear model, due to division by depth Z

Vanishing points

Vanishing points

Simpler Model: Orthographic Projection



When the object is very far ($Z \to \infty$)

$x = X, y = Y$

Can be used as long as the depth variation within the object is small compared to the distance of the object.

CAHV camera model

- Vector C: location of the pinhole
- Vector A: Camera axis (normal to the image plane)
- Vector H:
  - Horizontal axis of image plane
  - H-coordinate of optical center of image plane
  - Horizontal focal length (in pixels)
- Vector V: same vertically

A number of other camera models available, all more sophisticated than pinhole model

2D Motion

2D (or apparent) motion that is created by moving objects depends on 3 things:

1. An image formation (or camera) model
   – Perspective, orthographic, ..
2. Motion model of a 3D object (rigid body with 3D translation and rotation, 3D affine motion)
3. Surface model of 3D object (planar, parabolic..)

Motion models for 3D objects

- 3D motion
- Projection of 3-D motion on 2-D image plane
- 2D motion caused by rigid object motion
  - Projective mapping
- Approximations to the 2D motion field
  - Affine model
  - Bilinear model

Rigid object 3-D motion

- Translation vector T, defines how object translates in x,y,z; $T=[T_x,T_y,T_z]'$
- Rotation matrix R
  - Defines how object rotates, first in X, then in Y, then in Z
  - Defined by rotation angles:

$$\theta_x, \theta_y, \theta_z$$

- 6 parameters, valid for all points on object

Rotation matricies

- R = $R_z R_y R_x$
- Order matters!
- Orthonormal ($R^T = R^{-1}$) and det(R)=+/-1

$$R_x = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta_x & -\sin\theta_x \\ 0 & \sin\theta_x & \cos\theta_x \end{bmatrix}$$

$$R_y = \begin{bmatrix} \cos\theta_y & 0 & \sin\theta_y \\ 0 & 1 & 0 \\ -\sin\theta_y & 0 & \cos\theta_y \end{bmatrix}$$

$$R_z = \begin{bmatrix} \cos\theta_z & -\sin\theta_z & 0 \\ \sin\theta_z & \cos\theta_z & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Linearization approximation

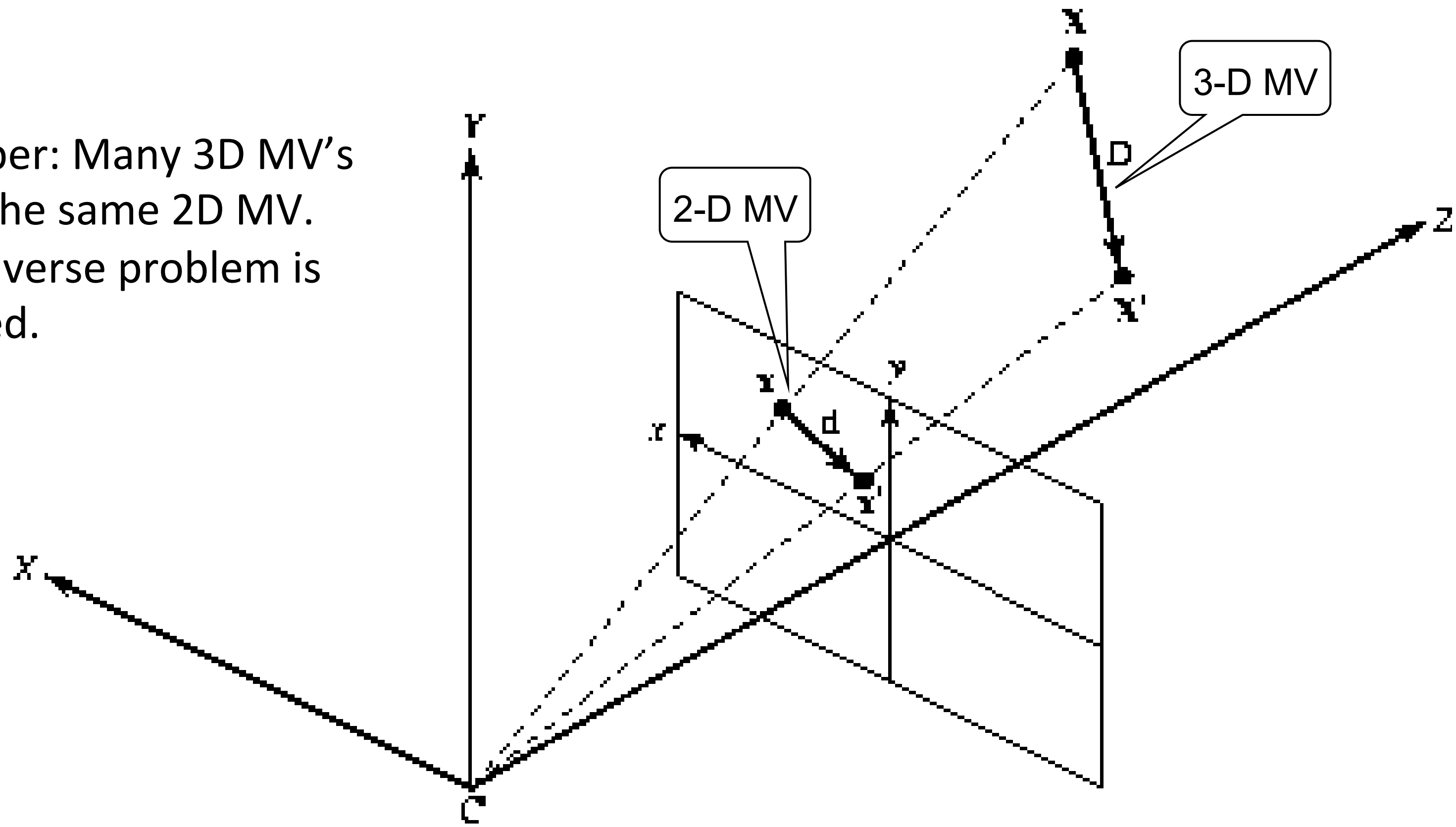
- Can use a small angle approximation
- $\cos\alpha \approx 1$; $\sin\alpha \approx \alpha$

$$R = \begin{bmatrix} 1 & -\theta_z & \theta_y \\ \theta_z & 1 & -\theta_x \\ -\theta_y & \theta_x & 1 \end{bmatrix}$$

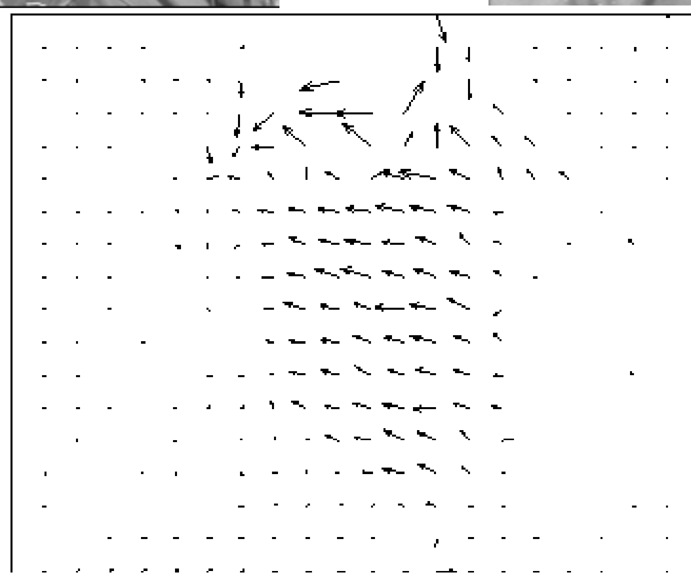
- Can define motion about the object center if desired

3-D Motion → 2-D Motion

Remember: Many 3D MV's map to the same 2D MV. So the inverse problem is ill-defined.



Sample 2-D Motion Field

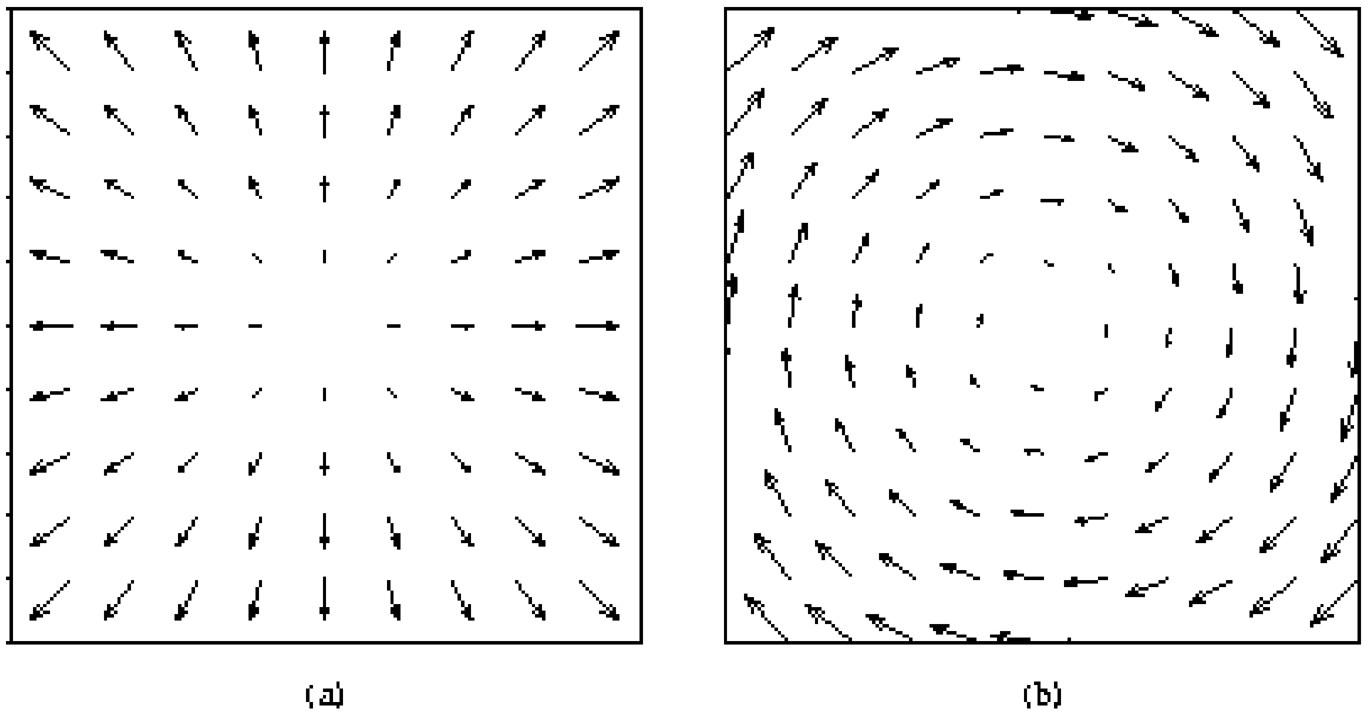


A "needle" plot helps to visualize motion, by describing where to go in first frame to get the "matching" pixel values for the second frame

2D Motion models

- How is the 2D motion represented in 2D
  - Each pixel can move on its own, but it's often more effective to describe how a region of pixels move – because remember, 2D motion is caused either by **camera** motion (which moves all pixels) or by **object** motion (which moves all the pixels on an object)
- So next, we look at how camera motion appears in 2D
  - Take a point (X,Y,Z) in 3D, map it into its new coordinates (X',Y',Z') based on camera motion, the apply perspective mapping to find relationship between (x,y) and (x',y')

2-D Motion Corresponding to **Camera** Motion



(a) Camera zoom

(b) Camera rotation around Z-axis (roll)

Also Pan and tilt (with small angle approximations)

Math: Camera Track and Boom

- Translations T_x and T_y
- $X' = X + T_x$
- $Y' = Y + T_y$
- $Z' = Z$

- $x'=x+FT_x/Z$ $\quad$ $d_x=FT_x/Z \approx t_x$ for Z large-ish
- $y'=y+FT_y/Z$ $\quad$ $d_y=FT_y/Z \approx t_y$ for Z large-ish

Math: Pan and Tilt

- Rotation angles θ_y and θ_x
- $X' = R_x R_y X$
- R_x and R_y are as given earlier

- $x'-x = d_x(x,y) = \theta_y F$
- $y'-y = d_y(x,y) = -\theta_y F$
  when $Y\theta_x << Z$ and $X\theta_y << Z$ so that $Z' \approx Z$

Math: Zoom

- F and F’ are focal length before and after
- $x' = \rho x$
- $y' = \rho y$
- $d_x(x,y) = (1-\rho)x$
- $d_y(x,y) = (1-\rho)y$

Math: Roll

- Rotation about Z axis; no change in depth
- $x' = x \cos \theta_z - y \sin \theta_z \approx x - y\ \theta_z$
- $y' = x \sin \theta_z + y \cos \theta_z \approx x\ \theta_z - y$

- $d_x(x,y) = -y\theta_z$
- $d_y(x,y) = x\theta_z$

Combining camera motions

- Translation, pan, tilt, zoom, and rotation, with small-angle approximations
- The result: A special case of affine mapping

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \rho \begin{bmatrix} \cos\theta_z & -\sin\theta_z \\ \sin\theta_z & \cos\theta_z \end{bmatrix} \begin{bmatrix} x + \theta_y F + t_x \\ y - \theta_x F + t_y \end{bmatrix}$$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} c_1 & -c_2 \\ c_2 & c_1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} c_3 \\ c_4 \end{bmatrix}$$

2-D Motion Corresponding to Rigid Object Motion

- General case:

$$\begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix} = \begin{bmatrix} r_1 & r_2 & r_3 \\ r_4 & r_5 & r_6 \\ r_7 & r_8 & r_9 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix}$$

Assume camera stationary. Object undergoes rigid motion.

$\xrightarrow{\text{Perspective Projection}}$

$$x' = F\frac{(r_1x + r_2y + r_3F)Z + T_xF}{(r_7x + r_8y + r_9F)Z + T_zF}$$

$$y' = F\frac{(r_4x + r_5y + r_6F)Z + T_yF}{(r_7x + r_8y + r_9F)Z + T_zF}$$

- Projective mapping:

(8 parameters)

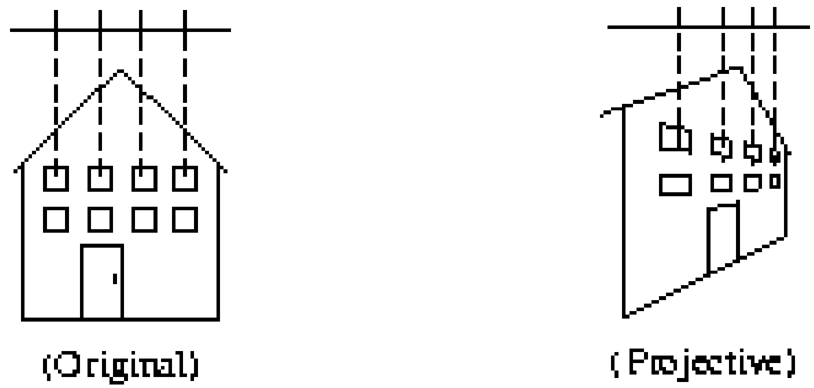
When the object surface is planar $(Z = aX + bY + c)$:

$$x' = \frac{a_0 + a_1x + a_2y}{1 + c_1x + c_2y}, \quad y' = \frac{b_0 + b_1x + b_2y}{1 + c_1x + c_2y}$$

More 2D motion models

- Projective mapping (8 parameters)
  - (3D rigid motion of planar surface using perspective projection)
- Affine
  - (3D rigid motion of planar surface using orthographic projection)
- Bilinear
- Translational

Perspective imaging



Two features of projective mapping:

- Chirping: increasing perceived spatial frequency for far away objects
- Converging (Keystone): parallel lines converge in distance

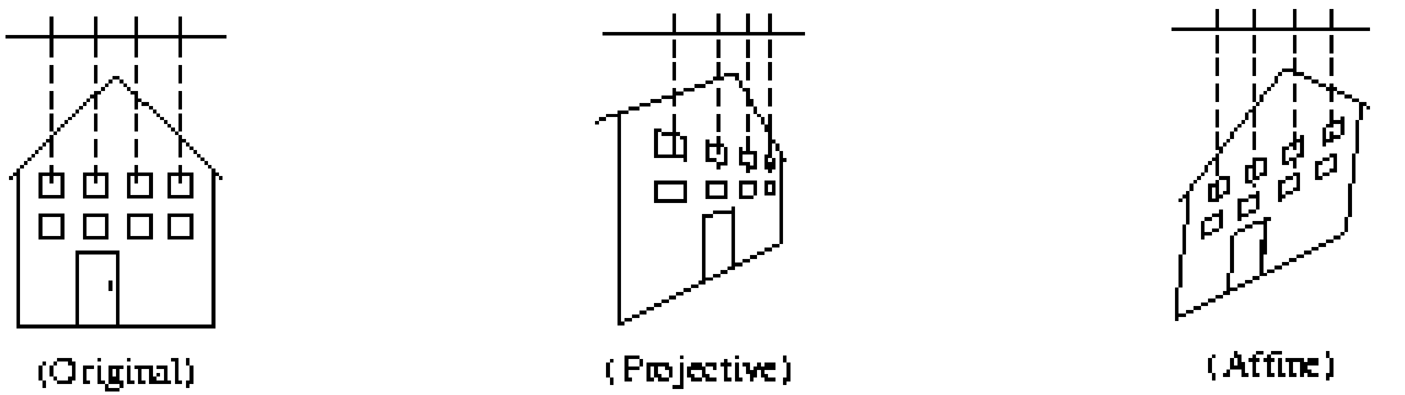
Affine Model

- Polynomial approximation to projective mapping
- 6 parameters

$$\begin{bmatrix} d_x(x,y) \\ d_y(x,y) \end{bmatrix} = \begin{bmatrix} a_0 + a_1 x + a_2 y \\ b_0 + b_1 x + b_2 y \end{bmatrix}$$

- Maps triangles to triangles: defined by motion vectors of the three corners

Affine model



Affine model:

- No chirping, no converging of parallel lines

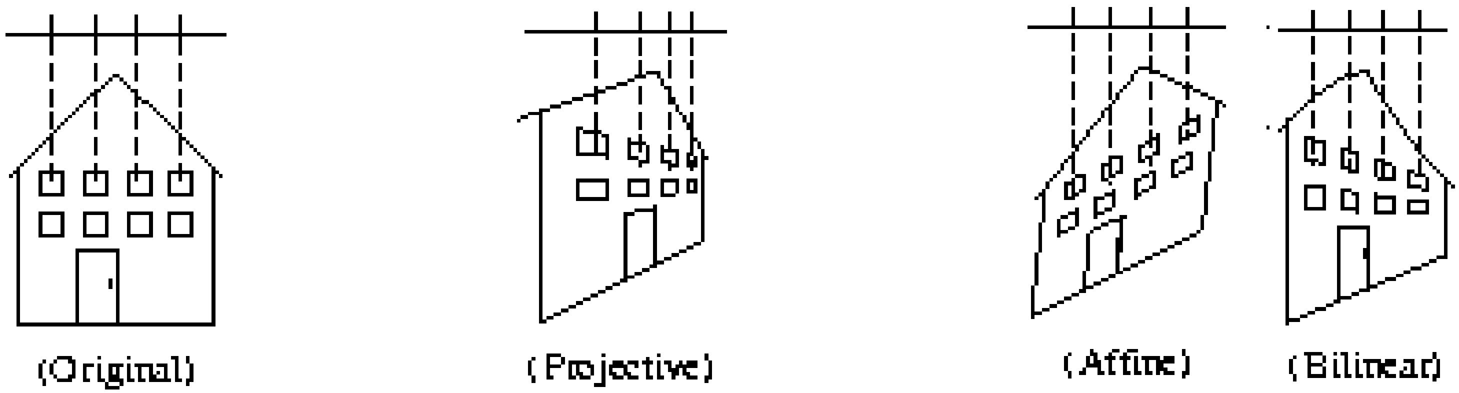
Bilinear Model

- Another approximation to projective mapping
- 8 parameters

$$\begin{bmatrix} d_x(x,y) \\ d_y(x,y) \end{bmatrix} = \begin{bmatrix} a_0 + a_1 x + a_2 y + a_3 xy \\ b_0 + b_1 x + b_2 y + b_3 xy \end{bmatrix}$$

- Maps a square into a quadrilateral
- CanNOT map between 2 arbitrary quadrilaterals

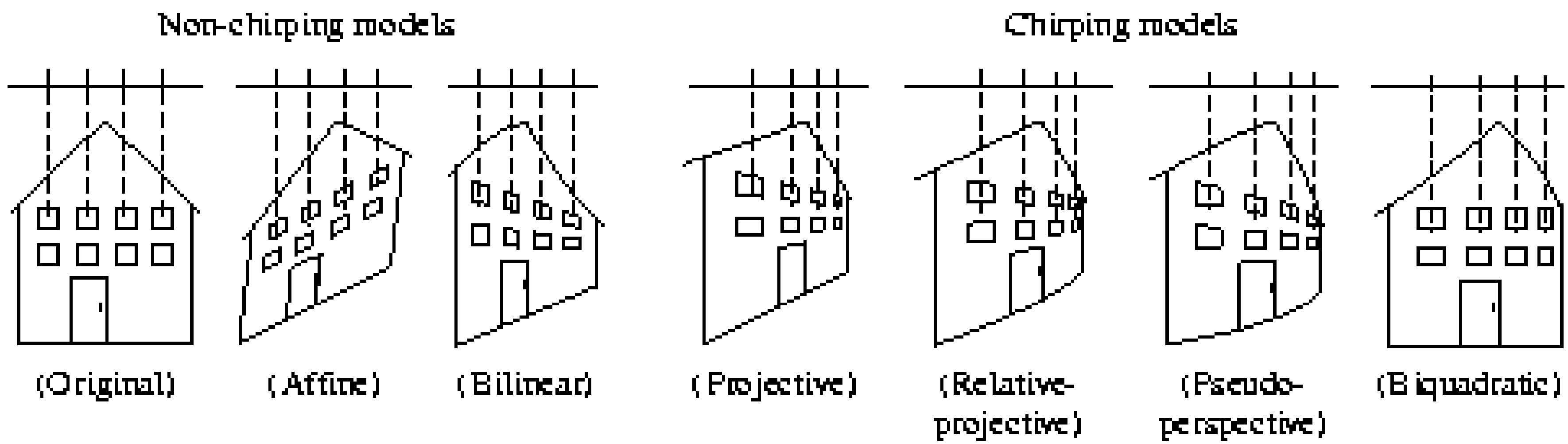
Bilinear model



Bilinear model:

- No chirping, but convergence of parallel lines

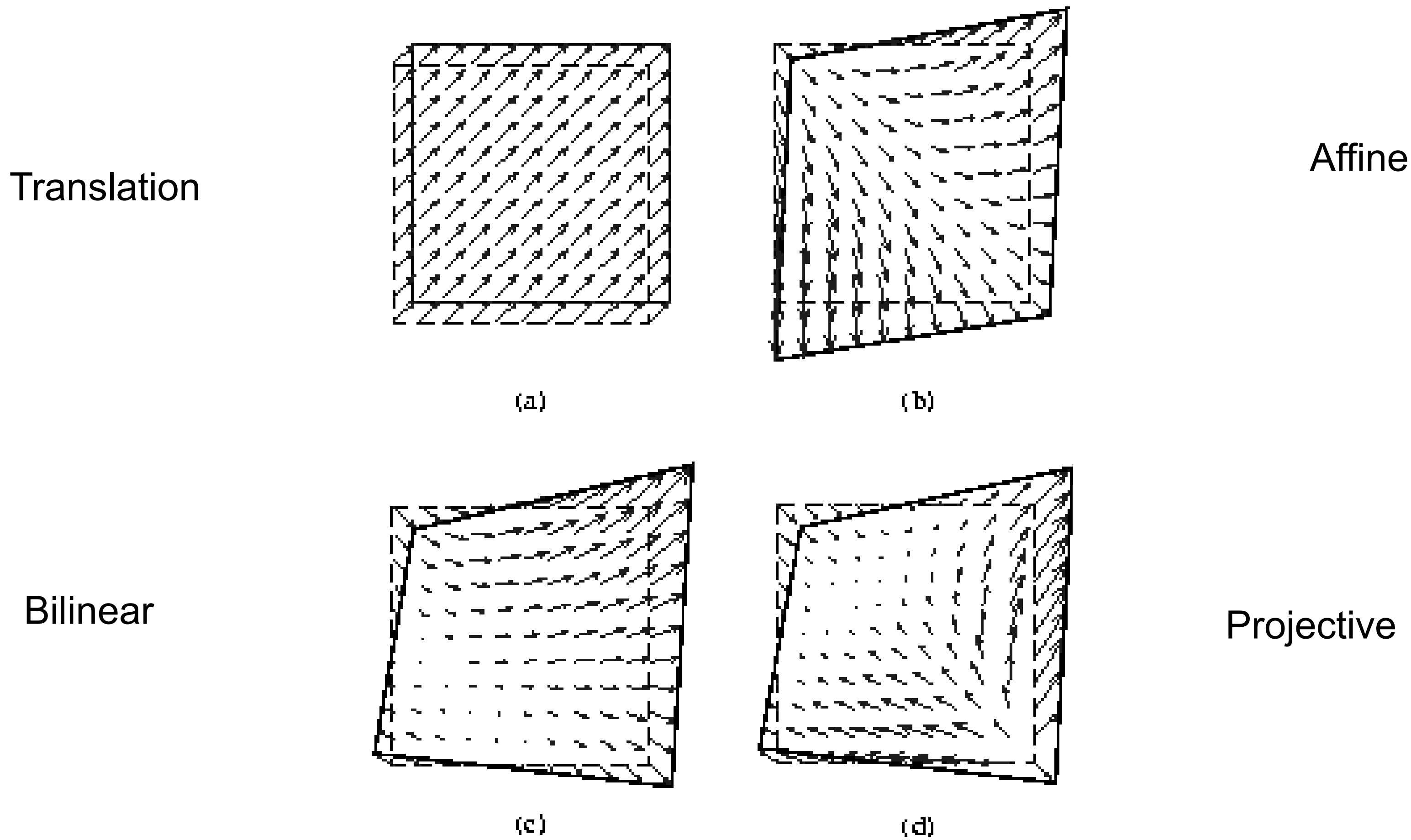
Other models exist



Translation only

$$\begin{bmatrix} d_x(x,y) \\ d_y(x,y) \end{bmatrix} = \begin{bmatrix} a_0 \\ b_0 \end{bmatrix}$$

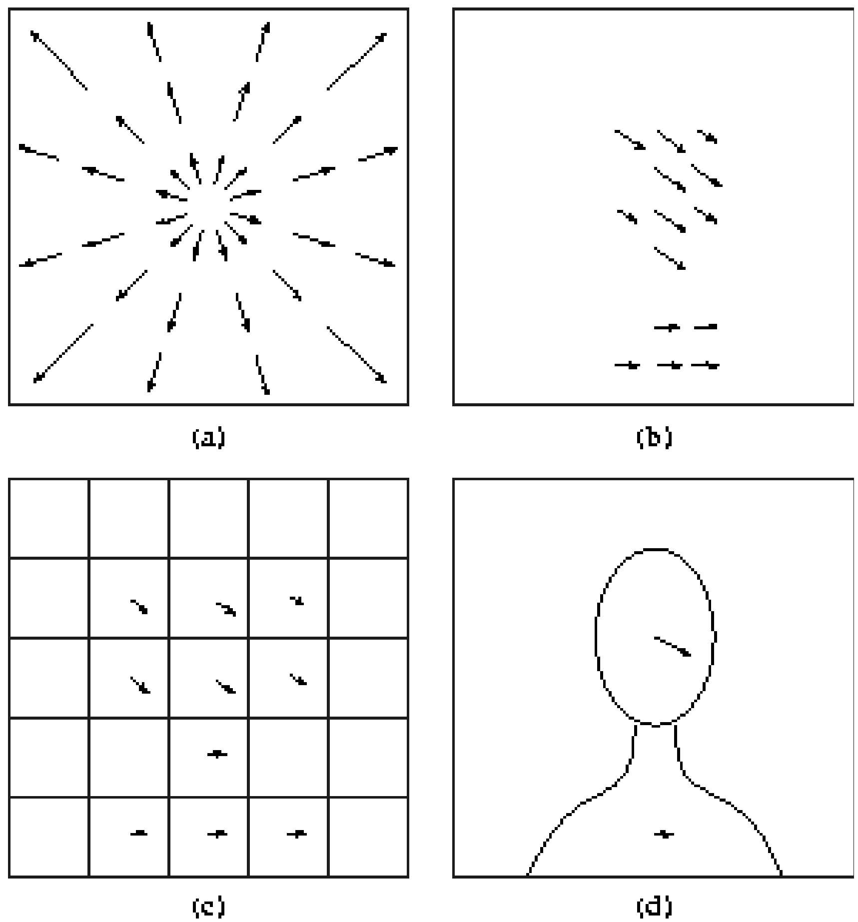
Motion Field Corresponding to Different 2-D Motion Models



Question: is it likely to have an entire image be composed of a planar object moving in a rigid fashion?

Region of support for representation of motion

Global:
Entire motion field is represented by a few global parameters



Pixel-based:
One MV at each pixel, with some smoothness constraint between adjacent MVs.

Block-based:
Entire frame is divided into blocks, and motion in each block is characterized by a few parameters.

Region-based:
Entire frame is divided into regions, each region corresponding to an object or sub-object with consistent motion, represented by a few parameters.