Robotics: Principles and Practice

Module 5: Robot Vision

Lecture 9: Inverse perspective transformation

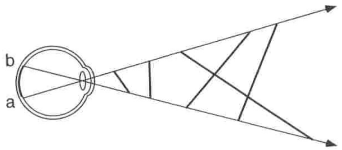
David Vernon
Carnegie Mellon University Africa

www.vernon.eu

Depth Perception: The Inverse Problem

Estimate depth (3D) of world from images

Inverse perspective transformation



Credit: Markus Vincze, Technische Universität Wien

The Inverse Perspective Transformation

- Once the camera model C has been determined, we are in a position to determine an expression for the co-ordinates of a point in the real world in terms of the co-ordinates of its imaged position

- Recalling equations (1) - (5) :

$$
\begin{aligned}
C_{11}x + C_{12}y + C_{13}z + C_{14} &= u = Ut \\
C_{21}x + C_{22}x + C_{23}x + C_{24} &= v = Vt \\
C_{31}x + C_{32}x + C_{33}x + C_{34} &= t
\end{aligned}
$$

The Inverse Perspective Transformation

Substituting the expression for t into the first two equations gives

$$U\left(C_{31}x + C_{32}y + C_{33}z + C_{34}C_{11}x\right) = C_{11}x + C_{12}y + C_{13}z + C_{14}$$
$$V\left(C_{31}x + C_{32}y + C_{33}z + C_{34}C_{11}x\right) = C_{21}x + C_{22}y + C_{23}z + C_{24}$$

Hence

$$\left(C_{11} - UC_{31}\right)x + \left(C_{12} - UC_{32}\right)y + \left(C_{13} - UC_{33}\right)z + \left(C_{14} - UC_{34}\right) = 0$$
$$\left(C_{21} - VC_{31}\right)x + \left(C_{22} - VC_{32}\right)y + \left(C_{23} - VC_{33}\right)z + \left(C_{24} - VC_{34}\right) = 0$$

The Inverse Perspective Transformation

Letting

$$a_1 = C_{11} - UC_{31}$$
$$b_1 = C_{12} - UC_{32}$$
$$c_1 = C_{13} - UC_{33}$$
$$d_1 = C_{14} - UC_{34}$$

and

$$a_2 = C_{21} - VC_{31}$$
$$b_2 = C_{22} - VC_{32}$$
$$c_2 = C_{23} - VC_{33}$$
$$d_2 = C_{24} - VC_{34}$$

we have

$$a_1 x + b_1 y + c_1 z + d_1 = 0$$
$$a_2 x + b_2 y + c_2 z + d_2 = 0$$

The Inverse Perspective Transformation

These are the equations of two planes

The intersection of these planes determines a line comprising the set of real-world points which project onto the image point $\begin{bmatrix} U \\ V \end{bmatrix}$

Solving these plane equations simultaneously (in terms of z)

$$x = \frac{z\left(b_1c_2 - b_2c_1\right) + \left(b_1d_2 - b_2d_1\right)}{\left(a_1b_2 - a_2b_1\right)}$$

$$y = \frac{z\left(a_2c_1 - a_1c_2\right) + \left(a_2d_1 - a_1d_2\right)}{\left(a_1b_2 - a_2b_1\right)}$$

The Inverse Perspective Transformation

Thus, for any given z_0, U and V, we may determine the corresponding x_0 and y_0, i.e. the real-world co-ordinates

The camera model and the inverse perspective transformation which we have just discussed allow us to compute the x and y real-world co-ordinates corresponding to a given position in the image

However, we must assume that the z coordinate, *i.e.* the distance from the camera, is known

The Inverse Perspective Transformation

- For some applications, e.g. where objects lie on a table at a given and constant height (*i.e.* at a given z_0), this is sufficient

- In general, however, we will not know the coordinate of the object in the third dimension and we must recover it

The Inverse Perspective Transformation

How we can compute z_0 ?

- If we have a second image of the scene, taken from another viewpoint
- If we know the image coordinates of the point of interest in this image
- Then we have two camera models and, hence, two inverse perspective transformations
- Instead of solving two plane equations simultaneously, we solve four plane equations

The Inverse Perspective Transformation

In particular, we have

$$a_1 x + b_1 y + c_1 z + d_1 = 0$$
$$a_2 x + b_2 y + c_2 z + d_2 = 0$$
$$p_1 x + q_1 y + r_1 z + s_1 = 0$$
$$p_2 x + q_2 y + r_2 z + s_2 = 0$$

where

$$a_1 = C1_{11} - U1C1_{31} \qquad a_2 = C1_{21} - V1C1_{31}$$
$$b_1 = C1_{12} - U1C1_{32} \qquad b_2 = C1_{22} - V1C1_{32}$$
$$c_1 = C1_{13} - U1C1_{33} \qquad c_2 = C1_{23} - V1C1_{33}$$
$$d_1 = C1_{14} - U1C1_{34} \qquad d_2 = C1_{24} - V1C1_{34}$$
$$p_1 = C2_{11} - U2C2_{31} \qquad p_2 = C2_{21} - V2C2_{31}$$
$$q_1 = C2_{12} - U2C2_{32} \qquad q_2 = C2_{22} - V2C2_{32}$$
$$r_1 = C2_{13} - U2C2_{33} \qquad r_2 = C2_{23} - V2C2_{33}$$
$$s_1 = C2_{14} - U2C2_{34} \qquad s_2 = C2_{24} - V2C2_{34}$$

$C1_{ij}$ and $C2_{ij}$ are the coefficients of the camera model for the first and second images, respectively.
$U1, V1$ and $U2, V2$ are the co-ordinates of the point of interest in the first and second images, respectively.

The Inverse Perspective Transformation

Since we now have four equations and three unknowns,
the system is over-determined

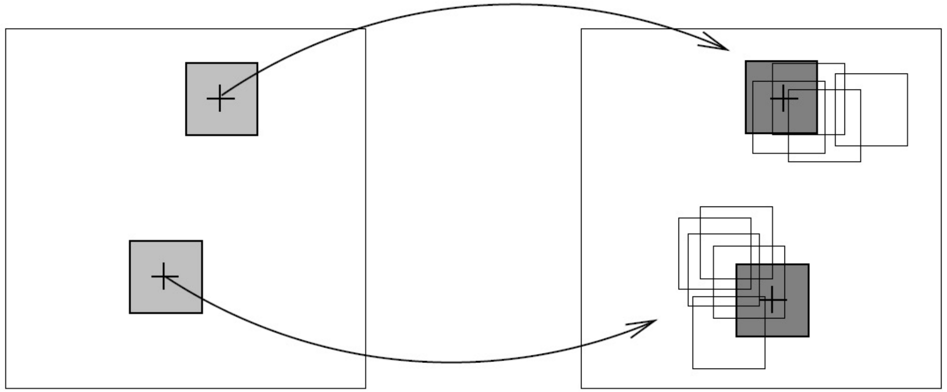
So, we compute a least-square-error solution using the pseudo-inverse technique

The Inverse Perspective Transformation

- It should be noted that the key here is not so much the mathematics which allows us to compute x_0 , y_0 and z_0 but, rather, the image analysis by which we identify the corresponding point of interest in the two images

- It is this correspondence problem which lies at the heart of most of the difficulties in recovery of depth information

The Inverse Perspective Transformation



[Young]

Credit: Markus Vincze, Technische Universität Wien

Reading

R. Szeliski, *Computer Vision: Algorithms and Applications*, Springer, 2010.

Section 2.1.5 3D to 2D projections

Section 6.3 Geometric intrinsic calibration

Vernon, D. 1991. Machine Vision: Automated Visual Inspection and Robot Vision, Prentice-Hall International; Section 8.6

OpenCV documentation on camera calibration:

http://docs.opencv.org/2.4/modules/calib3d/doc/camera_calibration_and_3d_reconstruction.html

Demo

Read "Camera Modelling and Camera Calibration.pdf"
Then walk through the following example applications:

cameraInvPerspectiveMonocular

cameraInvPerspectiveBinocular