Perspective Matching Example

Perspective Matching is the process of determining the position of a camera when it took a photograph so that 3D graphics can be combined with that photograph. To combine 3D graphics with a photograph you have to tell the 3D rendering program where to place the camera, how to orient the camera, and the specification for the camera's internals so during rendering the 'perspective matches'.

When PhotoModeler is solving a project to create a 3D model, it computes the exact position and orientation of the camera for each photograph. It also knows the accurate specifications for the camera's internal parameters (focal length, field of view, etc). This information can be exported from PhotoModeler for use in rendering and animation programs (such as 3D Studio Max, Truespace, Wavefront, Real3D, etc.).

This example starts off with an image of a cardboard box of known dimension but we know nothing about the camera that took the photo - we don't know what type of camera, what its focal length is, or where it was in space when the photo was taken. PhotoModeler's Inverse Camera feature and control points are used to solve the camera parameters and orientation. That information is then given to the Real3D animation/rendering program along with the 3D box model to perform a perspective match and rendering. Note that control points and inverse camera are used in this example because we have a single frame from an unknown camera but perspective matching can be used in other circumstances without control points also. Control points are known x,y,z locations.

The steps of this example are shown on the next page:

1) Start with an unknown photo of a box and import into PhotoModeler.

2) Load the known control points (box corners) and mark them on the photo in PhotoModeler. Solve the project and then export the camera station information (shown here in Notepad).

3) In the rendering program (Real3D in this case) load the box model from PhotoModeler and then set up the camera given information from PhotoModeler.

4) Add graphical elements to the box scene.
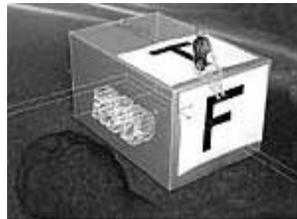
5) Render the result using the original photograph as a background. See how the perspectives match and how the shadows of the objects fall correctly on the box.

Real-3D images (steps 3-5) Copyright © 1997 Wolf-Dieter Grabner