Homography from Polygon in $R^3$ to Image plane

Li Zhang      Steve Seitz

1 A 2D Coordinate System for a Plane in 3D Space

![Diagram of a 2D coordinate system for a plane in 3D space](image)

Figure 1: plane coordinate system from 3 points in $R^3$

Let $p, q, r$ be 3 points in $R^3$. If they are not colinear, they define a unique plane. We want to set up a 2D coordinate system in the plane such that each point $a$ in the plane has a two dimensional coordinate $(u, v)$.

Let $r$ be the origin of the coordinate system and

$$e_x = \frac{p - r}{\|p - r\|}$$

be the base vector for $x$ axis in the plane. We then decompose vector $q - r$ into two components, one is parallel to $e_x$ and one is orthogonal to $e_x$. The parallel component is

$$s = <q - r, e_x> e_x$$

and the orthogonal component is

$$t = (q - r) - s$$

where $<\cdot, \cdot>$ is the dot product of two vectors. The base vector for $y$ axis in the plane is

$$e_y = \frac{t}{\|t\|}$$
For any point \( \mathbf{a} \) in the plane, its two dimensional coordinate in the plane with respect to \( \mathbf{e}_x \) and \( \mathbf{e}_y \) is

\[
(\langle \mathbf{a} - \mathbf{r}, \mathbf{e}_x \rangle, \langle \mathbf{a} - \mathbf{r}, \mathbf{e}_y \rangle)
\]

2 The Homography from a Polygon to its Image

Let \( \{\mathbf{p}_1 = (X_1, Y_1, Z_1), \mathbf{p}_2 = (X_2, Y_2, Z_2), \ldots\} \) be the vertices of a polygon in 3D space. Assuming \( \mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3 \) are not colinear, e.g. define a unique plane, they will introduce a 2D coordinate system in the plane. Using the method described in Section 1, each \( \mathbf{p}_i \) can have a 2D coordinate \((u_i, v_i)\) in the plane.

Let

\[
u_{\min} = \min\{u_i\}, u_{\max} = \max\{u_i\}, v_{\min} = \min\{v_i\}, v_{\max} = \max\{v_i\}\]

and we normalize \((u_i, v_i)\) by

\[
\tilde{u}_i = \frac{u_i - u_{\min}}{u_{\max} - u_{\min}}, \quad \tilde{v}_i = \frac{v_i - v_{\min}}{v_{\max} - v_{\min}}
\]

Now each \((\tilde{u}_i, \tilde{v}_i)\) is between \([0, 1]\) and can be used as texture coordinates.

Suppose the image coordinate of \( \mathbf{p}_i \) is \((x_i, y_i)\), we estimate the homography \( \mathbf{H} \) which maps each \((\tilde{u}_i, \tilde{v}_i)\) to \((x_i, y_i)\).

(Note: this is the \( H \) in the skeleton code in SVMPolygon structure. \((\tilde{u}_i, \tilde{v}_i)\) is used as texture coordinate for point \( \mathbf{p}_i \).)

The estimation algorithm is covered in Steve’s lecture. That is, \( \mathbf{h} \) is the eigenvector of the 9 by 9 semi-positive-definite matrix, whose eigenvalue is the smallest. The code for eigen-decomposition is provided in jacob.h/cpp. The main.cpp file associated with them show how to use the function jacob(\(\cdots\)). These files are downloaded seperately from our skeleton code.

2.1 More Accurate Estimation

As we do for solving vanishing point, we recommend you normalize \((x_i, y_i)\) first before estimating the homograph \( \mathbf{h} \). That is, we first compute

\[
x_{\min} = \min\{x_i\}, x_{\max} = \max\{x_i\}, y_{\min} = \min\{y_i\}, y_{\max} = \max\{y_i\}\]

and normalize \((x_i, y_i)\) as

\[
\hat{x}_i = \frac{x_i - x_{\min}}{x_{\max} - x_{\min}}, \quad \hat{y}_i = \frac{y_i - y_{\min}}{y_{\max} - y_{\min}}
\]

The normalization can be written as

\[
\begin{pmatrix}
\hat{x}_i \\
\hat{y}_i \\
1
\end{pmatrix} = \begin{pmatrix}
x_i \\
y_i \\
1
\end{pmatrix}
\]
where \( S = \begin{bmatrix}
\frac{1}{x_{\max} - x_{\min}} & 0 & \frac{-x_{\min}}{x_{\max} - x_{\min}} \\
0 & 1 & 0 \\
0 & 0 & 1 
\end{bmatrix} \).

You want to first estimate a homograph \( H_n \) from \((\hat{u}_i, \hat{v}_i)\) to \((\hat{x}_i, \hat{y}_i)\) and then compute \( H = S^{-1} H_n \). Some theoretical analysis proves that \( H_n \) can be more accurately estimated, which is beyond the scope of the class.