

Orthogonal projection

MAK

25.7.2014

1 Introduction

An orthogonal projection is a mathematical function that projects a vector $\vec{v} \in \mathbb{R}$ onto a (hyper-)plane X . So it maps the given vector onto that one, that has the shortest distance to it and lies in X .

This article deals with the corresponding projection matrix (for embedding in CE3D).

2 The orthogonal projection

An orthogonal projection has the following characteristics.

- The projected vector is a linear combination of the base vectors $\vec{v}_1, \vec{v}_2, \dots, \vec{v}_m$ of the projection space X .
- The projected vector minus the original vector to project shall be perpendicular to the projection space ($\langle p(\vec{x}) - \vec{x}, \vec{v} \rangle = 0$).

A list follows with definitions for this article:

- The superior vector room where projection happens: V . The dimension of this room is defined as $\dim(V) = n$.
- The projection space where is projected into: $X = \text{span}(\vec{v}_1, \vec{v}_2, \dots, \vec{v}_m)$. The dimension is defined as $\dim(X) = m$.
- The projection function: $p(\vec{x})$. $p = \begin{cases} \mathbb{R}^n \rightarrow \mathbb{R}^n \\ \vec{x} \mapsto p(\vec{x}) \end{cases}$

3 The projection matrix

To get the projection matrix we use the attributes from above und give them an generalized mathematical form:

$$\begin{array}{ll} \text{I} & p(\vec{x}) = \sum_{i=1}^m \mu_i \vec{v}_i, \quad \mu_i \in \mathbb{R} \\ \text{II} & \langle p(\vec{x}) - \vec{x}, \vec{v}_i \rangle = 0, \quad \forall i \in \{w \in \mathbb{N} | 1 \leq w \leq m\} \end{array} \quad (1)$$

The first constraint can be inserted into the second one and rewritten into a linear equation system:

$$\left\langle \sum_{j=1}^m \mu_j \vec{v}_j - \vec{x}, \vec{v}_i \right\rangle = 0, \quad \forall i \in \{w \in \mathbb{N} | 1 \leq w \leq m\} \quad (2)$$

$$\begin{aligned} \Leftrightarrow \quad & \begin{aligned} \langle \mu_1 \vec{v}_1 + \mu_2 \vec{v}_2 + \dots + \mu_m \vec{v}_m - \vec{x}, \vec{v}_1 \rangle &= 0 \\ \langle \mu_1 \vec{v}_1 + \mu_2 \vec{v}_2 + \dots + \mu_m \vec{v}_m - \vec{x}, \vec{v}_2 \rangle &= 0 \\ &\vdots \\ \langle \mu_1 \vec{v}_1 + \mu_2 \vec{v}_2 + \dots + \mu_m \vec{v}_m - \vec{x}, \vec{v}_m \rangle &= 0 \end{aligned} \end{aligned} \quad (3)$$

Move all constant factors to the right side of the equation:

$$\begin{aligned} \Leftrightarrow \quad & \begin{aligned} \langle \mu_1 \vec{v}_1 + \mu_2 \vec{v}_2 + \dots + \mu_m \vec{v}_m, \vec{v}_1 \rangle &= \langle \vec{x}, \vec{v}_1 \rangle \\ \langle \mu_1 \vec{v}_1 + \mu_2 \vec{v}_2 + \dots + \mu_m \vec{v}_m, \vec{v}_2 \rangle &= \langle \vec{x}, \vec{v}_2 \rangle \\ &\vdots \\ \langle \mu_1 \vec{v}_1 + \mu_2 \vec{v}_2 + \dots + \mu_m \vec{v}_m, \vec{v}_m \rangle &= \langle \vec{x}, \vec{v}_m \rangle \end{aligned} \end{aligned} \quad (4)$$

Use the distributivity and homogeneity:

$$\begin{aligned} \Leftrightarrow \quad & \begin{aligned} \mu_1 \langle \vec{v}_1, \vec{v}_1 \rangle + \mu_2 \langle \vec{v}_2, \vec{v}_1 \rangle + \dots + \mu_m \langle \vec{v}_m, \vec{v}_1 \rangle &= \langle \vec{x}, \vec{v}_1 \rangle \\ \mu_1 \langle \vec{v}_1, \vec{v}_2 \rangle + \mu_2 \langle \vec{v}_2, \vec{v}_2 \rangle + \dots + \mu_m \langle \vec{v}_m, \vec{v}_2 \rangle &= \langle \vec{x}, \vec{v}_2 \rangle \\ &\vdots \\ \mu_1 \langle \vec{v}_1, \vec{v}_m \rangle + \mu_2 \langle \vec{v}_2, \vec{v}_m \rangle + \dots + \mu_m \langle \vec{v}_m, \vec{v}_m \rangle &= \langle \vec{x}, \vec{v}_m \rangle \end{aligned} \end{aligned} \quad (5)$$

Represent the linear equation system with a matrix:

$$\Leftrightarrow \begin{pmatrix} \langle \vec{v}_1, \vec{v}_1 \rangle & \langle \vec{v}_2, \vec{v}_1 \rangle & \dots & \langle \vec{v}_m, \vec{v}_1 \rangle \\ \langle \vec{v}_1, \vec{v}_2 \rangle & \langle \vec{v}_2, \vec{v}_2 \rangle & \dots & \langle \vec{v}_m, \vec{v}_2 \rangle \\ \vdots & \vdots & \ddots & \vdots \\ \langle \vec{v}_1, \vec{v}_m \rangle & \langle \vec{v}_2, \vec{v}_m \rangle & \dots & \langle \vec{v}_m, \vec{v}_m \rangle \end{pmatrix} \begin{pmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_m \end{pmatrix} = \begin{pmatrix} \langle \vec{v}_1, \vec{x} \rangle \\ \langle \vec{v}_2, \vec{x} \rangle \\ \vdots \\ \langle \vec{v}_m, \vec{x} \rangle \end{pmatrix} \quad (6)$$

And we get the Gram matrix. Now we must transform the equation so that we get a map in the form

$$P\vec{x} = p(\vec{x}) \quad (7)$$

with P as our corresponding projection matrix. Furthermore we define the Gram matrix and the coefficient vector shorter for simplification:

$$G := \begin{pmatrix} \langle \vec{v}_1, \vec{v}_1 \rangle & \langle \vec{v}_2, \vec{v}_1 \rangle & \dots & \langle \vec{v}_m, \vec{v}_1 \rangle \\ \langle \vec{v}_1, \vec{v}_2 \rangle & \langle \vec{v}_2, \vec{v}_2 \rangle & \dots & \langle \vec{v}_m, \vec{v}_2 \rangle \\ \vdots & \vdots & \ddots & \vdots \\ \langle \vec{v}_1, \vec{v}_m \rangle & \langle \vec{v}_2, \vec{v}_m \rangle & \dots & \langle \vec{v}_m, \vec{v}_m \rangle \end{pmatrix} \quad (8)$$

$$\vec{\mu} := \begin{pmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_m \end{pmatrix} \quad (9)$$

Our equation system is rearranged after $\vec{\mu}$ and inserted into the first attribute of the orthogonal projection (every projected vector shall be a linear combination of the base of the projection space X):

$$G\vec{\mu} = \begin{pmatrix} \langle \vec{v}_1, \vec{x} \rangle \\ \langle \vec{v}_2, \vec{x} \rangle \\ \vdots \\ \langle \vec{v}_m, \vec{x} \rangle \end{pmatrix} \iff G^{-1} \begin{pmatrix} \langle \vec{v}_1, \vec{x} \rangle \\ \langle \vec{v}_2, \vec{x} \rangle \\ \vdots \\ \langle \vec{v}_m, \vec{x} \rangle \end{pmatrix} = \vec{\mu} \quad (10)$$

Before inserting we transform the first constraint into a matrix form:

$$\sum_{i=1}^m \mu_i \vec{v}_i = p(\vec{x}) = \left\langle \vec{\mu}, \begin{pmatrix} \vec{v}_1 \\ \vec{v}_2 \\ \vdots \\ \vec{v}_m \end{pmatrix} \right\rangle = (\vec{v}_1 \quad \vec{v}_2 \quad \cdots \quad \vec{v}_m) \vec{\mu} \quad (11)$$

Again we define for simplification:

$$A := (\vec{v}_1 \quad \vec{v}_2 \quad \cdots \quad \vec{v}_m) \quad (12)$$

So A is our matrix that contains the base of our projection plane X .

All left to do is mathematical rearranging to get the desired projection matrix P :

$$p(\vec{x}) = A\vec{\mu} = AG^{-1} \begin{pmatrix} \langle \vec{v}_1, \vec{x} \rangle \\ \langle \vec{v}_2, \vec{x} \rangle \\ \vdots \\ \langle \vec{v}_m, \vec{x} \rangle \end{pmatrix} = AG^{-1} \begin{pmatrix} \vec{v}_1^T \vec{x} \\ \vec{v}_2^T \vec{x} \\ \vdots \\ \vec{v}_m^T \vec{x} \end{pmatrix} = AG^{-1} A^T \vec{x} \quad (13)$$

The Gram matrix is the same as

$$G = A^T A \quad (14)$$

At least we get our final projection matrix:

$$p(\vec{x}) = A(A^T A)^{-1} A^T \vec{x} \quad (15)$$

$$P = A(A^T A)^{-1} A^T \quad (16)$$

with our base matrix

$$A = (\vec{v}_1 \quad \vec{v}_2 \quad \cdots \quad \vec{v}_m) \quad (17)$$

4 Projection onto the coefficients $\vec{\mu}$ in projection space

It's useful for 3D applications to get the coefficients of the projection space base vectors. For example to use for a camera implementation that gives you the x- and y-coordinates of the image objects on the sensor respectively on the display.

The projection matrix for the coefficients can be fastly obtained with the first projection attribute (after the translation into the matrix form):

$$p(\vec{x}) = A\vec{\mu} = A(A^T A)^{-1} A^T \vec{x} \quad (18)$$

$$\iff \vec{\mu} = (A^T A)^{-1} A^T \vec{x} \quad (19)$$

So the projection matrix for the coefficients is:

$$P_\mu = (A^T A)^{-1} A^T \quad (20)$$

5 Distance between $p(\vec{x})$ and \vec{x}

The distance between the projected vector and the one to project is very important in computer graphics applications. To arrange three-dimensional objects on screen right the distances must be known. Elsewise the objects are not displayed correct when they interfere with each other. Therefore an extra line is appended to the projection matrix that calculates the so-called depth. But there is a problem: The euclidean distance between the vectors is a non-linear function! So it is not representable with a matrix.

$$\|\vec{x} - p(\vec{x})\|_2 = \sqrt{\sum_{i=1}^n (x_i - p(\vec{x})_i)^2}, \quad \vec{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} \quad (21)$$

We need a function that expresses the distance and is linear function. So we use the 1-norm (and ignore the component-wise absolute value).

$$\|\vec{x} - p(\vec{x})\|_1 = \sum_{i=1}^n x_i - p(\vec{x})_i = \begin{pmatrix} 1 & \cdots & 1 \end{pmatrix} (\vec{x} - p(\vec{x})) \quad (22)$$

We see: A matrix form for this map exists. So it's suited for our purpose. Note: The 1-norm doesn't give us the real distance but it is a good alternative because the order of objects is calculated correctly.

This idea can be heavily improved when we make a base-change into our projection space inclusive the rest of vectors orthonormal to it. The vectors left are the degrees of freedom of our depth function, so in fact we just need to project our \vec{x} onto these and get automatically the 1-norm in our new base. At least the projections must be summed up.

So our depth function $d_1(\cdot, \cdot)$ is:

$$d_1(\vec{x}, p(\vec{x})) = \sum_{k=1}^j \langle \vec{n}_k, \vec{x} - p(\vec{x}) \rangle, \quad j = n - m \quad (23)$$

with \vec{n} as our base vectors left and orthonormal to projection space X .

We rearrange again to a matrix form:

$$\iff d_1(\vec{x}, p(\vec{x})) = \left\langle \vec{x} - p(\vec{x}), \sum_{k=1}^j \vec{n}_k \right\rangle \quad (24)$$

$$\iff d_1(\vec{x}, p(\vec{x})) = \left(\sum_{k=1}^j \vec{n}_k \right)^T (\vec{x} - p(\vec{x})) \quad (25)$$

For the projection function we insert the matrix (with E as identity matrix):

$$\iff d_1(\vec{x}, p(\vec{x})) = \left(\sum_{k=1}^j \vec{n}_k \right)^T (\vec{x} - P\vec{x}) \quad (26)$$

$$\iff d_1(\vec{x}, p(\vec{x})) = \left(\sum_{k=1}^j \vec{n}_k \right)^T (E - P)\vec{x} \quad (27)$$

And we get the depth matrix T that we can append to our projection matrix P .

$$T = \left(\sum_{k=1}^j \vec{n}_k \right)^T (E - P), \quad j = n - m \quad (28)$$

This projection has an extra advantage: Is $m = n - 1$, so $j = 1$, the depth function equals the Hesse normal form and retrieves the exact distance (2-norm)! Because there is only one orthonormal vector in this case the sum drops out:

$$d_1(\vec{x}, p(\vec{x})) = \vec{n}^T \vec{x} = \langle \vec{n}, \vec{x} - P\vec{x} \rangle = d_2(\vec{x}, p(\vec{x})) \quad (29)$$