Contents

COP290: Mathematical formulation

Indian Institute of Technology Delhi

Sunil Kumar(2016CS10314) & Anubhav Palway(2016CS10368

January 2018

1 Abstract

Here we will present 1)approach of reconstructing solids with planer surfaces from three-view engineering drawing.Our discussion will be mostly try to solve the problem of Reconstruction for polyhedron but is easily applicable for all sorts of solid bodies involving any spacial conic.Here we will present a matrix theory based approach.Later we will consider the reverse problem of 2)generating orthographic projection from a given 3D solid.First we will develop a relationship between a conic(including straight line) and its orthographic projections.At the end we approach the problem of 3)getting projection on oblique plane and projections of cutting section.

2 Introduction

1)The three views of engineering drawings has been widely used to illustrate product design.There are various approach to reconstruct 3D solid from these 2D drawings.Here we will use the B-rep oriented method which is a bottom up approach.Major steps involves:

- Generate 3D solids vertices from 2D vertices in each view.
- Generate 3D solids edges from 3D vertices using 2D drawings.
- Construct faces / planes using 3D solid's edges on same surface.
- Lastly construct 3D objects from faces/ Planes.

This complete process can be splited in 3 major steps

- Prepossessing of orthographic projections to prepare for 3D reconstruction. we will check the validity of input data at this stage.
- Analyzing relationship between 3D edges and their projection to generate a wire-frame from 2D drawings.Most of our mathematical modeling will be for this stage.
- Generating 3D solid by searching planes and solid parts within wire-frame.

Section 2 will explain relation between 3D edge and it's orthographic projection and section 3 will discuss about minimum views needed to generate a 3D solid.Algorithm for reconstructing 3D objects are presented in section 4.

2)For generating orthographic views from 3D solid we take projection of every vertices from 3D solid onto the projection plane. then we will construct 2D edges(may even be a conic) from 3D edge.We will use the modelling explained in section 2 and try to get a complete matrix based approach for getting 2D graph that can be used to print different orthographic projections.

3) For taking projection on oblique plane, we use rotation matrix to first rotate the points of 3D solid so that problem of getting can narrow down to first rotating the 3D solid and take orthographic projection in that view.For getting view of cutting section we have to eliminate all the points that lie of other side of plane while taking cutting section now we can take a projection on oblique plane of rest of solid for getting a cutting section view.

3 Matrix representation

In this section we develop relationship between conics(like circles, straight lines) and symmetric matrices, we know the general equation of conic in a plane can be given by algebraic expression

$$a_{11}x^2 + 2a_{12}xy + a_{22}y^2 + 2a_{13}x + 2a_{23}y + a_{33} \quad (1)$$

As our discussion main focus is polyhedron so we can clearly see if

$$a_{11} = 0,\ a_{22} = 0,\ a_{12} = 0$$

we get general equation of straight line so our major concern is a sub case of discussion. Using matrix notation, we get equation

$$f(u) = u^T A u \ = \ 0 \quad (2)$$

where u =$[x\ , y,\ 1]^T$, u^T is transpose of u, and **A** is the symmetric 3X3 matrix given by

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{12} & a_{22} & a_{23} \\ a_{13} & a_{23} & a_{33} \end{bmatrix}$$

The matrix **A** is required to be non-singular for (**u**) to be a non-degenerate conic[1]. This representation of a conic as symmetric matrix will help us in characterization of its orthographic projection.

definition 1: projection is linear transformation **P** from vector space to itself such that $P^2 \ = \ P$.This implies that **P** is idempotent.

For example, any Linear transformation of form **u=pu** is reflected by change of matrix A to A_p

$$A_P \ = \ P^T A P$$

So conic(including lines) and its orthographic projection in our representation is linear.

4 Minimum number of orthographic views needed

Definition 2:Non degenerate projection If plane containing conic is not perpendicular projection plane, then parallel projection is non-degenerate .Under non-degenerate parallel projections, all conics are equivalent,i.e conics are mapped to conics[1].

Now lets take an conic edge of 3D solid(edges connecting vertices), Its safe to assume that taking projection of this edge will lie in some plane α. Now lets take a coordinate system c_α whose axis x_α and y_α lie on plane α. Let the global coordinate system c in space, then global representation $x = [x\ y\ z\ 1]^T$ in c of point $x_\alpha = [x_\alpha \ \ y_\alpha \ \ z_\alpha \ \ 1]$ in c_α can be derived by using transformation.

$\mathbf{x} = \mathbf{R} x_\alpha + \mathbf{t}$

Where **R** , **t** are rotation and translation matrix respectively

Definition 3:Rotation Matrix [2] A matrix used to perform rotation in

Euclidean space. for example

$$\mathbf{R} = \begin{bmatrix} cos\theta & -sin\theta \\ sin\theta & cos\theta \end{bmatrix}$$

using above matrix we can rotate points in xy-plane counter clockwise through angle θ. To perform rotation using rotation matrix **R**, the position of point must be represented in coloumn vector **v**, containing coordinates of the point.Rotated vector is obtained by **Rv**.As in case of rotation we don't want to change relative distance of points so detR = 1.For any square matrix to be rotation matrix $R^T = R^{-1}$ and detR = 1.
A basic rotation (also called elemental rotation) is a rotation about one of the axes of a Coordinate system. The following three basic rotation matrices rotate vectors by an angle θ about the x-, y-, or z-axis, in three dimensions, using the right-hand rule—which codifies their alternating signs.

$$R_x(\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos\theta & -sin\theta \\ 0 & sin\theta & cos\theta \end{bmatrix}$$

$$R_y(\theta) = \begin{bmatrix} cos\theta & 0 & sin\theta \\ 0 & 1 & 0 \\ -sin\theta & 0 & cos\theta \end{bmatrix}$$

$$R_z(\theta) = \begin{bmatrix} cos\theta & -sin\theta & 0 \\ sin\theta & cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

General rotations

Other rotation matrix can easily be obtained by using above three matrix.
R $= R_z(\alpha)R_y(\beta)R_x(\gamma)$
this matrix produce desired result only when premultiplied as matrix multiplication is not commutative in nature.
Lets use homogeneous coordinates system[3]. the rotation matrix described above changes to

$$\begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} r_{00} & r_{01} & r_{02} & 0 \\ r_{10} & r_{11} & r_{12} & 0 \\ r_{20} & r_{21} & r_{22} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_\alpha \\ y_\alpha \\ z_\alpha \\ 1 \end{bmatrix} + \begin{bmatrix} t_0 \\ t_1 \\ t_2 \\ 0 \end{bmatrix}$$

Since we know any point on plane will have $z_\alpha = 0$, thus matrix transforms to

$$\mathbf{x} \;=\; \begin{bmatrix} r_{00} & r_{01} & t_0 \\ r_{10} & r_{11} & t_1 \\ r_{20} & r_{21} & t_2 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_\alpha \\ y_\alpha \\ 1 \end{bmatrix} \;=\; \mathbf{P}u_\alpha \qquad (4)$$

where $u_\alpha = [x_\alpha, y_\alpha, 1]^T$ and space geometry of matrix i given by $\mathbf{P}$.

Lets now discuss relationship between a space conic and it's orthographic projections onto some projection planes.let c_i for i = 1,2,...,q denote a 2D co-ordinate system associated with i^{th} projection.Now consider C_i is 3X4 matrix whose 3 columns form an orthogonal basis for this projection subspace, then transformation from 3D to this plane $u_i \;=\; C_i x \quad (5)$.
Now substituting in (5) from earlier equations we get(6)

$$u_i = C_i P u_\alpha \;=\; G_i u_\alpha \quad (6)$$

Now as stated in [1] we will show 3 distinct orthographic projections are sufficient to uniquely recover a space conic.
Let $\mathbf{A}$ be a space conic that lies on plane p so equation of space conic from discussion section 2

$$u_p^T A u_p \;=\; 0 \qquad (7)$$

and its projections on different curves A_i are represented by

$$u_i^T A_i u_i \;=\; 0 \qquad (8)$$

using equation (6) in equation (8)

$$G_i^T A_i G \;=\; P^T C_i^T A_i C_i P \;=\; A \qquad i \;=\; 1,2,3....q \quad (10)$$

where $\mathbf{A}$ and $\mathbf{P}$ are unknown matrices. It can be inferred that from 3 orthographic projection we will get 18 equations in 15 unknowns (As $\mathbf{A}$ is symmetric matrix. By Bernstein's seminal theorem[4], we can derive that equations are solvable. But for case of degenerate orthographic projections we still can find centre of conic from two other views.Without loss of generality consider we can get centre from front view So for we can found $\mathbf{A}$ from $P^T C_f^T A_f C_f P \;=\; A$, where subscript f indicates front view.Hence if there is at least one non-degenerate view we can get 3D reconstruction.

5 Reconstruction of wire-frame

After our discussion on basic relation between 3D representation and 2D view. To reconstruct lets use idea of Kuo[8]. A vertex from 2D to 3D can be generated from 3 views.Let a node N_f represents a vertex generated into 3d from front view so $N_f = (N_f(x), N_f(z))$ where $N_f(x)$ is x coordinate of point in 3D generated from a point in front view. we can't get value of $N_f(y)$ from front view so it can be any arbitrary value but we know get this information from other views as that point is other views too.

Now let δ be tolerance factor, that is error in getting co-ordinates from drawing while reading file using image scanner.So for a point in 3D to exit there should be

$|N_f(x) - N_t(x)| \ < \delta$, $|N_t(y) - N_s(y)| \ < \delta$ and $|N_f(z) - N_s(z)| \ < \delta$

but this method only works for polyhedron objects so for generating conic edges of 3D object we must determine **A** from equation (10) for all three orthographic projections. So for $G_i^T A_i G \ = \ P^T C_i^T A_i C_i P \ = \ A \quad i = 1, 2, 3$ for three orthographic views we know

$$C_1 \ = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad C_2 \ = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad C_3 \ = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Thus from $G_i = C_i P$ we get

$$G_1 \ = \begin{bmatrix} r_{00} & r_{01} & t_0 \\ r_{10} & r_{11} & t_1 \\ 0 & 0 & 1 \end{bmatrix} \quad G_2 \ = \begin{bmatrix} r_{00} & r_{01} & t_0 \\ r_{20} & r_{21} & t_2 \\ 0 & 0 & 1 \end{bmatrix} \quad G_3 \ = \begin{bmatrix} r_{10} & r_{11} & t_1 \\ r_{20} & r_{21} & t_2 \\ 0 & 0 & 1 \end{bmatrix}$$

A_i (i=1,2,3) can be described by an algebraic expression

$$a_{11}x^2 + 2a_{12}y^2 + a_{22}y^2 + a_{13}x + a_{23}y + a_{33} = 0$$

this expression will give equation of conic in i^{th} projection from where we can get **A** by left multiplying by P.

But in case of straight lines this approach will not work as for lines **A** is singular ie. $\mathbf{det(A) = 0}$.So we need work round the problem at hand.By Kou's method we already have all the vertex of 3D solid we just need to connect right set of points to generate the edges by in depth exploration of 2D drawings.If there is edge between $u_1 = [x_1, y_1, z_1]$ and $u_2 = [x_2, y_2, z_2]$ in 3D solid then in there is edge corresponding this edge in at least two views.So if u_1 is generated from Nodes $N_f^{(1)}$,$N_s^{(1)}$ and $N_t^{(1)}$ and u_2 is generated from Nodes $N_f^{(2)}$,$N_s^{(2)}$ and $N_t^{(2)}$.We know that 2D drawings can be represented as a graph representing all points in that view and edges that exist between them.So for an edge to exist between u_1 *and* u_2 for all 3 views either

1) both corresponds to same point eg $N_f^{(1)} = N_f^{(2)}$, or
2) the points in 2d has edge between them.
So we can run an exhaustive search for all the edges which will consider all possible cases $\mathbf{C}_2^n$.So it will turn out to be an order n_2 algorithm.

Algorithm:

```
1) take two points from set of all points such that u1! = u2
2)  for all 3 views:
        if(corresponds to same point in 2D)
            continue;
        else if(has an edge between them in 2D)
            continue;
        else
            return false;
    return true;
3) If there result of previous step is true add an edge
in 3D representation and run this for next pair of point.
4) At the end we will get wire frame of 3D solid.
```

Now we have obtained wire frame of solid and only thing needed is get faces and 3D solid.

6 Reconstructing 3D solids

At this stage we already have a wire frame model generated from previous stage. we now find faces of solid.

6.1 Face Generation

Definition 4:Surface unbounded face is called surface and face is region of a surface that is bounded by edge loop.Surface is generated by two edges having

same vertex.
Now we will first generate all the surface that can contain a face, for this we know by face that two edges that have a common vertex can make a surface.Although type of two edges and their relationship can uniquely determine the type and features of the surface[1].
If we explore all possible cases of faces it will cost a lot of computation so using[1] we know.
Property1: Each face of 3D-object corresponds to at least a closed area in one pf the views.
Once a surface is identified our algorithm searches for all possible on surface for tracing face.For every vertex on surface all the edges form it are checked those on surface are recorded and other are stored for further checking.Now using all these points we need to find edge loop which will give face in our case. The last step is to find all the solids that can constructed from given set of faces.

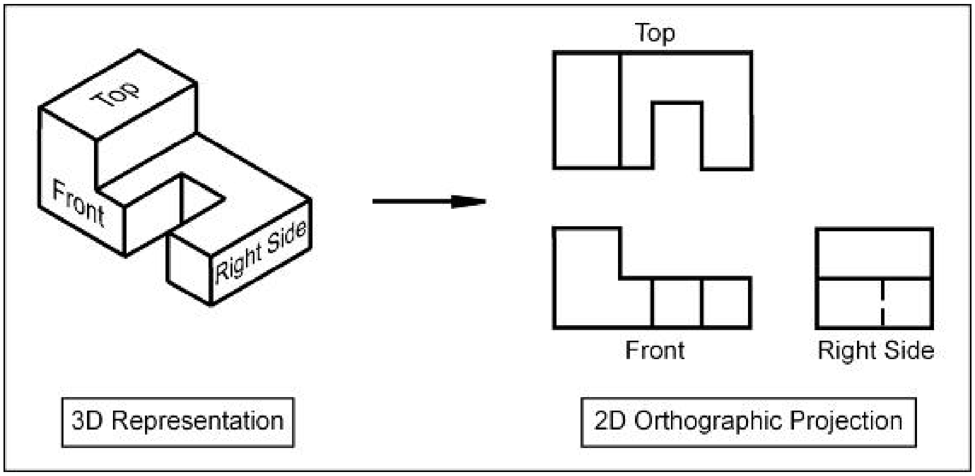
6.2 Searching solid

Now we have generated all possible pair of face some of them are even holes so we need to choose which face are really required for 3D solid. We can either run a exhaustive search for for all the solids that are possible which are possible with a subset of faces generated and trace 2D drawings back and check which is the best match but will cost alot of time in computation.
So to resolve ambiguities we use Moebius rule and definition of manifolds.These state that an edge can be part of two faces in manifold solids and orientation of edge is inverted by each face[9].Using divide and conquer check that every edge is only part of two planes and remove false planes.

7 3D to 2D: Projection(Linear Algebra)

Projection is a linear transformation P from a vector space to itself such that P^2 = P.

7.1 Orthogonal Projection



The function which maps the point (x, y, z) in three-dimensional space R^3 to the point (x, y, 0) is an orthogonal projection onto the x–y plane. This function is represented by the matrix

$$P = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

When we left multiply P with arbitrary coulum vector

$$P \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} x \\ y \\ 0 \end{bmatrix}$$

Often it's more useful to use homogeneous coordinates[5].The transformation matrix becomes

$$P = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

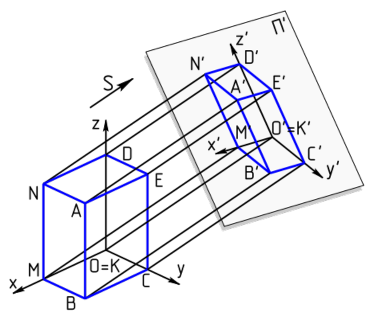
and for each homogeneous vector **V**$= (v_x, v_y, v_z, 1)$ the transformed vector would be

$$P = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \mathrm{v}_x \\ v_y \\ v_z \\ 1 \end{bmatrix} = \begin{bmatrix} \mathrm{v}_x \\ v_y \\ 0 \\ 1 \end{bmatrix}$$

for converting V=$[v_x, v_y, v_z, 1]^T$ to $P_v = [v_x, v_y, 1]^T$ we need to use 3X4 matrix.

$$C_f = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad C_t = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad C_s = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

7.2 Projection on Skew plane



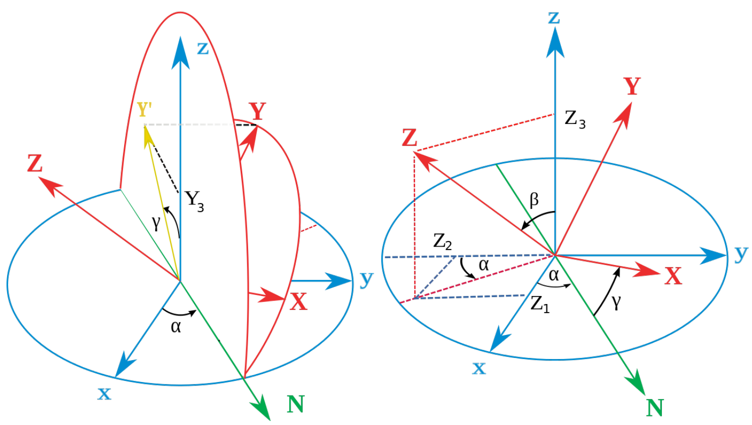
It is type of parallel projection.In this projectors in projection plane intersect the projection plane at an oblique angle to produce the projected image, as opposed to the perpendicular angle used in orthographic projection.

In Parallel projection of a point (x,y,z) on xy-plane gives (x+az,y+bz,0) , here a and b uniquely specify the parallel projection and in case of orthogonal projection these are a=b=0
For taking projection about any arbitrary plane P we need first find Euler angles[6] α, β, γ about axes x,y and z of the plane P.So for getting Projection we can rotate all the point on body by α, β, γ in negative direction and rotation will be

$$\mathbf{R} = R_z(-\gamma)R_y(-\beta)R_x(-\alpha)$$

.

To get values of Euler angles[6] $\alpha,\ \beta\ and\ \gamma$ we will use normal vector of of plane of projection.we will try to rotate body so that plane **P** will take place of xy plane. So assuming a fame with unit vectors as in diagram it can be



observed that:
$cos(\beta) = Z_3$ and since we know $sin^2x \ = \ 1-cos^2x$ we have $sin(\beta) \ = \ \sqrt{1 - Z_3^2}$.

As Z_2 is the double projection of a unitary vector,

$$cos(\alpha)sin(\beta) \ = \ Z_2$$

$$cos(\alpha) = Z_2/\sqrt{1 - Z_3^2}$$

There is a similar construction for $\mathbf{Y}_3$, projecting it first over the plane defined by the axis z and the line of nodes. As the angle between the planes is $\pi/2 - \beta$ and $cos(\pi/2 - \beta = sin(\beta)$, this leads to:

$$sin(\beta)cos(\gamma) = \mathbf{Y}_3,$$

$$cos(\gamma) = \mathbf{Y}_3/\sqrt{1 - Z_3^2,}$$

So finally we get

$$\alpha = \arccos(Z_2/\sqrt{1 - Z_3^2}$$

$$\beta = \arccos(Z_3)$$

$$\gamma = \arccos(Y_3/\sqrt{1 - Z_3^2})$$

If user has not defined the coordinate system for plane we can take $Y_3 = 0$ then $\gamma = 90$ degree and if coordinate system is defined we can calculate value of γ.

These matrices produce the desired effect only if they are used to premultiply column vectors, and (since in general matrix multiplication is not commutative) only if they are applied in the specified order.

8 Sectional View from 3D Solids

Now here we assume if we have 3D solid then it implies that we already know all the vertex of solid and their coordinates.So to get sectional view we first need a section that cut one face and direction of section. Then we will take projection of all the points on other side that plane.All the lines that cut plane will also give us points whose projection is themselves for plane.

So now problem is how will we decide about hashing the area cut by section.For that we already know the points where lines intersect with plane now we need to make closed faces which had solid part there then these faces will be hashed and rest can assumed a cavity.

9 References

1) http://www.shixialiu.com/publications/matrixreconstruction/paper.pdf

2) https://en.wikipedia.org/wiki/Rotation$_matrix$

3) https://en.wikipedia.org/wiki/Orthographic$_projection$

4) https://en.wikipedia.org/wiki/Bernstein-Kushnirenko$_theorem$

5) https://en.wikipedia.org/wiki/Homogeneous$_coordinates$

6) https://en.wikipedia.org/wiki/Euler$_angles$

7) Lequette R. Automatic construction of curvilinear solids from wire frame view.Computer Aided Design 1988;20(4):171-179

8) MH Kuo, Reconstruction of quadric surface solids from three-view engineering drawings, Computer Aided Design, 1998,30(7), pp. 517-527

9) https://web.stanford.edu/ jchw/WOMPtalk-Manifolds.pdf