Simulation of Volume Measurement of Glass Gobs

Master’s Thesis in Electrical Engineering

Sohail Muzamil
Muhammad Shoaib Perveiz

School of Information Science, Computer and Electrical Engineering
Halmstad University
Simulation of Volume Measurement of Glass Gobs

School of Information Science, Computer and Electrical Engineering
Halmstad University
Box 823, S-301 18 Halmstad, Sweden

JUNE 2010

Copyright © Sohail Muzammil
Muhammad Shoaib Perveiz
Abstract

We present a geometrical and mathematical solution to a problem faced in the glass industry in this work. Volume measurement of the glass gob is vital in making glassware. Geometric models were used to represent the glass gob.

A line scan camera system takes the images of the glass gob and the volume information of the glass gob is obtained by the image processing in the industry. This work is carried out to implement a simulator which estimates the change in the volume measurement of glass gob through line scan when it is rotated or when its shape is changed. A mixture of graphical and mathematical approaches is used to carry out this study. Geometric models have been used to represent the different gob models. Geometric models facilitate the manipulation of volumetric data.

A simple and effective technique is used in this work. The problem is divided into steps. Volume measurement through a line scan technique is simulated. An easy to use graphical user interface (GUI) is designed to interact with the gob model and check the results of volume measurements.

Keywords

Glass Gob, 3D model, Rotation, Projection, Line Scan
Preface

This thesis is submitted to the Halmstad University for partial fulfilment of the requirements for the degree of Master of Science.

This Master’s work has been performed at the School of Information, Computer and Electrical Engineering (IDE), with Kenneth Nilsson as supervisor.
Acknowledgements

We are very thankful to Dr. Kenneth Nilsson for his guidance, supervision and especially for his patience.

We are very thankful to our parents for helping us to study abroad.

We are also thankful to our all friends for their support especially Mr. Haseeb Ullah and Last but not the least we are thankful to IDE Halmstad for providing us the facilities to carry out our research work freely and with ease.
Contents

Abstract.......................................................................................................................... ii
Keywords ....................................................................................................................... ii
Preface......................................................................................................................... iii
Acknowledgements...................................................................................................... iv
Contents ...................................................................................................................... vi
List of Figures and Tables........................................................................................... 8

Introduction............................................................................................................... 10
  1.1 Company: ........................................................................................................... 10
  1.2 Project Objectives: ............................................................................................ 10
  1.3 Glass gob: ......................................................................................................... 10
  1.4 Environment: .................................................................................................... 11
  1.5 Project formulation: ......................................................................................... 11

Background ............................................................................................................... 12
  2.1 Model: .............................................................................................................. 12
  2.2 3D Modelling: .................................................................................................. 12
  2.3 3D Objects Models: ......................................................................................... 13
  2.4 Rotation: ........................................................................................................... 13
    2.4.1 2D Rotation: .............................................................................................. 13
    2.4.2 3D Rotation: .............................................................................................. 14
  2.4 Projection: ........................................................................................................ 15
  2.5 Line Scanning: .................................................................................................. 16
  2.6 Software Tools: ............................................................................................... 16
    2.6.1 AC3D: ........................................................................................................ 16
    2.6.2 Java 3D API: .............................................................................................. 16
    2.6.3 3D Studio Max: .......................................................................................... 17
    2.6.4 AutoCAD: ................................................................................................ 17
  2.7 Related Work: .................................................................................................. 18

Selected Method ....................................................................................................... 20
  3.1 Software Tool: ................................................................................................. 20
    3.1.1 Selected Approach .................................................................................... 20
  3.2 Representation: .............................................................................................. 21
    3.2.1 Ideal Gob Model: ..................................................................................... 21
    3.2.2 Non Ideal Gob Model: ............................................................................. 22
    3.2.3 Memory Representation: ........................................................................ 23
  3.3 Rotation: .......................................................................................................... 24
List of Figures and Tables

Figure 1.1: Environment .................................................................11
Figure 2.1: 2D Rotation .................................................................14

Figure 3.1: (a) Cylinder parameters with respect to axis.
(b) Top view of cylinder. .............................................................22
Figure 3.2: Examples of Non ideal gobs ..........................................23
Figure 3.3: Memory representation .................................................23
Figure 3.4: Rotation in X, Y, Z axis ..................................................24
Figure 3.5: (a) Camera position in 3D environment (On Y-axis) ..........26
(b) Camera visualization of 3D Cylinder .......................................27
Figure 3.6: Boundary pixels Selection ............................................28
Figure 3.7: Process of Line scan .....................................................30

Figure 4.1: Graphical User Interface ...............................................31
Figure 4.2: Ideal gob model ..........................................................32
Figure 4.3: Examples of Non ideal gobs ..........................................33
Figure 4.4: 0, 45 & -45-degrees rotation in X,Y,Z .........................33
Figure 4.5: Camera view of the rotation shown in figure 4.4 ...............34
Figure 4.6: Boundary point extraction from camera view ..................34
Figure 4.7: Rotation Vs Line Scan Volume for ideal model for table 4.1 ..36
Figure 4.8: Rotation Vs Line Scan Volume for table 4.2 .....................36
Figure 4.9: Rotation Vs Line Scan Volume for table 4.3 .....................38
Figure 4.10: Rotation Vs Line Scan Volume for table 4.4 ...................38
Figure 4.11: Rotation Vs Line Scan Volume for table 4.5 ...................39
Figure 4.12: Rotation Vs Line Scan Volume for table 4.6 ...................40

Figure A.1: Ideal gob model Line scanning for different rotation in X ....44
Figure A.2: Ideal gob model Line scanning for different rotation in Y ....45
Figure A.3: Ideal gob model Line scanning for different rotation in X & Y 46
Figure A.4: Non Ideal gob model 1 Line scanning for different rotation in X 46
Figure A.5: Non Ideal gob model 1 Line scanning for different rotation in Y 47
Figure A.6: Non Ideal gob model 1 Line scanning for different rotation in X & Y 48
Figure A.7: Non Ideal gob model 2 Line scanning for different rotation in X 49
Figure A.8: Non Ideal gob model 2 Line scanning for different rotation in Y 49
Figure A.9: Non Ideal gob model 2 Line scanning for different rotation in X & Y 50

Table 4.1: Rotation in X, Y & respective line scan volume for dz =0.1 ......35
Table 4.2: Rotation in X, Y and respective line scan volume for dz =0.03 ....35
Table 4.3: Rotation in X, Y & respective line scan volume for dz =0.1 ......37
Table 4.4: Rotation in X, Y & respective line scan volume for dz =0.03 ....37
Table 4.5: Rotation in X, Y & respective line scan volume for dz =0.1 ......39
Table 4.6: Rotation in X, Y & respective line scan volume for dz =0.03 ....39
1. Introduction

1.1 Company:

Gedevelop is a company involved in the glass manufacturing industry based in Sweden. It manufactures inspection, measurement and control systems. It is involved in the manufacturing, development, design, supply and services of inspection, control and measurement equipment for the world’s glass industry.

1.2 Project Objectives:

The objective of this project is to implement a simulator with a Graphical User Interface (GUI) and to study the effect of rotation and shape changes on the volume estimation of glass gob through the line scan.

1.3 Glass gob:

Glass gob is the molten glass formed by the cutting of the stream of glass as it flows from a feeder into a spout of variable diameter. The length of the gob depends on its diameter. A forming machine moulds these gobs into bottles and other glass objects. Information about glass volume is important to minimize cost and maximize the quality of glass containers. Gedevelop is using a non-contact measurement system GIA for the above mentioned purpose.
1.4 Environment:

Figure 1.1 is the environment that shows us that how the gob is analyzed. A line camera takes the image of gob and then this image is processed to take the measurements of the gob.

![Figure 1.1: Environment](image)

1.5 Project formulation:

This project was divided into following steps.

- We represented the ideal and non ideal glass gob with a suitable graphical model. A method was designed to interact with this model.
- We carried out the implementation of our design in Matlab.
- This implementation was verified and checked to establish how successful the above procedure has been.
- A GUI was created to help user to interact with graphical model of glass gob.
- The line scan procedure of the camera was simulated and the volume of the model was calculated.
2 Background

2.1 Model:

A model is a representation which is designed to show the main object or workings of an object, system, or concept [1].

A mathematical representation of any object or anything such as: a person, a building, a vehicle, a tree is called a “computer model”. It can represent a process: a weather pattern, traffic flow etc.

We can design models using a set of data, equations and computations that emulate the actions of things represented. Models are usually a graphical display of these numbers, equations that can be seen on a computer screen or by means of some other visual device [2].

2.2 3D Modelling:

3D modelling comprises the method to develop a graphical representation of a three dimensional object. This graphical representation is based on mathematics. This development process is carried out on with the help of specialized software. The product of this process is called a “3D” model. This model is displayed as a 2D image by 3D rendering. 3D models are created using a set of points in 3D space. These points are connected by geometric entities like lines, triangles or surfaces etc [3].
Object representation is a very essential task in 3D modelling. Recognition of object is done by the comparison of object description and model description. Representation of complex, curved objects is still a challenging and difficult problem. Face-based, local representations using primitives based on orientation of boundaries and Gaussian curvatures have been developed for representation of curved objects [4].

2.3 3D Objects Models:

There is a whole range of the 3D geometric models present to represent the physical objects. Popular models include cylinder, 3D rectangle, ellipsoid, cones, spheres etc.

2.4 Rotation:

Rotation is turning of an object around a centre point. It is rigid body transformation around a point. This point is fixed. There is a certain axis of the rotation which is perpendicular to plane of this motion. For rotation this axis of rotation has to lie inside the body of the object being rotated. If the object is rotated around x, y or z axis then these rotations are called “principal rotations”. Any rotation of the object can be done by rotating it in a combination of principal rotations.

2.4.1 2D Rotation:

2D Rotation is the rotation of a 2D object around any fixed point. Mathematically we can describe the 2D rotation of an object around an origin (0,0) from a point (x, y) to a new point(x’, y’) as

\[ x = r \cos \vartheta \]  

(2.1)
\[ y = r \sin \theta \]  

(2.2)

\[ x' = r \cos(\theta + \phi) = r \cos \theta \cos \phi - r \sin \theta \sin \phi \]  

(2.3)

\[ y' = r \sin(\theta + \phi) = r \sin \theta \cos \phi - r \cos \theta \sin \phi \]  

(2.4)

**Figure 2.1: 2D Rotation**

Where \( \theta \) = initial angle and \( \phi \) = angle of rotation.

### 2.4.2 3D Rotation:

3D Rotation has more complexity to it in comparison to 2D rotation. The axis of rotation is perpendicular to the XY plane in 2D rotation case. An axis of rotation need to be specified in a 3D case. There are three principal axes (X, Y, Z) around which a 3D object can be rotated [5]. The order is very important in 3D rotation as in the case of equation 2.5, it is important to first do rotation in Z then Y and X. Changing the order leads to different results.

\[ P' = R_z \times R_y \times R_x \times P \]  

(2.5)
where

\[
R_x = \begin{bmatrix}
1 & 0 & 0 \\
0 & \cos\vartheta_x & \sin\vartheta_x \\
0 & -\sin\vartheta_x & \cos\vartheta_x
\end{bmatrix}
\]  \hspace{1cm} (2.6)

\[
R_y = \begin{bmatrix}
\cos\vartheta_y & 0 & \sin\vartheta_y \\
0 & 1 & 0 \\
-\sin\vartheta_y & 0 & \cos\vartheta_y
\end{bmatrix}
\]  \hspace{1cm} (2.7)

\[
R_z = \begin{bmatrix}
\cos\vartheta_z & -\sin\vartheta_z & 0 \\
\sin\vartheta_z & \cos\vartheta_z & 0 \\
0 & 0 & 1
\end{bmatrix}
\]  \hspace{1cm} (2.8)

Equations 2.6, 2.7 and 2.8 are the rotation matrices for X, Y and Z rotation respectively.

\subsection*{2.4 Projection:}

Projection is point by point shifting or transformation of an object from one plane to another. This is done by using parallel lines which connect the corresponding points on both planes [6].

\subsubsection*{2.4.1 3D projection:}

This is the method of projecting a 3D object to a 2D plane. It involves the plotting of 3D points to a 2D plane. It is one of most current methods for displaying graphical data. There are two commonly used methods for 3D projection namely orthographic and perspective projection. Orthographic projection gives the profile, bird eye, cross section or elevation view of a 3D object. It is obtained by considering the view plane as being
parallel to any of the three axes. Orthographic projections are not used for camera view although they give information about the 3D object.

Perspective projection is used where the object is considered to be viewed by a camera. This view is obtained by considering the camera as being parallel to X, Y or Z axis.

2.5 Line Scanning:

A line-scan camera reads an object using a single array of pixels of sensors. It reads the object in line by line. The output of this camera is usually processed by a computer. A two dimensional images is created by this computer based on the data that it receives from the camera system. Area cameras use the whole matrix of pixel sensor for capturing the image.

2.6 Software Tools:

Following are some of the tools that are used for 3D modelling and graphics

2.6.1 AC3D:

AC3D is used for creating 3D models in games, rendering images and for visualizing scientific and general data. This software runs on a wide range of hardware. It requires only a good 3D graphics card [7].

2.6.2 Java 3D API:

This is a special Java 3D application programming interface (API) used for writing three-dimensional graphics applications and applets. It allows the creation, construction and
manipulation of 3D geometry. Virtual words are described using this tool. High level constructs are also used which provide Java 3D with enough information to render these worlds efficiently. Java 3D delivers portability benefit to developers of 3D graphics applications. It is part of the Java Media suite of APIs and it is available on a wide range of platforms. Since application and applets which are written in Java 3D have greater access to the Java classes so it integrates very well with the internet [8].

2.6.3 3D Studio Max:

3D studio max was developed as a tool to provide 3D visualization of their 2D CAD/CAM drawings. It provided design visualization with animating extensions thus providing a full animated environment. There are hundred of plug-in components which are organized around a core engine. It has a scripting language (Maxscript) and it is basically a polygonal modelling tool. A 3D environment called world space is used in 3D Max which allows the user to move, rotate and scale 3D objects. 3D object modelling is done by sculpting it by manipulating its geometric components [9].

2.6.4 AutoCAD:

AutoCAD is a Computer Aided Design or Computer Aided Drafting tool that is used for both 2D and 3D design and drafting. Primitive entities like lines, polylines, circles, arcs and text are basis of the complex objects used in Auto CAD. AutoCAD has supported custom objects through its C++ API. Modern AutoCAD includes a full set of basic solid modelling and 3D tools. Improved 3D tools and solid modelling was introduced in AutoCAD 2007. These features provide better modelling and interaction with 3D models [10].
2.7 Related Work:

There is an immense body of work on 3D modelling, rotation, rendering and shape modelling using geometric entities. Jen-Hui Chuang, Narendra Ahuja, Chien-Chou Lin, Chi-Hao Tsai and Cheng-Hui Chen presented a very good approach to represent complex 3D objects in their work [11]. They used generalized cylinder (GC) for different shape representations. A scaling function, a cross section curve and an axis is required to define a GC. GCs are very flexible and capable of modelling many types of real world objects. They proposed the decomposition of 3D objects into simple volumetric parts, designing these simple parts and then interlinking them with each other through different relations.

Jean Ponce, David Chelberg, and Wallace B. Mann in their work *Invariant Properties of Straight Homogeneous Generalized Cylinders and Their Contours* [12] also advocated the use of planar face objects for the problems which required the projection of three dimensional objects on two dimensional planes. There is loss in the information present in a three dimensional object when it is projected on a two dimensional plane. The generalized cylinder is very suitable for visual representation because of its expressiveness and also because it is a concise combination of both volume and surface information.

In a more recent work of J.C. Can, R. K. Beatson, J. B. Cherrie, T. J. Mitchell, W. R. Fright, B. C. McCallum, and T. R. Evans [13] Radial Basis functions were used for representing different types of 3D objects. A zero set of RBF fitted to a given surface data is used to implicitly define any objects surface. Large data sets consisting of millions of data points are modelled using only one single RBF. This is possible due to availability of fast methods for fitting and evaluating RBFs. RBF uses interpolation and extrapolation in the functional representation.

Martin Kilian and Niloy J. Mitra used computing with geometric shapes in their work [14]. They termed computing with geometric shapes as the basis of geometric modelling and processing. They viewed a shape as set of points and by using available data they
represented these shapes. They used an entirely geometric approach. They based their design and modelling process on geodesic curves.

Wu-Yuin Hwang, Jia-Han Su, Yueh-Min Huang and Jian-Jie [15] designed a Virtual Manipulatives and Whiteboard (VMW) system for solving complex mathematical area and volume problems. They used the geometric entities in their system. Users had the facility to manipulate the virtual objects in 3D space which helped them in solving the geometric problems. By using the three dimensional models users expressed their mathematic solutions and reasoning.
3

Selected Method

3.1 Software Tool:

We have selected Matlab for our problem. The MATLAB (Matrix Laboratory) is a computation and visualization software tool. It is a versatile and flexible tool which allows users with even the most elementary programming capabilities to produce sophisticated graphics and graphical user interfaces. The reason to choose Matlab is the strong mathematical programming capability combined with superb graphics modelling and rendering.

Since we are dealing with the numerical data and graphical representation, Matlab is one of the best applications available for providing both the computational capabilities of generating data and the means for displaying it in a variety of graphical representations. The latest version of MATLAB has taken on a new look, it has a new integrated development environment (IDE), new graphics development tools, and introduces some new functions. In a three-dimensional world, much of our information is best revealed with 3-D techniques. MATLAB provides a wealth of graphics functions that let you make quick 3-D plots and visualizations of your data [16].

3.1.1 Selected Approach

The idea was to use an approach similar to generalized cylinders. This allowed us to have a nice representation for the different gob models. It gave us the flexibility to manipulate
the numerical data with ease, since there was a lot of computation involved in rotation, projection and line scanning.

3.2 Representation:

The ideal gob shape is a cylinder. We used a cylinder based on parametric equations for the representation of the ideal gob shape. We represented the non ideal gob shape with a curved cylinder. Ideal and non-ideal gobs are represented in such a way that there are $N$ circles each comprising of $M$ pixels placed over one and other. The height of the gob model is $h$.

3.2.1 Ideal Gob Model:

A cylinder is defined by a surface which is made of the points at a fixed distance from the axis of cylinder. It is one of the most commonly used geometric shapes. The axis of the cylinder is the line that passes through the centre of the cylinder bases. The axis is parallel to the curved face of the cylinder. A cylinder can be parametrically defined by eq. 3.1, 3.2 and 3.3

\[
x = r \cos \theta \\
y = r \sin \theta \\
z = z
\]

Where $r =$radius and $z \in [0, h]$ and $\theta \in [0, 2\pi]$

The surface area (S) and volume (V) of a cylinder of height ‘$h$’ and a radius ‘$r$’ is given by eqs. 3.4 and 3.5 i.e.

\[
S = 2\pi rh
\]

\[
V = \pi r^2 h
\]
\[ V = \pi r^2 h \]  \hspace{1cm} (3.5)

Figure 3.1 (a) shows that how width, depth and height is represented in our model of the cylinder, where ‘x’ represents the WIDTH, ‘y’ represents the DEPTH and ‘z’ is the HEIGHT of the cylinder. Figure 3.1 (b) is its top view.

3.2.2 Non Ideal Gob Model:

A non ideal gob model can be curved and have a different radius. We designed such models to represent non ideal gob by making radius ‘r’ as a function of the ‘z’ i.e.:

\[ r = f(z) \]  \hspace{1cm} (3.6)

By using 3.6 we rewrote eqs. 3.1 and 3.2 as

\[ x = f(z)\cos \phi \]  \hspace{1cm} (3.7)
$y = f(z) \sin \theta$ \hspace{1cm} (3.8)

Figure 3.2 shows two examples of the non ideal gob.

![Figure 3.2: Examples of Non ideal gob](image)

### 3.2.3 Memory Representation:

The memory representation of both ideal and non ideal gob models is shown in figure 3.3. Each point has three coordinates X, Y and Z and they are placed in their own respective matrices.

![Figure 3.3: Memory representation](image)
3.3 Rotation:

Since we had to deal with the 3D model hence we implemented 3D rotation. The tilting of gob the model was done with the help of rotation along the X, Y and Z axes as shown in figure 3.4. The rotation along the Z axis is done first of all of the three rotations and it has no effect on the shapes because all used gob models in this work are symmetrical around the Z axis.

![Figure 3.4: Rotation in X, Y, Z axis](image)

3.3.1 Rotation Matrix:

The 3D rotation is based on a rotation matrix which is given as:

\[
R_x = \begin{bmatrix}
1 & 0 & 0 \\
0 & \cos \theta_x & \sin \theta_x \\
0 & -\sin \theta_x & \cos \theta_x
\end{bmatrix}
\]  

(3.9)

\[
R_y = \begin{bmatrix}
\cos \theta_y & 0 & \sin \theta_y \\
0 & 1 & 0 \\
-\sin \theta_y & 0 & \cos \theta_y
\end{bmatrix}
\]  

(3.10)
The rotation matrices are defined as:

\[
R_x = \begin{bmatrix}
\cos \theta_x & -\sin \theta_x & 0 \\
\sin \theta_x & \cos \theta_x & 0 \\
0 & 0 & 1
\end{bmatrix}
\]

\[
R_y = \begin{bmatrix}
\cos \theta_y & 0 & \sin \theta_y \\
0 & 1 & 0 \\
-\sin \theta_y & 0 & \cos \theta_y
\end{bmatrix}
\]

\[
R_z = \begin{bmatrix}
\cos \theta_z & -\sin \theta_z & 0 \\
\sin \theta_z & \cos \theta_z & 0 \\
0 & 0 & 1
\end{bmatrix}
\]

Where

\[
\theta_x = \text{rotation angle in x direction}
\]

\[
\theta_y = \text{rotation angle in y direction}
\]

\[
\theta_z = \text{rotation angle in z direction}
\]

New position points of the whole surface \( P' = [x', y', z']' \) are calculated on the basis of these rotation matrices.

\[
P' = R_x \times R_y \times R_z \times P
\]

(3.12)

where \( P = [x, y, z]' \) are the original points.

Any rotation of the gob model requires it to be rotated first along the Z axis as the order in equation 3.12.

The models are symmetric around the central line which is equal to the height of the model. Whenever the model is rotated around any axis the central line \( Q = [x_c, y_c, z_c]' \) is also rotated. The new position \( Q' = [x_c', y_c', z_c']' \) is obtained by multiplying the central line points \( Q \) with the rotation matrices as in equation 3.12. Where \( P \) and \( P' \) are replaced by \( Q \) and \( Q' \) respectively.

### 3.4 Camera View:

We had to implement a camera view and since we have a line scan camera so we implemented the 3D projection to simulate the angle at which the camera views the gob. This was achieved by simulating the optical axis of the camera along the Y axis. For this
purpose we zeroed the values of Y and took the data points of X and Z according to eq. 3.13:

\[
[x', z'] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} x' \\ y' \\ z' \end{bmatrix}
\]  

(3.13)

Where \([x', z']\) is the matrix containing all the surface points in XZ plane. Figure 3.5 (a) shows that how the camera view is simulated. Figure 3.4 (b) shows the projection of a 3D object (cylinder) in the camera.

![Figure 3.5: (a) Camera position in 3D environment i.e on Y-axis](image)
3.5 Boundary Points Extraction:

In the 2D view we got the whole surface points of the gob after rotation (figure 3.5 (b), to the left) but since we were only concerned with the boundary so we extracted the boundary points (figure 3.5 (b), to the right) of this 2D view with a step wise procedure. This procedure is divided in two parts.

(a) For left and right sides of the model.

(b) For top and bottom of the model.

a) **FOR LEFT AND RIGHT SIDES OF THE MODEL:**

1. For every circle, centre to point vectors are calculated. The vectors which are toward the Y-axis have a positive Y-coordinate see figure 3.6. These vectors are called “positive vectors”.

2. The positive vectors which have minimum Y-coordinates are the left and right border points of each circle.
b) **FOR TOP AND BOTTOM OF THE MODEL:**

1. Calculated the maximum and minimum Z positions for top and bottom circles respectively. These are the maximum and minimum Z-coordinates see green in figure 3.6.
2. Find the *start* and the *last* coordinate positions of top and bottom circles. These coordinates are the left and the right border points of the top and bottom circles respectively see red points in figure 3.6.
3. For top circle draw the upper curve and for bottom circle draw the lower curve of the circle.

![Figure 3.6: Boundary pixels Selection](image)
3.6 Line Scan Procedure and Volume Estimation:

The basic idea that we used to simulate volume measurement through line scan was that we divided the object in XZ-plane in small portions. Since a line scan is based on line by line reading of an object by using this technique we consider the portion between two consecutive lines as a separate portion having its own radius and height (see figure 3.7). Volume measurement of each portion was estimated and then these volumes were summed up to get the volume of whole object in XZ plane.

The line scan process is simulated as below:

1. Calculated the maximum and minimum values of the Z-coordinates.
2. Set the resolution value ‘dz’ for line scan, which is the distance between two consecutive lines.
3. Find the border points for each line ‘i’. The distance between these two lines is the diameter.

\[
Diameter(i) = x_2(i) - x_1(i)
\]  
(3.14)

Where

\[x_1(i) \text{ and } x_2(i)\] are the left and right border points respectively.

4. Estimated the volume of the portion between two consecutive lines considering it as a small cylinder using the formula:

\[
V(i) = \pi \times Radius(i)^2 \times dz
\]  
(3.15)

Where \[Radius(i) = Diameter(i) / 2.\]

This process is continued till the last scanning line.
5. Added the volumes of all portions to get the overall volume of the model which can be shown as:

\[ Volume = V(1) + V(2) + V(3) + \ldots + V(k) \]  \hspace{1cm} (3.16)
4

Experiments and Results

4.1 Graphical user Interface:

We designed a user friendly graphical user interface (GUI) for our simulation. The user can select from one of the ideal or non ideal gob models. The user can easily interact with the 3D model through this GUI shown in figure 4.1. The model can be easily rotated along X, Y, Z axes by using the sliders to get the required view. The default rotation of the model is around its centre. Two separate buttons are also available which gives the choice to fix either the top or the bottom of the model. After rotation the projection is obtained by the projection button and then the estimated line scan volume is obtained.

Figure 4.1: Graphical User Interface
4.2 Ideal Gob Model:

We designed a cylinder using data computed from parametric equations 3.1, 3.2, 3.3. The ideal gob is represented by a cylinder in which $N=200$ and $M=200$ points. See figure 4.2.

![Figure 4.2: Ideal Gob Model](image)

4.3 Non Ideal Gob Model:

We designed the non ideal gob models on the base of a curve along the z axis. These models are composed of $N=200$ and $M=200$ points. These circles have different radius according to equation 3.6. The curved shapes as shown in figure 4.3, are obtained by choosing a function which gives us the variance in the radii of the circles. These models have the same memory representation as the ideal gob model as in figure 3.3.
4.4 Rotation:

Some rotations in X, Y, Z are shown in Figure 4.4. Figure 4.4 (a) shows the 0-degree rotation of the ideal gob model along the XYZ axis. Figures 4.4 (b), (c) are the cases when the gob models are rotated by 45 and -45 degree respectively.

Figure 4.4: (a) 0-degree rotation (b) 45-degrees (c) -45-degrees rotation in X,Y,Z
4.5 3d Projection:

A camera view is shown in Figure 4.5 by considering the camera to be on the y axis. As discussed in 3.4 this view is simulated by supposing the optical axis of the camera along the y axis.

Figure 4.5: Camera view of the rotation shown in figure 4.4
(When M = 100 & N = 20)

Figure 4.5 presents the simulated camera view of the gob models. This view contains all the surface points present in the XZ plane. These points of XZ were calculated by multiplying the projection matrix with the rotated matrix points.

4.6 Boundary Points Extraction:

The results of the boundary points extraction is displayed in figure 4.6. The boundary points have been extracted from the above camera view respectively.

Figure 4.6: Boundary point extraction from the camera view in figure 4.5
4.7 Line Scan and Volume Estimation:

The results of line scan procedure for ideal gob model are illustrated in table 4.1 and 4.2. Table 4.1 shows volume estimation of ideal gob for resolution ‘dz’ equal to 0.1, while table 4.2 shows the volume estimation when ‘dz’ is 0.03. The figures 4.7 and 4.8 show the respective graph for each table.

The true volume of ideal gob for 0-degree rotation is \(4\pi = 12.57\).

<table>
<thead>
<tr>
<th>ROTATION</th>
<th>LINE SCAN VOLUME</th>
<th>ROTATION</th>
<th>LINE SCAN VOLUME</th>
<th>ROTATION</th>
<th>LINE SCAN VOLUME</th>
</tr>
</thead>
<tbody>
<tr>
<td>X+,X-,Y</td>
<td></td>
<td>X+,X-,Y,</td>
<td></td>
<td>X+,X-,Y,</td>
<td></td>
</tr>
<tr>
<td>0 0 12.25</td>
<td>0 0 12.25</td>
<td>0 0 12.25</td>
<td>0 0 12.25</td>
<td>0 0 12.25</td>
<td>0 0 12.25</td>
</tr>
<tr>
<td>15 0 13.25</td>
<td>0 15 12.55</td>
<td>15 0 13.25</td>
<td>15 0 13.32</td>
<td>15 0 13.77</td>
<td>15 0 13.84</td>
</tr>
<tr>
<td>30 0 13.00</td>
<td>30 30 13.32</td>
<td>30 0 13.00</td>
<td>30 30 14.17</td>
<td>30 0 13.32</td>
<td>30 0 14.29</td>
</tr>
<tr>
<td>45 0 11.90</td>
<td>45 15 15.08</td>
<td>45 0 11.90</td>
<td>45 15 14.77</td>
<td>45 0 11.90</td>
<td>45 0 14.17</td>
</tr>
<tr>
<td>60 0 9.96</td>
<td>60 60 18.33</td>
<td>60 0 9.96</td>
<td>60 60 14.29</td>
<td>60 0 9.96</td>
<td>60 0 14.29</td>
</tr>
<tr>
<td>75 0 7.37</td>
<td>75 75 14.17</td>
<td>75 0 7.37</td>
<td>75 75 10.51</td>
<td>75 0 7.37</td>
<td>75 0 10.51</td>
</tr>
<tr>
<td>80 0 6.38</td>
<td>80 80 8.45</td>
<td>80 0 6.38</td>
<td>80 80 8.45</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**TABLE 4.1: Rotation in X, Y & respective line scan volume for dz =0.1**

<table>
<thead>
<tr>
<th>ROTATION</th>
<th>LINE SCAN VOLUME</th>
<th>ROTATION</th>
<th>LINE SCAN VOLUME</th>
<th>ROTATION</th>
<th>LINE SCAN VOLUME</th>
</tr>
</thead>
<tbody>
<tr>
<td>X+,X-,Y</td>
<td></td>
<td>X+,X-,Y,</td>
<td></td>
<td>X+,X-,Y,</td>
<td></td>
</tr>
<tr>
<td>0 0 12.53</td>
<td>0 0 12.53</td>
<td>0 0 12.53</td>
<td>0 0 12.53</td>
<td>0 0 12.53</td>
<td>0 0 12.53</td>
</tr>
<tr>
<td>15 0 13.23</td>
<td>0 15 12.50</td>
<td>15 0 13.23</td>
<td>15 0 13.24</td>
<td>15 0 13.32</td>
<td>15 0 13.40</td>
</tr>
<tr>
<td>30 0 13.00</td>
<td>30 30 13.24</td>
<td>30 0 13.00</td>
<td>30 30 14.10</td>
<td>30 0 13.24</td>
<td>30 0 14.06</td>
</tr>
<tr>
<td>45 0 11.90</td>
<td>45 45 14.62</td>
<td>45 0 11.90</td>
<td>45 45 14.62</td>
<td>45 0 11.90</td>
<td>45 0 14.62</td>
</tr>
<tr>
<td>60 0 9.97</td>
<td>60 60 17.97</td>
<td>60 0 9.97</td>
<td>60 60 14.06</td>
<td>60 0 9.97</td>
<td>60 0 14.06</td>
</tr>
<tr>
<td>75 0 7.36</td>
<td>75 75 21.46</td>
<td>75 0 7.36</td>
<td>75 75 21.46</td>
<td>75 0 7.36</td>
<td>75 0 21.46</td>
</tr>
<tr>
<td>80 0 6.37</td>
<td>80 80 22.60</td>
<td>80 0 6.37</td>
<td>80 80 22.60</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**TABLE 4.2: Rotation in X, Y and respective line scan volume for dz =0.03**

Table 4.1 and 4.2 shows that the accuracy of line scan procedure increases as dz decreases.
Table 4.1, 4.2 and figure 4.7, 4.8 show the effect of rotation on volume estimation for ideal gob. When the gob is rotated along X-axis only, volume is increased initially and then decrease as rotation angle increases. When the gob is rotated along Y-axis only, volume is decreased initially and then increases as rotation angle increases. In case of
rotation along both axes, volume is increased initially and then decreases as rotation angle increases.

The results of non ideal gob model 1 for different resolutions are shown in table 4.3 and 4.4. Their respective graphs are in 4.9 and 4.10. While results of non ideal gob model 2 for different resolutions are shown in table 4.5 and 4.6. Their respective graphs are in figures 4.11 and 4.12.

<table>
<thead>
<tr>
<th>ROTATION</th>
<th>LINE SCAN VOLUME</th>
<th>ROTATION</th>
<th>LINE SCAN VOLUME</th>
<th>ROTATION</th>
<th>LINE SCAN VOLUME</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 0</td>
<td>6.16</td>
<td>0 0</td>
<td>6.16</td>
<td>0 0</td>
<td>5.92</td>
</tr>
<tr>
<td>15 0</td>
<td>6.60</td>
<td>0 15</td>
<td>6.14</td>
<td>15 15</td>
<td>6.66</td>
</tr>
<tr>
<td>30 0</td>
<td>6.63</td>
<td>0 30</td>
<td>6.44</td>
<td>30 30</td>
<td>7.11</td>
</tr>
<tr>
<td>45 0</td>
<td>6.19</td>
<td>0 45</td>
<td>7.32</td>
<td>45 45</td>
<td>7.57</td>
</tr>
<tr>
<td>60 0</td>
<td>5.33</td>
<td>0 60</td>
<td>9.46</td>
<td>60 60</td>
<td>8.32</td>
</tr>
<tr>
<td>75 0</td>
<td>4.12</td>
<td>0 75</td>
<td>13.83</td>
<td>75 75</td>
<td>7.1</td>
</tr>
<tr>
<td>80 0</td>
<td>3.63</td>
<td>0 80</td>
<td>16.61</td>
<td>80 80</td>
<td>5.6</td>
</tr>
</tbody>
</table>

**TABLE 4.3: Rotation in X, Y & respective line scan volume for dz =0.1**

<table>
<thead>
<tr>
<th>ROTATION</th>
<th>LINE SCAN VOLUME</th>
<th>ROTATION</th>
<th>LINE SCAN VOLUME</th>
<th>ROTATION</th>
<th>LINE SCAN VOLUME</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 0</td>
<td>6.14</td>
<td>0 0</td>
<td>6.14</td>
<td>0 0</td>
<td>6.1414</td>
</tr>
<tr>
<td>15 0</td>
<td>6.62</td>
<td>0 15</td>
<td>6.0763</td>
<td>15 15</td>
<td>6.6531</td>
</tr>
<tr>
<td>30 0</td>
<td>6.63</td>
<td>0 30</td>
<td>6.3857</td>
<td>30 30</td>
<td>7.0723</td>
</tr>
<tr>
<td>45 0</td>
<td>6.19</td>
<td>0 45</td>
<td>7.2408</td>
<td>45 45</td>
<td>7.4801</td>
</tr>
<tr>
<td>60 0</td>
<td>5.33</td>
<td>0 60</td>
<td>9.3636</td>
<td>60 60</td>
<td>8.0417</td>
</tr>
<tr>
<td>75 0</td>
<td>4.0951</td>
<td>0 75</td>
<td>13.4667</td>
<td>75 75</td>
<td>6.8342</td>
</tr>
<tr>
<td>80 0</td>
<td>3.6228</td>
<td>0 80</td>
<td>14.78</td>
<td>80 80</td>
<td>5.4936</td>
</tr>
</tbody>
</table>

**TABLE 4.4: Rotation in X, Y & respective line scan volume for dz =0.03**
FIGUE 4.9: Rotation Vs Line Scan Volume for table 4.3

FIGUE 4.10: Rotation Vs Line Scan Volume for table 4.4
### Table 4.5: Rotation in X, Y & respective line scan volume for $d_z = 0.1$

<table>
<thead>
<tr>
<th>ROTATION</th>
<th>LINE SCAN VOLUME</th>
<th>ROTATION</th>
<th>LINE SCAN VOLUME</th>
<th>ROTATION</th>
<th>LINE SCAN VOLUME</th>
</tr>
</thead>
<tbody>
<tr>
<td>$X+,X-$</td>
<td>$Y$</td>
<td>$X$</td>
<td>$Y+,Y-$</td>
<td>$X+,X-$</td>
<td>$Y+,Y-$</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>6.51</td>
<td>0</td>
<td>6.51</td>
<td>0</td>
</tr>
<tr>
<td>15</td>
<td>0</td>
<td>6.41</td>
<td>0</td>
<td>6.72</td>
<td>15</td>
</tr>
<tr>
<td>30</td>
<td>0</td>
<td>5.84</td>
<td>0</td>
<td>7.31</td>
<td>30</td>
</tr>
<tr>
<td>45</td>
<td>0</td>
<td>4.90</td>
<td>0</td>
<td>8.51</td>
<td>45</td>
</tr>
<tr>
<td>60</td>
<td>0</td>
<td>3.91</td>
<td>0</td>
<td>10.60</td>
<td>60</td>
</tr>
<tr>
<td>75</td>
<td>0</td>
<td>4.17</td>
<td>0</td>
<td>13.92</td>
<td>75</td>
</tr>
<tr>
<td>80</td>
<td>0</td>
<td>4.17</td>
<td>0</td>
<td>13.92</td>
<td>80</td>
</tr>
</tbody>
</table>

### Table 4.6: Rotation in X, Y & respective line scan volume for $d_z = 0.03$

<table>
<thead>
<tr>
<th>ROTATION</th>
<th>LINE SCAN VOLUME</th>
<th>ROTATION</th>
<th>LINE SCAN VOLUME</th>
<th>ROTATION</th>
<th>LINE SCAN VOLUME</th>
</tr>
</thead>
<tbody>
<tr>
<td>$X+,X-$</td>
<td>$Y$</td>
<td>$X$</td>
<td>$Y+,Y-$</td>
<td>$X+,X-$</td>
<td>$Y+,Y-$</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>6.54</td>
<td>0</td>
<td>6.54</td>
<td>0</td>
</tr>
<tr>
<td>15</td>
<td>0</td>
<td>6.42</td>
<td>0</td>
<td>6.70</td>
<td>15</td>
</tr>
<tr>
<td>30</td>
<td>0</td>
<td>5.86</td>
<td>0</td>
<td>7.27</td>
<td>30</td>
</tr>
<tr>
<td>45</td>
<td>0</td>
<td>4.91</td>
<td>0</td>
<td>8.48</td>
<td>45</td>
</tr>
<tr>
<td>60</td>
<td>0</td>
<td>3.93</td>
<td>0</td>
<td>10.55</td>
<td>60</td>
</tr>
<tr>
<td>75</td>
<td>0</td>
<td>4.17</td>
<td>0</td>
<td>13.20</td>
<td>75</td>
</tr>
<tr>
<td>80</td>
<td>0</td>
<td>4.17</td>
<td>0</td>
<td>13.72</td>
<td>80</td>
</tr>
</tbody>
</table>

### Figure 4.11: Rotation Vs Line Scan Volume for table 4.5

![Graph showing rotation vs line scan volume for non-ideal model 2.](image)
FIGURE 4.12: Rotation Vs Line Scan Volume for table 4.6

The figures of line scan of some of these rotations are given in the Appendix.
Summary and Conclusions

A solution to an industrial problem is provided by using 3D graphical models and mathematical calculations.

In the first step representation was designed to imitate the gob models. The next step comprised of the rotation of the model. The rotated object was projected to a 2D plane. After that the boundary points were extracted to perform the required line scan procedure.

The final stage consists of the simulation of the volume estimation of the gob model through a line scan procedure. The accuracy of volume estimation can be enhanced by increasing the resolution ‘d_z’ in the line scan procedure.

Improvements can be made in this work by adding more choices in the GUI and hence increasing the level of flexibility of the simulator. For example:

- An interactive way to change the shape for non ideal gob can be added.
- Non ideal gobs which are not symmetric around the Z axis can be added.
References


[9] Introduction to 3D Studio Max Lectures by Jeffrey Abouaf, Transcribed and Illustrated by Blue Bactol 2000.


[16] MATLAB Product Documentation
Appendix

The following figures show how the line scanning is done for different rotation angles on ideal and non-ideal gob models.

Figure A.1: Ideal gob model line scanning for different rotation in X
Figure A.2: Ideal gob model Line scanning for different rotation in Y
Figure A.3: Ideal gob model Line scanning for different rotation in X & Y
Figure A.4: Non Ideal gob model 1 Line scanning for different rotation in X

Figure A.5: Non Ideal gob model 1 Line scanning for different rotation in Y
Figure A.6: Non Ideal gob model 1 Line scanning for different rotation in X & Y
Figure A.7: Non Ideal gob model 2 Line scanning for different rotation in X

Figure A.8: Non Ideal gob model 2 Line scanning for different rotation in Y
Figure A.9: Non Ideal gob model 2 Line scanning for different rotation in X & Y