

**Interactive visualization of Hyperbolic geometry using the
Weierstrass model**

A Thesis

submitted to the faculty of the Graduate School
of University of Minnesota

by

Kedar Bhumkar

In partial fulfillment of the requirements
for the degree of
Master of Science

July 2006

University of Minnesota

This is to certify that I have examined this copy of the Master's thesis

by

Kedar Bhumkar

and have found that it is complete and satisfactory in all respects ,
and that any and all revisions required by the final
examining committee have been made

Dr Douglas Dunham

Name of the faculty advisor

Signature of faculty advisor

Date

GRADUATE SCHOOL

Acknowledgements

I would like to thank to Dr. Douglas Dunham for his support, encouragement and directions during the course of the thesis. I am also thankful to Dr. Gary Shute and Dr. Joseph Gallian for being on my committee.

I am grateful to the Department of Computer Science, University of Minnesota Duluth for their cooperation throughout my graduate studies.

Abstract

Euclidean geometry, as taught in schools, is based on five postulates. Efforts to derive the fifth postulate from the other four led to the discovery of hyperbolic geometry about 150 years ago. Hyperbolic geometry is a subset of a large class of geometries called non-Euclidean geometries. Euclidean geometry is usually the most convenient to describe the physical world surrounding us. On the other hand hyperbolic geometry describes space-time more conveniently. Albert Einstein's Special Theory of Relativity is based on hyperbolic geometry. However hyperbolic geometry is difficult to visualize as many of its theorems are contradictory to similar theorems of Euclidean geometry which are very familiar to us. For example in hyperbolic geometry, the sum of angles in a triangle is less than 180 degrees. Rectangles do not exist, and the distance between nonintersecting lines increases without limit, etc.

These inherent contradictions to Euclidean geometry force us to visualize hyperbolic geometry by constructing geometric models in Euclidean space. Viewing one such model – the Weierstrass model – from specific coordinates leads us to other models. The primary goal of this thesis is to provide an interactive and animated visualization of the Weierstrass model. Such visualization can provide us with a better insight into hyperbolic geometry and its models.

CONTENTS

1.	INTRODUCTION.....	1
2.	DISCOVERY OF HYPERBOLIC GEOMETRY	2
	2.1 EUCLIDEAN GEOMETRY	2
	2.2 PARALLEL POSTULATE CONTROVERSY	3
	2.3 THE QUEST FOR EUCLID V	3
	2.4 ACCEPTANCE OF NON-EUCLIDEAN GEOMETRIES	5
3.	FUNDAMENTALS OF HYPERBOLIC GEOMETRY	7
	3.1 WHAT IS HYPERBOLIC GEOMETRY?	7
	3.2 VISUALIZING HYPERBOLIC SPACE.....	8
	3.3 CROCHET MODELS	11
	3.4 IMPORTANCE OF HYPERBOLIC GEOMETRY	14
4.	MODELS OF HYPERBOLIC GEOMETRY	16
	4.1 MATHEMATICAL MODEL.....	16
	4.2 HYPERBOLIC MODELS	17
	4.3 ISOMORPHISM.....	26
5.	DEVELOPMENT MECHANICS.....	29
	5.1 IMPLEMENTATION CHOICE: JAVA 3D.....	29
	5.2 JAVA 3D BASICS	30
	5.3 CREATING THE 3D WEIERSTRASS MODEL.....	32
	5.4 MAPPING THE TEXTURE	34
	5.5 VIEWING THE MODELS	35
	5.6 PERSPECTIVE PROJECTION.....	36
	5.7 PARALLEL PROJECTION	37
6.	HYPERVIEW3D	38
	6.1 SPLIT SCREEN GUI.....	38
	6.2 MODES OF OPERATION	41
	6.3 DYNAMIC TEXTURE MAPPING	44
	6.4 CONTROL PANEL OPTIONS	45

7.	OUTPUT AND RESULTS	48
7.1	WIREFRAME MODE ON	48
7.2	WIREFRAME MODE OFF	52
8.	PROPOSED ENHANCEMENTS.....	56
8.1	INTEGRATION OF DR. DUNHAM'S PROGRAM	56
8.2	RUN-TIME PATTERN GENERATION	56
8.3	CUSTOMIZED ANIMATION	56
8.4	GEOMETRIC CONSTRUCT DRAWING	56
	APPENDIX	57
	REFERENCES	60

LIST OF FIGURES

FIGURE 3.1: HYPERBOLIC AXIOM	7
FIGURE 3.2: HORSE SADDLE DEPICTING A NEGATIVE CURVATURE [11].....	9
FIGURE 3.3: LETTUCE LEAF DEPICTING HYPERBOLIC SURFACE	10
FIGURE 3.4: MARINE FLATWORM WITH HYPERBOLIC RUFFLES	10
FIGURE 3.5: PAPER MODEL OF HYPERBOLIC SURFACE	10
FIGURE 3.6: HYPERBOLIC SOCCER BALL.....	10
FIGURE 3.7: A CROCHET HYPERBOLIC PLANE	11
FIGURE 3.8: VIOLATION OF EUCLID V.....	12
FIGURE 3.9: ANGLE SUM LESS THEN 180.....	12
FIGURE 3.10: HYPERBOLIC EQUIVALENT OF A CONE.....	13
FIGURE 3.11: GEOMETRY OF THE UNIVERSE	15
FIGURE 4.1: FINITE DISC OF THE KLEIN MODEL	17
FIGURE 4.2: A LINE IN THE KLEIN MODEL	18
FIGURE 4.3: PARALLEL LINES IN KLEIN MODEL	19
FIGURE 4.4: LENGTH IN KLEIN MODEL	20
FIGURE 4.5: A LINE IN THE POINCARÉ MODEL	21
FIGURE 4.6: PARALLEL LINES IN POINCARÉ MODEL.....	22
FIGURE 4.7: A LINE IN THE GANS MODEL.....	23
FIGURE 4.8: HYPERBOLOID OF 2 SHEET	24
FIGURE 4.9: ONE SHEET OF A HYPERBOLOID	24
FIGURE 4.10: ASYMPTOTIC CONE	25
FIGURE 4.11: A LINE IN THE WEIERSTRASS MODEL.....	26
FIGURE 5.1: JAVA3D API	30
FIGURE 5.2: HYPERVIEW3D SCENEGRAPH	31
FIGURE 5.3: 3D WEIERSTRASS MODEL BUILDING STEPS	33
FIGURE 5.4: M.C. ESCHER'S FAMOUS CIRCLE LIMIT I.....	34

FIGURE 5.5: PROJECTION TYPES	36
FIGURE 6.1: HYPERVIEW3D SPLIT SCREEN GUI.....	38
FIGURE 6.2: SIDE VIEW.....	39
FIGURE 6.3: MODEL VIEW	40
FIGURE 6.4: CONTROL PANEL	40
FIGURE 6.5: COMPARISON MODE ACTIVATION BUTTONS.....	41
FIGURE 6.6: COMPARISON MODE DISPLAY	42
FIGURE 6.7: WIREFRAME MODE ACTIVATION BUTTON.....	42
FIGURE 6.8: WIREFRAME MODE DISPLAY.....	43
FIGURE 6.9: DYNAMIC TEXTURE MAPPING.....	44
FIGURE 6.10: CAMERA IN MOTION	45
FIGURE 6.11: BOUNDING CONE	46
FIGURE 6.12: PRECISE POSITION USING TEXT BOX INPUTS.....	47
FIGURE 6.13: SLIDER	47
FIGURE 7.1: INPUT FOR WIREFRAME MODE	48
FIGURE 7.2: POINCARÉ MODEL VIEW	49
FIGURE 7.3: KLEIN MODEL VIEW	50
FIGURE 7.4: GANS MODEL VIEW	51
FIGURE 7.5: ESCHER'S CIRCLE LIMIT I: INPUT IMAGE (PRE-EXISTING).....	52
FIGURE 7.6: ESCHER'S CIRCLE LIMIT I: POINCARÉ MODEL (HYPERVIEW3D).....	53
FIGURE 7.7: ESCHER'S CIRCLE LIMIT I: KLEIN MODEL (HYPERVIEW3D).....	54
FIGURE 7.8: ESCHER'S CIRCLE LIMIT I: GANS MODEL (HYPERVIEW3D).....	55

1. Introduction

Euclidean geometry, as taught in schools, is based on five postulates. Efforts to derive the fifth postulate from the other four led to the discovery of hyperbolic geometry almost 200 years ago. Hyperbolic geometry is a subset of a large class of geometries called non-Euclidean geometries. Euclidean geometry is usually the most convenient to describe the physical world surrounding us. On the other hand hyperbolic geometry describes space-time more conveniently. Albert Einstein's Special Theory of Relativity is based on hyperbolic geometry.

However hyperbolic geometry is difficult to visualize as many of its theorems are contradictory to similar theorems of Euclidean geometry which are very familiar to us. For example in hyperbolic geometry, the sum of angles in a triangle is less than 180 degrees. Rectangles do not exist, and the distance between nonintersecting lines increases without bound, etc.

These inherent contradictions to Euclidean geometry force us to visualize hyperbolic geometry by constructing geometric models in Euclidean 3-space. Viewing one such model – the Weierstrass model – from specific coordinates leads us to other models. The primary goal of this thesis is to provide an interactive and animated visualization of the Weierstrass model. Such visualization can provide us with a better insight into hyperbolic geometry.

Subsequent chapters will deal with the basic principles of hyperbolic geometry, its models, Java 3D, construction of the Weierstrass model, information about the different views, and finally the results.

2. Discovery of Hyperbolic Geometry

2.1 Euclidean Geometry

Euclidean geometry is the geometry taught to us in our schools and practiced in day to day life. It was first proposed by Euclid of Alexandria, a Greek Mathematician in 3rd century BCE, in his seminal work, The Elements. The Elements is an epic mathematical and geometric work comprised of 13 books. It is considered to be one of the most important books ever written in the history of mankind. In terms of number of editions it is second only to Bible with over a thousand editions and innumerable reproductions [2].

The Elements is an example of an axiomatic deductive system. It starts by defining the basic terms, and further concepts (theorems) are developed using these terms and principles of logic. Book I contains 23 definitions (point, line etc.), five postulates and five axioms.

Postulates are statements that are accepted without any demonstration or proof. The following are Euclid's 5 postulates described in the Elements in an informal form [2]

- First Postulate
Two distinct points determine a unique line.
- Second Postulate
Any segment AB can be extended by a segment BE congruent to a given segment CD
- Third Postulate
For two distinct points O and A there exists a circle with center O and radius equal to OA.
- Fourth Postulate
All right angles are congruent to each other.

- Fifth Postulate (Parallel Postulate or Euclid V)

Given a line L and an external point P not on L there exists a unique line m passing through P and parallel to L .

2.2 Parallel Postulate Controversy

Mathematicians including Euclid himself questioned the need for the parallel postulate as soon as it was proposed. Euclid refrained from using this postulate until his 29th proposition by just making use of the first four postulates to derive the first 28 propositions [2]. One of the reasons the parallel postulate was questioned was because the other 4 seemed very obvious and could be verified by observation or doing experiments (with a compass and an unmarked straight edge).

It was not possible to consider the parallel postulate as self-evident truth just by observation or measurement as it involved the concept of lines stretching to infinity. Mathematicians then decided to prove the veracity of the parallel postulate by one way or the other. Thus began one of the greatest quests in mathematical history. It was to continue for 2000 years and destined to come to a shocking conclusion.

2.3 The Quest for Euclid V

The first attempt to establish the parallel postulate was made by Ptolemy (90 – 168 BCE). It was followed by Proclus (412 – 485). The next important attempt was made by Persian mathematician Nasiraddin (1201 – 1274), who was followed by John Wallis (1616 – 1703). All of them assumed that the 5th postulate was not independent. Hence they tried to derive it from the first four. However this process proved to be fruitless because they either made some unjustified assumptions or used circular reasoning.

Italian priest and mathematician Giovanni Saccheri (1667 – 1733) decided to vindicate Euclid by establishing the authenticity of Euclid V. He wrote a book “Euclides ab omni

naevo vindicatus” (Euclid Freed of Every Flaw) where he tried to derive the parallel postulate using the Reductio ad absurdum method [2]. Although he was not able to contradict his opposite assumption, he refused to believe his results and falsely tried to convince himself that he had indeed arrived at a contradiction. However, his work was more expansive and elaborate and he managed to derive many theorems and results by just using the first four postulates. His work paved the way to the development of *absolute geometry* also known as the neutral geometry. This is a geometry in which all theorems and results are derived using just the first four postulates of Euclid. An attempt on a similar line was later made by Lambert (1728 - 1777). However, it left many questions unanswered. With the advent of the 19th century, the quest for Euclid V entered its final showdown and 3 people were to be its nemesis.

Carl Friedrich Gauss (1777 – 1855), one of history’s greatest mathematician, investigated the parallel postulate problem for some time. From his private diary and letters to his colleagues it is now known that Gauss had indeed discovered the existence of a geometry that was completely different than that of Euclid’s. In one of his letters he describes clearly the existence of such a geometry, his inability to find a contradiction to the negation of Euclid V, the theorems and results obtained, and their counter-intuitiveness with respect to the Euclidean geometry. However Gauss did not publish any of his findings. Generally two reasons are attributed to his reluctance. Gauss was reluctant to get entangled in any controversy with the metaphysicians – followers of Immanuel Kant, one of the most influential thinkers of that time. Kant believed that Euclidean geometry was absolute and there was no other geometry besides this, a view held by many people at that time. Gauss’s discovery would have refuted this belief. Another reason attributed to Gauss’s reluctance was that he was a perfectionist working on several things at one time. His range of work encompassed fields such as astronomy, geodesy, physics and mathematics. Gauss believed in publishing only completed works that were above any criticism. This belief is reflected by his motto on his seal ‘*pauca sed matura*’ (few but ripe) [2]. Despite the fact the Gauss did not publicize any results he is equally recognized for solving the Euclid V.

Janos Bolyai (1802 – 1860) a Hungarian mathematician published his discovery of non-Euclidean geometry as a 26 page Appendix in his father's book Tentamen in 1831. His father knew Gauss well and sent Gauss a letter describing his son's discovery. Gauss in his reply informed the Bolyai's of his own research in this area. The response from Gauss came as a shock to Janos who thought Gauss was stealing his ideas and declaring them as his own. Janos fell into a mental depression and never recovered. He did not publish anything after this episode.

Nikolai Lobachevsky (1792 – 1856) a prominent Russian mathematician is regarded as the first person to discover Non-Euclidean geometry by virtue of its publication date. He first published his results in a Russian journal in 1829. However this revolutionary concept went unnoticed as it was a Russian journal and the Russian mathematicians were severely critical of the concept. In 1840 Lobachevsky published his discovery in German which came to the attention of Gauss. Gauss upheld the views of Lobachevsky and also indicated his contribution in this field.

Gauss, Bolyai and Lobachevsky all tried to find a contradiction to the negation of Euler V but were unable to find one. This was similar to the approach adopted by others before them. However their predecessors were unable to comprehend the fact that Euclid V could be invalid in certain situations. Gauss, Bolyai and Lobachevsky were able to demonstrate situations under which Euclid V was invalid and developed an altogether new field of non-Euclidean geometry.

2.4 Acceptance of Non-Euclidean Geometries

After the death of Gauss (1855) his correspondence over non-Euclidean geometry was published. From this time onwards people started taking interest in the field of non-Euclidean geometry. They rediscovered the publications of Janos Bolyai and Lobachevsky. The method of contradiction allowed for the assumption of two negations to the parallel postulate. The first, made by Lobachevsky (there are at least two lines passing through the external point not on a given line), led to the development of hyperbolic geometry. The second assumption

made by German mathematician Bernard Riemann (there are no lines passing through the external point not on a given line) led to the development of spherical geometry. Prominent mathematicians such as Beltrami, Klein and Poincare took up the work of establishing the field of non-Euclidean geometries.

In 1868 a moment arrived which put curtains on the great quest for Euler V, a quest which took 2000 years and the work of many prominent mathematicians. Beltrami proved the consistency of hyperbolic geometry relative to Euclidean geometry. Thus if hyperbolic geometry contains any contradiction then so would Euclidean geometry. This implied that no proof for the parallel postulate exists.

Euler V was solved.

3. Fundamentals of Hyperbolic Geometry

3.1 What is Hyperbolic Geometry?

As discussed above hyperbolic Geometry is one of the non-Euclidean geometries discovered during the quest for Euclid V. Hyperbolic geometry is the geometry obtained by considering only the first four postulates of Euclid and replacing the parallel postulate by its negation, called the hyperbolic axiom. The hyperbolic axiom states that there exists a line l and an external point not lying on this line such that there are at least 2 lines passing through this point and parallel to this line l [2].

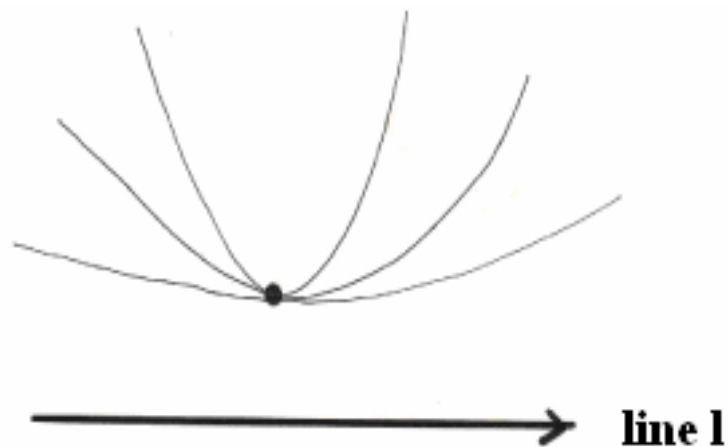


Figure 3.1: Hyperbolic axiom

The following are some important properties of hyperbolic geometry

- All triangles have angle sum less than 180.

- Rectangles do not exist.
- If two triangles are similar they are congruent.
- All convex quadrilaterals have angle sum less than 360.
- The hyperbolic axiom holds for all lines.

The above properties seem counter-intuitive to us. This is because we are conditioned from our childhood to think in terms of Euclidean geometry. Hence in a Euclidean plane our notion of sum of angles in a triangle or parallel lines holds and seems natural to us. But these notions will not hold if we are dealing with non-Euclidean planes, specifically here the hyperbolic plane. Thus this brings us to the question of how hyperbolic geometry appears to an observer. We must remember that when we are viewing any hyperbolic objects or planes we (the observer) are situated in Euclidean world.

3.2 Visualizing Hyperbolic Space

Euclidean planes are flat. They have a zero curvature. On the other hand, non-Euclidean geometric surfaces have non-zero curvature. A sphere has a positive curvature. No matter where you move from a point on the sphere it curves the same way in every direction. This is positive curvature. On the other hand, hyperbolic surfaces are characterized by negative curvature. An example of negative curvature is a horse saddle. If you start at the front end of the saddle and start going backward, the saddle curves down then curves up again. Similarly if you go from side to side, it curves up then curves down. So there are opposite curvatures going in different directions.

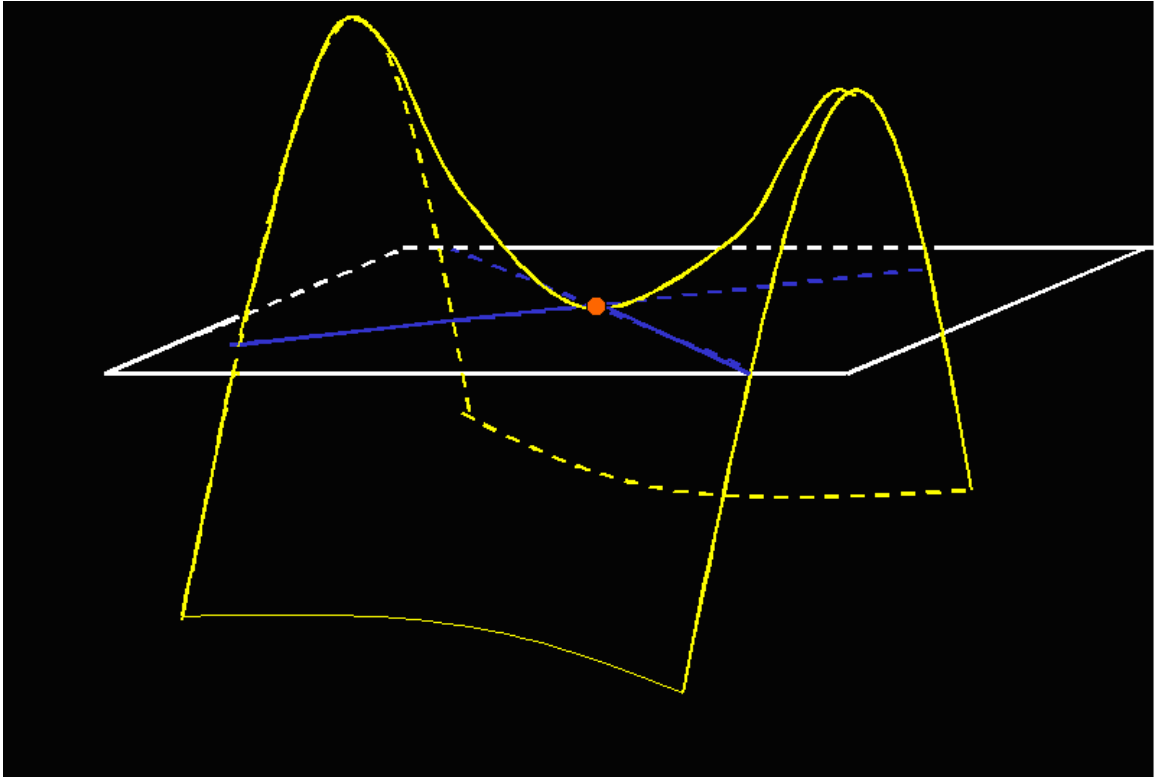


Figure 3.2: Horse Saddle depicting a negative curvature [11]

Technically if a surface has negative curvature at a point, a plane tangent to the surface at this point would lie on both sides of the surface.

When we say “line,” we immediately have a picture of the flat Euclidean plane with a straight line on it. If we imagine now this flat plane is replaced by the horse saddle, then the line will now look at some locations like a curve. Thus to visualize hyperbolic geometry and its properties we ought to discard the flat plane image of the Euclidean plane and think in terms of a shape like that of the saddle.

Although we don’t see hyperbolic surfaces around us, often nevertheless nature does possess a few. For example, lettuce leaves and marine flatworms exhibit hyperbolic geometry.



Figure 3.3: Lettuce leaf depicting hyperbolic surface



Figure 3.4: Marine flatworm with hyperbolic ruffles



Figure 3.5: Paper Model of hyperbolic surface

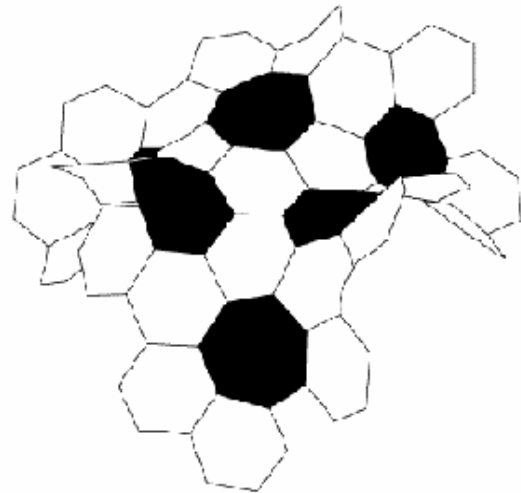


Figure 3.6: Hyperbolic Soccer ball

3.3 Crochet Models

Cornell University professor William Thurston had designed a paper model of the hyperbolic plane. His idea was that if we move away from a point in hyperbolic plane, the space around that point expands exponentially. However his models were very fragile and difficult to make. In 1997 another professor from Cornell, Dr. Diana Tiamina, developed a new technique of creating hyperbolic surface models [8]. She used crochets. Her models (seen below) made it possible to actually feel hyperbolic space and see many of its properties in action.



Figure 3.7: A crochet hyperbolic plane



Figure 3.8: Violation of Euclid V



Figure 3.9: Angle sum less than 180

In the above figures, the lines sewn on the crochet depict some of the properties of hyperbolic space.



Figure 3.10: Hyperbolic equivalent of a cone

3.4 Importance of Hyperbolic Geometry

The discovery of hyperbolic geometry has led to applications in many fields. Most of the applications, although theoretical, are nevertheless fundamental to our understanding of nature and our existence. Hyperbolic geometry has found applications in fields of mathematics, physics, and engineering. It has been found that the visual information seen through our eyes and interpreted by our brain is better explained using hyperbolic geometry [9]. Hyperbolic geometry also has practical aspects such as orbit prediction of objects within intense gravitational fields. Einstein's invented his special theory of relativity based on hyperbolic geometry [9].

Perhaps the most important application of hyperbolic geometry is to find the fate of the universe. This is because the shape of the universe will determine its future. Although recent measurements have given evidence that the geometry of the observable universe is Euclidean, nothing can be concluded about the overall topology of the universe at this time. Thus, there are three primary possibilities: Euclidean, hyperbolic or spherical topologies.

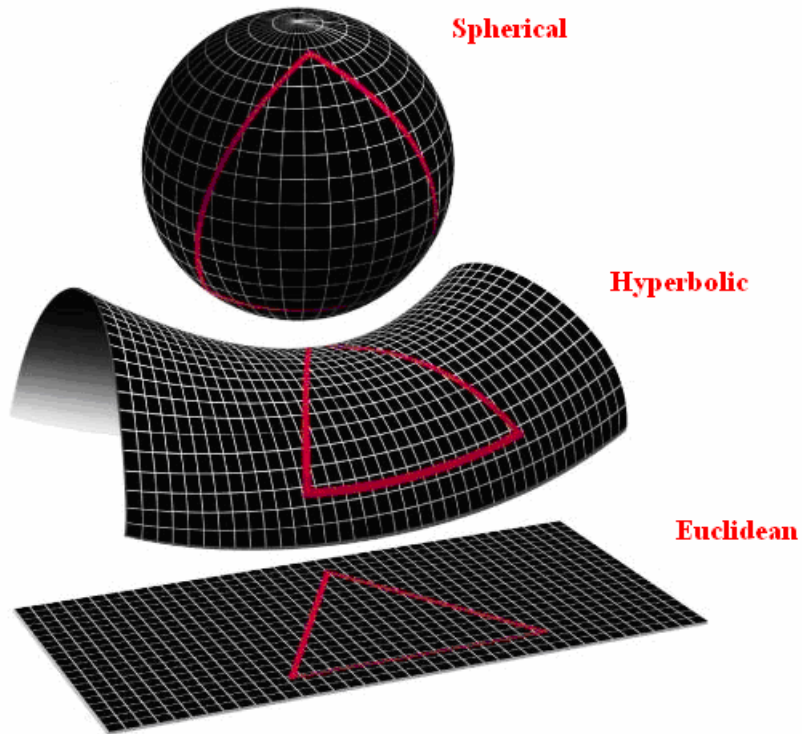


Figure 3.11: Geometry of the Universe

It is speculated that if the geometry of the universe is spherical then it is of finite size. It will continue expanding up to a certain time and then would eventually contract. If the geometry is either hyperbolic or Euclidean then it is of infinite size. If it is hyperbolic it will continue expanding forever at an increasing rate. However, if it is Euclidean then it will expand forever but the rate of expansion will decrease with the passage of time.

Current data from NASA's Wilkinson Microwave Anisotropy Probe (WMAP) satellite indicates a Euclidean universe. However the scientific community is not certain about the findings. Future NASA satellite missions similar to WMAP, the Clover and Plank Surveyor will hopefully furnish enough evidence to support or reject the claim [10].

4. Models of Hyperbolic Geometry

At the end of Chapter 2, it was mentioned that Beltrami showed the validity of hyperbolic geometry by proving it to be consistent with Euclidean geometry. He did this using a mathematical model of hyperbolic geometry.

4.1 Mathematical model

A mathematical model can be considered as a system where the axioms of a system become true statements. Consider for example the statement that for two distinct points a unique line passes through them. When we read this statement we visualize it in our mind with two dots and a line between them. Assuming this representation, the above statement seems correct. We can consider the dots and line segments to be a model for this statement if it were the only axiom of the system.

In general, to define a model we assign an interpretation to the ‘undefined terms’ of a mathematical system. Undefined terms could mean terms such as point, line, “is between”, etc. in case of geometry. Then we check to see if the axioms of the system are true under this interpretation. If they are found to be correct, then our interpretation becomes the model.

Theorems are logical statements deduced using axioms. One remarkable property of models is that all theorems for a system are true in a model. This is because all axioms are true in the model and theorems are logical deductions from axioms.

Models demonstrate whether a statement can be derived from the axioms. Models also establish consistency of any mathematical axiomatic system. Models made it possible to show the equiconsistency of hyperbolic and Euclidean geometry, which put an end to the quest for Euler V.

4.2 Hyperbolic Models

There are 2 types of hyperbolic models

- Finite
- Infinite

Finite models are those which can be represented in a finite portion of the Euclidean plane such as a disc. Examples of finite models are

- Klein model
- Poincare model

Infinite models are those which are represented in the entire Euclidean plane which is infinite. Examples of infinite model are

- Weierstrass model
- Minkowski model
- Gans model

Klein Model

The Klein model was first proposed by the German mathematician Felix Klein. The entire hyperbolic space is represented in a finite disc in this model [2].

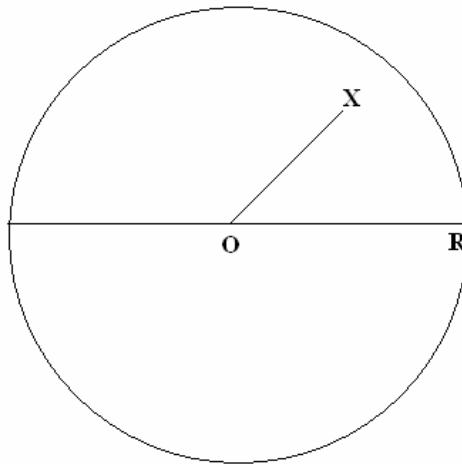


Figure 4.1: Finite disc of the Klein model

If O is the center of this circle and OR is its radius then all points X such that $OX < OR$ determine the interior of this circle. In this model the points in this interior represent the points on the hyperbolic plane.

As discussed above any model requires the interpretation of the undefined terms. Following are the interpretations for the fundamental undefined terms in this model.

- **Point**

It has the same meaning as in Euclidean geometry (but only within the circle).

- **Line**

A line in the hyperbolic plane is defined as an open chord in this model. An open chord is any chord of the circle excluding its endpoints. For example, AB is an open chord if we consider all points between A and B excluding A and B .

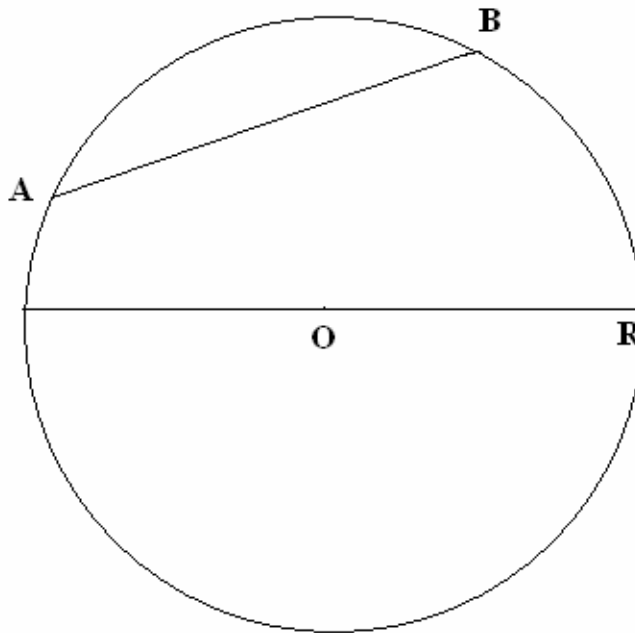


Figure 4.2: A line in the Klein model

- **Lies On**

It has the same meaning as in Euclidean geometry.

- **Between**

It has the same meaning as in Euclidean geometry.

Parallel Lines

The following figure shows the violation of Euclid V. On the other hand it shows how the hyperbolic axiom holds true in this model.

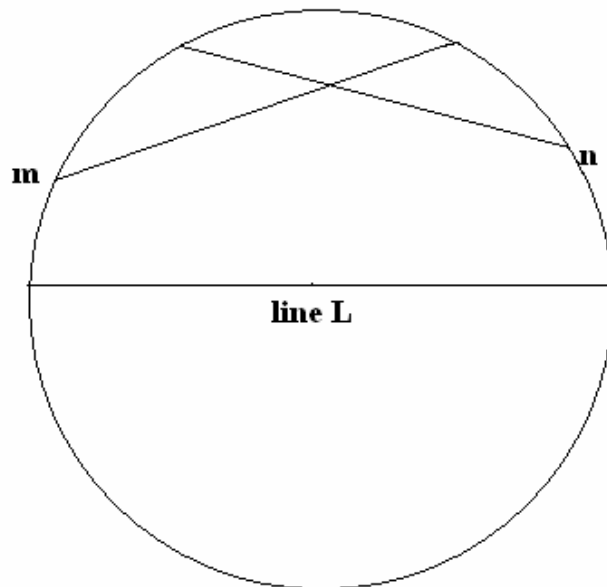


Figure 4.3: Parallel lines in Klein model

Hyperbolic Length

The measure of length is not the same as Euclidean length. If it were, then all lengths would have to be less than the radius of the disc, but in reality hyperbolic lengths are infinite.

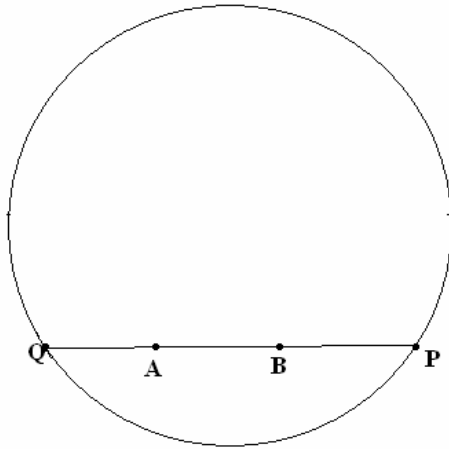


Figure 4.4: Length in Klein model

The following formula is used to compute the length between points A and B.

$$\text{Distance}(A,B) = 1/2 | \log (AP \cdot BQ / BP \cdot AQ) |$$

It can be seen that if point B approaches P then distance(A,B) approaches infinity making it possible to represent arbitrary lengths. Here AP, BQ, BP, and AQ are Euclidean distances.

Hyperbolic Angles

One potential drawback of this model is that the angles in the hyperbolic space get distorted when represented in the Klein disc. This model is therefore called non-conformal or angle distorting.

Poincare Model

This model was proposed by French mathematician Henri Poincare. Like Klein's model, this model too is finite and represents the entire hyperbolic space within the interior of the circle [2].

The undefined terms are interpreted in the following manner.

- **Point**

It has the same meaning as in Euclidean geometry (within the circle).

- **Line**

A line in hyperbolic plane is defined in two possible ways. One kind of line is represented by open diameters. Open diameters are open chords (defined in Klein model) passing through the center of the circle. Secondly, lines are also represented by open arcs of circle that are orthogonal to the Poincare disc as shown below. The angle is computed by measuring the angle between the tangents of the lines.

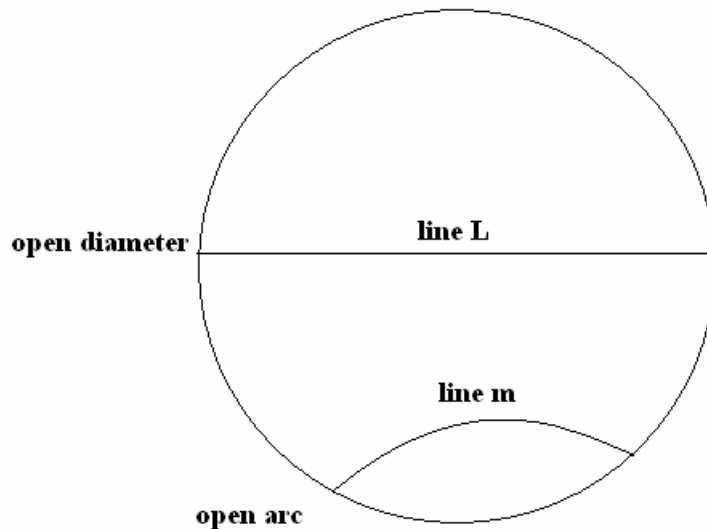


Figure 4.5: A line in the Poincare model

This is fundamentally different from the Klein model, since lines are straight in the Klein model and can never be arcs.

- **Lies On**

It has the same meaning as in Euclidean geometry.

- **Between**

It has the same meaning as in Euclidean geometry.

Parallel Lines

The figure below shows how Euclid V is violated. It shows both types of lines: open arcs and an open diameter.

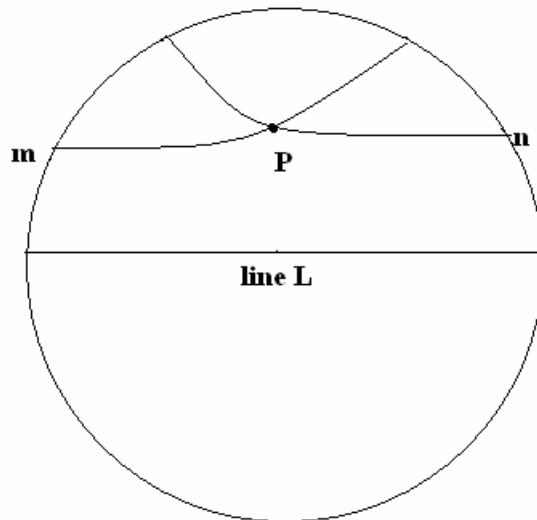


Figure 4.6: Parallel lines in Poincare model

Hyperbolic length

The measure of length is twice of the length in the Klein model above. Thus

$$\text{Distance}(A,B) = | \log (AP \cdot BQ / BP \cdot AQ) |$$

Hyperbolic Angles

This model preserves the angles of hyperbolic space and hence is called *conformal*. This is a major advantage over the non-conformal Klein model. The angle is measured as the number of degrees of angle between the tangents of two circular arcs intersecting at a point.

Gans Model

In 1966 David Gans proposed this model in the journal *American Mathematical Monthly* [3]. This model is not as widely used as other models but nevertheless is quite useful in the understanding of hyperbolic geometry. Unlike the Klein or the Poincare model, this model utilizes the entire Euclidean plane.

The lines in this model are interpreted as the branches of a hyperbola. The undefined terms: point, line, between, and “lies on” have the same meaning as in Euclidean geometry.

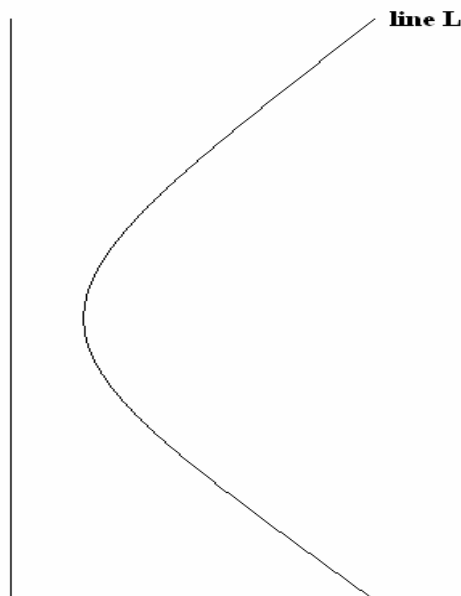


Figure 4.7: A Line in the Gans model

Weierstrass Model

This model is an infinite model of hyperbolic geometry. In this model the entire hyperbolic space is represented on the surface of a hyperboloid. A hyperboloid is a 3D figure which looks similar to a cone and has an analogue in the form of a light cone central to Einstein's theory of relativity.

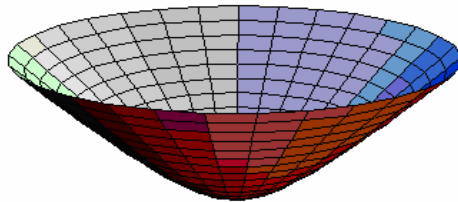


Figure 4.8: Hyperboloid of 2 sheet

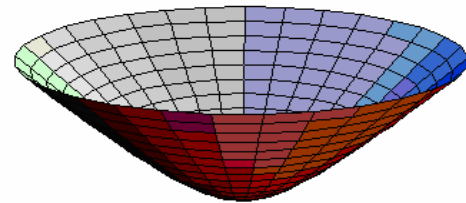
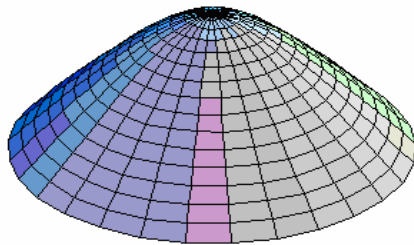


Figure 4.9: One sheet of a hyperboloid

The main advantage of this model is that all the other three models, Klein, Poincare, and Gans, are obtained by projections of this model as discussed.

Mathematically a hyperboloid is represented as $\langle \mathbf{X}, \mathbf{X} \rangle = x^2 + y^2 - z^2 = -k^2$ where \mathbf{X} is a vector (x, y, z) . This gives a hyperboloid of 2 sheets: upper and lower (Fig 4.8). The points on the lower sheet are reflections of the points on the upper sheet. Hence for simplicity, the lower sheet is discarded. The new equation that reflects this is $\langle \mathbf{X}, \mathbf{X} \rangle = -k^2$ and $z > 0$ [1].

The asymptotic cone bounding the upper sheet consists of all points that satisfy $\langle \mathbf{X}, \mathbf{X} \rangle = 0$.

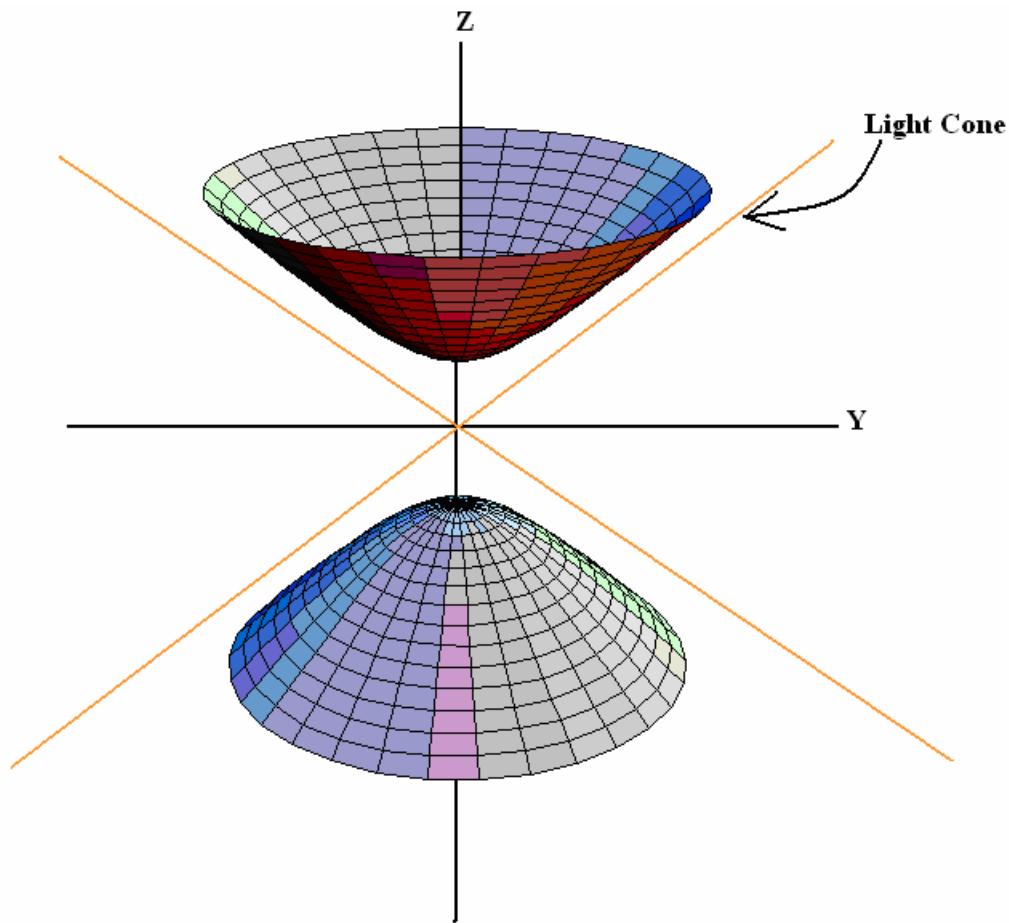


Figure 4.10: Asymptotic Cone

The undefined terms are interpreted in the following manner.

- **Point**

A point is defined as a point X satisfying the equation of upper sheet i.e. $\langle \mathbf{X}, \mathbf{X} \rangle = -\mathbf{k}^2$ and $z > 0$.

- **Line**

A section where a plane passing through the origin intersects the hyperboloid gives a line. The line is given by $\langle \mathbf{X}, \mathbf{L} \rangle = 0$ where \mathbf{L} is a h-normal to the intersecting plane. An h-normal is similar to the normal of the plane but reflected in the xy -plane.

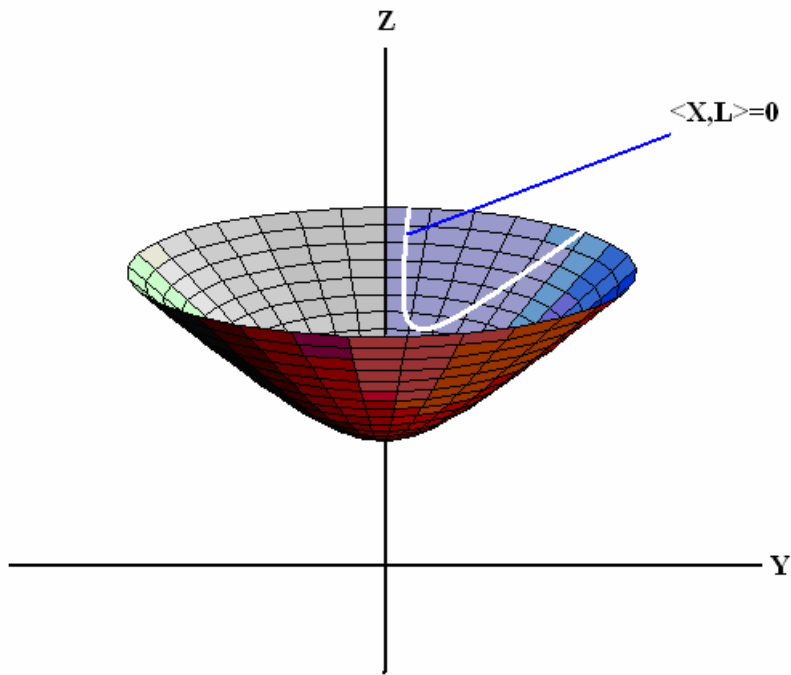


Figure 4.11: A Line in the Weierstrass model

- **Lies On**

If a point satisfies the equation $\langle X, L \rangle = 0$ and $z > 0$ then it is said to lie on a Line L.

- **Between**

Given 3 points P, Q and R we say that R lies between P and Q if R can be expressed as a linear combination of P and Q.

$$R = xP + yQ \text{ where } x, y > 0$$

4.3 Isomorphism

An isomorphism is a mapping from one model to another model. Formally, isomorphism means setting up one-to-one correspondences between the points and lines in one model with the points and lines in another such that the relations of betweenness, incidence and congruence are preserved [2].

The Weierstrass model is isomorphic to the Klein, Poincare, and Gans models. This property is used as a central theme in the development of this thesis.

Weierstrass – Poincare model Isomorphism

The Poincare model is obtained by stereographic projection of the 3D Weierstrass model onto the x-y plane. The projection is towards the point (0, 0, -1). It is given by

$$[x, y, z] \rightarrow 1/(z+1) [x, y, 0]$$

The Poincare to Weierstrass inverse projection is given by

$$[x, y, 0] \rightarrow 1/(1 - x^2 - y^2) [2x, 2y, 1 + x^2 + y^2]$$

Weierstrass – Klein model Isomorphism

The Klein model is obtained by stereographic projection of the Weierstrass model onto the z=1 plane. The projection is directed towards the point (0, 0, 0). It is given by

$$[x, y, z] \rightarrow [x/z, y/z, 1]$$

The Klein to Weierstrass inverse projection is given by

$$[[x, y, 1] \rightarrow 1/\sqrt{1 - x^2 - y^2} [x, y, 1]$$

Weierstrass – Gans model Isomorphism

Instead of a stereographic projection, a parallel or orthographic projection is used to obtain the Gans model from Weierstrass model. The projection is onto the x-y plane. It is given by

$$[x, y, z] \rightarrow [x, y, 0]$$

The Gans to Weierstrass inverse projection is given by

$$[x, y, 0] \rightarrow [x, y, \sqrt{1 + x^2 + y^2}]$$

5. Development Mechanics

The basic approach taken to implement this thesis project was to develop a 3D Weierstrass model. The software program used to implement the thesis is called HyperView3D. Different options were available to develop it.

5.1 Implementation choice: Java 3D

Various 3D graphics API's and languages are currently available in the market. Popular among those are OpenGL, VRML, JOGL, Xith3D and Java3D. Java3D was chosen as the API of choice to develop the HyperView3D application.

Java3D was more convenient for the development of Hyperview3D as compared to other choices. OpenGL is a C, C++ based API which is not portable across platforms and is a low level API. VRML is more useful for modeling simple applications. It can only be viewed by a VRML capable browser, which may or may not be available freely for all platforms. It also lacks the programming capabilities required to develop HyperView3D. Although JOGL is a java based API that sits on top of OpenGL it is primarily developed using a conversion tool called GLUEGEN. Hence it lacks the abstraction and extensibility required to develop HyperView3D. Xith3D is more suitable for the development of computer games using Java.

On the other hand Java3D is a powerful high level graphics API written in Java. Java3D allows the programmer to specify the actual program (objects and their interactions) and it handles the rendering and displays itself, shielding the programmer from the low level details. This makes the task of development easier. Although it may not include all the features offered by OpenGL or Direct3D, it has the most essential and used features of these two. It supports texturing, lighting, fog, spatial sound, customized behaviors, animation, interpolators, 3D object loaders supporting wide variety of 3D formats, support for wide

variety of input/output devices, etc. Java 3D has everything required for the efficient development of HyperView3D.

Being a Java API, it has all the features of the Java programming language, notably being that it is platform independent. Since it is envisaged that HyperView3D would be used as a desktop application as well as an applet, it makes sense to develop the application in Java. New Java technologies such as JNLP / Java Web Start allow the distribution of Java applications very easily.

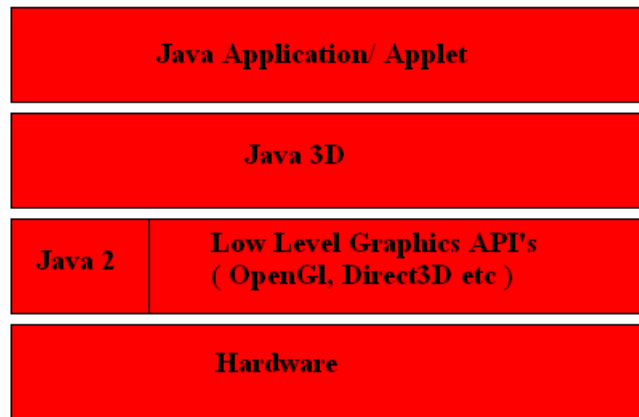


Figure 5.1: Java3D API

5.2 Java 3D basics

Java3D extends the Java platform by adding 3D development capabilities to the Java language. It sits on top of OpenGL or Direct3D hiding the details of the underlying API it is using. Besides acting as a wrapper over these two, it also uses a powerful object oriented paradigm in the form of the *scenegraph*. This feature makes it highly modular, extensible and flexible. Java3D development started in 1997 and it became a JCP (Java Community Process) project in 2004. The current version of Java3D is version 1.4 as of June 2006.

A *scenegraph* is a collection of nodes arranged in a tree format. They are used to store, organize and render 3D scene information. An operation applied to any node affects the nodes below it. Every Java3D application requires the specification of a scenegraph, which the

Java3D runtime system optimizes to produce an internal representation of the objects to be rendered including their behaviors and interactions [5]. A scenegraph-based approach allows us to define the logical and spatial relationship between objects efficiently. The following diagram represents a high level view of the HyperView3D scenegraph.

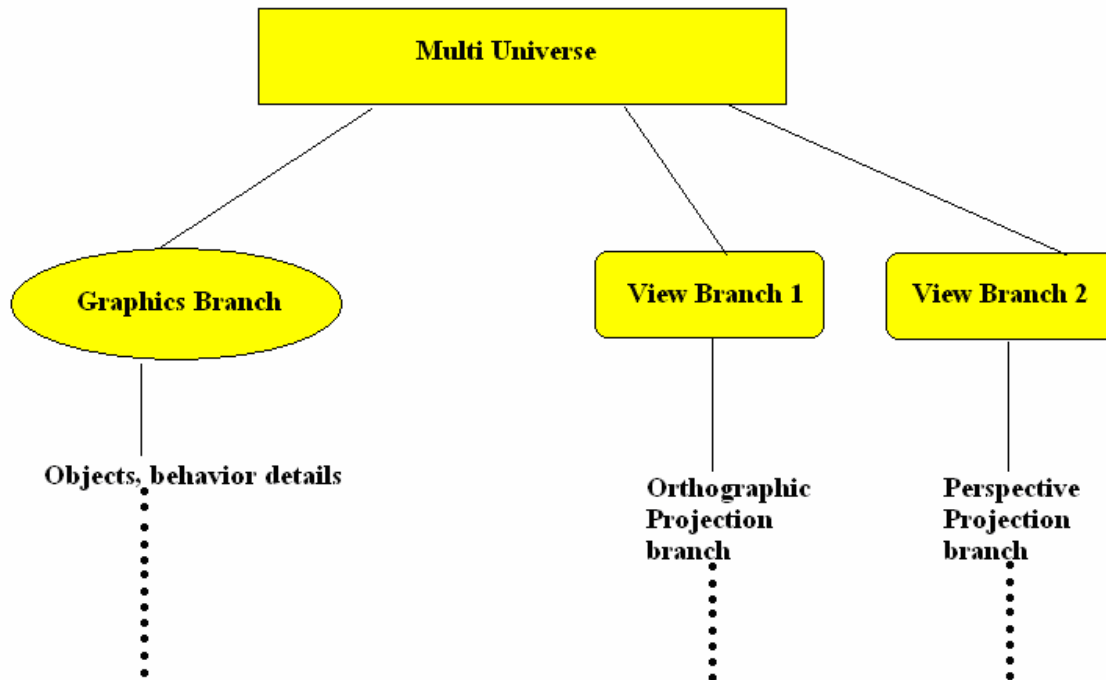


Figure 5.2: Hyperview3D Scenegraph

MultiUniverse is a utility Java3D class required for HyperView3D.

Note the scenegraph consists of 3 main branches

- Graphics Branch
- View Branch1 / View Branch2

The graphics branch basically defines the structure along with interactions among objects. The view branches specify how the structure looks on the screen. Various Java3D parameters allow specification of the view (Parallel/Orthographic or Perspective) and numerous other variables which fine tune the way the 3D object looks on the screen.

In HyperView3D, Java3D features such as real time texture mapping, customized behaviors, and customized interpolators are used extensively.

5.3 Creating the 3D Weierstrass model

The 3D hyperboloid was created using a four step process. First the 3D co-ordinates of the points required were calculated (Fig 5.3(a)). Then they were joined as polygons to create a wireframe image (Fig 5.3 (b)). This wireframe image was then filled to have a smooth appearance (Fig 5.3 (c)). Finally, a previously created jpeg image was wrapped around this filled wireframe using texture mapping (Fig 5.3 (d)).

The following figure was generated at default settings. The numbers of points generated were 4096 and the total numbers of polygons created were 1024. The user can change the default settings to alter the number of points and polygons created.

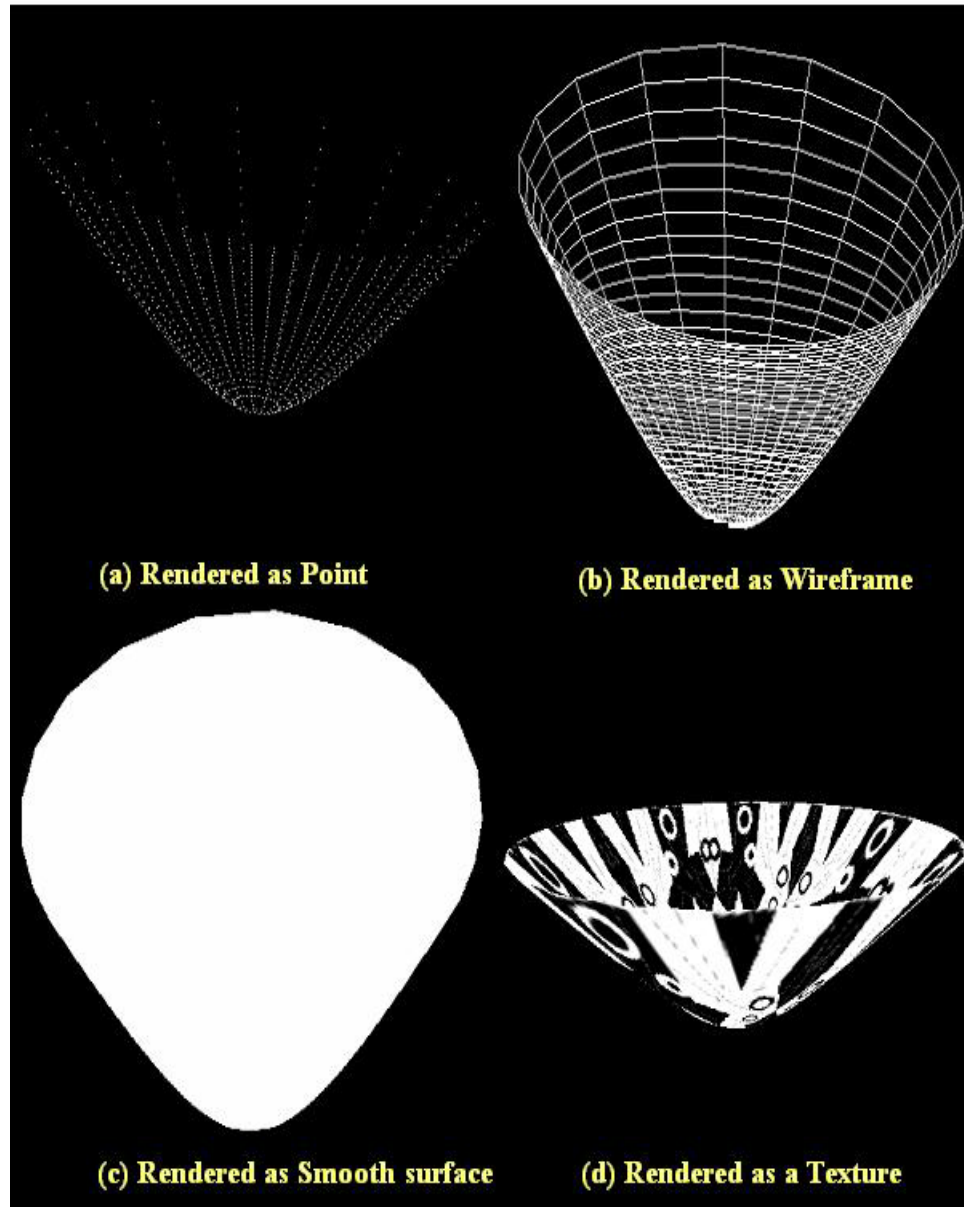


Figure 5.3: 3D Weierstrass model building steps

5.4 Mapping the texture

Java3D's built in texture mapping utilities were used to wrap the image around the hyperboloid. The image pattern selected was M.C. Escher's famous Circle Limit I (black and white fishes) [7].

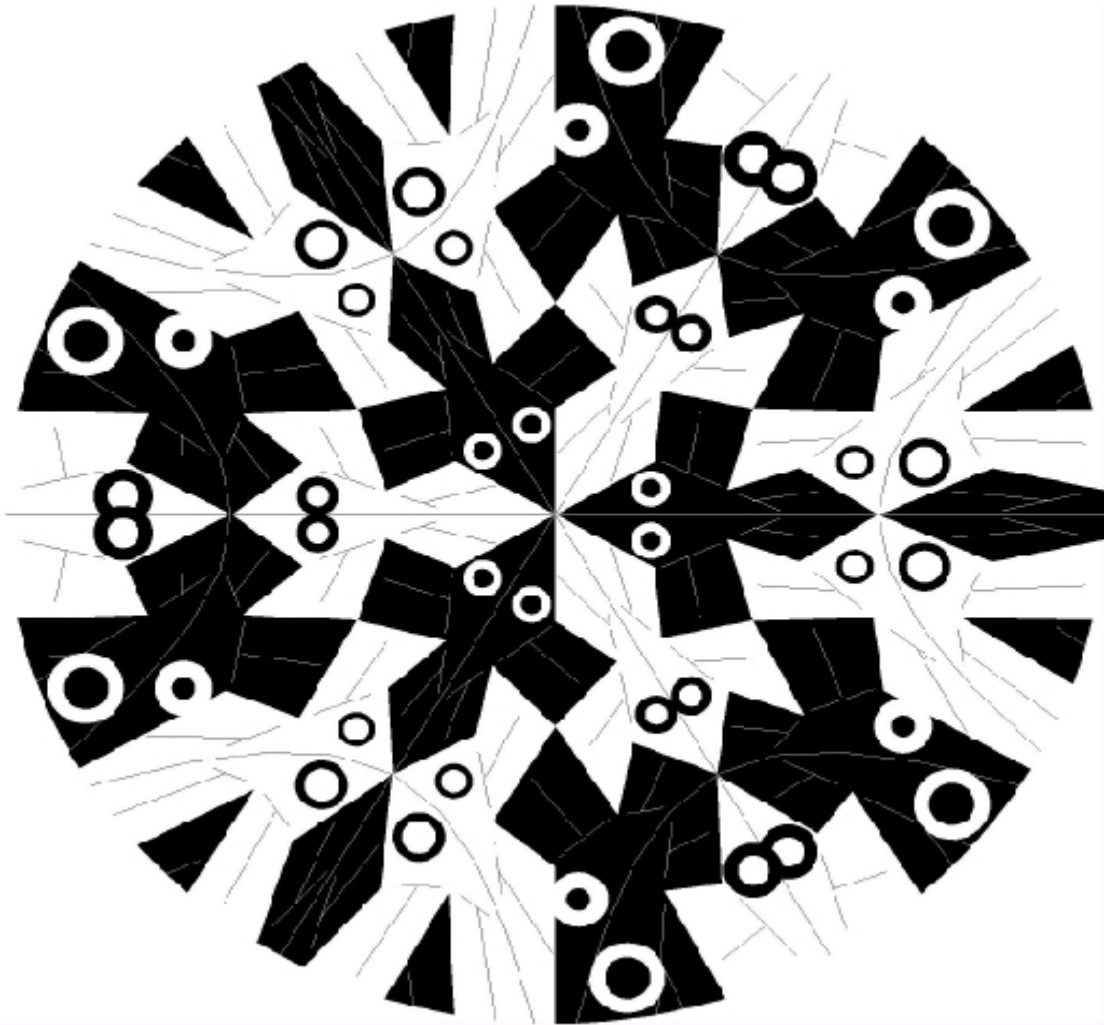


Figure 5.4: M.C. Escher's famous Circle Limit I

The actual jpeg image was created by Dr. Douglas Dunham using his computer program. The image was created as a Gans model image. Hyperview3D supports dynamic texture mapping.

5.5 Viewing the Models

One of the important properties of the 3D Weierstrass model is that one can see different models by viewing it from specific positions. This technique is used to see other models in HyperView3D. Thus, the cumbersome mathematical calculations discussed in the isomorphism of models can be avoided. This helps improve the performance in rendering. It also allows us to switch between models easily.

If the user is situated at the location $(0, 0, -1)$ and looks at the Weierstrass model vertically he sees the Poincare model projected onto the xy plane. If the user is located at $(0, 0, 0)$ he would see the Klein model projected onto the plane $z = 1$. If the user now shifts to infinity along $-z$ axis and the 3D Weierstrass model is projected orthographically he would see the Gans model. Thus a shifting of viewpoint and type of projection would enable the user to view the three models.

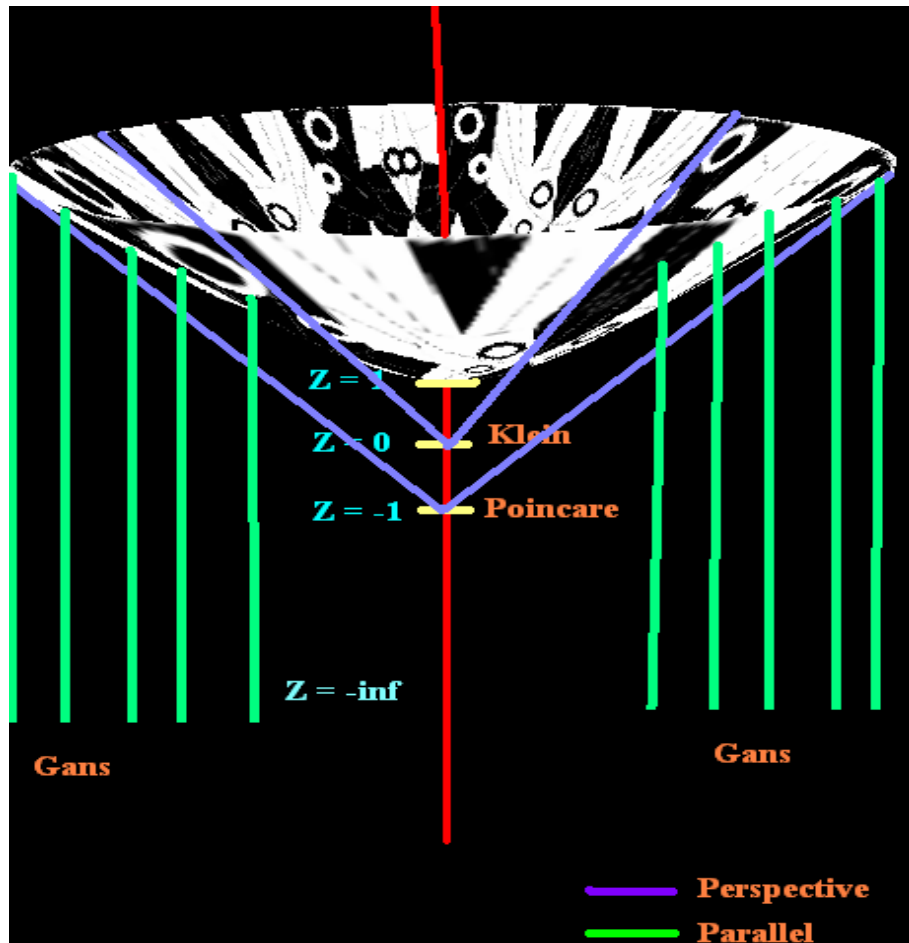


Figure 5.5: Projection types

5.6 Perspective Projection

In order to view the Klein and Poincare models, a perspective projection is required. In this type of projection, the point of projection is located at a finite distance from the object. The rays of projection meet at this point to produce a perspective image on the view plane in front of the projection point. A perspective projection makes objects located at a distance appear smaller than the objects located closer. It simulates the human eye but distorts angles.

5.7 Parallel Projection

In order to view the Gans model, a parallel orthographic projection is used. In this projection, the point of projection is located at infinity and the rays of projection are parallel to one another. They are assumed to intersect at infinity. A parallel projection does not distort angles. The size of the object remains the same irrespective of the distance from which it is viewed.

6. Hyperview3D

The primary aim of this thesis was to design a program that would improve understanding of hyperbolic geometry by studying its various models. Hyperview3D was designed by keeping this aim in mind. It has many features which might prove useful in the understanding of various models. The following sections introduce the reader to its various features.

6.1 Split Screen GUI

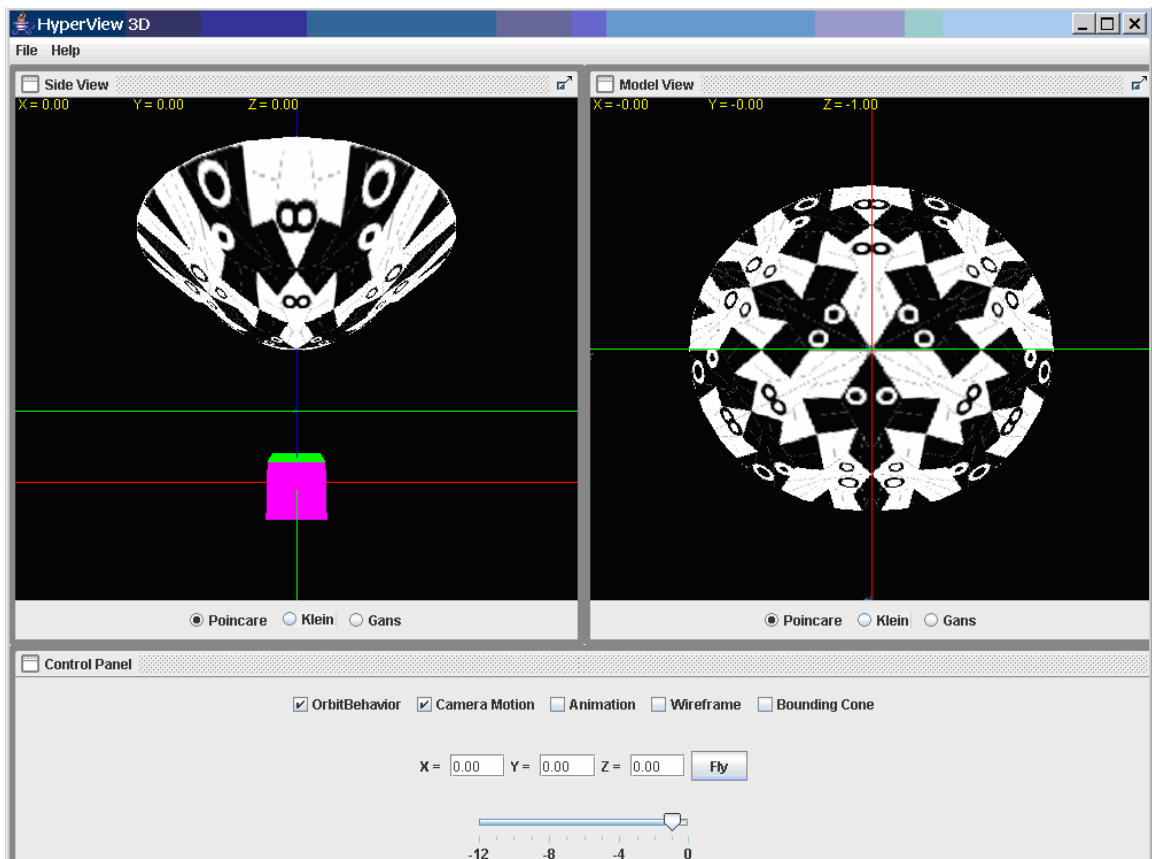


Figure 6.1: Hyperview3D split screen GUI

As seen in Figure 6.1, Hyperview3D has two split screens to view the models and a GUI panel with buttons at the bottom. Thus there are 3 main components:

- Side View
- Model View

- Control Panel

Side View

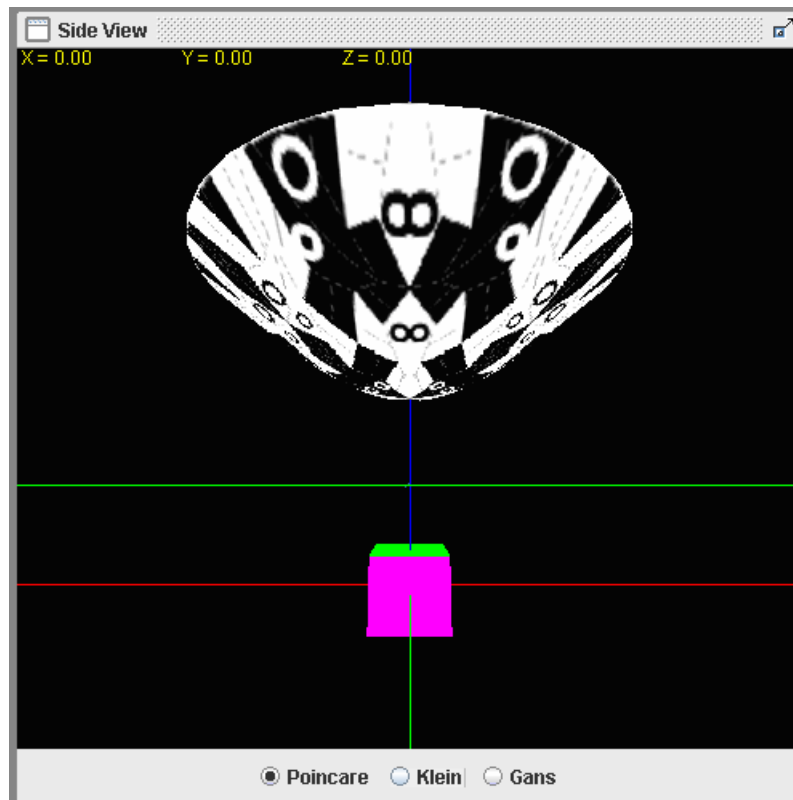


Figure 6.2: Side view

The basic use of the side view is to change the orientation of the 3D hyperboloid. This can be done by using mouse buttons. One can translate (right button), rotate (left button) and zoom (middle button) the hyperboloid in this view.

Model View

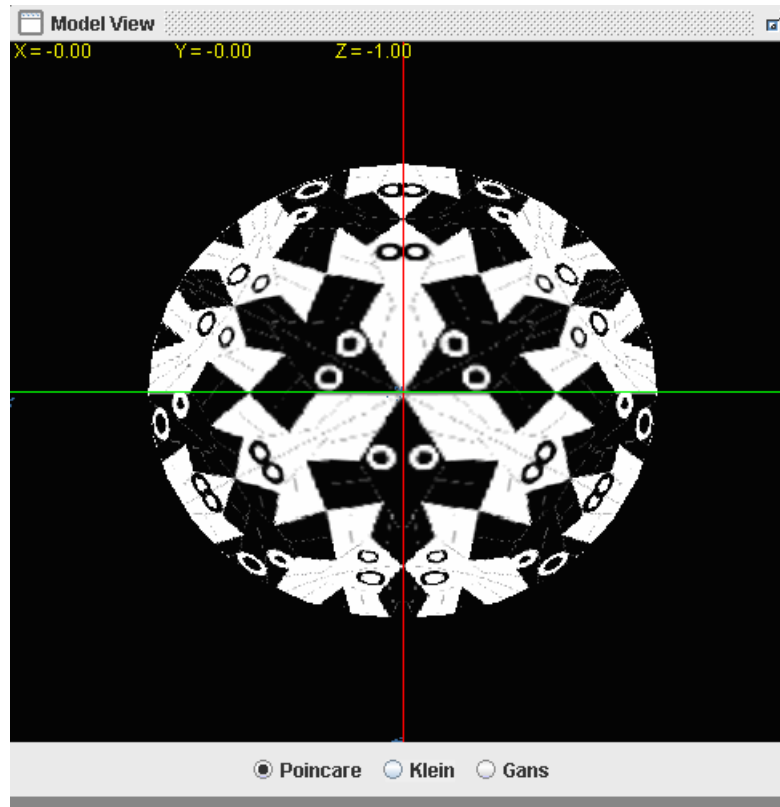


Figure 6.3: Model view

This view cannot be manipulated by the mouse. However it can be changed by manipulating the camera [depicted as cube] in the Side view.

Control Panel

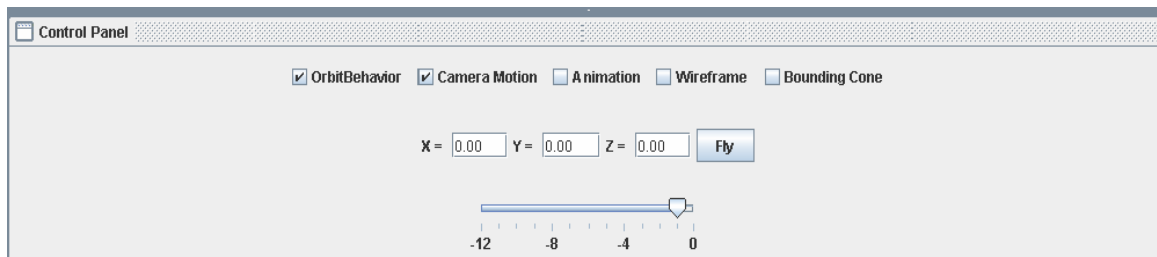


Figure 6.4: Control panel

This panel located at the bottom gives a centralized control on Hyperview3D. It has various radio buttons which enable/disable a particular feature on the Hyperboloid. It also has textboxes and a slider.

6.2 Modes of Operation

Hyperview3D operates in 3 modes:

- Comparison Mode
- Wireframe Mode
- Animation Mode

Comparison Mode

This mode is activated when the user clicks on any of the 3 radio buttons shown below in the Model and Side view.



Figure 6.5: Comparison mode activation buttons

The comparison mode allows the user to see any two models side by side. This helps in understanding the common features and differences between the models. The following figure shows a comparative view of the Klein and Gans Model.

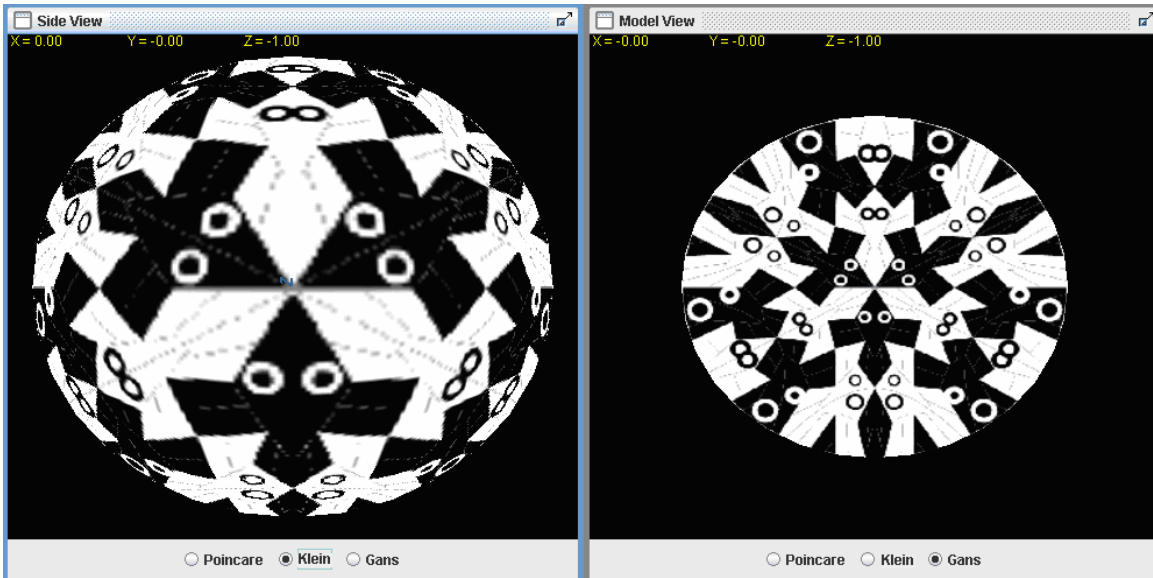


Figure 6.6: Comparison mode display

Wireframe Mode

This mode is activated when the user selects the Wireframe button from the Control Panel

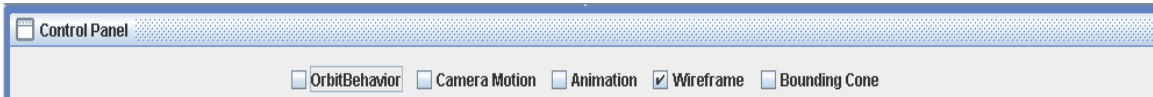


Figure 6.7: Wireframe mode activation button

This mode allows the user to see the models as a polygon structure. Running the comparison mode with wireframe mode allows the user to understand how space gets distributed within the models. The following figure shows Klein and Poincare space distributions when comparison mode is also activated along with wireframe mode.

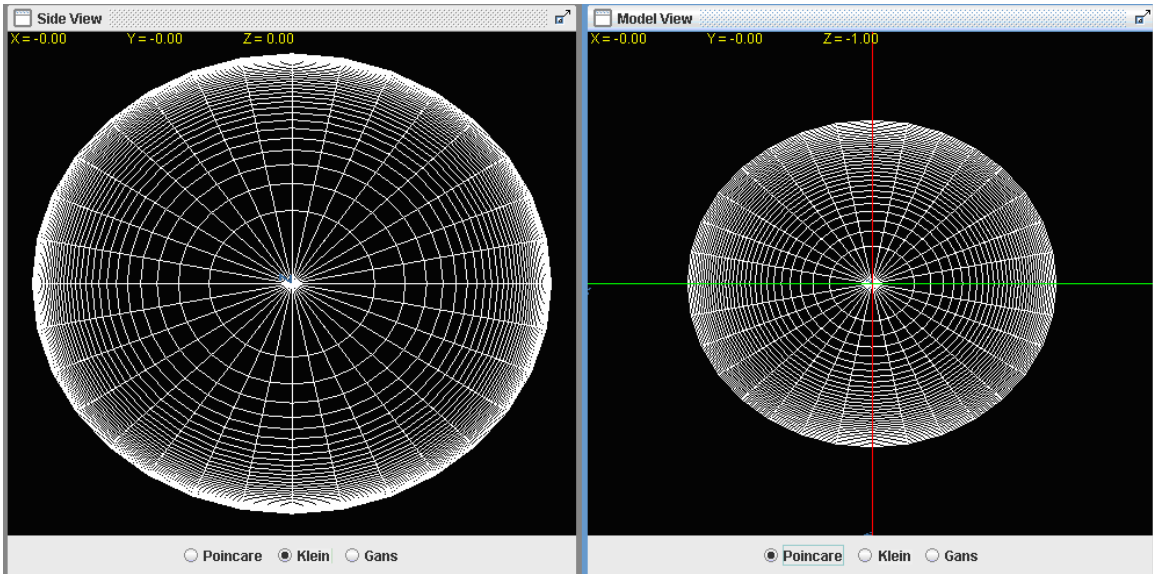


Figure 6.8: Wireframe mode display

Animation Mode

This mode is activated by selecting the Animation button from the Control Panel.

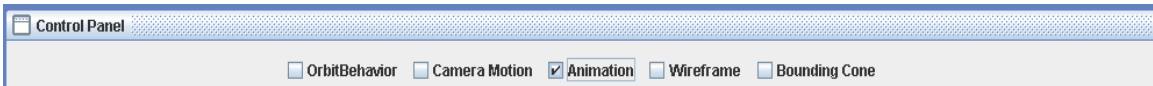


Fig 6.9: Animation mode activation buttons

Animation mode allows the user to see a predefined animation which makes a transition from the Poincare model to the Klein model and back in an infinite loop. This allows the user to see the intermediate steps between these two models. The animation is seen only in the Model view. The user can also see the camera in motion in the Side view.

6.3 Dynamic Texture Mapping

Hyperview3D allows the user to select a previously created image and texture map it onto the hyperboloid. This can be done from the File Menu. The figures below show the result of applying a new texture.

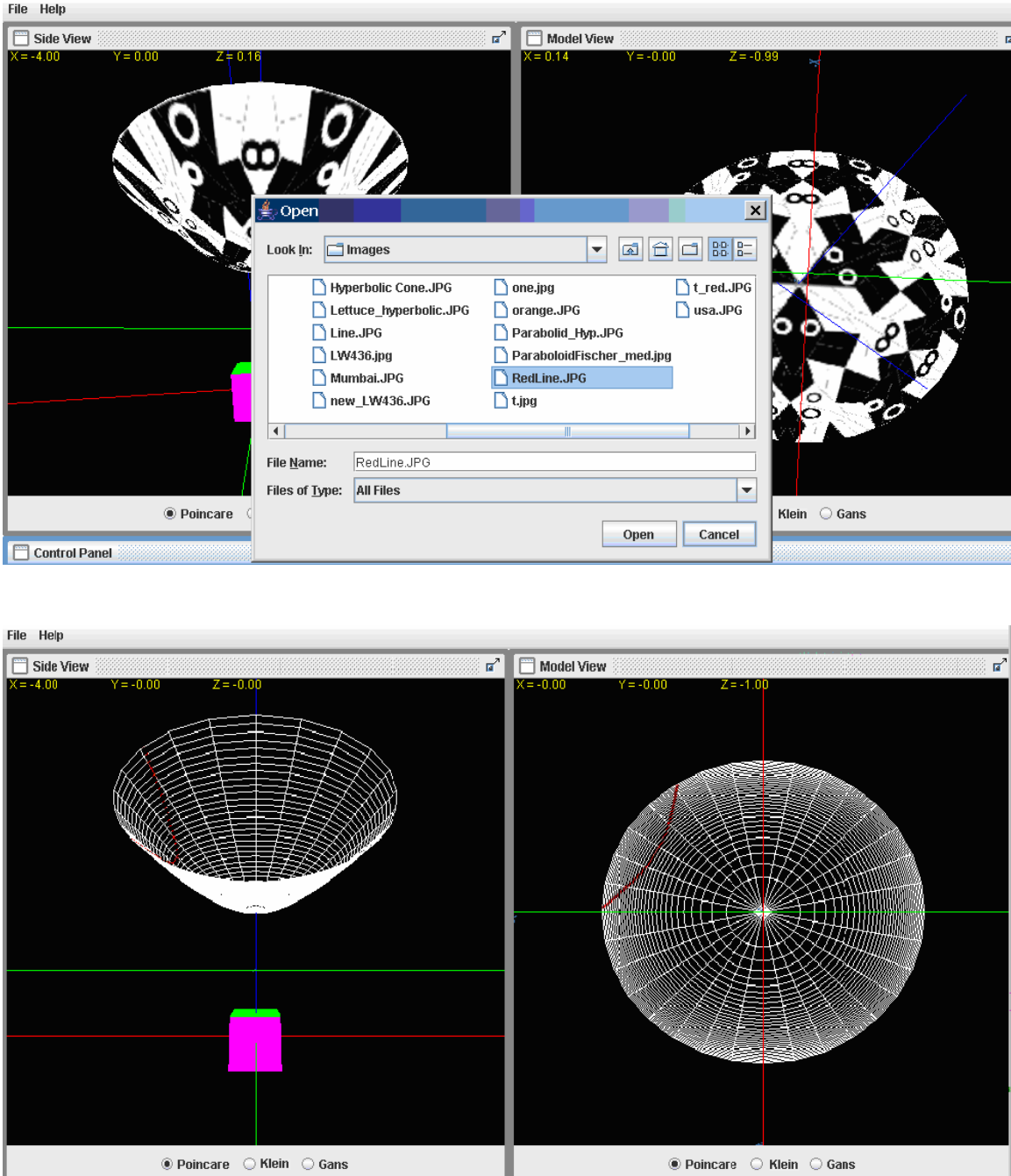


Figure 6.9: Dynamic Texture Mapping

6.4 Control Panel Options

OrbitBehaviour

Enabling/Disabling this button activates/deactivates mouse movement inside Side View. Thus when the button is selected the user can manipulate the 3D hyperboloid in the Side View.

Camera-Motion

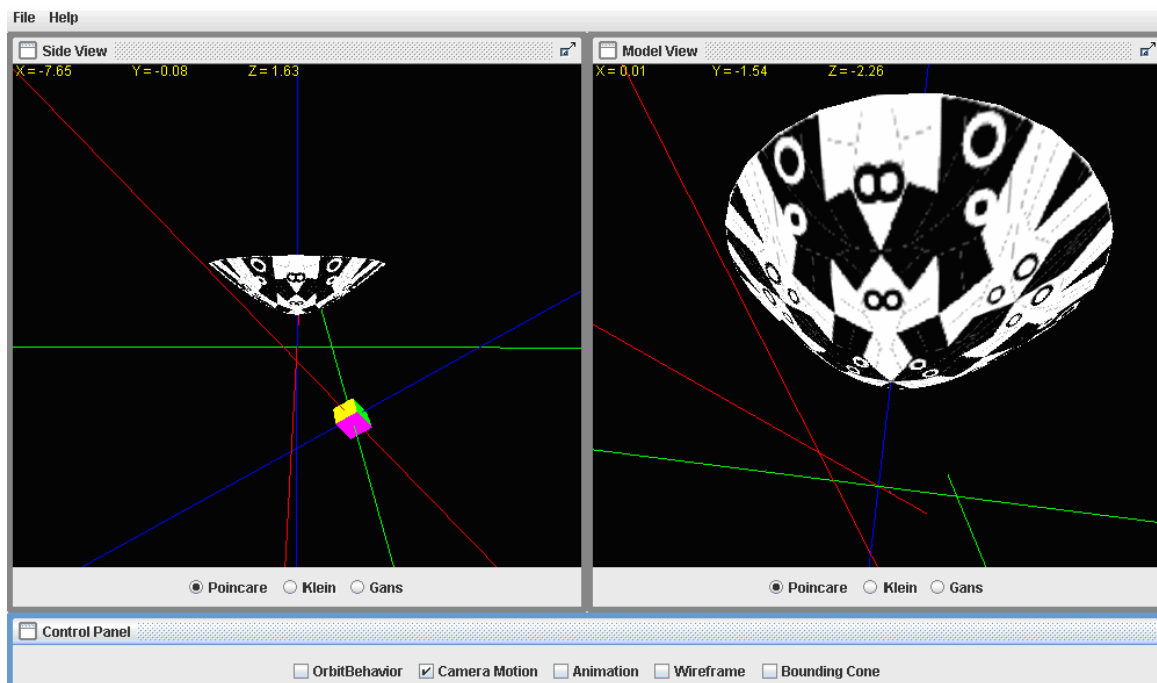


Figure 6.10: Camera in motion

Enabling/Disabling this button activates/deactivates camera manipulation in the Side View. This camera (represented as a cube with its green surface acting as the lens) is the place from which the user sees the hyperboloid in the Model View. The camera can be considered as the observer with the green surface acting as his eyes. Any change in the position and/or orientation of the camera would change the Model View.

Bounding Cone

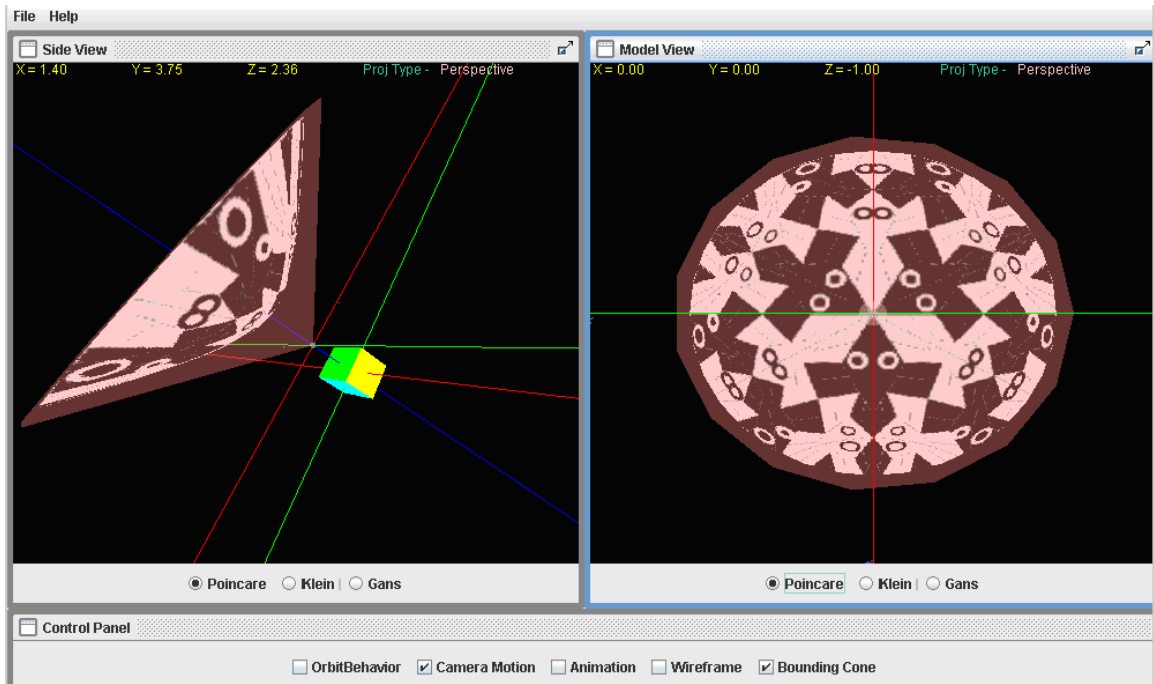


Figure 6.11: Bounding Cone

Enabling this button makes the asymptotic cone around the hyperboloid visible in both the views. The cone is semi-transparent. It becomes invisible if this button is deselected.

Text Boxes

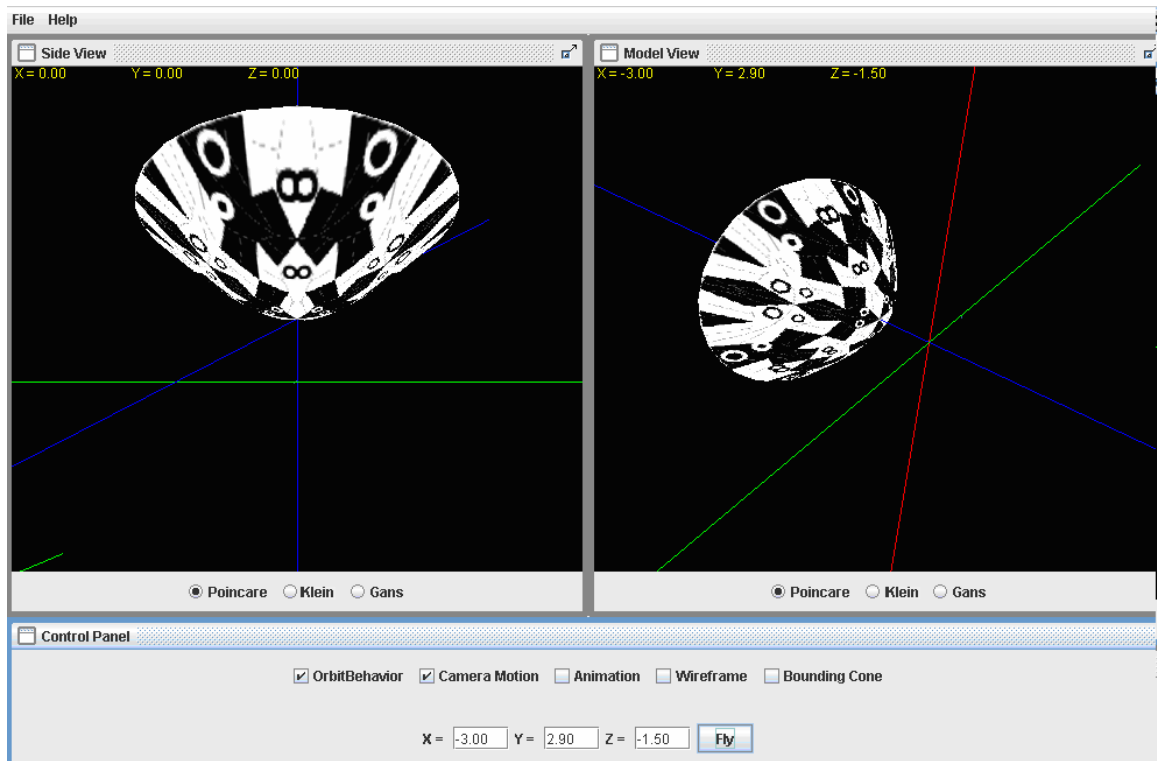


Figure 6.12: Precise position using Text box inputs

The three textboxes allow the user to enter the precise coordinates from which he would like to see the hyperboloid in the Model View. The precision is set to 2 decimal places. This feature gives finer control on the viewing of the 3D hyperboloid.

Slider

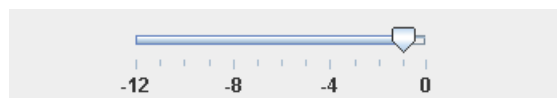


Figure 6.13: Slider

The Slider is useful in the Comparison mode. Moving the slider changes the z coordinate of the camera in the Side View which in turns changes the Model View. It acts a zoom lens.

7. Output and Results

Visual output was adopted as a mode of verification for the results obtained from Hyperview3D. The output verification was done by using the Wireframe mode.

7.1 Wireframe Mode ON

It is known that a line looks different in each of the models of hyperbolic geometry. In the Klein model it is chord on the disc. In the Poincare model it is circular arc. In Gans model it is one branch of a hyperbola. An image was created by modifying Dr. Dunham's image of Escher's Limit I shown below. This image was then texture mapped onto the hyperboloid with the Wireframe mode ON. The following output was observed in the Model View which confirmed the previous claims.

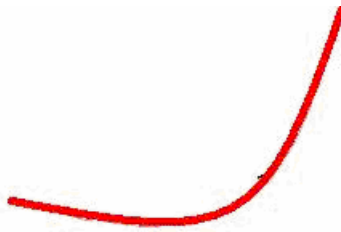


Figure 7.1: Input for Wireframe mode

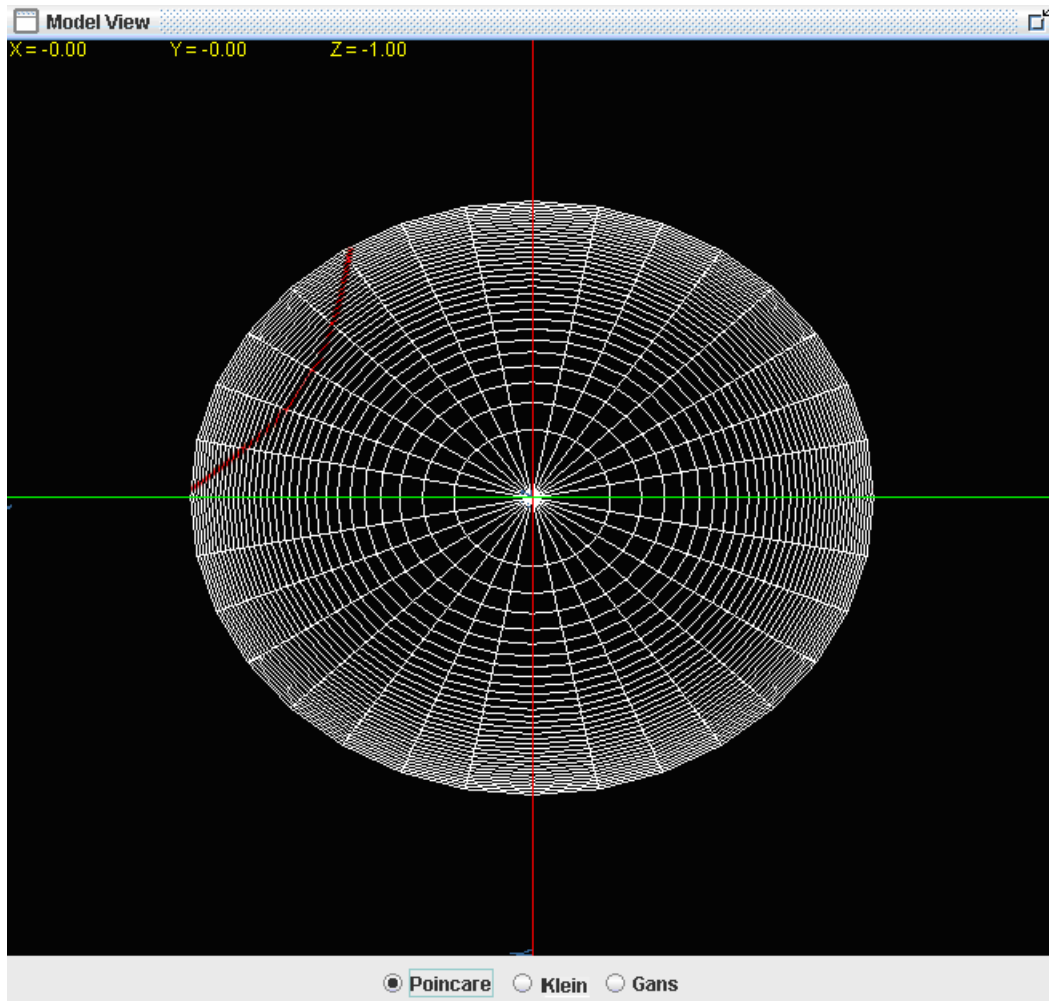


Figure 7.2: Poincare Model View

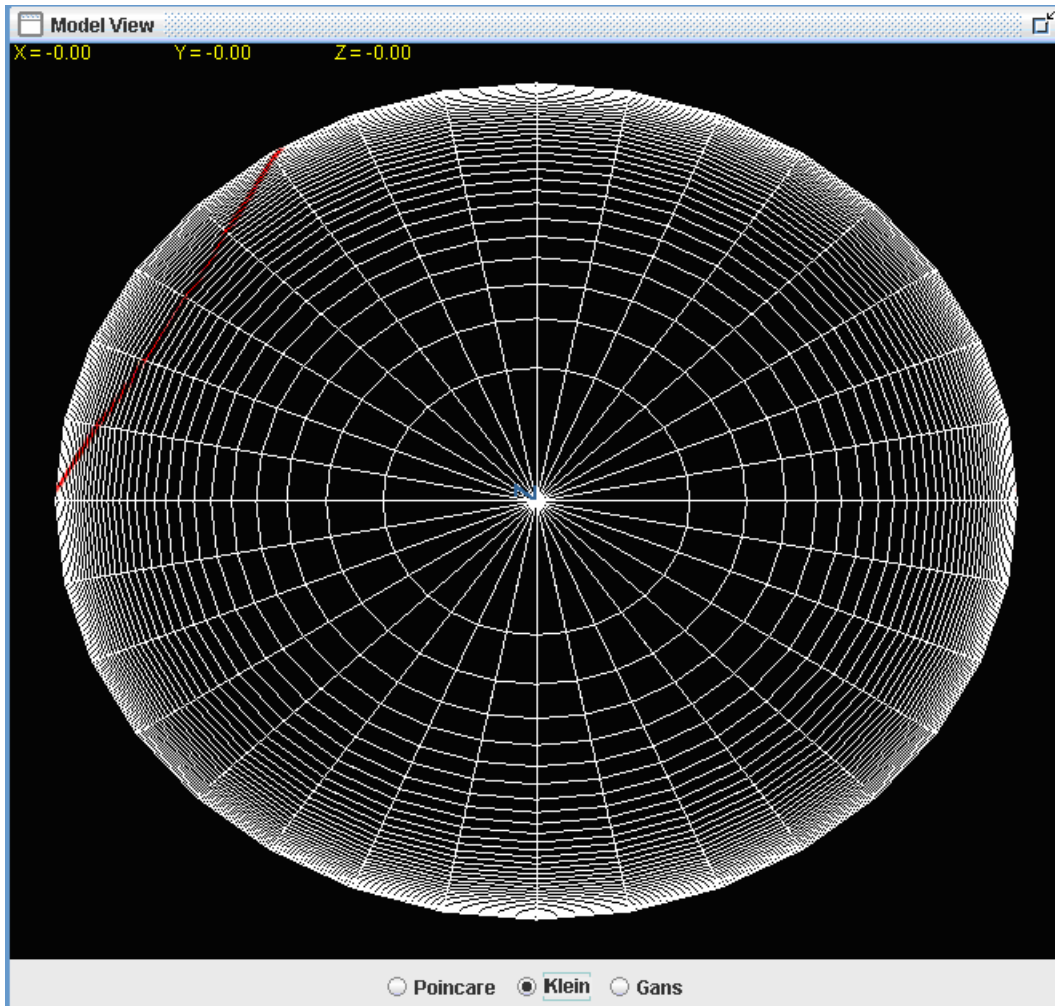


Figure 7.3: Klein Model View

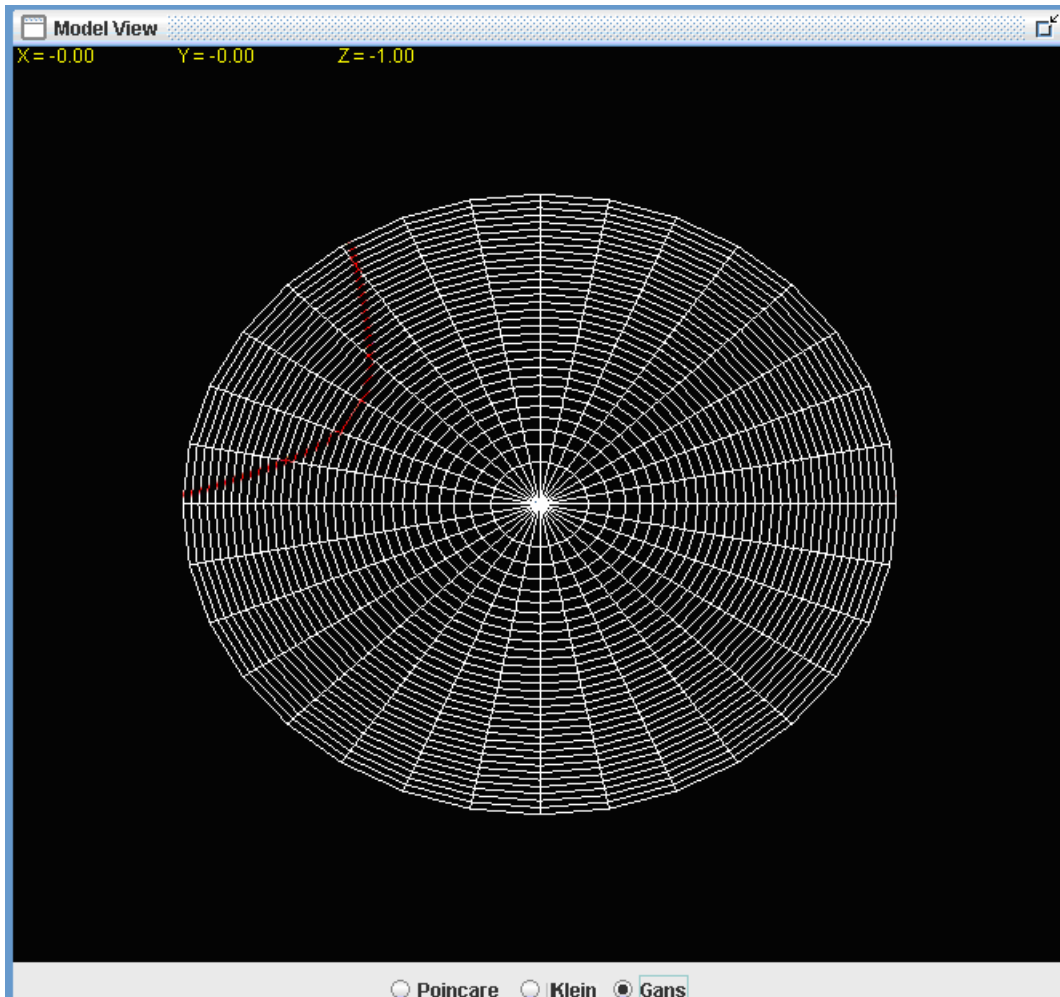


Figure 7.4: Gans Model View

7.2 Wireframe Mode OFF

Here Dr. Dunham's pre-existing Escher's Circle Limit I image for the Gans model was used to verify the output produced by Hyperview3D. The results were as follows.

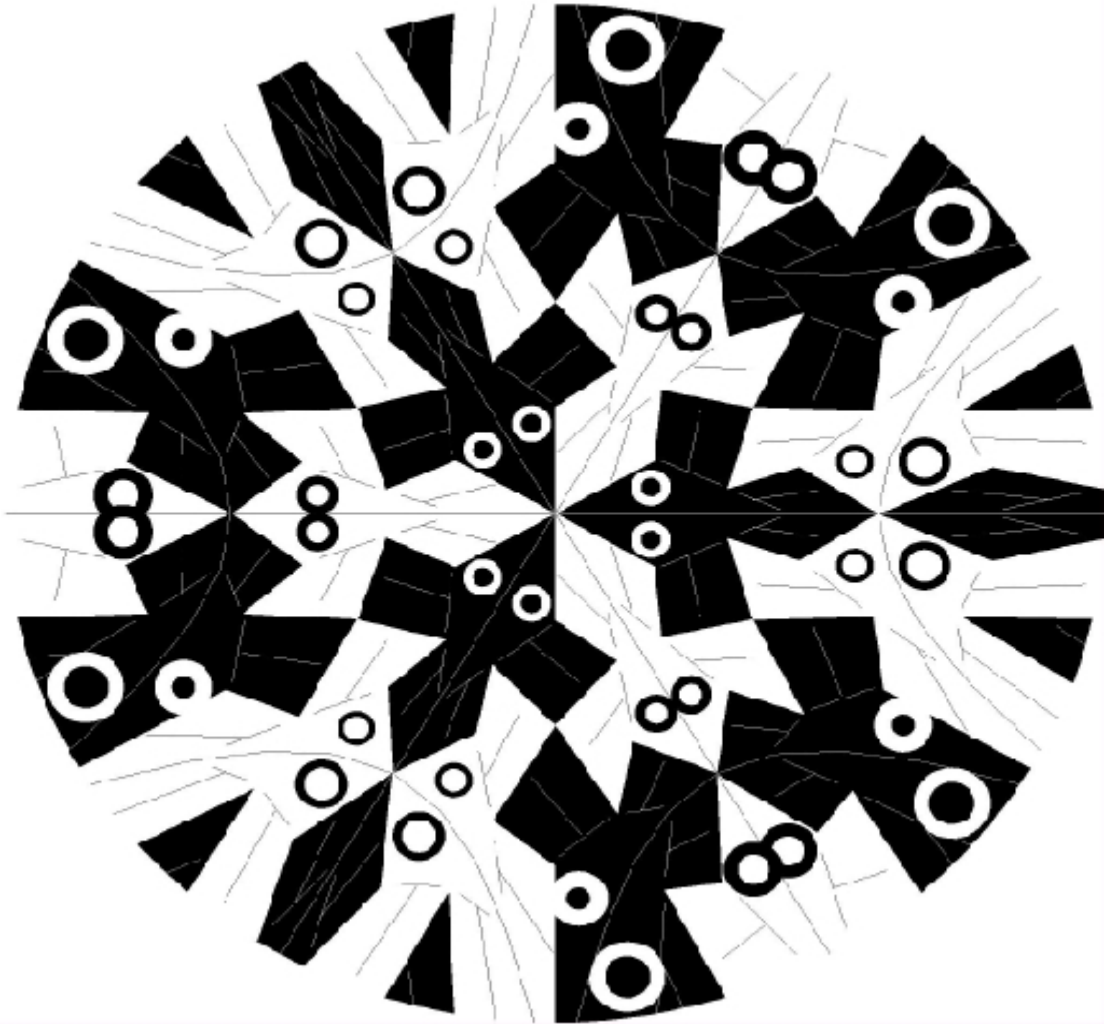


Figure 7.5: Escher's Circle Limit I: Input Image (pre-existing)

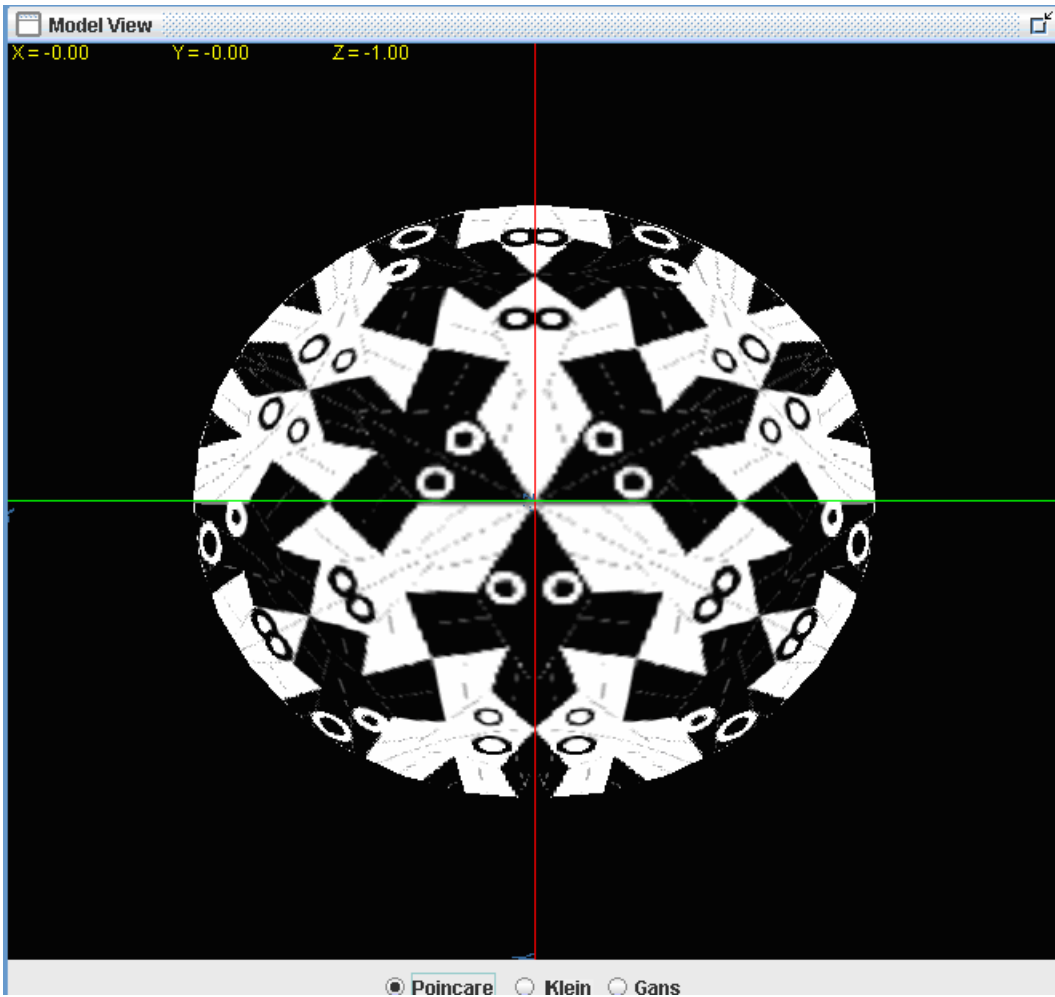


Figure 7.6: Escher's Circle Limit I: Poincare Model (HyperView3D)

Expected view: In the Poincare model the backbone lines in this pattern are circular arcs.

Observed view: The above image confirms with the expected view

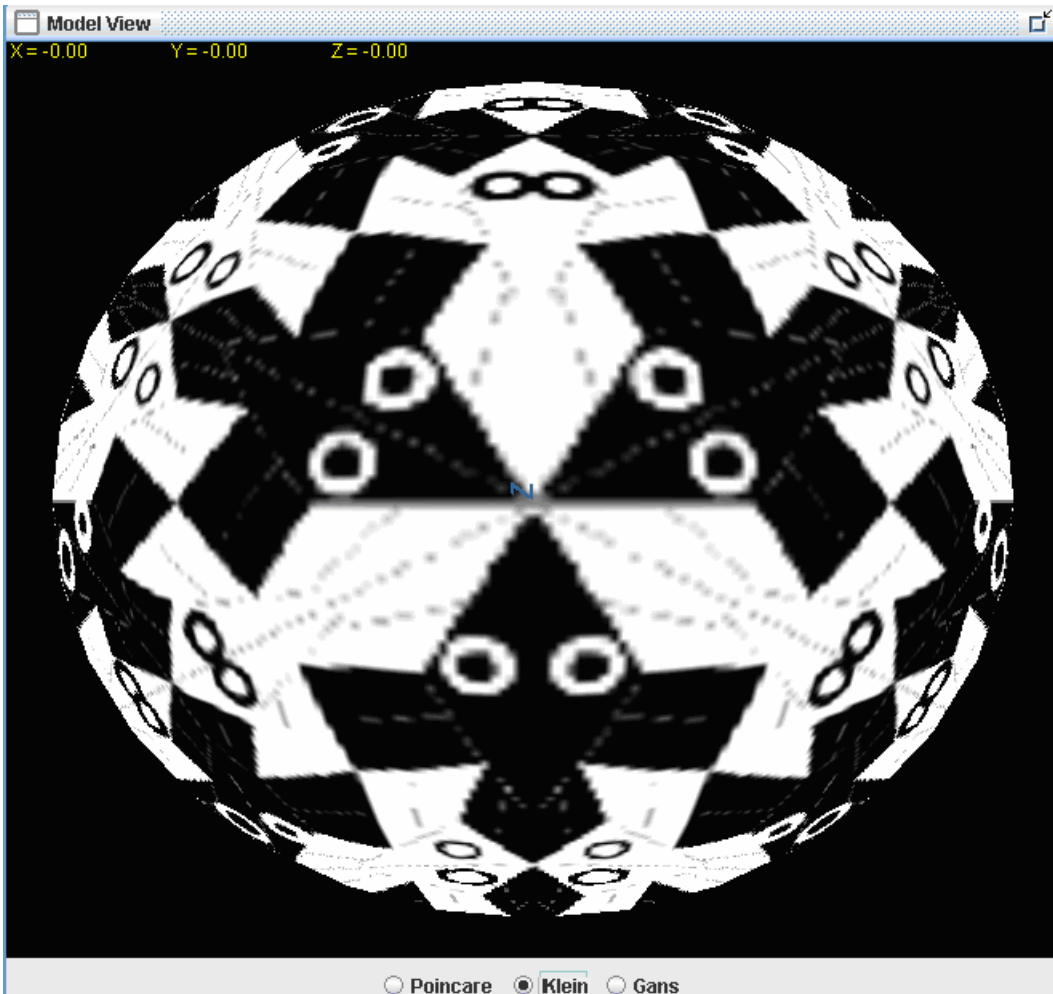


Figure 7.7: Escher's Circle Limit I: Klein Model (HyperView3D)

Expected view: In the Klein model the backbone lines in this pattern are chords (straight lines).

Observed view: The above image confirms with the expected view.

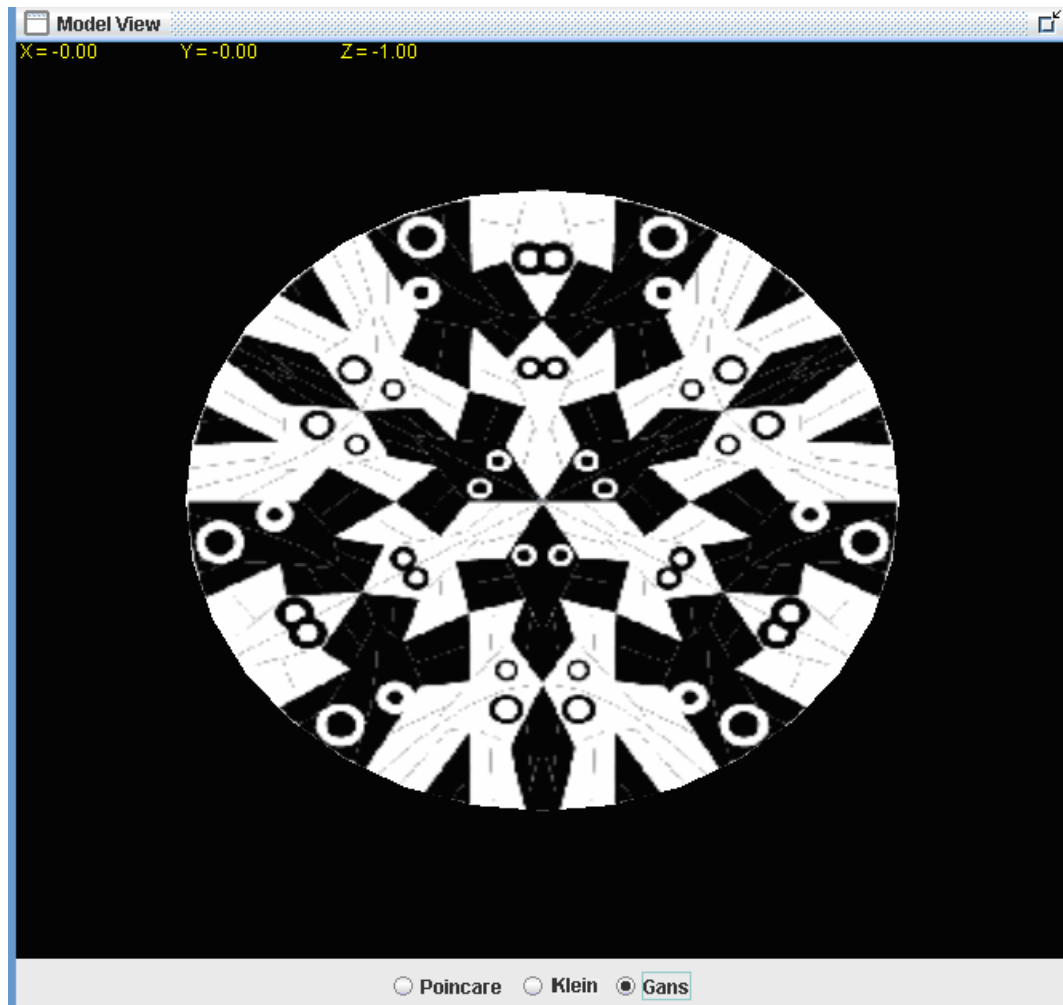


Figure 7.8: Escher's Circle Limit I: Gans Model (HyperView3D)

Expected view: In the Gans model the backbone lines in this pattern are arcs of a hyperbola.

Observed view: The above image confirms with the expected view.

8. Proposed Enhancements

Although Hyperview3D has various features there are many things which can make it even better in the future versions.

8.1 Integration of Dr. Dunham's program

The test image (Escher's Circle Limit I) in the Gans model was created by Dr. Dunham using his program written in the C language. One important addition would be to integrate this program directly in Hyperview3D. This would allow the user to create test images on the fly.

8.2 Run-time pattern generation

Another option would be to allow the user to select a motif (repeating pattern) file. An entire image can then be created on the hyperboloid directly. An animated version of this would be a good enhancement.

8.3 Customized Animation

The current version of Hyperview3D has a predefined animation which depicts the transition between Poincare and Klein models. The program can be enhanced to allow the user to specify a customized animation. One option would be to use the coordinates entered by the user in the textboxes. An animated transition can be shown from the current hyperboloid position and these new coordinates.

8.4 Geometric construct drawing

Hyperview3D's Model view shows the three models of Poincare, Klein and Gans. The behaviors of lines, triangle properties, angle and length measurements on the models is very different from their Euclidean counterparts. An advanced improvement could be to allow the user to directly draw geometric constructs such as points, lines and triangles on these three models in the Model view. Different properties pertaining to this drawing such as angle sum of triangles, lengths, etc. could then be displayed as text messages.

APPENDIX

Settings Information

The following table contains the default setting for which Hyperview3D works. This information is present in a file which is read at runtime by the software program.

Some of the parameters have an effect on how the 3D hyperboloid is constructed. Other parameters have an effect on the camera based viewing mechanism (Parallel/ Perspective) of Java3D.

The user can change this information before starting the program.

Parameter	Value	Info
SCALE_FACTOR	1	Factor by which 3D model is scaled.
NO_OF_SLICES	32	Used in construction of 3D model. The more vertical slices, the more detailed the figure.
NO_OF_ANGLE_DIVS	32	Used in construction of 3D model. The more angle divisions, the more detailed the figure.
Z_HEIGHT	2.45	Used in construction of 3D model. Decides the height of the 3D hyperboloid.
FIELD_OF_VIEW	3.1428	Defines the view field of the camera. This value is scaled by 4 to obtain a 45 degree field of view.
ASPECT_RATIO	1.0	Ratio of width to height. Used for camera setting.
FCP_DIST	0.1	Front Clipping Plane distance.
BCP_DIST	30	Back Clipping Plane distance.
OBSERVER_POSITION_X	0.0	Initial x-coordinate of observer eye position.
OBSERVER_POSITION_Y	0.0	Initial y-coordinate of observer eye position.
OBSERVER_POSITION_Z	-1.0	Initial z-coordinate of observer eye position.
VIEW_X_VALUE	1.05	View frustum's x-coordinate.
VIEW_Y_VALUE	-0.85	View frustum's y-coordinate.

Parameter	Value	Info
VIEW_Z_VALUE	0.0	View frustum's z-coordinate.
RIGHT_EDGE	3.3	Right edge distance of the rectangular parallelepiped. Used in parallel projection
BOTTOM_EDGE	-3.3	Bottom edge distance of the rectangular parallelepiped . Used in parallel projection.
TOP_EDGE	3.3	Top edge distance of the rectangular parallelepiped. Used in parallel projection.
NEAR_DIST	2	Front edge distance of the rectangular parallelepiped. Used in parallel projection
FAR_DIST	300	Back edge distance of the rectangular parallelepiped. Used in parallel projection.

References

1. Faber Richard L. (1983). *Foundations of Euclidean and Non-Euclidean Geometry*. Marcel Dekker Inc, New York.
2. Greenberg M.J. (1994). *Euclidean and Non-Euclidean Geometries*. W.H.Freeman and Company, New York.
3. Gans David (1966). A New Model of the Hyperbolic Plane. *American Mathematical Monthly*, Vol. 73, Issue 3, March 1966.
4. Foley James, van Dam Andries, Feiner Steven, Hughes John, (1997), *Computer Graphics: Principles and Practice, 2nd Edition*, Addison-Wesley Publishing Company, Inc.
5. Walsh E. Aaron, Gehringer Doug (2002). *Java 3D API Jump-Start*, 1st Edition, Prentice Hall PTR, Inc.
6. Dunham D. (1986). Hyperbolic Symmetry. *Computers & Mathematics with Applications*, Vol. 12B, Nos. 1 and 2, 1986, pp. 139—153.
7. Dunham D. (1986). Creating Hyperbolic Escher Patterns. In *M.C. Escher, Art and Science*, edited by H.S.M. Coxeter, et al. New York: North-Holland.
8. The Institute of Figuring.
<http://www.theiff.org/oexhibits/>

9. Interactive Java Software for Creating Ruler and Compass Constructions in both the Poincaré Disk and the Upper Half-Plane Models of Hyperbolic Geometry.

<http://www.cs.unm.edu/~joel/NonEuclid/>

10. Wilkinson Microwave Anisotropy Probe.

<http://map.gsfc.nasa.gov/>

11. Relativity and Black Holes Curvature and Geometry.

<http://www.etsu.edu/physics/plntrm/relat/curv.htm>