

Using Cohen Sutherland Line Clipping Algorithm to Generate 3D Models from 2D Drawings

Marah M. Taha

marah.neamy@Alnoor.edu.iq

Medical Laboratory Techniques

Al-Noor University College, Mosul, Iraq

Received on: 05/05/2019

Accepted on: 06/10/2019

ABSTRACT

This paper provides an efficient algorithm to generate three dimensional objects from simple uncomplicated 2D environment, lead to reduce processor effort, limit of using complex mathematical operations. Most of the previous researches used the idea of drawing by vanishing point to generate 3D objects from 2D environment, But the algorithm designed in this paper provides an idea of how to draw three-dimensional shapes from two-dimensional drawings when applying Cohen Sutherland Line clipping algorithm, so that a basic two-dimensional shape is inserted from a set of points connected with each other must be within vision borders with a vanishing point outside of vision that is connected with all points of basic shape to consist a group of lines with partial intersections. So that any point has specific limited vision border which represents one of its coordinates of depth vertex, finally 3d object is generated when all clipping processes are completed to obtain other coordinates for all points.

Keywords: 3D model, 2D drawing, vision field, vanishing point.

أستخدام خوارزمية قطع المستقيمت كوهين -سذرلاند لتوليد نماذج ثلاثية الابعاد من رسومات ثنائية الابعاد

مرح محمد طه حسن

كلية النور الجامعة

الموصل، العراق

تاريخ قبول البحث: 2019/10/06

تاريخ استلام البحث: 2019/05/05

الملخص

يقدم هذا البحث خوارزمية فعالة لتوليد نماذج ثلاثية الابعاد من بيئة ثنائية الابعاد بسيطة غير معقدة, يؤدي الى تقليل جهد المعالج والحد من استخدام العمليات الرياضية المركبة. استخدمت معظم الأبحاث السابقة فكرة الرسم بنقطة التلاشي لإنشاء كائنات ثلاثي الابعاد من بيئة ثنائية الابعاد , لكن الخوارزمية المصممة في هذا البحث تقدم فكرة عن كيفية رسم أشكال ثلاثية الابعاد من رسومات ثنائية الابعاد عند تطبيق خوارزمية كوهين لقطع المستقيمت (Cohen- Sutherland Line Clipping Algorithm). يتم ادخال الشكل الاساسي الثنائي الابعاد من مجموعة نقاط متصلة مع بعضها البعض ضمن حدود الرؤيا مع نقطة تلاشي تكون خارج حدود الرؤيا تتصل بكل نقاط الشكل الاساسي لتشكل مجموعة من المستقيمت ذات قطع جزئي (Partial Intersection), بحيث كل

نقطة ادخال تمتلك حد رؤيا معين والذي يمثل احد احداثياتها لنقطة العمق, في النهاية يتم إنشاء كائن ثلاثي الأبعاد بعد اكتمال جميع عمليات القطع عند الحصول على الاحداثيات الاخرى لكل النقاط.
الكلمات المفتاحية: نماذج ثلاثية الأبعاد, نماذج ثنائية الأبعاد, حدود الرؤيا, نقطة التلاشي.

1- Introduction

In the recently years, three Dimensional models have become important because they are involved in many fields, especially video games, movies, architecture and art, the importance of these drawings has also increased because they suggest something of imagination and sophistication[4] such as design 3D walls and floors[6].

There are many benefits when generate 3D models from 2d environment specially the ease of understanding composition and avoiding complexity [4] because 3D model is the evolution of computer graphic which is accompanied by difficult processes as “determining projection equations, hidden surfaces remover, arranging the objects in space, add many effects to make the scene more realistic such as lighting, fog, texture mapping, render 2D image on screen from 3D object, etc.”[7].

Since a long time, many of professional painters succeed in their attempts to draw these complex 3d objects on "a piece of paper" using specific methods like “organization to draw all elements in the scene via perspective point”, which help them to draw a complex object carries a great meaning of trick and attractive, it seems to be a model with actual depth [6], programmatically these achieves lead to many benefits such as to facilitate the design, stay away from the complexity, shorten the time [2], reduce the effort and give as a basic idea or an introduction to the understanding of the three dimensional environment [6].

Cohen Sutherland line clipping algorithm is characterized by its ease and speed in obtaining results and it is one of the most famous and simplest clipping algorithms [8], Because of these features this algorithm is used in this paper to generate 3d Models from 2d drawing that will be explained later.

Through this paper, many sections will be reviewed such as the characteristics of three dimensional environment, perspective point and the types of projection, Cohen Sutherland line clipping algorithm, and the generation of 3D models will be explain step by step via the proposed algorithm, In addition there are many reached conclusions that will be mentioned later during explanation the practical part.

2- Previous works

There are many previous researches in computer graphics relied on concepts on how design complex 3D objects from 2D drawings, following is some of what has reviewed the work of the researchers in this field.

1. Jeehyung Lee and Thomas Funkhouser in 2008 presented a work which was provided to generate 3D elaborated objects from 2D layout drawing via real interfaces with specific database contain 3D pieces, these are selected by user to be merged with original layout.
2. Junfang Zhu, Hui Zhang, and Yamei Wen in 2008 presented a work which was provided to design 3D objects (building) from 2D floor plans when introduce shape-opening graph with structure components provided by odd-even- based edges breaking algorithm to put wall lines instead of openings. Finally 3d object obtained after generation all internal spaces contour such as rooms or aisles and external spaces contour.

3. ESTELLE NOÉ in 2017 presented a work which was provided to obtain 3D modeling objects of rigid layered parts used in video games and the production of movies that are somewhat complex and take a long time to execute such as animals with rigid shells and armors when fabricate 2D photo with computed 2D patterns and using Circular and Bezier curve to obtain depth information and hidden surfaces.
4. Alec Rivers, Fredo Durand and Takeo Igarashi in 2018 presented a work which was provided to design 3D complex objects by combined their sub pieces from different views using CSG possessing.

3- Computer graphic environment

Computer graphics is one branch of computer science [10]. It has grown exponentially, developed in recent decades, and progressed from 2D initial drawing to high quality and 3D complex environment, It has wide use in many fields such as (art, design, education, computer games) [7].

3D computer graphics are technical works made up of software and special equipment, as it may differ from 2D drawings. In the 3D graphical environment, there are many complex basic operations to obtain 2d images from different views of 3D models. These operations may be classified in to three basic collection steps [7]

1. Modeling: - is the stage of forming or designing an object or group of individual objects that are displayed in a scene using set of points in space connected by different geometric entities such as lines, triangles, polygons etc. [6].
2. Scene layout setup: - The rearrangement of virtual objects may include lights and cameras, and some basics used in the process of generating static or moving images, for example, interest in making the movement of models complex and varied by designing a set of key frames that defines the starting and ending points of each object, The object is shaped from one body to another or the object moves from one location to another. This may help to provide several complex motions instead of specifying simple transformations applied on objects [7].
3. Rendering:- is the final stage of displaying a two-dimensional image on the screen from a three-dimensional model such as taking a picture or scene from a film after the design finished [10].

Three-dimensional computer graphics are artistic designs by calculating and presenting two-dimensional images from different perspectives, arranging objects in the landscape, calculating projections, and so on. In general, they are closer to sculpture and photography, while two-dimensional drawings are in the form of paintings [6].

4- Projection and perspective point

Projection is a set of mathematical processes, the first stage in the process of converting an object from three-dimensional space into two-dimensional space to be drawn on a two-dimensional screen [7]. It is an equivalent process for selecting a lens for the camera. Through this conversion, we can define the vision field of the objects to be seen and to some extent determine what it will look like [10], and there are two basic types of it, perspective projection and parallel projection [6].

When designing shapes simulated the reality, we use the projection perspective, through which the shapes look like they are seen by the eye [10], so that nearby objects look larger than others with the same size [7]. The lines extending from the object to be projected may intersect with the level of vision, and the vanishing point (the center of projection) represents the new points of the object after projection [10].

In the art, perspective is an expression used by painters to obtain 3d scene from 2d flat surface, this way provide different views of a scene, It looks like a three-dimensional model with real depth, as Shawn in figure (1), part (a): draw 2d patch from a set of lines, but part (b): add perspective point to previous shape to obtain deep sense when draw many lines intersected with it, part(c) when remove perspective point [6].

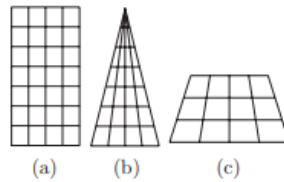


Figure (1): intersecting lines [6].

5- Cohen Sutherland Line clipping algorithm

Cohen Sutherland algorithm is considered the primary clipping line algorithm, characterized by easy and simple to apply [3]. It includes two basic elements, frame (field of vision) and line (with two ending vertexes). To draw a line on the specified vision field borders, there are three basic prospects: (a) if the two vertexes of the line, are output the vision field, then this line has status "rejection" because it is out the limits of the vision completely, leading to non-drawing. (b) If the two vertexes of the line, are inside the vision field, then this line has the status "acceptation" because it is in the limits of vision borders completely, leading to draw it totally. (c) if the one vertex of the line inside the borders of the vision field and another vertex is out, then this line has status "partial", leading to draw a line between internal point and intersection point located on the near border to external point [8], As shown in the figure (2).

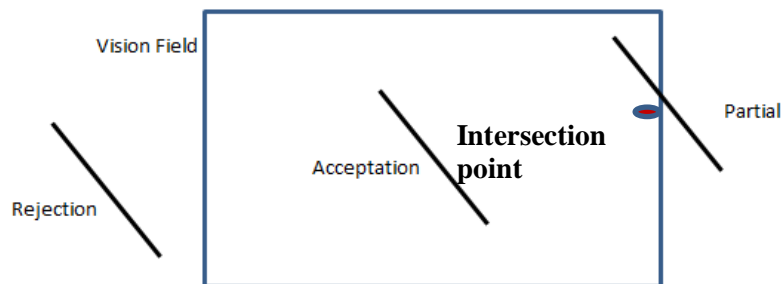


Figure (2): the vision field of Cohen Sutherland clipping algorithm [1].

There are nine areas assigned by code with four bits, explained in the next steps and as shown in the figure (3) [8].

1001	0001	0101
1000	0000 Window	0100
1010	0010	0110

Figure (3) vision field with encoded [1].

1. Given the line with two endpoint vertexes $v1(x1,y1)$, $v2(x2,y2)$
2. Compute 4 bit codes for each endpoint.
3. If both codes are 0000(result of logical OR equal 0000), then the line is completely inside, leading to draw two endpoints.

4. If both codes are not 0000 (result of logical AND not equal 0000), then the line is completely outside, leading to reject it.
5. If both codes are not 0000(result of logical AND equal 0000), then the line has partial status, leading to find intersection point.
 - a. If clipping from above, $x_{new} = x_1 + (x_2 - x_1) * (y_{max\ of\ vision\ field} - y_1) / (y_2 - y_1)$,
 $y_{new} = Y_{max\ of\ vision\ field}$ (1).
 - b. If clipping from bottom, $x_{new} = x_1 + (x_2 - x_1) * (y_{min\ of\ vision\ field} - y_1) / (y_2 - y_1)$,
 $y_{new} = Y_{min\ of\ vision\ field}$ (2).
 - c. If clipping from right, $x_{new} = X_{max\ of\ vision\ field}$, $y_{new} = y_1 + (y_2 - y_1) * (x_{max\ of\ vision\ field} - x_1) / (x_2 - x_1)$ (3).
 - d. If clipping from left, $x_{new} = X_{min\ of\ vision\ field}$, $y_{new} = y_1 + (y_2 - y_1) * (x_{min\ of\ vision\ field} - x_1) / (x_2 - x_1)$ (4) [3].

6- Proposed algorithm

There are many previous works designed to generate 3D models from 2d drawings in less expensive and easier ways. One of these design interface collect many 3D pieces there are chosen by user with 2d layout to design 3D models [2]. The other design 3d models by using the fabrications of 2d photo when apply Circular and Bezier curve [9], etc. But in this paper Cohen Sutherland line clipping algorithm used to design 3D models from 2D drawing by using partial intersection to find one coordinates for all points of 3D model after detect the other coordinate from clipping window in the intersection position.

The proposed algorithm in this paper contains three basic items: vanishing point, 2d input drawing (point, line, square, polygon, etc.) and Cohen Sutherland line clipping algorithm idea.

Algorithm steps:

1. Draw 2d basic points (must be inside limited vision borders).
2. Prepare vanishing point (must be outside limited vision borders).
3. Specify the vision border of all input points depended on direction of vanishing point for any one (represent one of coordinates of depth vertex).
4. Apply Cohen algorithm to find intersection points (find another coordinate of depth vertex)
5. Connect each 2d input with intersection point as analogous.
6. Generate 3d model.

All the input points are connected with the vanishing point to form a set of lines starting from the points of the input shape and fading at the other side. So that 2D shape represent the farthest level relative to the vanishing point and the clipping window is closest level to it. One coordinates of clipping window for all input points represent the amount form these points and represent one coordinates of intersection point as shown in the figure (2). then execute the clipping depended on the location of each point from vanishing point and call one of partial intersection equations (X_{Min} , X_{Max} , Y_{Min} , or Y_{Max}) to find other coordinate of all points to obtain 3d model

7- Experiment results

In this section, many cases of 3D models generation via proposed algorithm will be explained gradually from the inputs, execute the clipping after detect what is the type of partial intersection must be used, until obtain 3D models. And explain many of reached results with discussions. Finally explain a set of steps to generate 3D chess patch with different statuses when apply the proposed algorithm of this paper

Case (1):-

Apply proposed algorithm to generate 3d model (three lines) after Specific 2d drawing (three connected pixels), vanishing point and vision border of any input point

(A):- generate 3d model as shown in the figure (4).

- a. Line1: vanishing point is in the right from Pixel1; we call partial intersection from X_{Max} . $x_{new} = x$ determined specified from vision border, $y_{new} = y1 + (y2 - y1) * (x_{determined} - x1) / (x2 - x1)$,
- b. Line 2: similar in generation of Line 1
- c. Line 3: vanishing point is in the bottom from Pixel3, we call partial intersection from Y_{Min} , $x_{new} = x1 + (x2 - x1) * (y_{determined} - y1) / (y2 - y1)$, $y_{new} = y$ determined from vision border.

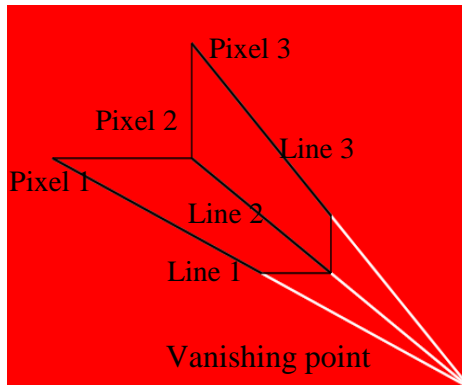


Figure (4): Generate 3d Model, case (1, A)

(B):- change vanishing point place from bottom right in the state (A) to bottom left, 3D model will be generated when apply partial intersection from X_{Min} to find Line1, Line2, and apply partial intersection from Y_{Min} to find Line3, as shown in figure (5).

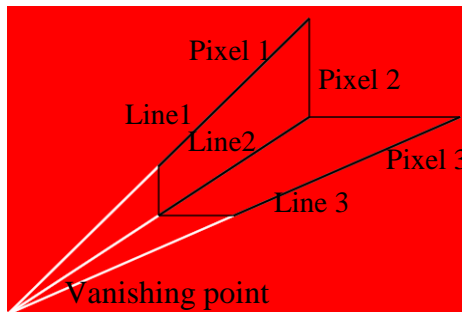


Figure (5): Generate 3d Model, case (1, B).

(C):- generate 3d Model with vanishing point located in the upper right side, as shown in the figure (6).

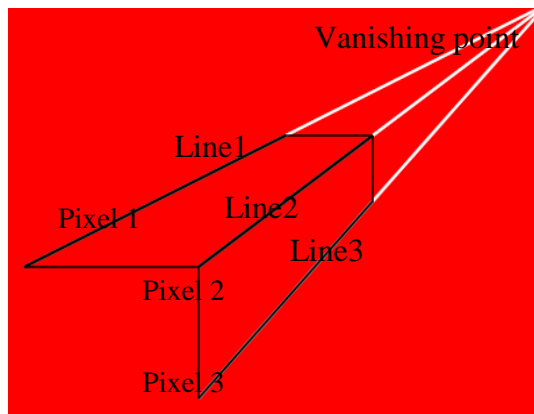


Figure (6): Generate 3d Model, case (1, C).

Case (2):-

Generate 3d model (cube) from 2d drawing (yellow box). Vanishing point located in the upper left side as shown in the figure (7), then Line1, Line2 generated when apply partial intersection from X_{min} , Line3 from Y_{max} .

The same result was obtained during cube generation when change the generation of Line2 from X_{min} to Y_{max} because there are cases observed when generate a cube by proposed algorithm, that is possible obtained same result when apply two states of partial intersection depended on location of vanishing point for input point (if this point is along the vanishing point) when apply the two following conditions:

1. Width of 2d input(square) = it' length in the two states
2. The distance between X value of vision border and X value of input point in the one of generation = distance between Y value of vision border and Y value of input point in the another

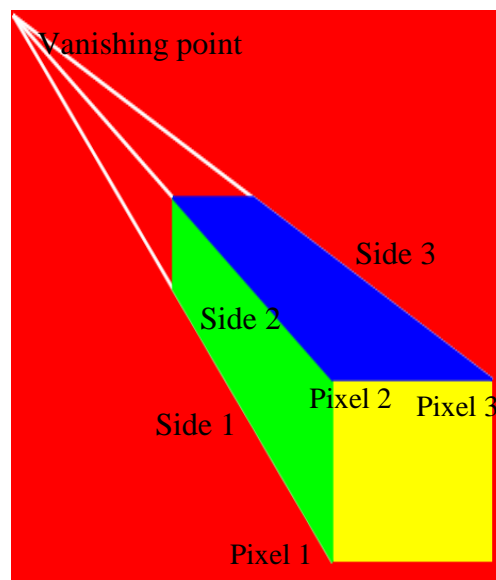


Figure (7): Generate 3d Model (cube).

As mention above when apply previous two conditions, Line2 can be generated with same result with two states of partial intersection. But when not applying one or both conditions, there are mismatch outputs for example figures (8) and (9), show the result is different in two states when it is not applied apply first, second condition as analogous.

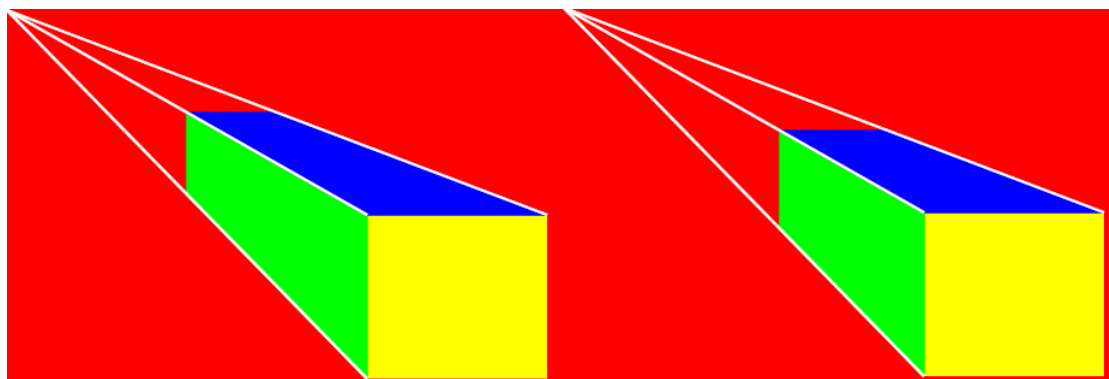


Figure (8): Generate mismatch 3d Models with don't apply first condition

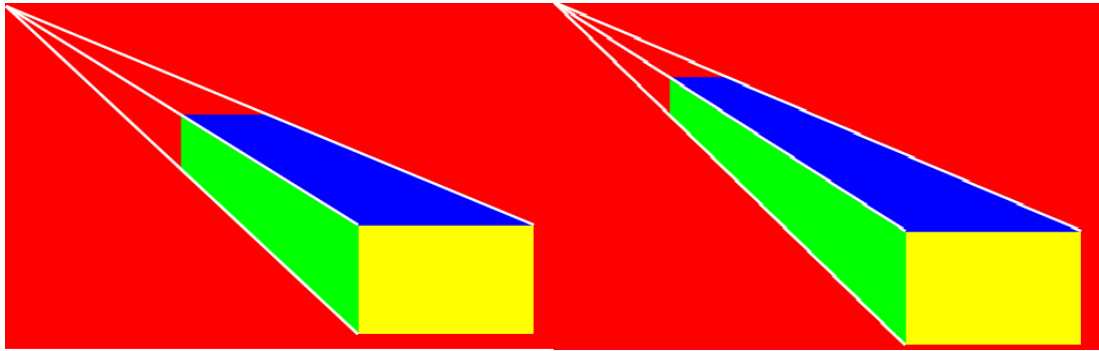


Figure (9): Generate mismatch 3d Models with don't apply second condition

Case (3):-

Generate 3d chess patch when using Cohen Sutherland algorithm not only specifies intersection points but also calculates the length of squares (border of vision field) in each row of chess patch except first row from the bottom, to generate squares with smaller area whenever move away from eye vision form the following steps.

1. Draw bottom lines of squares in the first row (started from bottom).
2. Specify the area (border of vision field) of the first row with vanishing point location.
3. Connect lines of each square in the first row with vanishing point.
4. Find intersection point, lead to generate 3D first row.
5. From second row to Row N , do the following.
 - a. The length of the squares in row n = distance between two 3D generation points of row $n-1$, as shown in the figure (10).
 - b. Repeat steps (3, 4).
 - c. $n = n + 1$
 - d. Repeat steps (a, b, c) until generate row N .

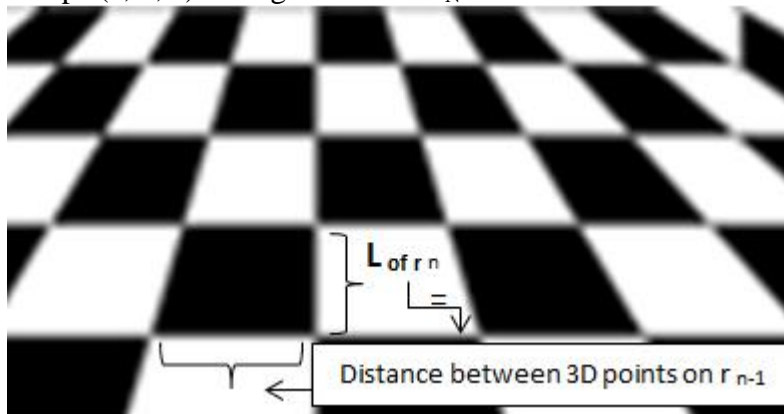


Figure (10): the length of any row equal the generated width distance in previous row

Add vanishing point to 2d square contain mini white and black squares, with applying above steps lead to generate 3D chess patch, as shown in the figure (11).

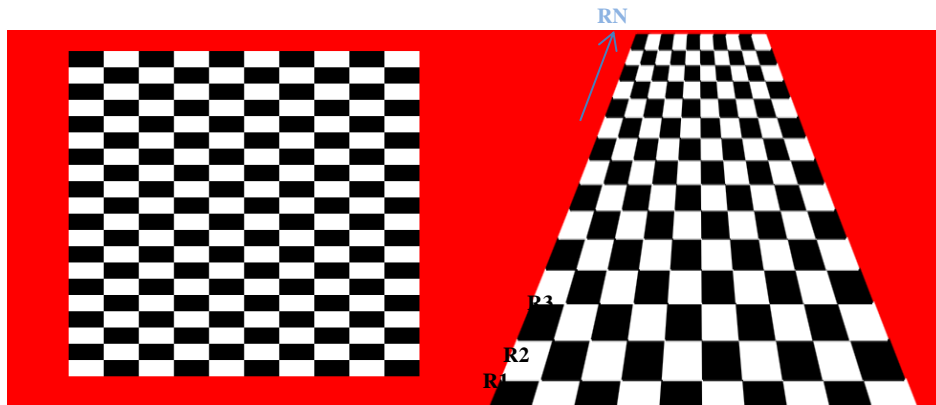


Figure (11): draw 3D model (chess patch).

Upper 3d chess patch was generated when partial intersection is applied from Y_{max} , that is depended on the location of vanishing point for 2d mini white and black squares. So when the vanishing point located bottom of 2D input squares, the 3d Chess patch will be generated with partial intersection from Y_{min} , as shown in upper side of the figure (12). But the shapes in the left and right sides will be generated when call partial intersection from X_{max} , X_{min} as analogous.

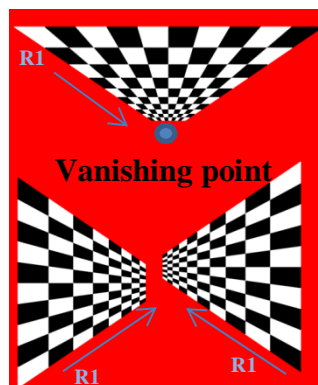


Figure (12): 3D chess patch generated of three designs.

When the four partial intersections (X_{min} , X_{max} , Y_{min} , Y_{max}) of chess patch are applied with same vanishing point and same specifics of first row, lead to generate a model similar a room viewed from out to in, as shown in the figure (13).

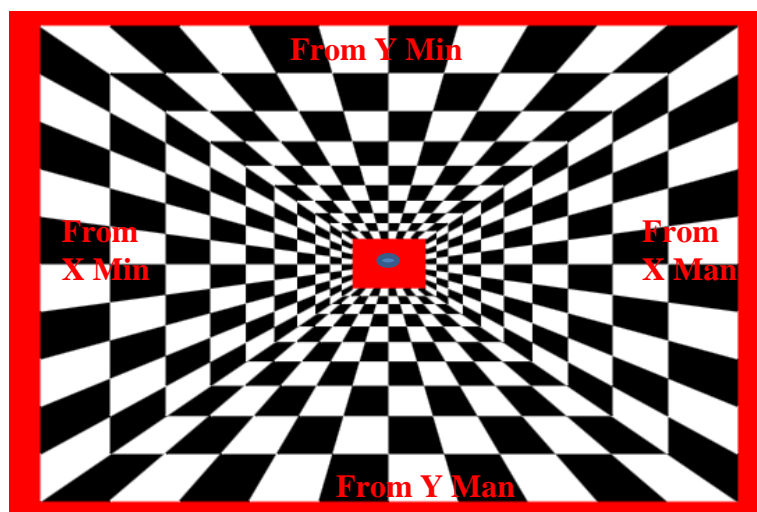


Figure (13): generate chess patch model similar a room.

8- Conclusion

At present, there is a great importance in generating three dimensional models because they are involved in many fields such as games and movies, especially if they are generated at the lowest possible cost, as many benefits obtained when generated them from 2D drawings, the most important of which is the ease and understanding of execution and reduce time required to extract the result. Cohen Sutherland line clipping algorithm was used in this paper for its easy implementing and understanding when prepare vanishing point that must be connected with all 2D input points to form a group of lines, they are clipped when apply partial intersection to obtain one of the depth coordinates of all points after specify limited view to generate 3D Model with easiest possible ways.

REFERENCES

- [1] Hui S, WANG Y., LI Y., and LI B., 2012, "Road conditions test in virtual driving system based on Cohen Sutherland algorithm", © (2012) Trans Tech Publications, Switzerland, www.scientific.net, Advanced Materials Research Vols 532-533 (2012) pp 1147-1151
- [2] Lee J., Funkhouser T., 2008, "Sketch-Based Search and Composition of 3D Models", EUROGRAPHICS Workshop on Sketch-Based Interfaces and Modeling (2008), C. Alvarado and M- P. Cani (Editors), © The Euro graphics Association 2008, ISBN: 978-3-905674-07-1.
- [3] LI H., ZUO D., 2009, "The research of security-mode based on Cohen Sutherland Encoding", IEEE explorer digital library, ISBN: 978-1-4244-3691-0.
- [4] NOÉ E., 2017, "3D layered articulated object from a single 2D drawing", Master of Science in Engineering in Media Technology, Kth Royal Institute of Technology School of Computer Science and Communication.
- [5] Rivers A., Durand F., and Igarashi T., 2010, "3D Modeling with Silhouettes", Association for Computing Machinery (ACM), New York, NY, USA, ISBN: 978-1-4503-0210-4.
- [6] Salomon D., 2006, "Transformations and Projections in Computer Graphics", British Library Cataloguing in Publication Data, ISBN-10: 1-84628-392-2 ISBN-13: 978-1-84628-392.
- [7] Shreiner D., 2010, "OpenGL® Programming Guide", Pearson Education, Inc, Rights and Contracts Department, Copyright © 2010 Pearson Education, ISBN:0201604582.
- [8] Wei W., Ma P., and Lin W, 2013, "An improved Cohen-Sutherland region encoding algorithm", Applied Mechanics and Materials Vols. 239-240 (2012) pp 1313-1317, © (2013) Trans Tech Publications, Switzerland, doi:10.4028/www.scientific.net/AMM.239-240.1313.
- [9] Zhu J., Zhang H., and Wen Y., 2014, "A new reconstruction method for 3D buildings from 2D vector floor plan", computer- Aided design and application, Volume 11, Issue 4.

[10] الهاللي، اياد 2004، "دليل المبرمج الى OpenGL"، البراق للطباعة والنشر والتوزيع، دار القلم العربي للطباعة والنشر والتوزيع، جميع الحقوق مطبوعة©2004، سوريا حلب، رقم التصنيف 006.6869 هـ. ل.

.ا