Intelligent scene understanding using geometry and lighting

Cyril LAM, Dimitri PLEMENOS
University of Limoges, XLIM laboratory, 83, rue d'Isle, 87000 Limoges (France)
lam@msi.unilim.fr, plemenos@unilim.fr

Abstract
In this paper we present techniques allowing to take into account the lighting in intelligent scene understanding and exploration. Starting from evaluation functions currently proposed to estimate the geometry-based quality of a point of view, we introduce illumination-based criteria and combine them in order to give the camera more realistic decision criteria during scene exploration. The first results obtained with our methods seem fully satisfactory and show that lighting parameters are well integrated in the viewpoint quality estimation process.
Keywords: *Good point of view, Heuristic search, Scene exploration, Lighting.*

1. Introduction

The problem of understanding a scene is currently a more and more pertinent problem because of the development of web applications and possibilities, for a user, to discover new, never seen, scenes on the net, which are generally difficult to well understand without a tool able to evaluate the pertinence of a view, to choose a good view for each world and even to allow to explore it with a virtual camera.
With the fast development of computers capabilities this last decade, the problem of well understanding complex virtual worlds becomes more and more crucial and several recent papers try to take into account this problem.
In order to evaluate the pertinence of a view, current techniques consider that geometry and topology of a virtual world are important elements to take into account but the problem is how to do it. The problem is difficult to resolve because its solution consists to quantify the human perception of an image.
Even if the real human perception is not perfectly taken into account with current visual pertinence evaluation techniques, these techniques give generally results good enough allowing to apply these techniques in several areas, such as computer games, virtual museums visiting, molecules visualisation or realistic rendering.
However, it is easy to understand that the only knowledge of geometry and topology of a scene is not enough to allow precise quantification of the human perception. If the virtual world is illuminated, it is important to take into account illumination of its elements in evaluation of visual pertinence. Everybody knows that, even if there are lots of pertinent details in a dark room, no one of them is visible and it is not pertinent to choose a point of view allowing to look inside the dark room.
In this paper we will present new techniques allowing to take into account the lighting parameters of a scene, together with its geometry, in order to get more pertinent criteria for choosing viewpoints and exploration paths allowing to understand this scene. For the moment it is supposed here that the camera always remains outside the scene.

The paper will be organised as follows: In section 2 a review of the main geometry-based proposed techniques for estimating the visual pertinence of a view and for understanding a virtual world will be presented. In section 3, new scene understanding techniques, allowing to take into account both geometry and lighting to estimate the quality of a view, will be presented. Section 4 will be dedicated to the presentation of first results obtained with the proposed new techniques. Critical evaluation of these results will be presented as well. In section 6 conclusions on the new proposed techniques to improve the visual pertinence of a view, in order to better understand a scene, will be given, as well as some indication on possible future work.

2. Geometry-based techniques

The very first works in the area of understanding virtual worlds were published at the end of 80's and the beginning of 90's. There were very few works because the computer graphics community was not convinced that this area was important for computer graphics. The purpose of these works was to offer the user a help to understand simple virtual worlds by computing a good point of view.

2.1 Best view computing for simple virtual worlds

When the virtual world to understand is simple enough, a single view of it may be enough to understand the virtual world. So, it is important to be able to propose an automatic computation of a "good" viewpoint.

Kamada et al. [1] consider a position as a good point of view if it minimises the number of degenerated images of objects when the scene is projected orthogonally. A degenerated image is an image where more than one edges belong to the same straight line.

The used method avoids the directions parallel to planes defined by pairs of edges of the scene.

The technique proposed by Kamada is very interesting for a wire-frame display. However it is not very useful for a more realistic display. Indeed, this technique does not take into account visibility of the elements of the considered scene and a big element of the scene may hide all the others in the final display.

The good point of view computing method proposed by Plemenos [2, 3] was developed and implemented in 1987 but it was first published only in 1991.

The good view criterion used by this method is the number of visible details combined with the projected area of the visible parts of the scene. More precisely, the importance of a point of view will be computed using the following equation:

$$I(V) = \frac{\sum_{i=1}^{n} \left[\frac{P_i(V)}{P_i(V)+1}\right]}{n} + \frac{\sum_{i=1}^{n} P_i(V)}{r} \qquad (1)$$

where:

I(V) is the importance of the view point V,

Pi(V) is the projected visible area of the polygon number i obtained from the point of view V,

r is the total projected area,

n is the total number of polygons of the scene.

In this equation, [x] denotes the smallest integer, greater than or equal to x, for any quantity x.

In practice, these measures are computed in a simple manner, with the aid of graphics hardware using OpenGL [5, 11]. A different color is assigned to every face, an image of the scene is computed using integrated z-buffer and a histogram of the image is computed. This histogram gives all information about the number of visible polygons and visible projected area of each polygon.

The process used to determine a good point of view works as follows:

The points of view are supposed to be on the surface of a virtual sphere whose the scene is the centre. The surface of the sphere of points of view is divided in 8 spherical triangles.

The best spherical triangle is determined by positioning the camera at each intersection point of the three main axes with the sphere and computing its importance as a point of view. The three intersection points with the best evaluation are selected. These three points on the sphere determine a spherical triangle, selected as the best one.

To select the best viewpoint on the best spherical triangle the following *heuristic search* technique is used:

If the vertex A (Figure 1) is the vertex with the best evaluation of the spherical triangle ABC, two new vertices E and F are chosen at the middles of the edges AB and AC respectively and the new spherical triangle ADE becomes the current spherical triangle. This process is recursively repeated until the quality of obtained points of view does not increase. The vertex of the final spherical triangle with the best evaluation is chosen as the best point of view.

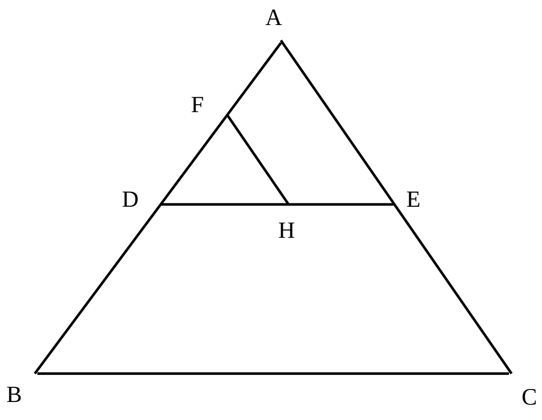


Figure 1: Heuristic search of the best point of view by subdivision of a spherical triangle

Colin [4] proposed a method to compute a good view for octree models. This method computes the "best" initial spherical triangle and then the "best" viewpoint is approximately estimated on the chosen triangle.

Sbert et al. [6] proposed to use information theory in order to establish an accurate criterion for the quality of a point of view. A new measure is used to evaluate the amount of information captured from a given point of view. This measure is called *viewpoint entropy*. To define it, the authors use the relative area of the projected faces over the sphere of directions centred in the point of view.

The best viewpoint is defined as the one that has the maximum entropy.

The selection of the best view of a scene is computed by measuring the viewpoint entropy of a set of points placed over a sphere that bounds the scene.

2.2 Virtual World Exploration

When we have to understand a complex virtual world, the knowledge of a single point of view is not enough to understand it. Computing more than one points of view is generally not a satisfactory solution in most cases because the transition from a point of view to another one can disconcert the user, especially when the new point of view is far from the current one. Of course, the knowledge of several points of view can be used in other areas of computer graphics, such as image-based modelling and rendering [10, 14] but it is not suitable for virtual world understanding. The best solution, in the case of complex virtual worlds is to offer an automatic exploration of the virtual world by a camera that chooses its path according to the specificities of the world to understand.

An important problem in automatic virtual world exploration is to make the camera able to visit the world to explore by using good points of view and, at the same time, by choosing a path that avoids brusque changes of direction.

In [5, 7] an initial idea of D. Plemenos and its implementations are described. The main principle of the proposed virtual world exploration technique is that the camera's movement must apply the following heuristic rules:

- It is important that the camera moves on positions which are good points of view.
- The camera must avoid fast returns to the starting point or to already visited points.
- The camera's path must be as smooth as possible in order to allow the user to well understand the explored world. A movement with brusque changes of direction is confusing for the user and must be avoided.

In order to apply these heuristic rules, the next position of the camera is computed in the following way:

- The best point of view is chosen as the starting position for exploration.
- Given the current position and the current direction of the camera, only directions insuring smooth movement are considered in computing the next position of the camera.
- In order to avoid fast returns of the camera to the starting position, the importance of the distance of the camera from the starting position is inversely proportional to the path of the camera from the starting to the current position.

Vazquez et al. [9, 10] use a similar method for outside and indoor exploration of a virtual world. They use the viewpoint entropy to compute the pertinence of a view.

2.3 More accurate definition of viewpoint complexity

Most of the better known methods using the notion of viewpoint complexity to evaluate the pertinence of a view are based to two main geometric criteria: number of visible polygons and area of the projected visible part of the scene. Thus, equation (1) of section 2 is often used to evaluate the viewpoint complexity for a given scene.

However, even if the methods using these criteria give generally interesting results, the number of polygons criterion may produce some drawbacks. Indeed, let us consider a scene made from a single polygon. This polygon may be subdivided in several other polygons and, in such a case, the number of visible polygons will depend on the number of subdivisions of the initial polygon. A viewpoint complexity evaluation function will give different results for the same scene, according to its subdivision degree.

In order to avoid this drawback, another criterion was proposed by Sokolov et al. [15, 16], which takes into account the curvature of the scene. More precisely, the *number of polygon* criterion is replaced by the criterion of *total curvature* of the scene. The total curvature of a scene is the sum of curvatures in all vertices of the scene.

The main advantage of the proposed criterion is that it is invariant to any subdivision of the scene elements maintaining the topology. Another advantage is that it can be extended in order to use the total integral curvature of curved surfaces.

The authors also propose a method to compute a pertinent trajectory for off-line exploration of the scene by a virtual camera. The trajectory is computed by assigning, at each step, a mass proportional to the number of new visible details, to each point of the discrete surrounding sphere. The superposition of gravitational forces for the camera current position is the vector of movement.

Another method to compute a minimal set of good viewpoints in order to define a camera trajectory for off-line scene exploration was proposed by Jaubert et al. [18, 19]. In this method a sufficient number of viewpoints is computed first and then the minimal set of good viewpoints is created by successively suppressing viewpoints which do not allow to see more details than the remaining ones.

The concept of *mesh saliency* was introduced in 2005 by Chang Ha Lee et al. [17]. The goal of this concept is to bring perception-based metrics in evaluation of the pertinence of a view. According to the authors, a high-curvature spike in the middle of a largely flat region is perceived to be as important as a flat region in the middle of densely repeated high-curvature bumps.

Mesh saliency is defined as the absolute difference between the Gaussian-weighted averages computed in fine and coarse scales.

The authors use mesh saliency to compute interesting points of view for a scene. The shown examples seem interesting. An important advantage of the method is that the notion of mesh saliency is defined and may be computed at multiple scales.

3. Scene understanding and lighting.

What is the lighting problem? There are rather two different problems which have to be resolved in different manners. The first problem is *absolute light source placement* and the second one is *taking into account light source position.*

3.1 Absolute light source placement

The problem is how to compute light source(s) position(s) in order to illuminate a scene in optimal manner. The resolution of this problem does not depend on the camera position. A good illumination of the scene should allow easier understanding by the user, if a camera explores the scene.

In the simple case of a single punctual light source, if only direct lighting is considered, the problem may be resolved in the same manner as the camera placement problem. What we have to do is to look for the best viewpoint from the light source.

In the general case, the problem is much more complex. Available today methods are not satisfactory. Most of them are based on inverse lighting techniques, where light source positions are deducted from the expected result. However, methods proposed by Poulingeas et al. [20] and Poulin et al. [21, 22] are not entirely satisfactory, especially because it is not easy to well describe and formalise the expected results.

Design Galleries [23] is a general system to compute parameters for computer graphics but computation is not fully automatic. Another not fully automatic system to compute light source positions is presented in [24]. The method presented in [13] is based on the notion of *light entropy* and automatically computes lighting parameters but results are not entirely satisfactory without the help of the user.

For a fixed point of view the problem is to find an optimal position for the light source, in order to better understand the scene. For a punctual light source, if we have a function automatically computing the quality of a point of view by taking into account not only the geometry of the scene but also lighting, it is possible to compute the best position for the light source by using the heuristic search described in 2.1, where the surrounding sphere of the scene is divided in 8 spherical triangles and the best one is subdivided in order to progressively reach a position with the best evaluation. This position will be the optimal light source position. In subsection 3.2, the problem of finding a viewpoint evaluation function allowing to take into account both geometry and lighting will be discussed.

3.2 Taking into account light source position

Up to now we have considered that the quality of a viewpoint is based on the geometry of the scene to be seen. However, a scene is often illuminated and several details, considered important according to the scene geometry, may be not visible for a given position of the light source, because they are shadowed. It is clear that, in such a case, it is important to take into account lighting in the computation of the quality of view from a viewpoint. If the number of scene details seen from a point of view is important, lighting of each visible detail has to be taken into account.

The problem of taking into account light source placement is quite different from the absolute source placement problem. Here the purpose is to take into account light source position in order to compute more precisely the pertinence of a view. The question to answer is: *Given a viewpoint P and a light source position L, how to compute the pertinence of the view from this viewpoint?* The problem is difficult to resolve in the general case but solutions may be proposed for some simpler cases.

Thus, Vazquez et al. [25] have proposed a perception-based measure of the illumination information of a fixed view. This measure uses Information Theory concepts. The authors use, as unit of information, the relative area of each region whose colour is different from its surrounding.

3.2.1 A naive first approach

It is possible to propose a method to compute the pertinence of a given view, taking into account the position of one (or more) punctual light source for direct lighting. This method is inspired from the method of viewpoint evaluation used in [5] and [11]. The general idea of this method was proposed by Plemenos et al. in [27]. We have already seen that in the method proposed in [5] and [11], equation (1) is used to compute de viewpoint quality. In order to compute information needed by this equation, OpenGL and its integrated z-buffer is used as follows:

A distinct colour is given to each surface of the scene and the display of the scene using OpenGL allows to obtain a histogram (Figure 2) which gives information on the number of displayed colours and the ratio of the image space occupied by each colour. As each surface has a distinct colour, the number of displayed colours is the number of visible surfaces of the scene from the current position of the camera. The ratio of the image space occupied by a colour is the area of the projection of the viewpoint part of the corresponding surface. The sum of these ratios is the projected area of the visible part of the scene. With this technique, the two viewpoint complexity criteria are computed directly by means of an integrated fast display method.

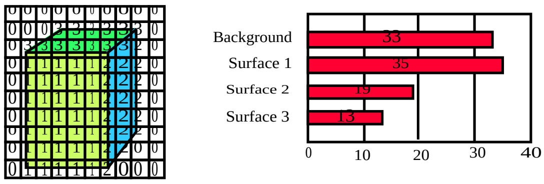


Figure 2: Fast computation of number of visible surfaces and area of projected viewpoint part of the scene by image analysis.

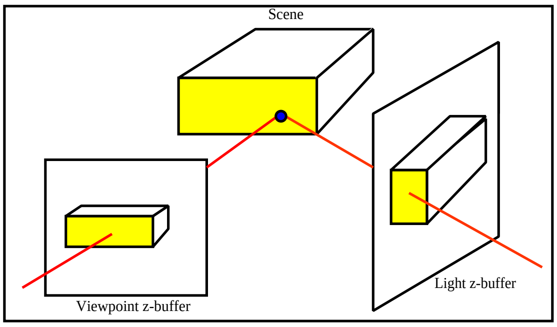


Figure 3: Two z-buffers to estimate the quality of a viewpoint taking into account lighting of the scene.

Let us suppose that equation (1) is used to compute the quality of a point of view when only the geometry of the scene is used. In order to get an accurate estimation of the quality of view of a polygon of the scene from a given point of view it is important to integrate the quality of lighting of this polygon. A simple method to do this is to consider the angle of lighting from the light source to, say, the centre of the polygon and to introduce the cosine of this angle in equation (1).

In practice we can use two z-buffers, one from the point of view and one from the light source and approximate the cosine of the angle with the projected area of the polygon from the light source position (Figure 3). For example, in equation (1), the considered visible projected area for a polygon will be the average value between the really visible projected area and the visible projected area if the light source position is taken as the centre of projection. That is, the number of pixels corresponding to the colour of the polygon of the first z-buffer will be added to the corresponding number of pixels of the second z-buffer and divided by 2.

In this method we use only the information that a polygon is lighted or not.

More generally, the formula to compute the quality of a point of view, taking into account the lighting, could be the following:

$$I_L(P, S) = \frac{\alpha\, I(P) + \beta\, I(S)}{\alpha + \beta}$$

where:

$I_L(P,S)$ is the global importance of a point of view P taking into account the light source S.

I(X) is the geometry-based importance of the point of view X.

α and β are coefficients used to refine the respective contribution of geometry and illumination of the virtual world.

This method may be easily generalised for n punctual light sources.

3.2.2 Refining the method

The formula proposed in the previous subsection is a very general one. We are going to refine it in three manners:

1. The geometry-based importance of a point of view will be decomposed in number of visible polygons and projected visible area.
2. The relative importance of number of visible polygons and projected visible area will be made more precise.
3. The projected area of the visible part of the scene will be expressed in number of pixels, by explicitly using the OpenGL histogram, as explained above.

So, the final formula used to take into account lighting is the following:

$$I_L(P) = n\, p_e(P)\, c + n\, p_v(P)$$

Where:

$I_L(P)$ is the global importance of a point of view P taking into account the lighting.

n is the number of visible polygons of the scene from the viewpoint P.

$p_v(P)$ is the number of visible pixels from the viewpoint P.

$p_e(P)$ is the number of lighted pixels, visible from the viewpoint P.

c is a coefficient allowing to modify the importance of the lighting part of the formula. The best results are obtained with a value of about 1.9 for c.

In this formula, the light source position doesn't appear explicitly. However, it is implicitly expressed by the number of lighted pixels.

In this method, only direct lighting is taken into account.

3.2.3 Indirect lighting

In the methods presented in the previous subsections, only direct lighting is taken into account. Even if obtained results are satisfactory, sometimes, when a higher degree of precision is required, it is necessary to take into account indirect lighting. In such a case, the use of the OpenGL z-buffer is not sufficient. Other techniques, such as ray tracing or radiosity must be used. In this subsection new good view criterion, taking into account both direct and indirect lighting, is proposed, together with a method to evaluate the quality of a view.

The formula evaluating the view of a scene from a given viewpoint P becomes now:

$$I_L(P) = n\,(p_v(P) + c_1\, p_e(P) + c_2\, p_r(P) + c_3\, p_t(P))$$

Where:

I_L(P) is the global importance of a point of view P taking into account direct and indirect lighting.
n is the number of visible polygons of the scene from the viewpoint P.
p_v(P) is the total number of directly visible pixels from the viewpoint P.
p_e(P) is the number of both directly and indirectly lighted pixels, visible from the viewpoint P.
p_r(P) is the number of indirectly visible pixels from the viewpoint P by reflection.
P_t(P) is the number of indirectly visible pixels from the viewpoint P by transparency.
c_1, **c_2** and **c_3** are coefficients allowing to modify the importance of indirect view and lighting. The best results are obtained with c_1=2, c_2=1 and c_3=1.
A ray tracing algorithm is used to evaluate the quality of a view. For each reflected or transmitted by transparency ray, if the ray intersects a polygon, p_r(P) or p_t(P) is incremented. If the indirectly visible polygon is lighted, p_e(P) is also incremented.

3.2.4 Automatic computation of the best viewpoint

For a scene and a fixed punctual light source it is now easy to compute the best viewpoint, using one of the viewpoint quality criteria proposed in this subsection. The *heuristic search* method proposed by Plemenos et al. [3] may be used again. The scene is placed at the centre of a surrounding sphere whose surface contains all possible points of view. The sphere is initially divided in 8 spherical triangles and the best one, according to the used criterion, is selected and subdivided as long as the quality of obtained viewpoints is improved (see Figure 1).
If neither light source nor viewpoint is fixed for the scene, the problem is to find on the surface of the surrounding sphere the couple (viewpoint, light source) which ensures the best evaluation, that is, the best view. There doesn't exist efficient manner to do this, even if we work on a discrete surface of the surrounding sphere. Every possible viewpoint has to be combined with all possible positions for the light source and the process is very time consuming.

3.2.5 Automatic scene exploration

For a scene and a fixed punctual light source, a variant of the method proposed in [5, 7] for incremental online exploration may be used. The best viewpoint position becomes the initial camera position and, for a given position of the camera, only a reduced number of next positions are considered, in order to warrantee smooth exploration, free of brusque changes of direction. Scene understanding may be improved by using plan-based incremental exploration proposed by S. Desroche et al. [26].
The problem with online exploration is that it supposes real time evaluation of a viewpoint. Exploration taking into account lighting is not yet implemented but we can already know that only offline exploration will be possible if indirect lighting is taken into account, as presented in subsection 3.2.3.

4. Results

Methods and heuristics presented in section 3 have been implemented and obtained results allow to conclude that it is possible to integrate lighting parameters in estimation of the quality of a point of view.
In Figures 4 to 7 one can see results obtained with the viewpoint quality criterion presented in subsection 3.2.2. In Figure 4 the triceratops scene is seen from the computed best viewpoint with the best lighting, whereas in Figure 5 the same scene is seen from the same viewpoint with different positions of the light source. It is easy to see that the chosen viewpoint and light position give an interesting view of the scene.

In Figure 6 the best view of the office scene is presented. In this scene the point of view is fixed and the viewpoint quality depends on the position of the light source. In Figure 7, two other views of the office scene are presented with different positions of the light source. The point of view selected in Figure 6 is not really the best one. What is presented is the best view from a chosen point of view, that is, the best position of the light source for this point of view. In Figure 7 are presented other views from the same point of view but with different positions of the light source.

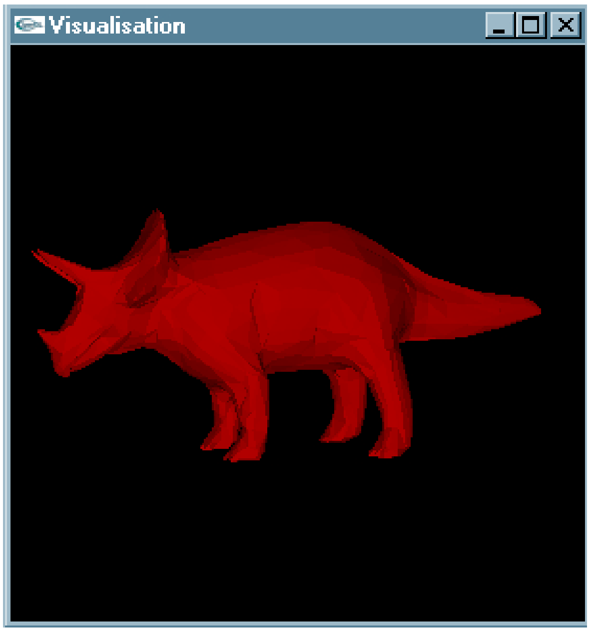
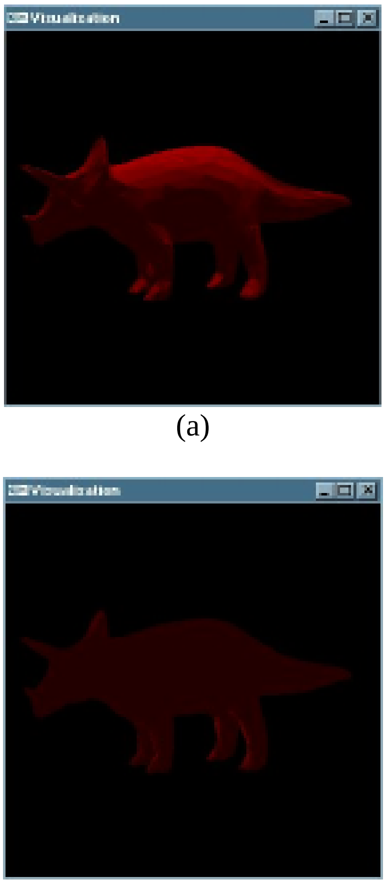


Figure 4: Best view for the triceratops scene.

The method of automatic computation of a good point of view based on the quality criterion of subsection 2.2.3 gives interesting and fast results, not dependent on the scene complexity by using the OpenGL z-buffer.

The main drawback of the method is that only direct lighting is taken into account. This limitation could imply less precision in results when considering scenes with many reflections and refractions.



(a)

(b)

Figure 5: Other views for the triceratops scene.

In Figure 8, the best view is presented for the best viewpoint computation method based on the viewpoint quality criterion presented in subsection 3.2.4 and allowing to take into account direct and indirect lighting. In Figure 9, two other, less interesting, views are presented. A ray tracing algorithm is used to compute the evaluation function.

The method based on the viewpoint quality criterion integrating indirect lighting gives very interesting results but it is very time consuming, as it uses a ray tracing algorithm to compute reflections and refractions. For this reason it is difficult to use in online scene exploration.

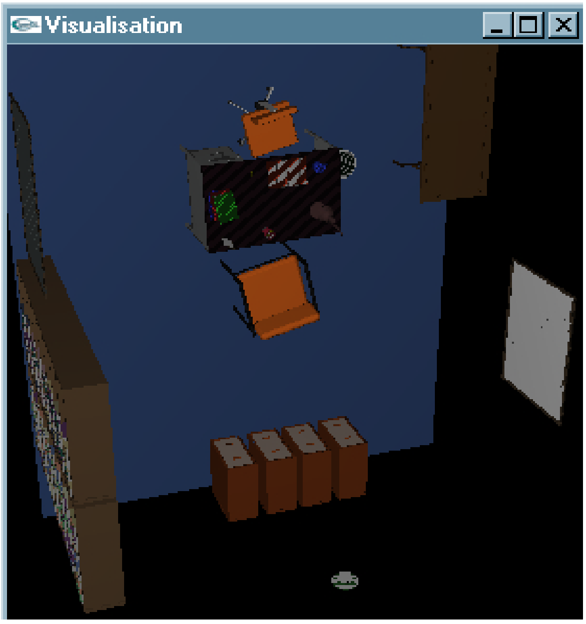
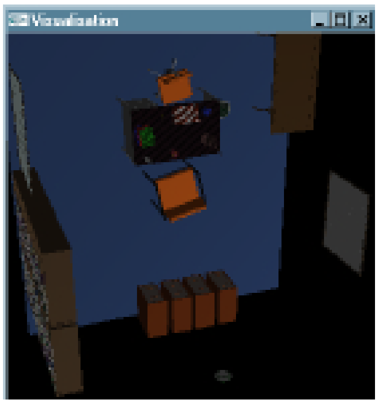


Figure 6: Best view for the office scene.



(a)

(b)

Figure 7: Other views for the office scene

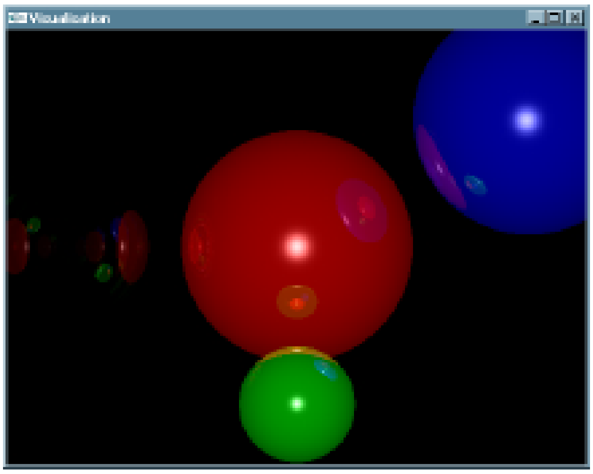
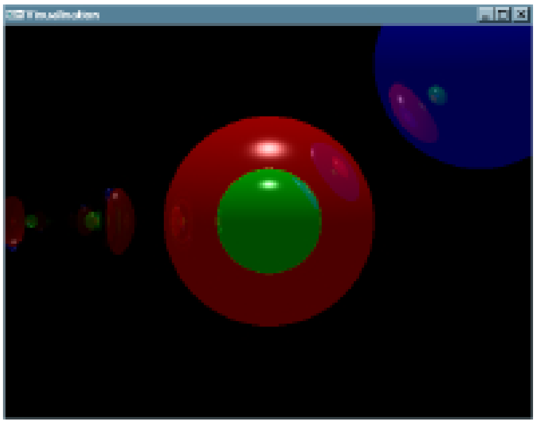
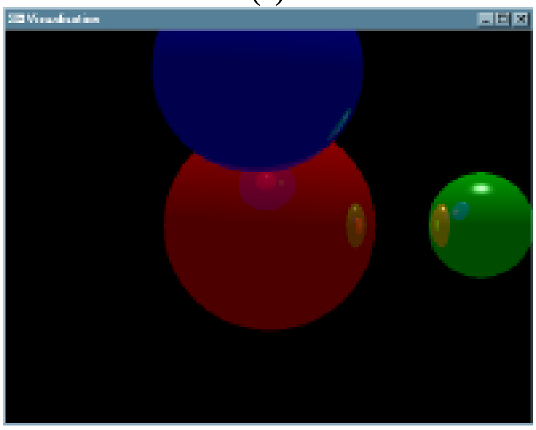


Figure 8: Best view for a scene with reflections.



(a)



(b)

Figure 9: Other views for a scene with reflections

5. Conclusion and future work

In this paper, after a presentation of the main geometry-based methods allowing to get a good view of a scene for a fast understanding, we have presented new proposals permitting to take into account not only the geometry of a scene but also its lighting. Indeed, lighting is very important to make possible to a human user to understand a scene.

Two viewpoint quality criteria were proposed. The first one supposes that the scene is lighted by one (or many) punctual light source(s). Only direct lighting is considered. The method computing a good viewpoint based on this criterion gives interesting results and is fast enough to be used in online scene exploration. The second viewpoint quality criterion supposes that the scene is lighted by one or more punctual light sources and takes into account both direct and indirect lighting. The method using this viewpoint quality criterion uses a ray tracing algorithm in order to compute indirect lighting. This method allows to obtain very precise results but its drawback is that it is very time consuming, that is difficult to be used in online scene exploration. Of course, it is possible to use this method in offline computation of a path for future exploration.

The proposed methods allow to mix geometric and lighting criteria in evaluation of a view. However, for the moment only punctual light sources are considered and taken into account by the proposed viewpoint quality criteria. It would be interesting to integrate non punctual light sources for more accurate results with realistic scenes.

6. Acknowledgements

This work has been supported and financed in part with funds of the European project GameTools. It has also been partly supported and financed by the Limousin Region (France). The authors would like to thank all people and organisations which have supported in any manner this project.

7. References

[1] KAMADA T., KAWAI S., A Simple Method for Computing General Position in Displaying Three-dimensional Objects, Computer Vision, Graphics and Image Processing, 41 (1988).

[2] PLEMENOS D., A contribution to the study and development of scene modelling, generation and visualisation techniques. The MultiFormes project., Professorial dissertation, Nantes (France), November 1991.

[3] PLEMENOS D., BENAYADA M., Intelligent display in scene modeling. New techniques to automatically compute good views., GraphiCon'96, Saint Petersburg, July 1996.

[4] COLIN C., A System for Exploring the Universe of Polyhedral Shapes., Eurographics'88, Nice (France), Sept. 1988.

[5] BARRAL P., DORME G., PLEMENOS D., Visual understanding of a scene by automatic movement of a camera, GraphiCon'99, Moscow (Russia), August 26 - September 3, 1999.

[6] SBERT M., FEIXAS M., RIGAU J., CASTRO F., VAZQUEZ P.-P., Applications of the information theory to computer graphics, International Conference 3IA'2002, Limoges (France), May 14-15, 2002.

[7] BARRAL P., DORME G. PLEMENOS D., Scene understanding techniques using a virtual camera, Short paper, Eurographics'2000, Interlagen (Switzerland), August 20 - 25, 2000.

[8] JARDILLIER F., LANGUENOU E., Screen-Space Constraints for Camera Movements: the Virtual Cameraman., Computer Graphics Forum, Volume 17, number 3, 1998.

[9] VAZQUEZ P.-P., SBERT M., Automatic indoor scene exploration, proceedings of the International Conference 3IA'2003, Limoges (France), May 14-15, 2003.

[10] VAZQUEZ P.-P., On the selection of good views and its application to computer graphics., PhD Thesis, Barcelona (Spain), May 26, 2003.

[11] DORME G., Study and implementation of 3D scene understanding techniques., PhD thesis, University of Limoges (France), June 2001. In French.

[12] MARCHAND E., COURTY N., Image-based virtual camera motion strategies., Graphics Interface Conference, GI'2000, Montreal (Canada), May 2000.

[13] GUMHOLD S., Maximum entropy light source placement., IEEE Visualization, October 27 - November 1, 2002.

[14] VAZQUEZ P.-P., FEIXAS M., HEIDRICH W., Image-based modeling using viewpoint entropy. In J. Vince and R. A. Earnshaw, editors, Advances in Modelling, Animation and Rendering (proc. Computer Graphics International), pages 267-279, Springer 2002.

[15] SOKOLOV D., PLEMENOS D., Viewpoint quality and scene understanding. In VAST 2005 Eurographics Symposium proceedings, pp. 67-73, Pisa, Italy (2005).

[16] SOKOLOV D., PLEMENOS D. ,, TAMINE K., Methods and data structures for virtual world exploration. "The Visual Computer", vol. 22, no 7, July 2006.

[17] LEE C. H., VARSHNEY A., JACOBS D. W., Mesh saliency. ACM TOG, 24(3), 659-666 (2005).

[18] PLEMENOS D., GRASSET J., JAUBERT B., TAMINE K., Intelligent visibility-based 3D scene processing techniques for computer games., GraphiCon'2005, Novosibirsk (Russia), June 2005.

[19] JAUBERT B., TAMINE K., PLEMENOS D., Techniques for off-line exploration using a virtual camera. International Conference 3IA'2006, Limoges (France), May 23 – 24, 2006.

[20] JOLIVET V., PLEMENOS D., POULINGEAS P., Inverse direct lighting with a Monte Carlo method and declarative modelling. Lecture Notes in Computer Science, 2002.

[21] POULIN P., FOURNIER A., Lights from highlights and shadows. Computer Graphics 25, 2, March 1992.

[22] POULIN P., RATIB K., JACQUES M., Sketching shadows and highlights to position lights. Computer Graphics International 97, June 1997.

[23] MARKS J., ANDALMAN B., BEARDSLEY P. A., FREEMAN W., GIBSON S., HODGINS J., KANG T., MIRTICH B., PFISTER H., RUML W., RYALL K., SEIMS J., SHIEBER S., Design galleries: A general approach to setting parameters for computer graphics and animation. International Conference SIGGRAPH 97.

[24] HALLE M., MENG J., Lightkit: A lighting system for effective visualization. IEEE Visualization (2003).

[25] VAZQUEZ P. –P. and SBERT M., Perception-based illumination information measurement and light source placement. Lecture Notes in Computer Science, 2003 (Proc. of ICCSA'2003).

[26] DESROCHE S., JOLIVET V., PLEMENOS D., Towards plan-based automatic exploration of virtual worlds. International Conference WSCG'2007, Plzen, Czech Republic, January 29 – February 2, 2007.

[27] PLEMENOS D., SOKOLOV D., Intelligent scene display and exploration. STAR Report. International Conference GraphiCon'2006, Novosibirsk (Russia), June 30- July 5, 2006.