CAMERA POSITIONING SUPPORT BASED ON POTENTIAL FIELD IN A STATIC VIRTUAL ENVIRONMENT

Adam Wojciechowski
Institute of Computer Science, Technical University of Lodz; adamwoj@ics.p.lodz.pl

Abstract: This paper introduces a new method of camera movement support. The task is aided by means of innovatively generated potential field [7]. A new construction of the potentials lets the camera be attracted by implicit goals and simultaneously prevents the observer from getting into the object – collisions detection. Paper introduces also a new way of evaluating final position of the camera to let the user come up to a visually selected object quite smoothly. It reduces discrepancy between movement trail and the user's intentions.

Key words: camera navigation, virtual environment, potential fields, collisions detection

1.   INTRODUCTION

In a virtual environment activities connected with camera (point of view) movements reflect the user's way of environment penetration. However, already proposed applications either suggest predefined or precalculated path camera steering or leave the camera control up to the user.

In this paper navigation support method is presented. It encompasses co-existence of user-intention movements and intelligent support while visually aimed goal achievement. This method is designed to let the user reach implicit goal while navigating quite freely without forcing certain way to the aim. It is also transparent to the user and concurrent with the user's movement intentions. Additionally the user is able to deal with rapidly changing interests in different objects of the scene and is able to reach

adequate position and orientation to present the resulting objects in their environment at their usual context and position.

In section 2 requirements for navigation in virtual environment are discussed and the context of necessity of the user's movement support is explained. Additionally section 2 deals with elementary definitions used in further discussions. Section 3 describes related works. Camera movements control under the influence of potential fields is presented in section 4. In the final section conclusions and ideas for future work are discussed.

2.   REQUIREMENTS FOR USER'S MOVEMENT SUPPORT

In virtual worlds control of camera (user's) movement can be roughly classified into two categories: *navigation* and *exploration* [1]. Classification simply depends on the task or purpose of navigation what is connected with the precision of a goal object definition. While exploring or navigating it is necessary to take into consideration an intention of the user. If the user explores, the decision of movement direction is made on the way and depends on visual context and temporal situation. While navigating the aim usually strictly determines the user's movement direction because the goal is already known and only choosing a proper path to the object could be a problem.

In both cases: navigating and exploring, the user wants to move quite smoothly to the goal and omit obstacles. However, it can happen that user alters his direction because of changing his mind before the goal is reached. User should be attracted by visually chosen goal and while pursuing towards it just the aimed object should influence camera for the most part. However after changing the goal influence should adjust to an actual context.

3.   RELATED WORK

First attitude towards using potential fields in navigation support appeared in robotics. In initially published papers in robotics [2-3] forces, coming out of potential fields, were used generally for obstacle avoidance or determining one predefined and pre-calculated aim. It resulted in planning predefined and optimal path from origin to the goal.

Quite a new method of navigation support in virtual environment was presented in Xiao's [4] paper. As previous works focused mainly on fixing a save path to the aim and controlling the point of view after collision with an object, Xiao does not force any certain route but deals with camera

obstacle avoidance. Even though Xiao analysed the problem quite thoroughly and proposed many context dependent camera behaviours, he faced some problems regarding collisions with objects and oscillation of a point of view in places where the fields counterbalance.

Continuation of the efforts begun by Xiao [4] is presented in Li [5-6] papers. The author's innovations concern mainly: replacing radial field propagation with wave field propagation determined by shape of the object, different way of collision problem solving, incrementing number of points considered while calculating total influence in order to deteriorate oscillation effect, completing wider context of the user (speed, orientation, history of motion) to select objects which influence should be taken into consideration. Li could not also get rid of collisions calculating problem but he proposed successful method of diminishing field influence while coming up to the object. As a final proposal Li suggested combining the methods based on potential fields and path planning.

Very interesting and quite general is the method published in Beckhaus' dissertation [1]. She concentrated on scene voxelisation, and by adding certain weights (gradients) to the voxels author characterizes total impact on a point of view. Even though the proposed method clear vantage is its universality, it seems to be very time consuming and as contends the author his method is not suitable for compound objects due to problems with voxelisation.

4.   CAMERA CONTROL

Camera is defined as a set of three independent vectors unambiguously defining its characteristics (*pos* – position, *dir* – horizontal orientation, *up* – vertical orientation; fig.1). Camera movement is in general a three dimensional phenomenon, however taking into consideration the shape of the potential function *U*, it is acceptable to divide camera managing into two stages: moving the camera in plane *XZ* and changing the camera altitude *Y*. Camera horizontal position is determined generally by visible goals and aims, but camera altitude depends on landscape and individual intensions of the user. The U function is constructed basing on objects within a scene that is why forces coming out of the U function would affect just a horizontal position of the camera [7]. From above reasons it is assumed that camera moves through the flat area and the height of the camera must be treated separately (must be steered by a different potential field) and it seems to be a good task for another article.

In the proposed method camera checks the value of the field depending on its position and then follows on under the influence of an actual force (force is retrieved from the closest left-bottom corner node of the mesh).

Traditionally, camera can be controlled by both a keyboard and a mouse in desktop interfaces or cyber-gloves and voice commands in immersive interfaces. Due to desktop interface popularity just first interface is considered. In desktop interface lack of degrees of freedom and problems with orientation in virtual environment causes certain simplification. Pressing UP or DOWN arrow on the keyboard generates vector K which is consistent with horizontal direction of the camera and its value rises up to fixed maximum value with passage of time (fig.1b). Such dependence on time causes effect of speeding up when the user launches the movement concurrent with a view direction. Pressing LEFT or RIGHT arrow causes rotation of the camera by angle α, what results in modifying movement direction (fig.1a). New *pos'* vector is calculated according to equation 1.
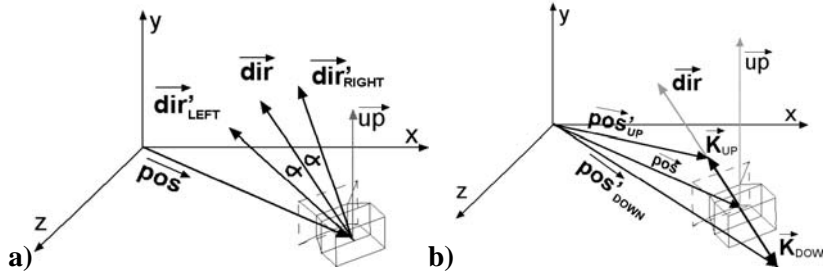

Figure 1. a) Camera horizontal orientation (dir) modification after pressing LEFT or RIGHT keyboard arrow; b) Camera position (pos) modification after pressing UP or DOWN arrow

$$\vec{K} = \vec{dir} * \alpha_1 \qquad \vec{pos}' = \vec{pos} \oplus \vec{dir} * \alpha_1 \qquad (1)$$

where $\oplus$ is an operator of a vector sum, and α_1 is a time dependant scalar coefficient which increases its value with a passage of time up to certain maximum value. This maximum value is a maximum length of vector *K* and determines maximum speed of position changeability.

Assigning camera movement direction to a keyboard and letting the user look around by means of mouse modification can cause loss of orientation especially in desktop environment.

On such defined input devices steering, it is necessary to apply potential field influence which is represented as vector *F* [7]. Value of *F* is known for each node $q_{i,j}$ of the mesh. Distance from the position of the camera to the nearest left-bottom corner node of the mesh determines which vector $F(q_{i,j})$ would be assigned to the camera. However, as mentioned above, for position

calculating only two dimensional projection of $F(q_{i,j})$ is taken into account. $F_y(q_{i,j})$ is left for altitude consideration.

To diminish the effect of rapid changeability of forces, 16 neighbouring values of $F(q_{i,j})$ are added as in the equation (2)

$$\vec{F}_{med}(\vec{pos}) = \sum_{r=i-1,\,s=j-1}^{r=i+2,\,s=j+2} \{\vec{F}_{XZ}(q_{r,s}) * \exp[-distance(pos;\, q_{r,s})]\} \quad (2)$$

$$\vec{pos'} = \vec{pos} \oplus \vec{dir} * \alpha_1 \oplus \vec{F}_{med}(\vec{pos}) * \alpha_2 \quad (3)$$

$$\vec{pos'} = \vec{pos} \oplus \vec{dir} * \alpha_1 \oplus \vec{F}_{med}(\vec{pos}) * \alpha_2 * \cos \sphericalangle (\vec{dir}, \vec{F}_{med}) \quad (4)$$

After obtaining $F_{med}(pos)$ which could be interpreted as medium value of 16 neighbouring forces' $F(q_{i,j})$ XZ projection, corresponding to an actual camera position *pos,* application of potential field influences results in a new direction of the camera movement and subsequently new position described as a total vector sum in the equation (3), where α_2 is a scalar coefficient determining the influence of a potential field on a camera position. The process described above in equations (2) and (3) is presented in figure 2.
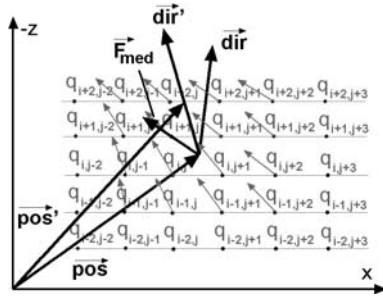

Figure 2. Process of camera direction and position modification

The method presented above could be supported with one additional serious improvement. Normally, the user moving inside the scene, can fall under influence of each force coming out of every object. Nevertheless, it seems to be reasonable to form some kind of a user context. Observer usually wants to be attracted by an object chosen visually by the direction of the camera. Even in immersive virtual environment before launching new movement user specifies new current direction and eventually can then look around. In such situations impact generated by forces coming out of objects lying outside the field of view should be considerably decreased. The proposed method achieves it by adding to equation (3) value of cosine between direction of camera *dir* and average force F_{med}. The suggested solution is described by the equation (4).

5.   SUMMARY

The method described above takes advantage of potential fields what is already known attitude. However, the solution met in literature treats potential fields strictly physically with its computing load. Contrary to former solutions, the proposed method makes use of one universal potential field which attracts user when he comes up to the object and stops him when he reaches the objects' edge. Additionally potential field is defined in such a way that it does not generate local maximums in accidental places what many authors treated as serious disadvantage of potential field usage. Such definition of potential field supports also collisions detection what, according to other authors, must be implemented additionally to repulsive and attractive potential fields. Undoubtly, one of big advantages accompanying this method is its mixture of flexibility and low calculation cost. Due to pre-calculation of all forces for each of the node, what was possible in case of a static scene, the camera context needs just 16 values taken from an array which influence camera position. It is much faster than real time calculation of influence generated by objects.

6.   REFERENCES

[1] S. Beckhaus, "*Dynamic Potential Fields for Guided Exploration in Virtual Environments*", doctoral dissertation presented to Otto-von-Gueicke-Universitat Magdeburg, September 2002

[2] Ronald C. Arkin, *Motor Scheme-based Mobile Robot Navigation,* The International Journal of Robotics Research, pages 92-l 12, August 1989.

[3] B.H. Krogh, *Integrated path planning and dynamic steering control from autonomous vehicles*, Proc. IEEE Int'l Conf. on Robotics and Automation, pages 1664-1669, April 1986.

[4] D. Xiao, R. Hubbold, *Navigation Guided by Artificial Force Fields*, Proceedings of CHI'98, Los Angeles, USA, 1998

[5] Tsen-Yen Li, Shu-Wei Hsu, *An Intelligent 3D User Interface Adapting to User Control Behaviours*, Proceedings of the 9th international conference on Intelligent user interface, 2004, Madeira, Funchal, Portugal, 184-190

[6] T.Y. Li, H.C. Chou, *Improving Navigation Efficiency with Artificial Force Fields,* Proceedings of 2001 14th IPPR Conference on Computer Vision, Graphics, and Image Processing, Taiwan, 2001

[7] A. Wojciechowski, *Generation of a static potential field for camera positioning support in a virtual environment*, unpublished