JOURNAL OF APPLIED
COMPUTER SCIENCE
Vol. 21 No. 2 (2013), pp. 165-179

Head-Coupled Perspective in Computer Games

Michał Szkudlarek, Maria Pietruszka

Lodz University of Technology
Institute of Information Technology
Wólczańska 215, 90-924 Łódź, Poland
[1]*800562@edu.p.lodz.pl*, [2] *maria.pietruszka@p.lodz.pl*

Abstract. *This work presents the new method of interaction in computer application, especially first- and third-person player games. The new control technique called "Head-Aim" is based on Head-Coupled Perspective imagery method and can be used as an independent method of aiming in games as alternative to conventional targeting (with use of popular control devices like computer mouse or gamepad) or combined with them, extending their functionality and interaction realism. Proposed pointing method can also be used in non-game application to allow people control them even if they are unable to use regular control devices. Effectiveness of Head-Aim technique was compared with convetional methods of aiming in specially designed game named "Facepace". Promising results of these tests are presented. Also a robust, fast and computationally cheap method of head tracking is proposed. Presented algorithm combines background subtrakction and new color-based tracking method enabling tracking user's head under various lightning conditions. Method allows Head-Coupled Perspective technique to be performed without latencies even with use of a low quality webcam and enables viewing scene in 3D (and use of presented Head Aim method) on every machine, without use of glasses or special devices.*

Keywords: *head tracking, Head Coupled Perspective, control, aiming, color-based tracking, Head-aim, Fish Tank VR.*

1. Introduction

Head-Coupled Perspective (HCP abbreviated) is one of the techniques presenting three-dimensional images in a two-dimensional medium. Perspective of the scene displayed on the screen is compatible with the temporary location of the observer's eyes to simulate the real, three-dimensional scene (which can be seen from different perspectives). When the user moves his head, perspective of the scene changes dynamically, adjusting to the new observation point, creating the effect of looking through a window rather than looking at the flat image.

Despite several successful approaches to create working system implementing HCP, the technique has still not become popular. It might be suprising as recent HCP implementation requires no special equipment and is not forcing user to wear any headset or glasses and as such represents a widely desired cheap glasses-free 3D. Altough researches show that HCP has greater impact on depth perception than stereoscopy [1, 2], it is almost ignored by developers and finds its place mainly in 3D CAD applications or as a way to show effects of newly invented head tracking methods. Games using Head-Coupled Perspective in some kind of interaction are completely based on its idea and represents niche genres like i.e. *object finding games*[1] or use just a plain head tracking as in [3]. In popular types of game using HCP only as 3D imagery technique is not worth computational cost of head tracking, especially high with use of not dedicated for HCP solutions like faceAPI used in [4].

Aim of this work was to develop a new control technique, based on the Head-Coupled Perspective (and enabling its usage fast, webcam-based head tracking method), which would work in popular genres of computer games such as First Person Shooters and other FPP and TPP games and expand their functionality with new methods of interaction.

2. Related Works

Although Head-Coupled Perspective is not a well known technique, it was studied for some time and there was some successful implementation.

One of the first was Fish tank virtual reality [1] which also combine the Head-Coupled Perspective with stereoscopic 3D image (with anaglyph method). It was

[1]Japanese game designed for NintendoDSi: *Hidden 3D Image: There It Is!*, `http://www.youtube.com/watch?v=dUC5JJCxx5k`

using mechanical arm connected to the head to track its position and to adjust the scene perspective respectively. This work coined the term Fish Tank VR (reffering to the achieved impression of looking through glass into an inner environment) widely used to describe similar systems. This work also first compare the impact of HCP and Stereoscopic imagery on user's understanding of the 3D scene and immersiveness of those methods (further investgate in [2]).

Jun Rekimoto [5] first proposed to track head for purpose of HCP with use of video camera. Implemented system combine background subtraction (used also in present work) with template matching giving robust tracking and rather low processing cost. The system required no expensive, dedicated devices nor putting on any equipment and thus providing real ease of use of Head-Coupled Perspective.

Probably the first (and only) widely known HCP implementation was presented by Johnny Lee using the hacked Ninendo Wii Remote [6]. This system track head very fast and accurately by tracking two infrared lamps mounted on the head. The effect presented on YouTube was very impressive and show great potencial of HCP, despite the drawback of wearing lamps on the head.

Similar solution to [5] was recently proposed for mobile devices, also using for track purpose camera, but built in cell phone or tablet [2][7]. The other difference is that the head is tracked in three dimensions (like in [6]), not only in a plane parallel to the screen.

In last few years the growing popularity of Microsoft's control device, Kinect, providing easy in use whole body tracking, results in some new projects based on its technology implementing HCP like [8].

Despite many working implementations, Head Coupled Perspective did not gain popularity. There are just a few games using it and almost none use it as control method – HCP is rather used just as a 3D imagery method. Some of the HCP game use for control purpose rather just the head position from head tracking, bypassing correct, dynamically changing viewpoint[3]. None of the popular games (representing popular genres) use Head-Coupled Perspective.

On the other hand there are ongoing researches for new methods of interaction, also based on tracking (i.e. via Kinect or webcam) useable in popular game genres as an alternative to usual control.

For example Blackwater video game[4] the alternative method of aiming is proposed – we change the position of the hindsight by moving our hand. Similar

[2]i3D – Head tracking for iPad: `http://www.youtube.com/watch?v=bBQQEcfkHoE`

[3]Browser HCP game *Facekat*: `http://www.shinydemos.com/facekat/`

[4]`http://www.youtube.com/watch?v=HRKQpffQeI4`

solution was proposed in the Ghost Recon: Future Soldier[5]. Unfortunetely it is not real-life imitating aim method as normally we aim in the eye-hand line and in these games we are actually pointing outside the view (this could probably be sloved better by track the head-hand line giving similar result to our approach). The second problem is the loss of control over character's movement as we can not use the gamepad simultaneously.

The alternative kinds of control are also search for non-game application i.e. to allow quadriplegic and nonverbal people its use. For people who can move their head, various methods of pointing in application via head move were proposed i.e. [9] and [10]. They are basing the coursor moves on the screen on the head position (maped directly or as a joystick) or its orientation (nose pointing). Proposed in this work Head-Aim method can be an natural alternative for such controls.

3. Head-Aim

Proposed method of control is based on observation that the object aimed by a viewfinder can change not only when we shift the viewfinder itself, but also when we move our head around immobile viewfinder.

Normally, in shooter games we aim at targets by moving virtual hand with use of a mouse or a gamepad and thereby targeting object seen in viewfinder of held gun which moves in accordance with the controler. In FPS games the virtual camera is rotating to target hand, but the point of view is not changing (until we move our character). By contrast, in developed Head-Aim method (used independently) the viewfinder is still (as placed in the plane of the screen) and we are changing the target by moving our head. As the perspective of the scene changes, adjusting to the new observation point, diffrent objects cross line defined by the points of the viewfinder and middle of our head, and thereby are visible in viewfinder and are aimed (figure 1). By moving our head, instead of moving virtual hand, we are shifting virtual torso and head of our character (although it is not directly seen in FPP games) which are placed in front of the screenplane and "fallows" our head. Results of the tests comparing Head-Aim with conventional control methods are presented in chapter 4.

Although proposed method may serve as an individual aiming method it can be combine with conventional aiming with i.e. mouse or gamepad. In that case mouse or gamepad is used to move viewfinder (virtual hand with gun) and rotate

[5] `http://www.youtube.com/watch?v=b47_X3S8Jbs`

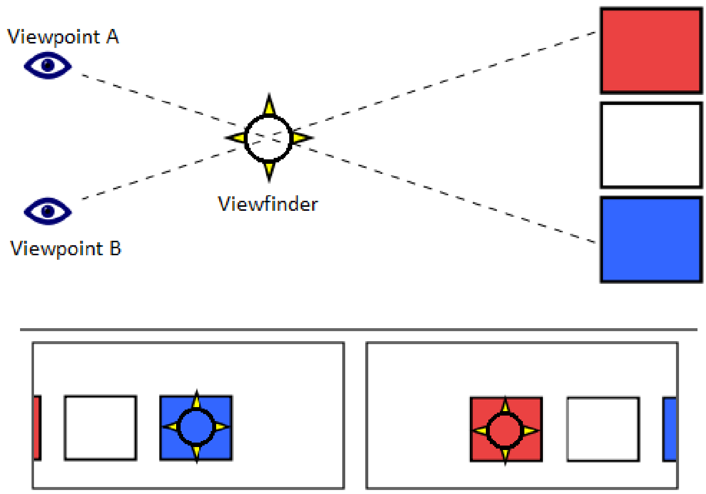

Figure 1. Head-Aim schema.

the camera (changing the camera target), while simultaneously Head-Aim (specifically our head) is moving the point of view (and virtual head of the character). This is probably the most natural way of dynamic aiming and moreover it is faster nad more precise then aiming with gamepad alone.

The Head-Aimed technique is designed primarly for use in games,but it can be use in any HCP-based interface. When we put a cursor (our non-game viewfinder) in the middle of the screen plane, user's head moves changes what it point on the (far) background of interface, where we can put its selectable elements. This method of interaction can be an alternative to other pointing techniques if the user can not use them, i.e. having disability to use the mouse, or (in case of mobile devices) having problem with the touchscreen due to worn winter gloves.

4. Head-Coupled Perspective

Head-Coupled Perspective technique requires tracking of the user's head location. In proposed solution head tracking is performed with use of web camera. It is assumed that the capturing device is situated directly above the screen (as it is on laptops). Altough proposed algorythms are designed to work even with low-quality camera, its viewing angle is important for head tracking - the narrower

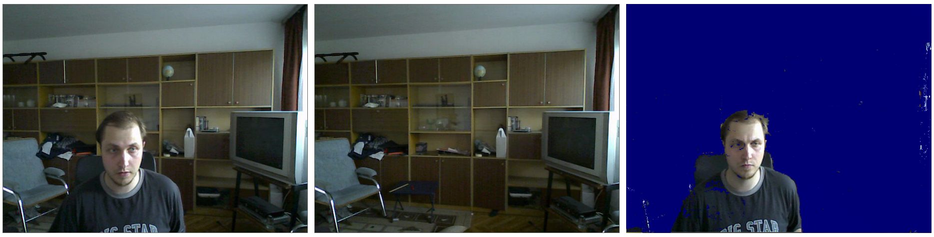

Figure 2. Captured frame (1), stored background image (2), frame with subtracted background (3).

it is and closer the user sits to the screen, the smaller became the captured area and lower is the range of head moves possible to track. On the other hand, when the user is far from the screen and viewing angle of the camera is wide (above 80°), face occupies small area of capture image and more distracting objects are captured, what may affect robustness of the algorithm. In normal use, the most desirable webcam viewing angle is around 60° (like in popular laptop cameras).

4.1. Finding face pixels

Proposed method of head tracking combines to techniques: background subtraction and color based filtering. Calculations are made for every pixels of the last frame obtained from the webcam without considering surrounding pixels and previous frames.

To perform background subtraction we need background image – frame obtained from webcam positioned in the same way as during tracking. If the difference between corresponding pixels in current frame and background image does not exceed the threshold the pixel is considered a background pixel and not included in further calculations. Same algorithm was proposed, by Jun Rekimoto [6], except we are considering distance between pixels in RGB space (in which the frame is obtained) instead YUV. Theoretically this operation discards all background pixels, leaving just those belonging to user's silhouette, in practice only still background elements are discarded (figure 2).

Second used method is color-based track technique. We can assume that the skin color differs from other elements of the scene (or at least from the user's outfit) and that in use of desktop (or mobile) applications, face is the only visible uncovered body part . The problem is to define filter in RGB space (transformation to other color space may be expensive) that reveal this difference and pass most of the head pixels while discarding majority of other pixels. Searching the color in a certain range in RGB channels around the pre-specified value covers an area of a cuboid in the RGB space. This approach is not good enough, because covering entire, variously lit skin gradient of the face, requires enlarging searched area and thereby accepting some of background pixels. Searching within an Euclidean distance changes the cuboid area into ellipsoid and gives better results but still not good enough as it is still oriented in accordance with the RGB axes of this space. It is possible to receive other shape (or orient it differently) but it requires more calculations and/or is not giving good results.

In method proposed in this work, instead of using a solid, the plane is used to split the RGB space into two parts – accepted and discarded. It allows to easily assign weights to pixels, which define the probability of their belonging to the user's face. Normal vector (r, g, b) of this plane defines the direction according to which this weights should increase. This weights should not be negative so the background pixels are not "repelling" calculated location of the head. They also should not grow unlimitedly. The proposed method calculates the weight F of each pixel with the following equation:

$$F = min(255, max(0, r \cdot R + g \cdot G + b \cdot B + d \cdot 100)) \tag{1}$$

Parameters r, g and b (with real, floatin point values) tells how much the skin is respectively red, green and blue, the d value changes directly the amount of pixels accepted by the method. Thereby user can easily set up manualy the filter to meet any condition, but even with pre-set values ($r = 3.0, g = -2, b = -0.5, d = -0.5$) gives satisfactory and usable weights in most cases (figure 3).

Although designed color-based tracking method works well by itself, the best results gives its combination with background subtraction (figure 3). Earlier use of background subtracktion removes many problematic unfiltered pixels, while reducing the computation cost.

The last thing that can be done to improve search results is discarding individual pixels, but checking whether the surrounding of each pixel also belongs to face would increase significantly computation cost of the algorythms. Cheaper and giving almost the same results method is to remember only if the previous pixel belong to the head and its weight exceeds certain threshold (what excludes also goups of pixels with small weights). The result of adding such filtering presents figure 3.

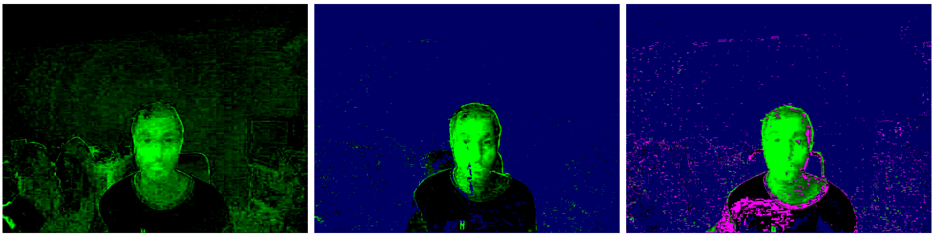

Figure 3. Frame captured in severe conditions (green pixels considered as detect face pixels): filtered with designed color-based filter (1), filtered frame after earlier background subtraction (2),with discarded individual pixels and small weight groups(3).

4.2. Estimating head location

Having weighted all of the frame pixels we can calculate head location on the captured image as their center of gravity. Such estimation is perfectly sufficient for the purpose of Head-Coupled Perspective. With pixels weights F, of an image of dimensions w and h, the head coordinates $head_x$ (horizontal) and $head_y$ (vertical) on a scale from $-\frac{1}{2}$ to $\frac{1}{2}$ can be calculate as:

$$head_x = \frac{\sum_{i=0}^{w-1} \sum_{j=0}^{h-1} F_{ij} \cdot i}{w} - \frac{1}{2} \tag{2}$$

$$head_y = \frac{\sum_{i=0}^{w-1} \sum_{j=0}^{h-1} F_{ij} \cdot j}{h} - \frac{1}{2} \tag{3}$$

Found center of the face on the captured frame gives no information about its distance from the camera (and the screen). Altough Jun Rekimoto [6] claims (and this is clearly true) that omission of the distance estimation is less noticeable than location in other dimensions and moreover it is not necessary in designed Head-Aim method. On the other hand, it affects correctness of perspective and allows user to see a wider space in HCP method.

User's head distance from the screen can be enough accurately estimate, by counting found face pixels – the closer user get to the camera, the larger area it

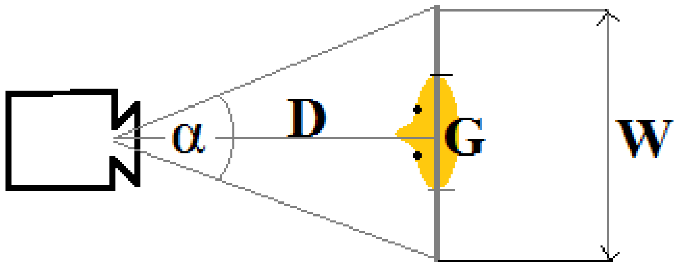

Figure 4. Schema illustrating relation between the camera and the user head.

occupies on the captured image and more face pixels should be found. Figure 4 illustrate the relation between the camera and the user head. Its distance from the screen in application units (D) is estimate by using the following equation:

$$D = \frac{\cot(\frac{1}{2}\alpha)}{2} \cdot \frac{G\sqrt{asp \cdot x}}{\sqrt{p}} \tag{4}$$

Where α is horizontal viewing angle of the camera, G is the approximate head width in game units (based on the assumed screen width), asp is the aspect of the camera, x is face area ratio to the square of its width and p is the relative area of the face (part of the frame it occupies). The $\frac{\cot(\frac{1}{2}\alpha)}{2}$ value can be precalculated and the $\sqrt{asp \cdot x}$ value can be accurately estimate as 1.

Knowing the distance of the user's head from the screen (D) we can calculate other head coordinates in game units (H_x and H_y) as fallow:

$$H_x = head_x \cdot W = head_x \cdot \frac{D}{c} \tag{5}$$

$$H_y = head_x \cdot \frac{W}{asp_s} = head_x \cdot \frac{D}{asp_s \cdot c} \tag{6}$$

Where $head_x$ and $head_y$ are previously found coordinates of the head on the frame, W is the width of area captured by camera at the distance of the head (in game units), asp_s is the aspect of the screen and the c is precalculated $\frac{\cot(\frac{1}{2}\alpha)}{2}$ value.

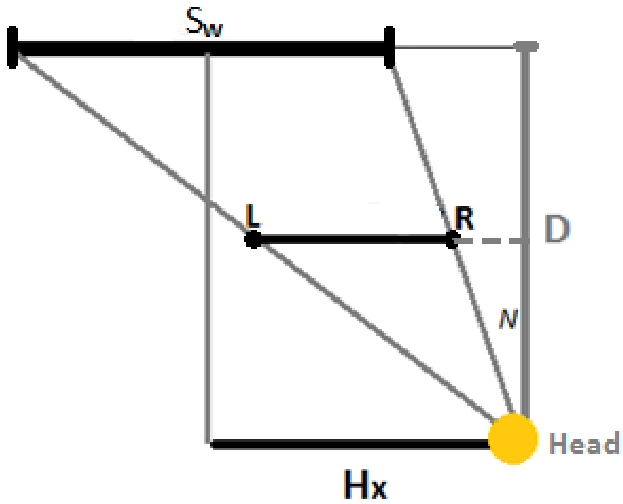

Figure 5. Schema illustrating relation between the camera and the user head.

4.3. Calculating projection

Estimated head position is used to create off axis perspective projection matrix. Unlike normal projection of the virtual scene, we are not assuming that the user's line of sight is perpendicular to the screen nor the viewing frustrum is symmetrical. To create projection matrix we need to specify $Left$, *Right*, Top and *Bottom* values (defining shifts of sides of the near plane of the viewing frustrum) so the image look correct from the estimate perspective. Figure 5 illustrate relations between the user, screen and the near plane of viewing frustrum. The following equations are used to calculate the $Left$ and *Right* values (the Top and *Bottom* are calculated analogously):

$$Left = (-H_x + (\frac{1}{2}S_w)) \cdot \frac{N}{D} \tag{7}$$

$$Right = (-H_x - (\frac{1}{2}S_w)) \cdot \frac{N}{D} \tag{8}$$

Where N is the *Nearplane* value, the distance to the near plane of the viewing frustrum (it must be set lower than D if we want objects from the game to be floating in front of the screen like in [6]).

5. Evaluation

To test robustness of head tracking method and to compare the effectiveness of Head-Aim with conventional control methods with mouse and gamepad, a simple game was designed enabling aiming with use of all these methods. The designed game, named *Facepace*, is a simple 3D Tunnel Shooter game in which player shoots incoming asteroids, flying in the cosmic crater, viewing the scene from first-person head-coupled perspective.

Twelve subjects not associated with the project were invited to test the application, 6 persons in each group. The first group was represented by experienced gamers who play a lot in games, including first-person shooters and other games viewed from a first-person perspective. All these subjects were also experienced in controlling both the computer mouse and gamepad. The second group consisted of people playing computer games occasionally and not playing in FPS games. In each group occurs age and sex differences, but the majority was 20 – 25 years old men.

In the experiment each subject played designed game, shooting down as many incoming asteroid as he can at a specified time using the Head-Aim control method. The participants were informed scoring rules, according to which for shooting down each asteroids they received 2 points, and lose one point for each unsuccessful shot or asteroid passed through. Each participant played three validated game rounds (2 minutes each) using Head-Aim, preceded by one non-measured round to learn game controls. In addition, for comparison purposes, each person played rounds with conventional control methods (mouse and gamepad), also three measured rounds after one training round. To ensure the independence of the average results, each participant tested three methods of control in a different order. All rounds were played under optimal conditions, in which there were no problems with head tracking. Taken into account was points earned by subjects and (independantly) the average shooting frequency. The results of each round of different control methods for both groups are shown on the graphs in figure 6 and 7.

After the test, participants were asked to play next five rounds of the game. This time they could use any of the given control method. Half of each group was informed that further results are no longer taken into account. All of these participants chose Head Aiming as control method. People who had not been informed that their test is finished choose equally often the mouse and Head-Aim method.

After the practical part of the experiment, participants were asked to complete a survey evaluating on the scale from 1 to 10 each control method and answer-

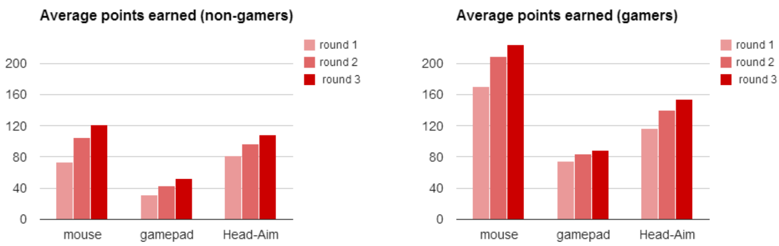

Figure 6. Average number of points earned by each group in trials.

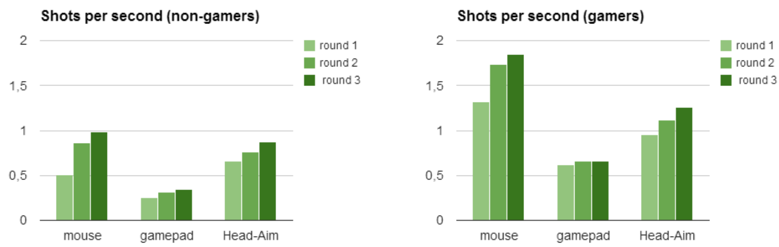

Figure 7. Average shots per second in every round.

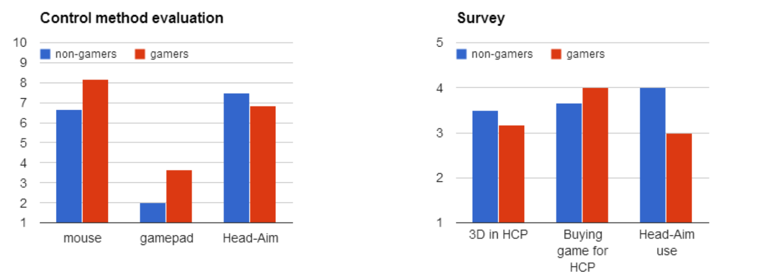

Figure 8. Average control methods valuations and answers to the survey's questions

ing three closed question. Results of the survey are presented on figure 8 and the survey was as fallow:

1. Review the different control methods on a scale of 1 to 10:
    - mouse
    - pad
    - Head-Aim
2. Rate impression of three-dimensionality of the Head-Coupled Perspective technique (very good, good, hard to say, poor, very poor)
3. Would you buy a video game for the contained Head-Coupled Perspective? (yes, probably, I don't know, probably no, no)
4. If you met Head-Aim control option in game, would you use it? (always, often, sometimes, rarely, never)

6. Result discussion and conclusion

The results of experiment shows that although aiming with mouse is the fastest and the most precise method, the Head-Aim technique is not left far behind. It does not seem obviously worse even for players who use mouse aiming in games every day. For people not accustomed so much to aiming with mouse, this methods give similar results.

Head-Aim comaparision with the gamepad aiming shows its great potential. As the figures 6 and 7 shows, although gamepad works great in a variety of games, in shooting games it apears slow and imprecise and clearly the Head-Aim seems to be better technique. However, it must be noted that participants of the test were more accustomed to the "PC's" method of control and console gamers given same task could get better results in the gamepad valuation. Still, console games developers were pushed to develop other type of controls and even use of Auto-Aim in console FPS. Tests show that in the absence of the mouse Head-Coupled Perspective and Head-Aim can be a good solution not only for inexperienced players, but also for computer gamers, not accustomed with FPS gamepad use.

Good test results of Head-Aim bode well its use on mobile devices. Having no mouse nor (usually) physical pad, mobile devices have very limited control options and adding new control method can be very beneficial on such a platform. Head-Aim method is also easier to use for mobile device users, as the perspective (and so the aimed target of the method) can be achieved not only by moving the head, but also by rotating or moving held device (and so the screen and the displayed scene).

Another adventage of Head-Aim is its naturalness of this control method and its novelty, making it more interesting than the old methods. It is difficult to measure such impressions but control method ratings chart (presented on figure 8) showing relatively much higher ratings of Head-Aim than practical tests, suggest a positive effect of these factors.

During experiment applied tracking method proved to be fast and robust even with use of poor quality cameras. Although severe tracking conditions may require filter calibration (or time consuming autocalibration), it is simple enough that all test subject have done it by themself.

The commercial potential of the presented methods reflect the results of surveys (figure 8) - respondents were interested in a new, previously unknown technique of 3D imaging and most of them were willing to buy the game for the fun of Head-Coupled Perspective and to take advantage of Head-Aim in the usual titles. Although these results do not automatically guarantee commercial success, other factors can also be taken into account, like the market demand for glasses-free and cheap 3D technology. Second is the the advertising potential of HCP. Stereoscopic 3D is practically impossible to demonstrate through the two-dimensional media, like the internet or television. Head-Coupled Perspective on the other hand not only looks great on a two-dimensional film, but also looks much more efficiently than in reality (enough to look at the videos of Johnny Lee, which cause automatically the “must have” effect).

This work shows the great potential of Head Coupled Perspective and disocover new fields for futher research. The developed control method allows its further development and use on other platoforms, both mobile and consoles.

References

[1] Ware, C., Arthur, K., and K.S, B., *Fish tank virtual reality*, In: INTERCHI’93 Conference Proceedings, 1993, pp. 37–42.

[2] Wright, E., Connolly, P. E., Sackley, M., McCollom, J., Malek, S., and Fan, K., *A comparative analysis of Fish Tank Virtual Reality to stereoscopic 3D imagery*, In: 67th Midyear Meeting Proceedings, 2012, pp. 37–45.

[3] Wang, S., Xiong, X., Xu, Y., Wang, C., Zhang, W., Dai, X., and Zhang, D., *Face-tracking as an augmented input in video games: enhancing presence, role-playing and control*, In: SIGCHI Conference on Human Factors in Computing Systems, 2006, p. 1097–1106.

[4] Sherstyuk, A. and Treskunov, A., *Head tracking for 3D games: Technology evaluation using CryENGINE2 and faceAPI*, In: Virtual Reality (VR), 2013 IEEE, 2013, pp. 67–68.

[5] Rekimoto, J., *A vision-based head tracker for fish tank virtual reality-VR without head gear*, In: Virtual Reality Annual International Symposium, 1995. Proceedings., 1995, pp. 94–100.

[6] Lee, J., *Hacking the Nintendo Wii Remote*, In: Pervasive Computing, IEEE, Vol. 7, 2008, pp. 39–45.

[7] Lopez, M., Hannuksela, J., Silven, O., and F., L., *Head-tracking virtual 3-D display for mobile devices*, In: Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference, 2012, pp. 27–34.

[8] Greuter, S. and Roberts, D., *Controlling Viewpoint from Markerless Head Tracking in an Immersive Ball Game Using a Commodity Depth Based Camera*, In: Distributed Simulation and Real Time Applications (DS-RT), 2011 IEEE/ACM 15th International Symposium, 2011, pp. 64–71.

[9] Betke, M., Gips, J., and Fleming, P., *The Camera Mouse: visual tracking of body features to provide computer access for people with severe disabilities*, In: Neural Systems and Rehabilitation Engineering, IEEE Transactions, Vol. 10, 2002, pp. 1–10.

[10] Tu, J.; Huang, T. H. T., *Face as mouse through visual face tracking*, In: Computer and Robot Vision, 2005. Proceedings. The 2nd Canadian Conference on Computer and Robot Vision, 2005, pp. 339–346.