Journal of Universal Computer Science, vol. 14, no. 19 (2008), 3127-3141
submitted: 23/7/08, accepted: 29/10/08, appeared: 1/11/08 © J.UCS

Using the Optical Flow to Implement a Relative Virtual Mouse Controlled by Head Movements

Tomàs Pallejà
(University of Lleida, Lleida, Spain
tpalleja@diei.udl.cat)

Edgar Rubión
(University of Lleida, Lleida, Spain
edgarrubion@gmail.com)

Mercè Teixidó
(University of Lleida, Lleida, Spain
merceteixido@hotmail.com)

Marcel Tresanchez
(University of Lleida, Lleida, Spain
mtresanchez@diei.udl.cat)

Alicia Fernández del Viso
(eInclusion Unit, Indra, Madrid, Spain
afernandezde@indra.es)

Carlos Rebate
(eInclusion Unit, Indra, Madrid, Spain
crebate@indra.es)

Jordi Palacin
(University of Lleida, Lleida, Spain
palacin@diei.udl.cat)

Abstract: The following paper introduces the work conducted to create a relative virtual mouse based on the interpretation of head movements and face gesture through a low cost camera and the optical flow of the images. This virtual device is designed specifically as an alternative non-contact pointer for people with mobility impairments in the upper extremities and reduced head control. The proposed virtual device was compared with a conventional mouse, a touchpad and a digital joystick. Validation results show performances close to a digital joystick but far away from a conventional mouse.

Keywords: Virtual mouse, human computer interaction, optical flow
Categories: D.2.2, D.2.3, D.2.5, H.5.2, J.5

1 Introduction

Currently the mouse is the most popular input device used in computers, providing access to computers and the capabilities of internet. In this work a new relative virtual mouse specifically designed to deal with some accessibility problems faced by people with mobility impairments in the upper extremities and reduced head control is presented. This relative virtual mouse is designed as a non-contact alternative to digital or button based joysticks [Casas, 06] used by some people with disabilities to interact with computers [Mauri, 06]. The idea of this virtual device was suggested by some therapist as a pointer alternative for people with reduced head control. The word *relative* means that the user gives direction orders to the pointer in the screen whereas acceleration and deceleration are allowed by repetitive direction orders. This way, an involuntary or uncontrolled head movement will have a small impact in the trajectory of the pointer device: accelerating (if the direction of the uncontrolled movement coincides with the actual direction of the pointer) or decelerating or stopping (otherwise). The *relative* implementation allows that, after any tic or spasm, the location of the pointer in the computer screen is completely predictable and easily compensated with other voluntary head movements.

The physical mouse has some virtual alternatives to control displacement in an absolute manner. Some of the non-contact alternatives are based on a camera pointed to the user to convert user's head movement in pointer displacement using different image vision algorithms. In [Gary, 98] the skin color distribution is used for user detection; in [Grauman, 04] the differences between two consecutive images are used for eye blink and eyebrow raise detection. Alternatively, some systems require a manual selection of the area of the face [Betke, 02][Gips, 01]; this area is used as a template for the following frames and must be located by correlation or template matching [Kim, 06]. Thus, the template offset displacement is converted in absolute mouse displacement. In [Tu, 07] the optical flow of the image and a reference template is computed and converted in mouse movements. In [Brox, 04] the optical flow is computed defining an energy function that combines brightness, gradient, and the spatio-temporal smoothness of the images outperforming the results obtained using only one of these concepts.

Alternatively, other non-contact virtual mouses [Hyun, 04][Reilly, 99] use infrared illumination for accurate detection and localization of the user. However, this technique requires a camera with a band pass filter tuned at the infrared wavelength. Other versions [Gareth, 00][Chen, 99] use infrared illumination but the source is attached to the user's head and the camera is used to detect the absolute position of the source (point with maximum emission).

There are a lot of different alternatives to the vision based systems: in [Pregenzer, 99] the encephalogram data is used for direct virtual mouse control. In [Norris, 97] the electric patterns of the muscles around the eye are used to estimate the absolute orientation of the eye. In [Kim, 02] a gyroscope is used to estimate absolute head orientation.

In this work, a new non-contact relative virtual mouse based on a simplified implementation of the optical flow [Horn, 81] of consecutive images is proposed. The virtual device is designed as an alternative to some digital joysticks used by users with mobility impairments in the upper extremities. The main objectives are fourfold:

1) detect small head movements and convert into cumulative direction and acceleration orders for the pointer, 2) detect face gestures and convert into simulated clicks, 3) use a standard USB low cost webcam as the input device, and 4) develop a virtual device with low CPU charge. The main improvements compared with other non-contact virtual mouses are the use of a low cost webcam as the input device and the detection of different face gestures to simulate the activation of the different mouse buttons.

This paper is structured as follows: section 2 presents the basic algorithm for image movement detection. Section 3 presents algorithms used in the implementation of the virtual mouse. Section 4 includes some validation results compared with other input devices. Finally section 5 outlines the conclusions.

2 Optical flow implementation

The principle of the proposed relative virtual mouse is to detect relative user head movements in one image and convert them in relative mouse displacements. The basic image processing algorithm used for image motion detection is based on the computation of the optical flow [Horn, 81][Sun, 02] from two consecutive images I^F and I^{F-1}, where F is the frame order number of an image with M columns and N rows. The procedure is as follows: first, the actual image is divided in sections of m by n pixels:

$$I_{i,j}^{F}(x,y) = I^{F}(i \cdot m + x, j \cdot n + y) \tag{1}$$

where (i,j) are the column and row section index, and (x,y) the relative position of the pixel in the section.

Next, each section of the actual image, F, is compared with the same area of a previous image, F-1, using a displacement offset (dx,dy):

$$S_{i,j}(d_x, d_y) = \sum_{x,y} \left[I_{i,j}^{F}(x,y) - I_{i,j}^{F-1}(x + d_x, y + d_y) \right]^2 \tag{2}$$

where d_x and d_y are the displacement offset applied to compare section (i,j) of both images and $S_{i,j}$ the squared similitude value obtained. As it is well known, the computation of (2) is time consuming [Brox, 04] and it could be very much optimized through gradient analysis [Chen, 98]. However, for the purpose of head motion detection where only a combination of horizontal and vertical movement must be expected, it could also be simplified to detect motion in these two main horizontal ($H_{i,j}$) and vertical ($V_{i,j}$) axis:

$$\begin{aligned} H_{i,j}(d_x) &= \sum_{x,y} \left| I_{i,j}^{F}(x,y) - I_{i,j}^{F-1}(x + d_x, y) \right| \\ V_{i,j}(d_y) &= \sum_{x,y} \left| I_{i,j}^{F}(x,y) - I_{i,j}^{F-1}(x, y + d_y) \right| \end{aligned} \tag{3}$$

Another advantage of the proposed optical flow implementation for user detection is that no additional segmentation or interpolation procedures are needed in the sections where motion is not detected. Additionally, the proposed formulation allows the creation of four independent motion matrices (*L* = left, *R* = right, *U* = up, *D* = down) with:

$$x_{\min} = x\left(\min H_{i,j}(x)\right)$$
$$\begin{cases} R_{i,j} = \left|x_{\min}\right|, & if \quad -p \le x_{\min} < 0 \\ L_{i,j} = x_{\min}, & if \quad 0 < x_{\min} \le p \end{cases} \tag{4}$$

$$x_{\min} = x\left(\min V_{i,j}(x)\right)$$
$$\begin{cases} U_{i,j} = \left|x_{\min}\right|, & if \quad -q \le x_{\min} < 0 \\ D_{i,j} = x_{\min}, & if \quad 0 < x_{\min} \le q \end{cases} \tag{5}$$

where x_{min} is the position of the minimum value in the functions H_{ij} and V_{ij}. Finally, *p* and *q* are the maximum horizontal (columns) and vertical (rows) offset applied to each image section.

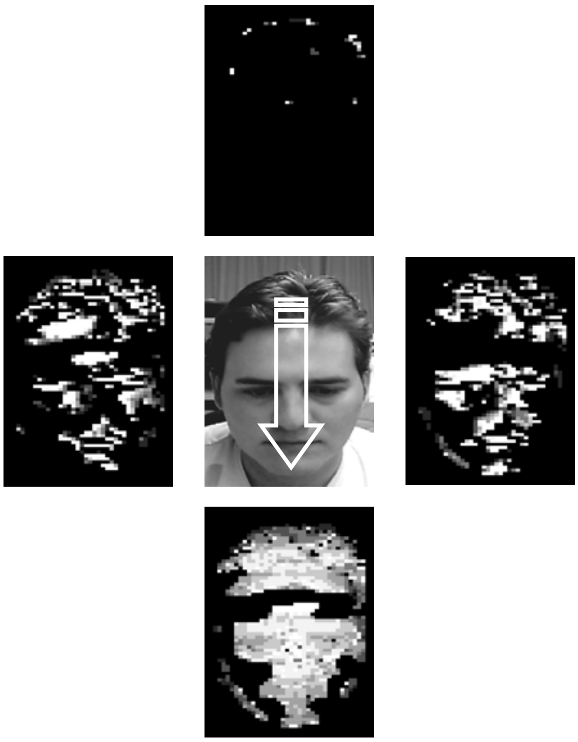

Figure 1: Motion images L, R, U and D when the user moves the head down.

Figure 1 shows an example of the motion matrices *L, R, U, D* when the user moves the head down. The number of nonzero pixels in the matrices is proportional to the area in movement while the value of the pixels denotes the intensity of the movement. Note that the motion matrices have lower resolution than the initial image because each pixel represents the motion of a complete image section without overlapping, allowing an extremely fast execution of the motion detection algorithm.

The most important parameters of this image motion detection algorithm are the size of the sections (*m* by *n*) and the limits of the displacement offsets (*p*,*q*). Figure 2 shows a representation of the influence of both parameters in the time spend by the algorithm when analyzing two images of 320x200 pixels of uncorrelated random noise using a Pentium® D at 3GHz. Figure shows that small section sizes (between 5x5 and 10x10 pixels) can be analyzed very fast, limiting the displacement offset although large amount of time is needed for larger displacement offsets. On the other hand, big section sizes require less time to analyze the images because the number of sections is lower, causing some steps in time dependence as shown in Figure 2.

The final implementation of the virtual mouse uses two values for these parameters; when the whole image of 640x480 pixels is analyzed the sections have 20x20 pixels and both offsets are limited to 10 pixels; when part of the image is analyzed (typically 150x200 pixels) the sections have 8x8 pixels and both offsets are limited to 6 pixels spending 8.3 ms in the analysis with a maximum theoretical frame rate of 120 frames per second.

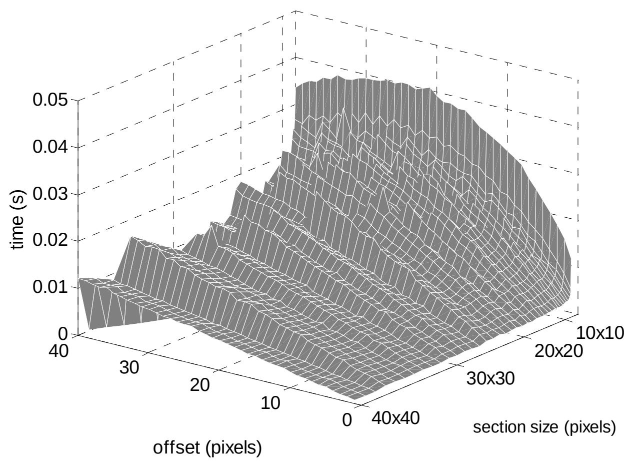

Figure 2: Time spent in optical flow computation using different sections sizes and offsets.

3 Virtual Mouse Implementation

The virtual mouse was implemented using the optical flow algorithm described in the previous section for motion detection. The final implementation has the following independent procedures: A) Initial user detection, B) Cursor movement, and C) Click detection. The final implementation is written in C for Windows® and the mouse is emulated sending typical mouse messages to the operating system. The image from the camera is obtained with a resolution of 640x480 pixels at 10 frames per second using the standard Video For Windows® library, allowing the use of any low costs USB compatible camera with video recording capabilities. Currently, the virtual mouse is fully developed and can be downloaded at http://robotica.udl.cat.

4 Initial user detection

User detection is the first procedure of the proposed virtual mouse. To this end the user must be in front of the camera, with the head centered in the images obtained. In this initial procedure, the four motion matrices of the whole image are computed and added in one overall cumulative movement image. Finally, ten frames of this cumulative image are averaged for motion noise reduction. Mouse activation is performed turning the head from left to right several times. Two turns are enough in standard illumination conditions (>10 lux) while up to ten times are needed in almost darkness, although this value is camera dependent. When the cumulative movement of the head reaches a threshold motion level the square contour of the head can be located by a simple analysis of the average cumulative image (Figure 3); this area is defined as a Region of Interest (ROI) for later face centered analysis. Compared with other methods for face detection [Viola, 04] the main advantage of this proposed procedure is that initial user detection does not depend on the skin color while its main disadvantage is that the background must be static during this initial detection. The robustness of this initial detection defines the robustness of the virtual mouse because user interaction is based on head and face movements inside this ROI.

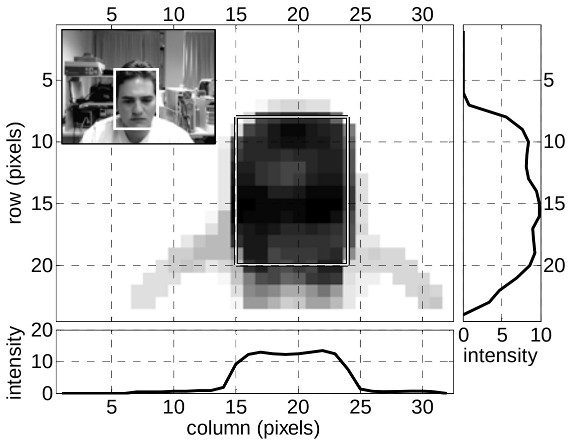

Figure 3: User detection using row and column sum applied to the cumulative displacement matrix.

This initial procedure could be avoided using a predefined ROI although then the virtual mouse will be camera-model dependent because typical low cost webcams from different manufacturers have also very different wide angle lens. Moreover, the interpretation of face gestures requires an accurate location in the image and other alternatives as the manual selection of the area of the face will make less useful and usable the virtual mouse for people with mobility impairment. As an example figure 4 shows the size of the ROI obtained in this initial user detection corresponding to the head of one volunteer user for different webcams placed over the monitor and different distances form the user to the webcam/monitor. The recommended working distance between monitor and user head is from 50 cm to 70 cm (Figure 4-dotted ellipse) and procedures for head and face analysis are optimized for ROI sizes from 90x120 to 210x279 pixels. Finally, figure 5 shows the relative error obtained when measuring the width of the head using the proposed procedure depending on the size of the sections of the optical flow analysis when the user is at a fixed distance of 55 cm. As section size increases the error also increases proportionally to the webcam angle of view (38º for Quickcam Pro, 42º for LifeCam, and 51º for Creative Live). The error is lower (and then the localization is better) for smaller section sizes but to reduce the time spent in the computation (see Figure 2) the initial user detection is performed by dividing the initial image of 640x480 pixels in sections of 20x20 pixels and by limiting the vertical and horizontal search offset to 10 pixels.

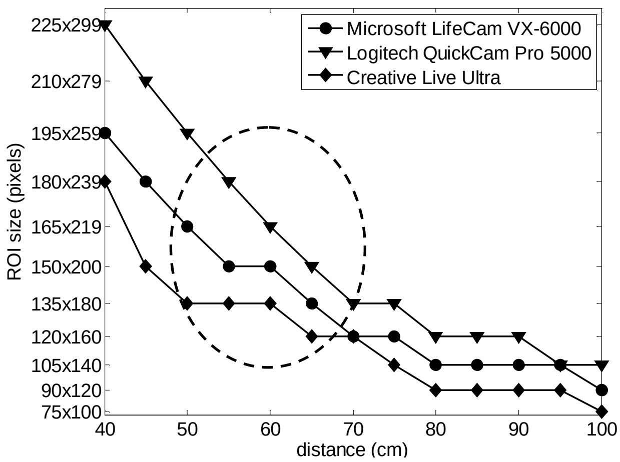

Figure 4: Detected ROI (head) size depending on the distance between user and webcam.

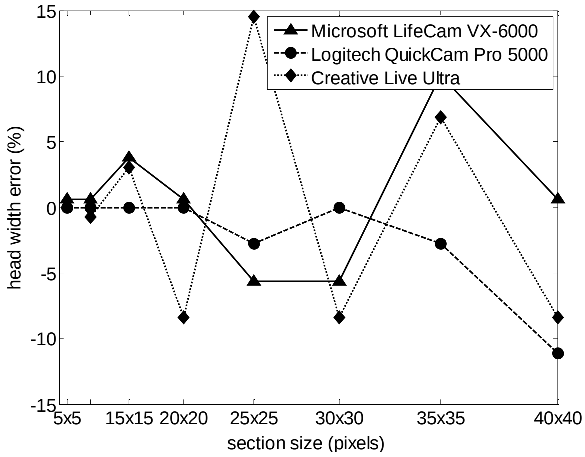

Figure 5: Relative error in automatic head width determination depending on section size used in the optical flow analysis.

5 Cursor Movement

After the initial user detection and localization the virtual mouse must convert head movements in pointer displacements. To this end, the head movement is obtained applying the optical flow procedure to the ROI selected (Figure 1) using sections of 8x8 pixels and a matching search offset limited to 6 pixels. Then, the pointer is moved in the direction of the motion matrix with a higher average value (see Figure 1). Repeating head movement in the same direction accelerates the pointer whereas a movement in the opposite direction stops pointer displacement. Combining vertical and horizontal head movements the user can move the pointer to any place in the screen. In such conditions any tic, spasm or other uncontrolled head movement has small impact in the trajectory of the pointer because just accelerates (keeping its trajectory) or stops the pointer. This is a functional improvement over a virtual mouse with absolute positioning where the pointer jumps to unexpected locations after an uncontrolled head movement.

6 Click detection

Finally, the virtual mouse must detect and emulate click actions that is a fundamental part of human computer interaction. Currently this detection is performed only when the pointer is in a stop state. Four different and configurable alternatives are considered using (Figure 6): eye blink, open mouth, eyebrow rise, and a timer. The automatic click after a predefined time without pointer movement does not require any optical flow analysis and is very useful at the beginning although very annoying while working seriously because a lot of unwanted situations are generated by unexpected clicks. The other methods use the results of the optical flow of the selected ROI to convert face gestures in click actions in the following manner:

Figure 6: Facial gestures considered: forced eye blink (left), open mouth (center) and eyebrow raise (right).

6.1 Eye Blink Detection

Forced eye blink is detected using the motion matrices obtained with the optical flow. Figure 7 shows the *D* motion matrix when the user closes the eye (Figure 6-left) making an unnatural blink; natural eye blinks are extremely fast and are unappreciable in the image obtained by typical webcams (with very low dynamic range). Then the typical pattern originated by this unnatural blink can be easily detected and converted into a click operation by simple analysis of the cumulative addition of rows and columns (see Figure 7): the cumulative sum of columns has two peaks while the cumulative sum of the rows has one peak in the middle of the ROI.

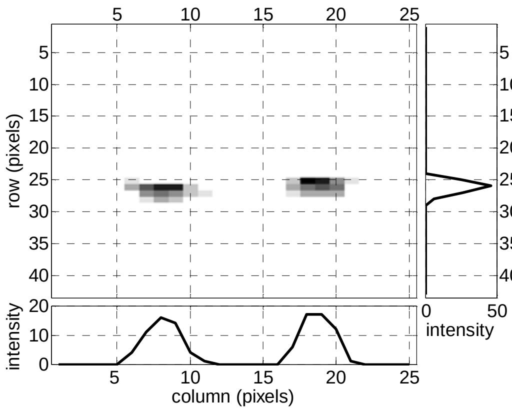

Figure 7: Eye blink detection using row and column sum on D motion matrix.

6.2 Open Mouth Detection

Open Mouth is detected using the motion matrices obtained with the optical flow. Figure 8 shows the *D* motion matrix when the user opens the mouth (Figure 6-center). In this case the motion is located in the lower part of the matrix. Then the typical pattern originated by this gesture can be easily detected and converted into a click operation by simple analysis of the cumulative addition of rows and columns (see Figure 8): the cumulative sum of columns has one peak in the center while the cumulative sum of the rows has one peak in the lower part of the ROI.

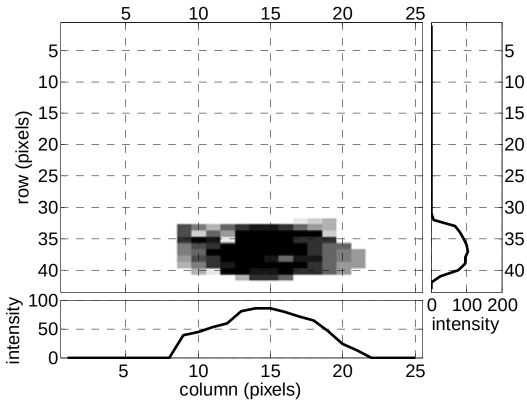

Figure 8: Open mouth detection using row and column sum on D motion matrix.

6.3 Eyebrow Raise Detection

Finally, eyebrow raise is detected using the motion matrices obtained with the optical flow. Figure 9 shows the *U* motion matrix when the user raises the eyebrow (Figure 6-right). Nevertheless, in this case, the color of the eyebrow must differ from the color of the face to be detected by the optical flow analysis. Figure 9 shows the typical pattern originated by this gesture: the cumulative sum of columns has two peaks while the cumulative sum of the rows has one peak in the center of the ROI. In this case the pattern can be easily confused with a forced eye blink because no precise location of the eyes is available.

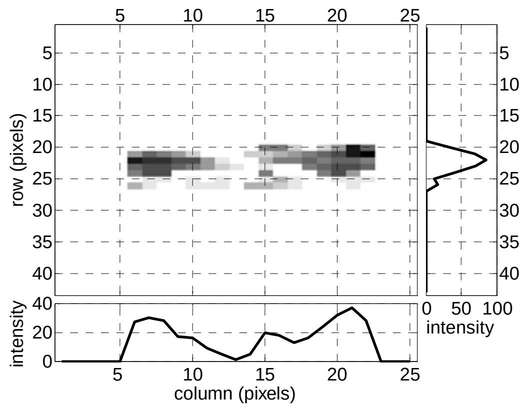

Figure 9: Eyebrow detection using row and column sum on U motion matrix.

7 Virtual Mouse Validation

Validation was planned just to compare the performances of several input devices: the proposed relative virtual mouse, the conventional mouse, a digital joystick, and a standard touchpad. Validation was performed with nine university volunteer users without mobility impairments and very familiar with computers; specific usability test with end-users will be performed in the near future. Results obtained for all users were very similar. Figures 10 to 13 shows the results obtained with one user to improve visual interpretation.

The validation experiment was performed over a computer screen of 1280x800 pixels with all pointer enhancements offered by the operating system disabled. During the experiment one target appears in a random position of the screen waiting for the user click, then a new target appears in another random position of the screen at a fixed radius distance from the previous target. The trajectory and clicks were registered using an additional software utility.

Figure 10 shows the absolute position of five targets and the trajectory followed by the all input devices selected. The trajectory of the conventional mouse is the most linear for all targets while the trajectory of the touchpad is the most erratic and requires more corrections during the movement.

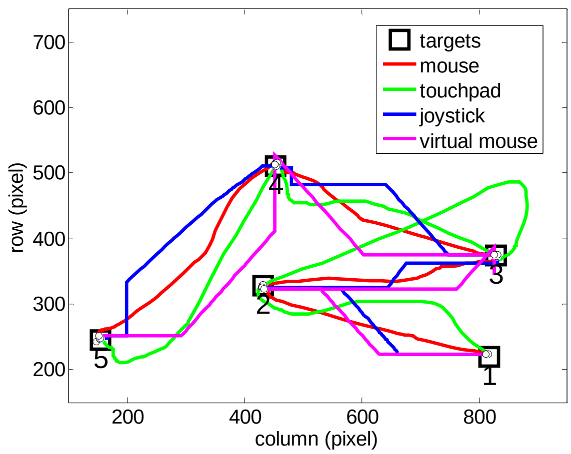

Figure 10: Position of five random targets (radius of 400 pixels) and input devices trajectory.

Figure 11 shows the relative error obtained when comparing the real distance between targets and the trajectory of the input devices. As expected, the maximum error (and the worst trajectory) corresponds to the touchpad and the minimum error corresponds to the conventional mouse. The error originated by the proposed virtual mouse and the joystick have intermediate values. Figure 12 shows the average speed between consecutive targets. The speed of the mouse and touchpad are very large whereas the speed of the joystick and the virtual mouse is very similar although highly depended on the device configuration as initial speed of the pointer. Finally, figure 13 shows one example of relative trajectory between two targets. The virtual mouse is the slower device because untrained users do not accelerate the pointer with additional head movements while the joystick continuously accelerates just keeping the stick pressed.

One important user report during the validation experiments performed with the virtual mouse is that the stress in the muscles of the face is lower when using the mouth for the clicks; this is because moving the mouth is more natural than closing the eye or raising the eyebrow. Finally, testing the virtual mouse in different computers equipped with Pentium® 4, Pentium® D, and Core™ 2 Quad the CPU charge was from 1% to 10% performing normal operations as text typing and web surfing.

Validation results confirm the utility of the optical flow algorithm for head motion and face gesture detection. Future work will be centered in the improvement of the control of the pointer through the reduction of the number of head turns needed to accelerate the pointer in large displacements. The detection of different facial gestures allows the configuration of the left-click, right-click, and double-click mouse functions but future work will be also focused on the analysis of the pointer trajectory for automatic click generation as a way to reduce the number of facial gestures performed by the user. After these improvements, the relative virtual mouse will be ready for a large usability test with a statistical representative sample of end-users.

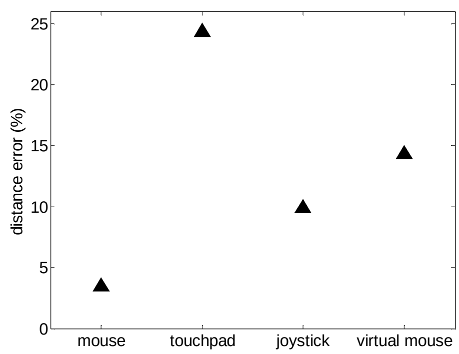

Figure 11: Average relative distance error between targets

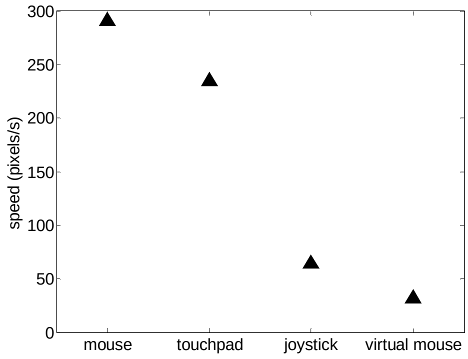

Figure 12: Average speed between targets.

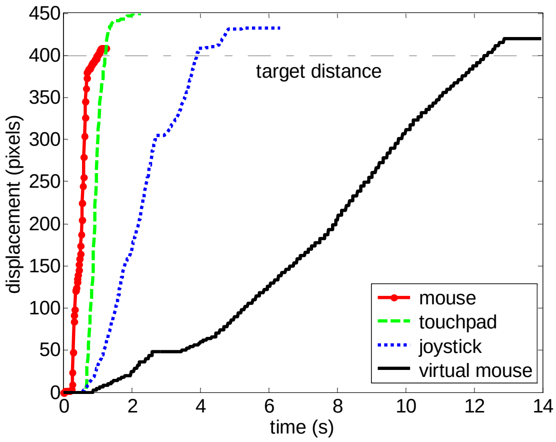

Figure 13: Relative trajectory followed by the input devices between two targets.

8 Conclusions

In this work, a new implementation of a non-contact relative virtual mouse is proposed as an alternative input device in the case of people with mobility impairments in the upper extremities and reduced head control. The virtual mouse is based on the interpretation of head movements and face gesture through a low cost camera and the optical flow of the images. Validation results obtained with people without impairments show performances close to a digital joystick but far away from a conventional mouse. Future work will be centered in two aspects: the reduction of the number of head turns needed to accelerate the pointer, and the supervision of the trajectory of the pointer for automatic click generation as a way to reduce the number of facial gestures performed by the user.

Acknowledgements

This work has been funded by Indra and Fundación Adecco under the Accessible Technologies Program. This program aims to foster innovation to make technology accessible for people with disabilities.

References

[Betke, 02] M. Betke, J. Gips and P. Fleming.: The Camera Mouse: Visual Tracking of Body Features to Provide Computer Access for People With Severe Disabilities, IEEE Trans. On Neuronal Systems and Rehabilitation Engineering, vol 10, no. 1, pp. 1–10, March 2002.

[Brox, 04] T. Brox, A. Bruhn, N. Papenberg, J. Weickert.: High Accuracy Optical Flow Estimation Based on a Theory for Warping, LNCS, vol. 3024/2004, pp. 25–36, May. 2004

[Casas, 06] R. Casas, M. Quilez, B. Romero and O. Casas.: NIBLUM: Non-Invasive Bluetooth Mouse for Wheelchair Users, Computers Helping People with Special Needs, vol 4061/2006, pp. 1006–1013, Jul. 2006

[Chen, 99] Y. Chen, F. Tang, W.Chang, M. Wong, Y. Shih, T. Kuo, The new design of an infrared-controlled human-computer interface for the disabled, IEEE Trans. on Rehabilitation Engineering, vol. 7, no. 4, pp. 474–481, Dec. 1999.

[Chen, 98] T. Chen, A cost-effective three-step hierarchical search block-matching chip for motion estimation, IEEE Journal of Solid-State Circuits, vol. 33, no. 8, pp. 1253–1258, Aug. 1998.

[Gareth, 00] D. Gareth, R. Drew and P. Blenkhorn.: Controlling mouse pointer position using an infrared head-operated joystick, IEEE Trans. on Rehabilitation Engineering, vol. 8, no. 1, pp. 107–117, March 2000.

[Gary, 98] Gary R. Bradski.: Computer Vision Face Tracking For Use in a Perceptual User Interface, Intel Technology Journal Q2, 1998, Available: http://citeseer.ist.psu.edu/bradski98computer.html

[Gips, 01] J. Gips, M. Betke and P. DiMattia.: Early Experiences Using Visual Tracking for Computer Access by People with Profound Physical Disabilities, Universal Access in HCI: Towards an Information Society for All, Ed. C. Stephanidis, vol. 3, pp. 914–918, Aug. 2001.

[Grauman, 04] K. Grauman, M. Betke, J. Lombardi, J. Gips and G.R. Bradski.: Communication via eye blinks and eyebrow raises: video-based human-computer interfaces, Universal Access in the Information Society, vol. 2, pp. 359–373, Feb. 2004.

[Horn, 81] B. Horn, B. Schunck.: Determining optical flow, Artificial Intelligence, nº17, pp, 185–203, 1981.

[Hyun, 04] D. Hyun and M. Jin.: Eye-mouse under large head movement for human-computer interface, Proc. Of the Int. Conf. on Robotics and Automation, vol. 1, pp. 237–242, May. 2004.

[Kim, 06] H. Kim and D. Ryu.: Computer Control by Tracking Head Movements for the Disabled, Computers Helping People with Special Needs, vol. 4061/2006, pp. 705–715, Jul. 2006

[Kim, 02] Y. Kim and J. Cho, A novel development of head-set type computer mouse using gyro sensors for the handicapped, Micro technologies in Medicine & Biology 2nd Annual International IEEE, pp 356–360, 2–4 May 2002.

[Mauri, 06] C. Mauri, T. Granollers, J. Lorés, M. García.: Computer vision interaction for people with severe movement restrictions, Human Technology, Vol. 2(1), pp. 38–54, Apr. 2006.

[Norris, 97] G. Norris and E. Wilson.: The Eye Mouse, an eye communication device, Bioengineering Conference, Proceedings of the IEEE 23rd Northeast, pp. 66–67, 21–22 May. 1997.

[Pregenzer, 99] M. Pregenzer and G. Pfurtscheller.: Frequency component selection for an EEG-based brain to computer interface, IEEE Trans. on Rehabilitation Engineering, vol. 7, no. 4, pp. 413–419, Dec. 1999.

[Reilly, 99] R. Reilly and M. O'Malley.: Adaptive Noncontact Gesture-Based System for Augmentative Communication, IEEE Trans. on Rehabilitation Engineering, vol. 7, no. 2, pp. 174–182, June 1999.

[Sun, 02] C. Sun, Fast optical flow using 3D shortest path techniques, Image and Vision Computing, vol. 20, no. 13–14, pp. 981–99, Dec. 2002.

[Tu, 07] J. Tu, H. Tao, T. Huang.: Face as mouse through visual face tracking, Computer Vision and Image Understanding, vol. 108, no. 1– 2, pp. 35–40, Nov. 2007.

[Viola, 04] P. Viola, M. J. Jones.: Robust Real-Time Face Detection, International Journal of Computer Vision, vol. 57(2), pp. 137–154, May. 2004