This full text paper was peer-reviewed at the direction of IEEE Instrumentation and Measurement Society prior to the acceptance and publication.

Evaluation of Control Modes for Head Motion-based Control with Motion Sensors

Nina Rudigkeit and Marion Gebhard
Sensors and Actuators in Medicine (SAM)
Westphalian University of Applied Sciences
45877 Gelsenkirchen, Germany
Email: {nina.rudigkeit, marion.gebhard}@w-hs.de

Axel Gräser
Institute of Automation (IAT)
University of Bremen
28359 Bremen, Germany
Email: ag@iat.uni-bremen.de

***Abstract*—Head-controlled Human-Machine Interfaces play an important role in restoration of the autonomy of severely disabled people, like tetraplegics. In the literature, different control modes to map head movement onto a single degree of freedom of an object to be controlled have been presented. However, the control modes have not been compared with each other under different conditions. With the work presented in this paper, we close this gap by evaluating two of the most promising control modes, namely position mode and velocity mode. These modes were tested under different conditions in order to highlight their advantages and disadvantages and to make suggestions which mode should be used for a particular application.**

Head movements were measured using a smart 9-axis motion sensor system. The mouse cursor control was considered as an example application for which control modes were compared subjectively as well as objectively. The objective comparison was carried out using a two-dimensional Fitts' Law Test at two different distances to the computer screen. Then, the modes were evaluated subjectively with an evaluation sheet. Position control mode turned out to be significantly faster than the velocity mode. On the other hand, the error rate for position mode increased significantly with the screen distance while velocity mode was insensitive to screen displacements. This was in line with the significantly higher ratings of accurate pointing for velocity mode. Because of this, the velocity mode is the preferred choice for safety-critical applications, such as robot control, while position mode should be used when speed is more important than accuracy, e.g. for cursor control. Therefore, the usefulness of the control modes depends on the application.

I. Introduction

Diseases such as cerebal palsy, amyothrophic lateral sclerosis, multiple sclerosis and traumatic injuries can lead to the total loss of motor functions of the arms and legs. This disease pattern is called tetraplegia.

Tetraplegic patients require extensive home care services. Also, they have to retire from working life when they do not meet needed physical conditions any more. The psychological consequences of this range from reduced self-esteem to serious depression. Assistive devices, such as the robot FRIEND [1], can support restoration of the autonomy of people with disabilities and can therefore substantially improve their quality of life.

Assistive devices for tetraplegic patients make use of the fact that most of these patients can still move their heads to produce input signals for Human-Machine Interfaces (HMIs).

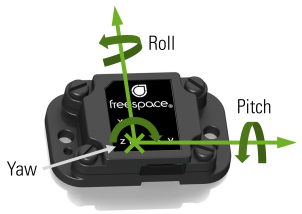

Fig. 1. FSM-9 sensor module of Hillcrest Labs [4] and corresponding coordinate system.

Interfaces that use head movements to produce control signals are referred to as head-joysticks. HMIs based on MEMS[1] motion sensors mounted on the user's head play an important role for assistive devices as the sensors are small, cheap, energy-efficient and self-contained [2]. However, motion sensors using MEMS technology are quite new, and so far they have not been enough exploited for HMI.

A. History of MEMS Motion Sensors for Head-Operated Interfaces

The first type of motion sensor using MEMS technology was the accelerometer, which appeared in 1979 [3]. The Earth's gravity provides a fixed reference for the accelerometers. As a result, accelerometers can be used as tilt-sensors after applying a low-pass filter in order to attenuate motion artifacts and noise. This means, the accelerometers can measure pitch and roll motion (Fig. 1) but cannot measure yaw motion. Head-operated interfaces based on accelerometers therefore provide only information about these two degrees of freedom (DOFs). Furthermore, they are mainly used to generate discrete control commands as quick movements induce additional noise which can hardly be removed.

MEMS gyroscopes, which were developed after the accelerometers in 1993 [5], measure rotation utilizing the Coriolis Effect. Sensor orientation can be obtained by integrating gyroscope data. However, due to the integration, small errors in sensor raw data accumulate over time, an effect known as drift. This means that gyroscopes alone are not suitable for precise motion measurement. In order to compensate for drift in pitch and roll directions, accelerometer data can be used. As a consequence, 6-axis Inertial Measurement Units

[1]Micro-Electro-Mechanical System

978-1-4799-6477-2/15/$31.00 ©2015 IEEE

(IMUs) [6] which integrate both a triaxial accelerometer and gyroscope have been developed. The interfaces using 6-axis IMUs are able to provide drift-free proportional control signals in pitch and roll directions, but they still suffer from drift in yaw direction.

MEMS magnetometers measure magnetic fields, and so they are able to provide a fixed reference due to the Earth's magnetic field. In order to compensate for gyroscope drift in yaw direction, 9-axis IMUs, in addition to a triaxial accelerometer and gyroscope, contain a triaxial MEMS magnetometer as well [7]. This means that 9-axis IMUs provide all the information necessary to measure drift-free 3D orientation. However, classical IMUs output only calibrated raw sensor data so that the user has to do the sophisticated sensor fusion himself. This is a major drawback with regard to the development of new HMIs.

Recently, smart IMUs with on-board signal processing for real-time sensor fusion became available [4]. These systems provide calibrated raw sensor data as well as almost drift-free sensor orientation.

B. Related Work

In 2007, Bureau et al [8] presented a universal interface which combined a triaxial accelerometer with a single axis gyroscope. The gyroscope was oriented to measure yaw rotation. The pitch and yaw DOFs were used to generate control signals for several applications. Depending on the direction of head movement, one out of four different control commands was classified. The operating speed could be varied by changing the amplitude of head movement. Basically, this control strategy is the discrete version of the velocity mode as presented in this paper in section II-B.

In the same year, Mandel et al [9] used a commercially available smart 9-axis IMU to control a wheelchair with head motion. Translational velocity of the wheelchair was controlled proportionally using pitch and roll motion of the head. Using of roll movement instead of yaw for control is unusual with respect to most common use of IMUs which is probably due to the fact that the integrated sensor fusion was not able to remove the drift in yaw direction.

The ENLANZA interface [10], which contained a 9-axis IMU, enabled children with cerebral palsy to navigate a mouse cursor directly with pitch and yaw motion. This control strategy corresponds to the position mode as described in this paper in section II-A.

Beside the above mentioned, some more control modes have been used in related work. However, most of these modes were only presented without comparison to other control modes, or they were compared for specific environmental conditions only. In this paper, we close this gap by evaluating two of the most promising control modes, i.e., position and velocity mode, under different conditions. The evaluation has been done both objectively and subjectively. The idea behind is to highlight the advantages and disadvantages of these control modes and to make suggestions which mode should be used for a particular application.

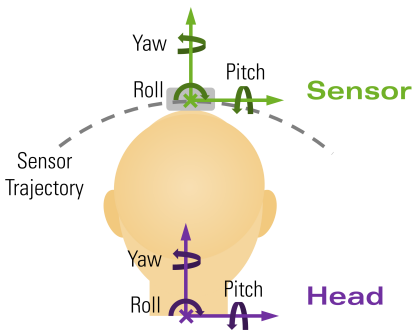

Fig. 2. Coordinate systems of the head (purple, seen from the back of the head) and of the sensor (green). Both head and sensor coordinate system have three rotational DOFs, i.e., roll φ, pitch ϑ and yaw ψ.

C. Sensor Placement and Signal Preprocessing

Within the work presented in this paper, a smart 9-axis IMU is used for precise head motion measurement. In this case, a coordinate transformation between sensor and head coordinate systems is necessary. However, in [11] we showed that the sensor was moved on a spherical surface when the sensor yaw-axis coincided with the approximated yaw-axis of the user's cervical spine (Fig. 2). Given this sensor placement, changes in head and sensor orientation are identical and a transformation of sensor orientation to head orientation is not needed. However, an offset calibration remains necessary. The head angles $(\varphi, \vartheta, \psi)_h$ are obtained by removing the offset $(\varphi, \vartheta, \psi)_0$ from the sensor angles $(\varphi, \vartheta, \psi)_s$ as follows:

$$\varphi_h = \begin{cases} \Delta\varphi + 2\pi & \text{if } \Delta\varphi < \pi \\ \Delta\varphi - 2\pi & \text{if } \Delta\varphi > -\pi \\ \Delta\varphi & \text{else} \end{cases}, \quad \text{with } \Delta\varphi = \varphi_s - \varphi_0 \quad (1)$$

$$\vartheta_h = \begin{cases} \Delta\vartheta + \pi & \text{if } \Delta\vartheta < \frac{\pi}{2} \\ \Delta\vartheta - \pi & \text{if } \Delta\vartheta > -\frac{\pi}{2} \\ \Delta\vartheta & \text{else} \end{cases}, \quad \text{with } \Delta\vartheta = \vartheta_s - \vartheta_0 \quad (2)$$

$$\psi_h = \begin{cases} \Delta\psi + 2\pi & \text{if } \Delta\psi < \pi \\ \Delta\psi - 2\pi & \text{if } \Delta\psi > -\pi \\ \Delta\psi & \text{else} \end{cases}, \quad \text{with } \Delta\psi = \psi_s - \psi_0 \quad (3)$$

Additionally, every head motion apart from the rotation around the yaw-axis results in additional linear sensor movement as the sensor is not rotated around its own center.

II. Control Modes

In order to control an object using the head movements, one needs to map the three DOFs of the head, namely roll φ_h, pitch ϑ_h and yaw ψ_h, to the DOFs of the object to be controlled. Such mapping for the purpose of controlling a robot arm was introduced in [12]. Here, the focus is on which physical quantity of the head motion should be mapped onto which quantity of the DOF to be controlled. The physical quantities of the head motion considered are: orientation, angular velocity and angular acceleration. In the following, two common intuitive mappings are presented.

A. Position Mode

In position mode, the user's head orientation is directly mapped onto the position or orientation of the object to be controlled. As a result, the limited range of motion of the head $\Delta\alpha$ leads to a limited range of motion Δs of the object. Therefore, the head motions have to be scaled to object motion in a way that all necessary control tasks can be carried out. This scaling directly influences sensitivity S, which is defined as:

$$S = \frac{\Delta s}{\Delta \alpha} \tag{4}$$

Without loss of generality, the following considerations are made for two-dimensional control tasks in which an object is moved within a plane which is perpendicular to the user's line of sight. Navigating a mouse cursor on a computer screen with the head motion is such an application.

We propose to map the ϑ_h- and ψ_h-angle of the user's head to his field of view (FOV). The user's FOV is assumed to be a two-dimensional projection of a 3D space with the shape of a frustrum of a pyramid. That means, the sensitivity S (Eq. 4) increases with increasing distance of the object from the user. As a result, the same amount of head motion leads to different amounts of object motion depending on its distance from the user. The 2D object coordinates in the FOV projected plane are denoted by (x, y). In case of cursor control, the FOV corresponds to the screen. That means, the sensitivity of cursor positioning is determined by screen size and the screen distance to the user. A linear relationship is used between the head movements $(\psi, \vartheta)_h$ and object position (x, y):

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} m_x & 0 \\ 0 & m_y \end{pmatrix} \cdot \begin{pmatrix} \psi \\ \vartheta \end{pmatrix}_h + \begin{pmatrix} b_x \\ b_y \end{pmatrix} \tag{5}$$

For the calibration, the user has to define the FOV with his head in order to obtain the parameters (m_x, m_y) and (b_x, b_y). This can be done by determining the head angles for two or more previously defined points, e.g. edges of the FOV.

B. Velocity Mode

In the velocity mode, the user's head angle is mapped onto the object's velocity. That means, if the user holds his head in a constant position, the object moves with constant velocity. Increasing or decreasing the head angle accelerates or decelerates the object's movement. As a consequence, the object's range of motion is not restricted by the head's range of motion. Furthermore, the sensitivity S is independent of the object distance from the user.

In analogy to position mode, the following considerations are made for two-dimensional control tasks in which an object is moved within an (x, y)-plane which is perpendicular to the user's line of sight. In this case, the ϑ_h- and ψ_h-angle of the user's head are mapped to the object's velocity in x- and y-direction.

We propose to use a Gompertz function [13], as shown in Fig. 3, to map the head displacement $(\psi, \vartheta)_h$ onto cursor velocity (x, y). The advantage of this mapping is that it enables the user to perform very accurate positioning while maintaining an appropriate operating speed because the velocity slowly increases at small angles and is almost linear at medium angles. At large head angles, object velocity saturates. This mapping can be formulized mathematically as:

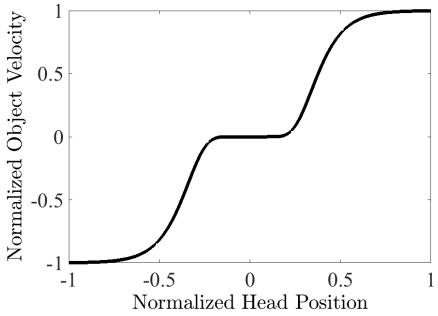

Fig. 3. Gompertz function (Eq. 6, 7) to map the head angle onto cursor velocity. Here, object velocity is normalized to maximum velocity.

$$\dot{x}(\psi_n) = \begin{cases} a \cdot e^{b \cdot e^{c \cdot \psi_n}} & \text{if } \psi_n \geq 0 \\ -a \cdot e^{b \cdot e^{-c \cdot \psi_n}} & \text{else} \end{cases} \quad \text{with } \psi_n = \frac{\psi_h}{\psi_{th}} \tag{6}$$

$$\dot{y}(\vartheta_n) = \begin{cases} -a \cdot e^{b \cdot e^{c \cdot \vartheta_n}} & \text{if } \vartheta_n \geq 0 \\ a \cdot e^{b \cdot e^{-c \cdot \vartheta_n}} & \text{else} \end{cases} \quad \text{with } \vartheta_n = \frac{\vartheta_h}{\vartheta_{th}} \tag{7}$$

The parameter a indicates the upper asymptote. In the case of cursor control, this corresponds to maximum cursor velocity in x- and y-direction, respectively. The negative coefficient b sets the displacements along the ψ_h- and ϑ_h-axis, and c indicates the growth rate. The parameters for the x- and y-DOFs are identical. The parameters ψ_{th} and ϑ_{th} define the user's range of motion along the ψ_h-axis and ϑ_h-axis, respectively. They are obtained during the calibration routine.

For every time step of length Δt, the object position is computed using the following two relationships:

$$\begin{pmatrix} x \\ y \end{pmatrix}_{new} = \begin{pmatrix} x \\ y \end{pmatrix}_{old} + \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} \tag{8}$$

$$\begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} = \begin{pmatrix} \dot{x} & 0 \\ 0 & \dot{y} \end{pmatrix} \cdot \begin{pmatrix} \Delta t \\ \Delta_t \end{pmatrix} \tag{9}$$

For the calibration of velocity mode, the ranges of motion of the user's head $(\psi, \vartheta)_{th}$ have to be recorded for the purpose of normalizing maximum head displacement to maximum object velocity a. The parameters a, b and c of the Gompertz function may be set according to the preferences of each user.

III. Methods

The goal of this study was to evaluate position and velocity mode both objectively and subjectively. Both control modes were tested for the case of mouse cursor control.

Fig. 4. (1) FSM-9 sensor module by Hillcrest Labs mounted on (2) velvet-like hair band. (3) A clip attached the USB cable to the hair band for mechanical strain relief.

A. Apparatus

The user's head movements were measured using an FSM-9 sensor module by Hillcrest Labs [4]. The FSM-9 is a 9-axis motion sensor system that provides various outputs such as real-time 3D orientation in terms of both Euler angles and quaternions, calibrated angular velocity or calibrated acceleration. In the work presented in this paper, Euler angles were used.

The sampling rate was set to $100\,\mathrm{Hz}$. That means, for every time step $\Delta t = 10\,\mathrm{ms}$, a triplet $(\varphi, \vartheta, \psi)_s$ with $\varphi_s \in [-\pi, \pi]$, $\vartheta_s \in [-\frac{\pi}{2}]$ and $\psi_s \in [-\pi, \pi]$ was obtained.

The FSM-9 was mounted on top of a hair band (Fig. 4). The tests were carried out on a 19" TFT screen with a resolution of 1280x1024 pixels.

B. Position Mode Calibration

For the calibration routine of position mode, users had to fix their gaze in a straight position and then turn their heads towards two different predefined points, $p_1 = (x, y)_1$ and $p_2 = (x, y)_2$, on the screen and save the corresponding head positions, $(\psi, \vartheta)_{h,1}$ and $(\psi, \vartheta)_{h,2}$, by clicking the mouse. Point p_1 was displayed in the upper left corner of the screen and point p_2 in the center. Point p_2 was used for offset removal. The parameters (m_x, m_y) and (b_x, b_y) were computed as described in section II-A.

C. Velocity Mode Calibration

At the beginning of the calibration in the velocity mode, users had to look at the center of the screen for offset removal. Afterwards, the users performed one repetition of each neck extension, neck flexion, neck rotation to the left and then to the right. They were instructed to move their heads as far as they could without feeling any discomfort and to save the corresponding position by clicking the mouse. In this way, two values were obtained for each DOF: one in positive direction and one in negative one. To guarantee a symmetrical mapping, only the smaller value was used to define the extent of movement. For the ψ_h-DOF, the threshold ψ_{th} was calculated as follows:

$$\psi_{th} = \begin{cases} \psi_+ & \text{if } \psi_+ \leq |\psi_-| \\ |\psi_-| & \text{else} \end{cases} \tag{10}$$

The threshold for the ϑ_h-DOF was computed analogously. After obtaining both thresholds, the calibration routine was finished.

For the experiments, the parameter a was set to $1000\,\mathrm{px/s}$. This is a relatively low value which was chosen so to enable unexperienced subjects to get used to this control scheme in an easy way. The coefficient b was set to -30 and c to -10.

D. Subjects

Ten able-bodied subjects with no known neck movement limitations participated in the experiment. Six of them were female and four were male. All participants were regular computer users. Their age ranged from 26 to 51 years. Four subjects did not have any prior experience in using the head-joysticks. Five subjects had experience in using the head-joystick of less than one hour of use. One subject had tested both position and velocity mode for several hours before.

E. Procedure

Subjects were instructed to perform a two-dimensional Fitts' Law Test according to ISO 9241-9, as explained in [14]. In this test, users have to click on circular targets of different sizes and at various distances as fast as possible but without making any mistakes. This is a standardized test to quantify the performance of non-keyboard input devices. In the work presented in this paper, the test was used to compare position and velocity mode. The distance between the targets on the screen was set to $500\,\mathrm{px}$, while the target sizes were $60\,\mathrm{px}$ and $30\,\mathrm{px}$. Users' performances were evaluated for both position and velocity mode at screen distances of $50\,\mathrm{cm}$ and $100\,\mathrm{cm}$, respectively. There were 15 trials for each of these 8 task conditions, resulting in 120 trials in total.

After the tests, all subjects were interviewed according to an evaluation sheet with the six following statements to evaluate the subjective usability of both control modes:

1) The smoothness of the operation was very high.
2) The mental effort required for operation was very low.
3) Staying at a certain position was very easy.
4) The operation speed was very fast.
5) Neck fatigue was very low.
6) Overall, control was very easy.

The subjects were able to choose between five answers ranging from 1 ("I do not agree at all") to 5 ("I totally agree"). In the end, all subjects were asked for suggestions to improve control.

Two-tailed Welch's t-tests [15] were used to compare the user ratings for position mode and velocity mode. To find out whether user performance decreased with distance, one-sided paired t-tests were performed for the data obtained at screen distances of $50\,\mathrm{cm}$ and $100\,\mathrm{cm}$. For all these tests, p-values of 0.05 were considered significant.

TABLE I. SUBJECTIVE USABILITY OF BOTH POSITION AND VELOCITY MODE

No.	Statement	Position Mode	Velocity Mode
1.	The smoothness of the operation was very high.	3.7 ± 0.9	3.6 ± 0.8
2.	The mental effort required for operation was very low.	3.4 ± 1.3	3.1 ± 1.2
3.	Staying at a certain position was very easy.	2.5 ± 1.0	3.6 ± 0.8
4.	The operation speed was very fast.	3.9 ± 1.0	2.7 ± 0.7
5.	Neck fatigue was very low.	3.4 ± 1.1	3.3 ± 1.1
6.	Overall, control was very easy.	3.6 ± 1.1	3.2 ± 0.8

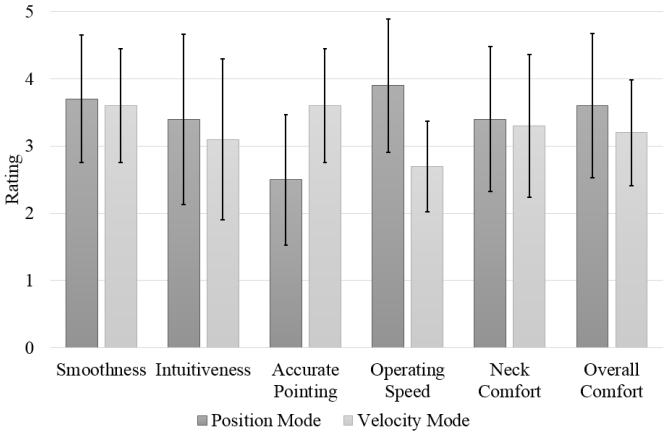

Fig. 5. Mean values and standard deviation of the subjective usability of both position and velocity mode. Accurate pointing was rated significantly better for velocity mode while operating speed was evaluated significantly higher for position mode.

IV. RESULTS AND DISCUSSION

The subjects assessed the smoothness of the operation for position mode 0.1 grades higher than for the velocity mode (Fig. 5 and Table I). One possible reason why the velocity mode was perceived less smooth than the position mode is the non-linear mapping of head position onto cursor velocity. Another possible reason is that the transfer function (Eq. 6, 7) is scaled depending on the range of motion for each DOF. This leads to different scalings as the range of motion is usually considerably greater for ψ_h- than for ϑ_h-movements. In the future, the effects of using equal scalings for both DOFs will be investigated. However, the subjective difference may also be an artifact as it is not statistically significant.

Mental effort was rated higher for the velocity mode than for the position mode. However, the difference of 0.3 grades can be considered as not significant from a statistical point of view, but it is comprehensible: Mapping the same physical quantities to each other, as done for position mode, is highly intuitive whereas using different physical quantities for cause and effect, requires higher mental effort.

The ease of dwelling on small targets was evaluated 1.1 grades higher for the velocity mode, which is statistically significant. With velocity mode, it is easy to stay at a certain position owing to the shape of the Gompertz function. The error rate stays stable when the screen distance is increased from $50\,\mathrm{cm}$ to $100\,\mathrm{cm}$ because the sensitivity does not depend on the screen distance. In position mode, every little head movement is directly translated into a cursor movement no matter where it originates from. Unintended movements, like breathing and natural tremor, make the cursor never stand still. As a consequence, it is impossible to dwell on targets below an individually determined size. This is in line with the significant increase of errors for position mode when the screen distance is changed from $50\,\mathrm{cm}$ to $100\,\mathrm{cm}$ (Fig. 6). Different filtering techniques can be applied in order to remove the jitter and make control more accurate.

The operating speed was rated significantly higher (1.2 grades) for position mode. This result is consistent with the one of the Fitts' Law Test (Fig. 7). One reason is the strong correlation between intuitiveness and operating speed. Furthermore, in the velocity mode, subjects tended to perform orthogonal movements even though diagonal movements at any angle were possible. This sequential way of control increased the movement time considerably. However, most subjects passed over to perform diagonal movements as soon as they got familiar with the way of control. It is therefore presumed that learning will have a strong effect on operating speed.

Neck fatigue was rated higher for the velocity mode than for the position mode. However, the difference of 0.1 grades can be considered as not statistically significant. From a physiological point of view the necessary movements for both control modes are comparable. The fact that the velocity mode was still experienced more tiring maybe stems from the higher mental effort which might made the users tense their neck muscles.

Overall, position mode was rated as the better control mode in this test. This result originates from the fact that accurate pointing was less important than speed and smoothness for the tested tasks. When controlling a mouse cursor, the position of the cursor usually does not have an immediate effect. There are regions of interest, like push buttons, within which actions are possible. Even then the combination of a certain cursor position and a click is required to initiate an action. Depending on the effect of the action, the user has to confirm his intent by again pointing and clicking. As a consequence, accurate pointing is of minor importance for this application.

A few subjects criticized the calibration routine for position mode when asked for suggestions to improve control. They found it difficult to turn their heads towards a certain point in a reproducible manner. Therefore, the mapping slightly varied between different calibration procedures. In future work, more sophisticated calibration routines will be tested.

To conclude, position mode is recommended for the applications which require quick, intuitive control. For safety-critical applications and precise control, the velocity mode is the preferred choice.

V. CONCLUSION AND FUTURE WORK

Within this paper, we evaluated position and velocity mode. While position mode is quick and intuitive, it lacks the required robustness for safety-critical applications. It is

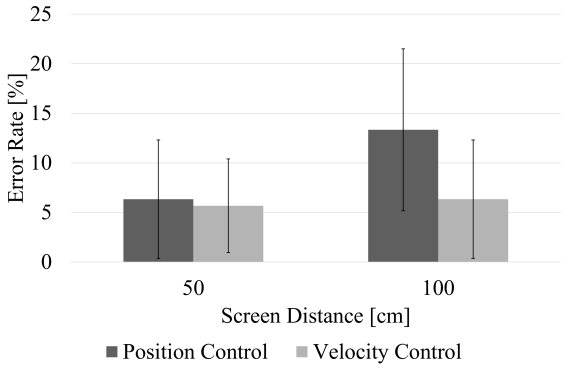

Fig. 6. Error rate for both position and velocity mode at screen distances of 50 cm and 100 cm. At a distance of 50 cm the average error rate of position mode was $6.33\% \pm 5.97\%$ and increased significantly to $13.33\% \pm 8.17\%$ when the screen was put further away. The error rate of velocity mode was not influenced by screen distance. It was $5.67\% \pm 4.73\%$ at 50 cm distance and $6.33\% \pm 5.97\%$ at 100 cm distance.

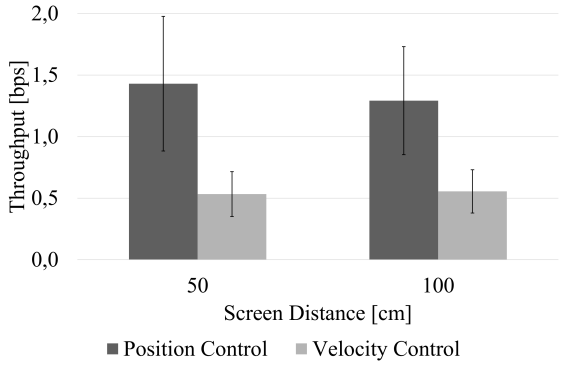

Fig. 7. Throughput measured in bits per second (bps) for both position and velocity mode at screen distances of 50 cm and 100 cm. At 50 cm distance the average throughput of position mode was $1.43\,\text{bps} \pm 0.55\,\text{bps}$. There was a slight decrease to $1.29\,\text{bps} \pm 0.44\,\text{bps}$ when the screen was moved to the 100 cm position. The throughput of velocity mode stayed almost constant. It was $0.53\,\text{bps} \pm 0.18\,\text{bps}$ at 50 cm distance and $0.56\,\text{bps} \pm 0.18\,\text{bps}$ at 100 cm distance.

therefore appropriate for applications like cursor control. Other applications may require precise positioning, such as direct robot control. In this case, the more robust velocity mode is the preferred choice although is it considerable slower and less intuitive.

As a result, whether the position or velocity mode is the better control mode, depends on the application and the preferences of each individual user. Therefore, both position and velocity mode will be part of our future investigations. This includes improving the calibration routine of position mode in order to increase its reproducibility. In addition, various filtering techniques, e.g. linear weighted moving average (LWMA), will be tested. The aim is to remove the jitter that occurs during position mode due to unintended movements, like natural tremor or breathing. The main drawbacks of velocity mode are the lower operating speed and the required mental effort. Future work will focus on reducing mental effort as operating speed increases with growing user experience and decreasing mental effort.

ACKNOWLEDGEMENT

The authors would like to thank Dr. Danijela Ristic-Durrant and Hillcrest Labs for their outstanding support as well as all the volunteers who took part in the experiments.

This work was funded by the Ministry for Innovation, Science and Research of North Rhine-Westphalia.

REFERENCES

[1] C. Martens, O. Prenzel, and A. Gräser, *The Rehabilitation Robots FRIEND-I & II: Daily Life Independency through Semi-Autonomous Task-Execution*. I-Tech Education Publishing, 2007, ch. 9. [Online]. Available: http://intechweb.org/book.php?id=19

[2] InvenSense Inc. (2012, Jun.) Motion sensors introduction. White Paper. [Online]. Available: http://invensense.com/mems/gyro/documents/whitepapers/1Sensor%20Introduction.pdf

[3] I. Lee, G. H. Yoon, J. Park, S. Seok, K. Chun, and K.-I. Lee, "Development and analysis of the vertical capacitive accelerometer," *Sensors and Actuators A: Physical*, vol. 119, no. 1, pp. 8–18, 2005.

[4] Hillcrest Laboratories, Inc. (2013, Sep.) FSM-9 Data Sheet. [Online]. Available: http://hillcrestlabs.com/

[5] J. Bernstein, S. Cho, A. King, A. Kourepenis, P. Maciel, and M. Weinberg, "A micromachined comb-drive tuning fork rate gyroscope," in *Micro Electro Mechanical Systems, 1993, MEMS '93, Proceedings An Investigation of Micro Structures, Sensors, Actuators, Machines and Systems. IEEE.* IEEE, 1993, pp. 143–148.

[6] Bosch Sensortec GmbH. (2014, Jul.) BMI055: Data sheet. [Online]. Available: https://ae-bst.resource.bosch.com/media/products/dokumente/bmi055/BST-BMI055-DS000-08.pdf

[7] ——. (2014, Nov.) BMX055: Data sheet. [Online]. Available: https://ae-bst.resource.bosch.com/media/products/dokumente/bmx055/BST-BMX055-DS000-02.pdf

[8] M. Bureau, J. Azkoitia, G. Ezmendi, I. Manterola, H. Zabaleta, M. Perez, and J. Medina, "Non-invasive, wireless and universal interface for the control of peripheral devices by means of head movements," in *IEEE 10th International Conference on Rehabilitation Robotics (ICORR)*. IEEE, 2007, pp. 124–131.

[9] C. Mandel, T. Rofer, and U. Frese, "Applying a 3DOF orientation tracker as a human-robot interface for autonomous wheelchairs," in *Rehabilitation Robotics, 2007. ICORR 2007. IEEE 10th International Conference on*, June 2007, pp. 52–59.

[10] R. Raya, J. Roa, E. Rocon, R. Ceres, and J. Pons, "Wearable inertial mouse for children with physical and cognitive impairments," *Sensors and Actuators A: Physical*, vol. 162, no. 2, pp. 248–259, 2010.

[11] L. Zhang, "Investigation of coupling patterns of the cervical spine," Master's thesis, University of Dundee, 2014.

[12] N. Rudigkeit, M. Gebhard, and A. Gräser, "Towards a user-friendly AHRS-based human-machine interface for a semi-autonomous robot," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS2014), Workshop on Assistive Robotics for Individuals with Disabilities: HRI Issues and Beyond*, Chicago, IL, 14 Sep. 2014.

[13] Wikipedia. (2015, Jan.) Gompertz function — Wikipedia, the free encyclopedia. [Online]. Available: http://en.wikipedia.org/wiki/Gompertz_function

[14] R. Soukoreff and I. MacKenzie, "Towards a standard for pointing device evaluation, perspectives on 27 years of Fitts' law research in HCI," *International Journal of Human Computer Studies*, vol. 61, pp. 751–789, 2004.

[15] Wikipedia. (2015, Mar.) Welch's t test — Wikipedia, the free encyclopedia. [Online]. Available: http://en.wikipedia.org/wiki/Welch%27s_t_test