Recognition of Simple Head Gestures Based on Head Pose Estimation Analysis

George Galanakis
Institute of Computer Science,
Foundation for Research and Technology Hellas
Computer Science Department,
University of Crete, Greece
e-mail: ggalan@ics.forth.gr

Pavlos Katsifarakis
Computer Science Department,
College of Science
Swansea University, UK
e-mail: 750183@swansea.ac.uk

Xenophon Zabulis, Ilia Adami
Institute of Computer Science,
Foundation for Research and Technology Hellas
e-mail: {zabulis, iadami}@ics.forth.gr

***Abstract*—A recognition method for simple gestures is proposed and evaluated. Such gestures are of interest as they are the primitive elements of more complex gestures utilized in natural communication and human computer interaction. The input to the recognition method is obtained from a head tracker that is based on images acquired from a depth camera. Candidate gestures are detected within continuous head motion and recognized, acknowledging that head pose estimates might be inaccurate. The proposed method is evaluated within the context of human-computer dialog.The reported results show that the proposed approach yields competitive recognition results to state-of-the-art approaches.**

***Index Terms*—head gesture recognition; head gesture detection.**

I. Introduction

The ability to recognize purposeful head motions, or gestures, is a special problem both in computer vision and human-computer interaction. Solving this problem accurately and robustly is of particular interest, because such head motions convey information that can be used in the natural communication of a person with a computer, or an intelligent environment. In this work, head gesture recognition targets purposeful head motions that are responses to a user interface dialog.

A central component of any head gesture recognition system is the estimation of head pose ([1]–[3]) and motion. Head pose information is of particular importance in a variety of applications and has received considerable attention in the recent years [4]. The selection of the sensory modality is important, as it relates to the reliability of this estimation which can, in turn, affect the performance of recognition. Practicality and applicability through cost-efficient and off-the-shelf hardware is also of concern and, thus, this work employs a state-of-the-art head tracker that is based on commodity depth cameras, nowadays widely available as Red Green Blue Depth (RGBD) sensors [5].

In most cases, natural gestures can be analyzed in simpler motions. For example, a horizontal shaking of the head to express negation, is usually repetitive. Moreover, each one of the repeated motions can be further analyzed. In the aforementioned example, the gesture can be regarded as a leftward and a rightward head rotation (or vice versa). In this work, we focus on the recognition of such simple motions, which we call primitive gestures. Our interest is twofold. First, due to their simplicity, these gestures are suitable for use in human computer dialogues. Second, primitive gestures are elements of higher order gestures and, thereby, their robust recognition is relevant to the recognition of more complex gestures.

In the context of this work, we use the notion of a reference head pose which, in our case, is the frontal (or, "looking straight ahead") pose. We also parameterize, human head 3D orientation upon the natural head rotations, which are called yaw, pitch, and roll (see Fig. 2). In this reference, primitive gestures correspond to a peak in the values of an angular component, while no significant rotation occurs in the remaining two angular components.

To determine the extent that the proposed method can be useful in human computer interaction we evaluate it through quantitative evaluation, in which recognition performance is measured. At the same time, this evaluation serves a secondary goal. By observing and profiling the way that subjects perform primitive gestures (i.e., how fast or how steep is a head rotation), information regarding the corresponding user motions is collected. In turn, this information can be exploited in the better recognition of these gestures and the design of systems that utilized them.

The rest of this paper is organized as follows. Section II presents related work on head gesture recognition methods and applications, Section III includes implementation details, Section IV discusses experiments and results, Section V briefly presents the applicability of head gestures within a specific example application, and Section VI concludes the presented work and suggests further applications in which head gestures can be employed.

II. Related Work

Work on the recognition of head gestures has started to emerge as long as two decades ago, but has been recently reinforced after the wide availability of depth cameras, which facilitate the pose estimation of the human head.

Some approaches to head gesture recognition capitalize on a special type of sensor (i.e., inertial [6] or pupil tracking [7]) and setup, which provides confidence to the input signal from head pose estimation. In turn, this input signal exhibits increased continuity and reduced noise and its processing is, thereby, simpler. At the other end, some approaches employ a fully passive (RGB or monochrome) camera to estimate head pose. Pertinent methods rely on facial feature detection (i.e., mouth, nose, eyebrows) and tracking to acquire head pose and motion [7]–[9]. In [10], direct measurements (pixel intensities) are utilized, but resorting to assumptions about the facial appearance of the subject and providing less accurate

 ISBN: 978-1-61208-356-8

results. In terms of sensory input, this work falls in the middle of the above range, utilizing a commodity RGBD sensor, as in [11]. Only the depth information is utilized to avoid sensitivity to illumination. However, although depth information is much more robust than color/intensity, input cannot be considered neither noise nor error free. In this context, this work accounts for poor, erroneous, or missing estimates provided as input.

The methods employed for head gesture recognition can be classified into two main categories, those which employ a Finite State Machine (FSM) and those which are based on learning, typically through an instantiation of Hidden Markov Models (HMMs).

Simple gestures, such as the one of interest in this work, have been recognized by a number of methods that employ FSMs. FSMs are simple to formulate but, in the other hand, do not scale with ease. In [7], an FSM recognizes nodding and shaking gestures, which are then used in the context of a dialog-based user interface. The same FSM is used by [8] in a self-portrait camera which is controlled by nodding and shaking gestures. In [12], an FSM is introduced for detecting nodding and shaking gestures, useful for interacting with avatars on mobile devices. In [13], FSM-detected head gestures have been used along with hand gestures in order to achieve interaction within a multi-modal user interface. The above methods lead to the use of rather complex FSMs in order to accommodate multiple gestures, while still support a smaller vocabulary of gestures (typically 4, based on up, down, left, and right motions).

Methods based on machine learning and recognition of temporal patterns techniques are also present in the literature. Recently, in [11], two HMMs are trained to recognize nod and shake gestures; "other" gestures are recognized by a third HMM as fallback. In [9], a HMM is trained for each of three different head movement gestures; right, left, left-forward, which are used in the context of sign language sentences. In [14], shaking, neutral and nodding gestures are detected by continuous HMMs and then provided to a dialog manager which operates a coffee machine. Similarly, in [15], Ordered means models (OMMs) are trained to recognize nod, shake, tilt and look gestures among two participants in a conversation; OMMs, are described as "rigorously reduced versions of HMMs. In [16], a multi-class Support Vector Machine (SVM) is augmented with contextual features, to recognize nod and shake gestures. These gestures are evaluated in the context of document browsing and dialog box confirmation. Finally, in [17], a multi-class SVM is trained to detect "Yes" and "No" head gestures, along with other hand gestures. In the same context, some methods learn gestures directly from posture data such as [6] which operates on head orientation readings to detect nodding and shaking gestures. In [10], a set of ten gestures is recognized by Continuous Dynamic Programming which compares live images with previously trained image sequences, annotated respectively.

The works above employ HMMs to treat gestures that contain multiple more simple gestures, resulting in complex gesture models, while considerable effort is required for the training of the system. In comparison to HMMs and SVMs the proposed work does not require a preceding training phase, but capitalizes upon the examination of each rotational component of the head pose. Moreover, it is concluded that the results of the proposed work, as shown in Section IV-A1, are not only comparable but, in most cases, outperform recognition rates in the literature.

Fig. 1. The raw input depth from the tracker (left) and the head pose estimate superimposed on the input color image.

III. METHOD

A. Sensory input

A head tracker [1], that receives input from an RGBD camera is employed to sense the current pose of the subject's head, in real-time. Fig. 1 illustrates the result of the tracker for a given input image. It is noted that any other head tracker (i.e., [2], [3]) could be used instead of this one, however the particular one was selected due to its reported increased accuracy and execution speed. The input is either the estimated 3D pose or null (in case of tracking failure) and is received multiple times per second. Acknowledging that erroneous or inaccurate estimates may be provided, as well as, that tracking may exhibit transient failures this information stream is adopted as the "sensory input" to the proposed system.

As the head tracker and the recognition system which we developed are implemented in different programming environments, their communication was achieved through a service interoperability platform [18].

3D pose is defined as the 3D translation and 3D rotation of the head from a reference pose and is, thereby, represented by 6 degrees of freedom (6 DoF). These DoF correspond to a translation 3D vector and a 3D rotation which is parameterized as in terms of Euler rotations, that is, as a rotation of the head about the xx', yy', and zz' axes. These rotations are referred as P_i, Y_i and R_i respectively (see Fig. 2).

In the context of this work, translation does not play a primary role as we assume the expression of gestures to be invariant to the translational motion of the head and that they can, also, be expressed while the subject is in motion. We also assume that rapid and large motions of the subject's head, which would influence the comprehension of a gesture do not occur as they are not typically performed by subjects.

B. Parsing of candidate gestures

Before describing recognition approach, we model the pursued primitive gesture, as a motion which starts and ends at the reference pose and, in between, a single peak of significant

 ISBN: 978-1-61208-356-8

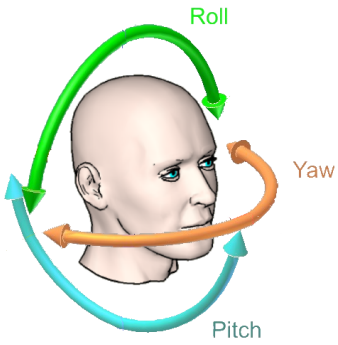

Fig. 2. Head rotation axes.

amplitude in the value of an orientation component occurs. Candidate gestures consist of a sequence of three head states;

1) postured and approximately motionless for a brief time interval at the reference pose,
2) performing a rotational motion and, possibly, a mild translation motion, and
3) postured again at the reference pose for a brief time interval,

Using this description we are able to "parse" the continuous sensory input into constituent, discrete elements. Each such element is then considered as candidate gesture. A candidate gesture is a head motion that might be expressing a gesture, or not. Each such candidate, is attempted to be, correctly, recognized as a gesture or as non-gesture. For candidates recognized as expressing a known gesture, labeling of the particular gesture is also attempted.

Depending on the type of motion of the second state, the gesture may be recognized as an instance of the known gestures, or not. The reference pose is defined as the pose of the head at approximately zero rotation in all the 3 axes. We have extrinsically calibrated our camera and estimated its relevant posture to the ground plane through, conventional, grid-based calibration [19]. In this way, we performed a change of reference coordinate and poses so that the reference pose, in our setup, this corresponds to the user's head facing frontally without any inclination of the head. The reference pose is defined to occur when $\forall\ p \in \{P_i, Y_i, R_i\},\ |p| < \tau_r$, where τ_r is a configurable threshold relaxing the requirement for exact frontal posture and is in the order of a few degrees ($10°$).

To parse candidate gestures we defined a simple state-machine, with parameterization in the transition of the states. We call it Buffered State Machine because the transition from a state to another is performed when a buffer is completed by a number n of valid tokens. This means that we have to acquire n consecutive poses in reference position to start identifying the gesture. This stabilizes the system against small estimation errors. The value of n is configurable with respect to the frame rate that the head tracker operates. In our implementation the value $n = 5$ was selected, based on preliminary observations of user behavior, adjusting the head rest at the reference pose

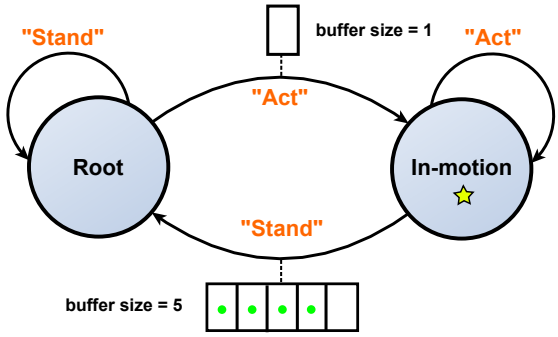

Fig. 3. The state machine which was employed to parse candidate gestures.

to have an (approximate) duration between $1/3 - 1/2$ $sec.$ Fig. 3 illustrates the described state-machine, being in "In-motion" state. In the particular case, if we receive one more "Stand" token (command), a transition to "Root" will be performed. Otherwise, the buffer will be invalidated, since we want 5 consequent "Stand" tokens.

When the subject's head is detected by the sensor, the pose estimation is continuous and the recognition component stores the estimations in a double buffer. Whenever the transition to the reference pose occurs, the current buffer is "parsed" and passed to the next stage for recognition, while the other buffer stores the more recent poses. The above is feasible because poses are received as events via the interoperability platform and are handled by a different thread. In cases of head pose estimation failures, a null result (estimate) is produced. In such cases, the recognition will stop receiving events until the tracker resumes operation. If such an event occurs during the expression of a gesture then, typically, the gesture fails to be recognized.

C. Gesture detection and recognition

Upon parsing of the gesture, the signal segment acquired during the "In-motion" state is assessed, in order to reason whether the candidate is indeed a primitive gesture and, if so, recognize which one it is.

To detect a gesture we investigate the content of the rotational components of this signal segment. We test for two conditions, for this purpose. The first is that the motion in the rotational component corresponding to a particular gesture matches the prototype of the gesture. Fig. 4 illustrates a prototype motion as assumed above. The second is that the remaining 2 rotational components do not correspond to a significant motion.

To test for the first condition, we consider the values of the 3 rotational components (pitch, yaw, roll) of the pose estimates. Each component is independently processed and its input is treated as a stream. Prior to its consideration, each stream is passed through a low-pass, Gaussian filter to eliminate tracking jitter. Henceforth, we call the signal of a rotational component within the time interval $[t_A, t_B]$ as dominant, if it is the sole one exhibiting significant motion. For example, Fig. 5(a) illustrates the acquired sensory input for the Y rotational component, at a time interval which is

 ISBN: 978-1-61208-356-8

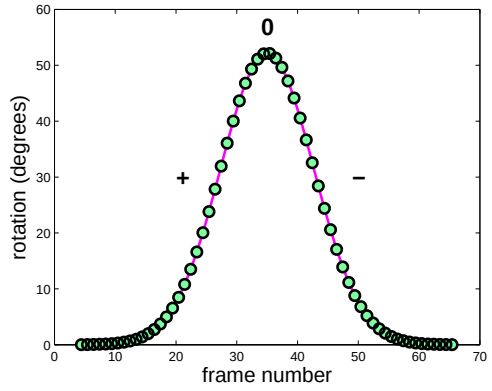

Fig. 4. Prototype of head motion as a function of the rotational component corresponding to the axis of the particular rotation.

segmented (dashed blue lines) by the Buffered State Machine, while Fig. 5(b) shows the output from its low-pass filtering. Note that, in Fig. 5(b), only the part between the two dashed blue lines is shown and, thus, they are omitted.

The second condition implies that, in addition to detecting a dominant motion we need to determine that motions in the remaining 2 components are insignificant, or henceforth "neutral" motions. For this purpose, "soft" thresholds (s_T) are defined. A "soft" threshold specifies the accepted amount of motion in a rotational component, when it is considered irrelevant to a gesture. For instance, when a "Head Up" gesture is performed, we do not expect significant motion in the yaw component. Henceforth, we call the signal of a rotational component within the time interval $[t_A, t_B]$ as neutral if it does not surpass the soft threshold s_T.

In order to recognize a gesture, each rotational component is investigated separately. Let f be a function of time which represents the value of the rotational component in consideration (pitch, yaw, roll). Let also $[t_A, t_B]$ the time interval for which the signal of the above component was acquired. As the primitive gestures to be recognized have the form of a peak, in the dominant rotational axis, candidate gestures are first tested as to whether they exhibit the potential of containing such a peak. This consists of the fulfillment of the following three conditions:

1) a single peak of f occurs during the entire interval
2) f advances in a strictly positive (negative) followed by a strictly negative (positive) manner around the peak
3) a threshold h_T is overcome, so that the peak exhibits significant amplitude to be attributed to an intensional gesture rather than an unintentional head motion.

In Fig. 5, characteristic data are shown for the Y component of rotation for the ideal model of a Head Down gesture, the acquired sensory input, and the output from its low-pass filtering. The implementation of these three conditions is as follows.

First, the peak has to be single; $\forall t_i \in [t_A, t_B]$, f has a single peak e. The reason is that recurring motions during the "In-motion" state should be omitted. Thereby, the zero-crossings

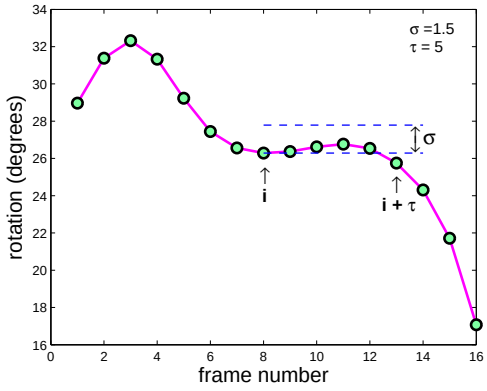

Fig. 6. A close-up of the special case displayed in Fig.5(d).

of the first derivative of f, f', are detected and counted. Detection of zero-crossings is performed by a, conventional rule, which is that if f' exhibits a zero-crossing within $[t_{i-1}, t_i]$ then $f'(t_{i-1}) \cdot f'(t_i) < 0$ should hold for exactly one i. Fig. 5(c) shows a filtered signal in which two peeks occur and recognition fails.

Second, the sign of the peak is specified. When the sign of the peak is positive the following condition must hold:

$$\forall t_i \in [t_A, e), sgn(f) = 1 \wedge \forall t_i \in (e, t_B], sgn(f) = -1, \quad (1)$$

while when it is negative the corresponding condition becomes:

$$\forall t_i \in [t_A, e), sgn(f) = -1 \wedge \forall t_i \in (e, t_B], sgn(f) = 1 \quad (2)$$

In the above, $sgn()$ denotes the sign function. Two options regulate how literate the requirement for the function f being strictly positive or negative. The accepted jitter is specified by threshold σ in rotation axis and τ in time axis, so that the following should hold:

$$\begin{cases} f(t_i) - f(t_i - 1) \leq \sigma & \forall t_i \in [t_A, e) \\ f(t_i) > f(t_j),\ t_j < t_i + \tau,\ j \in [i, i + \tau] & \\ & \\ f(t_i - 1) - f(t_i) \leq \sigma & \forall t_i \in (e, t_B] \\ f(t_i) < f(t_j),\ t_j < t_i + \tau,\ j \in [i, i + \tau] & \end{cases} \quad (3)$$

Fig. 5(d) illustrates such case of a permitted peak, that is treated as jitter. The segment of interest is presented along with the fulfilled requirements σ and τ in Fig. 6.

Third, a "hard" threshold (h_T) has to be overcome, such that $|e| > h_T$. Thresholds may be different across rotation axes, due to anatomical differences in head rotation about each axis. In particular for the pitch axis, the positive and negative thresholds are different as well; $|h_T^+| \neq |h_T^-|$ and $|s_T^+| \neq |s_T^-|$.

Both thresholds h_T, s_T, are empirically adjusted, based on the experimental user studies of Section IV. In total, 12 different thresholds were adjusted, based on the following combinations of h_T, s_T with each rotational component and the sign of the peak as note by the Cartesian product of the corresponding sets: $\{h_T, s_T\} \times \{pitch, yaw, roll\} \times \{positive, negative\}$.

 ISBN: 978-1-61208-356-8

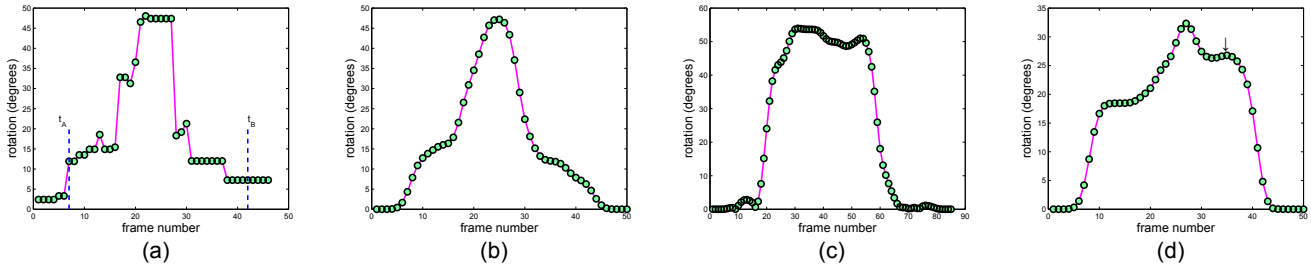

Fig. 5. Characteristic cases of acquired data, showing head motion as a function of the Y rotational component, for expressed "Head Down" gestures. See Section III-C.

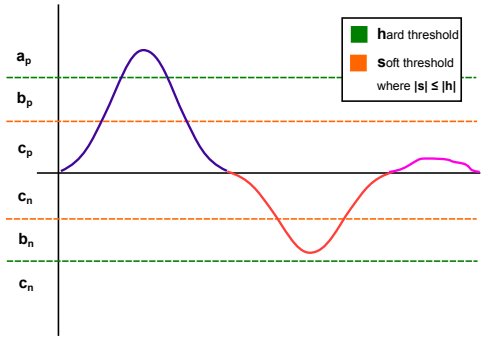

Fig. 7. A demonstration of the thresholds. See Section III-C.

The combination of the the three rotational axes with the sign of the peak results into 6 primitive gestures. Thereby, following the examination of the motion, a gesture is recognized as follows:

- "Head Up" when pitch is negative and dominant, while yaw and roll are neutral
- "Head Down" when pitch is positive and dominant, while yaw and roll are neutral
- "Head Right" when yaw is positive and dominant, while pitch and roll are neutral
- "Head Left" when yaw is negative and dominant, while pitch and roll are neutral
- "Roll Right" when roll is negative and dominant, while pitch and yaw are neutral
- "Roll Left" when roll is positive and dominant, while pitch and yaw are neutral

Fig. 7 illustrates the recognition processing by providing an example of a rotational component, in this case pitch. The signal has been already parsed in three segments, indicated by the corresponding three colors of the curve, by virtue of the process described in Section III-B.

The blue segment is positive and dominant, so we have to examine yaw and roll; if they are neutral then a "Head Down" is recognized. The red segment is negative. Its peek is below h_T but above s_T, which means none of the gestures will be recognized. The magenta segment is positive and neutral on the shown axis; a gesture might be recognized if one of the other rotational components (not shown) is dominant during this time interval.

IV. Experiments

The system was run on a personal computer (PC) with an Intel Core i7, at $2.67\,GHz$ with $6\,GB$ of random access memory and an NVIDIA GTX680 graphics processing unit (GPU). The head tracker was executed on the GPU while gesture recognition on the central processing unit of that PC. The head tracker offered estimates at a rate of $15\,Hz$.

The system was evaluated with the help of 13 test users, all naive to the experimental hypotheses. All test users had normal hearing and did not experience any kinetic problems.

The setup of the experiment included a 480×640 depth camera (an RGBD Kinect sensor) adjusted to a floor mount, and a chair in front of the mount at a distance of $\approx 1\,m$ (see Fig. 8). The sensor was adjusted so that it was at a height comfortable for each individual user. To avoid visual disruption during the experiment, the monitor of the PC was not present.

The evaluation task was enabled by a software module that was developed for the purposes of this evaluation. The system employed a speech synthesizer to prompt the user to perform a gesture and to provide feedback regarding its recognition. During an evaluation session, the system attempted to recognize gestures performed by the user, in individual trials. Each evaluation session, was comprised of 18 trials, testing the recognition of the 6 studied head gestures; 3 trials were dedicated for each gesture type. The execution order of the trials was decided randomly at each session, by the system.

The evaluation task was the following. The system would prompt the user to perform a particular gesture. Upon announcement of this prompt, the system monitored the user. If a gesture was recognized thereafter, the user was informed of the occurrence of the recognition event and the label of recognition. If a gesture was not recognized or if a different than the prompted gesture was recognized the system provided feedback. This feedback pointed out the unexpected outcome and, also, prompted the user to repeat the trial, up to two additional times. A trial was complete upon recognition of the prompted gesture or if three recognition failures occurred. When a trial was complete the system proceeded to the next trial. During the evaluation, the user had the option to pause the process in order to rest and continue later.

 ISBN: 978-1-61208-356-8

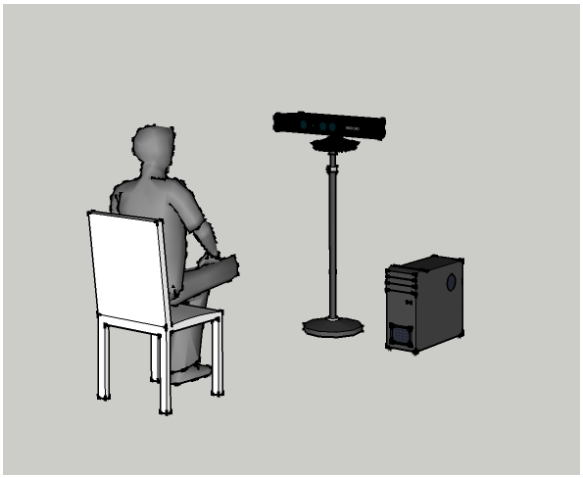

Fig. 8. The experiment setup. A user is sitting in front of a sensor which is adjusted to his height.

Before each session the test user was informed about the required gestures, and was asked to rehearse, in order to validate their comprehension. Also, in the beginning of each session, the system was initialized by acquiring a frontal head pose that was the reference pose for the individual user.

During the evaluation, we kept notes of the events. At the end of each session, the test users were interviewed about their experience, whether they had any difficulties achieving the task, whether they fully comprehended the provided feedback etc.

A. Quantitative analysis

1) Recognition accuracy: The recognition accuracy of the method was measured in terms of percentage of correct recognitions. The results are shown in Tab. I and in Tab. II. The first column of Tab. I shows the percentage of correct detection in the first gesture attempt, while the second column shows the percentage of correct detection after the third attempt. The high percentages of the second column indicate that users adjusted their gestures after the first or second failure to match the user expected recognition requirements of the system.

The first column of Tab. II, shows the sum of misses in all trials and the second column shows the proportion of the recognition errors, or otherwise, how many of the misses were recognized as another gesture. In all cases, recognition errors occurred due to pose estimation failures.

It is important to note that no false positives detections of gestures occurred in any of the experiments and this is due to mainly two reasons: the head tracker is very accurate in the pose estimates that it provides and the time interval for the detection of gesture was constrained by the experiment task (i.e., the time the user had to perform the gesture was guided by the system).

In further analysis of the results shown in Tab. I and II, more conclusions can be drawn for the gestures that received lower recognition accuracy scores. For example, in the case of the "Head Down" gesture, it was concluded that the lower score is due to the inability of the head tracker to calculate the position

TABLE I
RECOGNITION ACCURACY

Gesture	First time recognized	Any time recognized
Head Up	95%	100%
Head Down	72%	100%
Head Right	92%	100%
Head Left	64%	90%
Roll Left	74%	100%
Roll Right	74%	97%

TABLE II
MISSES DURING THE EXPERIMENTS

Gesture	Sum of misses in all trials	Recognition errors
Head Up	2	0%
Head Down	12	16.67%
Head Right	3	0%
Head Left	22	9.09%
Roll Left	13	0%
Roll Right	13	7.69%

of the head because the face becomes self-occluded and the image avails less facial information. In the case of the "Roll left" and "Roll Right" gestures, the lower recognition accuracy percentages are not due to any shortcomings of the tracker, but rather due to the fact that since this gesture is not a commonly performed gesture, its execution range varies from person to person. Finally, an interesting result is related to the accuracy of the "Head Left" gesture in contrast to its relevant "Head Right". From our investigation, half of the failures occurred because s_T^- of the pitch component was surpassed, meaning that participants unexpectedly inclined their head to the up direction while turning to the left. Such behavior should be investigated in further experiments though.

Some of the proposed works mentioned in Section II provide accuracy evaluations in order to prove the reliability of their systems. Though they are not directly comparable due to differences in gestures and head pose estimation method, we discuss the relationship of the proposed work to the state of the art. In [10], where a training phase is preceded, a ratio of 97% of the gestures are successfully recognized when the test user is the same with the person used for the training, while this ratio falls to 80% when they are different persons. In [14], accuracy depends on the states of the trained HMM, and it varies from 88% to 100%. In [17], "yes" and "no" gestures are recognized with a ratio of 88% and 77% respectively, while most of the other hand-based gestures are recognized at higher ratios. In [13], the recognition rate on the head gestures is over 92%, while in [6], 76.4% of the "nodding" and 80% of the "shaking" gestures are recognized. In [11], a recognition average ratio of 86% is reported. The

 ISBN: 978-1-61208-356-8

context-based approach in [16], increases the recognition ratio of the "nod" gestures which reaches 91%. Finally, in [15], the classification rates range from 75.95% to 98.4% when the training subjects are different from the testing. For a particular gesture, the ratio is 44.84%, though. It is noted that all classification-based methods include mismatches in recognitions, because a decision is made among all classes, but in most cases such a ratio is acceptable. Furthermore, all recognition methods with support of natural interaction have a failure rate. In general, the recognition accuracy depends on the number of the recognized gestures, on their complexity, but are also related to the proposed method. As it was presented in Section IV-A1, the average recognition ratio of our method ranges from 78.5% average, to 97.8% average when users familiarize with the system. We conclude that the proposed work offers results that are not only comparable but, in most cases, outperform reported recognition rates.

In our case, the results indicate that the proposed method can be reliably employed in human-computer dialog applications. As shown, false positive recognitions are rare, but are also undesirable in many cases. In order to overtake such situations, a dialog could expect from the user an extra confirmation. For example if a "Head Down" gesture is utilized as a "yes", then the dialog could expect it twice. Alternatively, the dialog could inform the user about the recognized gesture, permitting a period of cancellation which will be triggered by a gesture or by a simple posture outside the reference position; an invalid gesture. A different option for reducing false positives is to place a restricted time interval for gesture expression, as discussed below.

2) Gesture execution time: Another measurement we acquired, was the execution time of each gesture. Fig. 9 shows the distribution of the recognized or non-recognized gestures at each time-slot. The chart shows that, for the majority of gestures, execution time was below $2\,sec$. As we noticed during the experiments, large execution times were sometimes present due to pose estimation failures and thereby measured execution times were greater than actual (that is, due to a recognition failure the system kept waiting for a gesture to occur but to no avail).

We conclude that as gestures typically occur during a $2\,sec$ limited interval, it is for the benefit of an application that uses such gesture to avail a similar time interval for gesture expression, during a user interface dialog (and, in case of recognition failure, prompt the user to execute the gesture again). In this way, gesture recognition becomes more reliable as potential false positive recognitions are avoided. In addition, in cases of recognition failure, the system becomes more responsive, quickly prompting the user to execute the unrecognized gesture again, instead of letting the user wait for an unnecessary longer timeout.

3) User investigation: In preliminary experiments, the hard and soft thresholds h_T and s_T were initially fixed at the same values for all of the rotational components. However, we noticed that subjects did not perform rotations of the same magnitude on each axis, due to anatomical reasons (i.e., users typically do not lower the head as many degrees as they rotate it horizontally). We also observed that when performing different rotations, amplitude of the neutral (irrelevant) rotations differed. Hence we analyzed the behavior of the subjects in order to see if the thresholds required for the recognition could be adapted for each rotational axis to the benefit of recognition rates.

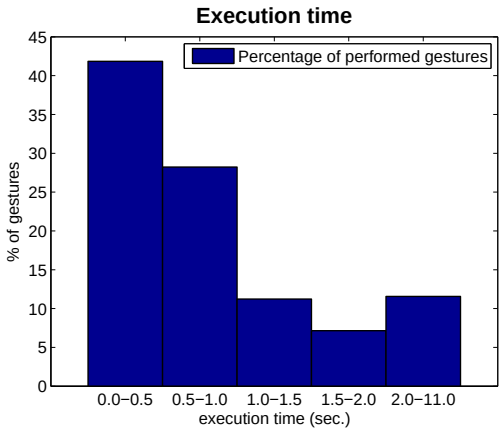

Fig. 9. Distribution of the gestures depending on the execution time. Most of them performed at below $2\,sec$..

The resulting per-axis maximum angle for each single gesture of the evaluation was stored and two types of diagrams were formed (Fig. 10 & Fig. 11). The dominant angle diagram (Fig. 10) displays the distribution of the angle on an axis when a gesture related to this axis was required. Fig. 10 shows the performed angles when a "Head Left" gesture was required; that is the graphs shows the values of the dominant rotational component. The yellow line depicts the h_T. The neutral angle diagram (Fig. 11) shows the distribution of the same angle for the neutral rotational components. Fig. 11 depicts the distribution in the yaw axis, when gestures different than "Head Left" were prompted. The yellow line shows the s_T, which is equal to h_T in this case. Both diagrams show additional information about the first attempt to perform the gesture, which is marked by the green dots, while magenta dots mark the repeats.

Following the preliminary experiments, the thresholds were tuned. The tuning accomplished for both h_T and s_T in the following ranges;

- $[10°, 25°]$ for the h_T^+ and s_T^+ of the pitch component
- $[15°, 25°]$ for the h_T^+ and s_T^+ of the yaw and roll components
- $[-15°, -25°]$ for the h_T^- and s_T^- of the pitch, yaw and roll components

Eventually, all the thresholds were adjusted as Tab III shows. We notice that the h_T^+ and s_T^+ of the pitch component, which are related to the "Head Down" gesture, have a lower absolute value than the others. This can be explained by the anatomy of the neck, which allows a smaller inclination of the head when it is directed down.

For the similar reasons as above, we measured also the ranges of rotational motions for the recognized gestures. As

 ISBN: 978-1-61208-356-8

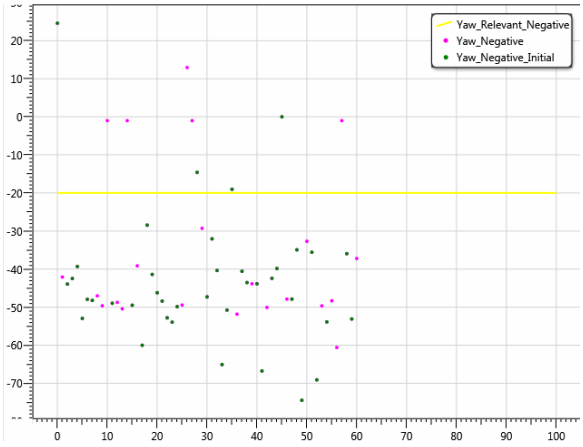

Fig. 10. Distribution of the peaks of the yaw rotational component, when a dominant to negative yaw gesture was expressed.

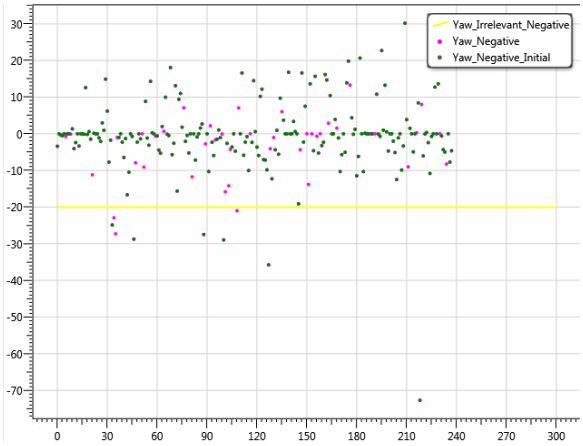

Fig. 11. Distribution of the peaks of the yaw rotational component, when an neutral to yaw gesture was expressed.

mentioned in SectionIII-C, each gesture is related with an axis of rotation. Tab.IV shows the ranges of the per gesture relevant axis, of the truly recognized gestures, and the mean and standard deviation, as well. These observations complement the analysis for tuning the thresholds and besides the general interest as a user study, can be used to tune parameters of the head tracker for optimization of its performance (i.e., by bounding the head pose estimation search space).

B. Qualitative analysis of head gestures

At the end of the experiment, the participants were asked to express any thoughts they had during the execution of the gestures, i.e., if they felt that the tracker was recognizing them easily, if they experienced any fatigue, and so on. Eight of

TABLE III
RECOGNITION THRESHOLDS

	h_T^+	h_T^-	s_T^+	s_T^-
pitch	15°	-20°	15°	-20°
yaw	20°	-20°	20°	-20°
roll	20°	-20°	20°	-20°

TABLE IV
RANGES OF RECOGNIZED GESTURES

	Min	Max	mean	stdev
Head Up	-51.44°	-21.49°	-35.41°	9.55°
Head Down	17.10°	58.49°	30.92°	8.30°
Head Right	26.65°	57.22°	42.82°	8.38°
Head Left	-65.08°	-28.45°	-43.94°	8.24°
Roll Right	-44.72°	-21.56°	-31.21°	6.09°
Roll Left	20.15°	54.01°	35.44°	8.89°

the participants said that the instructions given were clear and that the tracker behaved as expected. One participant said he was uncertain of the 'required' speed the gesture had to be performed in, in order to be recognized by the system. Another participant said that he intentionally performed the gestures in a wider than usual range in order to facilitate the system in recognizing it. The above two comments indicate that some users are just not aware or familiar with the hardware system capabilities. In addition, three participants mentioned neck fatigue especially caused by the "Roll" gestures, another thought that "Roll" is an unnatural motion, suggesting diagonal ones instead. Finally, two participants named issues with the "Head Down" gesture, but as of our observation during the evaluation, these caused by estimation errors due to the self-occlusion of largely bent head relative to the camera.

V. PILOT APPLICATION

The evaluation discussed in Section IV targeted the interaction with dialogs. In a dialog application the system prompts the user to provide input in the form of gestures. Other applications though, let the user interact with the system in a continuous manner; they handle events which are emitted by the available input devices (i.e., mouse clicks or keyboard strokes). Considering this, every gesture recognition system can be regarded as an input device. A primary difference of these systems with an everyday input device, is that the user is supposed to concentrate in the interaction, with limited habitual or natural movements, in favor of preventing false recognitions. In a spectrum of applications this user cooperation can be assumed, as gesture interaction is an essential communication modality for people with mobility difficulties.

For our demo the publicly available labyrinth/puzzle game called Bloxorz [20] was adopted. The recognized gestures were associated with keyboard events, which were then operated the subject of the game, which is a box. The box has two degrees of freedom, controlled with "Head{Up,Down}" and "Roll{Left,Right}" gestures, forming a natural mapping. Moreover the game's puzzle nature doesn't expect successive fast movements, qualifying the head gestures modality as suitable for the interaction. In Fig. 12 a user is shown using the system.

Following the employment of the application, it is concluded that the utilized gestured recognition system provided the ability to fully control it. However, it is noted that further

 ISBN: 978-1-61208-356-8

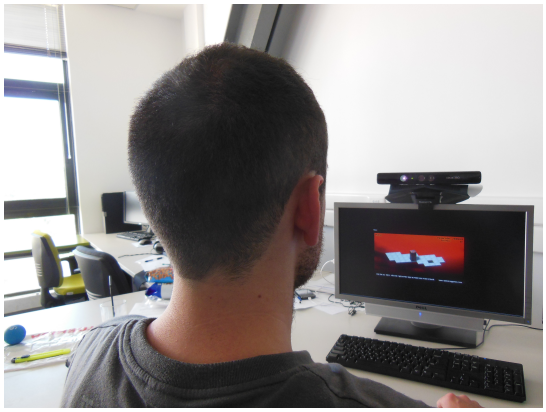

Fig. 12. The game application is shown in the left screen. Right screen shows the output window from the pose estimation.

work is required in order to use head gestures as the sole method of user interaction with an application.

VI. Conclusion

A method for simple gesture detection and recognition that is based on 3D head tracking was presented along with its evaluation. The proposed work explores the potential of recognizing robustly primitive head motions as a means for natural human computer interaction.

In this context, the proposed method was evaluated indicating that recognition provides sufficiently reliable recognition rates for employment in human-computer dialogs. The proposed method has been, also, utilized beyond the context of such a dialog. We concluded that the detection and parsing of gestures from continuous head motion of the proposed method, is a property that sets the foundations for the generic use of these gestures in human computer interaction. In that respect, investigation of usability issues is the topic of future work.

The evaluation of the proposed method indicated that advances in head tracking accuracy are the most important topic of future work, as recognition failures are mainly due to shortcomings of the underlying head pose estimation technology. Based on this finding, we conclude that the proposed technology is suitable to be applied at the spatial range of operation of the corresponding head trackers. In turn, this range is determined by the accuracy of the utilized depth sensor, which is in the order of $.5\,m$ to $1.5\,m$. As a consequence, in the context of an intelligent environment one could envisage utilization of head gesture at special locations, such as when the user is situated at location related to a particular activity.

In the evaluation, a study of user behavior in terms of gesture execution time and steepness of head rotation was performed. We have observed that downward head rotations are, usually, performed in smaller amount of rotation, with reasons that can be probably traced to the head rotation ergonomy and anatomy. Given a constant, with respect to axis of rotation, accuracy of head pose estimation this indicates the increased vulnerability of downward gestures, which can be of interest in the design of pertinent applications. Alternatively, the purposeful placement of the imaging sensor may be considered, so as to better image corresponding head motions.

Acknowledgment

This work has been supported by the FORTH-ICS RTD Programme “Ambient Intelligence and Smart Environments”.

References

[1] P. Padeleris, X. Zabulis, and A. A. Argyros, “Head pose estimation on depth data based on particle swarm optimization,” in CVPR Workshops, 2012, pp. 42–49.

[2] Q. Cai, D. Gallup, C. Zhang, and Z. Zhang, “3d deformable face tracking with a commodity depth camera,” in 11th European Conference on Computer Vision: Part III. Springer-Verlag, 2010, pp. 229–242.

[3] L. Morency, A. Rahimi, N. Checka, and T. Darrell, “Fast stereo-based head tracking for interactive environments,” in Fifth IEEE International Conference on Automatic Face and Gesture Recognition, May 2002, pp. 390–395.

[4] A. Riener and A. Sippl, “Head-pose-based attention recognition on large public displays,” Computer Graphics and Applications, IEEE, vol. 34, no. 1, 2014, pp. 32–41.

[5] Z. Zhang, “Microsoft kinect sensor and its effect,” MultiMedia, IEEE, vol. 19, no. 2, Feb 2012, pp. 4–10.

[6] Z. Yu, Z. Yu, H. Aoyama, M. Ozeki, and Y. Nakamura, “Capture, recognition, and visualization of human semantic interactions in meetings,” in IEEE International Conference on Pervasive Computing and Communications. IEEE, 2010, pp. 107–115.

[7] J. W. Davis and S. Vaks, “A perceptual user interface for recognizing head gesture acknowledgements,” in Workshop on perceptive user interfaces. ACM, 2001, pp. 1–7.

[8] S. Chu and J. Tanaka, “Head nod and shake gesture interface for a self-portrait camera,” in International Conference on Advances in Computer-Human Interactions, 2012, pp. 112–117.

[9] D. Kelly, J. Reilly Delannoy, J. McDonald, and C. Markham, “Automatic recognition of head movement gestures in sign language sentences,” in China Ireland International Conference on Information and Communication Technology. Dept. of Computer Science, National University of Ireland, Maynooth, 2009, pp. 142–145.

[10] H. Wu, T. Shioyama, and H. Kobayashi, “Spotting recognition of head gestures from color image series,” vol. 1, 1998, pp. 83–85.

[11] H. Wei, P. Scanlon, Y. Li, D. S. Monaghan, and N. E. O’Connor, “Real-time head nod and shake detection for continuous human affect recognition,” in International Workshop on Image Analysis for Multimedia Interactive Services. IEEE, 2013, pp. 1–4.

[12] R. Li, C. Taskiran, and M. Danielsen, “Head pose tracking and gesture detection using block motion vectors on mobile devices,” in International conference on mobile technology, applications, and systems. ACM, 2007, pp. 572–575.

[13] A. Agrawal, R. Raj, and S. Porwal, “Vision-based multimodal human-computer interaction using hand and head gestures,” in IEEE Conference on Information & Communication Technologies. IEEE, 2013, pp. 1288–1292.

[14] J. Gast et al., “Did I get it right: Head gestures analysis for human-machine interactions,” in Human-Computer Interaction. Novel Interaction Methods and Techniques. Springer, 2009, pp. 170–177.

[15] N. Wohler et al., “A calibration-free head gesture recognition system with online capability,” in International Conference on Pattern Recognition. IEEE, 2010, pp. 3814–3817.

[16] L.-P. Morency and T. Darrell, “Head gesture recognition in intelligent interfaces: the role of context in improving recognition,” in International conference on Intelligent user interfaces. ACM, 2006, pp. 32–38.

[17] K. Biswas and S. K. Basu, “Gesture recognition using Microsoft Kinect®,” in Automation, Robotics and Applications (ICARA), 2011 5th International Conference on. IEEE, 2011, pp. 100–103.

[18] Y. Georgalis, D. Grammenos, and C. Stephanidis, “Middleware for ambient intelligence environments: Reviewing requirements and communication technologies.” in HCI (6), ser. Lecture Notes in Computer Science, C. Stephanidis, Ed., vol. 5615. Springer, 2009, pp. 168–177.

[19] Z. Zhang, “Flexible camera calibration by viewing a plane from unknown orientations,” in IEEE International Conference on Computer Vision, vol. 1. IEEE, 1999, pp. 666–673.

[20] “Bloxorz puzzle game,” URL: http://www.miniclip.com/games/bloxorz [retrieved: June, 2014].

 ISBN: 978-1-61208-356-8