Head-to-Body-Pose Classification in No-Pose VR Tracking Systems

Tobias Feigl*
Programming Systems Group
Friedrich-Alexander University
Erlangen-Nürnberg (FAU)

Christopher Mutschler†
Machine Learning and
Information Fusion Group
Fraunhofer IIS

Michael Philippsen
Programming Systems Group
Friedrich-Alexander University
Erlangen-Nürnberg (FAU)

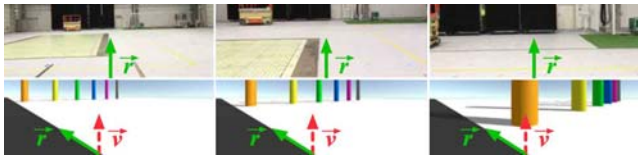


Figure 1: Real (top) and virtual world (bottom).

ABSTRACT

Pose tracking does not yet reliably work in large-scale interactive multi-user VR. Our novel head orientation estimation combines a single inertial sensor located at the user's head with inaccurate positional tracking. We exploit that users tend to walk in their viewing direction and classify head and body motion to estimate heading drift. This enables low-cost long-time stable head orientation. We evaluate our method and show that we sustain immersion.

Keywords: VR, head tracking, inertial sensor fusion, immersion, large-scale, machine learning, motion sickness.

Index Terms: Computing methodologies [Supervised learning by classification] Human-centered computing [Virtual reality]

1 INTRODUCTION

VR drives innovation in applications for theme parks, museums, architecture, training, simulation, etc. They all can benefit from multi-user interaction, from areas beyond 20 $m \times 20$ m, and from natural movement without motion sickness. However, SLAM-based pose estimation only works reliably under restricted conditions (small rooms, static scenes/no moving objects, homogeneous lightning) [2].

Low-cost no-pose tracking systems work on larger tracking areas and for more users, but as they only provide positions we need to estimate the head orientation separately. Low-cost Head-Mounted Display (HMD) units with their inertial measurement units (IMU) yield inaccurate estimates, as (i) magnetometers are unreliable in many indoor environments and provide a wrong absolute head orientation [6], (ii) dead reckoning based on relative IMU data leads to drift and (after a while) to a wrong estimate [3], and (iii) state-of-the-art filters fail to provide reliable motion direction estimates as they require either accurate sensor models or military-grade sensors [1]. While Human Activity Recognition can detect activities such as standing, walking, and sleeping with IMUs placed on body parts, they do not estimate the absolute head orientation [4, 6].

A wrong orientation estimation results in a mismatch of the real world and the VR display. The upper row in Fig. 1 shows the real world view of a user who walks straight ahead with his/her head oriented ($\vec{r}$) in the direction of the movement $\vec{m}$, i.e., $\vec{m}$=$\vec{r}$. However, under drift (the bottom row shows a 45° sensor drift) the same movement leads to a displacement from right to left as a wrong head orientation ($\vec{v}$) is used to render the VR images. For the user the direction of the movement does not fit to the VR view. The sensor drift inevitably causes a wrong head orientation. If there is almost no drift the user's real head orientation $\vec{r}$ is close to his/her virtual head orientation $\vec{v}$; movements feel natural as the VR image is rendered with the correct orientation. Otherwise the user moves in the direction of $\vec{m}$ and recognizes an unnatural translation of the rendered image towards $\vec{v}$.

Our key idea is to combine inaccurate positional tracking (error ± 20 cm) with relative IMU data to achieve a long-time stable head orientation while the user is walking naturally and rotating his/her head. Under the assumption that humans mostly walk in their viewing direction we extract features from sensor signals, classify the relation between real movement direction and real head orientation, and combine this with absolute tracking information. This yields an estimation of the absolute head orientation that we use to adapt the offset into a user's virtual view.

2 LONG-TERM STABLE HEAD ORIENTATION ESTIMATION

Assume that a fine-grained absolute position tracking (both with respect to coordinates and timestamps) of users is available. Then we can record a user's positions over time and extract a trajectory vector. With the assumption that users look forward in forward movements ($\vec{m}$=$\vec{r}$) a VR system *can* then deduce $\vec{r}$ from the trajectory vector, adjust $\vec{v}$, and thus eliminate the drift that causes motion sickness.

Instead of directly estimating the head orientation from IMU sensor data, we use machine learning to detect $\vec{m}$=$\vec{r}$-moments, because then we know how to adjust the drift (we set $\vec{v}$=$\vec{m}$).

Fig. 2 outlines our processing pipeline. First, we preprocess the raw IMU sensor signals (i.e., accelerometer and gyroscope) with digital filters. With supervised machine learning we then classify the movement along various head-to-body orientations ω. In a training step we extract features for known ranges of ω on labeled training samples to train the classifier. At runtime, the trained classifier processes the features of preprocessed unknown signals and returns the best-fitting ω-range class and its classification confidence. In $\vec{m}$=$\vec{r}$-moments, i.e., ω=0, we determine the head orientation drift and use a linear interpolation to reduce the drift (i.e., to adjust $\vec{v}$ to $\vec{r}$) in an immersive way so that users do not notice it.

Signal Processing. Raw accelerometer (acc) and gyroscope (gyr) data from a low-cost IMU sensor needs a preprocessing before we can extract reliable features. Our low-cost acc tracks gravity and acceleration at 200 Hz up to ± 16 g. Our gyr tracks the angular velocity at 200 Hz up to ± 2000 $^\circ/s$. To preprocess the IMU data we use 6 sliding windows for the acc- and gyr-streams (3 axes each). We smooth the raw acc^{raw} and gyr^{raw} data to eliminate noise with an Savitzky-Golay-filter (SG) (window length 25 and a polynomial order 3) into acc^{SG} and gyr^{SG}. To describe the user's

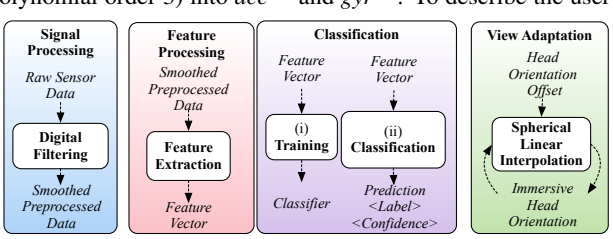


Figure 2: Head orientation estimation processing pipeline.

*Also with Fraunhofer IIS, tobias.feigl@fau.de
†Also with Friedrich-Alexander University Erlangen-Nürnberg (FAU)

2018 IEEE Conference on Virtual Reality and 3D User Interfaces
18-22 March, Reutlingen, Germany
978-1-5386-3365-6/18/$31.00 ©2018 IEEE

head-to-body-pose accurately, we also split acc^{raw} into its gravity (describing the pose) and linear (describing the motion) components acc^{raw}_{grav} and acc^{raw}_{lin} with low-/high-pass (LP/HP) IIR-filters (HP/LP cut-off frequencies of 5/40 Hz, window length 12,000). This yields 4 streams of preprocessed data (with 3 axes each) that we use to extract the features: gyr^{SG} (rotations), acc^{SG} (pose and motion), acc^{IIR}_{grav} (pose), and acc^{IIR}_{lin} (motion).

Feature Processing. We turn the 4·3 preprocessed signal streams into feature streams that capture the intuition and that a classifier can make use of. The following small set of features uniquely describes a class (ω-angle): (1) mean μ (starting spot of the current gait cycle), (2) standard deviation *std* (intensity of signal fluctuations), (3) correlation $corr_{yz}$ between the Y- and Z-axes (variant head orientation), and (4) Principal Component Analysis *pca* (highest variance/information density). From the 4·3 signal streams we thus extract a total of 4·3·3=36 streams (μ, *std*, and *pca*), plus 4 streams for $corr_{yz}$. From a pre-examination we know that the signals of one full gait cycle suffice to classify the ω-angle and that the necessary feature computations keep the required CPU resources low. Adding more only slows down the feature extraction. When users walk in a typical VR setup, one cycle fits into about 1 *s*, i.e., into 200 (filtered) signal values.

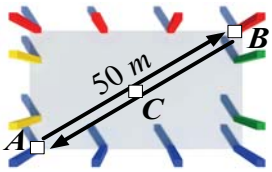
Classification. We studied 3 classifiers: a Classification & Regression Tree, a k-Nearest Neighbor, and a Support Vector Machine (SVM). The SVM (with a cubic kernel function) provided the most confident and reliable results. Since we need a multi-class classification we use a One-vs-All SVM.

View Adaptation. To adapt $\vec{v}$ so that it feels natural to the user we linearly interpolate and gradually apply the estimated orientation error to the current view orientation $\vec{v}$. We interpolate from the start orientation $\vec{v}$ to the end orientation by a small and immersive portion of the drift ω_{imm} between consecutive frames and subtract it from the error. Pre-tests taught us for a better immersion to only adapt the view in the direction of the current turn, i.e., to exaggerate turns and to only adapt while the user performs a yaw rotation.

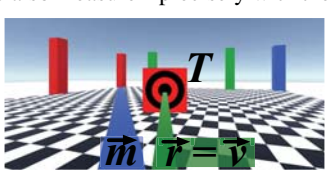
3 Evaluation

On a tracking space of about 40 $m \times 35$ m, all experiments use a Samsung Galaxy Note 4 smartphone that has a 6 *DoF* IMU sensor from InvenSense (MPU-6500) attached to a Samsung Gear VR HMD (SM-R320). In addition to the IMU data, for the training and the evaluation of the classifiers we also have highly precise head orientation measurements to label the IMU data from an optical laser-based Nikon iGPS system tracking system with an accuracy $<$10 mm at 20 Hz. We use a head-mounted apparatus that carries two locatable objects at a distance of 50 cm to calculate the absolute head orientation. We always introduced the participants to the setup and to the purpose of the measurements beforehand.

Classification. To evaluate the feature selection from Sec. 2 and the classifier performance we collect data with a group of 34 subjects (avg. age 23.16 [18, 36] years; avg. height 1.74 [1.49 to 1.81] m; 20 ♂, 14 ♀; nobody disabled or handicapped) and let them walk 10 times naturally on a 50 m path ($\vec{AB}$ and $\vec{BA}$, see Fig. 3(a)) within a VR environment (rendered at const. 60 $frames/s$, walking speed avg.: 0.87 m/s, min.: 0.58 m/s, max.: 1.19 m/s, SD: 0.18 m/s). A target T and 2 colored lines, see Fig. 3(b), helped them. While we collect the IMU data from the HMD we also measure $\vec{r}$ precisely with the



(a) Users walk between A and B or are positioned at C (center).



(b) Blue = walking path, green = view direction.

Figure 3: Top-view (a) and first-person-view (b) of our VR scenario.

Table 1: Success rate of a 10-fold cross-validation in % for the SVM.

Features				Streams: acc^{SG}	gyr^{SG}	acc^{IIR}_{grav}	acc^{IIR}_{lin}	acc^{SG}	gyr^{SG} acc^{SG}	acc^{IIR}_{grav} gyr^{SG} acc^{SG}	acc^{IIR}_{lin} gyr^{SG} acc^{SG}	acc^{IIR}_{lin} acc^{IIR}_{grav} gyr^{SG} acc^{SG}
μ	-	-	-	66	47	68	34	61	61	62	62	
-	*std*	-	-	63	75	64	69	57	56	58	59	
-	-	$corr_{yz}$	-	14	14	6	9	65	73	69	68	
-	-	-	*pca*	64	12	65	11	64	68	70	71	
μ	*std*	$corr_{yz}$	*pca*	73	42	72	41	81	84	83	**86**	

iGPS system for the labeling. Users were asked to keep their heads at a fixed ω-angle $[-45°; -30°; -15°; 0°; +15°; +30°; +45°]$. To obtain a close to natural head pose while walking, we did not enforce rigid head pitch and roll. We recorded about 8 h of movement data.

Table 1 shows the results of a 10-fold cross-validation on different combinations of signal and feature streams. The more features and input streams are used the better the result gets. The complete feature set on all streams yields a correct classification in 86% of the cases.

When the participants are walking, the classifier easily separates $\vec{m}$=$\vec{r}$-moments from others, i.e., the ω=0°-class. Assuming that humans tolerate a drift of 20% without noticing it [5] we even achieve a correct classification in 95% of the cases. To process all the 40 features of all streams, it takes 259 µs (per update).

View Adaptation. The user's rotation in the VR-display can be exaggerated or reduced by ±30% without the user noticing it [5]. Thus, for dynamic scenarios (with moving and turning users) we pick ω_{imm}=0.3·d to adapt the drift while the user is turning by d°. For static scenarios (without moving and turning), we use a group of 52 (other) subjects (avg. age 25.82 [19, 41] years; avg. height 1.72 [1.51 to 1.87] m; 34 ♂, 18 ♀; nobody disabled or handicapped). Users stand stationary at C, see Fig. 3. We asked them to announce significant jitter and/or jumps of the camera view as early as they notice them while we iteratively reduce the heading error by ω_{imm} + 0.1. We found that we can always (even in static scenarios) adapt the view by picking an unnoticeable $\omega_{imm} \leq 0.9°/s$. In movements we can set the upper bound below ω_{imm}=4.8 °/s.

4 Conclusion

The paper shows how to estimate absolute head-to-body orientations from inaccurate positions and noisy inertial sensors mounted at the head. With a set of features extracted from filtered sensor data (after some training with labeled data) an SVM classifier can reliably detect moments in which users walk with their heads facing forward and in which we can thus determine the accumulated drift. VR users do not notice our maximal error of ±15 ° when the estimation is used to immersively correct the drift of the displayed VR images. We successfully use the proposed method in VR, e.g., in museums.

Acknowledgments

This work was supported by the Bavarian Ministry for Economic Affairs, Infrastructure, Transport and Technology and the Embedded Systems Initiative (ESI).

References

[1] K. Al Nuaimi and H. Kamel. A survey of indoor positioning systems and algorithms. In *Proc. Intl. Conf. Innovations in Information Techno.*, pp. 185–190. Abu Dhabi, UAE, 2011.

[2] M. Buerli and S. Misslinger. Introducing ARKit - Augmented Reality for iOS. WWDC - Session 602, June 2017.

[3] E. Foxlin. Pedestrian tracking with shoe-mounted inertial sensors. *IEEE Compu. Graphics and Appli.*, 25(6):38–46, 2005.

[4] A. Steed and S. Julier. Behaviour-aware sensor fusion: Continuously inferring the alignment of coordinate systems from user behaviour. In *IEEE Intl. Symp. ISMAR*, pp. 163–172. Adelaide, Australia, 2013.

[5] F. Steinicke, Y. Visell, J. Campos, and A. Lécuyer (Eds.). *Human walking in virtual environments: perception, technology, and applications*. Springer, New York, 2013.

[6] J. Windau and L. Itti. Walking compass with head-mounted IMU sensor. In *Proc. 2016 IEEE Intl. Conf. Robotics and Automation*, pp. 5542–5547. Stockholm, Sweden, 2016.