Article

Hybrid Orientation Based Human Limbs Motion Tracking Method

Grzegorz Glonek [1,†,*] and Adam Wojciechowski [1,†]

[1] Institute of Information Technology, Faculty of Technical Physics, Information Technology and Applied Mathematics, Lodz University of Technology, Skorupki 6/8, 90-924 Lodz, Poland

* Correspondence: grzegorz@glonek.net.pl; Tel.: +48-660-758-666

† These authors contributed equally to this work.

Academic Editor: name

Version March 19, 2017 submitted to Sensors; Typeset by LaTeX using class file mdpi.cls

Abstract: One of the key technologies that lay behind human-machine interaction is limbs motion tracking. To make the interaction effective, the motion tracking must be able to estimate precise and unambiguous position of each tracked human body part and joint. In recent years, motion tracking became very popular and broadly available for home user because of easy access to cheap and robust tracking devices. The paper presents the novel approach of using and data fusion for two of such cheap tracking devices: Microsoft Kinect v.1 and inertial measurement units (IMU). The detailed review of their working characteristics leads to the description of the method of the fusion of data from both devices that compensates their imprecisions. The paper also describes the series of performed experiments that verified the method accuracy. This novel approach allowed improving joints positioning precision up to 18% in terms of the most relevant methods described in the literature.

Keywords: data fusion;Microsoft Kinect;IMU;motion tracking

0. How to Use this Template

The template details the sections that can be used in a manuscript. Note that the order of article sections may differ from the requirements of the journal (e.g. for the positioning of the Materials and Methods section). Please check the instructions for authors page of the journal to verify the correct order. For any questions, please contact the editorial office of the journal or support@mdpi.com. For LaTeX related questions please contact Janine Daum at latex-support@mdpi.com.

1. Introduction

Human limbs motion tracking can be defined as a process of unambiguous limb's joints spatial positioning process devoid of significant delays. Nowadays, it might be exploited within several areas, such as entertainment issues, virtual environments interaction or medical rehabilitation treatment. However system precision determines its possible application.

Since 1973 when Gunnar Johanson invented his motion capture system [1], such techniques were available mainly for professionals. However, since 2010, when Microsoft Company released Kinect v.1 controller, motion capture became available to almost everyone. Microsoft Kinect controller is an exemplary implementation of the optical marker-less motion capture system. Though it was designed for and was mainly used in the field of home physical activity games, Kinect became popular subject for many researchers, whose goal was to find other, more advanced scenarios where this controller might be applied [2,3].

Easy access to the motion capture has been additionally supplemented and magnified by the fact that inertial devices such as gyroscopes and accelerometers (IMU – inertial measurement units) have become an integral and mandatory part of current electronic devices. These units are built-in every smartphone device and they are used in various applications like pedometers [4,5], screen rotators [6] or digital camera orientation detectors.

Inertial measurement units (IMU) are also used as a measuring devices in non-optical inertial human limbs motion tracking systems i.e. XSense [7]. Despite IMU, used in products available for home users, are improving continuously, their measurements precision and work limitations can be recognized as blockers for motion capture scenarios that require high accuracy of joints positioning.

The broad and easy access to mentioned devices became a trigger to work on these controllers data fusion methods that increase the accuracy of joints positioning for many researchers around the world. Bo et al. [8] focused on knee joint angle estimation. He proposed accelerometer and gyroscope data fusion, by linear Kalman filter, to calculate the angle and align it to the Kinect's estimation. Destelle et al. [9] have proposed method based mainly on IMU devices. In this method, the shape of skeleton model has been estimated on inertial data only, and sizes of each skeleton bone were set based on Kinect estimation. Also the skeleton placement was defined by selected Kinect's skeleton joint position. The data from accelerometer, gyroscope and magnetometer were fused by Madgwick's filter [10]. Kalkbrenner et al. [11] proposed to combine together Madgwick's filter with linear Kalman filter. In their approach they fused accelerometer and gyroscope data by Madgwick's filter to estimate selected bones orientation. Basing on this information and the bone length, estimated by Kinect, joints positions were calculated. In the last step joints positions estimated by Kinect were fused with linear Kalman filter with joints positions estimated with IMUs orientations. In the method proposed by Feng and Murray-Smith [12] accelerometer data has been fused with Kinect measurement by linear Kalman filter, modified by authors to work with frequency unaligned signals. Tannous et al. [13] proposed extended Kalman filter to fuse bones orientations estimated by Kinect and by IMU devices. It is worth mentioning that the majority of data fusion methods, described in the literature, are based on different variants of Kalman filter (linear, extended or unscented variant). It is also noticeable that authors of these methods are selectively taking into consideration characteristics and limitations of measurement devices.

nigdzie nie pojawiają się wartości dokładności. Może warto aby się pojawiły to wtedy od razu pokaze się przyrost dokładności

This paper presents the novel approach to Microsoft Kinect and IMU data fusion that systemically makes up for limitations of both measurement devices and compensates their imperfections. Orientation based, human limbs' joints positions tracking method outperforms state-of-art hybrid systems accuracy by 18%.

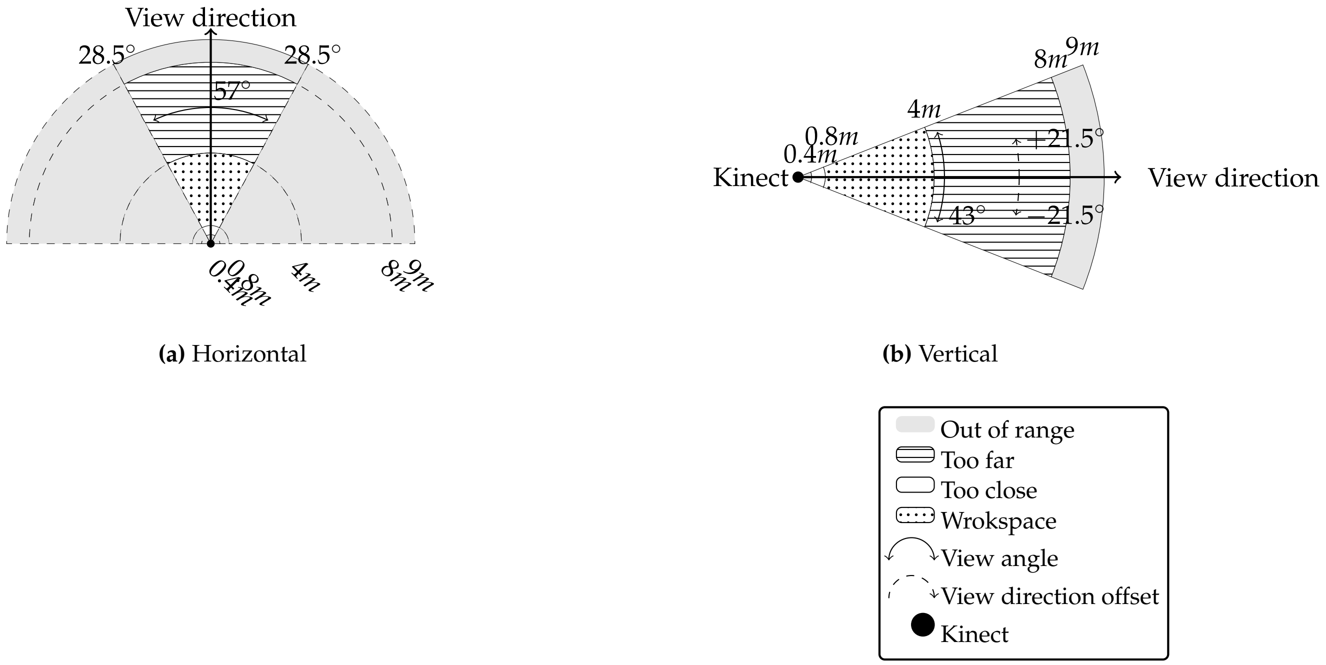
2. Materials and Methods

2.1. Microsoft Kinect controller characteristics

Microsoft Kinect controller might be described as a RGB-D camera. The first version of this device, originally designed for Xbox 360 video game console, was built from two CMOS cameras and integrated infrared (IR) projector. One of these CMOS was responsible for a RGB signal and the second one was calibrated to record IR beam's view. Considering limbs motion tracking and body gesture recognition, the most crucial part of Microsoft Kinect controller is the chip made by the company PrimeSense. All recognition algorithms, that stand behind Kinect's motion tracking, were implemented as a firmware of this processing unit. The Microsoft Company has not revealed detailed description of tracking algorithms and key characteristics, so their retrieval became a subject for scientists and hobbyist [14–16]. Good understanding of the Kinect controller constraints seems to be indispensable to create an accurate method that fuse its data with data collected from any other source. Official device specification stands that the operation range of Kinect is between $0.8m$ and $4m$ with the field of view 57° horizontally (Figure 1a) and 43° vertically (Figure 1b). However, the specification doesn't reveal the possible accuracy of nonlinearity in the working area. Researchers [17] and hobbyists [18] reported that operation range is different between device series.

Moreover, conducted experiments showed that there are distance measurement nonlinearities within the Kinect's working area. Comparison of fluently changing Kinect-limb's joints distances, collected simultaneously from Microsoft Kinect controller and Vicon motion capture system, showed that Kinect had tendencies to underestimate the distance in the close range and to overestimate it in the far range. Collected values allowed to define distance estimation error model in a form of 3rd order polynomial as it is presented in equation (1).

$$f(z) = 0.02z^3 - 0.11z^2 + 0.27z - 0.25 \tag{1}$$



(a) Horizontal

(b) Vertical

Figure 1. Microsoft Kinect v.1 operation range diagram in horizontal (a) and vertical (b) directions

Figure 2 presents the plot of function defined with equation (1). Analysis of this chart shows that Kinect's distance estimation is optimal when the user stands about $2m$ from the device ($2m$ – $2.3m$). Outside this distance range, where Kinect works with noticeable error, systematic sensor depth correction should be taken into consideration.

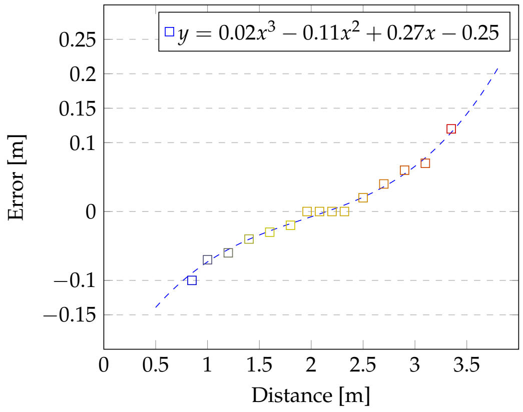


Figure 2. Microsoft Kinect depth measurement accuracy as a function of Kinect-joints distance.

The reason of such inaccuracy and its tendencies is probably caused by the algorithm used by Kinect controller. Though Microsoft hasn't revealed any precise technical description of their method for measuring distance between camera and objects at scene, original patent forms, that PrimeSense owned [19–21], combined with independent research results [22], provide some general overview of how this recognition algorithm works. It is known that analysed scene, in front of the Kinect controller, is enlighten with IR light dots structured pattern, which is skew symmetric, so the Kinect can work up or upside down. Then Kinect analysis the IR pattern's dots distortion, and based on that estimates the subject distance. For the depth map reconstruction, Kinect uses two techniques in parallel: dots blurriness analysis [23] and stereo-vision based on a single IR camera and a projector [24]. Achieved depth map is a foundation of a human skeleton estimation algorithm. Depth retrieved human body parts estimations are subsequently classified into body poses by means of machine learning approach: random decision forest algorithms as well as on object detection algorithms, such as the one prosed by Viola-Jones [25,26]. As the only source of data, for that estimation process, is IR camera (RGB camera is not used at all), the observed scene must be free of any external source of IR light. This requirement restricts the usage of Kinect in outdoor scenarios. A need for scene isolation, from any IR sources other than Kinect itself, is the reason why Microsoft doesn't recommend using two or more controllers simultaneously. Nevertheless scientists worked out and published methods on combining signals from several Kinect devices [27–29]. It is noticeable that information gathered from the Kinect is uncompleted, due to the lack of information about joints rotation along the bone. This is the result of skeleton model design, where each one of 20 tracked joints is described as a single point. The last major flaw of Microsoft Kinect controller are occlusions, which occur when a part of user's body is covered by another object or is hidden behind any other body part (self-occlusion). The occlusion by an external object seems to be intuitive and doesn't require any additional explanation, but self-occlusions are connected with Kinect's sensitivity to user's rotation to the camera (α angle in Figure 3) - Microsoft recommends operating Kinect in, not precisely defined, the face off pose. Self-performed experiments allowed observing α angle changes while user rotated in front of the camera. Assuming that P_{Sh_L} is the position of left shoulder and P_{Sh_R} is the position of right shoulder, both defined in a Kinect controller coordinating system (limited to X and Z axes in Figure 3), then α can be calculated according to equation (2).

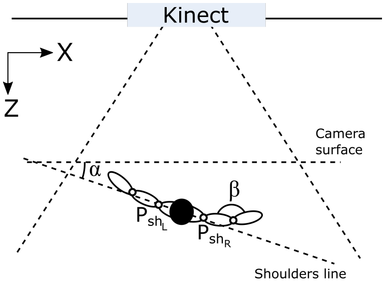


Figure 3. Rotation angle α between user and Kinect

$$\alpha = \begin{cases} atan(\frac{|p^K_{Sh_R,Z} - p^K_{Sh_L,Z}|}{|p^K_{Sh_R,X} - p^K_{Sh_L,X}|} & , |p^K_{Sh_R,X} - p^K_{Sh_L,X}| \neq 0 \\ \frac{\Pi}{2} & , |p^K_{Sh_R,X} - p^K_{Sh_L,X}| = 0 \end{cases} \tag{2}$$

There are two possible strategies that Kinect uses when occlusion happens. Either the device tries to estimate the location of the covered joint or, when it is not able to do even rough estimation of the joint position, Kinect stops tracking such joint. The results of a self-performed experiments show that occlusion occurs when the user is rotated by more than 50° ($\alpha > 50°$).

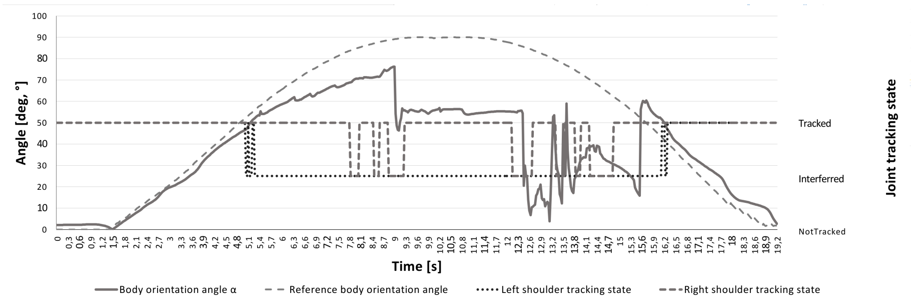


Figure 4. Shoulders joints tracking state in relation to body rotation angle α

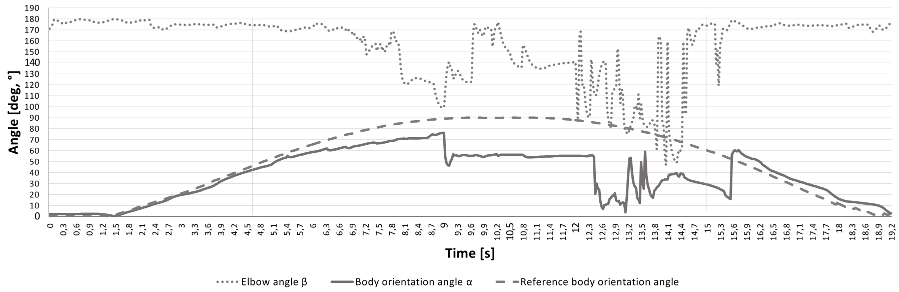


Figure 5. Elbow angle β estimation in relation to body rotation angle α

Charts presented in Figure 4 and Figure 5 show measured joints tracking states and elbow angle (angle β in Figure 3) changes during rotation respectively. In Figure 4 we can see that tracking the states of both shoulders joints fluctuate between *Tracked* and *Interferred* states when subject was rotated more than 50° in relation to Kinect's camera. It is worth mentioning that even well visible right shoulder joint lost tracking state during that rotation. In Figure 5 we can see that estimation of right elbow angle β. It was unstable for considerable body rotation ($\alpha > 50°$) though the hand was fully visible all the time.

2.2. *IMU characteristics*

IMU devices have been in professional usage from decades. Within scope of interest there are two types of sensors that measure inertial forces affecting them: accelerometers and gyroscopes. All experiments, presented in this paper, were performed with the usage of IvenSense MPU-6050 module which integrates these both types of inertial devices and thermometer. Even though accelerometer and gyroscope were integrated on a single PCB, their data must be processed individually, as they are affected by the different set of external noises with different frequencies that must be filtered out to make them usable.

An accelerometer is an inertial sensor that measures linear forces, affecting it along device coordinating system axes. Usually, the measurements are defined in relation to gravity force ($1g \approx 9.81 m/s^2$) so it is possible to estimate its linear acceleration during the motion. When the device rests the theoretically correct measurement should be $1g$ in upward direction and $0g$ in two other directions. However, due to the external noises (mainly high frequency) resting device measurements are oscillating around theoretical values. Analysis of these oscillations, during calibration, lets design low pass filter that would be able to filter out at least major part of these high frequency noises. Accelerometers are also sensitive to operating temperature changes and its influence, on sensor's accuracy, was the subject of some researches [30,31]. The reason of temperature sensitivity is due to the architecture of the sensor which is built of several capacitors hence the operating temperature has influence on their capacity. During self conducted experiment the measurement of the g-force was observed when the resting device was heated and then cooled down multiple times in the temperature range $10°C - 50°C$. The result of this experiment is presented in the Figure 6. According to the MPU-6050 specification and results of this experiment, the neutral operating temperature for considered module was $25°C$. Due to the fact that the operating temperature of IMU device, placed on a human body, rises and stabilizes at approx. $30°C$, some sort of compensation is required there.

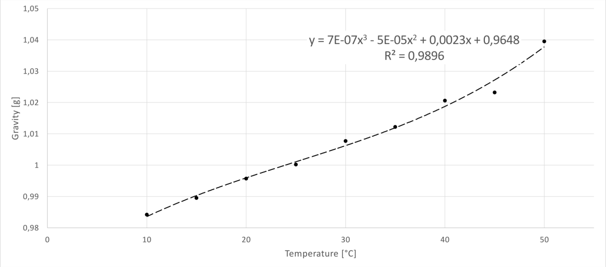


Figure 6. Accelerometer gravity measurements in temperature range $10°C - 50°C$

A gyroscope – the second type of inertial sensors – measures its angular velocity in deg/s units. If the sensor is at the resting state, all measurements should equal 0 for each of 3 axes. However device suffers from long-term bias that should be limited (ideally – removed). Analysed noise has low frequency characteristics, thus appropriate high pass filter, limiting its influence, should be applied. As MEMS based gyroscope is also built from multiple capacitors, their bias is influenced by the temperature as well. As the distortions have low frequency nature, there was no difference observed for high-pass filtered signals over the temperature range. It meant that no additional compensation is required here. The problem with incomplete information is also valid for IMU devices. To calculate accurate orientation of the IMU module, data from accelerometer and gyroscope (orientation around corresponding axes) should be fused together. The accelerometer allows to calculate orientation around two axes, even though it measures forces along all 3 orthogonal directions. The orientation (or the rotation) around the gravity vector is unmeasurable for such device. Theoretically, gyroscope should retrieve orientation measures around all 3 directions, however, even filtered signal, contains some error. Its numerical integration over time results in significant data drift that makes such measurements useless. An example of data drift is presented in Figure 7. Despite the sensor remaining motionless during the experiment, the estimations based on the numerically integrated data shows that the sensor rotated about 60° within 2 minutes.

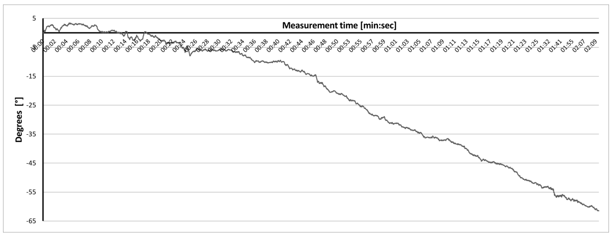


Figure 7. Gyroscope drift for non-moving device

2.3. Hybrid, orientation based, human limbs motion tracking method

The authors' novel hybrid, orientation based, human limbs motion tracking method, presented in this paper, is based on the continuous linear fusion of skeleton bones orientations (contrary to state-of-art skeleton joints positions fusion) with respect to the current motion context. It takes into consideration controllers' reliability and compensates measurement devices imperfections described in previous chapters. The overall data fusion, implemented in this method, has been split into several phases. Some of them are performed once, while the others are repeated while motion tracking. The subsequent phases are listed below:

1. IMU initialization (performed once).
2. IMU data fusion filters initialization (performed once).
3. IMU and Kinect controller data noise correction.
4. IMU and Kinect synchronization (performed once).
5. IMU and Kinect data fusion.
6. Joints final position estimation.

gdzieś przy hierarchii stawó przydałoby się odwołanie do rysunku szkieletu

The main goal of described method is to estimate selected human skeleton joint (j) position ($P^F_{j,t} = [p^F_{j,x}, p^F_{j,y}, p^F_{j,z}]_t$) within reference of a three dimensional coordinating system, in a particular moment of time t. Estimation is based on the joint skeleton attached accelerometer measurements ($A = [a_x, a_y, a_z]$), gyroscope measurements ($G = [g_x, g_y, g_z]$), current operational temperature (T) of IMU module and position of this joint (j) measured by Kinect controller ($P^K_{j,t} = [p^K_{j,x}, p^K_{j,y}, p^K_{j,z}]_t$) as well as the position of the parent joint ($j-1$) in the Kinect's hierarchical skeleton model ($P^K_{j-1,t} = [p^K_{j,x}, p^K_{j,y}, p^K_{j,z}]_t$). The estimation is performed with some time interval (Δt) and is defined as function presented in equation 3. The function presented in equation 3 relates to each IMU module individually and will need to be executed as many times as many IMU modules are used.

$$f(A, G, T, P^K_{j-1,t}, P^K_{j,t}, \Delta t) \implies P^F_{j,t} \tag{3}$$

2.3.1. IMU initialization

The goal of IMU initialization procedure is to calculate the data (G, A) correction factor matrix ($cor = [cor_A \quad cor_G]^T = [[c_a x, c_a y, c_a z] \quad [c_g x, c_g y, c_g z]]^T$). During this phase, IMU modules must lay down without any motion in face-up position, as this is the position that we know exactly what data to expect ($A_0 = [0, 0, 1]$ for accelerometer and $G_0 = [0, 0, 0]$ for gyroscope). In fact exact reference values are impossible to achieve because of the noise affecting IMU modules, thus the initialization algorithm should work iteratively (s - iteration index) as long as the any element of average measurement error matrix ($[\bar{A} \quad \bar{G}]^T = [[\bar{a_x}, \bar{a_y}, \bar{a_z}]^T \quad [\bar{g_x}, \bar{g_y}, \bar{g_z}]^T]^T$) is greater than the element with the same index in error threshold matrix ($[A_{th} \quad G_{th}]^T = [[a_{x,th}, a_{y,th}, a_{z,th}]^T \quad [g_{x,th}, g_{y,th}, g_{z,th}]^T]^T$). As a result, the data correction factors matrix (cor_s) was calculated to allow gathering the measurements with acceptable accuracy. If we define ideal measurements as ($[A_0 \quad G_0]$), ($[A \quad G]_{s,i}$) series on n subsequent sensor's indications, for s-th iteration, inertial sensor's average measurement error is defined by equation 4 and data correction factor for single measurement is defined by equation 5. While cor_s has been calculated, its values are used to correct the influence of static device measurement error in raw accelerometer and gyroscope measurements.

zbyt długie zdanie!!

$$\begin{bmatrix} \bar{A} \\ \bar{G} \end{bmatrix}_s = \begin{cases} \frac{1}{n}\sum_{i=1}^{n} \begin{bmatrix} A \\ G \end{bmatrix}_{s,i} & s = 1 \\ \frac{1}{n}\sum_{i=1}^{n} \begin{bmatrix} A \\ G \end{bmatrix}_{s,i} - \begin{bmatrix} cor_A \\ cor_G \end{bmatrix}_{s-1} & s > 1 \end{cases} \tag{4}$$

$$cor_s = \begin{bmatrix} cor_A \\ cor_G \end{bmatrix}_s = \begin{cases} \frac{1}{8}\left(\begin{bmatrix} A_0 \\ G_0 \end{bmatrix} - \begin{bmatrix} \bar{A} \\ \bar{G} \end{bmatrix}_1\right) & s = 1 \\ cor_{s-1} - diag(1/a_{x,th}, 1/a_{y,th}, 1/a_{z,th}, 1/g_{x,th}, 1/g_{y,th}, 1/g_{z,th})\left(\begin{bmatrix} A_0 \\ G_0 \end{bmatrix} - \begin{bmatrix} \bar{A} \\ \bar{G} \end{bmatrix}_1\right) & s > 1 \end{cases} \tag{5}$$

2.3.2. IMU data fusion filters initialization

As an IMU module inherent data fusion, retrieved both from accelerometer and gyroscope, Madgwick filter [10] has been used. This filter estimates the IMU module orientation in quaternion form (Q^I) using accelerometer data (A), gyroscope data (G) and filtration factor (f_m) based on gyroscope drift characteristics ome time interval (Δt) that measures time between two consecutive filter executions. Filter initialization requires multiple executions, when IMU module is not-moving, with some average accelerometer and gyroscope measurements as well as relatively high value of filtration factor (i.e. $f_m = 2$). After completion of the initialization process f_m should be defined according to the average angle random walk value that can be calculated with Allan's variance. In case of exploited MPU-6050, IMU modules coefficient $f_m = 0.082$. Madgwick's filter formula is presented with equation 6.

$$Q^I = m(A, G, f_m, \Delta t) \tag{6}$$

2.3.3. IMU and Kinect controller data noise correction

Data noise correction is a crucial phase for the data fusion. Its responsibility is to remove as much noise as possible from the raw data. All methods, used in this phase, are related to measurement devices characteristics described earlier in this article.
The first step to improve the IMU module data quality is the accelerometer measurements (A) error compensation due to the device operating temperature (T). For neutral temperature ($T_0 = 25°C$) and correction factor $f_T = 0.0011$, the correction is defined with equation 7. The value of f_T has been estimated in self-conducted experiments.

$$A' = A/(1 + f_T(T - T_0)) \tag{7}$$

Low and high pass filtration, of accelerometer and gyroscope data, is a part of Madgwick filter implementation so there is no necessity to repeat that process. The second measurement device – Kinect controller – requires two data correction steps that need to be performed as to improve the measurement quality: correction of distance estimation and smoothing the joints position estimations. Distance estimation correction is done by function presented in equation 9 which is the opposite of the distance estimation error model (eq. 1). Argument of this function (z) is directly taken from the Kinect's distance estimation API.

$$f(z) = z' = -0.02z^3 + 0.11z^2 - 0.27z + 0.25 \tag{8}$$

Smoothing of the Kinect's joints positions (P_j^K) is necessary as coordinates fluctuations may happen, especially when occlusions occur. Then positions of the same joints, in two consecutive frames (measurements), substantially differ, which is physically and anatomically impossible, while refresh rate of the Kinect is 30Hz. Smoothing can be done by Kinect's firmware, but this approach gives significant delay of the signal interpretation. Alternative approach to smooth positions, which was exploited in the method, was simple low pass filter. Method used 1st order exponential filter to remove unreliable positions estimations defined with equation 9 and filtration factor $f_{LPF} = 0.065$.

$${P'}_{j,t}^K = f_{LPF} * P_{j,t}^K + (1 - f_{LPF}) * {P'}_{j,t-1}^K \tag{9}$$

2.3.4. IMU and Kinect synchronization

Data fusion of any two signals requires signals alignment in time domain. This guarantees fusion of samples that were collected in the same time – synchronically. The goal of IMU and Kinect signals synchronization is to find the time offset τ between two data streams. To synchronize both signals, they must have the same frequency so the IMU signal has been down sampled from 100Hz to 30Hz, which is a nominal frequency of Kinect controller. Down sampling has been implemented in the form of decimation where Kinect's new sample availability defines the time of IMU sample picking. Then the time offset τ between IMU signal ($I(t)$) and Kinect signal ($K(t)$) was defined as a time offset argument of cross-correlation algorithm that gives the maximum value of correlation between these two signals (variable τ_{max}) (eq. 10a and 10b). The value of τ_{max} is added to the timestamp of IMU samples to align it to the Kinect's signal.

$$(I * K)(\tau) = \int_{-\infty}^{+\infty} I(t)K(t+\tau)dt \tag{10a}$$

$$\tau_{max} = \underset{\tau}{argmax}((I * K)(\tau)) \tag{10b}$$

2.3.5. IMU and Kinect data fusion

The data fusion process worked on time aligned filtered data and took into consideration their reliability. The most serious concerns about the quality of data used within fusion were related to the data provided by Microsoft Kinect controller. Decision about Kinect's data quality was based on the joints tracking state combined together with their positions estimation noise level, the current value of angle α between the user and the camera that cannot exceed 50° with the angle measurement variance lower than 1.5°, and information about tracking state of both shoulder joints. If all these conditions were satisfied, signals were fused according to equation 11. The novelty of the presented method, in comparison with literature approaches, is that the fused data represent skeleton bones orientations in the form of Euler angles $E = \begin{bmatrix} \phi & \theta & \psi \end{bmatrix}$, instead of joints resulting positions. Orientation of the bones is estimated from both sources and is represented in form of quaternions. They must be converted to the form of Euler angles before they can be fused. The conversion from quaternion to Euler angles form can be found in [32].

$$E_t^F = \begin{bmatrix} \phi^F \\ \theta^F \\ \psi^F \end{bmatrix}_t = diag(w_\phi, w_\theta, w_\psi) \begin{bmatrix} \phi^I \\ \theta^I \\ \psi^I \end{bmatrix}_t + diag(1-w_\phi, 1-w_\theta, 1-w_\psi) \begin{bmatrix} \phi^K \\ \theta^K \\ \psi^K \end{bmatrix}_t \tag{11}$$

The originally elaborated weights $[w_\phi, w_\theta, w_\psi]$ describe the importance level of IMU measurements. They were based both on the device precision and data completeness, and were defined as $[0.98, 0.05, 0.65]$ respectively. If Kinect's data turned out to be unreliable, they cannot be fused with the IMU data. Averaged bones orientation are then estimated based on the previous fused estimation and current change of IMU orientation, according to equation 12.

$$E_t^F = \begin{bmatrix} \phi^F \\ \theta^F \\ \psi^F \end{bmatrix}_t = \begin{bmatrix} \phi^F \\ \theta^F \\ \psi^F \end{bmatrix}_{t-1} + diag(w_\phi, w_\theta, w_\psi) \left(\begin{bmatrix} \phi^I \\ \theta^I \\ \psi^I \end{bmatrix}_t - \begin{bmatrix} \phi^I \\ \theta^I \\ \psi^I \end{bmatrix}_{t-1} \right) \tag{12}$$

However, in case of Kinect data unreliability, weights $[w_\phi, w_\theta, w_\psi]$ were defined differently. In equation 12 these weights were defined as

$$[0.98, (1 - t_{noise}/10) * 0.65, (1 - t_{noise}/10) * 0.65]$$

. t_{noise} coefficient defines the time, in seconds, that Kinect stayed unreliable and its maximum accepted value was defined to 10s. After this time, estimation of the new bone orientation (only around two axes) stops, until Kinect is available again and the method privileges the IMU signal.

2.3.6. Joints final position estimation

Final joint position ($P_{j,t}^F$) estimation requires the information regarding the position of joint's parent in the hierarchical skeleton model ($P_{j-1,t}^F$), fixed length of the bone between current joint and its parent (l) and the bone orientation estimation ($E_t^F = [\phi^F, \theta^F, \psi^F]_t$). As a result of data fusion, estimated orientations were presented in the form of Euler angles. However, this form of orientation representation may complicate further calculations. To simplify these calculations, conversion from Euler angles to quaternion Q^F_t, according to the formula described in [32] is required. It is worth noticing that Q^F_t represents orientation relative to the initial T-Pose defined at the beginning of motion sequence. At the last step, the bone in the current orientation must be aligned to the position of parent joint. This final joint position calculation procedure is described with equation 13.

$$P^F_{j,t} = P^F_{j-1,t} + Q^F_t * [l,0,0] * {Q^F_t}^{-1} \tag{13}$$

Subsequent limbs' joints estimation requires consecutive hierarchical analysis of skeleton joints and corresponding bones.

3. Results

In order to verify the accuracy of elaborated method, several experiments were performed. They were conducted with the VICON motion capture system, as a source of ground-truth reference data. For 3 subjects hand movements were monitored, being tracked simultaneously with Kinect controller, two IMUs attached to hand arm and forearm bones, as well as VICON tracking system. Each user had to perform 4 exercises, each of them twice with 5 repetitions per single try. Each try started with devices synchronization in the T-Pose and then 5 repetitions of motion began. Exercises focused on horizontal and vertical hand flexion in elbow joint, horizontal arm motion in shoulder and keeping straight arm in T-pose without motion for at least 60s. All these exercises schemes are presented in figure 8.

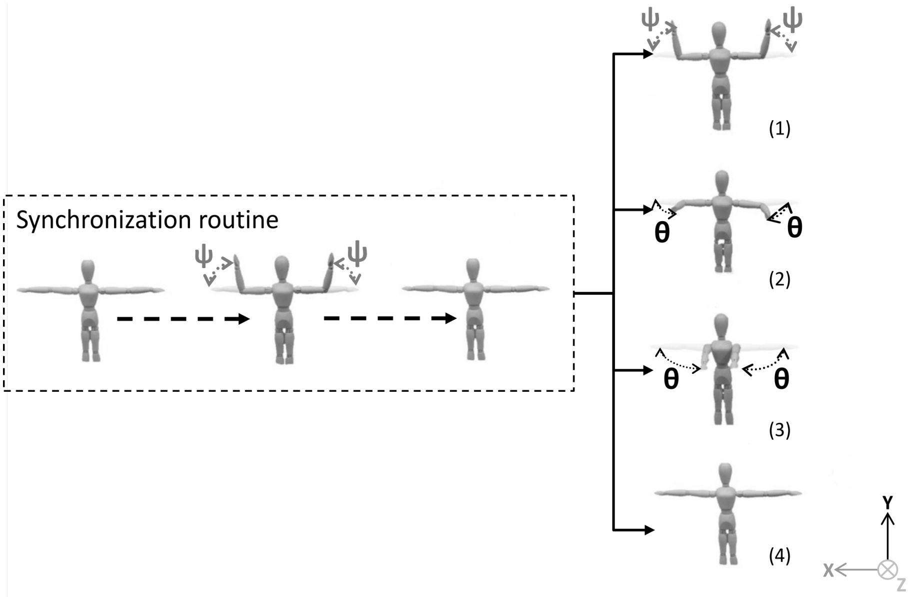


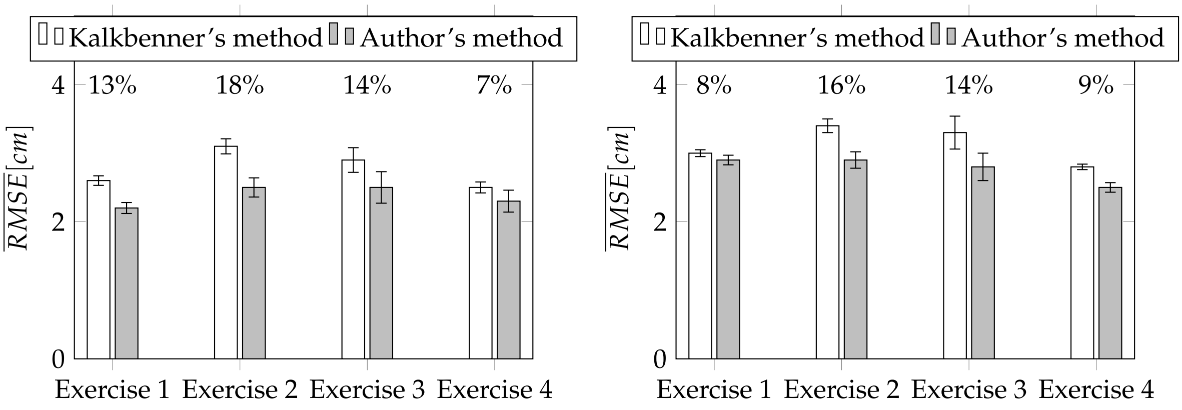
Figure 8. Movement sequences performed during tests

The accuracy of 3 parameters estimations has been observed: the position of elbow joint, the position of wrist joint and the angle between arm and forearm measured in elbow joint. Those values have been compared with the corresponding parameters estimated with Kalkbrenner's method [11] which, on the contrary, fused positions rather than orientations, of selected joints. The position astimation error has been defined as root mean squared error ($RMSE$) of Euclidean distance (d_e) between joint's reference posistion measured by VICON motion capture system (P^V_j) and joint's position estimated by Kalkbrenner method (P^{FP}_j) or between joint's position estimated by authors' orientation-based fusion method (P^{FO}_j). The $RMSE$ value has been calculated with equation 14 [33]. The error for elbow joint angle α estimation has been calculated in the similar way.

$$RMSE_j^F = \sqrt{\frac{1}{n}\sum_{i=1}^{n} d_e(P_{j,i}^V, P_{j,i}^F)^2} \tag{14}$$

The overall error of both methods for each test exercise has been calculated as a mean of $RMSE$ ($\overline{RMSE}$) of each motion tracking session. To compare the Kalkbrenner's method with authors' novel method the ratio r between the difference of $\overline{RMSE}$ of both methods to the $\overline{RMSE}$ of Kalkbrenner's method has been calculated according to the equation 15 and expressed in the form of percents. The figures 9a and 9b show the summary of $\overline{RMSE}$ for tracked upper limb joints: elbow and wrist. Above each pair of bars in these charts, the value of r ratio has been presented. Based on the results presented in figures 9a and 9b the improvement of proposed method in reference to the best in literature - Kalkbrenner's method – by up to 18% for elbow joint position estimation and up to 16% for wrist joint position estimation. Figure 10 presents the chart of $\overline{RMSE}$ of elbow angle β estimation (calculated similarly to eq. 14). In this value estimation the improvement has been also noticed by up to 11%.

$$r = \frac{\overline{RMSE_j^P} - \overline{RMSE_j^O}}{\overline{RMSE_j^P}} \tag{15}$$



(a) Elbow

(b) Wrist

Figure 9. Average root square mean error $\overline{RMSE}$ of elbow (a) and wrist (b) joints position estimation

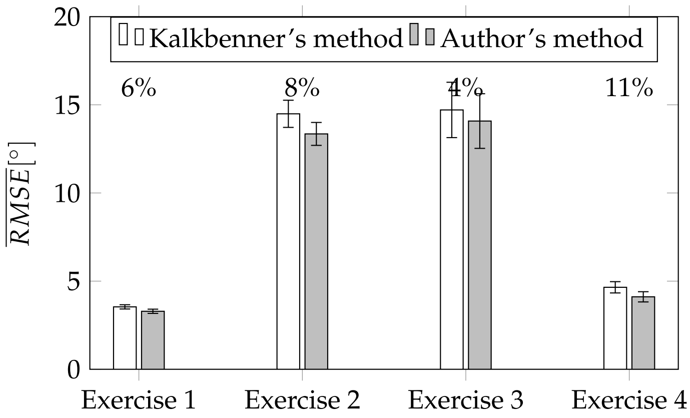


Figure 10. Average root square mean error $\overline{RMSE}$ of elbow flexion angle β

4. Discussion

The presented results show the comparison of estimation errors of two data fusion methods: the well-known method based on joints position fusion and a new data fusion method based on bones orientations. 3 factors estimated by both methods have been taken into consideration in this comparison: the error of elbow joint position estimation, the error of wrist joint position estimation and the error of the estimation of angle between the arm and the forearm of the tracked hand. The motions used to test these methods have been chosen to check how methods work when source data are poor quality due to the limitations and imperfections of measurement devices. The results of the comaprison of the error of elbow and wrist joints positions estimations are presented in figures 9a and 9b respectively. Charts present that the author's novel data fusion method reduces the joints position estimation error, especially during motions that emphasize the imperfections of Microsoft Kinect device (exercises 2 and 3). In these motions the most significant improvement in estimation error has been noticed - up to 18% for the elbow position esitmation and up to 16% for the wrist position estimation. In two other motions, the improvement has been noticed as well, however it was slightly less substantial than in the exercises 2 and 3. This shows that the proposed orientation-based data fusion method is able to detect the poor Kinect data quality faster than the method proposed by Kalkbrenner ~~et al.~~ and roduce the influence of these data on the final result. Figure 10 shows the results of the elbow angle estimation during the motion. In this factor, the estimation error reduction has been noticed as well, and the best result was close to 11%. However, in exercises 2 and 3, which might be considered as the most difficult from the data quality detection point of view, the improvement is slighter than in the case of joints position estimation. That means that both methods have similar accuracy of estimation of the body/limbs shape, however, the authors' method is more accurate in the joints position estimation. The most possible explanation for this, is the fact that the authors' method uses the fixed bones lenght definition,while the Kalkbrenner's method uses the Kinect's bone lenght temporary estimation that may vary between subsequent measurement frames.

5. Conclusions

The authors presented a new, orientation-based method for skeleton joints positioning, that compensates devices' imperfections and consideres the context of the motion. It was tested on a variety of right hand movements and results were compared with those gathered from the position-based fusion method and the reference, professional, ground-truth Vicon tracking system. With the results achieved from the experiments, one can notice the improvement of the elbow positioning accuracy up to 18%, the wrist positioning up to 16% and the elbow joint angle estimation accuracy up to 11%. Obtained results prove that the novel data fusion approach, based on the bones orientation, might be considered as an improved alternative to the well-known, joint position-based methods and it is worth continuing the research in this subject.

Author Contributions: G. Glonek designed the method and experiments and analyzed the data; A.Wojciechowski contributed and supervised the work; G. Glonek and A. Wojciechowski wrote the paper.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

RMSE	Root Mean Squared Error
IMU	Inertial Measurement Units
LPF	Low Pass Filter

Bibliography

1. Johansson, G. Visual perception of biological motion and a model for its analysis. *Perception & Psychophysics* **1973**, *14*, 201–211.

2. Lange, B.; Koenig, S.; McConnell, E.; Chang, C.Y.; Juang, R.; Suma, E.; Bolas, M.; Rizzo, A. Interactive game-based rehabilitation using the Microsoft Kinect. 2012 IEEE Virtual Reality (VR). IEEE, 2012, pp. 171–172.
3. Chang, Y.J.; Chen, S.F.; Huang, J.D. A Kinect-based system for physical rehabilitation: A pilot study for young adults with motor disabilities. *Research in Developmental Disabilities* **2011**, *32*, 2566–2570.
4. Huang, B.; Wu, X. Pedometer Algorithm Research Based-Matlab. In *Advances in Computer Science and Information Engineering: Volume 1*; Jin, D.; Lin, S., Eds.; Springer Berlin Heidelberg: Berlin, Heidelberg, 2012; pp. 81–86.
5. Jayalath, S.; Abhayasinghe, N. A gyroscopic data based pedometer algorithm. 2013 8th International Conference on Computer Science & Education. IEEE, 2013, number October, pp. 551–555.
6. Pedley, M. Tilt Sensing Using a Three-Axis Accelerometer. Technical report, Freescale Semiconductor, 2013.
7. Xsens Corp. Products - Xsens 3D motion tracking.
8. Bo, A.P.L.; Hayashibe, M.; Poignet, P. Joint angle estimation in rehabilitation with inertial sensors and its integration with Kinect. 2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE, 2011, Vol. 2011, pp. 3479–3483.
9. Destelle, F.; Ahmadi, A.; O'Connor, N.E.; Moran, K.; Chatzitofis, A.; Zarpalas, D.; Daras, P. Low-cost accurate skeleton tracking based on fusion of kinect and wearable inertial sensors. 22nd European Signal Processing Conference (EUSIPCO); , 2014; pp. 371–375.
10. Madgwick, S.O.H.; Harrison, A.J.L.; Vaidyanathan, R. Estimation of IMU and MARG orientation using a gradient descent algorithm. 2011 IEEE International Conference on Rehabilitation Robotics. IEEE, 2011, pp. 1–7.
11. Kalkbrenner, C.; Hacker, S.; Algorri, M.e.; Blechschmidt-trapp, R. Motion Capturing with Inertial Measurement Units and Kinect - Tracking of Limb Movement using Optical and Orientation Information. Proceedings of the International Conference on Biomedical Electronics and Devices. SCITEPRESS - Science and and Technology Publications, 2014, pp. 120–126.
12. Shimin Feng.; Murray-Smith, R. Fusing Kinect Sensor and Inertial Sensors with Multi-rate Kalman Filter. IET Conference on Data Fusion & Target Tracking 2014: Algorithms and Applications. Institution of Engineering and Technology, 2014, pp. 2.3–2.3.
13. Tannous, H.; Istrate, D.; Benlarbi-Delai, A.; Sarrazin, J.; Gamet, D.; Ho Ba Tho, M.; Dao, T. A New Multi-Sensor Fusion Scheme to Improve the Accuracy of Knee Flexion Kinematics for Functional Rehabilitation Movements. *Sensors* **2016**, *16*, 1914.
14. Skalski, A.; Machura, B. Metrological analysis of microsoft kinect in the context of object localization. *Metrology and Measurement Systems* **2015**, *22*, 469–478.
15. Gonzalez-Jorge, H.; Riveiro, B.; Vazquez-Fernandez, E.; Martínez-Sánchez, J.; Arias, P. Metrological evaluation of Microsoft Kinect and Asus Xtion sensors. *Measurement* **2013**, *46*, 1800–1806.
16. Khoshelham, K.; Elberink, S.O. Accuracy and Resolution of Kinect Depth Data for Indoor Mapping Applications. *Sensors* **2012**, *12*, 1437–1454.
17. DiFilippo, N.M.; Jouaneh, M.K. Characterization of Different Microsoft Kinect Sensor Models. *IEEE Sensors Journal* **2015**, *15*, 4554–4564.
18. Stackoverflow Community. Precision of the kinect depth camera.
19. Freedman, B.; Shpunt, A.; Machline, M.; Yoel, A. DEPTH MAPPING USING PROJECTED PATTERNS, 2010.
20. Shpunt, A. DEPTH MAPPING USING MULTI-BEAM ILLUMINATION, 2010.
21. Shpunt, A.; Petah, T.; Zalevsky, Z.; Rosh, H. Depth-varying light fields for three dimensional sensing, 2008.
22. Reichinger, A. Kinect Pattern Uncovered, 2011.
23. Fofi, D.; Sliwa, T.; Voisin, Y. A comparative survey on invisible structured light. SPIE Electron. ImagingMachine Vis. Appl. Ind. Insp. XII San Jos{é} USA; Price, J.R.; Meriaudeau, F., Eds., 2004, p. 90.
24. Rzeszotarski, D.; Strumiłło, P.; Pełczyński, P.; Więcek, B.; Lorenc, A. SYSTEM OBRAZOWANIA STEREOSKOPOWEGO SEKWENCJI SCEN TRÓJWYMIAROWYCH. *Zeszyty Naukowe Elektronika* **2006**, *10/2005*, 165–181.

25. Shotton, J.; Johnson, M.; Cipolla, R. Semantic texton forests for image categorization and segmentation. 2008 IEEE Conference on Computer Vision and Pattern Recognition. IEEE, 2008, pp. 1–8.
26. Shotton, J.; Sharp, T.; Kipman, A.; Fitzgibbon, A.; Finocchio, M.; Blake, A.; Cook, M.; Moore, R. Real-time human pose recognition in parts from single depth images. *Communications of the ACM* **2013**, *56*, 116, [1111.[illegible]9v1].
27. Asteriadis, S.; Chatzitofis, A.; Zarpalas, D.; Alexiadis, D.S.; Daras, P. Estimating human motion from multiple Kinect sensors. Proceedings of the 6th International Conference on Computer Vision / Computer Graphics Collaboration Techniques and Applications - MIRAGE '13; ACM Press: New York, New York, USA, 2013; p. 1.
28. Kitsikidis, A.; Dimitropoulos, K.; Douka, S.; Grammalidis, N. Dance Analysis using Multiple Kinect Sensors. 2014 International Conference on Computer Vision Theory and Applications (VISAPP), 2014, pp. 789–795.
29. Schröder, Y.; Scholz, A.; Berger, K.; Ruhl, K.; Guthe, S.; Magnor, M. Multiple Kinect Studies. Technical report, Technische Universität Braunschweig, Braunschweig, 2011.
30. Gebhardt, S.; Scheinert, G.; Uhlmann, F.H. Temperature influence on capacitive sensor structures. INFORMATION TECHNOLOGY AND ELECTRICAL ENGINEERING - DEVICES AND SYSTEMS, MATERIALS AND TECHNOLOGIES FOR THE FUTURE, 2006, number September, pp. 11–15.
31. Grigorie, M.; de Raad, C.; Krummenacher, F.; Enz, C. Analog Temperature Compensation for Capacitive Sensor Interfaces. Proceedings of the 22nd European Solid-State Circuits Conference, 1996. ESSCIRC '96., 1996, pp. 388–391.
32. Dunn, F.; Parberry, I. Rotation in Three Dimensions. In *3D Math Primer for Graphics and Game Development, 2nd Edition*, 2 ed.; A K Peters/CRC Press, 2011; chapter 8, pp. 217–294.
33. Armstrong, J.S.; Collopy, F. Error measures for generalizing about forecasting methods: Empirical comparisons. *International Journal of Forecasting* **1992**, *8*, 69–80.