

Article

Ski Jumping Trajectory Reconstruction Using Wearable Sensors via Extended Rauch-Tung-Striebel Smoother with State Constraints

Xiang Fang ^{1,*}, Benedikt Grüter ¹, Patrick Piprek ¹, Veronica Bessone ²,
Johannes Petrat ^{2,3} and Florian Holzapfel ¹

¹ Institute of Flight System Dynamics, Technical University of Munich, 85748 Garching, Germany; benedikt.grueter@tum.de (B.G.); patrick.piprek@tum.de (P.P.); florian.holzapfel@tum.de (F.H.)

² Department of Biomechanics in Sports, Faculty of Sport and Health Sciences, Technical University of Munich, 80992 Munich, Germany; veronica.bessone@tum.de (V.B.); johannes.petrat@tum.de (J.P.)

³ Olympic Training Center of Bavaria, 80809 Munich, Germany

* Correspondence: xiang.fang@tum.de; Tel.: +49-89-289-16059

Received: 13 January 2020; Accepted: 30 March 2020; Published: 2 April 2020



Abstract: To satisfy an increasing demand to reconstruct an athlete's motion for performance analysis, this paper proposes a new method for reconstructing the position and velocity in the context of ski jumping trajectories. Therefore, state-of-the-art wearable sensors, including an inertial measurement unit, a magnetometer, and a GPS logger are used. The method employs an extended Rauch-Tung-Striebel smoother with state constraints to estimate state information offline from recorded raw measurements. In comparison to the classic inertial navigation system and GPS integration solution, the proposed method includes additional geometric shape information of the ski jumping hill, which are modeled as soft constraints and embedded into the estimation framework to improve the position and velocity estimation accuracy. Results for both simulated measurement data and real measurement data demonstrate the effectiveness of the proposed method. Moreover, a comparison between jump lengths obtained from the proposed method and video recordings shows the relative root-mean-square error of the reconstructed jump length is below 1.5 m depicting the accuracy of the algorithm.

Keywords: state estimation; constrained filtering; position and velocity estimation; inertial sensors; GPS; sensors fusion

1. Introduction

Acquiring an accurate estimation of the position and velocity of the athlete during a ski jump has always been of great interest to athletes, coaches, and sport researchers to analyze the movement for improving the jumping performance. Traditional solutions to meet this demand are video analysis techniques based on camera recorded videos [1]. However, such a system has the disadvantage that it generally only covers limited parts of the jumping area [2]. To record the entire jump, a large number of cameras would be required making the system expensive and hard to calibrate. Furthermore, camera systems merely provide information about the athlete's position, while velocity and acceleration have to be estimated via numeric differentiation, which may be subject to large error.

In recent years, wearable sensors, such as inertial measurement units (IMUs) and global navigation satellite system (GNSS) receivers, have been widely employed for motion analysis in sports including ski jumping [3]. Chardonens et al. [4] presented an IMU-based system measuring ski jumping dynamics including the position and velocity of the center of mass perpendicular to the table during take-off. Groh et al. [2] proposed a measurement system with IMUs and a light barrier to estimate

the ski velocity and jump length. Blumenbach [5] designed and tested a helmet with a high-precision Global Positioning System (GPS) receiver for the positioning of ski jumpers. Fasel et al. [6] presented a solution to estimate the center of mass and its velocity using differential GNSS and IMUs for alpine skiing. In comparison to video analysis techniques, trajectory estimation using wearable IMUs and GNSS sensors has the advantages of being easy to use and maintain and having full coverage over all phases in a ski jump. For coaches and athletes, such a system provides the possibility to give feedback after each jump regarding the jumper's performance, such as jump length and take-off velocity data.

Trajectory reconstruction techniques are a class of methods to achieve an accurate state estimation of a moving object by properly combining information from the kinematic model with measurement data from sensors, including but not limited to IMUs, magnetometers, and GNSS receivers. Trajectory reconstruction techniques are also termed flight path reconstruction in the aerospace field. A systematic overview of such methods for an aircraft can be found in [7]. Göttlicher and Holzapfel [8] reconstructed an aircraft's flight path using low-cost sensors with the extended Rauch-Tung-Striebel (RTS) smoother. Similar techniques can be also applied to ski jumping applications.

From a methodological point of view, a trajectory reconstruction problem can be interpreted as a state estimation problem: One powerful tool to solve this is the classic RTS smoother, which was first proposed for linear systems in [9] using a linear Kalman filter [10] as the forward-time estimator. The extended RTS smoother is a modification of the classical RTS smoother capable of considering nonlinear systems. It employs the extended Kalman filter (EKF) as the forward-time filter to deal with the nonlinearity of the system. In this paper, the extended RTS smoother is adopted as an estimator. Moreover, we can formulate state equality constraints in a state estimation problem. Different methods to solve state estimation problems with state constraints, which are termed as constrained filtering (or smoothing) problems, are discussed in the survey [11].

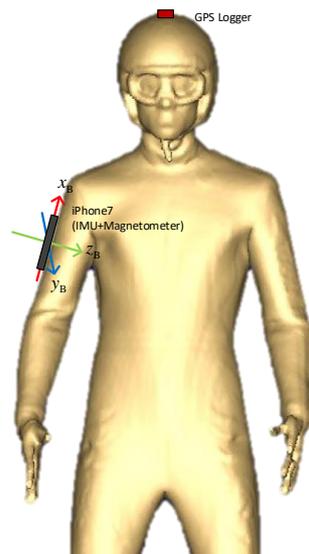
In this paper, we present an offline method based on the extended RTS smoother to estimate the trajectory, including the position and velocity, of a ski jumper by the inertial navigation system and GPS integration. Additionally, to fully utilize all available information to improve the estimation quality, the geometric shape of the ski jumping hill is modeled as a set of additional soft constraints and included in the smoothing framework.

2. Data Acquisition

The measurement data were collected during a summer session in June 2018 on the jumping hill Schattenbergschanze (Hill Size = 106 m) in Oberstdorf, Germany. During this summer session, the hill slope was covered with water-soaked plastic. The measurement setup consists of the following sensors: a Qstarz BT Q1000eX GPS data logger and an iPhone 7, which contains an InvenSense ICM-20600 IMU, and an Alps HSCDTD008A magnetometer. Table 1 summarizes the performance characteristics of the sensors provided by the respective manufacturers. The GPS data logger provides position and velocity measurements with a 10 Hz sample rate. The IMU measures triaxial rotational rates as well as specific forces (i.e., non-gravitational accelerations) at 100 Hz frequency. The magnetometer provides triaxial local magnetic field flux density measurements also at 100 Hz. To the best of our knowledge, the IMU and magnetometer sensor chips in the iPhone 7 provide raw measurement data with a similar accuracy level as previous studies, e.g., [2,4,6,8]. The raw data measured by the smartphone was logged via the application Phyphox [12] and was attached to the jumper's right upper arm using a running armband with the orientation shown in Figure 1. The GPS logger was fixed to the top of the helmet by adhesive tapes for better signal reception.

Table 1. Sensors and their key performance information.

Sensor	Type	Frequency	Performance Characteristics
GPS logger	Qstarz BT Q1000eX	10 Hz	Position accuracy: 3 m circular error probable (50%), velocity accuracy: 0.1 m/s. [13]
IMU	InvenSense ICM-20600	100 Hz	Gyroscope: measurement range: $\pm 2000^\circ/\text{s}$, rate noise spectral density: $\pm 0.004^\circ/\text{s}/\sqrt{\text{Hz}}$; Accelerometer: measurement range: $\pm 16\text{ g}$, noise spectral density: $100\ \mu\text{g}/\sqrt{\text{Hz}}$. [14]
Magnetometer	Alps HSCDTD008A	100 Hz	Measurement range: $\pm 2.4\text{ mT}$. [15]

**Figure 1.** Illustrative figure on the placement of the sensors (based on a body scanning figure).

3. Methods

In this section, we first introduce the extended RTS smoother framework for a general system in Section 3.1. Next, modifications made for the joint state and parameter estimation as well as the state constraints in Section 3.2. Finally, the system model for ski jumping, which is implemented in the extended RTS smoother framework, is presented in Section 3.3.

3.1. Extended Rauch-Tung-Striebel Smoother

The extended RTS smoother is an offline estimation method utilizing the entire batch of measurements over a fixed time interval [16]. It combines a forward filtering pass using the EKF with a backward smoothing pass with the RTS smoother.

A state-space representation for a general nonlinear system described by ordinary differential equations (ODEs) can be written as

$$\dot{x} = f(x, u, w), \quad x(t_0) = x_0, \quad (1)$$

$$y = g(x), \quad (2)$$

$$z = y + v, \quad (3)$$

where $\mathbf{x} \in \mathbb{R}^{n_x}$ is the state vector, $\mathbf{u} \in \mathbb{R}^{n_u}$ is the known input vector, $\mathbf{y} \in \mathbb{R}^{n_y}$ is the model outputs, and $\mathbf{z} \in \mathbb{R}^{n_z}$ represents the measurements. Symbols $\mathbf{w} \in \mathbb{R}^{n_w}$ and $\mathbf{v} \in \mathbb{R}^{n_v}$ denote process and measurement noise vectors. This system can be linearized and discretized into the following form [16]:

$$\mathbf{x}_k = \Phi_{k-1}\mathbf{x}_{k-1} + \Gamma_{u,k-1}\mathbf{u}_{k-1} + \Gamma_{k-1}\mathbf{w}_{k-1}, \quad (4)$$

$$\mathbf{y}_k = \mathbf{H}_k\mathbf{x}_k, \quad (5)$$

$$\mathbf{z}_k = \mathbf{y}_k + \mathbf{v}_k, \quad (6)$$

where $\mathbf{H}_k = \partial \mathbf{g}(\mathbf{x}_k) / \partial \mathbf{x}$. The transformation matrices Φ_{k-1} , $\Gamma_{u,k-1}$, and Γ_{k-1} are computed as

$$\Phi_{k-1} = \exp(\mathbf{A}_{k-1}\Delta t_{k-1}) = \sum_{i=0}^{\infty} \frac{(\mathbf{A}_{k-1}\Delta t_{k-1})^i}{i!}, \quad (7)$$

$$\Gamma_{u,k-1} = \int_0^{\Delta t_{k-1}} \exp(\mathbf{A}_{k-1}\tau) \mathbf{B}_{k-1} d\tau = \left(\sum_{i=1}^{\infty} \frac{\mathbf{A}_{k-1}^{i-1} \Delta t_{k-1}^i}{i!} \right) \mathbf{B}_{k-1}, \quad (8)$$

$$\Gamma_{k-1} = \int_0^{\Delta t_{k-1}} \exp(\mathbf{A}_{k-1}\tau) \mathbf{F}_{k-1} d\tau = \left(\sum_{i=1}^{\infty} \frac{\mathbf{A}_{k-1}^{i-1} \Delta t_{k-1}^i}{i!} \right) \mathbf{F}_{k-1}, \quad (9)$$

where $\mathbf{A}_{k-1} = \frac{\partial f(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1})}{\partial \mathbf{x}}$, $\mathbf{B}_{k-1} = \frac{\partial f(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1})}{\partial \mathbf{u}}$, $\mathbf{F}_{k-1} = \frac{\partial f(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1})}{\partial \mathbf{w}}$, and $\Delta t_{k-1} = t_k - t_{k-1}$. The noise processes $\{\mathbf{w}_k\}$ and $\{\mathbf{v}_k\}$ are assumed to be zero-mean, uncorrelated, white Gaussians, i.e., $\mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k)$, $\mathbf{v}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k)$, $\mathbb{E}[\mathbf{w}_j \mathbf{w}_k] = \mathbf{Q}_k \delta_{jk}$, $\mathbb{E}[\mathbf{v}_j \mathbf{v}_k] = \mathbf{R}_k \delta_{jk}$, and $\mathbb{E}[\mathbf{w}_j \mathbf{v}_k] = \mathbf{0}$, where δ_{jk} is the Kronecker delta.

3.1.1. Extended Kalman Filter

The EKF is a sequential state estimation method for nonlinear systems [11,16]. First, the initial state $\hat{\mathbf{x}}_{0|0}$ and covariance $\mathbf{P}_{0|0}$ for the EKF are given as

$$\hat{\mathbf{x}}_{0|0} = \bar{\mathbf{x}}_0, \quad (10)$$

$$\mathbf{P}_{0|0} = \bar{\mathbf{P}}_0, \quad (11)$$

where $\bar{\mathbf{x}}_0$ is an a priori estimate of the initial states \mathbf{x}_0 and $\bar{\mathbf{P}}_0$ is the covariance of the estimation error of $\bar{\mathbf{x}}_0$. Then, for each time point t_k with $k \in \{1, 2, \dots, N\}$, the EKF consist of a prediction and an update step as follows:

1) Prediction step:

$$\hat{\mathbf{x}}_{k|k-1} = \hat{\mathbf{x}}_{k-1|k-1} + \int_{t_{k-1}}^{t_k} \mathbf{f}(\hat{\mathbf{x}}_{k-1|k-1}, \tilde{\mathbf{u}}_{k-1}, \mathbf{0}) d\tau, \quad (12)$$

$$\mathbf{P}_{k|k-1} = \Phi_{k-1}^\top \mathbf{P}_{k-1|k-1} \Phi_{k-1} + \Gamma_{k-1}^\top \mathbf{Q}_{k-1} \Gamma_{k-1}, \quad (13)$$

where $\hat{\mathbf{x}}_{k|k-1}$ represents the a priori state estimate at time t_k based on measurements up to time t_{k-1} , i.e., $\hat{\mathbf{x}}_{k|k-1} = \mathbb{E}(\mathbf{x}_k | \mathbf{z}_1, \dots, \mathbf{z}_{k-1})$, and $\mathbf{P}_{k|k-1}$ denotes the covariance of the estimation error of $\hat{\mathbf{x}}_{k|k-1}$. To evaluate Φ_{k-1} and Γ_{k-1} , matrices \mathbf{A}_{k-1} and \mathbf{F}_{k-1} in Equation (7) and Equation (9) are linearized as $\mathbf{A}_{k-1} = \frac{\partial f(\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{u}_{k-1}, \mathbf{0})}{\partial \mathbf{x}}$ and $\mathbf{F}_{k-1} = \frac{\partial f(\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{u}_{k-1}, \mathbf{0})}{\partial \mathbf{w}}$. The integral in Equation (12) is numerically solved via the classic 4-th order Runge–Kutta method with $\tilde{\mathbf{u}}_{k-1} = \frac{t_k - \tau}{t_k - t_{k-1}} \mathbf{u}_{k-1} + \frac{\tau - t_{k-1}}{t_k - t_{k-1}} \mathbf{u}_k$.

2) Update step:

$$\mathbf{K}_k = \mathbf{P}_{k|k-1} \mathbf{H}_k^\top [\mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}_k^\top + \mathbf{R}_k]^{-1}, \quad (14)$$

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k [\mathbf{z}_k - \mathbf{g}(\hat{\mathbf{x}}_{k|k-1})], \quad (15)$$

$$\mathbf{P}_{k|k} = [\mathbf{I}_{n_x} - \mathbf{K}_k \mathbf{H}_k] \mathbf{P}_{k|k-1}, \quad (16)$$

where $\hat{x}_{k|k}$ denotes the a posteriori state estimate at time t_k based on measurements up to time t_k , i.e., $\hat{x}_{k|k} = \mathbb{E}(x_k | z_1, \dots, z_k)$, and $P_{k|k}$ denotes the covariance of the estimation error of $\hat{x}_{k|k}$. The symbol I_{n_x} denotes an n_x by n_x identity matrix, and H_k is approximated by $H_k = \partial g(\hat{x}_{k|k-1}) / \partial x$.

3.1.2. Rauch-Tung-Striebel Smoother

After the forward-time filtering pass, the EKF estimation of the states $\hat{x}_{N|N}$ and its error covariance $P_{N|N}$ is used to initialize the corresponding values in the RTS smoother for the final time point t_N :

$$\hat{x}_N^s = \hat{x}_{N|N}, \quad (17)$$

$$P_N^s = P_{N|N}, \quad (18)$$

where the superscripts "s" indicate the smoothed estimates. Then, the RTS smoother runs backward in time for t_k with $k \in \{N-1, N-2, \dots, 0\}$ as

$$K_k^s = P_{k|k} \Phi_k^\top (P_{k+1|k})^{-1}, \quad (19)$$

$$\hat{x}_k^s = \hat{x}_{k|k} + K_k^s [\hat{x}_{k+1}^s - \hat{x}_{k+1|k}], \quad (20)$$

$$P_k^s = P_{k|k} + K_k^s [P_{k+1}^s - P_{k+1|k}] (K_k^s)^\top, \quad (21)$$

where the smoothed estimate \hat{x}_k^s can be regarded as the state estimate of time point t_k utilizing all measurements, i.e., $\hat{x}_k^s = \mathbb{E}(x_k | z_1, \dots, z_N)$ [17], and P_k^s is the covariance of the estimation error of \hat{x}_k^s .

3.2. Adaption for Parameter Estimation and Soft Constraints

To account for unknown parameters and known constraints for the estimation problem, the extended RTS smoother for state estimation is modified by the following adaptations.

3.2.1. Joint State and Parameter Estimation

When the model also includes parameters $p \in \mathbb{R}^{n_p}$ to be estimated, these parameters are treated as additional states with $\dot{p} = \mathbf{0}$ so that they are compatible with the extended RTS smoother framework. Therefore, with the augmented state vector defined as $x_a = [x^\top \ p^\top]^\top \in \mathbb{R}^{n_x+n_p}$, the augmented system ODEs become

$$\dot{x}_a = f_a(x_a, u, w) = \begin{bmatrix} f(x, u, p, w) \\ \mathbf{0} \end{bmatrix}. \quad (22)$$

3.2.2. Constrained Filtering

Suppose that a system satisfies the nonlinear state equality constraints as

$$c_k = h(x_k). \quad (23)$$

In order to include these into the extended RTS smoother framework, the constraints can be treated as additional pseudo measurements in the form of

$$\begin{bmatrix} z_k \\ c_k \end{bmatrix} = \begin{bmatrix} y_k \\ h(x_k) \end{bmatrix} + \begin{bmatrix} v_k \\ v_{c,k} \end{bmatrix}. \quad (24)$$

For the case that equality constraints are satisfied exactly, which is termed as hard constraints in [11,18], the pseudo measurements can be treated as perfect with pseudo measurement noises $v_{c,k} = \mathbf{0}$. Contrary, soft constraints are only required to be fulfilled approximately, i.e., $c_k \approx h(x_k)$. In this case, we assume $v_{c,k}$ are zero-mean Gaussian noises with small covariance Q_c , i.e., $v_{c,k} \sim \mathcal{N}(\mathbf{0}, Q_c)$.

We further write Equation (24) as

$$\mathbf{y}_a = \mathbf{g}_a(\mathbf{x}_a), \tag{25}$$

$$\mathbf{z}_a = \mathbf{y}_a + \mathbf{v}_a, \tag{26}$$

where $\mathbf{z}_a = [z_k^\top c_k^\top]^\top$, $\mathbf{y}_a = [y_k^\top h(x_k)^\top]^\top$, and $\mathbf{v}_a = [v_k^\top v_{c,k}^\top]^\top$. By Equations (22), (25), and (26), we formulated the state-space presentation of an augmented system for state and parameter estimation with state constraints in the form of Equations (1)–(3).

3.3. System Model

After introducing the estimation framework for a general system in the previous two subsections, the system model for the ski jumping application is presented in this subsection.

3.3.1. Coordinate Frame Definitions

The first necessary frame is the hill reference frame $Ox_Ny_Nz_N$. As shown in Figure 2, its origin O is located at the end of the in-run table. The x_N axis and z_N axis are both within the symmetric plane of the jumping hill and point horizontally forward and vertically down, respectively. The y_N axis is perpendicular to x_N and z_N axes to form a right-handed coordinate system. $\mathbf{r}_N = [x_N y_N z_N]^\top$ denotes the coordinate matrix in the hill reference frame of the jumper's (or, to be exact, the IMU sensor's) relative position vector \mathbf{r} with respect to the origin O . Another necessary frame is the local north-east-down (NED) frame $Ox_Oy_Oz_O$. The origin of the local NED frame coincides with point O , and the three axes x_O , y_O and z_O point to north, east, and down directions respectively. Finally, the sensor's body frame $O_Bx_By_Bz_B$ is defined as following: the origin O_B is located at the center of the cellphone, and the three axes are fixed to its three axes of symmetry as shown in Figure 3. The coordinates in the NED frame of the relative position vector \mathbf{r} are denote by $\mathbf{r}_O = [x_O y_O z_O]^\top$.

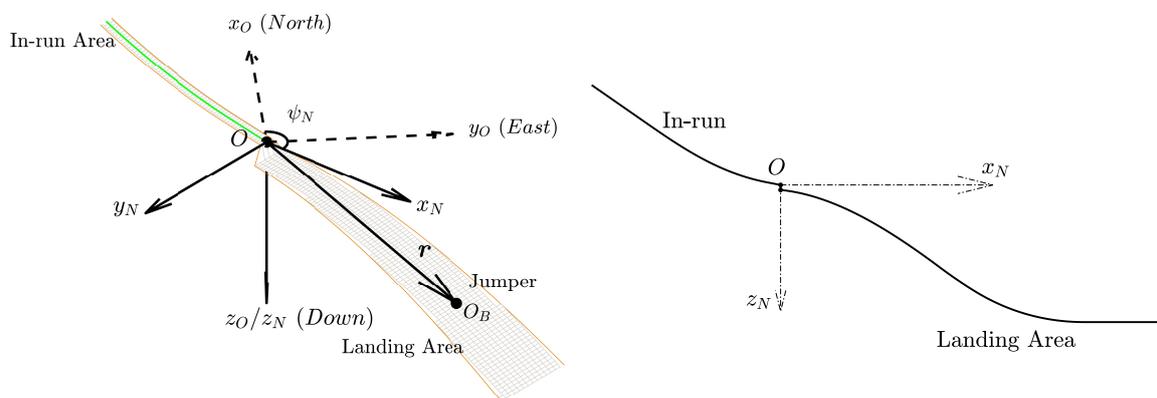


Figure 2. Definitions of the local north-east-down frame and the hill reference frame.

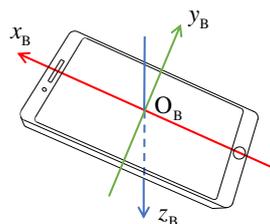


Figure 3. Definition of the sensor body frame.

3.3.2. Conversion of GPS Position Measurements to the Hill Reference Frame

The position measurements from GPS receivers are usually provided in geodetic coordinates, such as the World Geodetic System 1984 (WGS84) frame [19], as a time sequence of latitude μ , longitude λ , and height h . For ski jumping applications, it is more useful to calculate relative positions of the ski jumper with respect to the jumping hill.

With the frame definitions stated before, the GPS-measured positions $\mathbf{r}_{\text{WGS,GPS}} = [\mu \ \lambda \ h]^\top$ in the WGS84 frame are first transformed into the local NED frame $\mathbf{r}_{\text{O,GPS}} = [x_{\text{O}} \ y_{\text{O}} \ z_{\text{O}}]^\top$ by the following equations:

$$x_{\text{O}} = (M_{\mu} + h)(\mu - \mu_0), \quad (27)$$

$$y_{\text{O}} = (N_{\mu} + h)\cos \mu_0(\lambda - \lambda_0), \quad (28)$$

$$z_{\text{O}} = -(h - h_0), \quad (29)$$

where μ_0 , λ_0 , and h_0 are the latitude, longitude, as well as height of the hill reference frame's origin O, and parameters for the WGS84 reference ellipsoid [19] can be found in Table 2.

Table 2. Parameters for the WGS84 reference ellipsoid.

Name	Symbol	Value
The semi-major axis	a	6378137.0 m
The first eccentricity	e	0.0818191908426
The meridian radius of curvature	M_{μ}	$a \frac{1-e^2}{(1-e^2 \sin^2 \mu_0)^{\frac{3}{2}}}$
The radius of curvature in the prime vertical	N_{μ}	$\frac{a}{\sqrt{1-e^2 \sin^2 \mu_0}}$

Then, the GPS-measured positions $\mathbf{r}_{\text{N,GPS}}$ in the hill reference frame are obtained by

$$\mathbf{r}_{\text{N,GPS}} = \mathbf{M}_{\text{NO}} \mathbf{r}_{\text{O,GPS}}, \quad (30)$$

where \mathbf{M}_{NO} is the transformation matrix from local NED to hill reference frame as

$$\mathbf{M}_{\text{NO}} = \begin{bmatrix} \cos \psi_{\text{N}} & \sin \psi_{\text{N}} & 0 \\ -\sin \psi_{\text{N}} & \cos \psi_{\text{N}} & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad (31)$$

where ψ_{N} denotes the hill azimuth angle, which is defined as the rotation angle from the x_{O} axis (the north) to the x_{N} axis, as shown in Figure 2.

3.3.3. Constraint Modeling of the Geometric Shape of a Ski Jumping Hill

The geometric shape of the ski jumping hill is used as additional a priori information and included in the trajectory reconstruction problem as a set of constraints for further improvement of the estimation quality. The mathematical model of a ski jumping hill is described in [20], which was published by the International Ski Federation (Fédération Internationale de Ski, FIS) as the guideline to construct ski jumping hills. Additionally, all necessary model parameters can be found in the jumping hill certificate, which is also issued by the FIS. In our case, we have the certificate of jumping hill - No. 215/GER 29 [21] for the Schattenbergschanze.

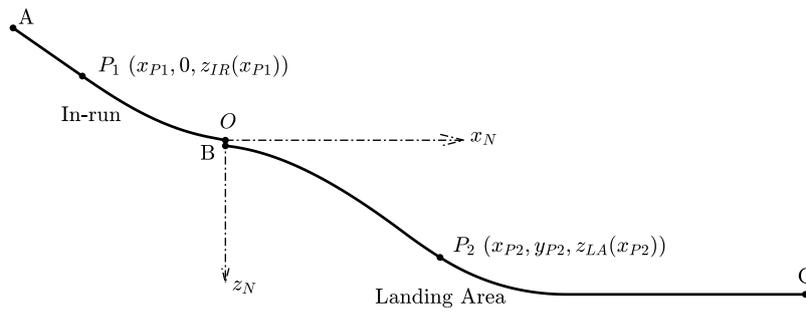


Figure 4. The Longitudinal Profile of a Ski Jumping Hill.

Figure 4 shows the geometric profile of a FIS-certified ski jumping hill in the longitudinal plane. The curve from the point A to the origin O is called the in-run where the jumper skis down following the track before take-off. For any point P_1 on the in-run curve, we assume its coordinates to be $(x_{P1}, 0, z_{IR}(x_{P1}))$ where $z_{P1} = z_{IR}(x_{P1})$ denotes the mapping relationship for the in-run in the vertical plane. Since the trajectory of the jumper before take-off must coincide with the in-run curve, we define the constraints for the in-run curve in the longitudinal and lateral plane as

$$c_{V,IR} = z_N - z_{IR}(x_N), \quad (32)$$

$$c_{H,IR} = y_N. \quad (33)$$

When the jumper is in the in-run area before take-off, i.e., $t_k \leq t_{TO}$, the constraints should fulfill that $c_{V,IR} \approx 0$ and $c_{H,IR} \approx 0$. The curve BC in Figure 4 is the landing area (including the landing slope and the out-run). For a point in the landing area P_2 , its position coordinate in the hill reference frame is $(x_{P2}, y_{P2}, z_{LA}(x_{P2}))$ where $z_{P2} = z_{LA}(x_{P2})$ is the mapping relationship of the landing area in vertical plane. The jumper would land somewhere in between and afterward skiing along this surface. Therefore, we can formulate another constraint as follows:

$$c_{V,LA} = z_N - z_{LA}(x_N). \quad (34)$$

After the jumper's landing, i.e., $t_k \geq t_{TD}$, the constraints should be active such that $c_{V,LA} \approx 0$. The method for detecting the take-off time point t_{TO} and the touch-down time point t_{TD} using raw measurements is introduced in Appendix A.

3.3.4. Measurement Error Models

To model the measurement error of the inertial sensor, constant bias terms $\Delta\omega$ and Δa for the gyroscope and accelerometer as well as zero-mean, white, Gaussian noise terms w_{gyro} and w_{acc} are explicitly considered as

$$\omega_{B,gyro} = \omega_B + \Delta\omega + w_{gyro}, \quad (35)$$

$$a_{B,acc} = a_B + \Delta a + w_{acc}, \quad (36)$$

where $\omega_{B,gyro}$ represents the rotational rates measurement from the gyroscope, $a_{B,acc}$ denotes the measurement from the accelerometer, and ω_B as well as a_B are the true rotational rates and specific forces respectively.

The measurement error model of the magnetometer is considered to be

$$m_{B,mag} = (1 + \text{diag}(\Delta S_m))m_B + \Delta m + v_{mag}, \quad (37)$$

where $m_{B,mag}$ is the magnetometer measurement of the local magnetic field strength, ΔS_m represents

the vector of the scaling factor error and $\text{diag}(\Delta S_m)$ denotes the corresponding square matrix with S_m being its diagonal elements, Δm denotes the constant bias vector, m_B is the true local magnetic field strength, and v_{mag} represents the zero-mean, white, Gaussian measurement noise.

The GPS measurements are modeled as

$$r_{N,\text{GPS}} = r_N + v_{\text{pos}}, \quad (38)$$

$$v_{O,\text{GPS}} = v_O + v_{\text{vel}}, \quad (39)$$

where r_N is the true position in the hill reference frame, v_O is the true velocity in the local NED frame, $r_{N,\text{GPS}}$ and $v_{O,\text{GPS}}$ are the GPS measurements accordingly, and v_{pos} and v_{vel} are the measurement noise terms.

3.3.5. State, Output, and Constraint Equations

The system states vector x is given by

$$x = \begin{bmatrix} r_N^\top & v_B^\top & q_{\text{BO}}^\top \end{bmatrix}^\top, \quad (40)$$

where v_B denotes the triaxial velocity in the body-fixed frame, and q_{BO} represents the attitude quaternions of the body-fixed frame with respect to the local NED frame. The input vector u contains the IMU measurements $\omega_{B,\text{gyro}}$ and $a_{B,\text{acc}}$ as

$$u = \begin{bmatrix} \omega_{B,\text{gyro}}^\top & a_{B,\text{acc}}^\top \end{bmatrix}^\top. \quad (41)$$

The parameter vector p to be estimated includes the biases of the gyroscope $\Delta\omega$, the accelerometer Δa , the magnetometer Δm , and the scaling factor error of the magnetometer ΔS_m :

$$p = \begin{bmatrix} \Delta\omega^\top & \Delta a^\top & \Delta m^\top & \Delta S_m^\top \end{bmatrix}^\top. \quad (42)$$

The process noise vector w consists of influences from both gyroscope and accelerometer:

$$w = \begin{bmatrix} w_{B,\text{gyro}}^\top & w_{B,\text{acc}}^\top \end{bmatrix}^\top. \quad (43)$$

With the definitions above, the state equations $\dot{x} = f(x, u, p, w)$ stated in Equation (22) can be assembled by the equations of inertial navigation mechanism and kinematics as follows:

$$\dot{r}_N = M_{\text{NO}} M_{\text{BO}}^\top v_B, \quad (44)$$

$$\dot{v}_B = a_B + M_{\text{BO}} g_O + \omega_B \times v_B, \quad (45)$$

$$\dot{q}_{\text{BO}} = \frac{1}{2} \begin{bmatrix} -q_1 & -q_2 & -q_3 \\ q_0 & -q_3 & q_2 \\ q_3 & q_0 & -q_1 \\ -q_2 & q_1 & q_0 \end{bmatrix} \omega_B, \quad (46)$$

where M_{BO} is the transformation matrix from the local NED frame to the body-fixed frame as

$$M_{\text{BO}} = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1q_2 + q_0q_3) & 2(q_1q_3 - q_0q_2) \\ 2(q_1q_2 - q_0q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2q_3 + q_0q_1) \\ 2(q_0q_2 + q_1q_3) & 2(q_2q_3 - q_0q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix}, \quad (47)$$

and $g_O = [0 \ 0 \ g_0]^\top$ represents the gravitational vector in the local NED frame. The term a_B

in Equation (45) can be substituted by $\mathbf{a}_B = \mathbf{a}_{B, \text{acc}} - \Delta \mathbf{a} - \mathbf{w}_{\text{acc}}$ using the accelerometer measurement model Equation (36). Similarly, $\boldsymbol{\omega}_B$ can be written as $\boldsymbol{\omega}_B = \boldsymbol{\omega}_{B, \text{gyro}} - \Delta \boldsymbol{\omega} - \mathbf{w}_{\text{gyro}}$ via the gyroscope measurement model Equation (35) and be further substituted into Equation (46).

The output vector \mathbf{y} is written as

$$\mathbf{y} = \begin{bmatrix} \mathbf{r}_N^\top & \mathbf{v}_O^\top & \mathbf{m}_{B, \text{meas}}^\top \end{bmatrix}^\top, \quad (48)$$

where \mathbf{r}_N can be directly obtained from the state vector, the velocity \mathbf{v}_O is calculated as

$$\mathbf{v}_O = \mathbf{M}_{BO}^\top \mathbf{v}_B, \quad (49)$$

and $\mathbf{m}_{B, \text{meas}}$ is the estimated magnetometer measurement. By taking the magnetometer model Equation (37) into account, $\mathbf{m}_{B, \text{meas}}$ is estimated by

$$\mathbf{m}_{B, \text{meas}} = (1 + \text{diag}(\Delta \mathbf{S}_m)) \mathbf{M}_{BO} \mathbf{m}_{O, \text{WMM}} + \Delta \mathbf{m}, \quad (50)$$

where $\mathbf{m}_{O, \text{WMM}}$ is the reference value of the local magnetic field strength in the local NED frame from the World Magnetic Model 2015 [22].

Accordingly, the measurement vector is $\mathbf{z} = \begin{bmatrix} \mathbf{r}_{N, \text{GPS}}^\top & \mathbf{v}_{O, \text{GPS}}^\top & \mathbf{m}_{B, \text{mag}}^\top \end{bmatrix}^\top$ with measurements from the GPS logger ($\mathbf{r}_{N, \text{GPS}}$ and $\mathbf{v}_{O, \text{GPS}}$) and the magnetometer ($\mathbf{m}_{B, \text{mag}}$). The measurement noise vector is therefore $\mathbf{v} = \begin{bmatrix} \mathbf{v}_{\text{pos}}^\top & \mathbf{v}_{\text{vel}}^\top & \mathbf{v}_{\text{mag}}^\top \end{bmatrix}^\top$.

The vector of constraints \mathbf{c} is defined to be

$$\mathbf{c} = \begin{bmatrix} c_q & c_{V, \text{IR}} & c_{H, \text{IR}} & c_{V, \text{LA}} \end{bmatrix}^\top, \quad (51)$$

where $c_q = \|\mathbf{q}_{BO}\|_2 - 1$ is the constraint for attitude quaternions. Here, c_q ensures a unit quaternion, i.e., $\|\mathbf{q}_{BO}\|_2 = 1$. The constraints $c_{V, \text{IR}}$, $c_{H, \text{IR}}$, and $c_{V, \text{LA}}$ are geometric constraints given by the shape of the jumping hill, which are introduced in Section 3.3.3.

4. Results and Discussion

In this Section, trajectory reconstruction results from both simulated and real measurement data are presented. The configuration for the extended RTS smoother is first introduced in Section 4.1. The simulated measurements are used to theoretically validate the purposed method in Section 4.2. Results based on real measurements are presented in Section 4.3. The jump length obtained from both the proposed method and video recordings are compared in Section 4.4 to further validate the results based on real measurement data.

4.1. Setting

We first introduce necessary configurations for the extended RTS smoother algorithm. For the initial states $\bar{\mathbf{x}}_0 = \begin{bmatrix} \bar{\mathbf{r}}_{N,0}^\top & \bar{\mathbf{v}}_{O,0}^\top & \bar{\mathbf{q}}_0 \end{bmatrix}^\top$, the GPS measurement at initial time t_0 is adopted as an estimation for the initial position and velocity as $\bar{\mathbf{r}}_{N,0} = \mathbf{r}_{N, \text{GPS},0}$ and $\bar{\mathbf{v}}_{O,0} = \mathbf{v}_{O, \text{GPS},0}$. The estimated value for initial attitude $\bar{\mathbf{q}}_0$ is calculated via the factored quaternion algorithm proposed in [23] using the accelerometer and magnetometer measurements at time t_0 . The initial guess for all measurement error parameters is chosen to be zero as $\bar{\mathbf{p}}_0 = \begin{bmatrix} \Delta \boldsymbol{\omega}_0^\top & \Delta \mathbf{a}_0^\top & \Delta \mathbf{m}_0^\top & \Delta \mathbf{S}_{m,0}^\top \end{bmatrix}^\top = \mathbf{0}$. For the augmented initial states $\bar{\mathbf{x}}_{a,0} = \begin{bmatrix} \bar{\mathbf{x}}_0^\top & \bar{\mathbf{p}}_0^\top \end{bmatrix}^\top$, the estimated covariance matrix for the initial states error $\bar{\mathbf{P}}_0$ is set as shown in Table 3. In addition, the covariance matrix of noises \mathbf{Q} and \mathbf{R} is considered constant and the estimated values of \mathbf{Q} and \mathbf{R} are listed in Tables 4 and 5 respectively. Within the model, the hill

azimuth angle ψ_N is set as 312.6° for the jumping hill Schattenbergschanze. The gravitational constant at the site is calculated to be $g_0 = 9.8053 \text{ m/s}^2$ according to the Earth Gravitational Model 1996 [19].

Table 3. Diagonal elements of the estimated initial states error covariance matrix \hat{P}_0 .

System States	Symbol	Estimated Variance
Position	$r_{N,0}$	$\text{diag}([5 \ 5 \ 15])^2 \text{ [m}^2\text{]}$
Velocity	$v_{B,0}$	$\text{diag}([1 \ 1 \ 3])^2 \text{ [(m/s)}^2\text{]}$
Attitude quaternions	$q_{BO,0}$	$(0.1\mathbf{I}_4)^2 \text{ [-]}$
Gyroscope bias	$\Delta\omega_0$	$(0.0873\mathbf{I}_3)^2 \text{ [(rad/s)}^2\text{]}$
Accelerometer bias	Δa_0	$(0.2\mathbf{I}_3)^2 \text{ [(m/s}^2\text{)}^2\text{]}$
Magnetometer bias	Δm_0	$(5\mathbf{I}_3)^2 \text{ [(\mu T)}^2\text{]}$
Magnetometer scaling error	$\Delta S_{m,0}$	$(0.1\mathbf{I}_3)^2 \text{ [-]}$

Table 4. Diagonal elements of the estimated process noise covariance matrix Q .

System Inputs	Symbol	Estimated Variance
Gyroscope	$\omega_{B, \text{gyro}}$	$(0.0175\mathbf{I}_3)^2 \text{ [(rad/s)}^2\text{]}$
Accelerometer	$a_{B, \text{acc}}$	$(0.1\mathbf{I}_3)^2 \text{ [(m/s}^2\text{)}^2\text{]}$

Table 5. Diagonal elements of the estimated measurement noise covariance matrix R .

System Measurements	Symbol	Estimated Variance
GPS position	$r_{N, \text{GPS}}$	$\text{diag}([3 \ 3 \ 5])^2 \text{ [m}^2\text{]}$
GPS velocity	$v_{O, \text{GPS}}$	$\text{diag}([0.1 \ 0.1 \ 0.3])^2 \text{ [(m/s)}^2\text{]}$
Magnetometer	$m_{B, \text{mag}}$	$(5\mathbf{I}_3)^2 \text{ [(\mu T)}^2\text{]}$
Quaternions constraint	c_q	0.01^2 [-]
In-run vertical constraint	$c_{V, \text{IR}}$	$0.1^2 \text{ [m}^2\text{]}$
In-run horizontal constraint	$c_{H, \text{IR}}$	$0.1^2 \text{ [m}^2\text{]}$
Land area vertical constraint	$c_{V, \text{LA}}$	$0.1^2 \text{ [m}^2\text{]}$

4.2. Validation by Simulated Measurement Data

To validate the accuracy of the proposed algorithm, an artificial measurement data set is generated by assuming a true value and adding sensor error and noise. This data set is generated based on the result from real measurements as a reference trajectory, ensuring similar observability of the estimation problem as in the real world. Trajectory optimization techniques with least-squares costs are used to obtain a trajectory with similar acceleration and angular velocities as the measured data. By adding path constraints to the trajectory optimization, the true trajectory also fulfills the constraints in Equations (32)–(34). The true value for the generated trajectory also complies with the kinematic models in Equations (44)–(46). The sensor errors and noises are added to the true values according to Equations (35)–(39), with the noise covariance values shown in Tables 4 and 5 to generate artificial measurements.

The trajectory is reconstructed with only the noisy artificial measurements known to the extended RTS smoother. The reconstruction result is presented in Figure 5. Here, it can be observed that the smoothed trajectory agrees closely with the true reference. To quantify the error of the estimated position and velocity, we calculated the error by

$$\Delta r_N = \hat{r}_N^s - r_{N, \text{ref}}, \quad (52)$$

$$\Delta v_O = \hat{v}_O^s - v_{O, \text{ref}}, \quad (53)$$

where variables with subscript “ref” are the true values from the generated data. The error of these estimated positions and velocities are presented in Figure 6a,b respectively. To intuitively display the

attitude, the quaternions q_{BO} are converted to the corresponding Euler angles (in z-y-x order): the bank angle ϕ , the pitch angle θ , and the azimuth angle ψ by

$$\phi = \arctan \left(\frac{2(q_2q_3 + q_0q_1)}{q_0^2 - q_1^2 - q_2^2 + q_3^2} \right), \quad (54)$$

$$\theta = \arcsin(-2(q_1q_3 - q_0q_2)), \quad (55)$$

$$\psi = \arctan \left(\frac{2(q_1q_2 + q_0q_3)}{q_0^2 + q_1^2 - q_2^2 - q_3^2} \right). \quad (56)$$

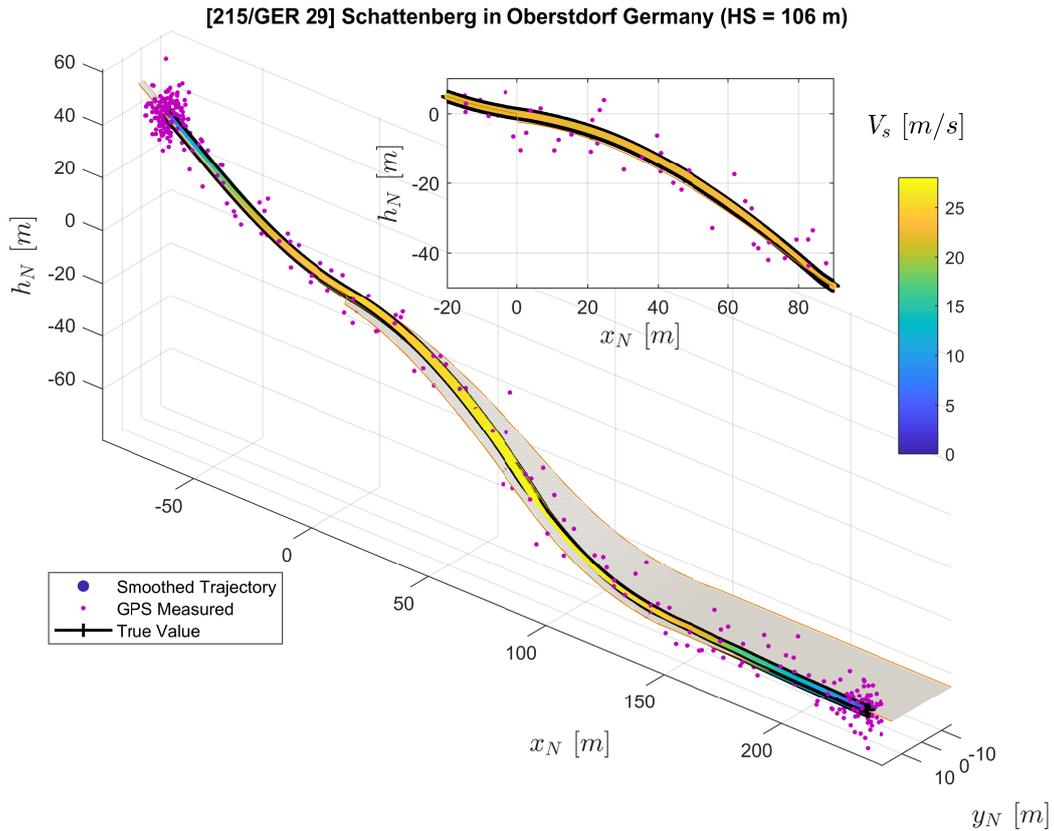


Figure 5. Comparison of the extended RTS smoother reconstructed trajectory (color-coded line), the generated reference trajectory (black line), and the GPS measured trajectory (purple dots) plotted on the ski jumping hill model.

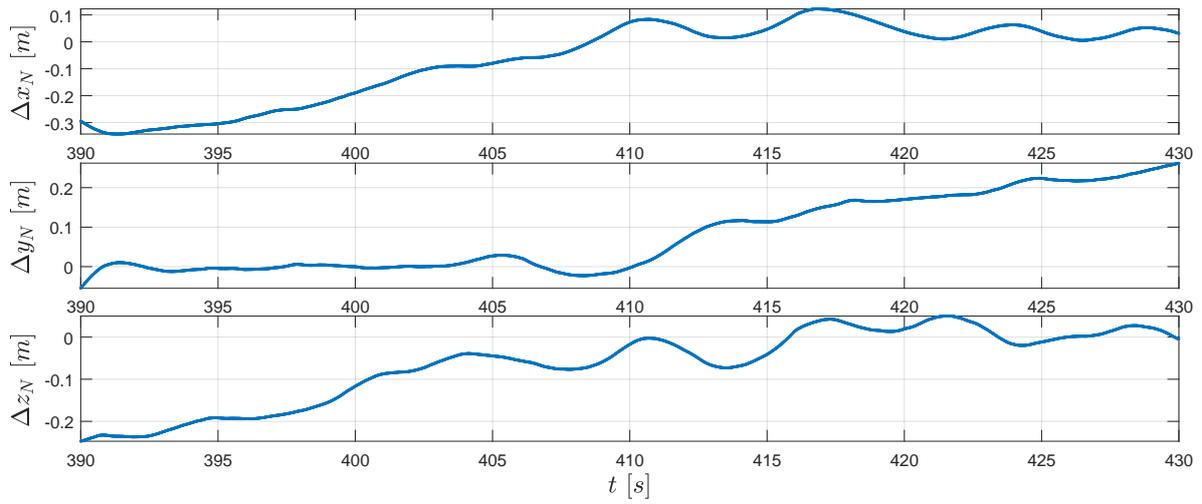
Then, we use the error in Euler angles to represent the attitude estimation error as

$$\Delta\phi = \hat{\phi}^s - \phi_{\text{ref}}, \quad \Delta\theta = \hat{\theta}^s - \theta_{\text{ref}}, \quad \text{and} \quad \Delta\psi = \hat{\psi}^s - \psi_{\text{ref}}. \quad (57)$$

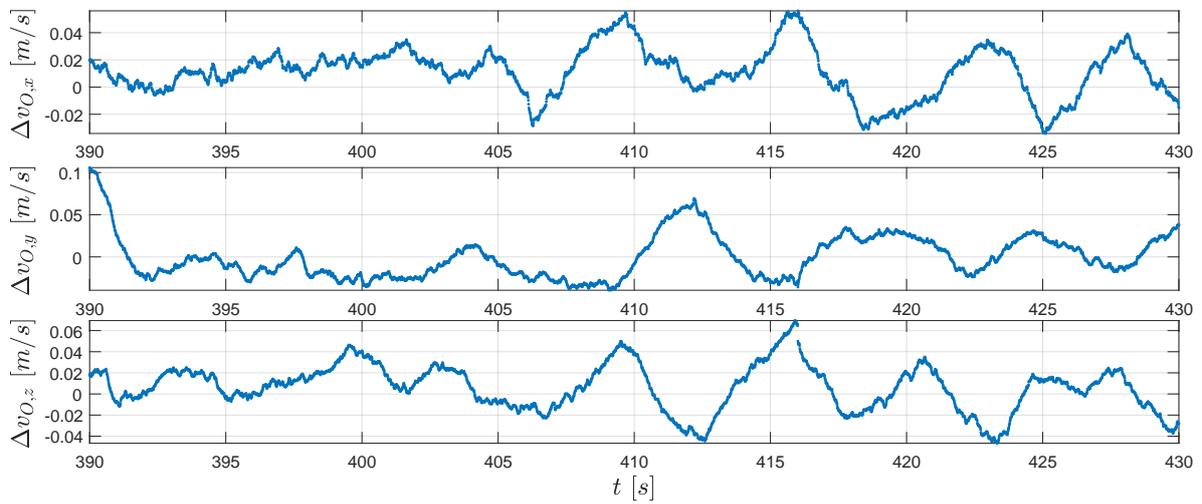
The Euler angle estimation error is presented in Figure 6c. The root-mean-square (RMS) error for position, velocities, and Euler angles are summarized in Table 6. This result shows that the proposed algorithm can successfully reconstruct the trajectory and that the position accuracy for the assumed noise covariance is in decimeter magnitudes.

Table 6. The root-mean-square error for estimated variables.

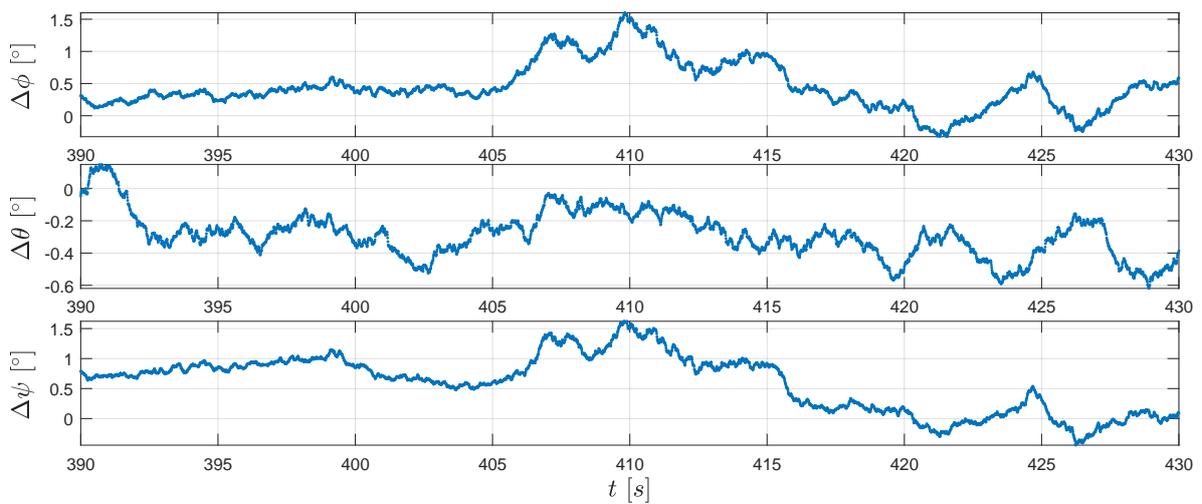
Variables	$\overline{\Delta x_N}$	$\overline{\Delta y_N}$	$\overline{\Delta z_N}$	$\overline{\Delta v_{O,x}}$	$\overline{\Delta v_{O,y}}$	$\overline{\Delta v_{O,z}}$	$\overline{\Delta\phi}$	$\overline{\Delta\theta}$	$\overline{\Delta\psi}$
RMS Error:	0.159 m	0.123 m	0.109 m	0.021 m/s	0.026 m/s	0.023 m/s	0.584°	0.319°	0.759°



(a) The position error Δr_N .



(b) The velocity error Δv_O .



(c) The attitude error in Euler angles $\Delta\phi$, $\Delta\theta$, and $\Delta\psi$.

Figure 6. The RTS smoother estimation error for the generated reference data.

4.3. Validation by Real Measurement Data

Now, we apply the extended RTS smoother on real measurement data to the ski jumping trajectory reconstruction problem. The results for a reconstructed ski jump are presented in the following part.

In Figure 7, the three-dimensional trajectories of the jumper are plotted with a digital ski jumping hill model based on [20,21]. The purple solid line with dots shows the converted raw GPS position measurements $r_{N, \text{GPS}}$ where each dot on the line represents a measurement sample. Furthermore, the line consisting of color-coded dots represents position reconstructed by the extended RTS smoother, where the color of one dot represent the velocity $V_s = \|v_O\|_2 = \sqrt{v_{O,x}^2 + v_{O,y}^2 + v_{O,z}^2}$. Note that we use the relative local height h_N here instead of z_N in the figure for convenience. The relationship between h_N and z_N is directly $h_N = -z_N$. The subplot on the top-right corner of Figure 7 presents the projection of the jumping trajectories on the x_N - z_N plane (focusing mainly on the flight part). The comparison between GPS-measured and extended RTS smoother reconstructed position as well as velocity time histories is presented in Figures 8 and 9 respectively.

Both GPS measurements and the smoother estimation in Figure 7 show the trajectory of an entire jump from start to stop. By comparing the GPS-measured trajectory to the jumping hill model for the in-run and out-run parts, it shows that the GPS height channel contains relatively large errors. After the jumper starts his run along the track during the in-run phase, the GPS-measured trajectory indicates that the jumper starts to move horizontally, but remains at the same height vertically. For part after the touch-down in the landing area, the GPS-measured trajectory shows large error leading to the position being above the ground. Therefore, in this case, the GPS-measured trajectory is not adequate for the demand for analyzing the ski jump limited to its accuracy. It is worth mentioning that the GPS measured trajectory does not necessarily encounter the same type of error in the height channel as in our case. Nevertheless, due to the trilateration positioning principle of the GPS technology employed, the GPS height measurement is far less accurate than the latitude and longitude measurements. The GPS service standard [24] reported that the GPS has a global average positioning accuracy of ≤ 9 m (95%) horizontal error and ≤ 15 m (95%) vertical error.

On the other hand, besides the GPS measurements as a data source, the extended RTS smoother uses IMU and magnetic measurements as well as geometric information of the jumping hill to perform data fusion. The application of the RTS smoother reconstructs the trajectory despite a lower GPS measurement frequency and the existence of some GPS height errors as shown in Figures 7–9. In Figure 7, the reconstructed trajectory shows the entire process of the jump. The jumper first slides down along the in-run track with increasing speed V_s . For the flight part, this can be clearly observed from both the three-dimensional plot and the subplot in Figure 7. It can be observed that the height above the ground is decreasing until the jumper's touch-down with the speed further increasing. After the touch-down, the jumper skis down on the landing area of the hill with his speed still accelerating. After reaching the flat part at the hill bottom, the jumper decelerates due to the friction and his break action and then comes to a full stop. In comparison to the GPS measurement, the estimated position and velocity are improved especially in the height channel, which reveals more information for the jump analysis, and the sampling frequency on the trajectory is also increased.

In Figure 8, we can observe that two lines fit well for x_N . For y_N in the extended RTS smoother result (the green line), the constraint $c_{H, IR}$ correct the position y_N to 0 before take-off. Afterward, the reconstructed trajectory matches the GPS measurements with some differences. For the h_N channel, as also mentioned before, the raw GPS measurement has a relatively large error which can be clearly observed in Figure 7. The smoother reconstructed height by considering the vertical constraints $c_{V, IR}$ and $c_{V, LA}$ is moving in the in-run curve before the take-off and on the surface of the landing area after the touch down (see Figure 7). The height history is closer to a real physical movement. In Figure 9, for horizontal velocity $v_{O,x}$ and $v_{O,y}$, the reconstruction and measurement fit accurately. The extended RTS smoother result gives a bit different estimation of $v_{O,z}$ compared to the GPS measured one. Considering the fact that after the jumper starts gliding down around 610.5 s, the vertical velocity afterward should be positive (moving downwards). So, the vertical speed measured by GPS also

contains an error for the in-run part which is associated with the height error. Since the extended RTS smoother considers kinetics $\dot{h}_N = -v_{O,z}$, the smoother estimated vertical speed is linked to the height estimation. Combining the result in Figures 7 and 8 for the in-run part (around 610.5 s to 616.8 s), the positive $v_{O,z}$ estimates the smoother giving out is more close to the fact that jumper is moving vertically down in the in-run. This indicates that the extended RTS smoother improves the velocity comparing to the GPS with the assistance of information from other sources.

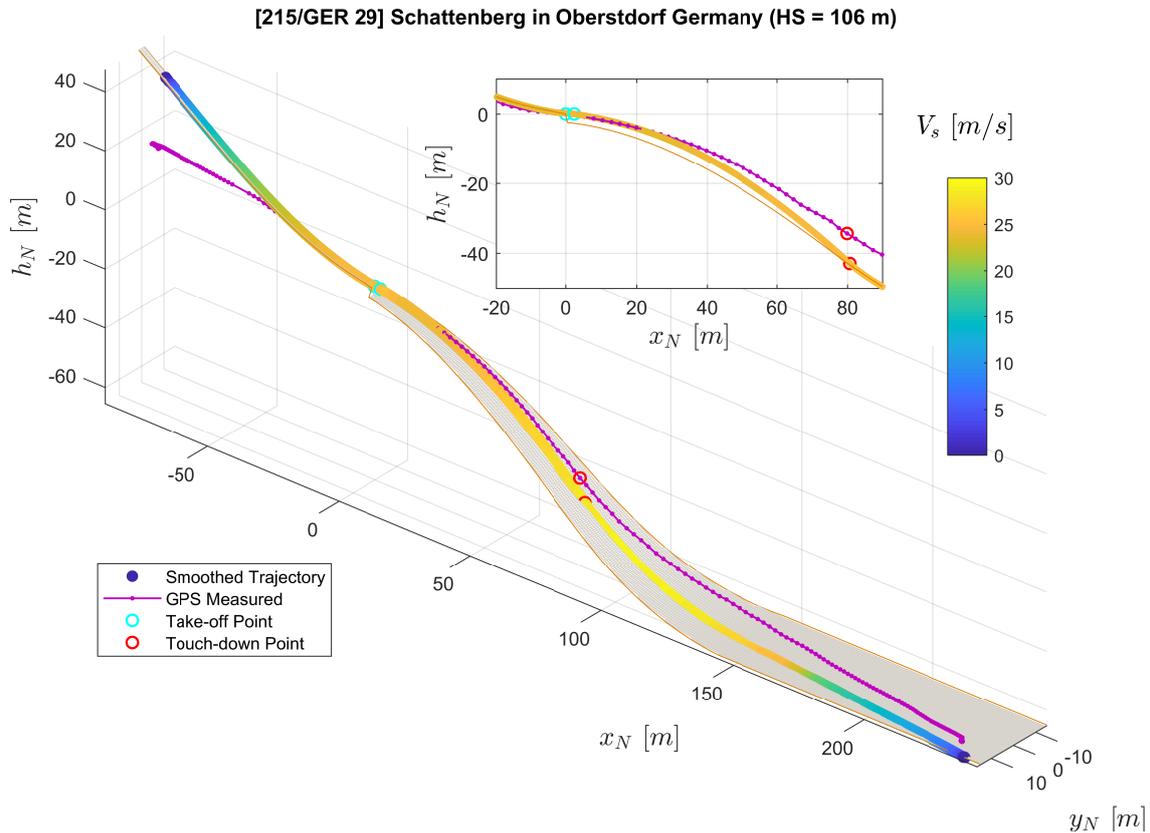


Figure 7. Comparison of the extended RTS smoother reconstructed trajectory (color-coded line) and the GPS measured trajectory (purple solid line with dots) plotted on the ski jumping hill model (orange lines and the gray surface).

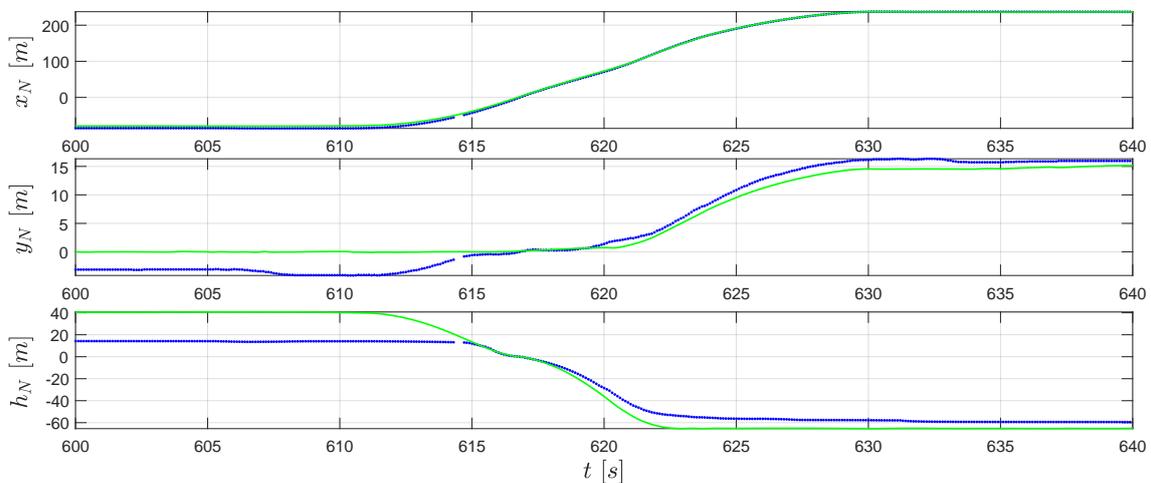


Figure 8. Converted GPS position measurements $r_{N,GPS}$ (blue dots) and the extended RTS smoother reconstructed relative positions r_N (green line).

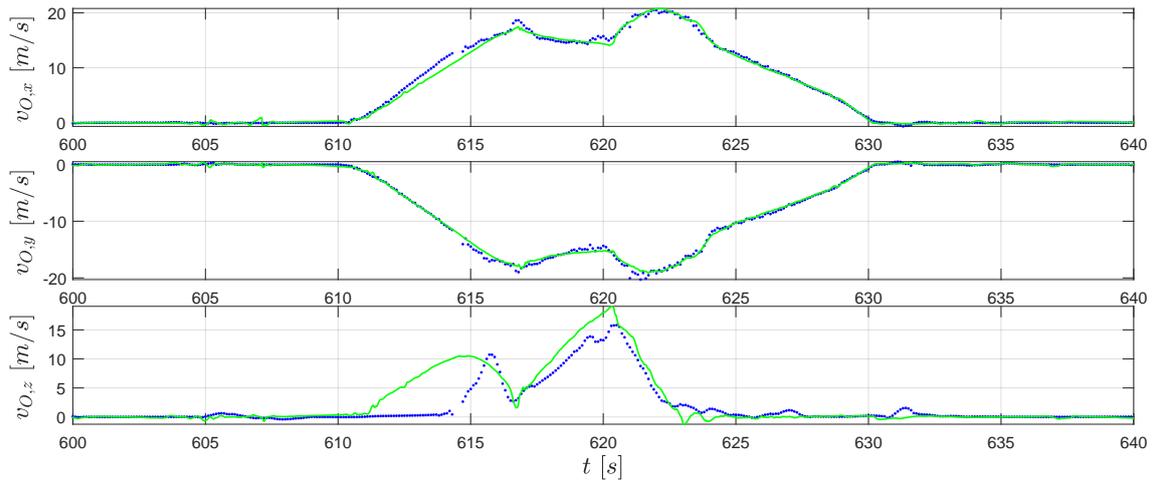


Figure 9. GPS velocity measurements $v_{O, \text{GPS}}$ (blue dots) and the extended RTS smoother reconstructed velocity v_O (green line).

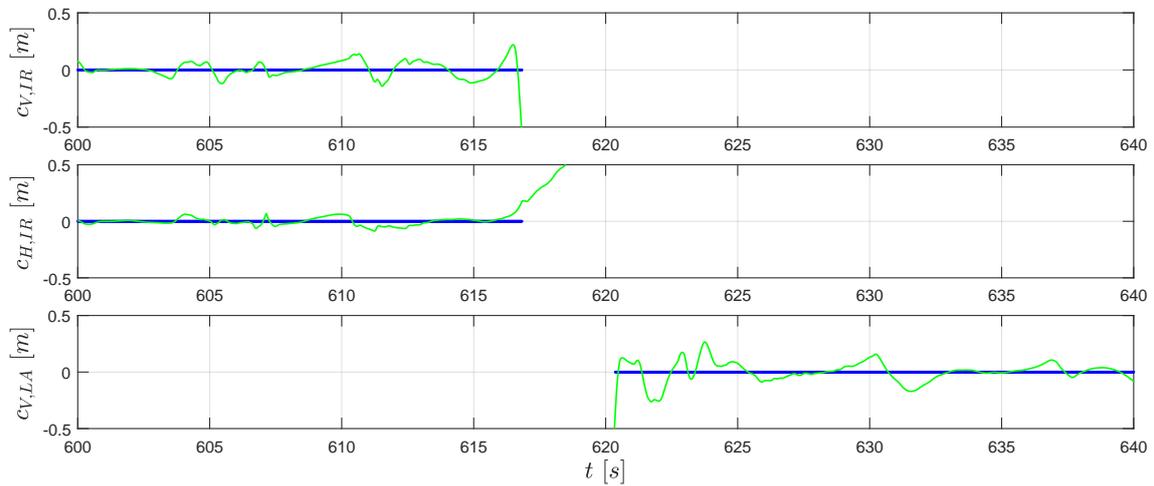


Figure 10. The extended RTS smoother estimated geometric constraints c (green line) and pseudo measurements (blue dots).

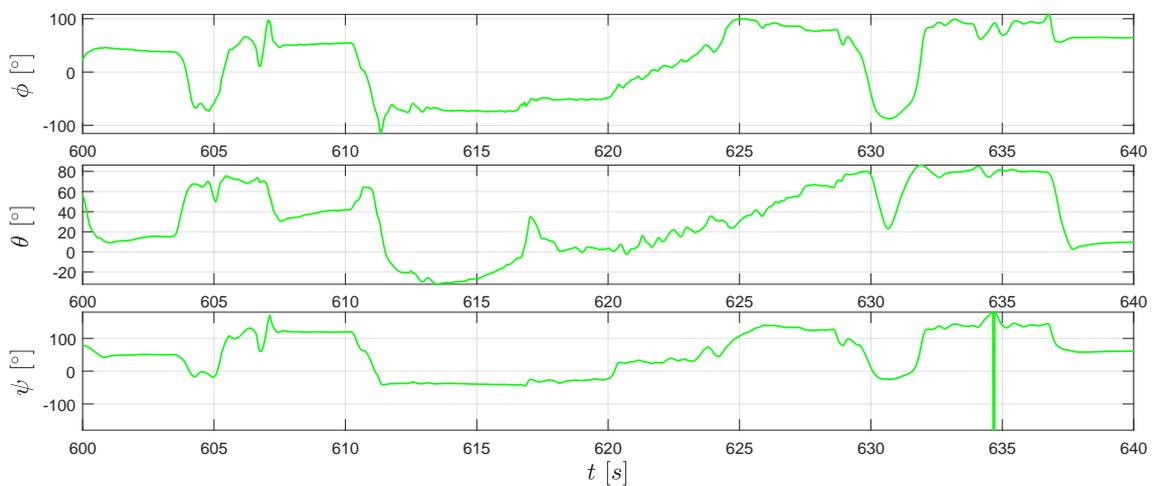


Figure 11. The extended RTS smoother estimated attitude angles: ϕ , θ , and ψ .

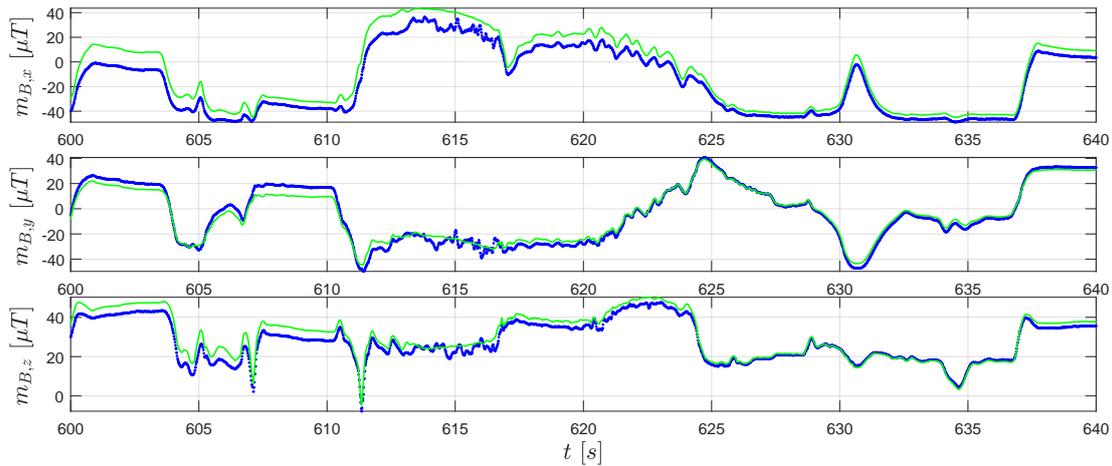


Figure 12. Real magnetometer measurements $m_{B, \text{mag}}$ (blue dots) and the extended RTS smoother estimated magnetometer measurements $m_{B, \text{meas}}$ (green line).

The improvement of the estimation accuracy is achieved by fusing data from different sources including measurement sensors, kinematics, and geometric information of the hill. The gyroscope and accelerometer data, after removing the estimated bias error, are used for propagating the states including attitude, velocity, and position together with a priori knowledge of the uncertainties. The GPS measurements and the magnetometer are included as additional data sources for the correction step. Furthermore, the introduction of the geometric constraints from the jumping hill model provides an additional information source, which contributes to the estimation. The constraints $c_{V, IR}$ and $c_{V, LA}$, as shown in Figure 10, establish a strong relationship between states x_N and z_N on the vertically plane for each part of the trajectory (before take-off and after touchdown).

Although the major aim of applying the extended RTS smoother is to obtain a better position and velocity estimation, it is worth to mention that the RTS smoother also provides an attitude estimation of the IMU sensor. Furthermore, the estimation of the IMU attitude is essential to attribute the accelerometer measured specific forces into the correct direction (see Equations (45) and (49)). By applying Equations (54)–(56), the converted Euler angles reconstructed by the extended RTS smoother are shown in Figure 11. From Equation (37), we know that the magnetic field measurements change with the attitude. Therefore, the similar tendency of extended RTS smoother estimated $m_{B, \text{meas}}$ and magnetometer measurements $m_{B, \text{mag}}$ in Figure 12 indicates good attitude estimation.

To conclude the result presentation, measurements from the gyroscope and the accelerometer are shown in Figures 13 and 14.

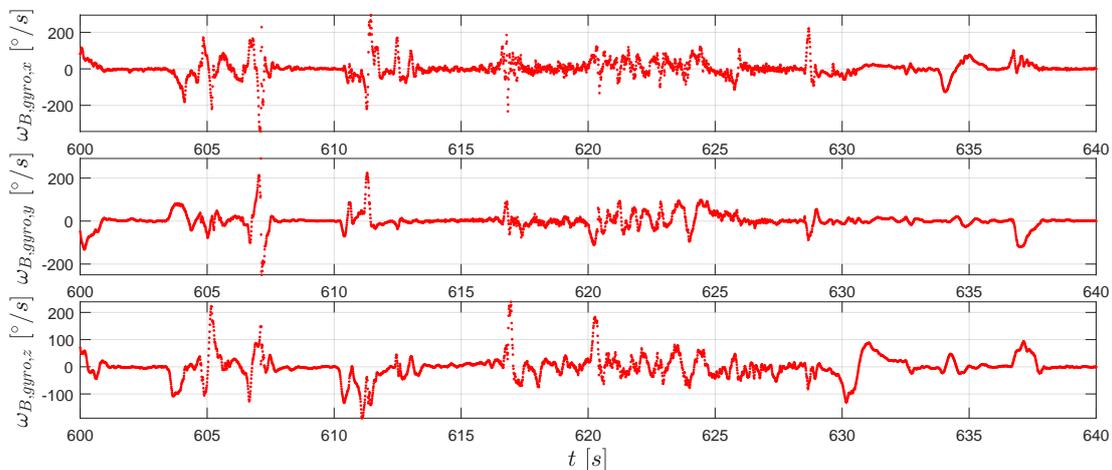


Figure 13. The gyroscope measurements $\omega_{B, \text{gyro}}$ (red dots).

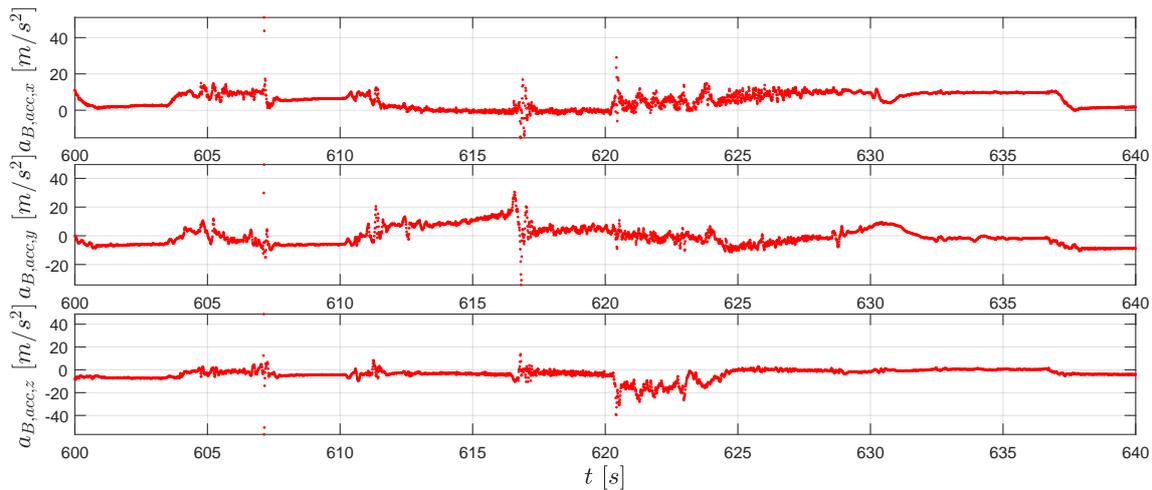


Figure 14. The accelerometer measurements $a_{B,acc}$ (red dots).

From the measurement campaign, we collected valid data sets for five different jumps in total. The reconstruction results for the rest of the four jumps are shown in Figures 15–18. Comparing to the result from Jump No. 1 (Figure 8), the results show similar characteristics as before which demonstrated the repeatability of the results.

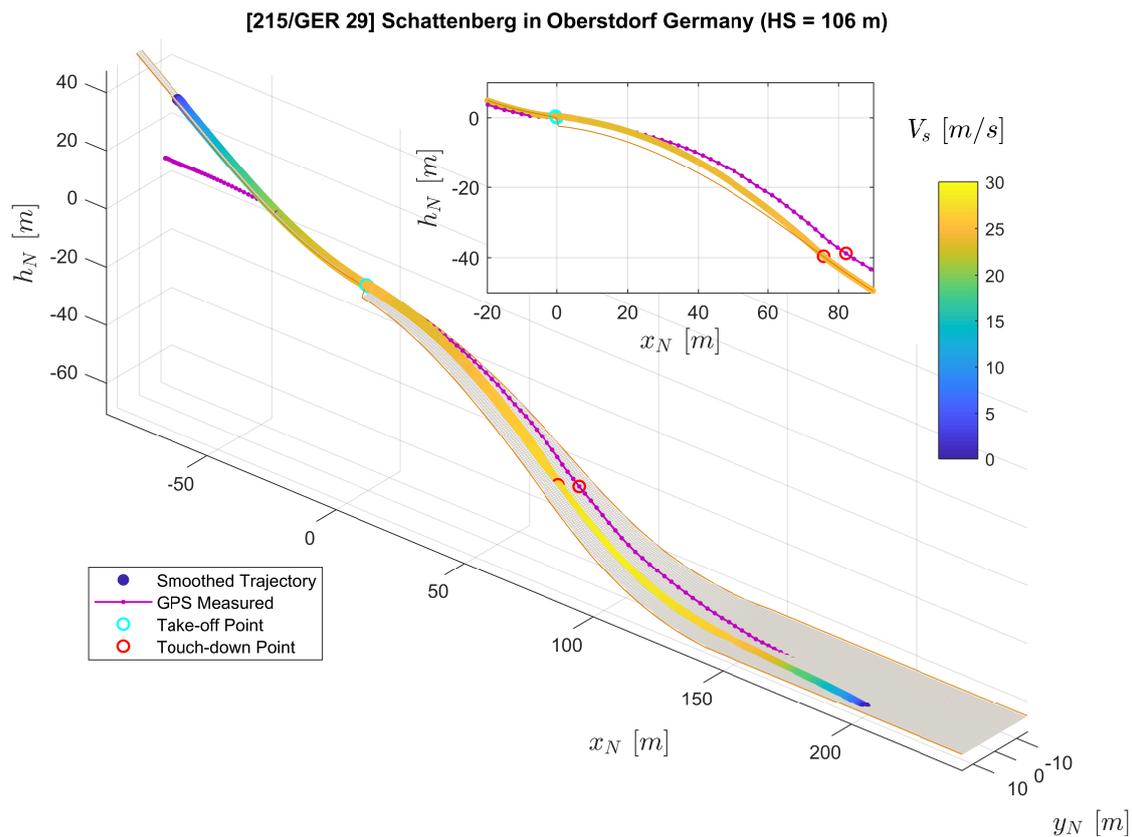


Figure 15. Trajectory Reconstruction result for Jump No. 2.

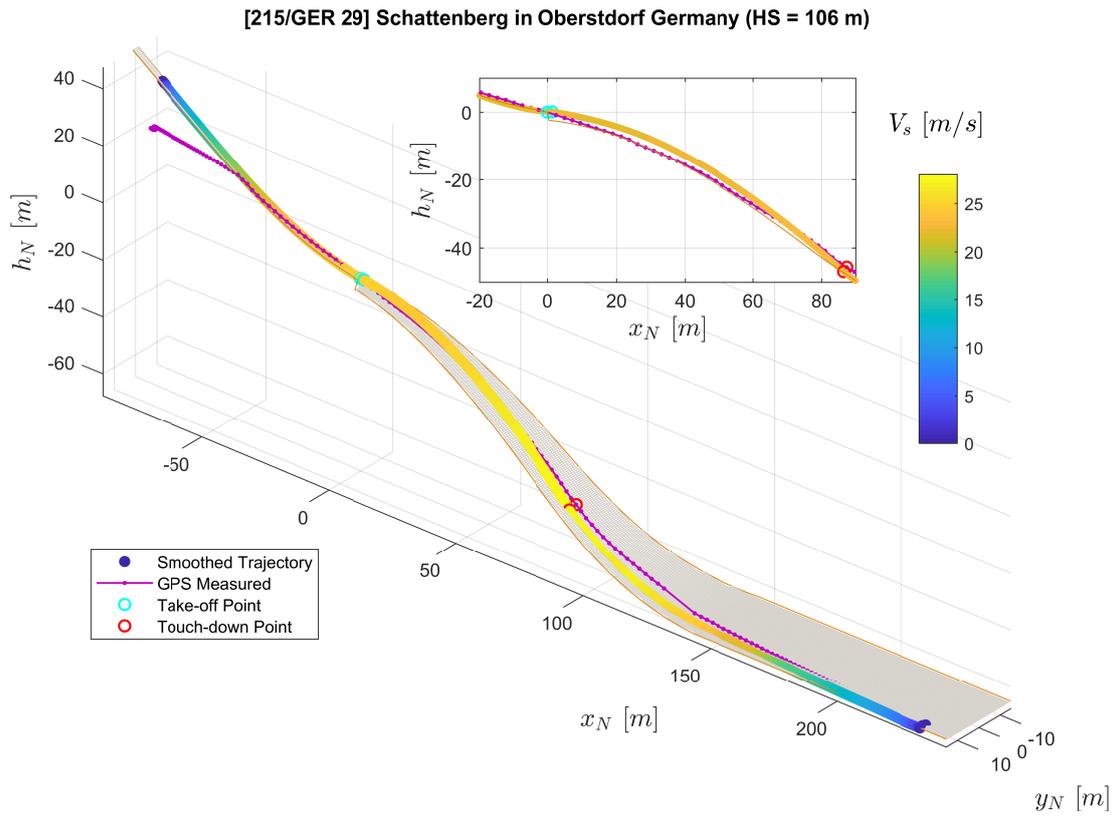


Figure 16. Trajectory Reconstruction result for Jump No. 3.

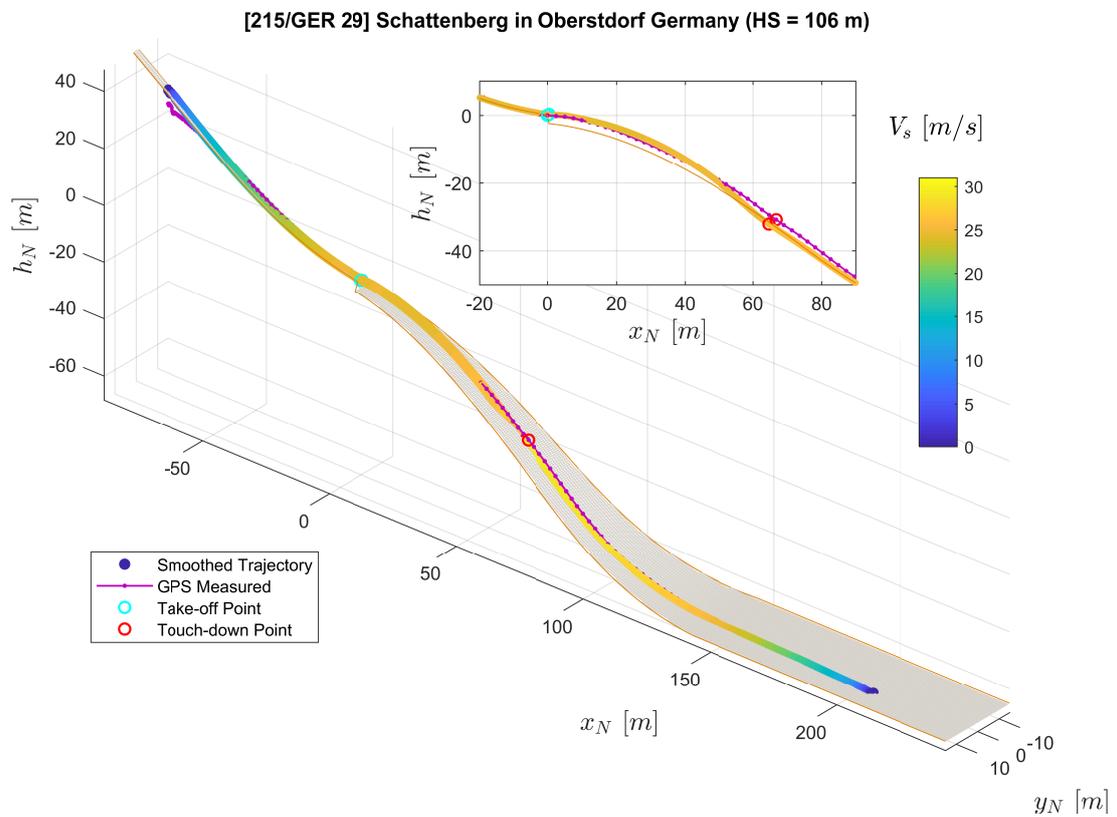


Figure 17. Trajectory Reconstruction result for Jump No. 4.

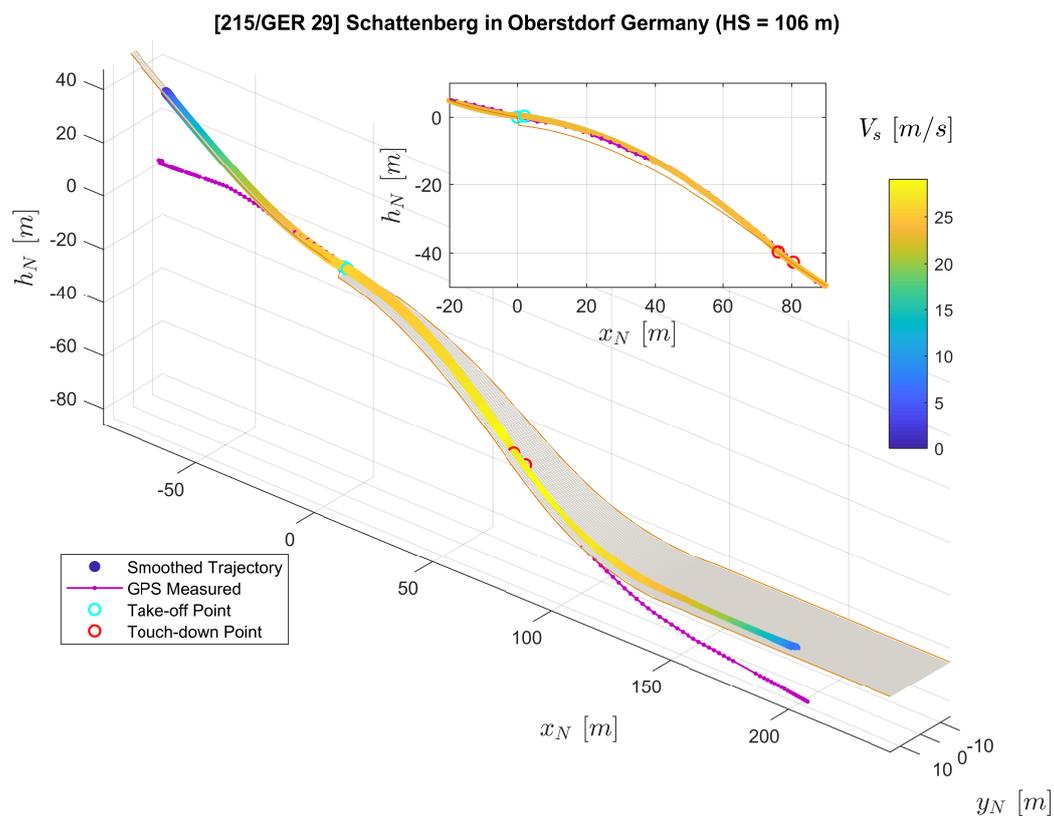


Figure 18. Trajectory Reconstruction result for Jump No. 5.

4.4. Validation by Jump Length

To validate the trajectory reconstruction results, we calculate the jump length results for the five collected trials obtained from the trajectory reconstruction L_{TR} for comparison with the jump length measurements from video recordings L_{VR} . The process of determining the jump length from the video recordings is explained in Appendix B. The validation result of five different jumps is shown in Table 7.

Table 7. Comparison between jump length obtained from the trajectory reconstruction results (proposed method) and video recordings (reference).

Jump Length:	From Trajectory Reconstruction L_{TR}	From Video Recording L_{VR}	Error ΔL
Jump No. 1	91.6 m	90.0 m	1.6 m
Jump No. 2	85.8 m	85.0 m	0.8 m
Jump No. 3	98.6 m	97.5 m	1.1 m
Jump No. 4	69.5 m	70.0 m	-0.5 m
Jump No. 5	86.0 m	85.5 m	0.5 m

The error ΔL in Table 7 is defined as $\Delta L = L_{TR} - L_{VR}$, and the RMS error for the jump length estimation is therefore $\overline{\Delta L} = \sqrt{\frac{1}{5} \sum_{i=1}^5 \Delta L^2} = 1.0$ m. Considering that the estimate of the L_{VR} value could contain an error of 0.5 m as worst case, we calculate the upper bound of the RMS error as $\overline{\Delta L}_{ub} = \sqrt{\frac{1}{5} \sum_{i=1}^5 (|\Delta L| + 0.5)^2} = 1.5$ m.

5. Conclusions and Discussion

This paper presents a novel method for trajectory reconstruction in ski jumping using wearable sensors by applying the extended RTS smoother with state constraints. The result of simulated measurement data validates the proposed method and shows that it has a theoretical accuracy of

decimeter magnitude in positions. The results based on real measurements show the method can successfully reconstruct the trajectories of ski jumps. By formulating the geometric information of the jumping hill as soft equality constraints, the estimation accuracy of positions and velocities is largely improved comparing to the GPS measurements. Jump length data for the five collected trails obtained from video recordings demonstrate good accordance with the reconstruction results with the root-mean-square error upper bound of 1.5 m. This provides strong supporting evidence for the validity of the proposed method.

For future research, a quantitative assessment of the accuracy of the proposed method by further real measurement data needs to be carried out. Therefore, another measuring system providing higher accuracy of position and velocity, e.g., a well-calibrated camera system, needs to be employed in order to compare the results of the two systems. For future device developments, wearable sensors providing better position and velocity measurement accuracy, such as differential GPS receivers, which could provide centimeter-level accuracy [6], are recommended to gain better estimation of the jumper's trajectories. In addition, GPS receivers with higher sampling frequency can be tested which could potentially improve the results. On the other hand, due to the importance of size and weight in the performance of ski jumping, the wearability of such sensors is still a major aspect to be considered. Future measurements should include more trials performed by different athletes in order to increase the statistical confidence. After further development, this technique is promising to be implemented to provide quantified motion feedback in ski jumping training.

Author Contributions: Conceptualization, X.F. and F.H.; methodology, X.F.; software, X.F.; validation, X.F. and B.G.; formal analysis, X.F.; investigation, X.F., B.G., P.P., V.B. and J.P.; resources, F.H.; data curation, X.F.; writing—original draft preparation, X.F.; writing—review and editing, B.G., P.P., V.B., J.P. and F.H.; visualization, X.F.; supervision, F.H.; project administration, B.G.; funding acquisition, F.H.

Funding: This research was supported by the Deutsche Forschungsgemeinschaft (DFG) through the TUM International Graduate School of Science and Engineering (IGSSE), GSC 81. The first author was funded by the China Scholarship Council(CSC) for his doctoral study in TUM.

Acknowledgments: The authors are thankful to the athletes who participated in the study and to the availability of the coach. The authors would also like to thank Ansgar Schwirtz and Matthias Bittner for their support in the TUM IGSSE research project for ski jumping.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the result.

Appendix A. Take-Off and Touch-Down Time Point Detection

The method implemented for detecting the time points for take-off t_{TO} and touch-down t_{TD} is introduced in this section: Due to the change of the ground reaction forces, both take-off and touch-down phase show abrupt changes in acceleration, which is measured by the accelerometer. Previous studies on ski jumping take-off [25] and landing [26,27] also presented similar acceleration changing behavior in their measurement-based results. Therefore, we propose to determine t_{TO} and t_{TD} by detecting these abrupt changes in the accelerometer measurement.

First of all, since the take-off and touchdown would happen when the jumper is moving with a high speed, we limit the analyzing time interval to the part when the GPS measured speed $V_{GPS}(t_k) = \|v_{GPS}(t_k)\|_2 = \sqrt{u_{GPS}^2(t_k) + v_{GPS}^2(t_k) + w_{GPS}^2(t_k)}$ is greater than 10 m/s. Next, within this time interval, the L2 norm of the triaxial specific force measured by accelerometer for discrete time points t_k are calculated by $a_{norm}(t_k) = \|a_{B,acc}(t_k)\|_2 = \sqrt{a_{x,acc}^2(t_k) + a_{y,acc}^2(t_k) + a_{z,acc}^2(t_k)}$ as shown

in Figure A1. Then, two peaks (local maximums) with very large a_{norm} and time difference more than 1.5 s (based on experience) are selected by the following rule as shown in Figure A1 as circles:

$$t_{\text{peak1}} = \underset{t_k}{\operatorname{argmax}}(a_{\text{norm}}(t_k) \mid V_{\text{GPS}}(t_k) > 10 \text{ m/s}), \quad (\text{A1})$$

$$t_{\text{peak2}} = \underset{t_k}{\operatorname{argmax}}(a_{\text{norm}}(t_k) \mid V_{\text{GPS}}(t_k) > 10 \text{ m/s}, -1.5 \text{ s} \leq t_k - t_{\text{peak1}} \leq 1.5 \text{ s}). \quad (\text{A2})$$

These two peaks indicate large ground reaction forces for take-off and landing. Finally, the take-off time points t_{TO} is determined as the earlier peak. As presented in [26,27], the actual touch-down point in the landing process is a short moment before the maximal ground reaction peak, approximately 0.05 s as we estimated. Therefore, the touch-down time points t_{TD} is determined as 0.05 s before the later peak as:

$$t_{\text{TO}} = \min\{t_{\text{peak1}}, t_{\text{peak2}}\}, \quad (\text{A3})$$

$$t_{\text{TD}} = \max\{t_{\text{peak1}}, t_{\text{peak2}}\} - 0.05 \text{ s}. \quad (\text{A4})$$

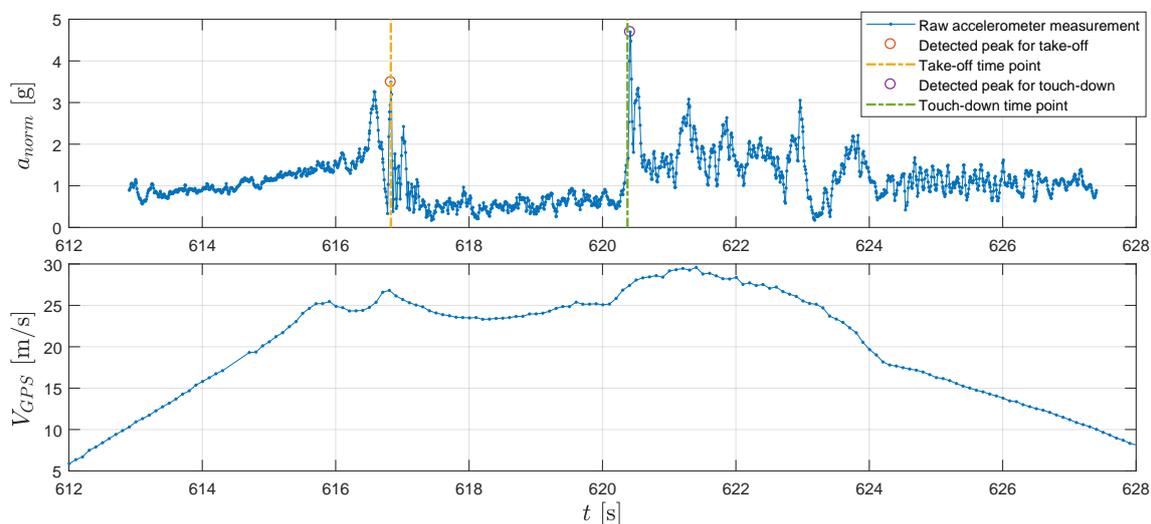


Figure A1. An illustrative example on detecting the take-off and touch-down time points by the raw measurement data.

Appendix B. Jump Length Detected by the Video Recordings

In this section, the process for detecting the jump length detected using the video recordings is explained. Cameras with a frame rate of 50 fps are set up at the side of the landing area to record the jumper's landing movement. Figure A2 shows a picture of overlaid snapshots from one of the video recordings. The time difference between each pair of neighboring snapshots is 0.1 s (with a distance of 5 frames). First, the frame for the jumper's touchdown determined indicated by both skis basically touching the ground. Then, the jump length is detected by the middle point of the ski with the help of the distance markers on the other side of the hill, as shown by the red dash lines in Figure A2. Overall, we consider the detected jump length is within an accuracy of 0.5 m.



Figure A2. A picture of overlaid snapshots from video recordings to illustrate the jump length detection from video recordings.

References

1. Virmavirta, M.; Isolehto, J.; Komi, P.; Brüggemann, G.P.; Müller, E.; Schwameder, H. Characteristics of the early flight phase in the Olympic ski jumping competition. *J. Biomech.* **2005**, *38*, 2157–2163. [[CrossRef](#)] [[PubMed](#)]
2. Groh, B.H.; Warschun, F.; Deininger, M.; Kautz, T.; Martindale, C.; Eskofier, B.M. Automated ski velocity and jump length determination in ski jumping based on unobtrusive and wearable sensors. *Proc. Acm Interact. Mob. Wearable Ubiquitous Technol.* **2017**, *1*, 53. [[CrossRef](#)]
3. Adesida, Y.; Papi, E.; McGregor, A.H. Exploring the role of wearable technology in sport kinematics and kinetics: A systematic review. *Sensors* **2019**, *19*, 1597. [[CrossRef](#)] [[PubMed](#)]
4. Chardonnes, J.; Favre, J.; Cuendet, F.; Gremion, G.; Aminian, K. Measurement of the dynamics in ski jumping using a wearable inertial sensor-based system. *J. Sport. Sci.* **2014**, *32*, 591–600. [[CrossRef](#)] [[PubMed](#)]
5. Blumenbach, T. High precision kinematic GPS positioning of ski jumpers. In Proceedings of the 17th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS 2004), Long Beach, CA, USA, 21–24 September 2004; pp. 761–765.
6. Fasel, B.; Spörri, J.; Gilgien, M.; Boffi, G.; Chardonnes, J.; Müller, E.; Aminian, K. Three-dimensional body and centre of mass kinematics in alpine ski racing using differential GNSS and inertial sensors. *Remote. Sens.* **2016**, *8*, 671. [[CrossRef](#)]
7. Mulder, J.; Chu, Q.; Sridhar, J.; Breeman, J.; Laban, M. Non-linear aircraft flight path reconstruction review and new advances. *Prog. Aerosp. Sci.* **1999**, *35*, 673–726. [[CrossRef](#)]
8. Göttlicher, C.; Holzapfel, F. Flight path reconstruction for an unmanned aerial vehicle using low-cost sensors. In Proceedings of the ICAS 30th International Congress of the International Council of the Aeronautical Sciences, Daejeon, Korea, 25–30 September 2016; pp. 25–30.
9. Rauch, H.E.; Striebel, C.; Tung, F. Maximum likelihood estimates of linear dynamic systems. *AIAA J.* **1965**, *3*, 1445–1450. [[CrossRef](#)]
10. Kalman, R.E. A new approach to linear filtering and prediction problems. *J. Basic Eng.* **1960**, *82*, 35–45. [[CrossRef](#)]
11. Simon, D. Kalman filtering with state constraints: A survey of linear and nonlinear algorithms. *IET Control. Theory Appl.* **2010**, *4*, 1303–1318. [[CrossRef](#)]
12. Staacks, S.; Hütz, S.; Heinke, H.; Stampfer, C. Advanced tools for smartphone-based experiments: Phyphox. *Phys. Educ.* **2018**, *53*, 045009. [[CrossRef](#)]
13. Qstarz BT Q1000eX GPS Data Logger Specifications. Available online: <http://www.qstarz.com/Products/GPS%20Products/BT-Q1000EX-10HZ-S.html> (accessed on 30 March 2020).

14. InvenSense ICM-20600 6-Axis IMU Datasheet. Available online: <https://www.invensense.com/download-pdf/icm-20600-datasheet/> (accessed on 30 March 2020).
15. Datasheet Alps HSCDTD008A. Available online: https://tech.alpsalpine.com/prod/c/pdf/sensor/geomagnetic/hscd/hscdtd008a_data.pdf (accessed on 30 March 2020).
16. Crassidis, J.L.; Junkins, J.L. *Optimal Estimation of Dynamic Systems*, 2nd ed.; Chapman and Hall/CRC: New York, NY, USA, 2011; ISBN 978-143-983-985-0.
17. Särkkä, S. *Bayesian Filtering and Smoothing*; Cambridge University Press: Cambridge, UK, 2013; ISBN 978-110-703-065-7.
18. Wang, L.S.; Chiang, Y.T.; Chang, F.R. Filtering method for nonlinear systems with constraints. *IEE Proc. Control. Theory Appl.* **2002**, *149*, 525–531. [[CrossRef](#)]
19. Mularie, W. *World Geodetic System 1984—Its Definition and Relationships with Local Geodetic Systems*, 3rd ed.; U.S. Department of Defense: Washington, DC, USA, 2000. Available online: <https://earth-info.nga.mil/GandG/publications/tr8350.2/wgs84fin.pdf> (accessed on 30 March 2020).
20. Gasser, H.H. *Jumping Hills Construction Norm 2018*; International Ski Federation: Oberhofen, Switzerland, 2018. Available online: https://assets.fis-ski.com/image/upload/v1542377868/fis-prod/assets/Construction-Norm_2018-2.pdf (accessed on 30 March 2020).
21. International Ski Federation. Certificate of Jumping Hill—No. 215/ GER 29. Available online: http://www.skisprungschanzen.com/photos/ger/oberstdorf_arena/HS106_2011.jpg (accessed on 30 March 2020).
22. Chulliat, A.; Macmillan, S.; Alken, P.; Beggan, C.; Nair, M.; Hamilton, B.; Wood, A.; Ridley, V.; Maus, S.; Thomson, A. *The US/UK World Magnetic Model for 2015–2020: Technical Report*; National Geophysical Data Center, NOAA: Boulder, CO, USA, 2015; [[CrossRef](#)]
23. Yun, X.; Bachmann, E.R.; McGhee, R.B. A simplified quaternion-based algorithm for orientation estimation from earth gravity and magnetic field measurements. *IEEE Trans. Instrum. Meas.* **2008**, *57*, 638–650. [[CrossRef](#)]
24. Grimes, J.G. *Global Positioning System Standard Positioning Service Performance Standard*, 4th ed.; U.S. Department of Defense: Washington, DC, USA, 2008. Available online: <https://www.gps.gov/technical/ps/2008-SPS-performance-standard.pdf> (accessed on 30 March 2020).
25. Logar, G.; Munih, M. Estimation of joint forces and moments for the in-run and take-off in ski jumping based on measurements with wearable inertial sensors. *Sensors* **2015**, *15*, 11258–11276. [[CrossRef](#)] [[PubMed](#)]
26. Bessone, V.; Petrat, J.; Schwirtz, A. Ground reaction forces and kinematics of ski jump landing using wearable sensors. *Sensors* **2019**, *19*, 2011. [[CrossRef](#)] [[PubMed](#)]
27. Groh, B.H.; Fritz, J.; Deininger, M.; Schwameder, H.; Eskofier, B.M. Unobtrusive and wearable landing momentum estimation in Ski jumping with inertial-magnetic sensors. In Proceedings of the 2018 IEEE 15th International Conference on Wearable and Implantable Body Sensor Networks (BSN), Las Vegas, NV, USA, 4–7 March 2018; pp. 102–105. [[CrossRef](#)]



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).