UAV attitude estimation based on MARG and optical flow sensors using gated recurrent unit

Xiaoqin Liu¹, Xiang Li¹✉, Qi Shi¹, Chuanpei Xu¹ and Yanmei Tang²

Abstract
Three-dimensional attitude estimation for unmanned aerial vehicles is usually based on the combination of magnetometer, accelerometer, and gyroscope (MARG). But MARG sensor can be easily affected by various disturbances, for example, vibration, external magnetic interference, and gyro drift. Optical flow sensor has the ability to extract motion information from image sequence, and thus, it is potential to augment three-dimensional attitude estimation for unmanned aerial vehicles. But the major problem is that the optical flow can be caused by both translational and rotational movements, which are difficult to be distinguished from each other. To solve the above problems, this article uses a gated recurrent unit neural network to implement data fusion for MARG and optical flow sensors, so as to enhance the accuracy of three-dimensional attitude estimation for unmanned aerial vehicles. The proposed algorithm can effectively make use of the attitude information contained in the optical flow measurements and can also achieve multi-sensor fusion for attitude estimation without explicit mathematical model. Compared with the commonly used extended Kalman filter algorithm for attitude estimation, the proposed algorithm shows higher accuracy in the flight test of quadrotor unmanned aerial vehicles.

Keywords
Unmanned aerial vehicle, attitude estimation, data fusion, optical flow sensor, magnetometer, accelerometer; gyroscope, gated recurrent unit

Date received: 10 December 2020; accepted: 11 March 2021
Handling Editor: Lyudmila Mihaylova

Introduction
Attitude estimation is the premise to ensure stable and controllable flight for unmanned aerial vehicles (UAV).¹–³ In recent years, attitude and heading reference systems (AHRS) are widely adopted in UAV flight control.⁴ The typical configuration of AHRS includes magnetometer, accelerometer, and gyroscope (MARG).⁵ With the continuous development of micro-electro-mechanical system (MEMS) technology, MARG-based AHRS provides a low-cost solution for UAV attitude estimation.

MARG-based AHRS has the advantages of low power consumption and compact size. Theories and techniques for MARG-based AHRS are also well developed. The ellipsoid fitting (also called scalar checking) and multi-position methods can help to calibrate and compensate MARG sensor errors,⁶–⁸ while Kalman filtering⁴,⁵ and complementary filtering algorithms⁹ can be used to implement data fusion for
MARG sensor. But still, the dynamic accuracy, robustness, and redundancy of AHRS needs further enhancement, since MARG-based three-dimensional (3D) attitude estimation is sensitive to external disturbances, for example, motional acceleration and magnetic interference.

Optical flow sensor can detect the movements of its carrier by processing the sequence of images and extract the differences between every two adjacent frames. At present, optical flow sensor has also been applied to UAV flight control. With the aid of optical flow sensor, UAV can achieve tracking flight and spot hover with higher accuracy. Besides, optical flow measurement can also be used for autonomous obstacle avoidance of UAV. Moreover, optical flow measurement also contains attitude information. Nevertheless, the attitude information contained in optical flow measurement is mixed with the translational movements in most cases, and thus, it is not easy to be fully utilized.

Therefore, when using optical flow sensor to augment MARG-based attitude estimation, one of the key issues is to distinguish the angular motion and translational motion that contained in optical flow measurement from each other. In most applications of optical flow sensors on UAV, the angular motion component of optical flow is eliminated with the aid of attitude information provided by IMU, and thus, only the translational motion component of optical flow is utilized. However, to make use of the attitude information contained in optical flow, the translational motion component of optical flow must be compensated using additional position and/or velocity information, for example, position/velocity provided by global navigation satellite system (GNSS) or inertial navigation system (INS).

There are three main purposes of the research presented in this paper: (a) to extract the attitude information contained in optical flow measurement; (b) to implement multi-sensor fusion, including MARG and optical flow sensors; and (c) to compensate sensor errors that probably exists. To achieve the above three goals, a gated recurrent unit (GRU) neural network is used for data fusion between MARG and optical flow sensors, in order to improve the performance of AHRS under dynamic conditions. As a special branch of artificial neural network, GRU is able to draw time-related information from time series signals, and thus, it can help to extract attitude information from optical flow measurement, without the need of complex mathematical models and additional position/velocity information.

The rest of this article is organized as follows. In section “Theoretical basis,” the theoretical basis will be briefly reviewed, including the principles of attitude estimation, data fusion, and optical flow sensor. Then in section “Attitude estimation algorithm based on GRU,” the design of GRU-based attitude estimation algorithm will be introduced, which incorporates MARG and optical flow sensors. In section “Experiment,” the proposed GRU-based algorithm will be evaluated through experiments on a quad-rotor UAV, and its performance will be compared with that of extended Kalman filter (EKF). After discussions of the experiment results in section “Discussion”, conclusions will be drawn in the sixth section.

Theoretical basis

Attitude representation and estimation

The 3D attitude of UAV can be defined by the relationship between the body frame (coordinate system that fixed to UAV) and the reference frame (coordinate system that used as the reference for navigation, e.g. North-East-Down reference frame). The transformation from one coordinate system to another can be decomposed into three consecutive rotations around the three coordinate axes, namely the Euler angles. In the case of UAV, these three Euler angles are called yaw (or heading), pitch, and roll, respectively. Besides, quaternion and direction cosine matrix (DCM) are also commonly used for UAV attitude representation.

In traditional INS, especially the strap-down INS (SINS), the 3D attitude are calculated according to the angular velocity measured by gyroscope, and thus, the long-term attitude accuracy is affected by gyro drift. In MARG-based AHRS, however, 3D attitude estimation mainly relies on the gravity vector \( g \) and geomagnetic vector \( h \), which can be obtained from the accelerometer and magnetometer, respectively. The gravity vector \( g \) is vertically downward and can be used to determine the tilt angles (i.e. pitch and roll). However, the horizontal component of geomagnetic vector \( h \) points to the magnetic north, and thus, it can help to determine the yaw angle. In a word, these two vectors can be directly used to calculate the attitude, and thus, accumulative attitude error caused by gyro drift can be avoided. Besides, the measurement of angular velocity provided by the gyroscope can be used to calculate the change of 3D attitude and thereby augmenting the attitude estimation. Therefore, the measurements of MARG sensor are sufficient for 3D attitude estimation, with appropriate data fusion algorithm.

Kalman filter (KF) is widely used for data fusion in MARG-based AHRS. It is worth mentioning that the mathematical relationship between attitude quaternion and vector observations (i.e. the gravity vector \( g \) and geomagnetic vector \( h \)) is nonlinear. As a result, nonlinear variants of KF are needed for MARG-based AHRS, such as the extended Kalman filter (EKF).
and unscented Kalman filter (UKF). EKF is based on the linearization of the system model using the first-order derivatives, while UKF makes use of unscented transform to approximate the nonlinear system. So far, EKF are more commonly used for UAV attitude estimation, mainly because its lower computational burden compared to UKF.

**Optical flow sensor**

Optical flow is the instantaneous velocity of the pixel motion on the imaging plane, which is caused by a spatially moving object. It can be calculated according to the correlation between the previous frame and the current frame in the image sequence. In another word, it contains the motion information of the object between every two adjacent frames. Based on the above principle, optical flow sensor can detect the movement of the UAV relative to the ground, as shown in Figure 1.

As shown in Figure 1(a) and (b), both the horizontal speed and rotation can generate corresponding optical flow. Therefore, in the case that optical flow sensor is attached to UAV and pointed to the ground, as shown in Figure 1(c), the optical flow measurement \( \delta \) can be described by equation (1). In equation (1), \( v \) is the horizontal velocity of the UAV, \( l \) is the height of the optical flow sensor above the ground, \( \gamma \) is the angle between the optical axis and the vertical direction, \( \omega \) is the rotation speed of the optical flow sensor, and \( f \) is the scale factor

\[
\delta = f \left( \frac{vcos^2\gamma}{l} + \omega \right)
\]  

(1)

According to equation (1), the optical flow measurement \( \delta \) is related to both the translational and rotational movements. Therefore, if the optical flow sensor is used to monitor ground speed, the optical flow component caused by rotation must be compensated, and vice versa.12,19

**Attitude estimation algorithm based on GRU**

**GRU neural network for attitude estimation**

GRU is a special version of recurrent neural network (RNN). RNN is a type of neural network that used to process sequence data. To better process sequence information, RNN has local feedback loop in its hidden layer, in order to recognize the relationship between the current input and the previous ones. GRU also has local feedback, but the feedback is controlled by two gates, namely the reset gate and the update gate. In other words, GRU can be viewed as a gated RNN. GRU has achieved great success in natural language processing and speech conversion. The schematic diagram of GRU-based attitude estimation is shown in Figure 2.

As shown in Figure 2, the GRU-based neural network takes the measurements from four different sensors as its inputs, namely the gravity vector \( g = (g_x, g_y, g_z) \) from accelerometer, the geomagnetic vector \( h = (h_x, h_y, h_z) \) from magnetometer, the angular velocity \( \omega = (\omega_x, \omega_y, \omega_z) \) from gyroscope, and the two-dimensional (2D) optical flow \( \delta = (\delta_x, \delta_y) \) from optical flow sensor. At the \( k \)th time step, all the above inputs are collectively denotes as \( x_k \). However, the output \( y_k \) consists of the estimated yaw, pitch, and roll angles. Besides, \( z_k \) is the hidden state of GRU, while the reset gate and update gate are denoted as \( f_k \) and \( g_k \), respectively.

![Figure 1. Geometric relationship of optical flow sensor: (a) optical flow caused by horizontal velocity, (b) optical flow caused by rotation, and (c) optical flow sensor on UAV with both translational and rotational movements.](image-url)
At the kth time step, the reset gate $f_k$ and update gate $g_k$ are calculated according to the input $x_k$ and the previous hidden state $z_{k-1}$, as described by equations (2) and (3), in which $\sigma(\cdot)$ denotes the activation function

$$
\begin{align*}
    f_k &= \sigma(U_f x_k + V_f z_{k-1} + b_f) \\
    g_k &= \sigma(U_g x_k + V_g z_{k-1} + b_g)
\end{align*}
$$

Then, the hidden state $z_k$ can be updated using equations (4) and (5)

$$
\begin{align*}
    \tilde{z}_k &= \sigma(U_z x_k + f_k z_{k-1} + b_z) \\
    z_k &= g_k \tilde{z}_k + (1 - g_k) z_{k-1}
\end{align*}
$$

Furthermore, the hidden state $z_k$ is transformed into the output $y_k$ through the output layer, which can be described by equation (6)

$$
y_k = Wz_k + d
$$

As can be seen in equation (5), the update gate works as a complementary multiplexer, in which larger value of $g_k$ will bring more information from the input $x_k$ to the current hidden state $z_k$. However, in the reset gate, smaller value of $f_k$ means that $\tilde{z}_k$ will further ignore the previous hidden state $z_{k-1}$, as indicated by equation (4). These two gates can help to handle long-term correlations in time domain, as well as the gradient vanishing problem in RNN training.

In most studies of GRU, the sigmoid function is used for the activation function in equations (2) and (3), which is defined as $\sigma(x) = 1/(1 + e^{-x})$, while hyperbolic tangent (tanh) function is used for the activation function in equation (4). But in order to reduce the computational complexity, in the following experiment, the “softsign” function is used to replace tanh function in equation (4), which is defined as $\sigma(x) = x/(1 + |x|)$. Meanwhile, the sigmoid function is also replaced by a biased version of softsign function for equations (2) and (3), which is defined as $\sigma(x) = 1 + 0.5x/(1 + |x|)$. Since the softsign function contains no exponential or trigonometric functions, it can help to promote the processing efficiency on hardware platforms with limited computing power, such as a micro-controller.

As stated above, GRU can deal with time series signals, and it is also able to approximate nonlinear functions. Therefore, it can be used to solve the problem of data fusion between MARG and optical flow sensors for UAV attitude estimation. After sufficient training, GRU neural network can effectively utilize the measurements of MARG and optical flow sensors for attitude estimation, without the need of rigorous mathematical modeling.

**GRU neural network training**

The training process of GRU network is to determine the parameters in both the hidden layer and the output layer, including $U_f$, $U_g$, $U_z$, $V_f$, $V_g$, $b_f$, $b_g$, $b_z$, $W$, and $d$ in Figure 2.

First of all, the cost function for GRU training is defined as equation (7), in which $y_{r,k}$ denotes the target output of GRU network (i.e. the true value of yaw, pitch, and roll) at the kth time step

$$
y_k = Wz_k + d
$$

As stated above, GRU can deal with time series signals, and it is also able to approximate nonlinear functions. Therefore, it can be used to solve the problem of data fusion between MARG and optical flow sensors for UAV attitude estimation. After sufficient training, GRU neural network can effectively utilize the measurements of MARG and optical flow sensors for attitude estimation, without the need of rigorous mathematical modeling.

**Figure 2.** Schematic diagram of GRU neural network. $U$, $V$, and $W$ are the weights, while $b$ and $d$ are the biases. These parameters will be determined during the training of GRU.
\[ e = \begin{pmatrix} y^T_1 - y_{r,1}^T \\ y^T_2 - y_{r,2}^T \\ \vdots \end{pmatrix} \]  
(7)

Meanwhile, all the parameters to be estimated can be rearranged as a column vector \( \theta \), so the training process can be described as an optimization problem in equation (8)

\[ \hat{\theta} = \arg \min_{\theta} |e|^2 \]  
(8)

Then, a hybrid algorithm is used for the optimization problem in equation (8), which is a combination of the commonly used Levenberg–Marquardt (L-M) method and extreme learning technique. The L-M method relies on the Jacobian matrix \( J \) for all the parameters to be estimated, and the Jacobian matrix \( J \) can be constructed according to the following partial derivatives.

(a) Derivatives for \( W \) and \( d \) in the output layer (\( e_k \) means the \( k \)th row of the cost function \( e \))

\[ \frac{\partial e_k}{\partial d} = 2(y_k - y_{r,k}) \]  
(9)

\[ \frac{\partial e_k}{\partial W} = \frac{\partial e_k}{\partial d} z_k^T \]  
(10)

(b) Derivatives for \( U_z \) and \( b_z \), with \( \sigma'(\cdot) \) and \( \otimes \) indicating the derivative of \( \sigma(\cdot) \) and Hadamard product (i.e. element-wise product of matrices), respectively

\[ \frac{\partial e_k}{\partial z_k} = W^T \frac{\partial e_k}{\partial d} \]  
(11)

\[ \frac{\partial e_k}{\partial z_k} = \frac{\partial e_k}{\partial d} \otimes g_k \]  
(12)

\[ \frac{\partial e_k}{\partial b_z} = \frac{\partial e_k}{\partial z_k} \otimes \sigma'(U_z x_k + f_k \otimes z_{k-1} + b_z) \]  
(13)

\[ \frac{\partial e_k}{\partial U_z} = \frac{\partial e_k}{\partial b_z} x_k^T \]  
(14)

(c) Derivatives for \( U_g, V_g, \) and \( b_g \) in the update gate

\[ \frac{\partial e_k}{\partial g_k} = \frac{\partial e_k}{\partial z_k} \otimes (z_k - z_{k-1})^T \]  
(15)

\[ \frac{\partial e_k}{\partial b_g} = \frac{\partial e_k}{\partial g_k} \otimes \sigma'(U_g x_k + V_g z_{k-1} + b_g) \]  
(16)

\[ \frac{\partial e_k}{\partial U_g} = \frac{\partial e_k}{\partial b_g} x_k^T \]  
(17)

\[ \frac{\partial e_k}{\partial V_g} = \frac{\partial e_k}{\partial b_g} z_{k-1}^T \]  
(18)

(d) Derivatives for \( U_f, V_f, \) and \( b_f \) in the update gate

\[ \frac{\partial e_k}{\partial f_k} = \frac{\partial e_k}{\partial z_k} \otimes z_{k-1} \]  
(19)

\[ \frac{\partial e_k}{\partial b_f} = \frac{\partial e_k}{\partial f_k} \otimes \sigma'(U_f x_k + V_f z_{k-1} + b_f) \]  
(20)

\[ \frac{\partial e_k}{\partial U_f} = \frac{\partial e_k}{\partial b_f} x_k^T \]  
(21)

\[ \frac{\partial e_k}{\partial V_f} = \frac{\partial e_k}{\partial b_f} z_{k-1}^T \]  
(22)

It is noteworthy that the relationship between \( z_k \) and \( y_k \) is linear, as shown in equation (6). Therefore, \( W \) and \( d \) can be determined directly via least square fitting, and this is the essence of extreme learning. The hybrid training algorithm can be summarized as follows:

1. Initialization: iteration number \( i = 0 \), damping factor \( \mu_i = 0.01 \), all the parameters \( (U_f, U_g, U_z, V_f, V_g, b_f, b_g, b_z) \) randomly chosen in \([-1, 1]\).
2. Construct vector \( \theta_i \) according to \( U_f, U_g, U_z, V_f, V_g, b_f, b_g, \) and \( b_z \).
3. Calculate the hidden state series \( \{z_k\} \), then solve \( W \) and \( d \) according to the target output series \( \{y_{r,k}\} \) via direct least square fitting.
4. Calculate the output series \( \{y_k\} \), the cost function \( e_i \), and the Jacobian matrix \( J = \partial e_i / \partial \theta_i \).
5. Update the parameter vector \( \theta_{i + 1} = \theta_i - (J_i^T J_i + \mu_i I_p \times p)^{-1} J_i^T e_i \), in which \( I_p \times p \) is an identity matrix, and \( p \) is the number of parameters in \( \theta_i \).
6. Update the hidden state series \( \{z_k\} \) according to the parameters in \( \theta_{i + 1} \), then solve \( W \) and \( d \) again.
7. Update the output series \( \{y_k\} \) and the cost function \( e_{i + 1} \).
8. If \( |e_{i + 1}|^2 < |e_i|^2 \), accept the updated parameters and let \( \mu_{i + 1} = 0.1 \mu_i \); otherwise, rollback all the parameters and let \( \mu_{i + 1} = 10 \mu_i \).
9. If \( |e_i|^2 - |e_{i + 1}|^2 < 10^{-4} |e_i|^2 \), or \( i > 5000 \), output all the parameters and then finish; otherwise \( i = i + 1 \) and go to 2).

**Experiment**

**Raw data acquisition**

PX4FLOW optical flow sensor is used in the experimental evaluation of the proposed algorithm, accompanied with PIXHAWK2.4.8 autopilot system. PX4FLOW has the original resolution of 752 × 480 pixels, and it uses a fourfold grading and clipping algorithm to calculate the optical flow. This sensor is connected to PIXHAWK2.4.8 autopilot and attached to the quadcopter platform. Meanwhile, the quadcopter platform also
integrates magnetometer, accelerometer, and gyroscope. After debugging, the quadcopter can fly stably in fixed height and hover modes, as shown in Figure 3. The PIXHAWK platform is equipped with 168 MHz Cortex M4 F CPU to process flight data. All the raw data, including the measurements from MARG and optical flow sensors, are recorded in the flight log.

Four datasets are collected using the apparatus shown in Figure 3, and they are denoted as Dataset 1, 2, 3, and 4, respectively. Figure 4 shows the trajectories of UAV during acquiring these four datasets. Dataset 1 is used for GRU training, while the other three datasets are used to evaluate the performance of GRU-based attitude estimation and compare it to EKF. Each dataset contains the raw data from magnetometer, accelerometer, gyroscope, and optical flow sensor, at the output rate of 50 Hz. Figure 5 illustrates the data in Dataset 1.

**GRU training**

The GRU neural network is trained using Dataset 1 with different numbers of hidden layer dimensions. The training results are evaluated in terms of root mean square error (RMSE), which can be calculated by equation (23)

\[
RMSE = \sqrt{\frac{1}{N} \sum_{k=1}^{N} (y_k - y_{r,k})^2}
\]

In equation (23), \(y_k\) is the output of trained GRU network, which contains yaw, pitch, and roll angles. Meanwhile, \(y_{r,k}\) is the corresponding true values that are based on the fusion of GPS/barometer/MARG sensor, which can be read from the flight log of PIXHAWK autopilot.

Figure 6 shows the RMSE and training time of GRU versus different hidden layer dimensions. The training is carried out on Intel Core i5-4590 CPU at 3.50 GHz with 8 GB RAM. It can be seen from Figure 6 that the relationship between GRU training time and hidden layer dimension is almost linear. Besides, the best performance can be achieved when hidden layer dimension is set to 15 (i.e. 15 GRUs in the hidden layer). The RMSE of roll, pitch, and yaw angles are 1.51°, 2.07°, and 0.87°, respectively. Furthermore, Figure 7 shows the estimated 3D attitude compared with the true values (including roll, pitch, and yaw)
when the dimension of hidden layer is 15. Compared to the attitude algorithms based on Elman neural network\textsuperscript{31,32}, the GRU-based attitude estimation method proposed in this article shows higher accuracy. Therefore, it is a feasible approach for data fusion.

**GRU validating**

After GRU training, the performance of GRU-based attitude estimation is further evaluated using Datasets 2, 3, and 4. Figure 8 shows the 3D attitude estimated by the proposed GRU-based algorithm in terms of Euler angles, as well as that of traditional EKF algorithm, compared to the ground truth from PIXHAWK flight logs. The root mean square error (RMSE) is still used to evaluate the overall performance for both GRU and EKF, and the results are listed in Table 1. It is worth mentioning that the proposed GRU-based algorithm is evaluated in two cases, that is, working with and without optical flow sensor, while EKF performs typical MARG-based attitude estimation and does not make use of optical flow sensor. According to the results in Table 1, the performance of GRU-based algorithm is better than that of EKF, especially when incorporating optical flow sensor.

To compare the computational complexity of GRU and EKF, the average execution time of both algorithms on STM32F103 micro-controller is listed in Table 2. The hidden layer dimension of GRU is still set to 15. It can be seen that GRU-based algorithm needs shorter execution time than EKF, either with or without the aid of optical flow.

**Discussion**

The experimental results evidently prove the feasibility of the proposed GRU-based attitude estimation algorithm. According to Figures 7 and 8, the proposed algorithm can effectively solve the problems of sensor data fusion and error compensation in attitude estimation of the quadcopter.

Furthermore, as can be seen in Table 1, the performance of the proposed GRU-based algorithm is comparable to that of EKF, in the case that optical flow sensor is unused. When the measurement of optical flow sensor is adopted, GRU-based algorithm can achieve higher attitude accuracy. Consequently, the
Figure 7. GRU-based attitude estimation results of Dataset 1 after training: (a) roll angle, (b) pitch angle, and (c) yaw angle.

Figure 8. Attitude estimation of different algorithms in the flight test: (a) Dataset 2, roll angle; (b) Dataset 2, pitch angle; (c) Dataset 2, roll angle; (d) Dataset 3, roll angle; (e) Dataset 3, pitch angle; (f) Dataset 3, yaw angle; (g) Dataset 4, roll angle; (h) Dataset 4, pitch angle; and (i) Dataset 4, yaw angle.
proposed GRU-based algorithm is able to make better use of the 3D attitude information in the measurements of MARG and optical flow sensors.

It is worth mentioning again that the proposed algorithm can implement data fusion for MARG and optical flow sensors, without considering explicit mathematical models of these sensors. It is the very point that the proposed algorithm differs essentially from the commonly used KF-based data fusion technique.

**Conclusion**

In this article, a 3D attitude estimation algorithm based on GRU network is proposed, which make use of MARG and optical flow sensors. Experimental results on a quadcopter show that the proposed algorithm can provide reliable attitude estimation according to the measurements of MARG and optical flow sensor. Compared with commonly used EKF-based attitude estimation, the proposed algorithm can further enhance 3D attitude accuracy by extracting the attitude information from the optical flow measurement, with no need to redesign the attitude algorithm according to complex mathematical equations. Therefore, it is a novel and feasible approach for the development of AHRS and UAV flight control.

**Declaration of conflicting interests**

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

**Funding**

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: The work reported in this article was funded by the National Natural Science Foundation of China (grant numbers 61603107 and 41761087).

**ORCID iD**

Xiang Li https://orcid.org/0000-0001-9764-1290

**References**

1. Yu Y, Zhang X and Khan MSA. Attitude heading reference algorithm based on transformed cubature Kalman filter. *Meas Control* 2020; 53: 1446–1453.
2. Zhang T and Liao Y. Attitude measure system based on extended Kalman filter for multi-rotors. *Comput Electron Agric* 2017; 134: 19–26.
3. Kada B, Munawar K, Shaikh MS, et al. UAV attitude estimation using nonlinear filtering and low-cost MEMS sensors. *IFAC-Papersonline* 2016; 49: 521–528.
4. Jurman D, Jankovec M, Kamnik R, et al. Calibration and data fusion solution for the miniature attitude and heading reference system. *Sens Actuators A Phys* 2007; 138: 411–420.
5. Valenti RG, Dryanovski I and Xiao J. A linear Kalman filter for MARG orientation estimation using the algebraic quaternion algorithm. *IEEE Trans Instrum Meas* 2016; 65: 467–481.
6. Sipos M, Paces P, Rohac J, et al. Analyses of triaxial accelerometer calibration algorithms. *IEEE Sens J* 2012; 12: 1157–1165.
7. Hemerly EM and Coelho FAA. Explicit solution for magnetometer calibration. *IEEE Trans Instrum Meas* 2014; 63: 2093–2095.

**Table 1.** RMSE of GRU- and EKF-based attitude algorithms.

| Dataset 2       | EKF  | GRU without optical flow | GRU with optical flow |
|-----------------|------|--------------------------|-----------------------|
| RMSE-yaw        | 4.47°| 1.55°                    | 1.32°                 |
| RMSE-pitch      | 2.54°| 2.81°                    | 2.41°                 |
| RMSE-roll       | 2.76°| 2.31°                    | 1.94°                 |

| Dataset 3       | RMSE-yaw | 3.67° | 3.33° | 3.27° |
|-----------------|-----------|-------|-------|-------|
| RMSE-pitch      | 5.36°     | 3.18° | 3.07° |       |
| RMSE-roll       | 3.94°     | 1.96° | 1.81° |       |

| Dataset 4       | RMSE-yaw | 3.56° | 2.91° | 2.33° |
|-----------------|-----------|-------|-------|-------|
| RMSE-pitch      | 4.47°     | 3.67° | 2.70° |       |
| RMSE-roll       | 3.26°     | 1.89° | 1.67° |       |

**Table 2.** Execution time of GRU and EKF on micro-controller.

|                   | EKF       | GRU without optical flow | GRU with optical flow |
|-------------------|-----------|--------------------------|-----------------------|
| Average execution time on STM32F103 at 72 MHz | 4.80 ms | 2.44 ms | 2.74 ms |
8. Sarkka O, Nieminen T, Suuriniemi S, et al. A multi-position calibration method for consumer-grade accelerometers, gyroscopes, and magnetometers to field conditions. *IEEE Sens J* 2017; 17: 3470–3481.

9. Mahony R, Hamel T and Pflimlin JM. Nonlinear complementary filters on the special orthogonal group. *IEEE Trans Autom Control* 2008; 53: 1203–1218.

10. Shabayek AER, Demonceaux C, Morel O, et al. Vision based UAV attitude estimation: progress and insights. *J. Intell Rob Syst Theor Appl* 2012; 65: 295–308.

11. Herissé B, Hamel T, Mahony R, et al. Landing a VTOL unmanned aerial vehicle in a moving platform using optical flow. *IEEE Trans Rob* 2012; 28: 77–89.

12. Zhang L, Xiong Z, Lai J, et al. Research of optical flow aided MEMS navigation based on convex optimization and ROF denoising. *Optik* 2018; 158: 1575–1583.

13. Ashraf S, Aggarwal P, Damacharla P, et al. A low-cost solution for unmanned aerial vehicle navigation in a global positioning system–denied environment. *Int J Distrib Sens Netw* 2018; 14: 1550147718781750.

14. Gageik N, Benz P and Montenegro S. Obstacle detection and collision avoidance for a UAV with complementary low-cost sensors. *IEEE Access* 2015; 3: 599-609.

15. Zhang L, Shi Z and Zhong Y. Attitude estimation and control of a 3-DOF lab helicopter only based on optical flow. *Adv Rob* 2016; 30: 505–518.

16. Zhang L, Xiong Z, Lai J, et al. Optical flow-aided navigation for UAV: a novel information fusion of integrated MEMS navigation system. *Optik* 2016; 127: 447–451.

17. Tehrani MH, Garratt MA and Anavatti SG. Low-altitude horizon-based aircraft attitude estimation using UV-filtered panoramic images and optic flow. *IEEE Trans Aerosp Electron Syst* 2016; 52: 2362–2375.

18. Dusha D, Mejias L and Walker R. Fixed-wing attitude estimation using temporal tracking of the horizon and optical flow. *J Field Rob* 2011; 28: 355–372.

19. Carrio A, Bavlé H and Campoy P. Attitude estimation using horizon detection in thermal images. *Int J Micro Air Veh* 2018; 10: 352–361.

20. Fasano G, Rufino G, Accardo D, et al. Satellite angular velocity estimation based on star images and optical flow techniques. *Sensors* 2013; 13: 12771–12793.

21. Romero H, Salazar S and Lozano R. Real-time stabilization of an eight-rotor UAV using optical flow. *IEEE Trans Rob* 2009; 25: 809–817.

22. Liu M, Wang Y and Guo L. 6-DOF motion estimation using optical flow based on dual cameras. *J Cent South Univ* 2017; 24: 459–466.

23. Shuster M. Survey of attitude representations. *J Astronaut Sci* 1993; 41: 439–517.

24. Gebre-Egziabher D, Hayward RC and Powell JD. Design of multi-sensor attitude determination systems. *IEEE Trans Aerosp Electron Syst* 2004; 40: 627–649.

25. Gebre-Egziabher D and Elkaim GH. MAV attitude determination by vector matching. *IEEE Trans Aerosp Electron Syst* 2008; 44: 1012–1028.

26. Crassidis JL and Markley FL. Unscented filtering for spacecraft attitude estimation. *J Guid Control Dyn* 2003; 26: 536–542.

27. Che Z, Purushotham S, Cho K, et al. Recurrent neural networks for multivariate time series with missing values. *Sci Rep* 2018; 8: 6085.

28. Wang J, Zhao R, Wang D, et al. Machine health monitoring using local feature-based gated recurrent unit networks. *IEEE Trans Ind Electron* 2017; 65: 1539–1548.

29. Alom MZ, Taha TM, Yakopcic C, et al. A state-of-the-art survey on deep learning theory and architectures. *Electronics* 2019; 8: 292.

30. Schmidhuber J. Deep learning in neural networks: an overview. *Neural Netw* 2015; 61: 85–117.

31. KolanoWSki K, Swietlicka A, Kapela R, et al. Multisensor data fusion using Elman neural networks. *Appl Math Comput* 2018; 319: 236–244.

32. Guo C, Yan J and Tian Z. Analysis and design of an attitude calculation algorithm based on Elman neural network for SINS. *Cluster Computing* 2019; 22: 15267–15272.