

INERTIAL-ONLY NAVIGATION SYSTEM FOR UNMANNED AERIAL VEHICLES

Yifan Lin

Student# 1009867128

i.lin@mail.utoronto.ca

Dingsheng Liu

Student# 1009891128

jackds.liu@mail.utoronto.ca

Xuanhao Lu

Student# 1009883360

georgexlu.lu@mail.utoronto.ca

Shengwen Zhao

Student# 1009796687

jm.zhao@mail.utoronto.ca

1 INTRODUCTION

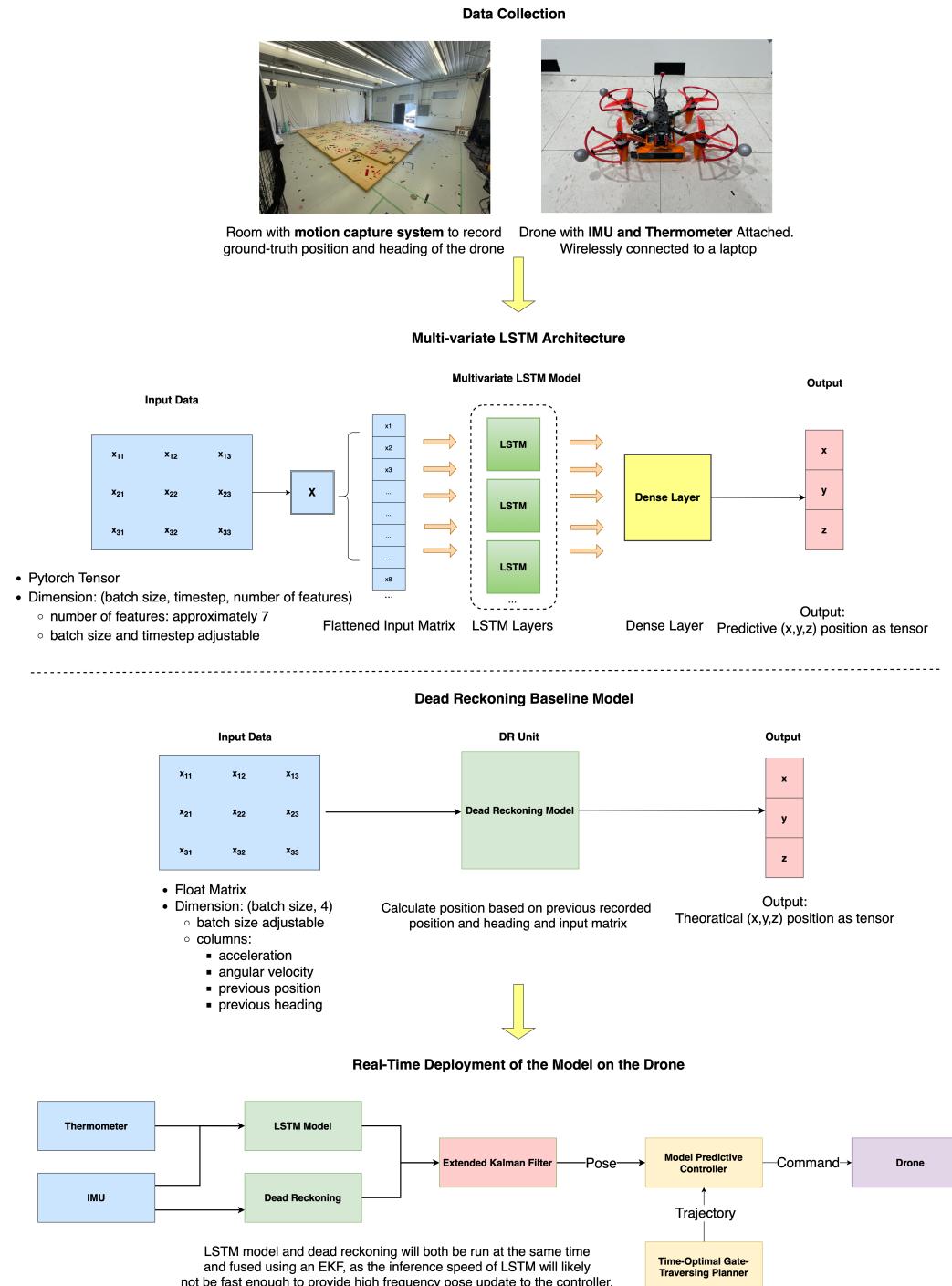
The rapid advancement of drone technology has created exciting opportunities across various fields, from aerial photography to disaster management. However, a challenge remains: accurately predicting a drone's position in real-time, especially in complex environments. Traditional methods like visual-inertial odometry (VIO) rely heavily on visual data, which can be unreliable in low-light conditions or when obstacles obstruct the view. Moreover, this approach requires high computational power, which is difficult to achieve under edge conditions on a drone. These challenges have inspired our project to seek out innovative solutions for more robust and precise drone localization.

Our project aims to develop a system that leverages deep learning, specifically Long Short-Term Memory (LSTM) models, to predict a drone's position using data from inertial measurement units (IMUs). This approach is both interesting and significant because it allows drones to operate independently of external visual cues, enhancing their reliability and versatility. This capability is especially important in scenarios where traditional navigation aids might fail, such as in dark or cluttered environments. Additionally, this approach decreases the computational power required and requires less hardware, making drones more accessible to those interested in such areas.

Deep learning is a powerful tool for this task due to its ability to process large amounts of sequential data and uncover complex patterns. LSTM models, a type of recurrent neural network (RNN), are particularly well-suited for time-series prediction, making them ideal for handling the continuous data streams from IMUs. By learning long-term dependencies in this data, LSTMs can provide more accurate and reliable position predictions.

In summary, our project aims to explore further in depth, the field of drone navigation by utilizing the strengths of deep learning. By overcoming the limitations of existing methods, we hope to create a more robust and accurate system for drone localization, contributing to the broader advancement of autonomous navigation technologies.

2 ILLUSTRATION/FIGURE



3 BACKGROUND & RELATED WORK

Currently, the most common conventional method for predicting a drone's position is VIO, which focuses on two different approaches: filtering methods and fixed-lag smoothing methods.

- **Filtering Method:** This method combines inputs from both the camera and the IMU to predict the position. It simplifies the processing by using camera data primarily for feature detection rather than for complete prediction, thereby increasing efficiency (Mourikis & Roumeliotis, 2007).
- **Fixed-Lag Smoothing Methods:** This method utilizes data from every previous frame, integrating it into the calculation of the current system's position before discarding the data. This ensures both accuracy and low computational overhead (Leutenegger et al., 2013).

However, due to the superior performance of IMUs in position prediction during complicated situations, scholars have begun to explore their integration with deep learning (DL). Innovations such as IONet, TLIO, and IMU Gyroscope Denoising with DL have paved the way for applying DL in this field.

- **IONet:** The system segments IMU data into smaller parts to decrease the error accumulated over time and shifts position prediction from traditional integration to feeding the input into an RNN, transforming it into a sequential learning problem (Chen et al., 2018).
- **TLIO:** This system introduces a novel approach by first processing the input data through a convolutional neural network (CNN) while maintaining the uncertainty of the calculation to ensure robustness in predictions. A method involving random perturbations is used to prevent the CNN from becoming idle when the data is straightforward. The processed data is then fed into an Extended Kalman Filter (EKF) to complete the prediction (Liu et al., 2020).
- **IMU Gyroscopes with DL:** The system utilizes a CNN to denoise data from the IMU, which decreases the potential errors accumulated over time, thus enhancing the precision of orientation estimation critical for autonomous navigation. Additionally, the network architecture effectively uses dilated convolutions, enabling it to capture temporal relationships over extended periods without significantly increasing computational complexity (Brossard et al., 2020).

4 DATA PROCESSING

4.1 SOURCE OF DATA

For consumer-grade IMUs, their noise profile can vary significantly between units, even if they share the same design. To our knowledge, there is no sufficiently large publicly available dataset for any particular IMU. Therefore, all the data used for training and validating the model will be collected by us. While this implies that the model needs re-training for each individual IMU, if we can streamline the data collection and training procedures, re-training the model for a new IMU should not be time-consuming.

4.2 DATA COLLECTION

4.2.1 PLATFORM

To collect the data, we will fly a lightweight 5-inch quadrotor in a 10 by 10 meter motion capture room at the University of Toronto Institute for Aerospace Studies (UTIAS). The flight will be performed autonomously using an existing and proven reliable autopilot system to avoid loss of control at high speed in a confined environment. We have obtained full permission to access all the related facilities and equipment mentioned in this proposal.

The motion capture room is equipped with 16 OptiTrack cameras along the edges. They can track the five reflective markers on the drone to determine its ground truth position and altitude at 120 Hz. This information will be collected and treated as our ground-truth label, while also being sent to the drone in real time to provide localization for the autopilot system.

The drone will be using a Betaflight-based low-level flight controller and powertrain capable of generating 40 N of thrust. It will also carry an Nvidia Jetson Orin NX onboard computer that provides computing power for the autopilot systems and records the onboard IMU readings. The autopilot will be able to follow trajectories through a number of waypoints at varying speeds generated by the

Time-Optimal Gate-Traversing (TOGT) Planner (Qin et al., 2024). It employs an Extended Kalman Filter (EKF) to fuse IMU readings and poses provided by the OptiTrack system for localization, as well as a Model Predictive Controller (MPC). The system will be communicating using the Robot Operating System (ROS) (Quigley et al., 2009). Close to the IMU, there will also be a temperature sensor. The IMU readings, temperature, and OptiTrack pose measurements will be recorded on the Jetson computer in the form of rosbag files. Both types of data will be timestamped at the time of collection.

Our IMU of interest will be mounted on the drone. It is a Bosch BMI270, capable of measuring linear acceleration up to 16 G and angular velocity up to 2000 degrees per second. The sensor's native readout rate is 3.2 kHz, but due to limitations in data transfer and inference time of the model, we will only be recording the data at 400 Hz.

4.2.2 PROCEDURE

We intend to execute a minimum of three flight trajectories: circular with a 3 m radius, Split-S, and sweep through the entire field; each at the speed limit of 1 m/s, 3 m/s, and no speed limit. The test session will be run twice - once at noon and once in the evening for different temperatures. These should enable us to cover a wide variety of working conditions for the IMU.

For each flight path, first, the position of the waypoints and their order will be fed into the TOGT planner to generate a trajectory in the form of a Comma-Separated Values (CSV) file. Then, the file will be loaded into the Agilicious flight stack onboard the drone (Foehn et al., 2022). After that, the rosbag recording for IMU readings, temperature, and ground truth pose will be started. Finally, the command to start the flight will be sent and the rosbag file will be saved after each flight.

4.2.3 PROCESSING

After obtaining the rosbag file, the portion of the recording before takeoff and after landing will be trimmed out. The IMU data will be parsed and converted to a CSV file in the format of $[t, a_x, a_y, a_z, \omega_x, \omega_y, \omega_z]$. The pose data will then be parsed in terms of translation and rotation represented by quaternions $[t, x, y, z, \varepsilon_x, \varepsilon_y, \varepsilon_z, \eta]$. The same applies for the temperature. The pose data along with temperature will be linearly interpolated according to the timestamp of IMU data and finally merged with IMU data. The final CSV file will have the following format:

$$[t, a_x, a_y, a_z, \omega_x, \omega_y, \omega_z, x, y, z, \varepsilon_x, \varepsilon_y, \varepsilon_z, \eta, \text{temp}] \quad (1)$$

which should be easily accessible by a PyTorch dataloader.

5 ARCHITECTURE

For this project of drone localization and position adjustment based on IMU data and ground-truth position obtained from the motion-capture system as mentioned in the previous sections, we will utilize the multivariate LSTM. This specialized RNN is selected due to its ability to capture long and short term dependencies in sequential positional data from IMU accelerometers and gyroscopes (Zhang et al., b).

A rough template of the architecture is depicted below: As multivariate LSTM is implemented, the input layer will be a pytorch tensor consisting of the data obtained from IMU accelerometers, gyroscopes, thermometer, and theoretical position and heading calculated by mathematical integration. The size of the tensor is (num_batches, timesteps, num_features) with:

- **Num_batches:** a time interval where data will be collected and reported to the system. In the meantime, data will be stored in the tensor.
- **Time_step:** LSTM timestep
- **Num_features:** features such as acceleration, temperature, and theoretical position.

We are planning to have two LSTM layers with a unit of 64 (subjected to adjustment). More layers can be added if required. Dense layers will have a unit of 32 and utilize Relu activation function to

introduce non-linearity. The output layer will have a unit of 3 (predictive x, y, z coordinate) with linear activation function.

Mean Square Error (MSE) is selected as the loss function as it is suitable for calculating the error between the predictive position and actual position. The error can then be utilized for position adjustment. Adam is selected as the optimizer. Note that Bahdanau Attention Mechanism might be included for more accurate prediction (Zhang et al., a).

6 BASELINE MODEL

Dead reckoning is selected as our baseline model which calculates instantaneous position of the drone based on mathematical integration. Specifically, the acceleration data obtained from the IMU is integrated twice to obtain the relative position. First, acceleration is integrated to obtain velocity, and then velocity is integrated to obtain position. Angular velocity data can also be integrated to adjust for orientation changes. The acceleration values are summed up over time and multiplied by the time step to obtain changes in velocity. Similarly, changes in velocity are integrated over time to obtain changes in position. Simultaneously, angular velocity values are integrated over time to calculate changes in orientation (i.e. Euler angles or quaternions). Ultimately, the relative position calculated through integration is adjusted over time based on the changes in acceleration and angular velocity (Advanced Navigation, n.d.).

7 ETHICAL CONSIDERATIONS

7.1 ETHICAL CONCERNNS

The introduction of an unmanned aerial vehicle equipped with onboard localization capabilities inevitably triggers ethical considerations. Our aspiration is for our model to enable drone navigation without reliance on external aids such as light, GPS, or other navigation systems. This autonomy opens doors to various military applications, envisioning scenarios where drones can navigate complex terrains and structures in complete darkness, even in the most remote areas, to execute essential tasks (Taylor & Guilmartin, 2024). Consequently, as we delve into the development of this project, it becomes imperative to contemplate the military implications of UAVs.

7.2 POSSIBLE LIMITATIONS

Nevertheless, this project comes with its limitations. Given that our training data will be confined to specific environments, there exists the risk of bias and inherent errors within our model, and reduces its generalizability. Additionally, the relative nature of starting points and positions introduces limitations tied to the volume of training data. As the model encounters diverse scenarios influenced by mere setup variations, the requirement for an extensive dataset becomes apparent to accommodate the array of situations encountered during testing. Furthermore, as we aim to gather our own data with the drone's IMU, which raises concerns regarding the accuracy and quantity of data collected (Inertial Labs, 2024).

8 PROJECT PLAN

Group Member	Responsibilities	Task Description	Accomplished so Far	Deadlines
Yifan Lin	Data collection, Data Processing	Set up flight platform, configure sensors, conduct data collection sessions, parse and clean data	Drone and data collection platform set up	June 24th, training data collection; July 10th, testing data collection

Shengwen Zhao	Basic model design, LSTM model design	Design initial baseline model, develop and train LSTM model, hyperparameter tuning	Architecture selection	July 1st, basic model set up; July 25th, testing of models accuracy
Xuanhao Lu	Optimization of models/Training	Optimize model performance, train models, integrate LSTM with drone navigation system	Environment set up and model test run	July 15th, basic model optimization for drone; July 31st, system optimization
Dingsheng Liu	Integration of model with drone movement	Implement dead reckoning model, integrate models with drone system, assist with data collection	Drone testing	July 15th, system integration; July 31st, bug fix and system testing with new environment
Entire Team	Testing and Validation	Conduct system integration, end-to-end testing, validate and debug the entire system	N/A	Aug 15th

Collaboration Plan:

- **Meetings:** twice a week via voice chat, Monday 4PM. - 6PM, Wednesday 4PM. - 6PM. with more meetings added if needed
- **Communication:** WeChat Group chat and Group voice call
- **Collaboration Mechanisms:** To ensure effective collaboration, all code and work will be put into a GitHub Repository, so that everyone can access and assess each person's work. Furthermore, we will use the fork function on Github to ensure members only work in their parts and enables us to compare contents to prevent conflict. In addition, a specific format is used, so that the work stays organized and documented, so that anyone can pick up someone else's work exactly where they left off efficiently if needed.

9 RISK REGISTER

Our project faces several significant risks, each requiring careful consideration and mitigation strategies. Firstly, the accuracy and quantity of the data collected using our own drones pose a high-risk factor. The reliability and effectiveness of our model heavily rely on the quality and volume of this data. To mitigate this risk, we will implement stringent data collection protocols, including precise drone calibration and validation against ground truth. Additionally, strategies such as data augmentation will be explored to enhance dataset diversity and robustness.

Another considerable risk is the limited availability of drones for testing. With a finite number of drones at our disposal, any complications during testing could impede data collection and subsequently delay project progress. To counteract this, meticulous planning and coordination of testing activities will be prioritized to minimize the likelihood of incidents. Furthermore, potential partnerships or collaborations with other teams could provide access to additional drones if needed.

Additionally, environmental factors pose a significant risk to data accuracy. Variations in temperature, wind, humidity, and other environmental conditions can significantly impact the reliability of the data collected. (Hayward, 2024) To address this risk comprehensively, measurement devices will be integrated to monitor and account for these factors during data collection. This proactive approach will ensure that data remains consistent and reliable across different environmental conditions.

Lastly, there's the risk of delays in model training. Training complex machine learning models for UAV navigation often proves time-consuming due to various factors such as computational limita-

tions or unforeseen technical challenges. To address this risk, realistic timelines for model training will be estimated through feasibility studies, with adequate resources allocated to support efficient training. In case of delays, a proactive reassessment of our approach and potential optimization techniques will be conducted to expedite training without compromising quality.

10 LINK TO GITHUB

https://github.com/snowstormnight/APS360_Project

REFERENCES

- Advanced Navigation. Pedestrian dead reckoning (pdr): An introduction. n.d. URL <https://www.advancednavigation.com/tech-articles/pedestrian-dead-reckoning-pdr-an-introduction/>. Accessed: 2024-06-05.
- Martin Brossard, Silvere Bonnabel, and Axel Barrau. Denoising imu gyroscopes with deep learning for open-loop attitude estimation, 2020.
- Changhao Chen, Xiaoxuan Lu, Andrew Markham, and Niki Trigoni. Ionet: learning to cure the curse of drift in inertial odometry. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence, AAAI'18/IAAI'18/EAAI'18*. AAAI Press, 2018. ISBN 978-1-57735-800-8.
- Philipp Foehn, Elia Kaufmann, Angel Romero, Robert Penicka, Sihao Sun, Leonard Bauersfeld, Thomas Laengle, Giovanni Cioffi, Yunlong Song, Antonio Loquercio, and Davide Scaramuzza. Agilicious: Open-source and open-hardware agile quadrotor for vision-based flight. *Science Robotics*, 7(67), June 2022. ISSN 2470-9476. doi: 10.1126/scirobotics.abl6259. URL <http://dx.doi.org/10.1126/scirobotics.abl6259>.
- Laura Hayward. Inertial measurement unit (imu): An introduction. In *Advanced Navigation*, 2024.
- Inertial Labs. Advantages and disadvantages of inertial measurement units (imus), 2024. URL <https://inertiallabs.com/advantages-and-disadvantages-of-inertial-measurement-units/>.
- Stefan Leutenegger, Paul Timothy Furgale, Vincent Rabaud, Margarita Chli, Kurt Konolige, and Roland Y. Siegwart. Keyframe-based visual-inertial slam using nonlinear optimization. In *Robotics: Science and Systems*, 2013.
- Wenxin Liu, David Caruso, Eddy Ilg, Jing Dong, Anastasios Mourikis, Kostas Daniilidis, Vijay Kumar, Jakob Engel, Abhinav Valada, and Tamim Asfour. Tlio: Tight learned inertial odometry. *IEEE Robotics and Automation Letters*, PP:1–1, 07 2020. doi: 10.1109/LRA.2020.3007421.
- Anastasios I. Mourikis and Stergios I. Roumeliotis. A multi-state constraint kalman filter for vision-aided inertial navigation. In *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pp. 3565–3572, 2007. doi: 10.1109/ROBOT.2007.364024.
- Chao Qin, Maxime S.J. Michet, Jingxiang Chen, and Hugh H. T. Liu. Time-optimal gate-traversing planner for autonomous drone racing. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 2024.
- Morgan Quigley, Brian Gerkey, Ken Conley, Josh Faust, Tully Foote, Jeremy Leibs, Eric Berger, Rob Wheeler, and Andrew Ng. Ros: an open-source robot operating system. In *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA) Workshop on Open Source Robotics*, Kobe, Japan, May 2009.
- John W. R. Taylor and John F. Guilmartin. Unmanned aerial vehicles (uavs). *Encyclopaedia Britannica*, 2024. URL <https://www.britannica.com/technology/military-aircraft/Unmanned-aerial-vehicles-UAVs>.

Aston Zhang, Zachary C. Lipton, Mu Li, and Alexander J. Smola. *Dive into Deep Learning.* D2L.ai, a. URL https://d2l.ai/chapter_attention-mechanisms-and-transformers/bahdanau-attention.html. Accessed: 2024-06-05.

Aston Zhang, Zachary C. Lipton, Mu Li, and Alexander J. Smola. *Dive into Deep Learning.* D2L.ai, b. URL https://zh.d2l.ai/chapter_recurrent-modern/lstm.html. Accessed: 2024-06-05.