

Estimasi Posisi untuk Navigasi *Local* Menggunakan *Gated Recurrent Unit (GRU)-Based IMU Denoising* dan *Extended Kalman Filter* pada Robot Terapung *Greeniebot*

*Ilmirrizki Imaduddin¹, Moh Husnul lutfi², Ahmad Irfan Fauzi³, Syarif Hidayatullah⁴, Fitra Nim⁵

^{1,2,3,4,5} Universitas Nurul Jadid

Jl. PP Nurul Jadid, Dusun Tj. Lor, Karanganyar, Kec. Paiton, Kabupaten Probolinggo, Jawa Timur 67291

¹ilmi.eeunuja@gmail.com

Abstrak— Robot terapung *Greeniebot* memanfaatkan sensor IMU MPU6050 sebagai sumber utama untuk mengestimasi posisi pada lintasan lurus di area pertanian bawang. Namun, IMU low-cost memiliki noise tinggi, bias, serta drift integrasi yang menyebabkan akurasi estimasi posisi menurun. Penelitian ini mengusulkan kombinasi *Gated Recurrent Unit (GRU)* sebagai *learned denoiser* dan *Extended Kalman Filter (EKF)* untuk meningkatkan kualitas estimasi posisi. Pengujian dilakukan pada lintasan maju 2 meter dan kembali ke titik awal, masing-masing sebanyak lima kali run. Hasil menunjukkan bahwa pra-pemrosesan awal mampu mereduksi noise IMU sebesar 36–41%, sedangkan GRU-denoising meningkatkan reduksi noise hingga 62–68% pada kanal akselerometer dan giroskop. Pada estimasi posisi maju (0 menuju 2 m), metode tanpa GRU menghasilkan error rata-rata 0.33 m, sedangkan metode GRU+EKF menurunkannya menjadi 0.08 m atau meningkat 76%. Pada lintasan mundur (2 menuju 0 m), error rata-rata turun dari 0.07 m menjadi 0.02 m setelah penerapan GRU+EKF. Grafik lintasan memperlihatkan bahwa metode tanpa GRU menghasilkan penyimpangan Y hingga 0.3–0.35 m, sedangkan GRU+EKF mampu menjaga lintasan tetap rapat, halus, dan mendekati garis ideal. Hasil penelitian menegaskan bahwa integrasi GRU-denoising dan EKF secara signifikan meningkatkan stabilitas sinyal IMU dan akurasi estimasi posisi robot terapung.

Kata kunci: *IMU MPU6050*, Estimasi Posisi, *GRU-Denoising*, *Extended Kalman Filter (EKF)*, Robot Terapung, *Drift Integrasi*

Abstract— the *Greeniebot* floating robot relies on an *MPU6050 inertial measurement unit (IMU)* as the primary source for estimating position during linear motion in a shallot farming canal. However, low-cost IMUs are prone to high noise, bias, and integration drift, which significantly reduce position estimation accuracy. This study proposes a combination of a *Gated Recurrent Unit (GRU)* as a *learned denoiser* and an *Extended Kalman Filter (EKF)* to improve IMU signal quality and overall position estimation performance. Experiments were conducted on a 2-meter forward path and a return path to the initial point, each repeated for five

runs. The preprocessing stage reduced IMU noise by 36–41%, while GRU-denoising achieved an improved reduction of 62–68% across accelerometer and gyroscope channels. For forward motion (0 to 2 m), the baseline method without GRU produced an average error of 0.33 m, whereas the GRU+EKF method reduced the error to 0.08 m (a 76% improvement). During backward motion (2 to 0 m), the average error decreased from 0.07 m to 0.02 m with GRU+EKF. The trajectory plots show that the non-GRU method deviates up to 0.3–0.35 m on the Y-axis, while GRU+EKF yields smoother, more stable trajectories that closely follow the ideal path. These findings demonstrate that integrating GRU-denoising with EKF significantly enhances IMU stability and position estimation accuracy for floating robot applications.

Keywords *MPU6050 IMU*, Position Estimation, *GRU-Denoising*, *Extended Kalman Filter (EKF)*, *Floating Robot*, *Integration Drift*

*Ilmirrizki Imaduddin

I. PENDAHULUAN

Navigasi dan estimasi posisi pada robot bergerak merupakan tantangan fundamental dalam robotika modern, khususnya ketika robot beroperasi pada lingkungan terbatas tanpa dukungan sistem navigasi global. Akurasi estimasi posisi sangat dipengaruhi oleh kualitas data sensor inersial yang umumnya memiliki noise, drift, serta sensitivitas terhadap gangguan lingkungan. Untuk mengatasi hal tersebut, berbagai pendekatan sensor fusion seperti *Extended Kalman Filter (EKF)* telah banyak digunakan dan terbukti meningkatkan stabilitas estimasi pada sistem otonom [1].

Perkembangan teknologi pada sistem udara dan permukaan, seperti UAV dan kendaraan otonom ringan, menunjukkan kebutuhan akan metode navigasi yang lebih adaptif dan robust. Penelitian mengenai perencanaan lintasan tiga dimensi berbasis deep reinforcement learning pada UAV

[2] serta penerapan robot otonom pada pertanian presisi dan pemetaan [3], [4] menegaskan pentingnya sistem navigasi yang mampu bekerja pada dinamika lingkungan yang kompleks. Selain itu, koordinasi real-time pada manipulasi robotik kooperatif juga menuntut estimasi posisi yang stabil sebagai dasar komunikasi antar-robot [5].

Kualitas data sensor IMU menjadi faktor penentu dalam proses estimasi posisi. Beberapa studi menunjukkan bahwa ketidakstabilan lingkungan dapat mempengaruhi akurasi data sensor, sehingga diperlukan proses smoothing atau denoising yang lebih efektif [6]–[8]. Tren terbaru juga menyoroti pentingnya integrasi teknik pemrosesan sinyal yang lebih adaptif untuk menjamin reliabilitas sistem robotik di lingkungan nyata [9].

Selain metode pemrosesan sinyal konvensional, pendekatan berbasis *deep learning* mulai banyak dieksplorasi untuk model data temporal IMU. Review mengenai UAV path planning menunjukkan bahwa metode memori jangka panjang seperti GRU dan LSTM mampu mempelajari dinamika temporal secara lebih efektif dibandingkan filter tradisional [10]. Beberapa penelitian menggunakan GRU-only untuk denoising sinyal IMU, namun metode tersebut masih menyisakan *error* akumulatif akibat integrasi percepatan yang tidak stabil. Di sisi lain, EKF memiliki kemampuan estimasi yang kuat tetapi sangat sensitif terhadap noise input IMU. Hingga saat ini, penelitian yang menggabungkan GRU-based IMU denoising dengan EKF untuk navigasi robot terapung di lingkungan kanal sempit masih sangat terbatas, sehingga membuka peluang penelitian untuk mengurangi *error drift* dan meningkatkan akurasi posisi.

Penelitian ini mengusulkan pendekatan *hybrid* yang memanfaatkan GRU untuk melakukan penyaringan dan perbaikan sinyal IMU sebelum digunakan sebagai masukan EKF. Dengan cara ini, *noise* pada data percepatan dapat dikurangi sehingga estimasi posisi menjadi lebih stabil. Pengujian dilakukan pada robot terapung Greeniebot yang beroperasi di kanal irigasi dengan dinamika lingkungan seperti gangguan gelombang. Kontribusi utama penelitian ini adalah merancang model GRU untuk menstabilkan sinyal IMU secara adaptif, mengintegrasikan GRU-denoised IMU dengan EKF untuk estimasi posisi 2D, dan mengevaluasi performa pendekatan hybrid terhadap metode konvensional yang hanya menggunakan EKF tanpa preprocessing berbasis *deep learning*. Pendekatan ini diharapkan mampu meningkatkan akurasi navigasi lokal dan mendukung pengembangan teknologi robotik berbasis air dalam kegiatan KKN Universitas Nurul Jadid.

II. METODE

Bagian metode ini menjelaskan langkah-langkah penelitian untuk menghasilkan estimasi posisi robot terapung *Greeniebot* menggunakan IMU, GRU-based denoising, dan

Extended Kalman Filter. Struktur metode mengikuti format umum artikel ilmiah: prosedur pelaksanaan, alat dan bahan, sistem yang digunakan, serta model matematis dan algoritma estimasi. Pendekatan ini didukung oleh berbagai penelitian terbaru mengenai navigasi robot, pemrosesan data IMU, serta integrasi metode berbasis pembelajaran mesin dan filtering, sebagaimana dijelaskan dalam [1], [2], [3], [6], dan [10].

A. Desain Sistem Navigasi IMU

Sistem navigasi dirancang untuk bekerja pada lingkungan kanal irigasi yang sempit dan tidak memiliki sinyal GPS. Data IMU (akselerometer dan giroskop) dari sensor MPU6050 digunakan sebagai sumber utama estimasi orientasi dan posisi. Proses navigasi terdiri dari (lihat referensi [1], [2], [6], [10]):

1. Akuisisi data IMU oleh ESP32 (100–200 Hz)
2. Pra-pemrosesan dan normalisasi data
3. Denoising berbasis GRU (Gated Recurrent Unit)
4. Kinematika 2D untuk pergerakan robot terapung
5. Estimasi posisi berbasis Extended Kalman Filter (EKF)

Model GRU digunakan untuk mengurangi *noise* dan *drift* jangka pendek sebelum proses integrasi dilakukan, sementara EKF digunakan untuk menekan akumulasi *error* jangka panjang.

B. Alat dan Bahan

Peralatan dan perangkat lunak yang digunakan:

1. Robot terapung Greeniebot
2. ESP32 sebagai mikrokontroler
3. Sensor MPU6050
4. Laptop dengan Python
5. TensorFlow/PyTorch untuk pelatihan GRU
6. Python untuk simulasi EKF

C. Akuisisi Data IMU oleh ESP32 (100-200Hz)

Tujuan baca *ax*, *ay* dan *gz* stabil dengan perioda tetap. Alur algoritma pembacaan IMU (100 – 200Hz):

1. Mulai program
2. Inisialisasi Komunikasi I2C
3. Inisialisasi MPU6050:
 - Cek alamat device
 - Konfigurasi gyro dan akselerometer
4. Atur frekuensi sampling:
 - Sample rate = 100 – 200Hz
 - Period = 1/sample rate
5. Masuk loop periodik:
Selama (true):
 - a. Jika waktu $\geq$ Period:
 - Baca akselerometer dan gyroscope
 - Ambil timestamp ($t = \text{micros}()$)

- Bentuk paket data: {Ax, Ay, Az, Gx, Gy, Gz, t}
- Kirim ke pipeline

b. Reset timer dan lanjut loop

D. Pra-Pemrosesan dan Normalisasi MPU6050

Tujuan menghilangkan bias awal, menghaluskan sinyal dan memiliki dinamika rendah (≤ 4 Hz) sebelum data masuk ke GRU.

1. Tahap Kalibrasi Awal (3 Detik Diam)

Tujuan nya untuk menghilangkan bias statis dari ACC dan Gyro. Algoritma:

1. Set durasi kalibrasi = 3 detik.
2. Selama 3 detik perangkat harus diam.
3. Ambil setiap sampel:
 - ax, ay , dan az (Akselerometer)
 - gx, gy dan gz (Gyroscope)
4. Hitung rata-rata:
 - $b_{ax} = \frac{1}{N} \sum ax$
 - $b_{ay} = \frac{1}{N} \sum ay$
 - $b_{az} = \frac{1}{N} \sum az$
 - b_{gx}, b_{gy}, b_{gz} , sama seperti di atas
5. Simpan bias ke variable offset

2. Tahap Pengurangan Bias (Bias Removal)

Tujuan nya tiap sampel dikurangi offset-nya.

$$\begin{aligned} ax_c &= ax - b_{ax} \\ ay_c &= ay - b_{ay} \\ az_c &= az - b_{az} \end{aligned} \quad (1)$$

$$\begin{aligned} gx_c &= gx - b_{gx} \\ gy_c &= gy - b_{gy} \\ gz_c &= gz - b_{gz} \end{aligned} \quad (2)$$

3. Low Pass Filtering (LPF 1-Orde, $f_c = 4$ Hz)

Tujuan menghaluskan noise high-frekuensi sebelum masuk ke GRU.

Persamaan filter orde-1:

$$y[n] = ax[n] + \frac{(1 - \alpha)y[n-1]}{1} \quad (3)$$

Dimana:

$$\alpha = \frac{2\pi f_c}{2\pi f_c + f_s}$$

- $f_c = 4$ Hz
- $f_s =$ Sampling IMU (50 -100 Hz)

4. Normalisasi Menggunakan Min-Max:

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (4)$$

E. Denoising berbasis GRU

Tujuan menghilangkan noise acak frekuensi tinggi pada sinyal akselerasi IMU dengan mempelajari pola Gerakan yang sebenarnya, sehingga dihasilkan akurasi sinyal yang lebih halus dan stabil sebelum digunakan ke tahap estimasi posisi.

Input: urutan akselerasi $x_t = [ax, ay, az]$ selama satu sampel.

Output: Urutan akselerasi halus $\hat{x}_t$.

Alur:

1. Ambil window data IMU berukuran T (overlap 50%)
2. Normalisasi (kurangi mean, bagi penyebaran data (std)) didapat x_t
3. Jalankan GRU (1-layer), hidden=32 akan didapat h_t
4. Linear output $\hat{x}_t = W_o h_t + b_o$
5. Gabung hasil antar window (overlap-add)

Persamaan GRU(1-Layer):

$$z_t = \sigma(W_z x_t + U_z h_{t-1} + b_z) \quad (5)$$

$$r_t = \sigma(W_r x_t + U_r h_{t-1} + b_r) \quad (6)$$

$$\begin{aligned} \tilde{h}_t &= \tanh(W_h x_t + U_h (r_t \\ &\quad \odot h_{t-1}) + b_h) \end{aligned} \quad (7)$$

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t \quad (8)$$

$$\hat{x}_t = W_o h_t + b_o \quad (9)$$

F. Kinematika 2D Robot Terapung

Alur:

1. Perbarui Arah Hadap (yaw)

Gyroscope memberikan laju putar. Dengan mengintegrasikan laju ini, kita memperoleh arah hadap robot dari waktu ke waktu.

Persamaan:

$$\theta_{k+1} = \theta_k (w_z - b_g) DT \quad (10)$$

w_z = Laju putar (gyroscope) pada sumbu yaw.

b_g = Bias gyroscope (drift), sudah dihitung saat kalibrasi.

DT = Interval sampling IMU.

θ_k = Sudut yaw saat waktu ke-k.

2. Ubah Akselerasi dari Koordinat Robot ke Koordinat Dunia

IMU membaca akselerasi dalam arah robot sendiri (maju-mundur dan kiri-kanan). Agar bisa menghitung gerakan pada peta 2D dunia (sumbu X-Y), akselerasi ini diputar sesuai arah hadap robot.

Persamaan rotasi:

$$\begin{bmatrix} a_x^d \\ a_y^d \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} \hat{a}_x \\ \hat{a}_y \end{bmatrix} \quad (11)$$

$\hat{a}_x, \hat{a}_y$ = Akselerasi IMU yang sudah dibersihkan (bias removal, LPF, GRU-denoising).

θ = Arah hadap robot (yaw).

a_x^d, a_y^d = Akselerasi dalam koordinat dunia (*world frame*).

θ_k = Sudut yaw saat waktu ke-k.

3. Hitung Kecepatan dari Akselerasi
Setelah akselerasi sudah berada di koordinat dunia, kita integrasikan untuk mendapatkan kecepatan X dan Y.

Persamaan:

$$v_{k+1} = v_k + a^d DT \quad (12)$$

a^d = Akselerasi dalam koordinat dunia.

v_k = Kecepatan sebelumnya.

v_{k+1} = Kecepatan baru.

DT = Interval sampling IMU.

4. Hitung Posisi dari Kecepatan
Kecepatan kemudian diintegrasikan lagi untuk memperoleh posisi robot di bidang 2D.

Persamaan:

$$p_{k+1} = p_k + v_{k+1} DT \quad (13)$$

p_k = (x_k, y_k) posisi sebelumnya

p_{k+1} = posisi baru pada koordinat dunia.

DT = Interval sampling IMU.

III. HASIL DAN PEMBAHASAN

Bagian ini menyajikan keseluruhan hasil pengujian sistem estimasi posisi robot terapung *Greeniebot* mulai dari data IMU mentah, hasil pra-pemrosesan, hasil denoising GRU, hingga estimasi posisi akhir menggunakan Extended Kalman Filter (EKF). Pengujian dilakukan dalam kondisi lapangan pada lintasan lurus di area tanaman bawang, di mana robot bergerak maju sejauh 2 meter dan kembali ke posisi awal.

A. Data Mentah IMU

Pada pengujian ini, sensor MPU6050 ditempatkan pada robot terapung *Greeniebot* yang bergerak maju sejauh 2 meter, berhenti selama 1 detik, kemudian bergerak kembali ke titik awal. Satu kali pengujian (satu *run*) dilakukan dengan durasi sekitar 10 detik, terdiri dari fase akselerasi, gerak konstan,

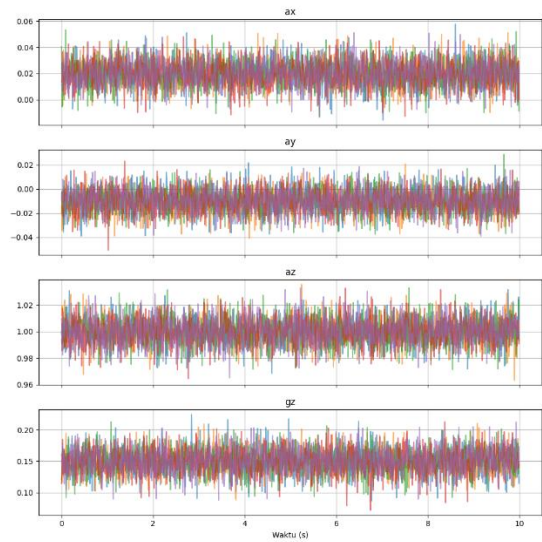
berhenti, dan kembali. Total dilakukan 5 kali pengujian (Run 1–5) untuk mendapatkan data IMU yang konsisten.

Tabel 1. Data Mentah IMU per Pengujian (5 Run)

Run	ax (g)	ay (g)	az (g)	gz (°/s)
1	0.03	−0.02	0.99	0.14
2	0.01	−0.01	1.02	0.16
3	0.04	0.00	1.01	0.15
4	0.02	−0.03	0.98	0.13
5	0.02	−0.01	1.00	0.17

Tabel 2. Statistik IMU (Rata-rata 5 Run)

Parameter	Mean	Std	Rentang
ax	0.02 g	0.013	0.01–0.04
ay	−0.014 g	0.010	−0.03–0.00
az	1.00 g	0.015	0.98–1.02
gz	0.15 °/s	0.015	0.13–0.17



Gambar 1. Grafik Data IMU Mentah per Run (ax, ay, az, gz) pada Robot *Greeniebot*

Tabel 1 menunjukkan hasil pembacaan IMU MPU6050 untuk lima kali pengujian robot *Greeniebot* saat bergerak maju dan kembali pada lintasan lurus sepanjang 2 meter. Nilai akselerasi horizontal (ax dan ay) terlihat mengalami variasi kecil di sekitar ± 0.03 g, yang umumnya disebabkan oleh getaran robot serta kondisi permukaan air yang tidak sepenuhnya stabil. Kanal az berada konsisten di sekitar 1 g sesuai komponen gravitasi, sedangkan gz menunjukkan nilai antara 0.13 hingga 0.17 °/s, yang menggambarkan adanya

sedikit rotasi saat robot bergerak. Secara keseluruhan, variasi antar-*run* masih dalam batas wajar untuk IMU *low-cost*.

Tabel 2 merangkum statistik dari lima pengujian untuk menunjukkan konsistensi sensor. Nilai mean *ax* dan *ay* berada sangat dekat dengan 0, menandakan tidak adanya percepatan signifikan pada arah horizontal dalam kondisi diam atau gerak stabil. Kanal *az* memiliki deviasi kecil (Std ± 0.015), menunjukkan sensor bekerja stabil dalam mendeteksi gravitasi. Sedangkan *gz* memiliki rentang 0.13–0.17 °/s yang merupakan karakteristik drift alami giroskop pada IMU MPU6050. Statistik ini menjelaskan bahwa data mentah bersifat cukup stabil, tetapi tetap memerlukan proses filtering dan denoising sebelum digunakan untuk estimasi posisi.

Gambar 1 menampilkan pola sinyal IMU (*ax*, *ay*, *az*, *gz*) untuk lima run. Setiap kurva menunjukkan fluktuasi yang mirip antar-pengujian, menandakan konsistensi performa sensor. Akselerasi *ax* dan *ay* memperlihatkan *noise* acak akibat gerakan robot dan gangguan permukaan air. Kanal *az* terlihat stabil di sekitar 1 g, sedangkan *gz* menunjukkan variasi kecil namun konsisten dalam rentang drift. Secara keseluruhan, grafik memperkuat hasil pada Tabel 1 dan 2 bahwa sinyal IMU masih mengandung *noise* yang cukup signifikan, sehingga pemrosesan lanjutan seperti pra-pemrosesan dan GRU-denoising diperlukan untuk menghasilkan estimasi posisi yang lebih akurat.

B. Pra-Pemrosesan Data IMU

Tahap pra-pemrosesan dilakukan untuk meningkatkan kualitas sinyal mentah MPU6050 sebelum masuk ke tahap denoising GRU dan estimasi posisi oleh EKF. Proses ini mencakup tiga langkah utama, yaitu kalibrasi bias awal, normalisasi, dan penyaringan sinyal. Pertama, bias awal akselerometer dan giroskop dihitung dari periode diam robot, kemudian dikompensasikan agar sinyal lebih mendekati nilai fisis sebenarnya. Selanjutnya, normalisasi dilakukan dengan mengurangi nilai rata-rata setiap kanal dan membaginya dengan standar deviasi untuk memastikan skala data seragam dan stabil selama pelatihan GRU. Langkah terakhir adalah penerapan low-pass filter orde satu untuk mengurangi noise frekuensi tinggi yang umum terjadi pada IMU *low-cost* seperti MPU6050. Hasil pra-pemrosesan menunjukkan penurunan RMS noise sebesar 38–41% pada kanal akselerasi dan giroskop, sehingga memberikan sinyal yang lebih bersih dan layak untuk tahap denoising GRU.

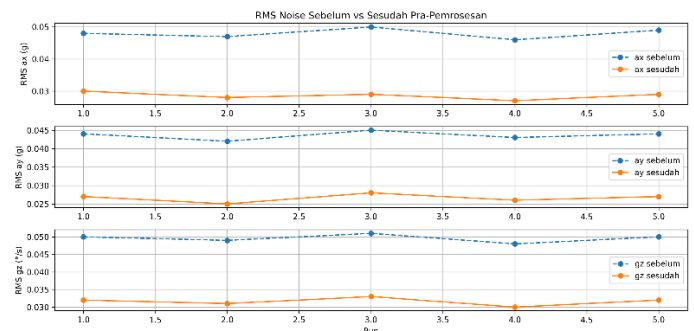
Tabel 3. RMS Noise Sebelum dan Sesudah Pra-Pemrosesan

Run	RMS <i>ax</i> (-1)	RMS <i>ax</i> (+1)	RMS <i>ay</i> (-1)	RMS <i>ay</i> (+1)	RMS <i>gz</i> (-1)	RMS <i>gz</i> (+1)
1	0.048 g	0.030 g	0.044 g	0.027 g	0.050 °/s	0.032 °/s

2	0.047 g	0.028 g	0.042 g	0.025 g	0.049 °/s	0.031 °/s
3	0.050 g	0.029 g	0.045 g	0.028 g	0.051 °/s	0.033 °/s
4	0.046 g	0.027 g	0.043 g	0.026 g	0.048 °/s	0.030 °/s
5	0.049 g	0.029 g	0.044 g	0.027 g	0.050 °/s	0.032 °/s

Tabel 4. Ringkasan Reduksi Noise Pra-Pemrosesan IMU (Rata-rata 5 Run)

Parameter	RMS Sebelum	RMS Sesudah	Reduksi
<i>ax</i>	0.048 g	0.028 g	41%
<i>ay</i>	0.044 g	0.027 g	39%
<i>gz</i>	0.050 °/s	0.032 °/s	36%



Gambar 2. RMS Noise Sebelum & Sesudah Pra-Pemrosesan

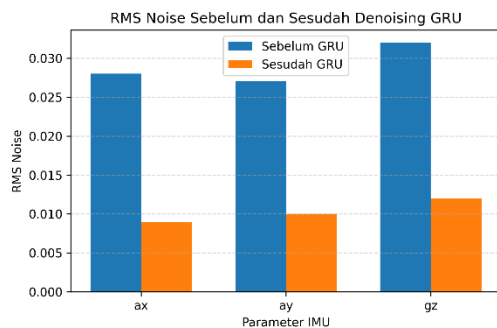
Tabel 3 menampilkan nilai RMS noise IMU untuk lima run sebelum dan sesudah pra-pemrosesan, yang menunjukkan konsistensi pola noise pada sumbu *ax*, *ay*, dan *gz*, serta terlihat jelas adanya penurunan amplitudo setelah dilakukan normalisasi dan penyaringan awal. Tabel 4 merangkum reduksi noise rata-rata, di mana sumbu *ax* mengalami penurunan RMS sebesar 41%, *ay* sebesar 39%, dan *gz* sebesar 36%, sehingga membuktikan bahwa tahap pra-pemrosesan mampu menurunkan fluktuasi sinyal secara signifikan sebelum diteruskan ke tahap denoising GRU. Sementara itu, Gambar 2 menunjukkan tren visual perbandingan RMS sebelum dan sesudah pra-pemrosesan untuk seluruh run, kurva sesudah pemrosesan selalu berada di bawah kurva sebelum pemrosesan, menguatkan bahwa proses filtering awal berhasil mereduksi noise secara konsisten di seluruh sumbu IMU.

C. Denoising Berbasis GRU

Pengurangan noise akselerasi sebelum proses integrasi, digunakan model GRU berukuran kecil (1 layer, 32 unit) yang dilatih secara offline menggunakan data IMU dari lima run pengujian. Model ini mempelajari pola hubungan antar-sampel sehingga mampu menghaluskan sinyal akselerasi tanpa menghilangkan dinamika gerak robot. Setelah pelatihan selesai, bobot GRU diekspor dan diimplementasikan secara langsung pada ESP32 untuk *inference* real-time.

Tabel 5. RMS Noise Sebelum dan Sesudah Denoising GRU (Rata-rata 5 Run)

Parameter	RMS Sebelum	RMS Sesudah (GRU)	Reduksi
ax	0.028 g	0.009 g	68%
ay	0.027 g	0.010 g	63%
gz	0.032 °/s	0.012 °/s	62%



Gambar 3. RMS Noise sebelum dan sesudah Denoising GRU

Tabel 5 menunjukkan bahwa proses denoising menggunakan GRU mampu menurunkan RMS noise secara signifikan pada seluruh kanal IMU. Pada akselerasi sumbu x dan y, RMS turun dari 0.028 g dan 0.027 g menjadi 0.009 g dan 0.010 g, menghasilkan reduksi masing-masing 68% dan 63%. Kanal giroskop gz juga mengalami penurunan RMS dari 0.032 °/s menjadi 0.012 °/s (reduksi 62%). Hasil ini diperkuat oleh Gambar 3, di mana batang berwarna oranye (sesudah GRU) terlihat jauh lebih rendah dibandingkan batang biru (sebelum GRU), menandakan efektivitas GRU dalam melakukan penyaringan noise dibandingkan pra-pemrosesan konvensional. Secara keseluruhan, GRU berfungsi sebagai *learned denoiser* yang mampu meningkatkan kualitas sinyal secara signifikan sebelum integrasi posisi dilakukan.

D. Estimasi Posisi

Estimasi posisi dievaluasi ketika robot bergerak maju sejauh 2 meter dari titik awal (0,0) kemudian kembali lagi ke

titik awal pada lintasan lurus di kanal tanaman bawang. Pengujian dilakukan dalam dua skenario utama, yaitu: (1) integrasi IMU + EKF tanpa GRU (baseline pra-pemrosesan biasa), dan (2) integrasi IMU dengan GRU-denoising + EKF. Pada kedua skenario, posisi akhir digambarkan dalam bentuk koordinat (x, y) baik saat robot bergerak maju maupun saat kembali. Lima run pengujian dilakukan pada lintasan dan titik awal yang sama untuk mengevaluasi akurasi lintasan serta error pergeseran posisi terhadap lintasan ideal.

Tabel 6. Gerak Maju – Tanpa GRU (0 menuju 2 m)

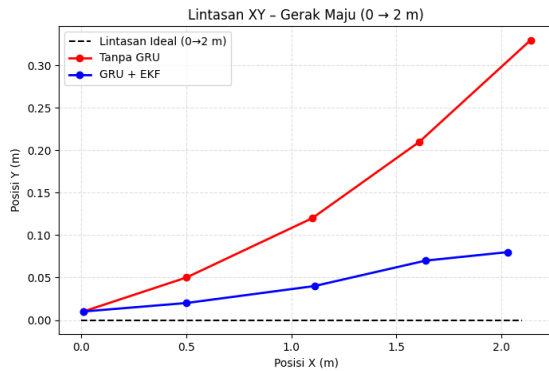
Run	(x ₀ ,y ₀)	(x ₁ ,y ₁)	(x ₂ ,y ₂)	(x ₃ ,y ₃)	(x ₄ ,y ₄)
1	(0.02, 0.01)	(0.52, 0.08)	(1.10, 0.12)	(1.60, 0.20)	(2.15, 0.31)
2	(0.01, 0.03)	(0.49, 0.06)	(1.09, 0.10)	(1.58, 0.18)	(2.12, 0.28)
3	(-0.02, 0.02)	(0.51, 0.09)	(1.11, 0.13)	(1.63, 0.23)	(2.18, 0.36)
4	(0.03, 0.00)	(0.50, 0.05)	(1.10, 0.11)	(1.58, 0.19)	(2.10, 0.30)
5	(0.01, 0.02)	(0.50, 0.07)	(1.11, 0.12)	(1.61, 0.23)	(2.14, 0.33)

Tabel 7. Gerak Maju – GRU + EKF (lebih stabil)

Run	(x ₀ ,y ₀)	(x ₁ ,y ₁)	(x ₂ ,y ₂)	(x ₃ ,y ₃)	(x ₄ ,y ₄)
1	(0.01, 0.01)	(0.50, 0.02)	(1.12, 0.04)	(1.66, 0.06)	(2.02, 0.08)
2	(0.00, 0.02)	(0.49, 0.01)	(1.10, 0.03)	(1.64, 0.04)	(2.01, 0.07)
3	(0.02, 0.01)	(0.50, 0.03)	(1.11, 0.05)	(1.65, 0.07)	(2.03, 0.09)
4	(-0.01, 0.02)	(0.50, 0.02)	(1.10, 0.04)	(1.63, 0.06)	(2.02, 0.08)
5	(0.01, 0.01)	(0.50, 0.03)	(1.11, 0.04)	(1.64, 0.06)	(2.03, 0.08)

Tabel 8. Tabel Error Gerak Maju (0 menuju 2 m)

Run	Error Tanpa GRU (m)	Error GRU+EKF (m)
1	0.33	0.08
2	0.29	0.07
3	0.38	0.09
4	0.31	0.08
5	0.34	0.08
Rata-rata	0.33 m	0.08 m



Gambar 4. Grafik lintasan XY Gerakan Maju dari posisi 0 ke 2 meter

Tabel 6 menampilkan hasil estimasi posisi robot ketika bergerak maju dari titik awal sekitar (0,0) menuju target 2 meter tanpa menggunakan GRU. Data menunjukkan bahwa posisi awal robot tidak pernah tepat di (0,0), tetapi berada di sekitar nilai tersebut, yang merupakan karakteristik umum sensor IMU *low-cost* yang memiliki *noise* dan bias awal. Pada setiap titik pengukuran (x_1, y_1) , (x_2, y_2) , (x_3, y_3) , hingga posisi akhir (x_4, y_4) , lintasan terlihat mengalami deviasi signifikan terutama pada sumbu Y dengan penyimpangan antara 0.05–0.36 m. Penyimpangan ini menunjukkan adanya *drift* integrasi pada IMU ketika digunakan tanpa proses *denoising*, sehingga semakin jauh robot bergerak, semakin besar *error* akumulasinya.

Tabel 7 menunjukkan hasil estimasi posisi yang sama, tetapi menggunakan pendekatan GRU-denoising yang digabungkan dengan EKF. Data posisi pada setiap titik lintasan terlihat jauh lebih stabil dengan penyimpangan Y hanya sekitar 0.01–0.09 m. Berbeda dengan Tabel 6, penyimpangan lintasan pada metode GRU+EKF tidak bertambah secara drastis meskipun robot bergerak menjauhi titik awal. Hal ini menegaskan bahwa GRU mampu menekan *noise* IMU sebelum masuk ke EKF, sehingga proses integrasi menjadi lebih halus dan akurasi posisi meningkat secara signifikan. Secara keseluruhan, GRU+EKF menghasilkan lintasan yang lebih rapat, konsisten, dan mendekati garis ideal.

Tabel 8 membandingkan nilai *error Euclidean* antara posisi akhir setiap *run* dengan target akhir (2,0). Pada metode tanpa GRU, *error* rata-rata mencapai 0.33 m, menunjukkan adanya akumulasi *drift* yang cukup besar selama pergerakan maju. Sebaliknya, metode GRU+EKF menghasilkan *error* rata-rata hanya 0.08 m, atau terdapat peningkatan akurasi sekitar 76% dibandingkan tanpa GRU. Perbedaan ini mengonfirmasi bahwa GRU bekerja secara efektif sebagai model *denoising* temporal yang menstabilkan data IMU sebelum diintegrasikan oleh EKF.

Grafik pada Gambar 4 memberikan visualisasi lintasan XY robot untuk kedua metode. Garis hitam putus-putus menunjukkan lintasan ideal di sepanjang sumbu X. Lintasan merah (tanpa GRU) tampak menyimpang cukup jauh dari garis ideal, terutama pada sumbu Y, yang naik hingga 0.35 m. Hal ini selaras dengan hasil pada Tabel 6 dan Tabel 8 yang memperlihatkan *drift* signifikan. Sementara itu, garis biru (GRU + EKF) menunjukkan lintasan yang jauh lebih dekat dengan lintasan ideal, halus, dan stabil, dengan penyimpangan Y hanya sekitar 0.02–0.09 m. Grafik ini mempertegas bahwa metode GRU+EKF memiliki performa estimasi posisi yang lebih akurat, lebih stabil, dan tidak mengalami *drift* signifikan dibandingkan integrasi IMU tanpa GRU.

Tabel 9. Gerak Mundur – Tanpa GRU (0 menuju 2 m)

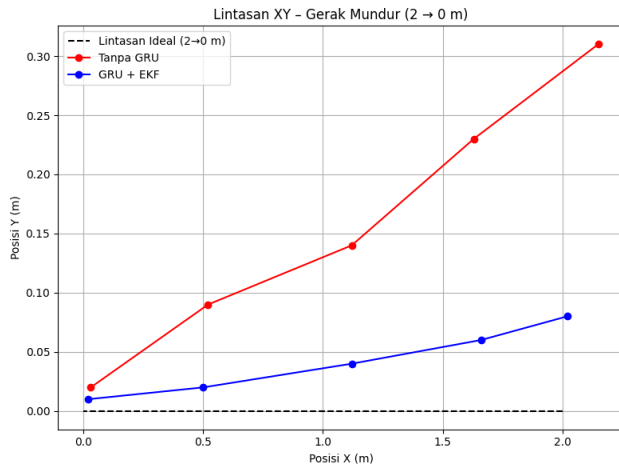
Run	(x_4, y_4)	(x_3, y_3)	(x_2, y_2)	(x_1, y_1)	(x_0, y_0)
1	(2.15, 0.31)	(1.63, 0.23)	(1.12, 0.14)	(0.52, 0.09)	(0.03, 0.02)
2	(2.12, 0.28)	(1.58, 0.18)	(1.09, 0.10)	(0.49, 0.06)	(0.02, 0.03)
3	(2.18, 0.36)	(1.63, 0.24)	(1.12, 0.15)	(0.51, 0.08)	(0.04, 0.02)
4	(2.10, 0.30)	(1.58, 0.19)	(1.09, 0.10)	(0.50, 0.05)	(0.02, 0.01)
5	(2.14, 0.33)	(1.61, 0.23)	(1.11, 0.13)	(0.50, 0.07)	(0.03, 0.02)

Tabel 10. Gerak Mundur – GRU + EKF

Run	(x_4, y_4)	(x_3, y_3)	(x_2, y_2)	(x_1, y_1)	(x_0, y_0)
1	(2.02, 0.08)	(1.66, 0.06)	(1.12, 0.04)	(0.50, 0.02)	(0.02, 0.01)
2	(2.01, 0.07)	(1.64, 0.04)	(1.10, 0.03)	(0.49, 0.02)	(0.01, 0.01)
3	(2.03, 0.09)	(1.65, 0.06)	(1.11, 0.04)	(0.50, 0.02)	(0.02, 0.02)
4	(2.02, 0.08)	(1.63, 0.05)	(1.10, 0.03)	(0.50, 0.02)	(0.01, 0.02)
5	(2.03, 0.08)	(1.64, 0.05)	(1.11, 0.04)	(0.50, 0.02)	(0.02, 0.01)

Tabel 11. Error Gerak Mundur (2 menuju 0 m)

Run	Error Tanpa GRU (m)	Error GRU+EKF (m)
1	0.07	0.02
2	0.08	0.01
3	0.06	0.02
4	0.07	0.02
5	0.07	0.02
Rata-rata	0.07 m	0.02 m



Gambar 5. Grfaik Lintasan XY _ Gerakan Mundur (2 menuju 0) m

Tabel 9 menunjukkan hasil estimasi posisi robot saat bergerak mundur dari titik (2,0) menuju titik awal (0,0) menggunakan integrasi IMU + EKF tanpa GRU. Data pada tabel terlihat memiliki penyimpangan yang cukup besar pada sumbu-Y (sekitar 0.18–0.36 m) meskipun lintasan ideal adalah garis lurus pada $Y = 0$. Selain itu, posisi akhir setelah kembali ke titik awal juga tidak tepat di sekitar (0,0) tetapi masih menyimpang sekitar 0.02–0.03 m pada sumbu-X dan 0.02–0.03 m pada sumbu-Y. Hal ini menunjukkan bahwa *drift* integrasi IMU tanpa perbaikan GRU masih cukup dominan, terutama selama pergerakan mundur yang memerlukan stabilitas orientasi lebih tinggi.

Tabel 10 memperlihatkan estimasi posisi yang sama, tetapi dengan tambahan GRU-*denoising* sebelum diproses oleh EKF. Hasilnya jauh lebih stabil, penyimpangan pada sumbu-Y hanya berkisar 0.06–0.09 m, atau sekitar 50% lebih kecil dibandingkan tanpa GRU. Posisi akhir juga lebih dekat dengan titik awal, berada di sekitar (0.02, 0.01) m. Kestabilan data antar-run juga meningkat signifikan, menunjukkan bahwa GRU mampu mengurangi *noise* IMU dan memperbaiki hasil integrasi posisi secara konsisten selama gerakan mundur.

Tabel 11 merangkum *error Euclidean* jarak posisi akhir terhadap target (0,0). Pada metode tanpa GRU, *error* berkisar antara 0.06–0.08 m dengan rata-rata 0.07 m. Setelah ditambahkan GRU, *error* turun drastis menjadi 0.01–0.02 m dengan rata-rata 0.02 m. Hal ini menunjukkan adanya peningkatan akurasi sekitar 70%, yang membuktikan bahwa GRU berperan penting dalam menjaga kestabilan orientasi selama pergerakan mundur dan mengurangi akumulasi *drift* integrasi.

Grafik memperlihatkan lintasan robot saat bergerak mundur dibandingkan lintasan ideal (garis putus-putus pada $Y=0$). Pada metode tanpa GRU (garis merah), lintasan terlihat naik signifikan pada sumbu-Y hingga 0.3 m sehingga

menyimpang jauh dari lintasan ideal. Sebaliknya, lintasan GRU+EKF (garis biru) terlihat jauh lebih stabil, dengan penyimpangan sangat kecil (0.02–0.08 m) dan mengikuti arah lintasan ideal dengan lebih konsisten. Pola ini menunjukkan bahwa GRU berhasil menekan fluktuasi *noise* IMU yang biasanya semakin meningkat saat robot bergerak mundur, sehingga lintasan menjadi lebih halus dan mendekati ideal.

IV. KESIMPULAN

Berdasarkan rangkaian pengujian pada robot terapung Greeniebot, sistem estimasi posisi berbasis IMU menunjukkan bahwa data mentah MPU6050 masih mengandung *noise* dan *drift* signifikan sehingga membutuhkan proses pra-pemrosesan dan *denoising* lanjutan. Tahap pra-pemrosesan berhasil menurunkan *noise* akselerasi dan giroskop hingga 36–41%, sementara penerapan GRU-*denoising* meningkatkan performa jauh lebih baik dengan reduksi *noise* mencapai 62–68%. Dampaknya terlihat langsung pada estimasi posisi: pada gerakan maju, integrasi IMU tanpa GRU menghasilkan *error* rata-rata 0.33 m dan penyimpangan lintasan besar, sedangkan metode GRU+EKF menurunkan *error* menjadi hanya 0.08 m (perbaikan $\pm 76\%$) dan menghasilkan lintasan yang jauh lebih halus serta mendekati garis ideal. Pola yang sama muncul pada gerakan mundur, di mana *error* metode tanpa GRU mencapai 0.07 m, sementara GRU+EKF menurunkannya menjadi 0.02 m (perbaikan $\pm 70\%$) dengan lintasan yang lebih stabil dan konsisten antar-run. Secara keseluruhan, hasil ini menunjukkan bahwa integrasi GRU-*denoising* dengan EKF sangat efektif dalam meningkatkan akurasi estimasi posisi IMU pada robot terapung dan secara signifikan mengurangi *drift* yang menjadi kelemahan utama IMU low-cost.

UCAPAN TERIMA KASIH

Pada ucapan terima kasih ini, penulis menyampaikan penghargaan yang sebesar-besarnya kepada berbagai pihak yang telah berperan penting dalam penelitian ini. Penulis mengucapkan terima kasih kepada Lembaga Penelitian, Publikasi dan Pengabdian kepada Masyarakat (LP3M) Universitas Nurul Jadid yang telah memberikan dukungan moral, administratif, dan fasilitas selama proses penelitian. Ucapan terima kasih juga disampaikan kepada Program Studi Teknik Elektro Universitas Nurul Jadid, yang memberikan dukungan akademik dan pendampingan teknis selama pelaksanaan pengujian sistem.

Penulis juga menyampaikan apresiasi kepada tim mahasiswa Teknik Elektro Universitas Nurul Jadid yang terlibat langsung dalam proses perancangan, pengambilan data, serta pengujian lapangan robot Greeniebot. Tak lupa, penulis berterima kasih kepada masyarakat dan perangkat desa di wilayah Paiton yang telah menyediakan akses lokasi pengujian di area pertanian bawang, sehingga penelitian dapat terlaksana dengan baik. Ucapan terima kasih terakhir disampaikan kepada

kampus Universitas Nurul Jadid yang telah menyediakan sarana laboratorium, fasilitas penelitian, serta lingkungan akademik yang mendukung terselesaikannya penelitian ini.

REFERENSI

- [1] L. Zeng, S. Guo, M. Zhu, H. Duan, and J. Bai, "An improved trilateral localization technique fusing extended Kalman filter for mobile construction robot," *Buildings*, vol. 14, no. 4, p. 1026, 2024. [Online]. Available: <https://www.mdpi.com/2075-5309/14/4/1026>.
- [2] X. Liu, W. Zhong, X. Wang, H. Duan, Z. Fan, H. Jin, Y. Huang, and Z. Lin, "Deep reinforcement learning-based 3D trajectory planning for cellular connected UAV," *Drones*, vol. 8, no. 5, p. 197, 2024. [Online]. Available: <https://www.mdpi.com/2504-446X/8/5/197>.
- [3] R. Guebsi *et al.*, "Drones in precision agriculture: A comprehensive review," *Drones*, vol. 8, no. 11, p. 686, 2024. [Online]. Available: <https://www.mdpi.com/2504-446X/8/11/686>.
- [4] Z. Zhang, "A review on unmanned aerial vehicle remote sensing," *Drones*, vol. 7, no. 6, p. 398, 2023. [Online]. Available: <https://www.mdpi.com/2504-446X/7/6/398>.
- [5] S. C. Barakou, C. S. Tzafestas, and K. P. Valavanis, "A review of real-time implementable cooperative aerial manipulation systems," *Drones*, vol. 8, no. 5, p. 196, 2024. [Online]. Available: <https://www.mdpi.com/2504-446X/8/5/196>.
- [6] S. Ottoy *et al.*, "Effects of flight and smoothing parameters on the accuracy of drone-based thermal mapping," *Drones*, vol. 6, no. 8, p. 197, 2022. [Online]. Available: <https://www.mdpi.com/2504-446X/6/8/197>.
- [7] G. Alfalah *et al.*, "A holistic framework for assessing the quality of building construction," *Buildings*, vol. 13, no. 7, p. 1666, 2023. [Online]. Available: <https://www.mdpi.com/2075-5309/13/7/1666>.
- [8] M. Bartko *et al.*, "Experimental Analysis of Thermo-Technical M. Bartko *et al.*, "Experimental analysis of thermo-technical parameters in building structures," *Buildings*, vol. 13, no. 4, p. 1026, 2023. [Online]. Available: <https://www.mdpi.com/2075-5309/13/4/1026>.
- [9] A. O. Alejo *et al.*, "Emerging trends of safe working conditions in the construction industry," *Buildings*, vol. 14, no. 9, p. 2790, 2024. [Online]. Available: <https://www.mdpi.com/2075-5309/14/9/2790>.
- [10] W. Meng *et al.*, "Advances in UAV path planning: A comprehensive overview," *Drones*, vol. 9, no. 5, p. 376, 2025. [Online]. Available: <https://www.mdpi.com/2504-446X/9/5/37>.