

Performance Evaluation of Direction Cosine Matrix and Madgwick's as 3D Orientation Estimation Algorithm for Estimating Ground Contour

Eri Prasetyo Wibowo (✉ eri@staff.gunadarma.ac.id)

Universitas Gunadarma <https://orcid.org/0000-0002-6146-1324>

Imam Purwanto

Universitas Gunadarma

Dennis Aprilla Christie

Universitas Gunadarma

Purnawarman Musa

Universitas Gunadarma

Busono Soerowirdjo

Universitas Gunadarma

Research Article

Keywords: DCM, Madgwick, Orientation, Trajectory.

Posted Date: July 7th, 2020

DOI: <https://doi.org/10.21203/rs.3.rs-34412/v1>

License:  This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Performance Evaluation of Direction Cosine Matrix and Madgwick's as 3D Orientation Estimation Algorithm for Estimating Ground Contour

Eri Prasetyo Wibowo¹, Imam Purwanto², Dennis Apriilla Christie³, Purnawarman Musa⁴, Busono Soerowirdjo⁵

Doctoral Program In Information Technology Gunadarma University

Jl. Margonda Raya No. 100, Depok 16424, Jawa Barat, Indonesia

{eri¹,imampur², dennis_christie³, p_musa⁴,busono⁵}@staff.gunadarma.ac.id

Abstract—Contour interval is elevation difference between two contour lines that are adjacent and parallel. Ground Contour interval is an important information that must be provided in a construction. Ground contour interval estimation problem could be viewed as a 3D trajectory estimation problem. One framework, called dead reckoning, is often used for motion (position and orientation) tracking of a mobile systems over time. Dead reckoning requires a precise 3D orientation estimation algorithm. Two competing algorithms are Direction Cosine Matrix and Madgwick's. To ensure the 3D trajectory estimation gives an accurate result, these two algorithms must be carefully and thoroughly evaluated. This research is part of a complete research on the development of a ground contour estimation and only focuses on comparison between two 3D orientation estimation algorithm.

Keywords—DCM, Madgwick, Orientation, Trajectory.

I. INTRODUCTION

A. Background

Measurement of ground contour maps is a common activity in geomatics concentration. One of the information that can be derived from ground contour maps is the contour interval, which is an elevation difference between two adjacent and parallel contour lines [1,8]. Contour interval is an important information that must be provided in the construction development [2,9]. Without it, building construction design planning becomes inaccurate and might results an unstable, not sturdy building. Unfortunately, contour intervals can't provide any information about the height difference between within two contour lines which involved. This is due to the sparsity of the contour lines in the map. The resolution provided by the contour map also varies depending on its usage.

Ground interval contour estimation problem (henceforth, will be referred as ground contour estimation), can be viewed from another perspective, namely three-dimensional (3D) trajectory estimation of a system. 3D trajectory estimation is one branch of computer science that is able to produce a collection of 3D point coordinates in space that are able to represent trace or movement of the system while the system is working [3,10].

One topic that intersects with this problem is called Dead Reckoning (DR). DR is a movement tracking technique of a mobile systems, it tracks the relative position and orientation from time to time. DR is also able to describe the full 3D orientation of the system [4,11], however, over time, the error given by the sensor in each measurement moment will accumulate and would lead to exaggerated error [5]. Measurement errors that is suffered by DR can actually be minimized by implementing a more accurate orientation estimation algorithm. There are two

algorithms that are commonly used in the world of robotics, namely Direct Cosine Matrix (DCM) [6,4,12], and Madgwick's algorithm [7,13].

This research is part of a complete research on the development of a ground contour estimation. This study focuses on the implementation and an in-depth performance evaluation of the two 3D orientation estimation algorithms, namely DCM and Madgwick's.

B. Problem Reformulation

Based on the background, some problems can be formulated into several points as follows:

- 1) How to estimate the 3D orientation of the system?
- 2) How can the algorithm testing be performed?
- 3) How can the algorithm be evaluated?

C. Objectives

Based on the formulation of the problem, some of the objectives of this study can be formulated into several points as follows:

- 1) Implement two 3D orientation estimation algorithms, Direct Cosine Matrix and Madgwick's.
- 2) Using the Inertial Measurement Sensor (IMU) dataset which is available to test whether the algorithm has been implemented properly.
- 3) Determine a proper metrics for evaluating numerically (quantitatively) to the two algorithms involved.

D. Contribution

This paper will perform a thorough performance evaluation between orientation estimation algorithms, DCM and Madgwick's, and summarizes the results which would be helpful for future research that involves this algorithm.

II. LITERATUR REVIEW

A. Orientation Representation

Orientation of a system can be described by several representations, including matrix representation and quaternions representation. Each has its own unique property requirement and operating rules.

1) Rotation Matrix

In linear algebra, rotation matrix is a matrix that is used to perform a rotation in Euclidean space. Rotation matrix that is used to represent an orientation in 3D space is a 3x3 matrix. Rotation matrix has a special properties, which are *orthogonal* and *normal*.

$$R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}$$

2) Quaternions

Quaternion is a four-dimensional (4D) number system that extends the definition of complex two-dimensional (2D) numbers. Similar to the rotation matrix, *quaternions* are used to perform a rotation in *Euclidean* space, but it has different set of rules of operations and properties. Quaternions should be represented by a 4D vector with a length of 1. Where $Q = [q_w \ q_x \ q_y \ q_z]$

B. Direction Cosine Matrix

A summary of the DCM algorithm is described in Algorithm 1 below. Details of DCM orientation estimation algorithm can be found in its original publication [6].

Algorithm 1 Orientation Estimation with DCM

Input:

Orientation at $t - 1$ in rotation matrix $\hat{R}_{t-1} \in SO(3)$

Current gravitation acceleration $a_t = [a_x, a_y, a_z] \in \mathbb{R}^3$,

Current angular velocity $\omega_t = [\omega_x, \omega_y, \omega_z] \in \mathbb{R}^3$,

Current magnetic field $m_t = [m_x, m_y, m_z] \in \mathbb{R}^3$

Time interval $dt \in \mathbb{R}$

Output:

Current orientation in rotation matrix $\hat{R}_t \in SO(3)$

Process:

- 1 $\omega' \leftarrow$ Koreksi error dengan proporsional dan integral term
 - 2 $[d\theta]_x \leftarrow$ Build a skew-symetric matrix of the gyroscope-based orientation (ω')
 - 3 $R_t \leftarrow$ Update the rotation matrix
 - 4 $\hat{R}_t \leftarrow \hat{X}, \hat{Y}, \hat{Z} \leftarrow$ Normalize the rotation matrix
 - 5 $e_{rp}, e_y \leftarrow$ Error detection (drift) for roll-pitch and yaw
-

C. Madwick's Algorithm

A summary of the Madgwick's algorithm is described in Algorithm 2. Details of Madgwick's orientation estimation algorithm can be found in its original publication [6], [13].

Algorithm 2 Orientation Estimation with Madgwick's

Input:

Orientation at $t - 1$ in Quaternions $q_{t-1} = [w, x, y, z] \in \mathbb{R}^4, \|q\| = 1$,

Current gravitation acceleration $a_t = [a_x, a_y, a_z] \in \mathbb{R}^3$,

Current angular velocity $\omega_t = [\omega_x, \omega_y, \omega_z] \in \mathbb{R}^3$,

Current magnetic field $m_t = [m_x, m_y, m_z] \in \mathbb{R}^3$.

Output:

Current orientation in Quaternions $q_t = [w, x, y, z] \in \mathbb{R}^4, \|q\| = 1$,

Proses:

- 1 $\dot{\hat{q}}_t \leftarrow$ Calculate *quaternions rate of change* with gyro
 - 2 $\dot{\hat{q}}_{\omega,t} \leftarrow$ Calculate *quaternions orientation* with gyro
 - 3 $L \leftarrow$ Calculate *Loss Function* $f_{g,b}$
 - 4 $J \leftarrow$ Calculate *Jacobian Matrix* $J_{g,b}$
 - 5 $\dot{\hat{q}}_{v,t} \leftarrow$ Perform *Gradient Descent* at q_{t-1}
 - 6 $\dot{\hat{q}}_{est,t} \leftarrow$ Fuse two quaternion measurements, $\dot{\hat{q}}_{\omega,t}$ dan $\dot{\hat{q}}_{v,t}$
 - 7 $\dot{\hat{q}}_{est,t} \leftarrow$ Normalize quaternions $\dot{\hat{q}}_{est,t}$
-

III. COMPARISON FRAMEWORK

To observe the performance of both Algorithm 1 and Algorithm 2, a test using real data must be performed. Secondary data, derived from [7], are used as input for both of the algorithms.

The available dataset is a pair of files with CSV (comma separated values) extension. The first file, *dataset.csv*, is a file that contains raw data from 9 Degree of Freedom (DoF) IMU measurements. The *dataset.csv* file has 6960 data rows including headers. The second file, *groundtruth.csv*, is a file that contains groundtruth in euler space. The number of rows of *groundtruth.csv* file has the same number as in the *dataset.csv* file.

TABLE 1 is a description of IMU measurement scenario within the dataset. The scenario is divided into seven steps / stages, arranged in such a way that all IMU's basic orientation are read.

TABLE 1 Scenario Description within Dataset.

Stage	Description
1	The IMU sensor is first placed statically
2	The IMU is moved 90° to the X axis, stops, - 90° to the X axis, stops.
3	The IMU was returned to its original position.
4	The IMU is moved 90° to the Y axis, stopped, - 90° to the Y axis, stopped.
5	The IMU was returned to its original position.
6	The IMU is moved 90° to the Z axis, stopped, - 90° to the Z axis, stopped.
7	The IMU was returned to its original position.

FIGURE 1 is a visualization graph from gyroscope, accelerometer, and magnetometer, sequentially from top to bottom, which derived from the dataset. Red, green, blue lines represent measurements on the X, Y, and Z axis respectively.

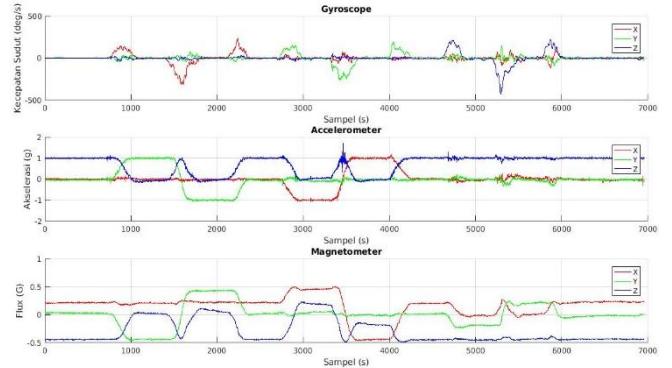


FIGURE 1 Values from *dataset.csv* (gyroscope, accelerometer, magnetometer) visualized as a graph.

Performance evaluation and comparison of the two algorithms is done using MATLAB 2016b software on a computer with Ubuntu Operating System 14.04 LTS with specifications of i5-6200U @ 2.3GHz CPU and 4 GB RAM.

A. Qualitative Evaluation

Qualitative evaluation is performed by representing the collection of estimation data and groundtruth into the same graph.

An algorithm that is said to have low performance is when there is a significant difference between the groundtruth orientation line and the estimated orientation line, indicating the algorithm is unable to estimate the orientation according the actual situation.

B. Quantitative Evaluation

To conduct a quantitative evaluation, orientation estimation error rate calculation to the groundtruth at each given moment must be performed. Errors can be calculated using the error function defined in equation 1 below.,

$$E(Q, \hat{Q}) = \frac{1}{N} \sum_{i=1}^N \|Q_i \otimes \hat{Q}_i^{-1} - \mathbf{1}\|_2 \quad (1)$$

where $\|\cdot\|_2$ is an l2-norm, $\mathbf{1} \in \mathbb{R}^4$ is an identity quaternion which has value of 1 for the real component (w) and 0 for all imaginary component (x, y, z), $Q \in \mathbb{R}^4$ is a ground truth orientation in quaternion, \otimes symbol is quaternion product, and $\hat{Q} \in \mathbb{R}^4$ is an orientation estimation in quaternion with superscript -1 represent the inverse of quaternion that is defined as,

$$\hat{Q}_i^{-1} = \frac{\hat{Q}_i^*}{\|\hat{Q}_i\|_1} \quad (2)$$

where \hat{Q}^* is a quaternion conjugate from \hat{Q} and $\|\cdot\|_1$ is an l1-norm from \hat{Q} .

$Q_i \otimes \hat{Q}_i^{-1}$ in the equation 1 will produce an identity quaternion when the estimation (\hat{Q}) precisely predict the true orientation (Q_i) without any mistakes. This condition is the most optimal condition, so that the difference between $Q_i \otimes \hat{Q}_i^{-1}$ with the identity quaternion ($\mathbf{1}$) will gives zero.

The error function defined by equation 1, shows that the performance of the corresponding orientation estimation algorithm, i.e. the smaller the value that is calculated using equation 1, the more accurate the estimation made by the algorithm; and vice versa. Value range produced by the error function is zero (0) to one (1).

IV. RESULTS AND EVALUATION

A. Orientation Estimation Results

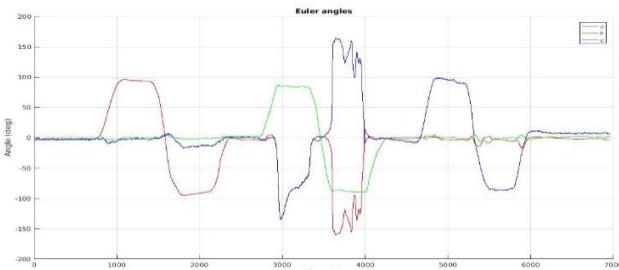


FIGURE 2 Orientation Estimation (in euler) Generated Using Algorithm 1, DCM.

The IMU measurement dataset (see FIGURE 1) at its nine degree of freedom is used as the input for Algorithm 1 (DCM) and Algorithm 2 (Madgwick's). Both of these algorithms produce outputs in the form of collection of orientation estimation in euler angle. These data collection is represented in FIGURE 2 and FIGURE 3

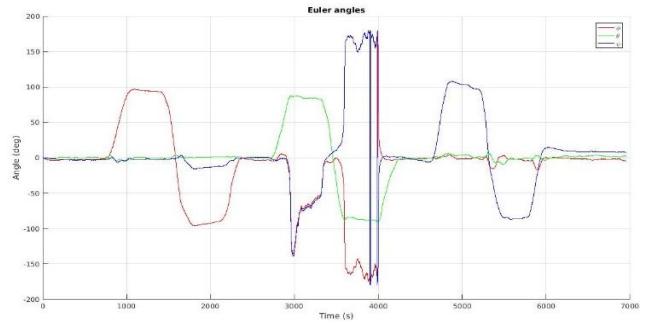


FIGURE 3 Orientation Estimation (in euler) Generated Using Algorithm 2, Madgwick's.

The red, green, and blue lines (see FIGURE 2 and FIGURE 3) represent the orientation estimation for the X, Y, and Z axis, respectively. It can bee seen in the graph, there are three main waves that are formed, each of which is the representation of stage number 2, 4, and 6 that is conducted (see TABLE 1).

B. Qualitative Evaluation

Performance evaluation of Algorithm 1 and Algorithm 2 can be seen by comparing the actual orientation (groundtruth) with the estimated orientation in one graph (see FIGURE 4a and FIGURE 4b). Qualitatively, the estimation (red line) is almost always coincides with the groundtruth (blue line), especially in the three main waves mentioned earlier. This indicates that Algorithm 1 and Algorithm 2 are able to estimate the orientation of the system properly.

Exciting things happening in sample ± 3000 to ± 4000 , i.e. there is a sudden spike that only happens in the estimated roll (FIGURE 4 - first column) and estimated yaw (FIGURE 4 - third column). This spike was not found in the estimated pitch (FIGURE 4 - second column).

This spike is an artifact of a gimbal-lock, a phenomenon that only occurs in euler angle representation, which is a specific situation where euler angle is not able to represent the full 3D orientation. The specific situation mentioned is when the system orientates itself towards 90° on the Y axis (pitch). This spike does not mean that Algorithm 1 and Algorithm 2 fail to estimate orientation, it just means that euler angle representation is not able to decribe a situation when the system orientates it self 90° on the Y axis (pitch).

Euler angle is indeed the most intuitive orientation representation of a system to be analyzed by humans, but due to limitation possesed by this representation, the performance of Algorithm 1 and Algorithm 2 should be done using another orientation representation. FIGURE 5 and FIGURE 6 are orientation estimation and groundtruth comparison in quaternion representation. Red, green, blue, magenta lines are the w, x, y, z components of quaternions, respectively.

The results of qualitative evaluation (graphical) do not show significant differences between the two algorithms. Although if we look carefully, there is actually a slight

deviation performance, but it is quite difficult to see by human eye. Therefore, quantitative testing had to be conducted.

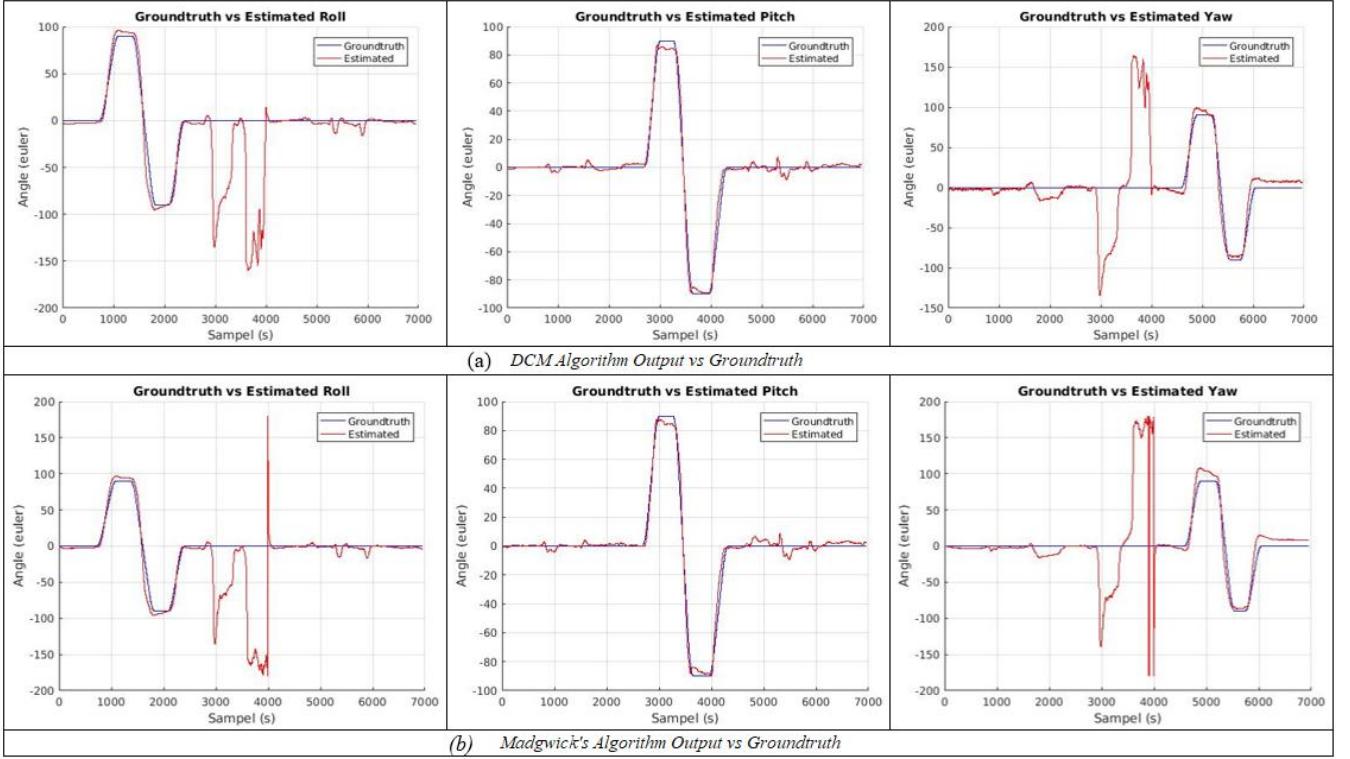


FIGURE 4 DCM and Madgwick's Orientation Estimation Comparison with Groundtruth using Euler Representation.

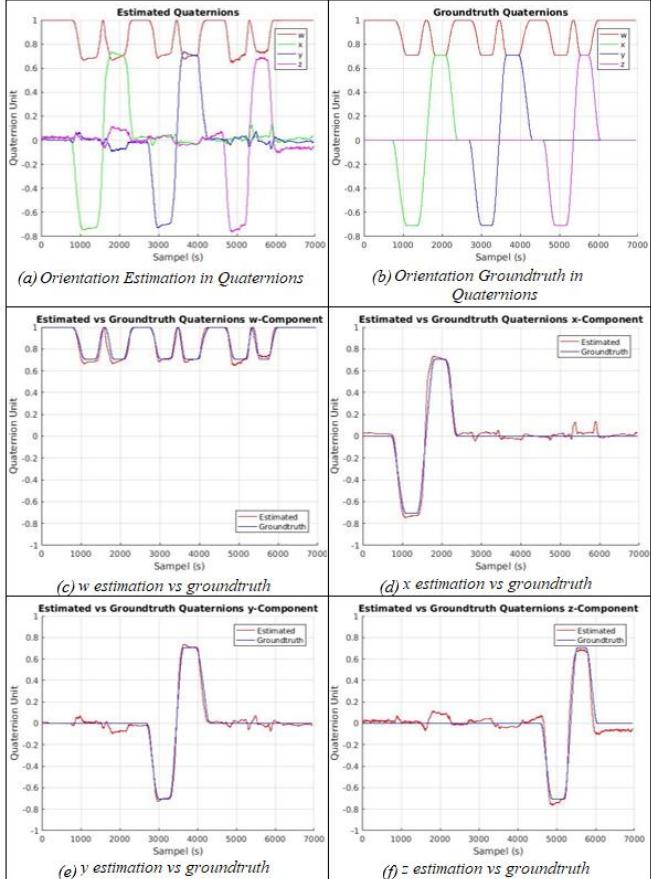


FIGURE 5 DCM Orientation Estimation Comparison with Groundtruth using Quaternion Representation.

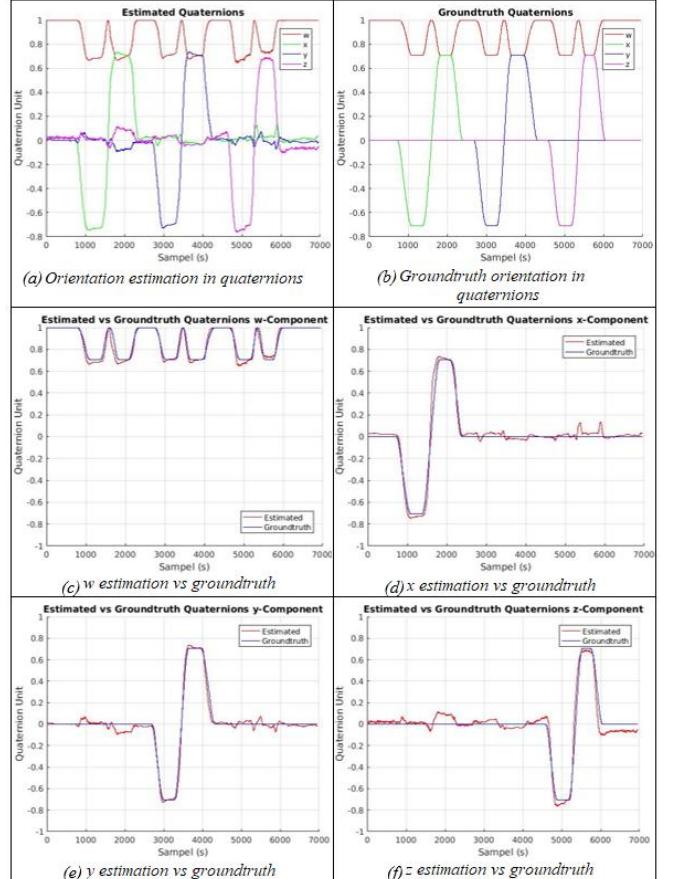


FIGURE 6 Madgwick's Orientation Estimation Comparison with Groundtruth using Quaternion Representation.

C. Quantitative Evaluation

The quantitative analysis was conducted in two related algorithms. To strengthen the analysis results, the two algorithms are compared at different noise levels. Noise reference (reference / baseline noise) became the basis for the comparison. Baseline noise is estimated by measuring the standard deviation (σ) of the mean-normalized data on each axis (x, y, z) on each sensor (accelerometer, gyroscope, magnetometer). FIGURE 7 shows the original measurements (red lines) and measurements with noise (green lines) at two noise levels, 2σ and 5σ .

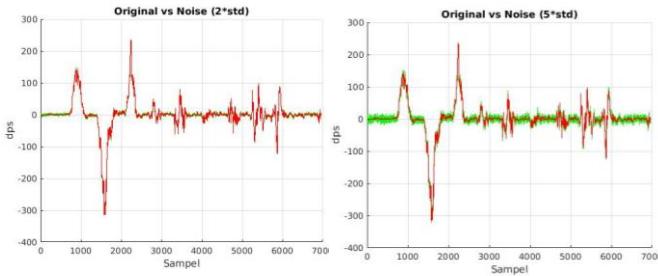


FIGURE 7 Comparison between Original and Noise Added Measurement with 2σ (left) and 5σ (right)

FIGURE 8 shows the comparison of error values owned by the two algorithms using dataset inputs that are subject to 11 different noise levels, from 0σ to 5σ . Estimation results in each noise level are compared with groundtruth using the error function (equation 1). To ensure quantitative measurements, the two algorithms were run n times ($n = 10$ in this study) so as to produce a collection of n error values. The mean of the error is plotted into a graph and becomes the reference for this quantitative analysis.

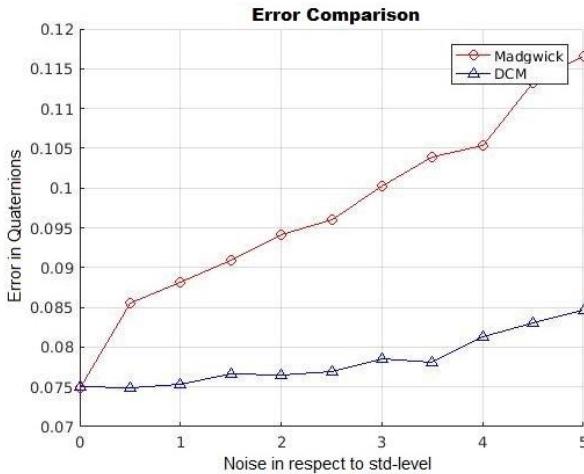


FIGURE 8 DCM and Madwick's Comparison in Resistance to Noise.

It is shown by FIGURE 8, the DCM algorithm (blue line) tends to always have a lower error value than the Madwick algorithm (red line). Therefore, it can be concluded that the DCM algorithm has the advantage in resistance to noise.

FIGURE 9 shows the comparison of the duration required for each algorithm in completing its operations for all data available in the dataset. Duration is measured in seconds. The blue bar indicates the duration of pure

operation required by each algorithm, i.e. without any data storage into variables or other things. The green bar indicates the operation duration with a storing process of the estimation orientation results into a variable. The yellow bar shows the operation duration with a transforming process from the body-frame coordinate system to the world-frame coordinate system (a process that must be done if we want to represent the orientation in the observer coordinate system).

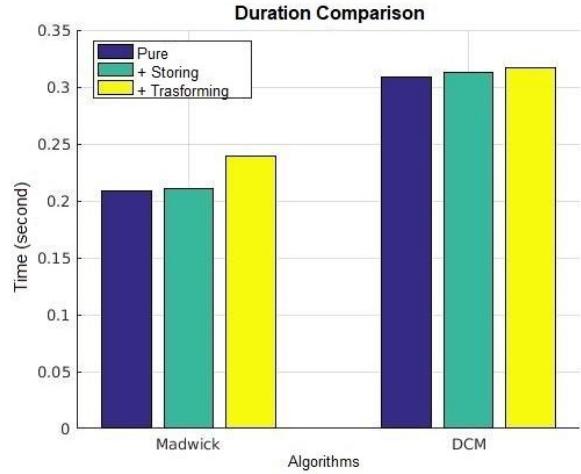


FIGURE 9 Comparison of Algorithm 1 and Algorithm 2 in the Level of Execution Speed.

As can be seen from FIGURE 9, the Madwick algorithm (left) is superior in speed compared to the DCM algorithm (right). This conclusion is consistent with one of the claims in the original publication of Madwick's Algorithm [6].

FIGURE 10 shows the comparison of the amount of memory needed for each algorithm to store the entire collection of estimated system orientation values from the dataset. Memory size is measured in bytes.

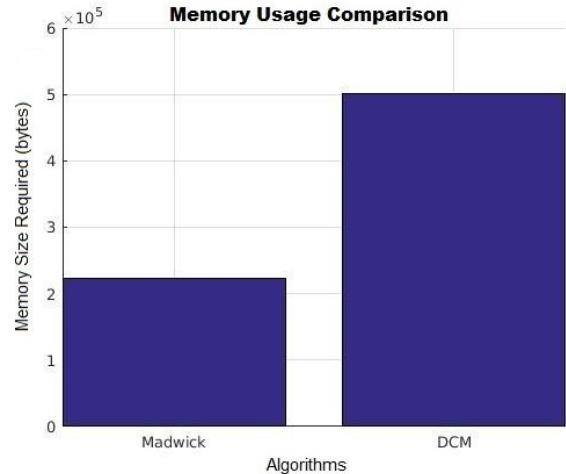


FIGURE 10 Comparison of Algorithm 1 and Algorithm 2 in the Amount of Memory Needed.

As can be seen from FIGURE 10, the Madwick algorithm (left) is superior in memory efficiency compared to the DCM algorithm (right). This conclusion also agrees with one of the claims in the original publication. This is because Madwick's works in quaternion space, which is a

four dimensional vector (requires 4 floating point values) while DCM works using a 3×3 transformation matrix (requires 9 floating point values).

Generally, in this study, the two orientation estimation algorithms are compared using three dimension of evaluations, namely error rate (and their resistance to noise), the execution duration, and the amount of memory needed. These three comparison is summarized in the TABLE 2 below, with (+) sign indicates superiority and (-) sign indicates inferiority.

TABLE 2 Algorithm Comparison Summary

Evaluation	DCM	Madgwick
Error + Noise	(+)	(-)
Execution Duration	(-)	(+)
Memory Efficiency	(-)	(+)

V. CONCLUSION AND FUTURE WORK

To perform ground contour estimation, one part of the algorithm that must be evaluated carefully is the orientation estimation algorithm. This algorithm is the key to estimating 3D coordinates / system trajectories, in other words, ground contours. Two algorithms that are evaluated in detail in this study are (1) the Direct Cosine Matrix (DCM) Algorithm, and (2) Madgwick Algorithm.

The test was conducted using the IMU sensor measurement dataset and the evaluation was performed with two different approach, qualitative and quantitative. Qualitatively, both algorithms are able to estimate orientation based on predetermined IMU measurement data. The difference is not significant, so quantitative testing is carried out.

Based on three quantitative tests, the DCM algorithm is superior in accuracy and resistance to noise, while Madgwick's algorithm is superior in execution duration and memory efficiency. Each algorithm does have advantages and disadvantages, but the conclusions from this test can be drawn as follows;

1. If the available hardware has a small and limited specification (example: a microcontroller), Madgwick's algorithm becomes a preference,
2. In contrast, if the available hardware has high specifications and is flexible (for example: a computer), the DCM algorithm becomes a preference.

Because for future research (estimation of land contours), a computer will be used as the machine for applying the algorithm, DCM algorithm was chosen to be the orientation estimation algorithm. The main priority in this research is to produce contour plots with a high degree of accuracy. The duration and amount of memory are the second priority because computer specifications can be increased according to algorithm requirements.

DECLARATIONS

* Ethics approval and consent to participate
Manuscript reporting studies is not involving human participants, human data or human tissue
“Not applicable” in this section”

* Consent for publication
Our research does not involve human participation; our study only compares a number of algorithms to develop more optimization algorithms
“Not applicable” in this section”

* Availability of data and material
All data is taken from the experiment itself using the equipment that we made. The data used for the experiment is not harmful to the public. Ground contour data for simulation data is used to test algorithm optimization
“Not applicable in this section”

* Competing interests
Currently it is not receiving research funding from the Indonesian Government due to the covid-19 outbreak. The publication fee will be carried out independently by authors, , so we ask for a relief in financing. and our request was approved by Springer without paying publication fees.

* Funding
The publication fee will be carried out independently by authors, , so we ask for a relief in financing. and our request was approved by Springer without paying publication fees.

* Authors' contributions
All authors contribute to research and writing as well as revisions and suggestions for improving the publication of the paper

* Acknowledgements
First we would like to thank the Gunadarma University that allows the use of laboratories. and software to do algorithm simulation.
Secondly, we would like to thank the Springer Nature Waivers team for approving a request for relief in financing publications in this journal



* Authors' information (optional)

Eri Prasetyo Wibowo born in Kendal, Indonesia, in 1966. He received his B.S. degree in electronics and instrumentation from Gadjah Mada University, Indonesia, in 1992, his M.S. degree in information system from Gunadarma University, Indonesia, in 1994, and his Ph.D. degree in electronics informatics from Burgundy University, France, in 2005. He is a member of IEEE, ACM, Professional Organization in the field of Information and Computer Technology (IPKIN), Indonesian Association of Higher Education in Informatics and Computing (APTIKOM). He has worked as a secretary of the doctoral program in information technology at Gunadarma University. He was a member of the EACOVIROE Project to promote masters in VIBOT, which was founded by Erasmus Mundus. His current interests include real time image processing applications and system on chips design.



Imam Purwanto born in Jakarta, Indonesia, in 1972. He received his B.S. degree in Computer System from Gunadarma University, Indonesia, in 1997; his M.S. degree in Information Management System from Gunadarma University, Indonesia, in 2000; and his doctoral degree in Computer Science from Gunadarma University, Indoensia, in 2020. He worked as an academic service administration staff and as a lecturer in Information System Engineering at Gunadarma University since 1997. His current research interest includes robotic navigation system.



Dennis Aprilla Christie born in Jakarta, Indonesia, in 1993. He received his B.S. degree in Informatics Engineering from Gunadarma University, Indonesia, in 2014, his M.S. degree in Information Management System from Gunadarma University, Indonesia, in 2015, and his M.S. degree in Computer Science from Université de Bourgogne Franche Comté, France, in 2016. He is a lecturer in computer science in Gunadarma University and currently conducting his doctoral degree research in Gunadarma University. His current research interests include 3D computer vision, robotics, machine learning and deep learning.



Purnawarman Musa born in Gorontalo, Indonesia, in 1975. He received his B.CS. Degree in Computer Science from Gunadarma University, Indonesia, in 1999, his M.T. degree in Electronics and Telecommunication from Gunadarma University, Indonesia, in 2006, and his Ph.D. degree in Electronics Informatics from Burgundy University, France, in 2013. He is a member of the Professional Organization in the field of Information and Computer Technology (IPKIN), Indonesian Association of Higher Education in Informatics and Computing (APTIKOM). He has worked as a Head of the Research Center Aeronautics and Robotics at Gunadarma University. His current interests include the Internet of Things, Control and Mechatronics System, Artificial Intelligent, Real-Time Image Processing Applications, Sensing, Embedded System, and System on Chips Design..



Busono Seerowirdjo was born in Purworejo, Indonesia in 1944. He received B.S degree in Electrical Engineering from ITB Bandung Indonesia in 1972. The M.S degree in Electronics design from Southampton University, England in 1975. The PhD degree in Microelectronics from Southampton University, England in 1983. He was Professor in Microelectronics from Gunadarma University. He was as Director of Doctoral program in Information technology at Gunadarma University Indonesia. His current interest is in Microelectronics Design.

BIBLIOGRAPH

- [1] Jenks, G.F., Steinke, T., Buchert, B. and Armstrong, L., 1971. Illustrating the concepts of the contour symbol, interval, and spacing via 3-D maps. *Journal of Geography*, 70(5), pp.280-288.
- [2] Yu, C., Lee, J.A.Y. and Munro-Stasiuk, M.J., 2003. Extensions to least-cost path algorithms for roadway planning. *International Journal of Geographical Information Science*, 17(4), pp.361-376.
- [3] Ohno, K., Nomura, T. and Tadokoro, S., 2006, October. Real-time robot trajectory estimation and 3d map construction using 3d camera. In 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems (pp. 5279-5285). IEEE.
- [4] P. Musa, D. Christie, E.P. Wibowo, 2016, "An implementation of direction cosine matrix in rocket payload dynamics attitude monitoring", International Conference on Informatics and Computing.
- [5] Arraigada, M. and Partl, M., 2006, March. Calculation of displacements of measured accelerations, analysis of two accelerometers and application in road engineering. In 6th Swiss transport research conference (No. 0.5, p. 1)
- [6] Premerlani, W. and Bizard, P., 2009. Direction cosine matrix imu: Theory. *Diy Drone: Usa*, pp.13-15
- [7] Sebastian O.H. Madgwick, 2010, "An efficient orientation filter for inertial and inertial/magnetic sensor arrays", http://www.samba.org/tridge/UAV/madgwick_internal_report.pdf.
- [8] Teck Wee Chua and Li Shen, 2017, "Contour Detection from Deep Patch-level Boundary Prediction", *IEEE 2nd International Conference on Signal and Image Processing*.
- [9] Xiao Yang, Rudolf Seethaler, Chengpeng Zhan, Dun Lu , Wanhuai Zhao,2019. A Novel Contouring Error Estimation Method for Contouring Control. *IEEE/ASME Transactions on Mechatronics*, 24(4), pp. 1902-1907.
- [10] Jianqi Zhong, Hao Sun, Wenming Cao , Zhihai He, 2020. Pedestrian Motion Trajectory Prediction With Stereo-Based 3D Deep Pose Estimation and Trajectory Learning. *IEEE Access*, 8, pp. 23480-23486.
- [11] E. TURQUIN , L. PETRUSCA , M. VIALLON , H. LIEBGOTT , F. VARRAY, 2019, "Full 3D anisotropic estimation of tissue in ultrasound imaging", *IEEE International Ultrasonics Symposium (IUS)*.
- [12] Fangjun Qin, Lubin Chang, Sai Jiang, 2018," Inertial Navigation System Calibration based on Direction Cosine Matrix Matching", *IEEE International Conference of Intelligent Robotic and Control Engineering (IRCE)*.
- [13] S.A. Ludwig K.D. Burnham,2018," Comparison of Euler Estimate using Extended Kalman Filter, Madgwick and Mahony on Quadcopter Flight Data", *International Conference on Unmanned Aircraft Systems (ICUAS)*.

Figures

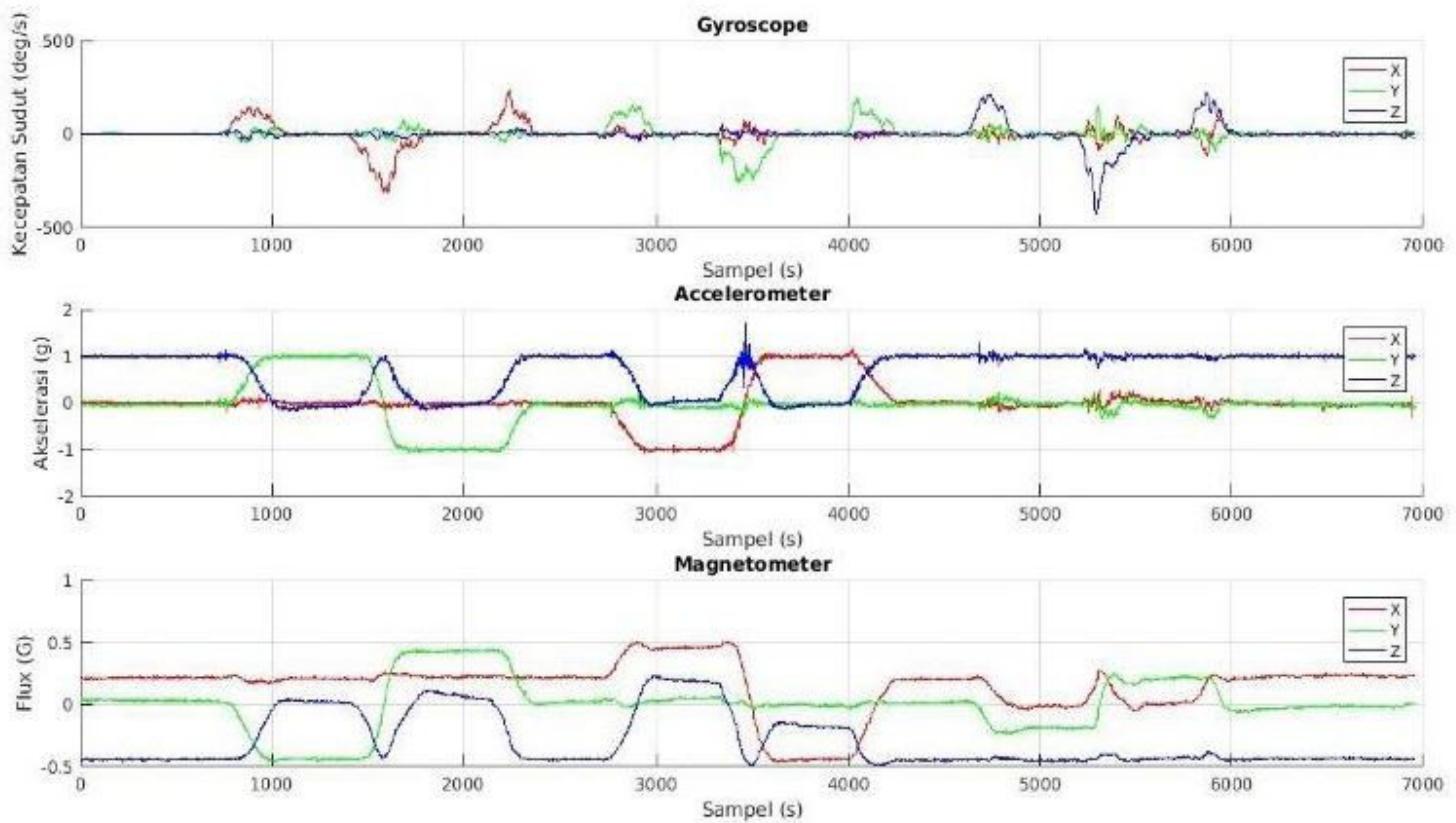


Figure 1

Values from dataset.csv (gyroscope, accelerometer, magnetometer) visualized as a graph.

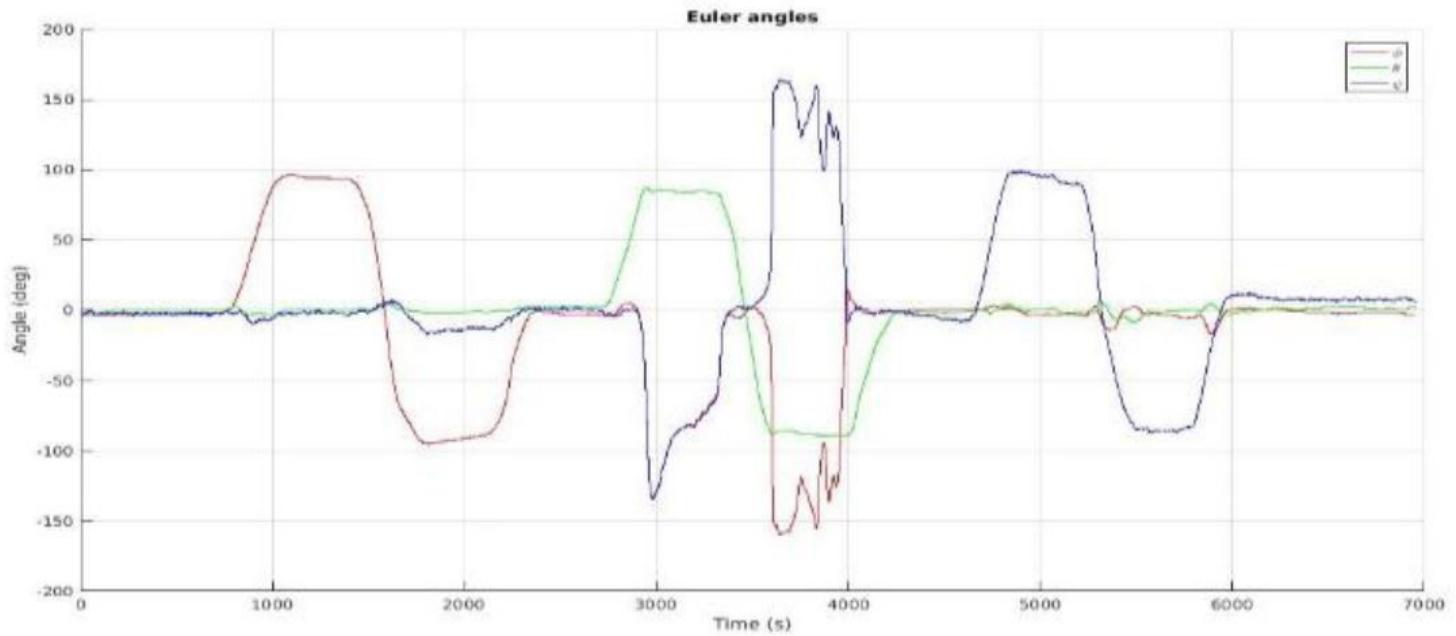


Figure 2

Orientation Estimation (in euler) Generated Using Algorithm 1, DCM.

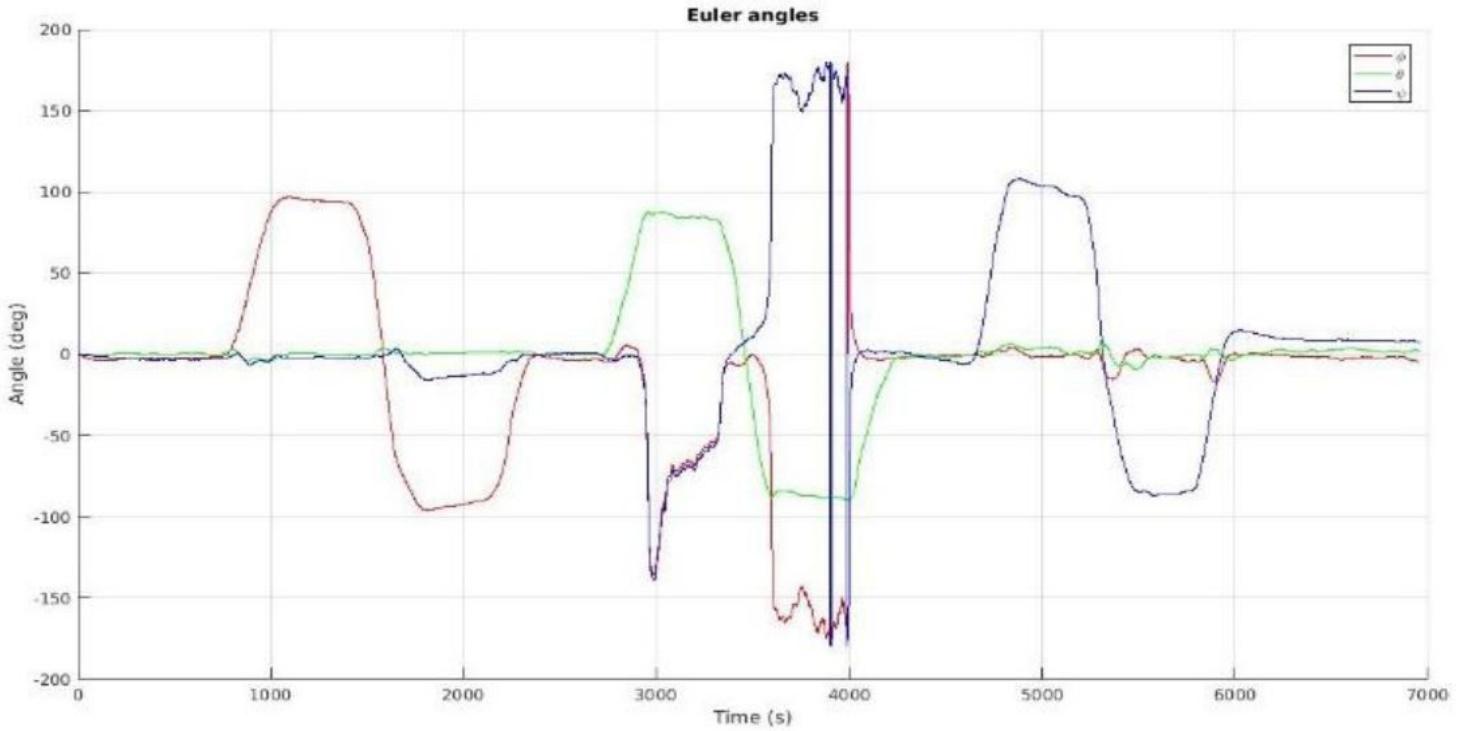


Figure 3

Orientation Estimation (in euler) Generated Using Algorithm 2, Madgwick's.

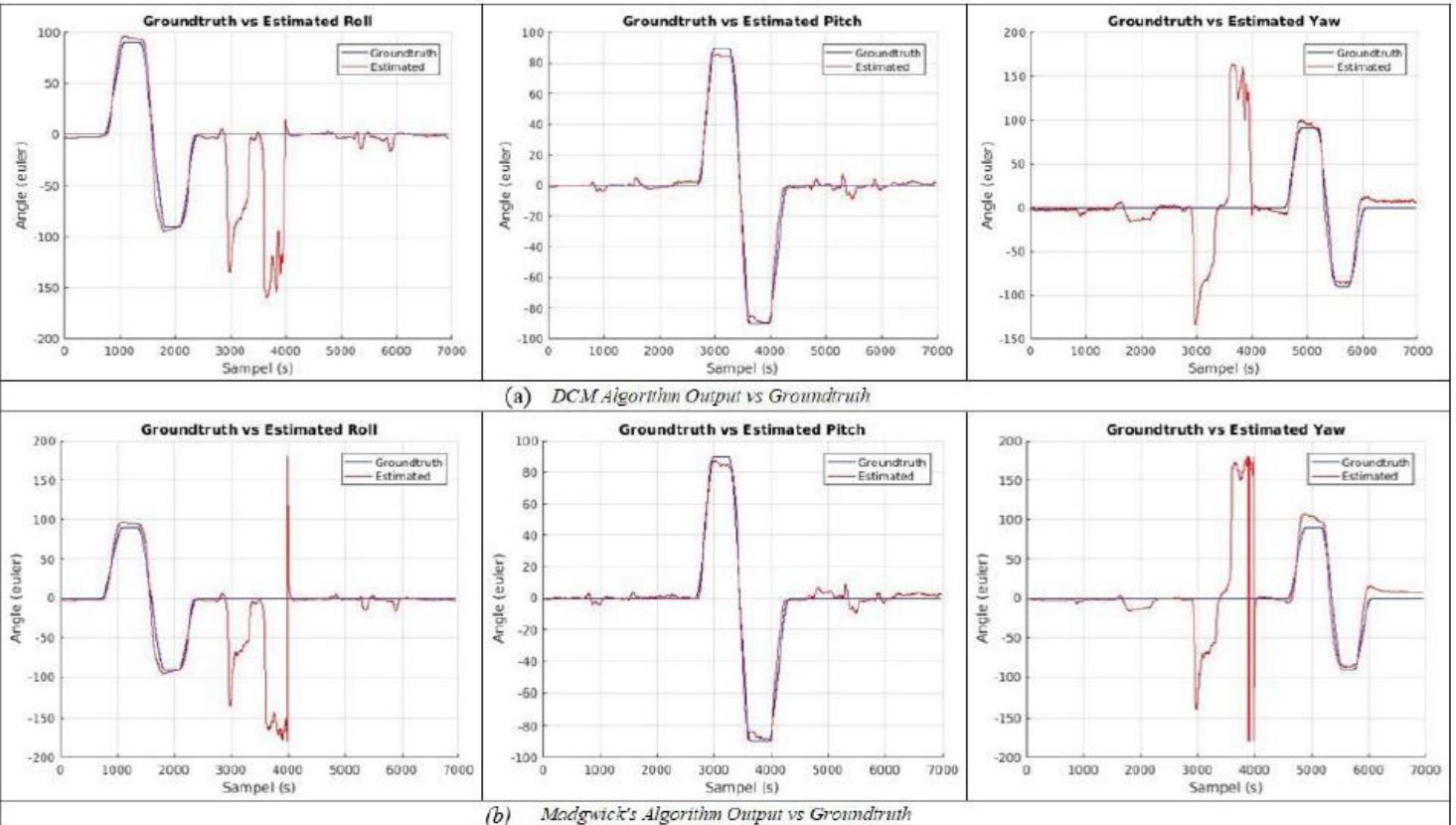


Figure 4

DCM and Madgwick's Orientation Estimation Comparison with Groundtruth using Euler Representation.

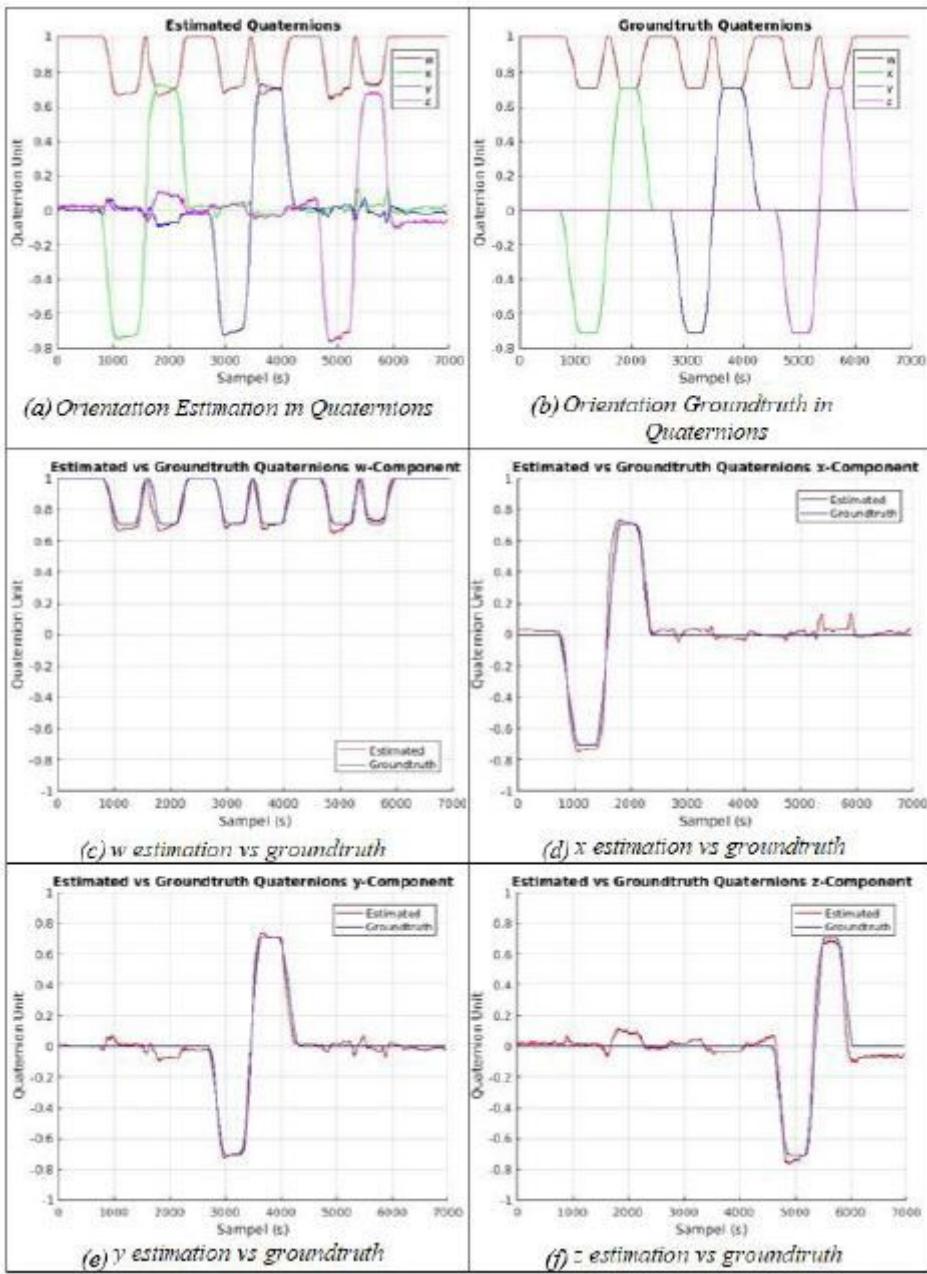


Figure 5

DCM Orientation Estimation Comparison with Groundtruth using Quaternion Representation.

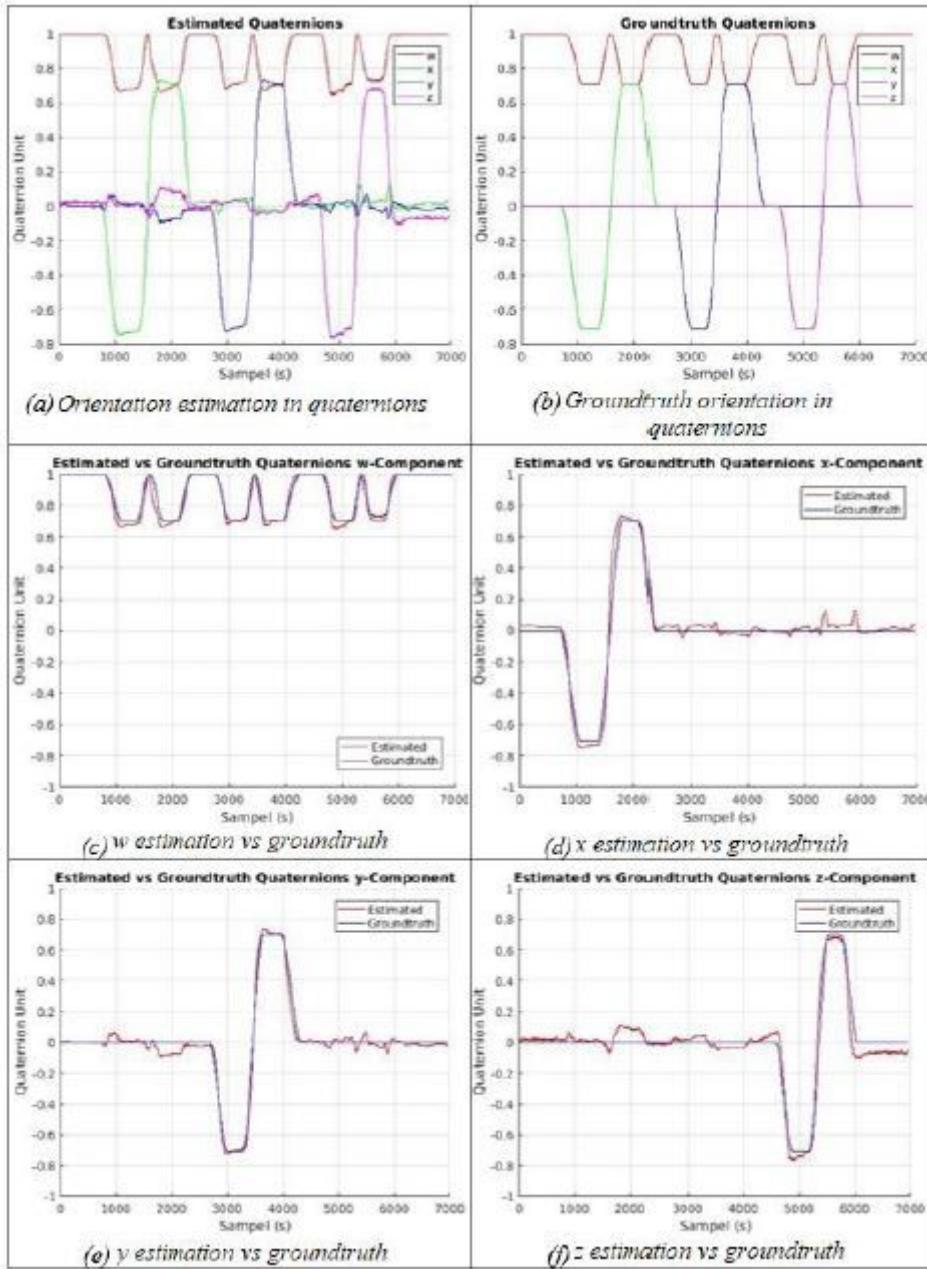


Figure 6

Madgwick's Orientation Estimation Comparison with Groundtruth using Quaternion Representation.

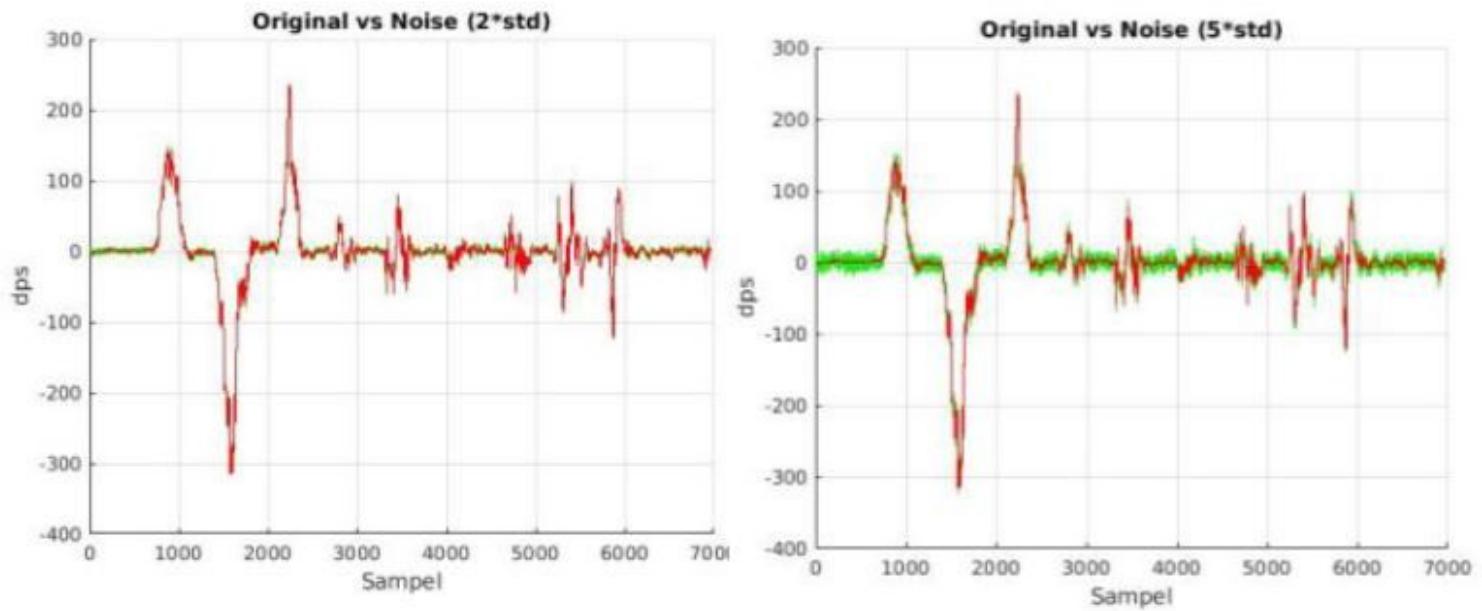


Figure 7

Comparison between Original and Noise Added Measurement with 2σ (left) and 5σ (right)

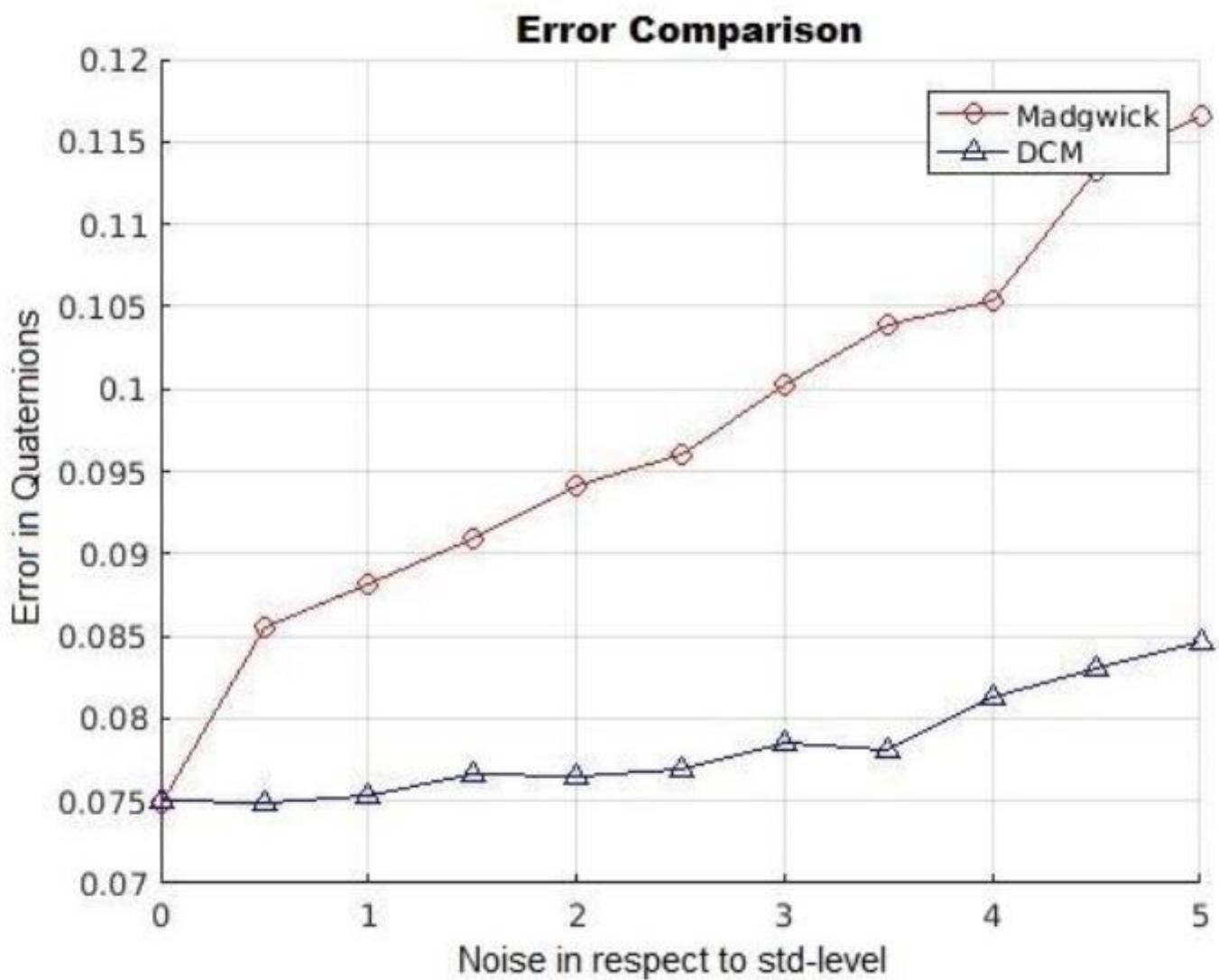


Figure 8

DCM and Madwick's Comparison in Resistance to Noise.

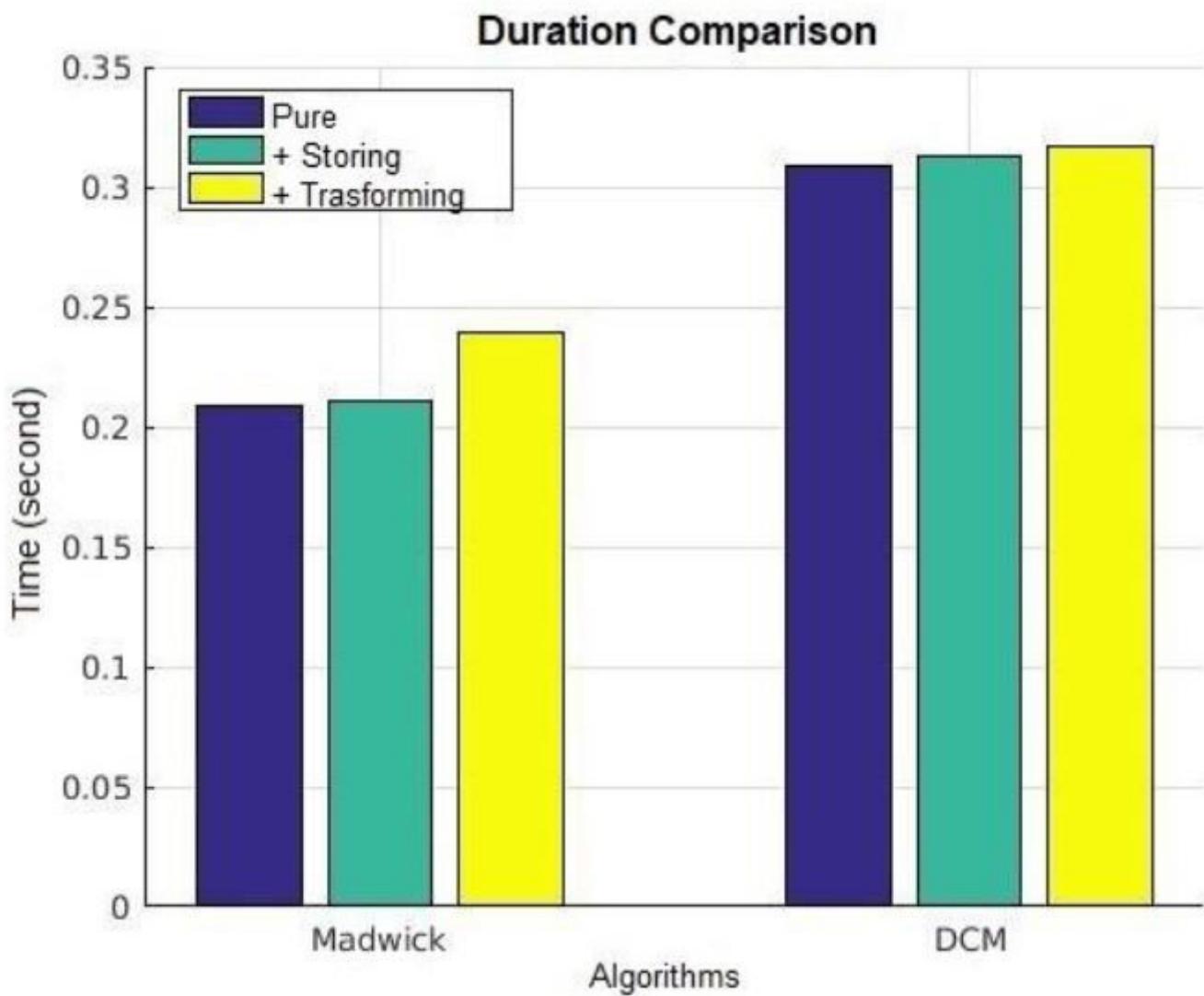


Figure 9

Comparison of Algorithm 1 and Algorithm 2 in the Level of Execution Speed.

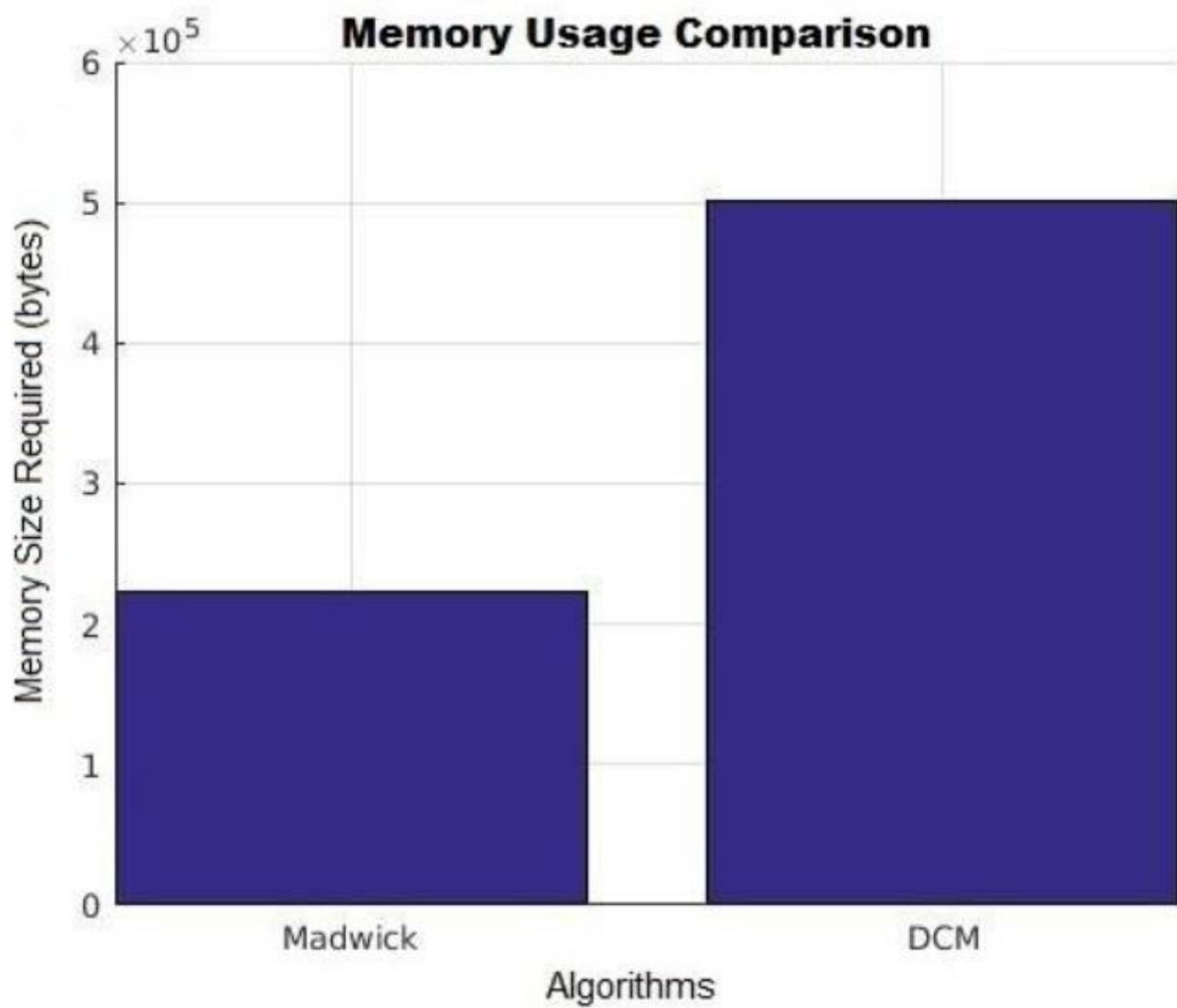


Figure 10

Comparison of Algorithm 1 and Algorithm 2 in the Amount of Memory Needed.