DOI: 10.22146/jnteti.v13i4.9403

Improving the Android Geopositioning Accuracy Using Graham Scan Algorithm and Moment Centroid

Rachmat Wahid Saleh Insani[1], Sucipto[1]

[1] Department of Informatics Engineering, Faculty of Engineering and Computer Science, Universitas Muhammadiyah Pontianak, Pontianak, Kalimantan Barat 78123, Indonesia

[Received: 18 September 2023, Revised: 4 June 2024, Accepted: 1 October 2024]
Corresponding Author: Rachmat Wahid Saleh Insani (email: rachmat.wahid@unmuhpnk.ac.id)

ABSTRACT — Geopositioning is the process of determining or estimating the geographic position of an object through the global positioning system (GPS). The calculations in geopositioning require measurements of distances or angles relative to known reference positions. In Android devices, achieving accuracy, speed, and power efficiency in geopositioning with GPS, cellular networks, and Wi-Fi can be challenging. This research aimed to improve the accuracy of the geopositioning process for cellular networks on Android devices through polygon triangulation using the Graham scan algorithm and determining a moment centroid for the improved estimation of geolocation data. The geolocation data were collected using an Android smartphone with a cellular network and disabled Wi-Fi. A filtering phase on the coordinates was established to obtain the closest distance coordinates from the other. The distances between each pair of coordinates were calculated using the haversine formula, and then the average distance of all pairs was calculated. Then, a polygon was formed by arranging the coordinates in a sequence, which was achieved using the Graham scan algorithm. After obtaining a set of triangles from the polygon triangulation results, the moment centroid of each formed triangle was determined. The centroid, as a result, was compared with another centroid calculation, the Lagrange interpolation polynomial. Based on the results obtained from quantifying the accuracy and precision using average Euclidean error (AEE) and root mean square error (RMSE), the coordinates derived from the moment centroid were more accurate and precise than the Lagrange interpolation polynomial.

KEYWORDS — Geopositioning, Graham Scan Algorithm, Moment Centroid, Android Phone.

I. INTRODUCTION

Navigation requires the utilization of vision, common sense, and directional indicators. It also relies on more precise knowledge, such as speed, distance, and electronic signals, to determine positions, known as radio navigational aids. Radio navigational aid is a network of transmitters that determines the user's position. This tool has several boundaries that may not be compatible with dynamic user positions, such as their limited range, lack of precision, and dependence on the users' latitude. Global positioning systems (GPS), a satellite-based navigation system, was designed to render accurate, continuous global 3D positional and speed information through a suitable device [1]. The GPS has been used in a wide range of areas such as sports [2], education [3], disaster management [4], healthcare [5], agriculture [6], and transportation [7]. Nevertheless, a few errors or flaws reduces the accuracy of the GPS. Failure to upload data results in these errors, which have a distorting effect on the user position estimates [8].

Geopositioning is a procedure to determine and forecast the geographical position of an object, which requires measurements of distances and angles concerning the recognized reference positions. Observations on three reference points from three satellites are used to determine the position of an object. However, there may be errors that lead to position uncertainties in the form of an area, known as an error ellipse [9]. It can be difficult to achieve accuracy, speed, and power efficiency when conducting geolocating with GPS, cellular networks, or Wi-Fi, specifically on an Android device. Moreover, continuous position recalculation is required because of dynamic changes in the user's location. In terms of accuracy, the location predictions are inconsistent, making it challenging to retrieve accurate user locations [10].

Android uses fused location provider (FLP) as one of the location application programming interfaces (API) that manage the location technology on Android devices while maintaining the device's use of battery power. It is a combination of GPS location provider and the network location provider services. FLP offers a way to access location data from GPS, Wi-Fi, cellular networks, and sensors. It combines data from multiple sources to provide accurate location information while reducing the need to keep high-power location sensors activated continuously [11]. It has access to Google data obtained from Android phones for more accurate location estimates [12]. Several application services on Android store geolocation data in the device. These data are more accurate than the network operator data [13].

There are several different location estimation techniques and algorithms used to detect the location of mobile devices, which can be divided into three categories: proximity detection, scene analysis, and triangulation [14]. Proximity detection determines the location using the cell of origin method, wherein the positions are known and ranges are limited; this category provides symbolic relative location information [15]. Scene analysis is based on radio frequency signals by collecting fingerprints of a scene and determining the location by matching online measurements with the closest previous location fingerprints [16]. Triangulation determines the location using the geometric properties of triangles by measuring its distance from multiple reference points. At least three fixed points are necessary to determine a position using this technique [17].

The important aspect of mobile devices is to have accurate and precise location determinations of location. Figure 1 shows accuracy and precision in coordinates. An accuracy is a

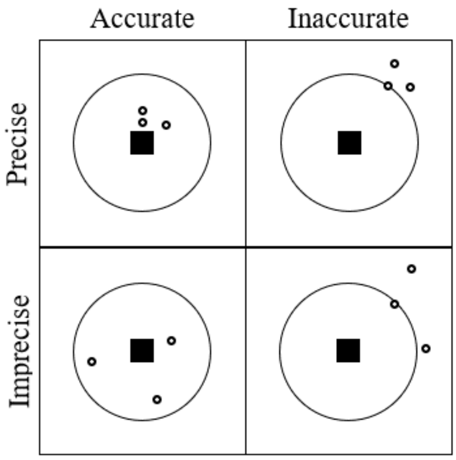

Figure 1. Accuracy and precision in coordinates.

quantification of how close measurements are to the true value, while precision is a quantification of how close replicate measurements are to each other [18]. Measurements are accurate if they cluster around the true location and precise if they cluster close to each other [19].

The relationships between points on the earth's surface, as projections on a flat plane, can be measured in polygons. The centroid point of a complex polygon is a value that can be determined through mathematical formulas and Cartesian points. The moment centroid is the most appropriate centroid for a polygon in geographic information systems (GIS) [20]. The Graham scan algorithm can be used in improved localization algorithms with high accuracy as an approximation technique to define a convex hull containing nodes and estimate the position of nodes [21]. The Graham scan algorithm also has been implemented in various studies, such as identifying traffic congestion and estimating air quality using crowd-sourced information [22], extracting characteristics from toothmarks in traditional Chinese medicine [23], and delineating convex-concave boundaries and measuring discontinuity size in rock mass [24].

Several research studies are focused on improvements of the Android GPS location: a centroid GPS model to improve the positioning accuracy of receivers estimates by calculating a triangular centroid sum of the individual positions of GPS receivers using geolocation data, which results in an improved accuracy of about 2–12 m compared to the original GPS receivers [25]. A model for better accuracy has adopted the enhanced Kalman filter, amended on the trilateration technique, and centroid localization algorithm. A center point was computed from the produced triangle-shaped model. The coordinates of this approach were taken and used together with the center point coordinates, which improved accuracy by more than 80% [26]. An error correction of GPS coordinate was calculated using the haversine formula and polynomial Lagrange interpolation using PL/SQL to coordinate roads, which resulted in their calculation accuracy depending on the number of points within a radius [27]. An implementation and test of the interpolation method of GPS satellite coordinates for real-time processing software, which computes the position and velocity of GPS satellites from broadcast and precise ephemerides [28]. Android location accuracy has also been improved using the relative and clustering analysis correction technique, eliminating common error components from pseudo-scope measurements and noncommon errors through cluster analysis. This resulted in lower accuracy in a multipath environment than in an open environment due to the reflection and refraction of signals [29].

This research proposed an approach for improving the location accuracy of Android smartphones by implementing the Graham scan algorithm to determine a convex hull and forming a polygon using coordinates to deploy the triangulation technique. The results were compared with the Lagrange interpolating polynomial method in [27]. Average Euclidean error (AEE) and root mean square error (RMSE) calculation were used to define and calculate the accuracy and precision of the proposed approach [30]. In comparison with previous studies, this research implemented triangulation as one of several location estimation techniques for mobile devices, integrated with earth's distance calculation methods, forming polygons in a plane, and calculates the accuracy and precision using methods by the Federal Geographic Data Committee (FGDC).

II. PROPOSED METHODOLOGY

The data collection phase was performed using the Android application. Then, a filtering phase on the data was established on a set of coordinates with the furthest distance from the other using the haversine formula. The coordinate result was formed into a polygon by determining the convex hull using the Graham scan algorithm. Then, polygon triangulation was performed to create a few triangles using the ear-clipping algorithm. Following the polygon triangulation, a moment centroid was calculated as the estimated coordinate. The shoelace formula was used to get the area values for each triangle. Then, a comparison result was performed with another approach, the Lagrange interpolating polynomial method, by calculating accuracy and precision using AEE and RMSE—these processes are described in Figure 2.

A. DATA COLLECTION

This research began with the data collection phase to obtain field data in several geographical regions in Pontianak, Indonesia. The location was determined by the following characteristics: (a) being in a spacious and open area, (b) fair cellular network signal, and (c) having a reference point in the satellite image with a recognizable shape, such as a cross sign. This allows the satellite image of the location to help determine the control point coordinates. The control point is the coordinate where the data collection phase is obtained using the Android device. It was measured by pinpointing a certain location on Google Maps.

This research also built an Android app (Figure 3) to conduct the data collection phase. The data for the control point were in the form of geodetic coordinates in decimal degree format, including latitude and longitude. A Xiaomi Redmi Note 10 smartphone with Android 11 operating system was used through cellular data connection from Indosat Ooredoo Hutchison provider, a 4G network type. Meanwhile, Wi-Fi was disabled to obtain several coordinates from a single location obtained in latitude and longitude format. Although there was no fixed time intervals elapsed between each measurement, the app would not save similar data and skip to the next measurement until disparate data were acquired. The ordinary utilization of a mobile device was simulated by holding it at waist level in front of the body at a length from the elbow to the hand of a student whose height was 160 cm. Simulation was conducted in an open area where the cellular signals were

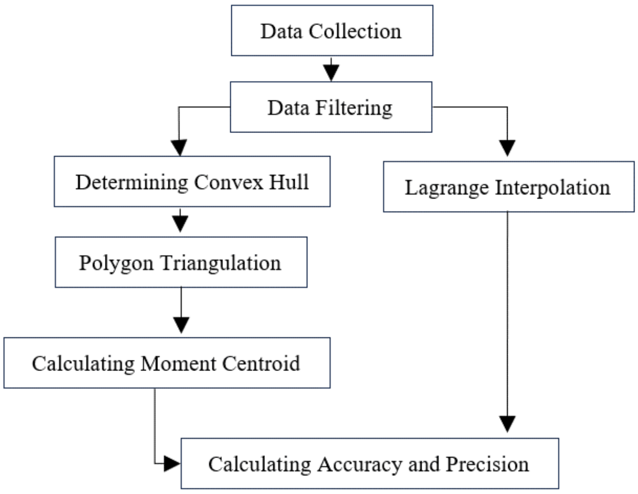

Figure 2. Proposed methodology flowchart.

relatively fair. Ten latitude and longitude measurements were taken at each of three different physical locations.

A filtering phase on the geolocation data was established on a set of coordinates that have the furthest distance from the other. The distances between coordinates were measured using the haversine formula. The haversine formula is a mathematical formula to calculate the distance between two points on the surface of a spherical object, such as the earth, using latitude and longitude coordinates. This formula considers the curvature of the earth's surface to provide a more accurate distance calculation compared to the Euclidean distance formula applied to a flat surface. The haversine formula is derived from the haversine law, which states that for any spherical trigonometry, the haversine of half the side length is equal to the haversine of the complementary angle of that side. If $\Delta\phi$ is the difference in latitude between the two points, $\Delta\lambda$ is the difference in longitude between the two points, ϕ_1 and ϕ_2 are the respective latitudes of the two points, R is the radius of the earth, which is 6,371 km, d is the distance between the two points in kilometers, then the haversine of an angle is defined as the square root of the sine of half of that angle [31].

$$a = \left(\frac{\Delta\phi}{2}\right) + cos\ cos\ (\phi_1)\ \times cos\ cos\ (\phi_2) \times \left(\frac{\Delta\lambda}{2}\right)$$
$$c = 2 \times \left(\sqrt{a}, \sqrt{(1-a)}\right) \quad (1)$$
$$d = R \times c$$

This formula has been used in various research, such as changes in the rate of ionosphere total electron content compression in pre-earthquake region estimation [32], early warning systems for ship collision based on automatic identification system data [33], and accuracy assessment of aircraft positioning using the self-evaluated method and quick access recorder data [34].

B. DETERMINING CONVEX HULL

For each location, the obtained coordinates were formed into a polygon by determining the convex hull using the Graham scan algorithm. This process required triangulation and calculation of the centroid. The Graham scan algorithm is a method used in computational geometry to find the convex hull of a set of points in a 2D plane. The convex hull of a set of points is the smallest convex polygon that encloses all the given points [35]. All the points of the convex hull were arranged in a counterclockwise sequence as they were added to the stack. Subsequently, the algorithm efficiently eliminated unnecessary points by considering the orientations of the points concerning the last two points on the stack. The Graham scan algorithm has a time complexity of $O(n\ log\ log\ n\ )$, where n is the number of input points.

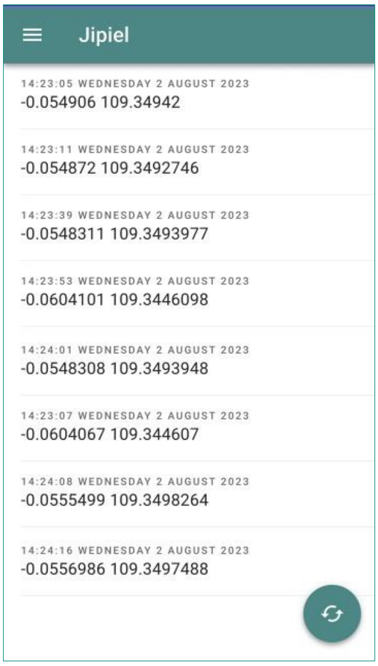

Figure 3. Coordinates from an Android application.

This algorithm starts by determining the point with the lowest y-coordinate as the pivot point. Afterward, the remaining points will be sorted in ascending order of polar angles. It pushes the pivot point onto a stack and iterates this process through the sorted points to build the convex hull incrementally. If adding the point to the convex hull results in a left turn, then the point is added to the convex hull. In contrast, if it is in a right turn, the previous point will be removed from the hull. Hence, the stack will contain the vertices of the convex hull in counterclockwise order. This algorithm can be described using the following pseudocode.

Algorithm 1. Pseudocode for Graham Scan Algorithm

```
function grahamScan(points):
  if length(points) < 3: return points
  pivot = point with lowest y-coordinate
  sort points by polar angle with respect to pivot
  stack = empty stack
  push pivot onto stack
  for i from 1 to length(points) - 1:
    while length(stack) >= 2 and orientation(nextToTop(stack),
top(stack), points[i]) is not counterclockwise:
      pop stack
    push points[i] onto stack
  return stack

function orientation(p, q, r):
  value = (q.y - p.y) * (r.x - q.x) - (q.x - p.x) * (r.y - q.y)
  if value == 0: return colinear
  return (value > 0) ? clockwise : counterclockwise

function nextToTop(stack):
  return stack[length(stack) - 2]
```

```
function top(stack):
  return stack[top index]
```

C. POLYGON TRIANGULATION

Polygon triangulation was performed to create a few triangles from the polygon. The ear-clipping algorithm was used for polygon triangulation, a process of dividing a polygon P into a set of nonoverlapping triangles. This decomposition ensures no overlap or self-intersections among the triangles. Triangulating a polygon is especially useful for solving various geometric problems, such as calculating areas, centroids, and other properties of the polygon. It plays an important role in various applications across fields like computer graphics, computer-aided design, and GIS [36].

Ear-clipping or ear-trimming algorithm is based on the observation that a simple polygon without holes with at least four vertices, has a minimum of two ears. Ears are triangles where one vertex belongs to the polygon, and the other two vertices are consecutive vertices of the polygon. This algorithm will identify ears and remove them one by one until all vertices form a set of connected triangles [37]. A simple polygon is formed using a sequence of vertices that can be connected to form the edges of the polygon. Then, an ear found, which is a vertex when combined with its adjacent vertices, will form a triangle that lies entirely inside the polygon. This process iterates through the vertices of the polygon. The ear will be removed by "clipping" it off the polygon and then connected to its adjacent vertices with an edge to form a new diagonal of the polygon. As it became much smaller, the polygon was treated as new input, and ears were found recursively within it. It terminates the process when there are no vertices left in the polygon. This algorithm can be described using the following pseudocode.

Algorithm 2. Pseudocode for Ear Clipping Algorithm

```
function earClipping(vertices):
  createLinkedList(vertices)

  triangles = []

  while linkedList.size >= 3:
    vertex = linkedList.getFirst()

    prevVertex = linkedList.getPrevious(vertex)
    nextVertex = linkedList.getNext(vertex)

    if isEar(vertex, prevVertex, nextVertex):
      linkedList.remove(vertex)

      triangle = (prevVertex, vertex, nextVertex)
      triangles.add(triangle)

return triangles
```

A number of research has implemented this algorithm, for instance, transformation and 3D geometry extraction of sets of polygons in a flat plane annotated in digital imaging and communications in medicine (DICOM) images [38], constructing 3D polygon models for a GIS through 2D vector data [39], and building a viewer for CityGML to optimize the visualization process of CityGML data with augmented reality [40].

D. CALCULATE MOMENT CENTROID

Following the polygon triangulation, the moment centroid was calculated as the estimated coordinate. The determination of centroid on a plane polygon is standard functionality in most GIS software [20]. The moment centroid or geometric centroid is a point that represents the center of mass or the average location of moments in an object or system. The moment centroid is used to simplify calculations related to the distribution of forces or moments acting on an object [41]. Given any flat surface with area A, the centroid is a point with coordinates (C_x, C_y) given by $\underline{x} = \frac{M_y}{A}$ and $\underline{y} = \frac{M_x}{A}$, where M_x and M_y is the first moment of the area along the y-axis and x-axis, respectively [42]. The moments M_x and M_y are determined based on the assumption that the polygon is shaped of several triangles with areas A_k, each with a centroid $(\underline{x_k}, \underline{y_k})$, so that

$$\underline{x} = \frac{M_y}{A} = \frac{M_{y_1} + M_{y_2} + \cdots + M_{y_n}}{A} = \frac{\sum_{k=1}^{n} A_k \underline{x_k}}{A}$$
$$\underline{y} = \frac{M_x}{A} = \frac{M_{x_1} + M_{x_2} + \cdots + M_{x_n}}{A} = \frac{\sum_{k=1}^{n} A_k \underline{y_k}}{A} \tag{2}$$

The shoelace formula was used to get the area values for each triangle. It is a method to calculate the area of a simple polygon using connections between vertices and edges in a 2D space. It is based on the concept that the area of a polygon can be determined by summing the x and y coordinate products of consecutive vertices and taking the absolute value of half of this sum [43]. In the case that each of the vertices is placed in a clockwise or counterclockwise direction, this formula may be utilized for a simple polygon. Given a set of n vertices of a simple polygon in the Euclidean plane in counterclockwise, $(x_0, y_0), \ldots, (x_{n-1}, y_{n-1})$, the area A of the polygon can be calculated as follows:

$$A = \frac{1}{2}(x_0 y_1 - x_1 y_0 + \cdots + x_{n-2} y_{n-1} - x_{n-1} y_{n-2} + x_{n-1} y_0 - x_0 y_{n-1}). \tag{3}$$

III. EVALUATION METHOD

A. LAGRANGE INTERPOLATING POLYNOMIAL

The results were compared with the Lagrange interpolating polynomial method in [27]. It is a method to generate a polynomial that passes over n data points, $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$ and takes certain values randomly, $y_i = f(x_i)$, the Lagrange interpolating polynomial $P(x)$ of degree $\leq (n-1)$ is given by (4).

$$P(x) = \sum_{j=1}^{n} P_j(x)$$
$$P_j(x) = y_j \prod_{k=1;k\neq j}^{n} \frac{x - x_k}{x_j - x_k}. \tag{4}$$

B. CALCULATING SPATIAL ACCURACY

The accuracy and precision are quantified using AEE and RMSE as described by FGDC in [30]. Given the (x, y) coordinates of n points, and the (x, y) coordinates of some special point (x_s, y_s), AEE is defined to be (5).

$$AEE = \frac{1}{n}\sum_{i=1}^{n} \sqrt{((x_i - x_s)^2 + (y_i - y_s)^2)}. \tag{5}$$

AEE is the average of the Euclidean distances between the n points and the special point. Given the (x, y) coordinates of n points and (x, y) coordinates of special point (x_s, y_s), RMSE is defined in (6) [44].

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left((x_i - x_s)^2 + (y_i - y_s)^2\right)}. \quad (6)$$

FGDC in [44] provides spatial data accuracy computation divided into RMSE in longitudinal direction $RMSE_x$ and RMSE in latitudinal direction $RMSE_y$, so that

$$RMSE_x = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - x_s)^2}$$
$$RMSE_y = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - y_s)^2}. \quad (7)$$

When $RMSE_x = RMSE_y$, the accuracy equals 1.7308. If $0.6 < RMSE_x/RMSE_y > \frac{5}{3}$, the accuracy $\approx 1.22385(RMSE_x + RMSE_y)$ [44]. When RMSE and AEE values are small, the accuracy and precision will be high. The calculation result will be in degrees (°). FGDC recommends the calculation result of RMSE to be represented in physical distances by converting the degrees of latitude and longitude to the state plane coordinate (SPC) systems.

IV. RESULT AND ANALYSIS

The Android device obtained geolocation data using a cellular data connection. Wi-Fi was disabled with the configuration set to the best estimation accuracy. However, this configuration caused active location computation to occur on the device and generates a data cache. As the geolocation data were acquired, the device provided cached data if the process was performed within a close time range. Therefore, the application was set to repeat the geolocation data retrieval and stored only dissimilar geolocation data to avoid redundant information from the cache. Afterward, a filtering phase on the geolocation data was established on a set of coordinates with the furthest distance from the other. The distances between each pair of coordinates were calculated using the haversine formula. Then, the average distance of all pairs was calculated. Among all coordinates, there were several with a distance to all pairs farther from the average value found. Some of them were located far from the other coordinates. Coordinates with distances from other coordinates exceeding the average value were removed from the dataset. This filtering process ensures that only the most relevant and meaningful geolocation data points are retained, removing any outliers or data that might be less accurate. Coordinates with a distance below the average value were formed into a polygon to undergo the polygon triangulation process using the ear-clipping algorithm. The Graham scan algorithm was used to form the polygon by obtaining the order of coordinates in a clockwise direction. The purpose is to find the convex hull of a set of points, the smallest convex polygon that encloses all the given points.

The process is shown on Figure 4. It was initiated by (a) converting the coordinates into a Cartesian coordinate system and mapping them into a Cartesian plane. Then, they were arranged in a sequence to form a polygon, (b) which was achieved using the Graham scan algorithm. After obtaining a set of triangles from the polygon triangulation results (c), the moment centroid of each formed triangle was determined (d).

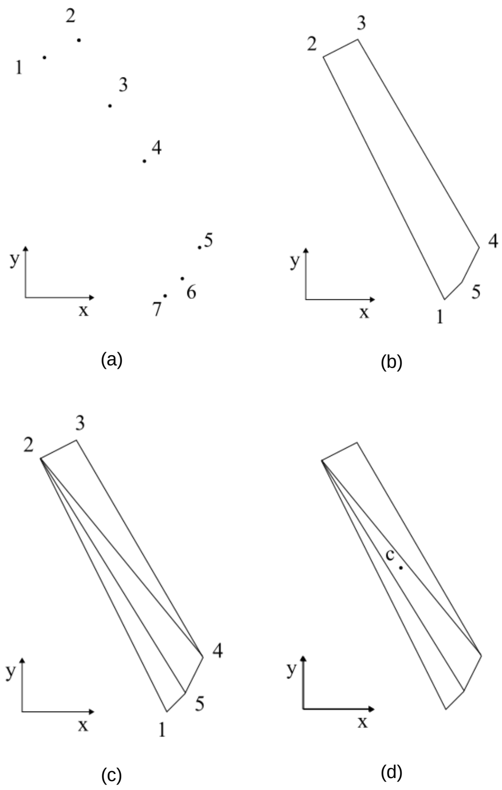

Figure 4. Polygon triangulation and calculating moment centroid, (a) coordinates in Cartesian plane, (b) in a form of a polygon, (c) partition polygon into a set of triangles, and (d) moment centroid of each triangle.

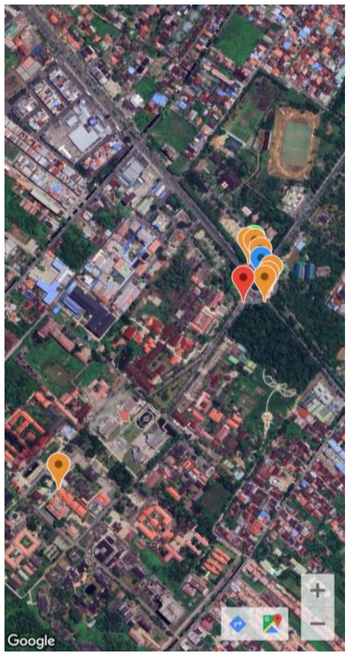

Figure 5. Google Maps API marker for coordinates.

Another estimation was performed using the Lagrange interpolation formula. Both coordinates were displayed on the Android application along with the coordinates of reference points and other input coordinates, as shown in Figure 5. The map was displayed using Google Maps API. The red marker indicates the control point, the green marker indicates coordinate from Lagrange interpolation formula, the blue marker indicates the result, and the orange markers indicate input coordinate data in a single measurement.

Subsequently, the spatial accuracy was calculated using AEE and RMSE, the results are described in Table I. Based on

TABLE I
COMPARISON RESULTS

Centroid	AEE	RMSE
Lagrange interpolation	0.000727995742	0.0005147707
Moment centroid	0.000642858562	0.0004545696

the results displayed in Table I, the coordinates from the moment centroid have the lowest error compared to the coordinates from the Lagrange interpolation polynomial. It indicates that the moment centroid method provides higher spatial accuracy when compared to the result from the Lagrange interpolation polynomial in determining the Android user's location.

V. CONCLUSION

The geolocation data were collected using an Android smartphone with a cellular network, while Wi-Fi was disabled. A filtering process was performed on the coordinates with the furthest distance from the other. The distance between each pair of coordinates was calculated using the haversine formula. Then, the average distance of all pairs was computed. All coordinates with distances below the average were used to form a polygon, which then underwent the polygon triangulation process using the ear-clipping algorithm. The polygon formation was done with the Graham scan algorithm to determine the order of coordinates in a clockwise direction, which would form the polygon. After obtaining a set of triangles from the polygon triangulation results, the moment centroid of each formed triangle was determined. Based on the results obtained from calculating the AEE and RMSE, the coordinates derived from the moment centroid determination were more accurate and precise to the exact points compared to the coordinates obtained from the Lagrange interpolation polynomial.

CONFLICTS OF INTEREST

The authors declare that there is no conflict of interest during the writing of this paper.

AUTHORS' CONTRIBUTIONS

Conceptualization, Rachmat Wahid Saleh Insani; methodology, Rachmat Wahid Saleh Insani; software, Rachmat Wahid Saleh Insani; validation, Sucipto; formal analysis, Rachmat Wahid Saleh Insani; investigation, Sucipto; resources, Rachmat Wahid Saleh Insani; data curation, Sucipto; writing—original draft preparation, Rachmat Wahid Saleh Insani; writing—reviewing and editing, Sucipto; visualization, Rachmat Wahid Saleh Insani; supervision, Rachmat Wahid Saleh Insani and Sucipto; project administration, Sucipto; funding acquisition, Rachmat Wahid Saleh Insani.

ACKNOWLEDGMENT

This research was supported by the Lembaga Penelitian dan Pengabdian Kepada Masyarakat, Universitas Muhammadiyah Pontianak. We thank our colleagues from the Software Engineering Laboratory, Universitas Muhammadiyah Pontianak, part of the Department of Informatics Engineering, who provided insight and expertise that greatly assisted the research.

REFERENCES

[1] M.J. Rycroft, "Understanding GPS. Principles and applications: Kaplan E. D. (ed.), 1996, 554 pp. Artech House, £75, hb, ISBN 0-89006-793-7," *J. Atmos. Sol. Terr. Phys.*, vol. 59, no. 5, pp. 598–599, Mar. 1997, doi: 10.1016/s1364-6826(97)83337-8.

[2] R.J. Aughey, "Applications of GPS technologies to field sports," *Int. J. Sports Physiol. Perform.*, vol. 6, no. 3, pp. 295–310, Sep. 2011, doi: 10.1123/ijspp.6.3.295.

[3] N. Lambrinos and F. Asiklari, "The introduction of GIS and GPS through local history teaching in primary school," *Eur. J. Geogr.*, vol. 5, no. 1, pp. 32–47, Mar. 2014.

[4] K.M. Kafi and M.B.A. Gibril, "GPS application in disaster management: A review," *Asian J. Appl. Sci.*, vol. 4, no. 1, pp. 63–69, Feb. 2016.

[5] S.B. Im, S. Hurlebaus, and Y.J. Kang, "Summary review of GPS technology for structural health monitoring," *J. Struct. Eng.*, vol. 139, no. 10, pp. 1653–1664, Oct. 2013, doi: 10.1061/(ASCE)ST.1943-541X.0000475.

[6] M.R. Yousefi and A.M. Razdari, "Application of GIS and GPS in precision agriculture (a review)," *Int. J. Adv. Biol. Biomed. Res.*, vol. 3, no. 1, pp. 7–9, Mar. 2015.

[7] S. Lee, G. Tewolde, and J. Kwon, "Design and implementation of vehicle tracking system using GPS/GSM/GPRS technology and smartphone application," in *2014 IEEE World Forum Internet Things (WF-IoT)*, 2014, pp. 353–358, doi: 10.1109/WF-IoT.2014.6803187.

[8] P.K. Enge, "The global positioning system: Signals, measurements, and performance," *Int. J. Wirel. Inf. Netw.*, vol. 1, no. 2, pp. 83–105, Apr. 1994, doi: 10.1007/BF02106512.

[9] A. Nait-Sidi-Moh, M. Bakhouya, J. Gaber, and M. Wack, Eds. *Geopositioning and Mobility*. London, England: Wiley-ISTE, 2013.

[10] C.R. Rani *et al.*, "Location based services in Android," *Int. J. Adv. Eng. Technol.*, vol. 3, no. 1, pp. 209–220, Mar. 2012.

[11] A. Nadaf and B.K. Srinivas, "Android based fused location provider api for localization accuracy," *Int. J. Latest Trends Eng. Technol.*, vol. 8, no. 3, pp. 278–282, May 2017, doi: 10.21172/1.83.042.

[12] N. Moayeri, C. Li, and L. Shi, "Indoor localization accuracy of major smartphone location apps," in *2019 IEEE Wirel. Commun. Netw. Conf. (WCNC)*, 2019, pp. 1–8, doi: 10.1109/WCNC.2019.8885639.

[13] M. Spreitzenbarth, S. Schmitt, and F. Freiling, "Comparing sources of location data from Android smartphones," in *Adv. Digit. Forensics VIII 8th IFIP WG 11.9 Int. Conf. Digit. Forensics*, 2012, pp. 143–157, doi: 10.1007/978-3-642-33962-2_10.

[14] Z. Farid, R. Nordin, and M. Ismail, "Recent advances in wireless indoor localization techniques and system," *J. Comput. Netw. Commun.*, vol. 2013, pp. 1–12, Sep. 2013, doi: 10.1155/2013/185138.

[15] M. Werner, *Indoor Location-Based Services: Prerequisites and Foundations*. Cham, Switzerland: Springer, 2014.

[16] H. Liu, H. Darabi, P. Banerjee, and J. Liu, "Survey of wireless indoor positioning techniques and systems," *IEEE Trans. Syst. Man Cybern. Part C (Appl. Rev.)*, vol. 37, no. 6, pp. 1067–1080, Nov. 2007, doi: 10.1109/TSMCC.2007.905750.

[17] M. Bin Ismail *et al.*, "Implementation of location determination in a wireless local area network (WLAN) environment," in *2008 10th Int. Conf. Adv. Commun. Technol.*, 2008, pp. 894–899, doi: 10.1109/ICACT.2008.4493911.

[18] "International Vocabulary of Metrology – Basic and General Concepts and Associated Terms (VIM) 3rd Edition," International Organization of Legal Metrology, 2007.

[19] S. Safavi and U.A. Khan, "Localization in mobile networks via virtual convex hulls," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 4, no. 1, pp. 188–201, Mar. 2018, doi: 10.1109/TSIPN.2017.2673238.

[20] R.E. Deakin, S.C. Bird, and R.I. Grenfell, "The centroid? Where would you like it to be be?" *Cartography*, vol. 31, no. 2, pp. 153–167, Dec. 2002, doi: 10.1080/00690805.2002.9714213.

[21] Y. Sabri and N. El Kamoun, "A distributed method for localization in large-scale sensor networks based on Jarvis," *Int. J. Comput. Appl.*, vol. 165, no. 3, pp. 1–9, May 2017, doi: 10.5120/ijca2017913826.

[22] S. Singh *et al.*, "A novel framework to avoid traffic congestion and air pollution for sustainable development of smart cities," *Sustain. Energy Technol. Assess.*, vol. 56, pp. 1–9, Mar. 2023, doi: 10.1016/j.seta.2023.103125.

[23] Y. Jiaxin *et al.*, "Objectification of tooth-marked tongue in Chinese medicine based on morphological feature extraction," *Laser Optoelectron. Prog.*, vol. 59, no. 11, pp. 1–9, Jun. 2022, doi: 10.3788/LOP202259.1117001.

[24] C. Na *et al.*, "Intelligent interpretation of rock mass discontinuity based on three-dimensional laser point cloud," *Earth Sci.*, vol. 46, no. 7, pp. 2351–2361, Jul. 2021, doi: 10.3799/dqkx.2020.282.

[25] M.R. Islam and J.M. Kim, "A centroid-GPS model to improving positioning accuracy for a sensitive location-based system," in *Ubiquitous Comput. Appl. Wirel. Sens.*, 2015, pp. 187–196, doi: 10.1007/978-94-017-9618-7_18.

[26] M.M. Din, N. Jamil, and N.F.N.A. Aziz, "Adopting centroid and amended trilateration for better accuracy of range-based non-GPS localization," in *2018 4th Int. Conf. Comput. Inf. Sci. (ICCOINS)*, 2018, pp. 1–6, doi: 10.1109/ICCOINS.2018.8510605.

[27] N.M.S. Wibawa, I.M. Sukarsa, and I.K.A. Purnawan, "Correction position of coordinates from data GPS logger in Google maps by using lagrange interpolation method," *J. Theor. Appl. Inf. Technol.*, vol. 59, no. 2, pp. 263–268, Jan. 2014.

[28] M. Horemuž and J.V. Andersson, "Polynomial interpolation of GPS satellite coordinates," *GPS Solut.*, vol. 10, no. 1, pp. 67–72, Feb. 2006, doi: 10.1007/s10291-005-0018-0.

[29] M. Jang *et al.*, "Improving positional accuracy using relative measurement between Android smartphones," *Sens. Mater.*, vol. 34, no. 1, pp. 349–366, Jan. 2022, doi: 10.18494/SAM3711.

[30] L. Lee *et al.*, "Comparison of accuracy and precision of GPS-enabled mobile devices," in *2016 IEEE Int. Conf. Comput. Inf. Technol. (CIT)*, 2016, pp. 73–82, doi: 10.1109/CIT.2016.94.

[31] G. Van Brummelen, *Heavenly Mathematics: The Forgotten Art of Spherical Trigonometry*. Princeton, NJ, USA: Princeton Univ. Press, 2013.

[32] G. Sharma, M.S. Singh, S.P. Aggarwal, and R. Romero-Andrade, "Integrated observations on crustal strain-ionosphere total electron content anomalies before the earthquake," *Acta Geophys.*, vol. 71, no. 3, pp. 1173–1185, Jun. 2023, doi: 10.1007/s11600-023-01030-7.

[33] G.R. Anggani, A.Z. Arfianto, A. Arleiny, and B.T. Setiawan, "Ship collision realtime early warning system based on automatic identification system (AIS) data," *Indones. J. Eng. Res.*, vol. 3, no. 1, pp. 18–23, May 2023, doi: 10.11594/ijer.v3i1.42.

[34] E. Wang *et al.*, "Aircraft positioning accuracy evaluation using QAR data: A case study on four aircraft types," *J. Aeronaut. Astronaut. Aviat.*, vol. 55, no. 1, pp. 65–75, Mar. 2023, doi: 10.6125/JoAAA.202303_55(1).06.

[35] R.L. Graham, "An efficient algorithm for determining the convex hull of a finite planar set," *Inf. Process. Lett.*, vol. 1, no. 4, pp. 132–133, Jun. 1972, doi: 10.1016/0020-0190(72)90045-2.

[36] J. O'Rourke, "Polygon triangulation," in *Computational Geometry in C*, 2nd ed. New York, NY, USA: Cambridge Univ. Press, 1998, pp. 1–43.

[37] D. Eberly, "Triangulation by ear clipping," *Geometric Tools.* Access date: 14-Oct-2023. [Online]. Available: http://www.geometrictools.com/

[38] J.F. Oliveira *et al.*, "Geometric and topological modelling of organs and vascular structures from CT data," in *Multi-Modality Imaging: Applications and Computational Techniques*, M.A. de Souza, H.R. Gamba, and H. Pedrini, Eds., Cham, Switzerland: Springer, 2018.

[39] C. Cai and Z. Tianlong, "An algorithm for triangulation applied to 3D GIS," in *2010 Int. Conf. Comput. Appl. Syst. Model. (ICCASM 2010)*, 2010, pp. V9-146–V9-149, doi: 10.1109/ICCASM.2010.5623065.

[40] C. Blut, T. Blut, and J. Blankenbach, "CityGML goes mobile: Application of large 3D CityGML models on smartphones," *Int. J. Digit. Earth*, vol. 12, no. 1, pp. 25–42, 2019, doi: 10.1080/17538947.2017.1404150.

[41] R.G. Budynas and J.K. Nisbett, *Shigley's Mechanical Engineering Design*, 9th ed. New York, NY, USA: McGraw Hill, 2011.

[42] F. Ayres and E. Mendelson, *Theory and Problems of Differential and Integral Calculus*, 3rd ed. New York, NY, USA: McGraw Hill, 1990.

[43] Y. Lee and W. Lim, "Shoelace formula: Connecting the area of a polygon and the vector cross product," *Math. Teach.*, vol. 110, no. 8, pp. 631–636, Apr. 2017, doi: 10.5951/mathteacher.110.8.0631.

[44] *Geospatial Positioning Accuracy Standards Part 3: National Standard for Spatial Data Accuracy*, FGDC-STD-007.3-1998, Subcommittee for Base Cartographic Data, Federal Geographic Data Committee, Virgina, USA, 1998.