

Peningkatan Akurasi Geopositioning Android Menggunakan Algoritma Graham Scan dan Moment Centroid

Rachmat Wahid Saleh Insani¹, Sucipto¹

¹ Program Studi Informatika, Fakultas Teknik dan Ilmu Komputer, Universitas Muhammadiyah Pontianak, Pontianak, Kalimantan Barat 78123, Indonesia

[Diserahkan: 18 September 2023, Direvisi: 4 Juni 2024, Diterima: 1 Oktober 2024]

Penulis Korespondensi: Rachmat Wahid Saleh Insani (email: rachmat.wahid@unmuhpnk.ac.id)

INTISARI — *Geopositioning* adalah proses penentuan atau estimasi posisi geografis suatu objek melalui *global positioning system* (GPS). Perhitungan dalam *geopositioning* memerlukan pengukuran jarak atau sudut relatif terhadap posisi referensi yang diketahui. Pada perangkat Android, proses memperoleh akurasi, kecepatan, dan efisiensi daya dalam *geopositioning* menggunakan GPS, jaringan seluler, dan Wi-Fi merupakan sebuah tantangan. Penelitian ini bertujuan untuk meningkatkan akurasi proses *geopositioning* jaringan seluler pada perangkat Android melalui triangulasi poligon dengan algoritma *Graham scan* dan menentukan *moment centroid* estimasi data *geolocation* yang lebih baik. Data *geolocation* dikumpulkan menggunakan ponsel pintar Android dengan jaringan seluler, sementara Wi-Fi dinonaktifkan. Fase penyaringan koordinat ditetapkan guna memperoleh jarak terdekat koordinat satu dengan yang lain. Jarak antara setiap pasangan koordinat dihitung menggunakan rumus *haversine*, kemudian jarak rata-rata semua pasangan dihitung. Setelah itu, poligon dibentuk dengan mengatur koordinat dalam suatu urutan menggunakan algoritma *Graham scan*. Setelah memperoleh sekumpulan segitiga dari hasil triangulasi poligon, *moment centroid* dari setiap segitiga yang terbentuk ditentukan. Hasilnya, *centroid* tersebut dibandingkan dengan perhitungan *centroid* lain, yaitu polinomial interpolasi Lagrange. Berdasarkan hasil yang diperoleh dari pengukuran akurasi dan presisi menggunakan *average Euclidean error* (AEE) dan *root mean square error* (RMSE), koordinat yang diperoleh dari *moment centroid* lebih akurat dan tepat daripada polinomial interpolasi Lagrange.

KATA KUNCI — *Geopositioning*, Algoritma *Graham Scan*, *Moment Centroid*, Ponsel Pintar Android.

I. PENDAHULUAN

Navigasi memerlukan pemanfaatan penglihatan, *common sense*, dan indikator arah. Navigasi juga bergantung pada pengetahuan kecepatan, jarak, dan sinyal elektronik yang saksama untuk menentukan posisi, atau yang dikenal sebagai alat bantu navigasi radio. Alat bantu navigasi radio adalah jaringan pemancar yang dapat menentukan posisi pengguna. Alat ini memiliki beberapa batasan yang mungkin tidak kompatibel dengan posisi pengguna yang dinamis, seperti jangkauannya yang terbatas, kurangnya presisi, dan ketergantungannya pada garis lintang pengguna. *Global positioning system* (GPS), yang merupakan sistem navigasi berbasis satelit, dirancang untuk memberikan informasi posisi dan kecepatan 3D global yang akurat dan berkelanjutan melalui perangkat yang sesuai [1]. GPS telah digunakan dalam berbagai bidang, seperti olahraga [2], pendidikan [3], manajemen bencana [4], perawatan kesehatan [5], pertanian [6], dan transportasi [7]. Meskipun demikian, beberapa kesalahan atau kekurangan mengurangi keakuratan GPS. Kegagalan mengunggah data mengakibatkan kesalahan ini, yang memiliki efek distorsi pada estimasi posisi pengguna [8].

Geopositioning adalah prosedur untuk menentukan dan memperkirakan posisi geografis suatu objek, yang memerlukan pengukuran jarak dan sudut posisi referensi yang dikenali. Pengamatan pada tiga titik referensi dari tiga satelit digunakan untuk menentukan posisi suatu objek. Namun, dapat terjadi kesalahan yang menyebabkan ketidakpastian posisi dalam bentuk area, yang dikenal sebagai elips kesalahan [9]. Pada perangkat Android, usaha mencapai akurasi, kecepatan, dan efisiensi daya saat melakukan proses *geolocation* dengan GPS,

jaringan seluler, atau Wi-Fi merupakan sebuah tantangan. Selain itu, perhitungan ulang posisi secara terus-menerus diperlukan karena lokasi pengguna dapat berubah secara dinamis. Dalam hal akurasi, tidak konsistennya prediksi lokasi menyebabkan sulitnya mendapatkan lokasi pengguna yang akurat [10].

Android menggunakan *fused location provider* (FLP) sebagai salah satu *application programming interface* (API) lokasi yang mengelola teknologi lokasi pada perangkat Android dengan tetap menjaga penggunaan daya baterai perangkat. FLP merupakan gabungan antara layanan *GPS location provider* dan *network location provider*. FLP menawarkan cara untuk mengakses data lokasi dari GPS, Wi-Fi, jaringan seluler, dan sensor. FLP menggabungkan data dari berbagai sumber untuk menyediakan informasi lokasi yang akurat dengan mengurangi kebutuhan untuk terus mengaktifkan sensor lokasi berdaya tinggi [11]. FLP memiliki akses ke data Google yang diperoleh dari ponsel pintar Android untuk estimasi lokasi yang lebih akurat [12]. Beberapa layanan aplikasi pada Android menyimpan data *geolocation* di dalam perangkat. Data ini lebih akurat dibandingkan data dari operator jaringan [13].

Terdapat beberapa teknik dan algoritma estimasi lokasi yang berbeda yang dapat digunakan untuk mendeteksi lokasi perangkat seluler. Teknik ini terbagi menjadi tiga kategori, yaitu *proximity detection*, *scene analysis*, dan triangulasi [14]. *Proximity detection* menentukan lokasi dengan metode sel asal, yang posisinya diketahui dan jangkauannya terbatas. Kategori ini memberikan informasi lokasi relatif simbolis [15]. *Scene analysis* didasarkan pada sinyal frekuensi radio dengan

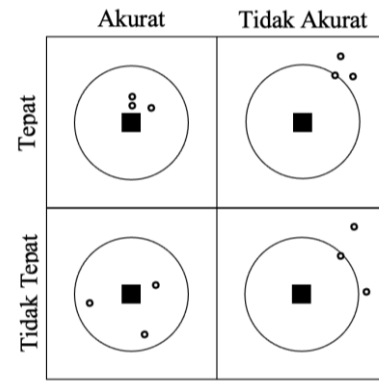
mengumpulkan *fingerprint* suatu tempat dan menentukan lokasinya dengan cara mencocokkan pengukuran daring dengan *fingerprint* lokasi sebelumnya yang terdekat [16]. Triangulasi menentukan lokasi menggunakan sifat geometris segitiga dengan cara mengukur jaraknya dari beberapa titik referensi. Setidaknya tiga titik tetap diperlukan untuk menentukan posisi menggunakan teknik ini [17].

Aspek penting dari perangkat seluler adalah memiliki penentu lokasi yang akurat dan tepat. Gambar 1 menunjukkan akurasi dan presisi pada koordinat. Akurasi adalah kuantifikasi dekatnya pengukuran dengan nilai sebenarnya, sedangkan presisi adalah kuantifikasi dekatnya pengukuran replikasi satu sama lain [18]. Pengukuran dikatakan akurat jika mengelompok di sekitar lokasi sebenarnya dan dikatakan tepat jika mengelompok dekat satu sama lain [19].

Hubungan antara titik-titik pada permukaan bumi sebagai proyeksi pada bidang datar dapat diukur dalam poligon. Titik pusat dari poligon yang kompleks merupakan nilai yang dapat ditentukan melalui rumus matematika dan koordinat Cartesian. *Moment centroid* adalah pusat yang paling tepat untuk poligon dalam *geographic information systems* (GIS) [20]. Algoritma *Graham scan* dapat digunakan pada algoritma lokalisasi yang ditingkatkan dengan akurasi tinggi sebagai teknik perkiraan untuk mendefinisikan *convex hull* yang berisi *node* dan memperkirakan posisi *node* [21]. Algoritma *Graham scan* juga telah diimplementasikan dalam berbagai penelitian, seperti mengidentifikasi kemacetan lalu lintas dan memperkirakan kualitas udara menggunakan informasi yang bersumber dari berbagai individu [22], mengekstraksi karakteristik bekas gigitan dalam pengobatan tradisional Tiongkok [23], dan menggambarkan batas cembung-cekung dan mengukur ukuran diskontinuitas dalam massa batuan [24].

Beberapa penelitian difokuskan untuk meningkatkan lokasi GPS Android. Model GPS *centroid* untuk meningkatkan akurasi perkiraan posisi penerima dengan menghitung jumlah *centroid* segitiga dari posisi individual penerima GPS menggunakan data *geolocation*, yang menghasilkan peningkatan akurasi sekitar 2–12 m dibandingkan dengan perangkat GPS asli [25]. Model yang menghasilkan akurasi yang lebih tinggi mengadopsi filter Kalman yang telah disempurnakan dengan mengubah teknik trilaterasi dan algoritma lokalisasi *centroid*. Titik pusat dihitung dari model berbentuk segitiga yang dihasilkan. Koordinat pendekatan ini diambil dan digunakan bersama dengan koordinat titik pusat, sehingga dapat meningkatkan akurasi lebih dari 80% [26]. Perhitungan koreksi kesalahan koordinat GPS dilakukan menggunakan rumus *haversine* dan interpolasi Lagrange polinomial menggunakan PL/SQL untuk mengoordinasikan jalan, yang menghasilkan akurasi yang perhitungannya tergantung pada jumlah titik dalam radius [27]. Implementasi dan pengujian metode interpolasi mengoordinasi satelit GPS perangkat lunak pemrosesan waktu nyata, yang menghitung posisi dan kecepatan satelit GSS dari *ephemerides* siaran dan presisi [28]. Akurasi lokasi Android juga telah ditingkatkan menggunakan teknik koreksi analisis relatif dan pengelompokan, yang menghilangkan komponen kesalahan umum dari pengukuran *pseudo-scope* dan kesalahan tidak umum melalui analisis kluster. Metode ini menghasilkan akurasi yang lebih rendah pada lingkungan multilintasan daripada lingkungan terbuka karena adanya refleksi dan refraksi sinyal [29].

Penelitian ini mengusulkan pendekatan untuk meningkatkan akurasi lokasi ponsel pintar Android dengan



Gambar 1. Akurasi dan presisi pada koordinat.

menerapkan algoritma *Graham scan* untuk menentukan *convex hull* dan membentuk poligon menggunakan koordinat untuk menerapkan teknik triangulasi. Hasilnya kemudian dibandingkan dengan metode polinomial interpolasi Lagrange dalam [27]. Perhitungan *average Euclidean error* (AEE) dan *root mean square error* (RMSE) digunakan untuk menentukan dan menghitung akurasi dan presisi pendekatan yang diusulkan [30]. Dibandingkan dengan penelitian sebelumnya, penelitian ini menerapkan triangulasi sebagai salah satu teknik estimasi lokasi untuk perangkat seluler, yang terintegrasi dengan metode perhitungan jarak bumi, membentuk poligon dalam bidang, dan menghitung akurasi dan presisi menggunakan metode dari Federal Geographic Data Committee (FGDC).

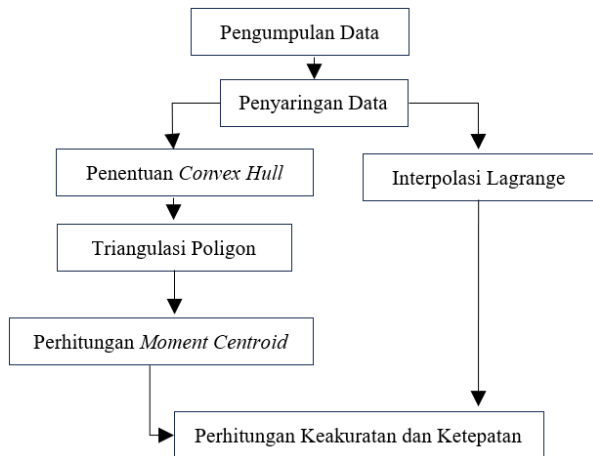
II. METODOLOGI YANG DIUSULKAN

Tahap pengumpulan data dilakukan dengan menggunakan aplikasi Android. Kemudian, tahap penyaringan data sekumpulan koordinat dengan jarak terjauh antar koordinat dilakukan menggunakan rumus *haversine*. Hasil koordinat tersebut dibentuk menjadi poligon dengan menentukan *convex hull* menggunakan algoritma *Graham scan*. Kemudian, triangulasi poligon dilakukan untuk membentuk beberapa segitiga dengan menggunakan algoritma *ear-clipping*. Setelah triangulasi poligon, titik *moment centroid* dihitung sebagai koordinat yang diperkirakan. Rumus *shoelace* digunakan untuk mengetahui luas setiap segitiga. Kemudian, hasil perbandingan dibandingkan dengan pendekatan lain, yaitu metode polinomial interpolasi Lagrange, dengan menghitung akurasi dan presisi menggunakan AEE dan RMSE. Proses ini dijelaskan pada Gambar 2.

A. PENGUMPULAN DATA

Penelitian ini dimulai dengan tahap pengumpulan data untuk memperoleh data lapangan di beberapa wilayah geografis di Pontianak, Indonesia. Lokasi tersebut dipilih dengan mempertimbangkan beberapa karakteristik: (a) berada di daerah yang luas dan terbuka, (b) terdapat sinyal jaringan seluler yang cukup baik, dan (c) memiliki titik referensi pada citra satelit dengan bentuk yang dapat dikenali, seperti tanda silang. Hal ini memungkinkan citra satelit dari lokasi tersebut untuk membantu menentukan koordinat titik kontrol. Titik kontrol adalah koordinat yang tahap pengumpulan datanya diperoleh dengan menggunakan perangkat Android. Titik kontrol diukur dengan cara menentukan lokasi tertentu pada Google Maps.

Penelitian ini juga membuat aplikasi Android (Gambar 3) yang digunakan untuk mengumpulkan data. Data titik kontrol berupa koordinat geodetis dalam format derajat desimal, termasuk lintang dan bujur. Sebuah ponsel pintar Xiaomi



Gambar 2. Diagram alir metode yang diusulkan.

Redmi Note 10 dengan sistem operasi Android 11 digunakan, dengan koneksi data seluler dari penyedia layanan Indosat Ooredoo Hutchison berjenis jaringan 4G. Sementara itu, Wi-Fi dinonaktifkan pada perangkat tersebut agar diperoleh beberapa koordinat dari satu lokasi dalam format lintang dan bujur. Meskipun tidak ada interval waktu yang tetap di antara setiap pengukuran, aplikasi tidak akan menyimpan data yang sama dan akan melompat ke pengukuran berikutnya sampai data yang berbeda diperoleh. Penggunaan perangkat seluler disimulasikan dengan memegangnya setinggi pinggang di depan tubuh, pada jarak dari siku hingga tangan seorang siswa dengan tinggi badan 160 cm. Simulasi dilakukan di area terbuka dengan sinyal seluler relatif baik. Sepuluh pengukuran lintang dan bujur dilakukan di tiga lokasi fisik yang berbeda.

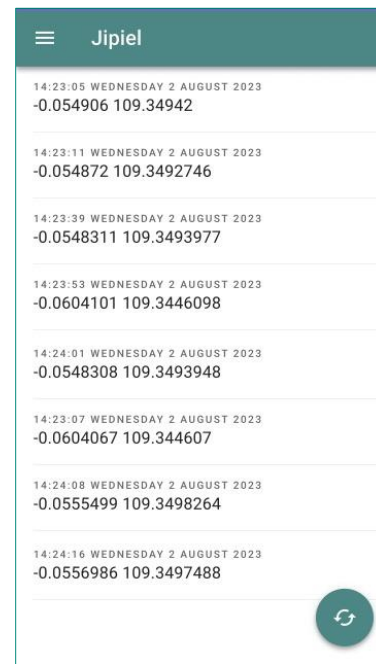
Tahap penyaringan data *geolocation* ditetapkan pada satu set koordinat yang memiliki jarak terjauh dari yang lain. Jarak antar koordinat diukur dengan menggunakan rumus *haversine*. Rumus *haversine* adalah rumus matematika untuk menghitung jarak antara dua titik di permukaan objek bulat, seperti bumi, menggunakan koordinat lintang dan bujur. Rumus ini mempertimbangkan kelengkungan permukaan bumi untuk memberikan perhitungan jarak yang lebih akurat dibandingkan dengan rumus jarak Euclidean yang diterapkan pada permukaan datar. Rumus *haversine* diturunkan dari hukum *haversine*, yang menyatakan bahwa untuk setiap trigonometri bola, *haversine* dari setengah panjang sisi sama dengan *haversine* dari sudut komplementer sisi tersebut. Jika $\Delta\phi$ adalah perbedaan lintang antara dua titik, $\Delta\lambda$ adalah perbedaan bujur antara dua titik, ϕ_1 dan ϕ_2 adalah lintang masing-masing dari dua titik, R adalah jari-jari bumi, yaitu 6.371 km, serta d adalah jarak antara dua titik dalam kilometer, maka *haversine* suatu sudut didefinisikan sebagai akar kuadrat dari sinus setengah sudut tersebut [31].

$$a = \left(\frac{\Delta\phi}{2}\right) + \cos \phi_1 \times \cos \phi_2 \times \left(\frac{\Delta\lambda}{2}\right)$$

$$c = 2 \times \left(\sqrt{a}, \sqrt{1-a}\right) \tag{1}$$

$$d = R \times c.$$

Formula ini telah digunakan dalam berbagai penelitian, seperti perubahan laju kompresi kandungan elektron total ionosfer dalam estimasi wilayah pragempa [32], sistem peringatan dini tabrakan kapal berdasarkan data sistem identifikasi otomatis [33], dan penilaian akurasi penentuan



Gambar 3. Koordinat dari aplikasi Android.

posisi pesawat terbang dengan menggunakan metode evaluasi mandiri dan data perekam akses cepat [34].

B. PENENTUAN CONVEX HULL

Untuk setiap lokasi, koordinat yang diperoleh dibentuk menjadi poligon dengan cara menentukan *convex hull* menggunakan algoritma *Graham scan*. Proses ini memerlukan triangulasi dan perhitungan *centroid*. Algoritma *Graham scan* adalah metode yang digunakan dalam geometri komputasi untuk menemukan *convex hull* dari sekumpulan titik pada bidang 2D. *Convex hull* dari sekumpulan titik adalah poligon cembung terkecil yang melingkupi semua titik yang diberikan [35]. Semua titik *convex hull* disusun dalam urutan berlawanan arah jarum jam saat ditambahkan ke tumpukan. Algoritma ini secara efisien menghilangkan titik-titik yang tidak digunakan dengan mempertimbangkan orientasi titik-titik yang berkaitan dengan dua titik terakhir pada tumpukan. Algoritma *Graham scan* memiliki kompleksitas waktu $O(n \log \log n)$, dengan n adalah jumlah titik masukan.

Algoritma ini dimulai dengan menentukan titik dengan koordinat y terendah yang dijadikan sebagai titik pivot. Setelah itu, titik-titik yang tersisa akan diurutkan berdasarkan urutan sudut polar. Algoritma ini mendorong titik pivot ke dalam tumpukan dan mengulangi proses ini pada titik-titik yang diurutkan untuk membangun *convex hull* secara bertahap. Jika penambahan titik ke *convex hull* menyebabkan belokan ke kiri, titik tersebut ditambahkan ke *convex hull*. Sebaliknya, jika terjadi belokan ke kanan, titik sebelumnya akan dihapus. Oleh karena itu, tumpukan akan berisi simpul-simpul *convex hull* dalam urutan berlawanan arah jarum jam. Algoritma ini dapat dijelaskan dengan menggunakan *pseudo-code* berikut.

Pseudo-code untuk Algoritma Graham Scan

```

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. TRIANGULASI POLIGON

Kemudian, triangulasi poligon dilakukan untuk membuat beberapa segitiga dari poligon. Algoritma *ear-clipping* digunakan untuk triangulasi poligon, sebuah proses membagi poligon P menjadi sekumpulan segitiga yang tidak tumpang tindih. Penguraian ini memastikan tumpang tindih atau perpotongan di antara segitiga tidak terjadi. Triangulasi poligon sangat berguna untuk memecahkan berbagai masalah geometris, seperti menghitung area, titik pusat, dan properti poligon lainnya. Kegunaan ini berperan dalam berbagai aplikasi di berbagai bidang, seperti grafik komputer, desain berbantuan komputer, dan GIS [36].

Algoritma *ear-clipping* atau *ear-trimming* didasarkan pada pengamatan bahwa poligon sederhana tanpa lubang dengan setidaknya empat simpul memiliki minimal dua telinga. Telinga adalah segitiga yang satu simpulnya merupakan bagian dari poligon dan dua simpul lainnya merupakan simpul yang berurutan pada poligon tersebut. Algoritma ini mengidentifikasi telinga dan menghapusnya satu per satu sampai semua simpul membentuk satu set segitiga yang terhubung [37]. Poligon sederhana dibentuk menggunakan urutan simpul yang dapat dihubungkan untuk membentuk tepi poligon. Kemudian, telinga yang ditemukan, yang merupakan sebuah simpul ketika digabungkan dengan simpul-simpul yang berdekatan, akan membentuk sebuah segitiga yang seluruhnya berada di dalam poligon. Proses ini berulang melalui simpul-simpul poligon. Telinga akan dihilangkan dengan “memotongnya” dari poligon dan kemudian dihubungkan ke simpul yang berdekatan dengan sebuah tepi untuk membentuk diagonal poligon yang baru. Ketika poligon menjadi jauh lebih kecil, poligon tersebut diperlakukan sebagai masukan baru dan telinga ditemukan secara rekursif di dalamnya. Algoritma ini mengakhiri proses ketika tidak ada simpul yang tersisa di dalam poligon. Algoritma ini dapat dijelaskan dengan menggunakan *pseudo-code* berikut.

Pseudo-code untuk Algoritma Ear Clipping

```

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

```

Sejumlah penelitian telah mengimplementasikan algoritma ini, misalnya transformasi dan ekstraksi geometri 3D dari kumpulan poligon pada bidang datar yang dianotasi pada *digital imaging and communications in medicine (DICOM)* [38], membangun model poligon 3D untuk GIS melalui data vektor 2D [39], dan membuat penampil untuk CityGML untuk mengoptimalkan proses visualisasi data CityGML dengan realitas tertambah (*augmented reality*) [40].

D. PERHITUNGAN MOMENT CENTROID

Setelah triangulasi poligon, *moment centroid* dihitung sebagai koordinat estimasi. Penentuan *centroid* pada poligon bidang merupakan fungsi standar pada sebagian besar perangkat lunak GIS [20]. *Moment centroid* atau *centroid* geometrik adalah titik yang merepresentasikan pusat massa atau lokasi rata-rata momen pada suatu objek atau sistem. *Moment centroid* digunakan untuk menyederhanakan perhitungan yang berkaitan dengan distribusi gaya atau momen yang bekerja pada suatu objek [41]. Jika diketahui sebuah permukaan datar dengan luas A , *centroid* adalah sebuah titik dengan koordinat (C_x, C_y) yang diberikan oleh $\underline{x} = \frac{M_y}{A}$ dan $\underline{y} = \frac{M_x}{A}$, dengan M_x dan M_y masing-masing adalah momen pertama dari daerah sepanjang sumbu y dan sumbu x [42]. Momen M_x dan M_y ditentukan berdasarkan asumsi bahwa poligon berbentuk beberapa segitiga dengan luas A_k , masing-masing dengan titik pusat $(\underline{x}_k, \underline{y}_k)$, sehingga

$$\begin{aligned} \underline{x} &= \frac{M_y}{A} = \frac{M_{y_1} + M_{y_2} + \dots + M_{y_n}}{A} = \frac{\sum_{k=1}^n A_k \underline{x}_k}{A}, \\ \underline{y} &= \frac{M_x}{A} = \frac{M_{x_1} + M_{x_2} + \dots + M_{x_n}}{A} = \frac{\sum_{k=1}^n A_k \underline{y}_k}{A}. \end{aligned} \quad (2)$$

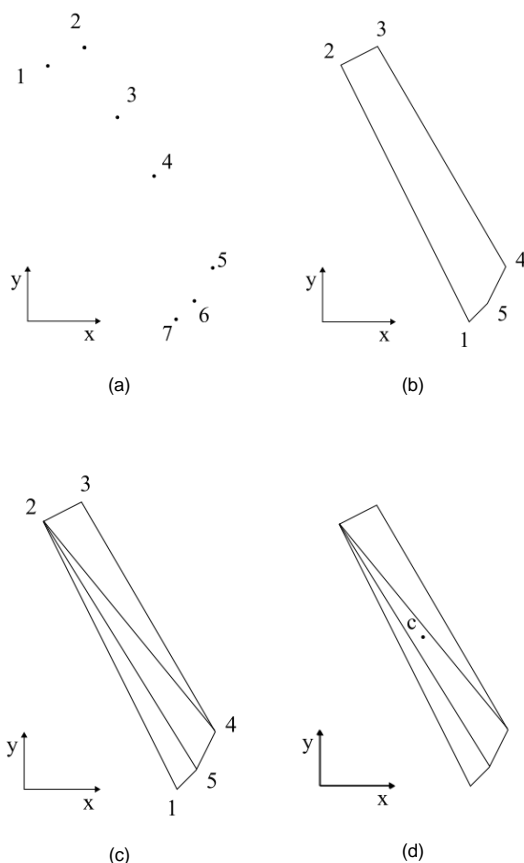
Rumus *shoelace* digunakan untuk mendapatkan nilai area untuk setiap segitiga. Rumus ini merupakan metode untuk menghitung luas area poligon sederhana dengan menggunakan hubungan antara simpul dan tepi dalam ruang 2D. Hal ini didasarkan pada konsep bahwa luas poligon dapat ditentukan dengan menjumlahkan produk koordinat x dan y dari simpul-simpul yang berurutan dan mengambil nilai absolut dari setengah jumlah ini [43]. Jika setiap simpul ditempatkan searah atau berlawanan dengan arah jarum jam, rumus ini dapat digunakan untuk poligon sederhana. Jika satu set n simpul poligon sederhana pada bidang Euclidean berlawanan arah jarum jam adalah $(x_0, y_0), \dots, (x_{n-1}, y_{n-1})$, area A dari poligon tersebut dapat dihitung sebagai berikut.

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

III. METODE EVALUASI

A. POLINOMIAL INTERPOLASI LAGRANGE

Hasil dibandingkan dengan metode polinomial interpolasi Lagrange dalam penelitian sebelumnya [27]. Metode ini merupakan metode untuk menghasilkan polinomial yang melewati titik data n , $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$ dan mengambil nilai tertentu secara acak, dengan $y_i = f(x_i)$. Polinomial interpolasi Lagrange $P(x)$ dengan derajat $\leq (n - 1)$ ditunjukkan pada (4).



Gambar 4. Triangulasi poligon dan perhitungan *moment centroid*. (a) koordinat bidang Cartesian, (b) dalam bentuk poligon, (c) membagi poligon menjadi sekumpulan segitiga, dan (d) *moment centroid* setiap segitiga.

$$P(x) = \sum_{j=1}^n P_j(x) \tag{4}$$

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

B. PERHITUNGAN AKURASI SPASIAL

Akurasi dan presisi dihitung menggunakan AEE dan RMSE, sebagaimana dijelaskan oleh FGDC dalam [30]. Jika koordinat (x, y) adalah titik n , dan koordinat (x_s, y_s) adalah titik khusus (x_s, y_s) , AEE didefinisikan sebagai (5).

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

AEE merupakan rata-rata dari jarak Euclidean antara titik n dan titik khusus. Jika koordinat (x, y) merupakan titik n dan koordinat (x_s, y_s) adalah titik khusus (x_s, y_s) , RMSE didefinisikan dalam (6) [44].

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

FGDC dalam [44] memberikan perhitungan akurasi data spasial yang dibagi menjadi RMSE pada arah longitudinal $RMSE_x$ dan pada arah *latitudinal* $RMSE_y$, sehingga

$$RMSE_x = \sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - x_s)^2}$$

$$RMSE_y = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - y_s)^2} \tag{7}$$

Apabila $RMSE_x = RMSE_y$, akurasinya adalah 1,7308. Jika $0,6 < RMSE_x/RMSE_y > \frac{5}{3}$, akurasinya $\approx 1,22385(RMSE_x + RMSE_y)$ [44]. Ketika nilai RMSE dan AEE kecil, akurasi dan presisinya akan tinggi. Hasil perhitungan akan ditampilkan dalam satuan derajat ($^{\circ}$). FGDC merekomendasikan hasil perhitungan RMSE untuk direpresentasikan dalam jarak fisik dengan mengonversi derajat lintang dan bujur ke *sistem state plane coordinate* (SPC).

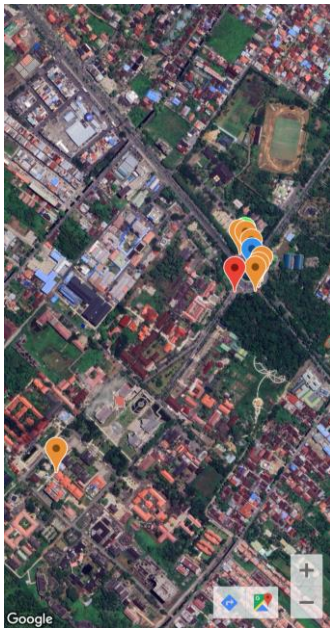
IV. HASIL DAN ANALISIS

Perangkat Android memperoleh data *geolocation* menggunakan koneksi data seluler. Wi-Fi dinonaktifkan dan konfigurasi yang ditetapkan untuk mendapatkan akurasi estimasi terbaik digunakan. Namun, konfigurasi ini menyebabkan komputasi lokasi aktif terjadi pada perangkat dan menghasilkan *cache* data. Saat data *geolocation* diperoleh, perangkat menghasilkan *cache* data jika proses dilakukan dalam rentang waktu yang dekat. Oleh karena itu, aplikasi diatur untuk mengulangi pengambilan data *geolocation* dan hanya menyimpan data *geolocation* yang tidak sama untuk menghindari informasi *cache* yang berlebihan. Setelah itu, fase penyaringan data *geolocation* ditetapkan pada serangkaian koordinat dengan jarak terjauh dari yang lain. Jarak antara setiap pasangan koordinat dihitung menggunakan rumus *haversine*. Kemudian, jarak rata-rata semua pasangan dihitung. Di antara semua koordinat, terdapat beberapa koordinat dengan jarak ke semua pasangan lebih jauh dari nilai rata-rata yang ditemukan. Beberapa di antaranya terletak jauh dari koordinat lainnya. Koordinat yang jaraknya lebih besar dari rata-rata terhadap koordinat lainnya akan dihapus dari kumpulan data. Proses penyaringan ini memastikan bahwa hanya titik data *geolocation* yang paling relevan dan bermakna yang dipertahankan, dengan membuang pencilon atau data yang mungkin kurang akurat. Koordinat dengan jarak di bawah nilai rata-rata akan dibentuk menjadi poligon untuk kemudian menjalani proses triangulasi poligon menggunakan algoritma *ear-clipping*. Algoritma *Graham scan* digunakan untuk membentuk poligon dengan memperoleh urutan koordinat searah jarum jam. Tujuannya adalah untuk menemukan *convex hull* dari sekumpulan titik, poligon cembung terkecil yang melingkupi semua titik yang diberikan.

Proses tersebut ditunjukkan pada Gambar 4. Proses ini diawali dengan: (a) mengubah koordinat menjadi sistem koordinat Cartesian dan memetakannya ke bidang Cartesian. Kemudian, koordinat tersebut disusun dalam urutan untuk membentuk poligon, (b) yang dicapai dengan menggunakan algoritma *Graham scan*. Setelah memperoleh sekumpulan segitiga dari hasil triangulasi poligon (c), ditentukan *moment centroid* dari setiap segitiga yang terbentuk (d).

Estimasi lain dilakukan dengan menggunakan rumus interpolasi Lagrange. Kedua koordinat ditampilkan pada aplikasi Android beserta koordinat titik referensi dan koordinat masukan lainnya, sebagaimana ditunjukkan pada Gambar 5. Peta ditampilkan menggunakan Google Maps API. Penanda merah menunjukkan titik kontrol, penanda hijau menunjukkan koordinat dari rumus interpolasi Lagrange, penanda biru menunjukkan hasil, dan penanda oranye menunjukkan data koordinat masukan dalam satu pengukuran.

Selanjutnya dilakukan perhitungan akurasi spasial menggunakan AEE dan RMSE. Hasilnya ditunjukkan pada Tabel I. Berdasarkan hasil yang ditampilkan pada Tabel I tersebut, koordinat dari *moment centroid* memiliki kesalahan



Gambar 5. Penanda koordinat Google Maps API.

TABEL I
HASIL PERBANDINGAN

Centroid	AEE	RMSE
Interpolasi Lagrange	0,000727995742	0,0005147707
Moment centroid	0,000642858562	0,0004545696

paling kecil dibandingkan dengan koordinat dari polinomial interpolasi Lagrange. Hal ini menunjukkan bahwa metode *moment centroid* memberikan akurasi spasial yang lebih tinggi dibandingkan dengan hasil dari polinomial interpolasi Lagrange dalam menentukan lokasi pengguna Android.

V. KESIMPULAN

Data *geolocation* dikumpulkan menggunakan ponsel pintar Android melalui jaringan seluler, dengan Wi-Fi dinonaktifkan. Proses penyaringan dilakukan pada koordinat dengan jarak terjauh dari yang lainnya. Jarak antar setiap pasangan koordinat dihitung menggunakan rumus *haversine*. Kemudian, jarak rata-rata semua pasangan dihitung. Semua koordinat dengan jarak di bawah rata-rata digunakan untuk membentuk poligon, yang kemudian menjalani proses triangulasi poligon menggunakan algoritma *ear-clipping*. Pembentukan poligon dilakukan dengan algoritma *Graham scan* untuk menentukan urutan koordinat searah jarum jam, yang akan membentuk poligon. Setelah memperoleh sekumpulan segitiga dari hasil triangulasi poligon, ditentukan titik berat momen dari setiap segitiga yang terbentuk. Berdasarkan hasil yang diperoleh dari perhitungan AEE dan RMSE, koordinat yang diperoleh dari penentuan *moment centroid* lebih akurat dan presisi hingga ke titik-titik eksak dibandingkan dengan koordinat yang diperoleh dari polinomial interpolasi Lagrange.

KONFLIK KEPENTINGAN

Penulis menyatakan bahwa tidak terdapat konflik kepentingan selama penyusunan makalah ini.

KONTRIBUSI PENULIS

Konseptualisasi, Rachmat Wahid Saleh Insani; metodologi, Rachmat Wahid Saleh Insani; perangkat lunak, Rachmat Wahid Saleh Insani; validasi, Sucipto; analisis formal, Rachmat Wahid Saleh Insani; investigasi, Sucipto; sumber daya, Rachmat Wahid Saleh Insani; kurasi data, Sucipto;

penulisan—penyusunan draf asli, Rachmat Wahid Saleh Insani; penulisan—penelaahan dan penyuntingan, Sucipto; visualisasi, Rachmat Wahid Saleh Insani; pengawasan, Rachmat Wahid Saleh Insani dan Sucipto; administrasi proyek, Sucipto; akuisisi pendanaan, Rachmat Wahid Saleh Insani.

UCAPAN TERIMA KASIH

Penelitian ini didukung oleh Lembaga Penelitian dan Pengabdian Kepada Masyarakat, Universitas Muhammadiyah Pontianak. Ucapan terima kasih disampaikan kepada rekan-rekan di Laboratorium Rekayasa Perangkat Lunak Universitas Muhammadiyah Pontianak, bagian dari Program Studi Teknik Informatika, yang telah memberikan wawasan dan keahliannya sehingga sangat membantu penelitian ini.

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, hal. 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, hal. 295–310, Sep. 2011, doi: 10.1123/ijsp.6.3.295.
- [3] N. Lambrinos dan F. Asiklari, "The introduction of GIS and GPS through local history teaching in primary school," *Eur. J. Geogr.*, vol. 5, no. 1, hal. 32–47, Mar. 2014.
- [4] K.M. Kafi dan M.B.A. Gibril, "GPS application in disaster management: A review," *Asian J. Appl. Sci.*, vol. 4, no. 1, hal. 63–69, Feb. 2016.
- [5] S.B. Im, S. Hurlbaus, dan Y.J. Kang, "Summary review of GPS technology for structural health monitoring," *J. Struct. Eng.*, vol. 139, no. 10, hal. 1653–1664, Okt. 2013, doi: 10.1061/(ASCE)ST.1943-541X.0000475.
- [6] M.R. Yousefi dan A.M. Razdari, "Application of GIS and GPS in precision agriculture (a review)," *Int. J. Adv. Biol. Biomed. Res.*, vol. 3, no. 1, hal. 7–9, Mar. 2015.
- [7] S. Lee, G. Tewolde, dan J. Kwon, "Design and implementation of vehicle tracking system using GPS/GSM/GPRS technology and smartphone application," dalam *2014 IEEE World Forum Internet Things (WF-IoT)*, 2014, hal. 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, hal. 83–105, Apr. 1994, doi: 10.1007/BF02106512.
- [9] A. Nait-Sidi-Moh, M. Bakhouya, J. Gaber, dan M. Wack, Eds. *Geopositioning and Mobility*. London, Inggris: Wiley-ISTE, 2013.
- [10] C.R. Rani dkk., "Location based services in Android," *Int. J. Adv. Eng. Technol.*, vol. 3, no. 1, hal. 209–220, Mar. 2012.
- [11] A. Nadaf dan B.K. Srinivas, "Android based fused location provider api for localization accuracy," *Int. J. Latest Trends Eng. Technol.*, vol. 8, no. 3, hal. 278–282, Mei 2017, doi: 10.21172/1.83.042.
- [12] N. Moayeri, C. Li, dan L. Shi, "Indoor localization accuracy of major smartphone location apps," dalam *2019 IEEE Wirel. Commun. Netw. Conf. (WCNC)*, 2019, hal. 1–8, doi: 10.1109/WCNC.2019.8885639.
- [13] M. Spreitzenbarth, S. Schmitt, dan F. Freiling, "Comparing sources of location data from Android smartphones," dalam *Adv. Digit. Forensics VIII 8th IFIP WG 11.9 Int. Conf. Digit. Forensics*, 2012, hal. 143–157, doi: 10.1007/978-3-642-33962-2_10.
- [14] Z. Farid, R. Nordin, dan M. Ismail, "Recent advances in wireless indoor localization techniques and system," *J. Comput. Netw. Commun.*, vol. 2013, hal. 1–12, Sep. 2013, doi: 10.1155/2013/185138.
- [15] M. Werner, *Indoor Location-Based Services: Prerequisites and Foundations*. Cham, Swiss: Springer, 2014.
- [16] H. Liu, H. Darabi, P. Banerjee, dan J. Liu, "Survey of wireless indoor positioning techniques and systems," *IEEE Trans. Syst. Man Cybern. Part C (Appl. Rev.)*, vol. 37, no. 6, hal. 1067–1080, Nov. 2007, doi: 10.1109/TSMCC.2007.905750.
- [17] M. Bin Ismail dkk., "Implementation of location determination in a wireless local area network (WLAN) environment," dalam *2008 10th Int. Conf. Adv. Commun. Technol.*, 2008, hal. 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 dan U.A. Khan, "Localization in mobile networks via virtual convex hulls," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 4, no. 1, hal. 188–201, Mar. 2018, doi: 10.1109/TSIPN.2017.2673238.
- [20] R.E. Deakin, S.C. Bird, dan R.I. Grenfell, "The centroid? Where would you like it to be?," *Cartography*, vol. 31, no. 2, hal. 153–167, Des. 2002, doi: 10.1080/00690805.2002.9714213.
- [21] Y. Sabri dan N. El Kamoun, "A distributed method for localization in large-scale sensor networks based on Jarvis," *Int. J. Comput. Appl.*, vol. 165, no. 3, hal. 1–9, Mei 2017, doi: 10.5120/ijca2017913826.
- [22] S. Singh dkk., "A novel framework to avoid traffic congestion and air pollution for sustainable development of smart cities," *Sustain. Energy Technol. Assess.*, vol. 56, hal. 1–9, Mar. 2023, doi: 10.1016/j.seta.2023.103125.
- [23] Y. Jiaxin dkk., "Objectification of tooth-marked tongue in Chinese medicine based on morphological feature extraction," *Laser Optoelectron. Prog.*, vol. 59, no. 11, hal. 1–9, Jun. 2022, doi: 10.3788/LOP202259.1117001.
- [24] C. Na dkk., "Intelligent interpretation of rock mass discontinuity based on three-dimensional laser point cloud," *Earth Sci.*, vol. 46, no. 7, hal. 2351–2361, Jul. 2021, doi: 10.3799/dqkx.2020.282.
- [25] M.R. Islam dan J.M. Kim, "A centroid-GPS model to improving positioning accuracy for a sensitive location-based system," dalam *Ubiquitous Comput. Appl. Wirel. Sens.*, 2015, hal. 187–196, doi: 10.1007/978-94-017-9618-7_18.
- [26] M.M. Din, N. Jamil, dan N.F.N.A. Aziz, "Adopting centroid and amended trilateration for better accuracy of range-based non-GPS localization," dalam *2018 4th Int. Conf. Comput. Inf. Sci. (ICCOINS)*, 2018, hal. 1–6, doi: 10.1109/ICCOINS.2018.8510605.
- [27] N.M.S. Wibawa, I.M. Sukarsa, dan 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, hal. 263–268, Jan. 2014.
- [28] M. Horemuž dan J.V. Andersson, "Polynomial interpolation of GPS satellite coordinates," *GPS Solut.*, vol. 10, no. 1, hal. 67–72, Feb. 2006, doi: 10.1007/s10291-005-0018-0.
- [29] M. Jang dkk., "Improving positional accuracy using relative measurement between Android smartphones," *Sens. Mater.*, vol. 34, no. 1, hal. 349–366, Jan. 2022, doi: 10.18494/SAM3711.
- [30] L. Lee dkk., "Comparison of accuracy and precision of GPS-enabled mobile devices," dalam *2016 IEEE Int. Conf. Comput. Inf. Technol. (CIT)*, 2016, hal. 73–82, doi: 10.1109/CIT.2016.94.
- [31] G. Van Brummelen, *Heavenly Mathematics: The Forgotten Art of Spherical Trigonometry*. Princeton, NJ, AS: Princeton Univ. Press, 2013.
- [32] G. Sharma, M.S. Singh, S.P. Aggarwal, dan R. Romero-Andrade, "Integrated observations on crustal strain-ionosphere total electron content anomalies before the earthquake," *Acta Geophys.*, vol. 71, no. 3, hal. 1173–1185, Jun. 2023, doi: 10.1007/s11600-023-01030-7.
- [33] G.R. Anggani, A.Z. Arfianto, A. Arleiny, dan B.T. Setiawan, "Ship collision realtime early warning system based on automatic identification system (AIS) data," *Indones. J. Eng. Res.*, vol. 3, no. 1, hal. 18–23, Mei 2023, doi: 10.11594/ijer.v3i1.42.
- [34] E. Wang dkk., "Aircraft positioning accuracy evaluation using QAR data: A case study on four aircraft types," *J. Aeronaut. Astronaut. Aviat.*, vol. 55, no. 1, hal. 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, hal. 132–133, Jun. 1972, doi: 10.1016/0020-0190(72)90045-2.
- [36] J. O'Rourke, "Polygon triangulation," dalam *Computational Geometry in C*, 2nd ed. New York, NY, AS: Cambridge Univ. Press, 1998, hal. 1–43.
- [37] D. Eberly, "Triangulation by ear clipping," *Geometric Tools*. Tanggal akses: 14-Okt-2023. [Online]. Tersedia: <http://www.geometrictools.com/>
- [38] J.F. Oliveira dkk., "Geometric and topological modelling of organs and vascular structures from CT data," dalam *Multi-Modality Imaging: Applications and Computational Techniques*, M.A. de Souza, H.R. Gamba, dan H. Pedrini, Eds., Cham, Swiss: Springer, 2018.
- [39] C. Cai dan Z. Tianlong, "An algorithm for triangulation applied to 3D GIS," dalam *2010 Int. Conf. Comput. Appl. Syst. Model. (ICCASM 2010)*, 2010, hal. V9-146–V9-149, doi: 10.1109/ICCASM.2010.5623065.
- [40] C. Blut, T. Blut, dan J. Blankenbach, "CityGML goes mobile: Application of large 3D CityGML models on smartphones," *Int. J. Digit. Earth*, vol. 12, no. 1, hal. 25–42, 2019, doi: 10.1080/17538947.2017.1404150.
- [41] R.G. Budynas dan J.K. Nisbett, *Shigley's Mechanical Engineering Design*, 9th ed. New York, NY, AS: McGraw Hill, 2011.
- [42] F. Ayres dan E. Mendelson, *Theory and Problems of Differential and Integral Calculus*, 3rd ed. New York, NY, AS: McGraw Hill, 1990.
- [43] Y. Lee dan W. Lim, "Shoelace formula: Connecting the area of a polygon and the vector cross product," *Math. Teach.*, vol. 110, no. 8, hal. 631–636, Apr. 2017, doi: 10.5951/matteacher.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, Virginia, AS, 1998.