

**VISUALISASI 3D RUPA BUMI
BERBASIS DATA *GDEM ASTER* 30 METER**

SKRIPSI

Oleh:

MOCHAMAD AGUNG TARECHA

NIM. 09650135



**JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI (UIN) MAULANA MALIK IBRAHIM
MALANG
2013**

**VISUALISASI 3D RUPA BUMI
BERBASIS DATA *GDEM ASTER* 30 METER**

SKRIPSI

**Diajukan kepada:
Universitas Islam Negeri Maulana Malik Ibrahim Malang
Untuk Memenuhi Salah Satu Persyaratan dalam
Memperoleh Gelar Sarjana Komputer (S. Kom)**

**Oleh:
MOCHAMAD AGUNG TARECHA
NIM. 09650135**

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI (UIN) MAULANA MALIK IBRAHIM
MALANG
2013**

HALAMAN PERSETUJUAN

**VISUALISASI 3D RUPA BUMI
BERBASIS DATA *GDEM* ASTER 30 METER**

SKRIPSI

Oleh:

MOCHAMAD AGUNG TARECHA

NIM. 09650135

Telah Diperiksa dan Disetujui untuk Diuji:

Tanggal: 1 Juli 2013

Pembimbing I,

Pembimbing II,

Dr. Cahyo Crysdian

NIP. 197404242009011008

Fatchurrochman, M.Kom

NIP. 197007312005011002

**Mengetahui,
Ketua Jurusan Teknik Informatika**

Ririen Kusumawati, M.Kom

NIP. 197203092005012002

HALAMAN PENGESAHAN

VISUALISASI 3D RUPA BUMI
BERBASIS DATA *GDEM ASTER 30 METER*

SKRIPSI

Oleh:
MOCHAMAD AGUNG TARECHA
NIM. 09650135

Telah Dipertahankan di Depan Dewan Penguji Skripsi
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
Untuk Memperoleh Gelar Sarjana Komputer (S. Kom)

Tanggal 9 Juli 2013

Susunan Dewan Penguji:

- | | | Tanda Tangan |
|--------------------|---|--------------|
| 1. Penguji Utama | : <u>Fresy Nugroho, M.T</u>
NIP. 197107222011011001 | () |
| 2. Ketua Penguji | : <u>Linda Salma Angreani, M.T</u>
NIP. 197708032009122005 | () |
| 3. Sekretaris | : <u>Dr. Cahyo Crysdian</u>
NIP. 197404242009011008 | () |
| 4. Anggota Penguji | : <u>Fatchurrochman, M.Kom</u>
NIP. 197007312005011002 | () |

Mengetahui,
Ketua Jurusan Teknik Informatika

Ririen Kusumawati, S.Si, M.Kom
NIP. 197203092005012002

PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan dibawah ini:

Nama : Mochamad Agung Tarecha

NIM : 09650135

Jurusan : Teknik Informatika

Fakultas : Sains dan Teknologi

menyatakan dengan sebenarnya bahwa skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri, bukan merupakan pengambilalihan data, tulisan atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka. Apabila dikemudian hari terbukti atau dapat dibuktikan skripsi ini hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 1 Juli 2013
Yang membuat pernyataan,

Mochamad Agung Tarecha
NIM. 09650135

MOTTO

يَتَأْتِيهَا الَّذِينَ ءَامَنُوا إِذَا قِيلَ لَكُمْ تَفَسَّحُوا فِي الْمَجَالِسِ فَافْسَحُوا
 يَفْسَحِ اللَّهُ لَكُمْ وَإِذَا قِيلَ انشُرُوا فَانشُرُوا يَرْفَعِ اللَّهُ الَّذِينَ ءَامَنُوا
 مِنْكُمْ وَالَّذِينَ أُوتُوا الْعِلْمَ دَرَجَاتٍ وَاللَّهُ بِمَا تَعْمَلُونَ خَبِيرٌ ﴿١١﴾

Hai orang-orang beriman apabila dikatakan kepadamu: "Berlapang-lapanglah dalam majlis", maka lapangkanlah niscaya Allah akan memberi kelapangan untukmu. Dan apabila dikatakan: "Berdirilah kamu", maka berdirilah, niscaya Allah akan meninggikan orang-orang yang beriman di antaramu dan orang-orang yang diberi ilmu pengetahuan beberapa derajat. Dan Allah Maha Mengetahui apa yang kamu kerjakan. (QS. Mujaadilah 58:11)



THE POWER OF DREAMS

BIARKAN IMPIAN MEMBAWAMU

KE PUNCAK - PUNCAK TERTINGGI

PERSEMBAHAN

Karya tulis ini penulis persembahkan untuk:

*Ayahanda dan Ibunda tercinta
yang telah sukses mendidik kami
dengan penuh kasih sayang dan cinta,
dengan harapan di hati
dengan do'a yang terucap,
dengan kesungguhan usaha.*

*Adik penulis Rachmad Imam Tarecha
yang telah memberikan dukungan atas semua ide - ide
dan gagasan yang telah diberikan.*

KATA PENGANTAR



Assalamualaikum Warahmatullahi Wabarakatuh

Syukur Alhamdulillah penulis haturkan kehadiran Allah SWT yang telah melimpahkan Rahmat dan Hidayah-Nya, sehingga penulis dapat menyelesaikan studi di Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang sekaligus menyelesaikan skripsi ini dengan baik.

Sholawat serta salam semoga selalu terlimpahkan kepada Nabi besar kita Nabi Muhammad SAW yang telah membimbing kita dari zaman jahiliah menuju zaman yang terang benderang yaitu Islam.

Selanjutnya penulis haturkan ucapan terima kasih seiring do'a dan harapan *jazakumullah ahsanal jaza'* kepada semua pihak yang telah membantu terselesaikannya skripsi ini. Penulis sampaikan terima kasih kepada Ayahanda dan Ibunda tercinta yang senantiasa memberikan do'a dan restunya kepada penulis selama ini, adik penulis Rachmad Imam Tarecha tercinta yang selalu aktif dalam membantu dan berdiskusi sehingga skripsi ini dapat diselesaikan dengan baik, dan juga kepada segenap sivitas akademika UIN Maulana Malik Ibrahim Malang terutama Prof. Dr. H. Mudjia Rahardjo, M.Si, selaku Rektor UIN Maulana Malik Ibrahim Malang saat ini dan juga tak lupa Prof. Dr. Imam Suprayogo selaku mantan rektor periode sebelumnya, Dr. Cahyo Crysdiyan dan Fatchurrochman, M.Kom selaku dosen pembimbing skripsi, yang telah banyak memberikan pengarahan.

Tak lupa kepada semua pihak yang turut mendukung sehingga skripsi ini dapat diselesaikan dengan baik.

Penulis menyadari bahwa dalam penyusunan skripsi ini masih terdapat kekurangan dan penulis berharap semoga skripsi ini bisa memberikan manfaat kepada para pembaca khususnya bagi penulis secara pribadi. Amin Ya Rabbal Alamin.

Wassalamu'alaikum Warahmatullahi Wabarakatuh

Malang, 1 Juli 2013

Penulis,

Mochamad Agung Tarecha

DAFTAR ISI

KATA PENGANTAR	viii
DAFTAR GAMBAR	xiii
DAFTAR TABEL	xv
ABSTRAK	xvi
ABSTRACT	xvii
BAB I	1
PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	6
1.3 Batasan Masalah	6
1.4 Tujuan Penelitian	7
1.5 Manfaat Penelitian	7
1.6 Sistematika Penulisan	8
1.7 Tahapan Penelitian	9
BAB II	10
TINJAUAN PUSTAKA	10
2.1 Landasan Peraturan	10
2.2 <i>Digital Elevation Model</i>	11
2.3 Model Data Raster	12
2.4 Data ASTER	14
2.5 <i>Remapping Point</i>	15
2.6 <i>Euclidean Distance</i>	17
2.7 <i>Algoritma Rounding</i>	19
2.8 <i>Algoritma Brute Force Closest-Pair</i>	21
2.9 <i>Forward Chaining</i>	23
2.10 Koreksi Tampilan	24
2.11 Interpolasi Linier	27
2.12 <i>Polygonal Modeling</i>	31
BAB III	32
ANALISIS DAN PERANCANGAN	32
3.1 Kebutuhan Sistem	32
3.1.1 Sumber Data	32

3.1.2	Kebutuhan <i>Platform</i>	33
3.1.3	Deskripsi Sistem	34
3.2	Perancangan dan Implementasi Algoritma	36
3.2.1	Normalisasi	38
3.2.2	<i>Remapping Point</i>	42
3.2.3	Seleksi <i>Round-half-down</i>	43
3.2.4	Seleksi <i>Brute-Force Closest-Pair</i>	47
3.2.5	Pilih.....	49
3.2.6	Seleksi Data	63
3.2.7	Interpolasi Linier.....	65
3.2.8	Visualisasi	68
3.2.9	Koreksi Tampilan.	69
3.2.10	<i>Map Matriks to Nearest Coordinat</i>	75
3.2.11	Batas Matriks Kecil	77
3.2.12	<i>Map Objek</i>	79
3.3	Perancangan dan Implementasi Antarmuka.....	81
3.4	Perancangan dan Implementasi <i>Database</i>	90
BAB IV	92
HASIL DAN PEMBAHASAN	92
4.1	Langkah – langkah pengujian	92
4.2	Hasil Pengujian	94
4.2.1	Pengujian Ketepatan dan Kecepatan Pencarian Titik Tengah	94
4.2.2	Pengujian Ketelitian Ketinggian Titik Tengah	97
4.2.3	Pengujian Ketepatan Penyusunan Empat Ubin Data Terdekat.....	100
4.2.4	Pengujian Interpolasi Linier.....	103
4.2.5	Pengujian Kompatibilitas pada Perangkat	106

4.3	Pembahasan Pengujian	109
4.4	Integrasi Visualisasi 3D Rupa Bumi dengan Islam	112
BAB V		115
PENUTUP		115
5.1	Kesimpulan	115
5.2	Saran	116
DAFTAR PUSTAKA		117
LAMPIRAN.....		119



DAFTAR GAMBAR

Gambar 1.1: (a) Lokasi pesawat Shukoi Super Jet 100 jatuh, (b) tim SAR sedang merencanakan strategi evakuasi.....	1
Gambar 1.2: Tahapan penelitian	9
Gambar 2.1: Struktur mode data raster	12
Gambar 2.2: Contoh raster elevasi	13
Gambar 2.3: Ilustrasi <i>remapping point</i>	15
Gambar 2.4: Ilustrasi <i>euclidean distance</i> satu dimensi	17
Gambar 2.5: Ilustrasi euclidian distance dua dimensi.....	18
Gambar 2.6: <i>Pseudocode</i> algoritma <i>brute force closest-pair</i>	21
Gambar 2.7: Ilustrasi <i>brute force closest-pair</i>	22
Gambar 2.8: Sudut pandang 3D.....	24
Gambar 2.9: <i>Colormap</i> MATLAB.....	26
Gambar 2.10: Ilustrasi perubahan <i>colormap</i>	27
Gambar 2.11: Interpolasi linier untuk meramal kurva nilai $y = f(\bar{x})$	28
Gambar 2.12: Contoh interpolasi linier.....	30
Gambar 2.13: Objek pohon.....	31
Gambar 3.1: Perancangan sistem	34
Gambar 3.2: <i>Flowchart</i> sistem secara garis besar.....	37
Gambar 3.3: Ilustrasi tipe input, (a) tipe titih tengah, (b) tipe titik tepi	38
Gambar 3.4: Ilustrasi normalisasi	39
Gambar 3.5: <i>Flowchart</i> normalisasi.....	40
Gambar 3.6: <i>Pseudocode</i> fungsi normalisasi	41
Gambar 3.7: Ilustrasi pembulatan radius	42
Gambar 3.8: <i>Flowchart remapping point</i>	42
Gambar 3.9: <i>Pseudocode</i> fungsi <i>remapping point</i>	43
Gambar 3.10: Ilustrasi <i>round-half-down</i>	44
Gambar 3.11: <i>Flowchart round-half-down</i>	45
Gambar 3.12: <i>Pseudocode</i> fungsi <i>round-half-down</i>	46
Gambar 3.13: Ilustrasi <i>brute force closest-pair</i>	47
Gambar 3.14: <i>Flowchart brute force closest-pair</i>	48
Gambar 3.15: <i>Pseudocode</i> fungsi <i>brute-force closest-pair</i>	49
Gambar 3.16: Pembagian empat kuadran	50
Gambar 3.17: Susunan empat ubin terdekat	51
Gambar 3.18: Ilustrasi dasar <i>rule basis</i> pengetahuan.....	52
Gambar 3.19: <i>Flowchart forward chaining</i>	55
Gambar 3.20: <i>Pseudocode</i> fungsi pilih	60
Gambar 3.21: <i>Pseudocode</i> fungsi <i>generatefile</i>	62
Gambar 3.22: <i>Flowchart</i> seleksi data	63
Gambar 3.23: <i>Pseudocode</i> seleksi data	64
Gambar 3.24: <i>Flowchart</i> interpolasi linier.....	65
Gambar 3.25: <i>Pseudocode</i> fungsi interpolasi linier	66
Gambar 3.26: Perbandingan interpolasi.....	67
Gambar 3.27: <i>Pseudocode</i> visualisasi.....	68
Gambar 3.28: Kontur tanah menggunakan fungsi <i>mesh</i>	69
Gambar 3.29: <i>Flowchar</i> koreksi skala	70
Gambar 3.30: <i>Pseudocode</i> koreksi skala	70

Gambar 3.31: Gunung Semeru sebelum koreksi skala	71
Gambar 3.32: Gunung Semeru setelah koreksi skala.....	72
Gambar 3.33: Gunung Semeru setelah koreksi poros z	72
Gambar 3.34: <i>Flowchart</i> koreksi <i>colormap</i>	73
Gambar 3.35: <i>Pseudocode</i> koreksi <i>colormap</i>	74
Gambar 3.36: Perbedaan koreksi <i>colormap</i>	74
Gambar 3.37: Ilustrasi fungsi <i>mapMatrikstoNearestCoordinat</i>	75
Gambar 3.38: <i>Flowchart</i> fungsi <i>mapMatrikstoNearestCoordinat</i>	76
Gambar 3.39: <i>Pseudocode</i> fungsi <i>mapMatrikstoNearestCoordinat</i>	77
Gambar 3.40: Ilustrasi fungsi <i>batasMatriksKecil</i>	78
Gambar 3.41: <i>Flowchart</i> fungsi <i>batasMatriksKecil</i>	78
Gambar 3.42: <i>Pseudocode</i> fungsi <i>batasMatriksKecil</i>	79
Gambar 3.43: <i>Flowchart</i> fungsi <i>mapObjek</i>	80
Gambar 3.44: <i>Pseudocode</i> fungsi <i>mapObjek</i>	80
Gambar 3.45: Perancangan halaman beranda	81
Gambar 3.46: Implementasi halaman beranda.....	82
Gambar 3.47: Perancangan halaman visualisasi	83
Gambar 3.48: Implementasi halaman visualisasi titik tengah.....	84
Gambar 3.49: Implementasi halaman visualisasi titik tepi	85
Gambar 3.50: Perancangan halaman manajemen data.....	86
Gambar 3.51: Implementasi halaman manajemen data	86
Gambar 3.52: Perancangan halaman tutorial	87
Gambar 3.53: Implementasi halaman tutorial.....	88
Gambar 3.54: Perancangan halaman tentang peneliti.....	89
Gambar 3.55: Implementasi halaman tentang peneliti	89
Gambar 3.56: Perancangan ERD	91
Gambar 4.1: Grafik waktu pencarian titik terdekat.....	96
Gambar 4.2: Grafik ketinggian	99
Gambar 4.3: Grafik persentase perbedaan ketinggian rata-rata	100
Gambar 4.4: Susunan ubin data <i>GDEM ASTER</i>	101
Gambar 4.5: Ilustrasi titik visualisasi.....	101
Gambar 4.6: Hasil visualisasi tanpa penyusunan empat ubin data	102
Gambar 4.7: Hasil visualisasi dengan penyusunan empat ubin data.....	103
Gambar 4.8: Pohon yang akan divisualisasikan.....	104
Gambar 4.9: Visualisasi pohon tanpa interpolasi.....	105
Gambar 4.10: Visualisasi pohon dengan interpolasi.....	105
Gambar 4.11: Pengujian pada Windows 7	106
Gambar 4.12: Pengujian pada Linux Mint.....	107
Gambar 4.13: Pengujian pada Tablet Android.....	107
Gambar 4.14: Pengujian pada Smartphone Android.....	108

DAFTAR TABEL

Tabel 2.1: Karakteristik <i>GDEM ASTER</i>	14
Tabel 3.1: Kategori data input.....	90
Tabel 3.2: Lokasi data input.....	90
Tabel 4.1: Pengujian algoritma <i>brute force closest-pair</i>	95
Tabel 4.2: Pengujian algoritma <i>round-half-down</i>	95
Tabel 4.3: Pengujian ketelitian ketinggian titik tengah.....	98
Tabel 4.4: Titik koordinat pohon	104



ABSTRAK

Tarecha, Mochamad Agung. 2013. **Visualisasi 3D Rupa Bumi Berbasis Data GDEM ASTER 30 Meter**. Skripsi. Jurusan Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.

Pembimbing: (I) Dr. Cahyo Crysdiyan (II) Fatchurrochman, M.Kom

Kata kunci : visualisasi 3D, rupa bumi, *GDEM ASTER*

Penggunaan informasi spasial sangat vital untuk mendukung kegiatan perencanaan pembangunan, pemanfaatan sumber daya alam, militer, strategi penanggulangan bencana dan lain sebagainya. Informasi spasial umumnya divisualisasikan dalam bentuk 2D, ketinggian wilayah direpresentasikan menggunakan kontur topografi. Pendekatan ini sulit untuk memperoleh detail kontur suatu wilayah, seiring dengan berkembangnya teknologi kontur topografi 2D kurang memadai dalam memberikan informasi kontur tanah karena mengharuskan pengguna mengimajinasikan garis kontur menjadi ketinggian tanah. Oleh karena itu, visualisasi 3D rupa bumi dewasa ini diperlukan guna mempermudah pengguna dalam mendapatkan informasi mengenai kontur tanah.

Visualisasi 3D rupa bumi membutuhkan data *Global Digital Elevation Model (GDEM)* yang merupakan data ketinggian tanah yang membentuk kontur bumi. Salah satu data *GDEM* tersebut adalah *Advanced Spaceborne Thermal Emission and Reflection Radiometer (ASTER)* yang memiliki ketelitian 30 meter. Data *GDEM ASTER* tersebut diolah menggunakan beberapa metode dan algoritma antara lain metode *euclidean distance* untuk menghitung jarak antara dua titik, metode *forward chaining* untuk menyusun empat ubin data *GDEM ASTER* yang terdekat dengan titik tengah, algoritma *round half down* dan *brute force closest-pair* untuk mencari pasangan titik terdekat. Algoritma *round half down* dan *brute force closest-pair* dibandingkan untuk membuktikan algoritma mana yang paling cepat mencari titik terdekat. Untuk meningkatkan ketelitian digunakan interpolasi linier sehingga ketelitian yang sebelumnya 30 meter menjadi lebih tinggi yaitu 15 meter. Aplikasi visualisasi disajikan dalam bentuk halaman *web* yang dapat diakses oleh pengguna melalui *web browser*. Hasil akhirnya sebuah aplikasi visualisasi yang dapat digunakan pada berbagai perangkat.

ABSTRACT

Tarecha, Mochamad Agung. 2013. **3D Earth Appearance Visualization Based Data ASTER GDEM 30 Meters**. Thesis. Department of Informatics, Faculty of Science and Technology of the State Islamic University of Maulana Malik Ibrahim Malang.
Supervisor: (I) Dr. Cahyo Crysdiandian (II) Fatchurrochman, M.Kom

Keywords: 3D visualization, earth appearance, GDEM ASTER

The use of spatial information is vital to support the development planning activities, use of natural resources, military, disaster management strategies and so forth. Spatial information is usually visualized in 2D, altitude regions represented using topographic contours. This approach is difficult to obtain detailed contour of an area, along with the development of technology, 2D topographic contour is inadequate in providing contour information by requiring users to imagine the elevation contour lines into height of land. Therefore, the 3D visualization of the earth today needed a way to facilitate users in getting information about the contour of the land.

3D visualization requires a Global Digital Elevation Model (GDEM) which is the ground elevation data that form the contours of the earth. One of these is the data GDEM Advanced Spaceborne Thermal Emission and Reflection Radiometer (ASTER) which has an accuracy of 30 meters. The ASTER GDEM data is processed using several methods and algorithms such as euclidean distance method to calculate the distance between two points, forward chaining method to compose four ASTER GDEM Data tiles closest to the midpoint, round-half-down algorithms and brute force closest-pair to find the closes pair point. Algorithms round half down and brute force closest-pair compared to prove which one is the fastest algorithm find the closes pair point. To improve the accuracy, linear interpolation used previous 30 meters of the accuracy to 15 meters higher. Visualization application is presented in the form of web pages that can be accessed by users through a web browser. The end result is a visualization application that can be used on various devices.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Saat ini informasi mengenai data spasial (referensi ruang kebumihan / *georeference*) banyak dibutuhkan di berbagai bidang. Sebagai contoh informasi data spasial dibutuhkan dalam perencanaan pembangunan, pemanfaatan sumber daya alam, militer, dan strategi penanggulangan bencana. Salah satu contoh adalah strategi penanggulangan bencana pesawat Sukhoi Super Jet 100 yang menabrak tebing Gunung Salak pada tanggal 9 Mei 2012, informasi mengenai kontur Gunung Salak diperlukan oleh Tim SAR guna perencanaan evakuasi korban pesawat Sukhoi Super Jet 100 di tebing gunung yang terjal. Lokasi pesawat tersebut jatuh dan bagaimana perencanaan strategi evakuasi korban ditunjukkan pada Gambar 1.1 berikut.



(Sumber: <http://beritabogor.com/>)

Gambar 1.1 (a) Lokasi pesawat Shukoi Super Jet 100 jatuh,
(b) tim SAR sedang merencanakan strategi evakuasi.

Karena pentingnya penggunaan data spasial guna perencanaan pembangunan nasional maka Pemerintah Republik Indonesia melalui Peraturan Presiden Republik Indonesia Nomor 85 Tahun 2007 membuat sebuah Peraturan Presiden yang mengelola data spasial yaitu Jaringan Data Spasial Nasional (JDSN). Informasi data spasial yang tersaji diharapkan dapat dicerna dengan mudah oleh pengguna, informasi tersebut biasanya ditampilkan dalam peta 2D seperti yang terlihat pada Gambar 1.1 diatas, untuk menyajikan data berupa ketinggian tanah digunakan peta topografi yang memiliki garis kontur yang menyatakan ketinggian tanah pada garis yang sama. Namun sejalan dengan perkembangan teknologi, visual 3D dapat memberikan informasi yang lebih mudah dicerna oleh pengguna. Informasi rupa bumi haruslah sesuai atau paling tidak mendekati bentuk rupa bumi yang sebenarnya, oleh karena itu dibutuhkan sebuah informasi yang konsisten untuk mendukung berbagai kebutuhan yang telah dipaparkan.

Setelah dikaji dalam Al-Qur'an ternyata Allah SWT. telah menyinggung mengenai rupa bumi sebagaimana Firman Allah dalam Surah Qaaf ayat 7-8 sebagai berikut.

وَالْأَرْضَ مَدَدْنَاهَا وَأَلْقَيْنَا فِيهَا رَوَاسِيَ وَأَنْبَتْنَا فِيهَا مِنْ كُلِّ زَوْجٍ بَهِيجٍ
تَبْصِرَةً وَذِكْرًا لِكُلِّ عَبْدٍ مُنِيبٍ ﴿٧﴾

Artinya :

Dan Kami hamparkan bumi itu dan Kami letakkan padanya gunung-gunung yang kokoh dan Kami tumbuhkan padanya segala macam tanaman yang indah

dipandang mata, untuk menjadi pelajaran dan peringatan bagi tiap-tiap hamba yang kembali (mengingat Allah).

Bagaimana mengenai bumi yang dihamparkan dan diletakkan gunung-gunung yang kokoh dan ditumbuhkan segala macam tanaman yang indah telah diisyaratkan dalam Al-Qur'an untuk menjadi pelajaran dan peringatan bagi tiap-tiap hamba yang kembali (mengingat Allah).

Ternyata dijelaskan lagi pada Surah Al Anbiyaa' ayat 31 sebagai berikut.

وَجَعَلْنَا فِي الْأَرْضِ رَوَاسِيَ أَنْ تَمِيدَ بِهِمْ وَجَعَلْنَا فِيهَا فِجَاجًا
سُبُلًا لَعَلَّهُمْ يَهْتَدُونَ ﴿٣١﴾

Artinya :

Dan telah Kami jadikan di bumi ini gunung-gunung yang kokoh supaya bumi itu (tidak) goncang bersama mereka dan telah Kami jadikan (pula) di bumi itu jalan-jalan yang luas, agar mereka mendapat petunjuk.

Ternyata gunung-gunung yang kokoh itu bertujuan agar bumi tidak goncang bersama manusia, bagaimana fungsi gunung yang dibahas disini adalah sebagai penyeimbang bumi dan disebutkan pula dijadikan jalan-jalan yang luas di bumi agar manusia mendapatkan petunjuk. Fungsi gunung yang kokoh sebagai pasak bumi dijelaskan lagi pada Surah An Naba' ayat 6-7 sebagai berikut.

أَلَمْ نَجْعَلِ الْأَرْضَ مِهْدًا ﴿٦﴾ وَالْجِبَالَ أَوْتَادًا ﴿٧﴾

Artinya :

Bukankah Kami telah menjadikan bumi itu sebagai hamparan?, dan gunung-gunung sebagai pasak?

Sekali lagi pada ayat ini disinggung kembali bahwa Allah menjadikan bumi itu sebagai hampan dan gunung-gunung sebagai pasak. Bagaimana gunung berfungsi sbagai pasak bumi yang menjadikan bumi tidak bergoncang.

Apa pelajaran dan peringatan mengenai bumi yang dihamparkan dan diletakkannya gunung-gunung yang kokoh ? Bagaimana bumi itu dijadikan jalan-jalan yang luas agar manusia mendapat petunjuk ? Inilah yang menjadi pembuka kajian ilmiah penelitian visualisasi rupa bumi dilakukan agar dapat mengetahui pelajaran dan peringatan apa yang terkandung dalam surah-surah tersebut. Penelitian ini ingin mengungkap bagaimana Allah memberikan pelajaran dan peringatan bagi manusia melalui visualisasi rupa bumi pada umumnya dan visualisasi gunung pada khususnya.

Rupa bumi dibentuk oleh elevasi (ketinggian) tanah, sebuah wilayah tentu memiliki elevasi tanah diatas permukaan laut (dpl) yang berbeda-beda, data elevasi tersebut bisa dimodelkan dalam bentuk visual 3D. Salah satu penyedia layanan visualisasi rupa bumi adalah Google Earth, layanan ini memungkinkan pengguna melihat kontur bumi secara 3D. Data yang digunakan pada layanan Google Earth adalah data *Global Digital Elevation Model (GDEM) Shuttle Radar Topography Mission (SRTM)* yang memiliki tingkat ketelitian 90 m (30 meter untuk wilayah Amerika Serikat).

Data dengan ketelitian tersebut dinilai kurang detail karena tingkat ketelitiannya rendah, namun pada tahun 2009 telah dirilis data *GDEM ASTER (Advanced Spaceborne Thermal Emission and Reflection Radiometer)* yang memiliki ketelitian 30 meter. Dan pada Oktober 2011 telah dirilis *GDEM ASTER*

versi 2 yang telah diperbaiki meliputi data kosong diisi dan banyak artefak dihapus. Aplikasi yang akan dibangun menggunakan data *GDEM ASTER* diharapkan lebih baik dan visual yang dihasilkan lebih detail daripada aplikasi yang telah ada sebelumnya dimana menggunakan data *GDEM SRTM*.

Sebuah fail data *GDEM ASTER* yang memiliki ukuran 3601 x 3601 dimana jarak antar titik nya adalah 30 meter memiliki panjang 108.000 meter, dan memiliki luas 11.664.000.000 m², masalah yang timbul untuk memvisualisasikan rupa bumi dengan satu file utuh adalah :

- Area terlalu luas sehingga kurang fokus ketika area yang lebih kecil yang diinginkan.
- Sumber daya sistem (processor, memory) akan terpakai sangat tinggi untuk memvisualisasikan 3D rupa bumi pada satu fail penuh sehingga dapat menyebabkan sistem *crash*.
- Ketika suatu wilayah terbagi dalam lebih dari satu fail data maka wilayah tersebut tidak bisa untuk divisualisasikan seluruhnya.

Sehingga untuk menanggulangi masalah-masalah yang timbul tersebut diperlukan beberapa metode. *Euclidean distance* merupakan metode yang dipakai untuk menghitung jarak antar dua titik. Untuk menentukan jarak terdekatnya menggunakan perbandingan antar 2 titik terdekat masing-masing poros (x dan y) menggunakan algoritma *round half down* dan *brute force closest-pair*, kedua algoritma tersebut dibandingkan untuk mengetahui mana yang lebih cepat. Sedangkan masalah ketika suatu wilayah terbagi lebih dari satu file data menggunakan metode *forward chaining* untuk mencari 4 file wilayah terdekat.

Setelah disusun kemudian data tersebut diseleksi untuk memvisualisasikan wilayah yang diinginkan saja. Untuk meningkatkan ketelitian digunakan interpolasi linier sehingga ketelitian yang sebelumnya 30 meter dapat lebih tinggi yaitu 15 meter. Karena ketelitian meningkat diharapkan gambar jauh lebih detail dan dapat memberikan informasi yang lebih akurat.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dikemukakan, maka rumusan masalah yang akan dijawab dalam penelitian ini adalah sebagai berikut:

1. Apakah algoritma *round-half-down* dan *brute force closest-pair* dapat mencari pasangan titik terdekat dengan tepat ? dan manakah yang lebih cepat ?
2. Apakah metode *euclidean distance* dan *forward chaining* dapat mendukung visualisasi 3D rupa bumi ?
3. Apakah metode interpolasi linier dapat meningkatkan ketelitian dalam visualisasi 3D rupa bumi ?

1.3 Batasan Masalah

Mengingat luasnya wilayah rupa bumi yang ada, maka dalam pembahasan visualisasi 3D rupa bumi dibatasi pada :

1. Data yang akan disediakan meliputi peta Pulau Jawa dan Bali, sedangkan objek akan difokuskan pada visualisasi 3D gunung. Namun jika ingin wilayah lain yang akan divisualisasikan cukup menambah file *GDEM ASTER* nya saja tanpa merubah aplikasi.
2. Mengingat keterbatasan sistem perangkat keras yang ada maka visualisasi maksimal dibatasi 501 x 501 titik dengan interval asli 30 meter sehingga luas

maksimal adalah $15.000 \text{ m} \times 15.000 \text{ m} = 225.000.000 \text{ m}^2$. Namun jika dikemudian hari sistem perangkat keras yang digunakan lebih canggih maka batasan visualisasi bisa ditambahkan bahkan dihilangkan.

3. Sistem koordinat yang digunakan adalah koordinat lintang dan bujur, satuan koordinat yang digunakan adalah *decimal degree*.
4. Menggunakan interpolasi linier untuk mendapatkan resolusi yang lebih tinggi dengan interval maksimal 15 meter.

1.4 Tujuan Penelitian

Tujuan dilakukannya penelitian ini adalah sebagai berikut.

1. Menerapkan algoritma *round-half-down* dan *brute force closest-pair* dalam mencari titik tengah terdekat kemudian membandingkan mana yang lebih cepat.
2. Menerapkan metode *euclidean distance* dan *forward chaining* untuk mendukung visualisasi 3D rupa bumi.
3. Menerapkan metode interpolasi linier untuk memperhalus tampilan dan membuktikan bahwa dengan interpolasi tersebut tingkat ketelitian visualisasi meningkat.

1.5 Manfaat Penelitian

Hasil dari penelitian ini diharapkan akan memberikan manfaat terhadap pengembangan data spasial antara lain :

1. Memudahkan pihak-pihak yang membutuhkan informasi mengenai visual rupa bumi mendapatkan visual secara cepat dan mudah.
2. Memberikan informasi mengenai visual rupa bumi dengan tingkat keakuratan yang tinggi.

3. Menjadi dasar pengembangan Sistem Informasi Geografis 3D yang membutuhkan data elevasi bumi.

1.6 Sistematika Penulisan

Penulisan skripsi ini tersusun dalam lima bab dengan sistematika penulisan sebagai berikut :

- BAB I Pendahuluan

Bab ini berisi pembahasan tentang latar belakang masalah, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, dan sistematika penulisan.

- BAB II Tinjauan Pustaka

Bab ini berisi pembahasan tentang visualisasi 3D rupa bumi : *state of the art, digital elevation model, model data raster, data ASTER, remapping point, euclidean distance, algoritma rounding, algoritma brute force closest-pair, forward chaining, koreksi tampilan, interpolasi linier, dan polygonal modeling.*

- BAB III Analisis dan Perancangan

Bab ini menjelaskan tentang perancangan dan implementasi visualisasi 3D rupa bumi berbasis data *GDEM ASTER 30 meter* yang meliputi kebutuhan sistem, perancangan dan implementasi algoritma, antarmuka, dan *database*.

- BAB IV Hasil dan Pembahasan

Bab ini berisi pembahasan tentang pengujian aplikasi meliputi pengujian ketepatan dan kecepatan pencarian titik tengah, pengujian ketelitian ketinggian titik tengah, pengujian ketepatan penyusunan empat ubin data terdekat, pengujian

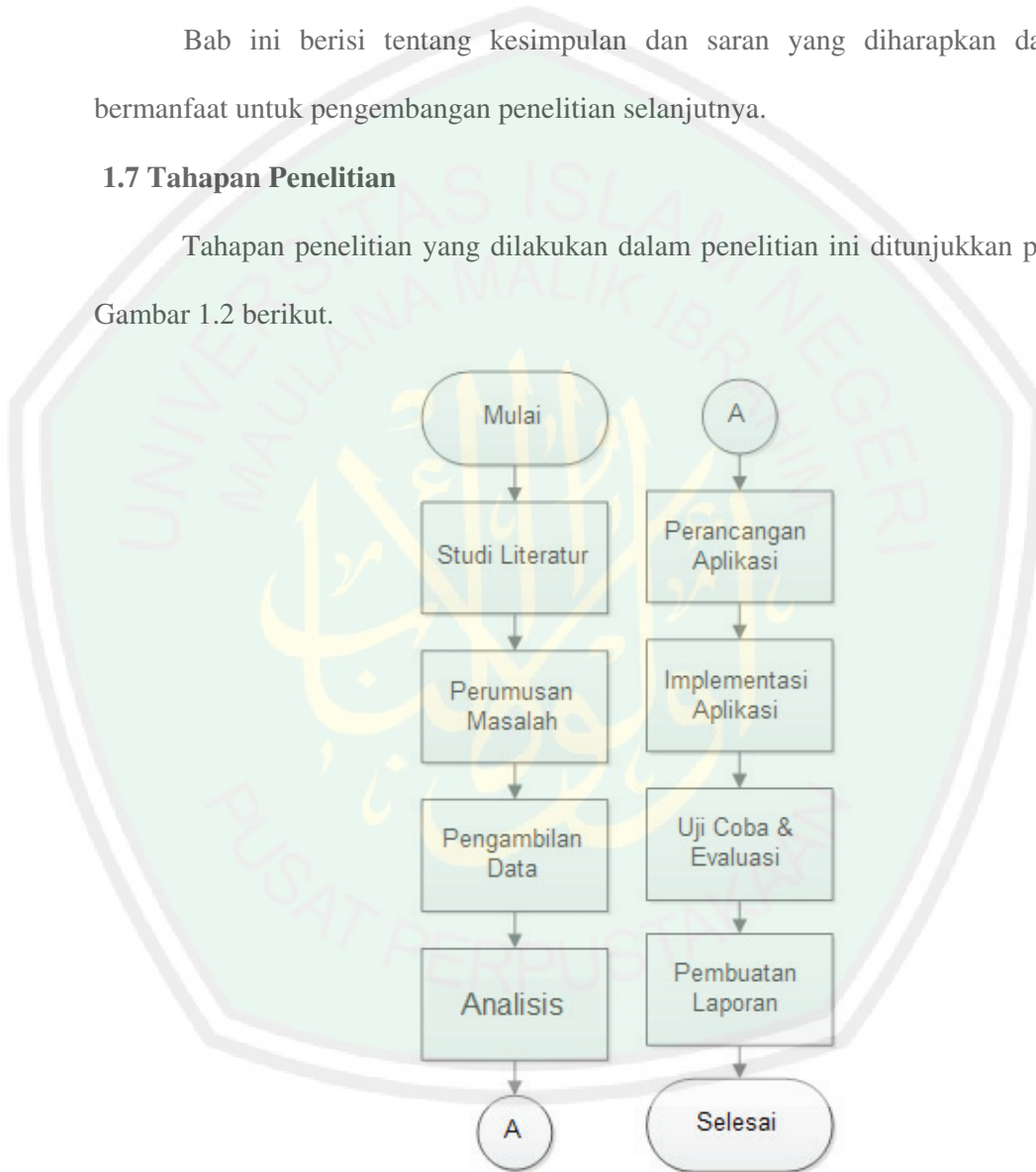
interpolasi linier, pengujian kompatibilitas pada perangkat, dan integrasi visualisasi 3D rupa bumi dengan Islam.

- BAB V Penutup

Bab ini berisi tentang kesimpulan dan saran yang diharapkan dapat bermanfaat untuk pengembangan penelitian selanjutnya.

1.7 Tahapan Penelitian

Tahapan penelitian yang dilakukan dalam penelitian ini ditunjukkan pada Gambar 1.2 berikut.



Gambar 1.2 Tahapan penelitian.

BAB II

TINJAUAN PUSTAKA

2.1 Landasan Peraturan

Penggunaan data spasial sangat penting dalam memberikan informasi untuk berbagai aspek sehingga Pemerintah Republik Indonesia menerbitkan peraturan yang menjadi payung hukum dalam pengelolaan data spasial. Data spasial adalah data hasil pengukuran, pencatatan, dan pencitraan terhadap suatu unsur keruangan yang berada di bawah, pada, atau diatas permukaan bumi dengan posisi keberadaannya mengacu pada sistem koordinat nasional. Ruang lingkup penelitian ini berada dalam wilayah data spasial maka dapat disimpulkan bahwa penelitian ini dibutuhkan, berikut beberapa landasan peraturan tentang data spasial.

- **Peraturan Presiden Republik Indonesia Nomor 85 Tahun 2007 Tentang Jaringan Data Spasial Nasional**

Data spasial memiliki peranan penting dalam memberikan informasi untuk menunjang pembangunan suatu negara. Informasi yang didapatkan diharapkan dapat menjadi salah satu sumber informasi dalam melakukan perencanaan pembangunan nasional, oleh karena pentingnya data spasial tersebut maka Pemerintah Republik Indonesia melalui Peraturan Presiden Republik Indonesia Nomor 85 Tahun 2007 membuat sebuah Peraturan Presiden yang mengelola data spasial yaitu Jaringan Data Spasial Nasional (JDSN).

- **Instruksi Presiden Republik Indonesia Nomor 6 Tahun 2012 Tentang Penyediaan, Penggunaan, Pengendalian Kualitas, Pengolahan dan Distribusi Data Satelit Penginderaan Jauh Resolusi Tinggi**

Dalam rangka pemanfaatan data satelit penginderaan jauh resolusi tinggi secara cepat untuk meningkatkan efisiensi penggunaan anggaran negara dan menghindari duplikasi alokasi anggaran dalam pengadaan data satelit penginderaan jauh resolusi tinggi, dengan ini menginstruksikan menggunakan citra tegak satelit penginderaan jauh resolusi tinggi yang disediakan oleh Badan Informasi Geospasial berdasarkan data satelit penginderaan jauh resolusi tinggi yang disediakan oleh Lembaga Penerbangan dan Antariksa Nasional. Data *GDEM ASTER* merupakan data satelit penginderaan jauh. Dimana Indonesia termasuk dalam cakupan data *GDEM ASTER*.

2.2 Digital Elevation Model

Sebuah model elevasi digital adalah model digital atau representasi 3D dari permukaan bumi yang dibuat dari data elevasi tanah. Menurut Jacobsen (2003) *Digital Elevation Model* (DEM) berdasarkan jumlah titik dengan x , y , dan z koordinat yang menggambarkan tanah gundul. DEM mungkin disusun dalam bentuk raster atau bentuk acak. Poros x dan y merupakan poros koordinat bumi, sedangkan poros z merupakan data ketinggian. DEM diperlukan untuk beberapa keperluan seperti generasi *ortoimages*, perencanaan penanggulangan banjir, pengendalian erosi, pertanian dan lain sebagainya.

2.3 Model Data Raster

Dalam bentuk paling sederhana, raster terdiri dari matiks sel / piksel yang diatur dalam baris dan kolom (atau grid), di mana setiap sel berisi nilai yang mewakili informasi seperti suhu, ketinggian, dan lain sebagainya, seperti yang ditunjukkan dalam Gambar 2.1 di bawah. Data Raster dapat diperoleh melalui foto udara, citra satelit, gambar digita, *remote sensing* atau peta yang dipindai (ESRI, 2009).

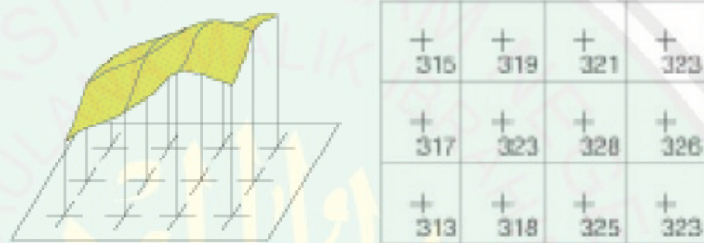


(Sumber: *Understanding Raster Data*)

Gambar 2.1 Struktur model data raster .

Pada Gambar 2.1 merupakan gambar struktur sel data raster, dalam set data raster setiap sel (yang juga dikenal sebagai piksel) memiliki nilai. Nilai-nilai sel merupakan fenomena digambarkan oleh data raster ditetapkan seperti kategori, magnitude, ketinggian, atau nilai spektral. Nilai data elevasi dapat disimpan dalam model data raster, dimana ketinggian tiap titik direpresentasikan dalam bentuk

nilai numerik. Untuk beberapa jenis data, nilai sel mewakili nilai yang diukur pada titik tengah sel. Sel tersebut memiliki posisi dan dikonversikan menjadi posisi baris dan kolom. Sel – sel tersebut disusun dalam suatu urutan yang membentuk data raster. Dalam penelitian ini tiap sel tersebut menyimpan data DEM. Contohnya adalah raster elevasi yang diilustrasikan melalui Gambar 2.2 berikut.



(Sumber: *Understanding Raster Data*)

Gambar 2.2 Contoh raster elevasi

Pada Gambar 2.2 sebuah data raster yang menyimpan nilai elevasi tanah. Posisi sel direpresentasikan dalam baris dan kolom, nilai sel merupakan data elevasi. Kumpulan sel tersebut membentuk matriks elevasi, dimana *longitude* direpresentasikan menjadi kolom matriks, dan *latitude* direpresentasikan menjadi baris matriks.

2.4 Data ASTER

Salah satu data elevasi tanah dengan model data raster adalah data *GDEM ASTER*. Untuk melakukan visualisasi rupa bumi dibutuhkan data DEM, data *GDEM ASTER* merupakan data *Global Digital Elevation Model (GDEM)* berbasis model data raster. *The Ministry of Economy, Trade, and Industry (METI) of Japan* dan *the United States National Aeronautics and Space Administration (NASA)* bersama-sama mengumumkan rilis dari *Advanced Spaceborne Thermal Emission and Reflection Radiometer (ASTER) Global Digital Elevation Model (GDEM)* versi 2 pada 17 Oktober 2011 (LP DAAC, 2011). Cakupan *GDEM ASTER* membentang dari 83° lintang utara hingga 83° lintang selatan, meliputi 99 persen dari daratan bumi (Melebihi cakupan SRTM dari 60° lintang utara hingga 56° lintang selatan). Karakteristik data *GDEM ASTER* ditunjukkan pada tabel 2.1 berikut.

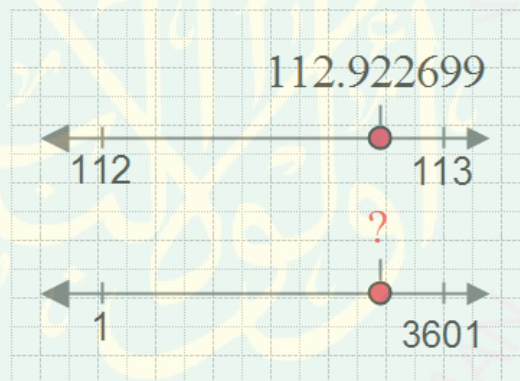
Tabel 2.1: Karakteristik *GDEM ASTER*

Ukuran ubin	3601 x 3601 (1° x 1°)
Ukuran piksel	1 <i>arc-second</i>
Sistem Koordinat Geografis	Geografis lintang dan bujur
Format <i>Output DEM</i>	GeoTIFF, signed 16-bit, vertikal dalam satuan meter Direferensikan ke the WGS84/EGM96 geoid
Nilai spesial <i>Digital Number</i>	-9999 untuk piksel kosong dan 0 untuk permukaan laut
Cakupan	Utara 83° sampai selatan 83°, 22.702 ubin

(Sumber : ASTER GDEM 2 README)

2.5 Remapping Point

Untuk mengkonversikan titik koordinat bumi lintang dan bujur menjadi titik baris dan kolom digunakan *remapping point*. Konsep *remapping* adalah mengubah nilai pada suatu *range* nilai menjadi nilai dengan *range* nilai yang lain, seperti konversi suhu °Celcius ke °Reamur atau sebaliknya. Misal sebuah titik lokasi di puncak Gunung Semeru memiliki koordinat lintang -8.111658° dan bujur 112.922699° . Pada file data ASTER satu file memiliki panjang 1° , dimana 1° terdapat 3601 titik artinya antara 112° sampai 113° memiliki 3601 titik. Untuk mengetahui posisi titik pada matriks dari bujur 112.922699° dapat dilakukan dengan cara *remapping point* seperti pada Gambar 2.3 berikut.



Gambar 2.3 Ilustrasi *remapping point*.

Mekanisme *remapping point* (disadur dari <http://arduino.cc/en/Reference/map>) ditunjukkan pada Rumus 2.1 berikut.

$$titikOut = lowOut + (valueIn - lowIn) * \frac{highOut - lowOut}{highIn - lowIn} \quad (2.1)$$

dimana :

titikOut = titik hasil map

lowOut = titik batas bawah hasil konversi yaitu 1

highOut = titik batas atas hasil konversi yaitu 3601

lowIn = titik batas bawah titik asal dalam contoh ini adalah 112

highIn = titik batas atas titik asal dalam contoh ini adalah 113

valueIn = titik asal yang akan dikonversi

sehingga :

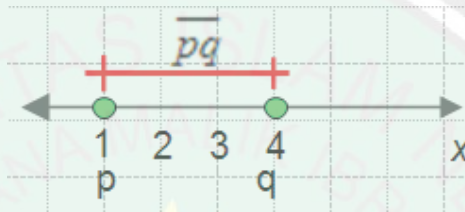
$$titikOut = 1 + (112.922699 - 112) * \frac{3601 - 1}{113 - 112}$$

$$titikOut = 1 + 0.922699 * \frac{3600}{1}$$

$$titikOut = 3322.7164$$

2.6 Euclidean Distance

Euclidean distance adalah sebuah metode untuk mengukur panjang antara dua titik, *euclidean distance* antara p dan q adalah panjang ruas garis yang menghubungkan mereka (\overline{pq}) (Wikipedia). Perhitungan *euclidean distance* satu dimensi diilustrasikan pada Gambar 2.4 berikut.



Gambar 2.4 Ilustrasi *euclidean distance* satu dimensi

Perhitungan jarak titik menghitung jarak tiap poros x dan y satu persatu maka menggunakan rumus *euclidean distance* satu dimensi yang ditunjukkan pada Rumus 2.2 berikut.

$$\begin{aligned} dx &= |x1 - x2| \\ dy &= |y1 - y2| \end{aligned} \quad (2.2)$$

dimana: $dx = distance$ / jarak $x1$ ke $x2$

$dy = distance$ / jarak $y1$ ke $y2$

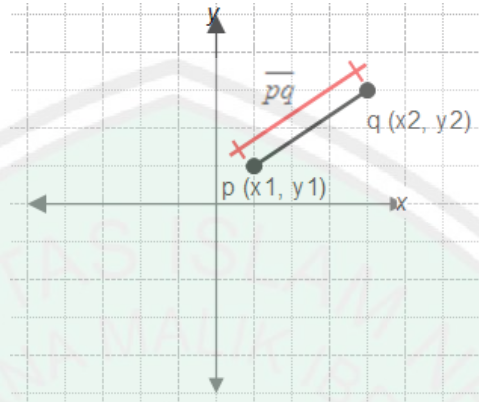
$x1 =$ titik koordinat $x1$ pada poros x

$x2 =$ titik koordinat $x2$ pada poros x

$y1 =$ titik koordinat $y1$ pada poros y

$y2 =$ titik koordinat $y2$ pada poros y

Perhitungan *euclidean distance* dua dimensi diilustrasikan pada Gambar 2.5 berikut.



Gambar 2.5 Ilustrasi *euclidian distance* dua dimensi.

Untuk perhitungan jarak titik dua dimensi dimana menghitung kedua titik pada poros x dan y secara bersamaan maka menggunakan rumus *euclidian distance* yang ditunjukkan pada Rumus 2.3 berikut.

$$d(p, q) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \quad (2.3)$$

dimana : $d(p, q) = \text{distance} / \text{jarak}$

x_1 = titik koordinat **p** pada poros x

x_2 = titik koordinat **q** pada poros x

y_1 = titik koordinat **p** pada poros y

y_2 = titik koordinat **q** pada poros y

2.7 Algoritma Rounding

Menurut Maxfield (2006) bahwa fakta paling mendasar terkait dengan pembulatan adalah mengubah beberapa kuantitas dari presisi yang lebih tinggi ke presisi yang lebih rendah. Misal pengukuran panjang pensil menggunakan jangka sorong mendapat hasil 10,967 cm bisa dibulatkan menjadi 11 cm. Ada beberapa jenis algoritma pembulatan ini, dapat dipilih yang sesuai untuk meminimalkan efek kehilangan presisi. Algoritma tersebut antara lain *round-up*, *round-down*, *round-toward-nearest*, *arithmetic rounding*, *round-half-up*, *round-half-down*, *round-half-even*, *round-half-odd*, *round-toward-zero*, *round-away-from-zero*, *round-ceiling*, *round-floor*, *truncation*, *round-alternate*, dan *round-random*. Dalam penelitian ini hanya menggunakan 3 algoritma pembulatan yaitu *round-ceiling*, *round-floor*, dan *round-half-down*, yang dapat dijelaskan sebagai berikut.

- *Round-ceiling* → Algoritma ini mengacu pada pembulatan ke arah positif tak terhingga. Dalam kasus angka positif, hasilnya akan tetap tidak berubah jika angka yang akan dibuang adalah nol, jika tidak maka akan dibulatkan. Contoh, 3.0 akan dibulatkan menjadi 3, tapi 3.1, 3.2, 3.3, 3.4, 3.5, 3.6, 3.7, 3.8, dan 3.9 semua akan dibulatkan menjadi 4. Sebagai perbandingan, dalam kasus angka negatif, angka yang tidak diinginkan dibuang begitu saja. Misalnya, -3.0, -3.1, -3.2, -3.3, -3.4, -3.5, -3.6, -3.7, -3.8, -3.9 dan semua akan dibulatkan ke -3.

Dalam MATLAB algoritma ini diimplementasikan menggunakan fungsi *ceil*.

- *Round-floor* → Lawan dari *round-ceiling*, algoritma ini mengacu pada pembulatan menuju tak terhingga negatif. Dalam kasus angka positif, angka yang tidak diinginkan dibuang begitu saja. Misalnya, 3.0, 3.1, 3.2, 3.3, 3.4, 3.5, 3.6, 3.7, 3.8, dan 3.9 semua akan dibulatkan menjadi 3. Dalam kasus angka negatif, hasilnya akan tetap tidak berubah jika angka yang akan dibuang adalah nol, jika tidak maka akan dibulatkan ke bawah. Misalnya, -3.0 akan dibulatkan ke -3, tapi -3.1, -3.2, -3.3, -3.4, -3.5, -3.6, -3.7, -3.8, -3.9 dan semua akan dibulatkan ke -4. Dalam MATLAB algoritma ini diimplementasikan menggunakan fungsi *floor*.
- *Round-half-down* → Algoritma ini mengacu pada pembulatan ke nilai terdekat, sedang nilai tengah akan dibulatkan ke bawah. Misal 3.0, 3.1, 3.2, 3.3, 3.4 akan dibulatkan ke 3, sedangkan 3.6, 3.7, 3.8, 3.9, 4.0 akan dibulatkan ke 4, dan 3.5 dimana nilai tersebut adalah nilai tengah maka karena *half-down* maka pembulatannya mulai nilai tengah kebawah yaitu 3. Algoritma ini nantinya akan digunakan untuk mencari titik terdekat diantara titik bilangan bulat pada baris dan kolom matriks. Untuk membandingkan jarak titik yang diperoleh dari hasil *remapping point* dengan titik terdekat pada batas atas dan batas bawah titik tersebut digunakan *euclidean distance* satu dimensi.

2.8 Algoritma Brute Force Closest-Pair

Menurut Levitin (2012) algoritma *brute force closest-pair* adalah menghitung jarak antara setiap pasang titik yang berbeda dan menemukan sepasang dengan jarak terkecil, dan tidak menghitung pasangan yang sama dua kali, untuk menghindarinya hanya pasangan titik (p_i, p_j) yang $i < j$. Diasumsikan bahwa titik-titik tersebut ditetapkan secara standar oleh titik (x, y) pada koordinat kartesius dan bahwa jarak antara dua titik $p_i(x_i, y_i)$ dan $p_j(x_j, y_j)$ adalah *euclidean distance* dua dimensi standar seperti pada Rumus 2.4 dibawah ini.

$$d(p_i, p_j) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \quad (2.4)$$

Pseudocode dari algoritma *brute force closest-pair* adalah ditunjukkan pada Gambar 2.6 berikut.

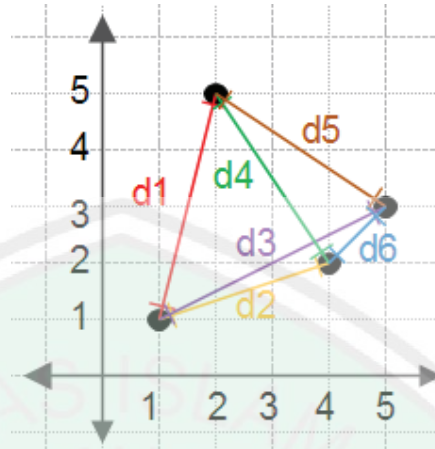
```

algorithm BruteForceClosestPair(P)
//Finds distance between two closest points
  in the plane by brute force
//Input: A list P of n (n ≥ 2) points p1(x1, y1),
  . . . , pn(xn, yn)
//Output: The distance between the closest pair of points

d ← ∞
for i ← 1 to n-1 do
  for j ← i + 1 to n do
    d ← min(d, sqrt((xi - xj)2 + (yi - yj)2))
return d

```

Gambar 2.6 *Pseudocode* algoritma *brute force closest-pair*.



Gambar 2.7 Ilustrasi *brute force closest-pair*.

Seperti pada Gambar 2.7 misal terdapat 4 buah titik yaitu $(1,1)$, $(2,5)$, $(4,2)$, dan $(5,3)$, masing-masing titik tersebut dihitung *euclidean distance* sebagai berikut.

$$(1,1) \rightarrow (2,5) = d1,$$

$$(1,1) \rightarrow (4,2) = d2,$$

$$(1,1) \rightarrow (5,3) = d3,$$

$$(2,5) \rightarrow (4,2) = d4,$$

$$(2,5) \rightarrow (5,3) = d5,$$

$$(4,2) \rightarrow (5,3) = d6.$$

Masing-masing dihitung *euclidean distance* pada tiap titik anggota yang didefinisikan sebanyak satu kali karena $(1,1) \rightarrow (2,5)$ atau $(2,5) \rightarrow (1,1)$ adalah titik yang sama dengan jarak yang sama. Jarak tersebut dibandingkan satu sama lain dan diseleksi mana yang paling kecil. Pasangan titik dengan jarak terkecil adalah hasil dari pencarian menggunakan algoritma ini.

2.9 Forward Chaining

Menurut Russell (2003) *forward chaining* adalah salah satu metode penalaran saat menggunakan *inference rule* dan dapat dijelaskan secara logis sebagai aplikasi modus ponens. *Forward chaining* dimulai dengan data yang tersedia dan menggunakan aturan inferensi untuk mengekstrak lebih banyak data sampai tujuan tercapai. Sebuah mesin inferensi yang menggunakan *forward chaining* diterapkan untuk orde pertama klausa dan dapat diimplementasikan secara efisien. Klausa seperti situasi → respon (jika situasi a maka respon b) berguna untuk sistem yang membuat kesimpulan untuk menanggapi informasi yang diberikan. Inferensi dimulai dari keadaan awal yaitu informasi atau fakta yang ada kemudian dicoba untuk dicocokkan dengan basis pengetahuan kemudian diperoleh tujuan yang diharapkan.

Forward chaining dapat diterapkan secara luas dalam berbagai bidang, salah satu contoh adalah sistem pakar untuk mendiagnosa suatu penyakit berdasarkan gejala yang timbul. Basis pengetahuan dapat diperoleh dari para ahli dalam hal ini dokter, petugas kesehatan, atau pihak-pihak yang memiliki pengetahuan tentang suatu penyakit dan gejala-gejala yang ditimbulkannya. Gejala penyakit digunakan sebagai input kemudian data tersebut diekstraksi menggunakan mesin inferensi berdasarkan basis pengetahuan yang diperoleh sebelumnya, hasil keluaran merupakan beberapa kemungkinan penyakit yang timbul dengan gejala yang diinputkan. Sedemikian luas penggunaan *forward chaining* dalam mengekstraksi data sehingga memungkinkan untuk digunakan dalam penelitian ini.

2.10 Koreksi Tampilan

- **Penyamaan skala poros x , y , dan z**

Menurut Crys dian (2009) ketika sebuah data *DEM* divisualisasikan terdapat masalah yaitu satuan pada poros x dan y berbeda dengan skala pada poros z . Korelasi tidak seimbang antara koordinat horizontal yang direpresentasikan oleh x - y ke vertikal koordinat yang direpresentasikan oleh z . Tampak bahwa nilai vertikal mendominasi nilai horizontal karena sumbu poros tersebut tidak sama satuannya. Pada data *GDEM ASTER* interval tiap titik tanpa interpolasi pada poros x dan y adalah 30 meter, sedangkan interval tiap titik pada poros z adalah 1 meter. Matriks data tersebut dikonversikan baris menjadi poros x , kolom menjadi poros y , isi tiap sel matriks menjadi poros z .

Untuk melihat visualisasi secara 3D sudut pandang dirumuskan seperti yang terlihat pada Rumus 2.5 berikut.

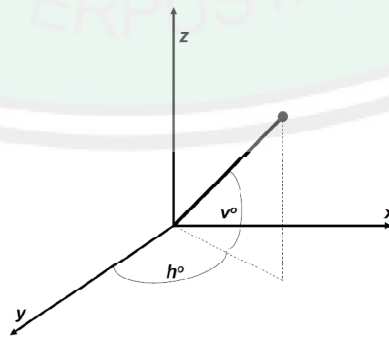
$$V = (h, v) \quad (2.5)$$

dimana : V = sudut pandang dari visualisasi 3D

h = ketinggian horizontal dari poros y

v = ketinggian vertikal dari bidang $x - y$

Berikut ilustrasi visualisasi 3D ditunjukkan pada Gambar 2.8 berikut.



(Sumber : 3D Visualization of Spatial Objects using Elevation Model)

Gambar 2.8 Sudut pandang 3D.

Untuk mengatasi masalah di atas, mekanisme untuk memaksa grid yang sama dibentuk untuk skala kedua sumbu vertikal dan horizontal dikembangkan dengan menggunakan pendekatan *thresholding*. Untuk menyamakan skala interval pada poros x dan y dengan poros z maka diambil poros x atau y terpanjang, Panjang dari poros tersebut dikalikan dengan interval kemudian dijadikan batas atas dari poros z (z_{max}) sedangkan batas bawah dari poros z (z_{min}) adalah 0 karena nilai 0 adalah permukaan laut. Setelah itu tampilan dikunci dengan mengunci rasio tampilan 3D menggunakan “*axis vis3d*” di MATLAB agar tampilan tidak berubah-ubah ketika rotasi objek 3D. Kemudian agar poros z yang kosong tidak perlu ditampilkan maka dipotong lagi z_{min} = nilai matriks minimal dan z_{max} = nilai matriks maksimal jika $z_{max} > z_{min}$.

Formula dari penyamaan satuan interval ditunjukkan pada Rumus 2.6 berikut.

$$z_{max} = \begin{cases} interval * (x_{max} - x_{min}) - 1 & \leftrightarrow x_{max} \geq y_{max} \\ interval * (y_{max} - y_{min}) - 1 & \leftrightarrow x_{max} < y_{max} \end{cases} \quad (2.6)$$

dimana:

x_{min} = 1 (index pertama matriks pada MATLAB adalah 1).

y_{min} = 1 (index pertama matriks pada MATLAB adalah 1).

x_{max} = panjang baris matriks.

y_{max} = panjang kolom matriks.

-1 = karena visualisasi dimulai pada ketinggian 0 pada poros z .

- **Penyesuaian Color Map**

Color Map adalah fungsi yang memetakan / mengubah warna dari satu (sumber) gambar untuk warna gambar lain (target) (Wikipedia) . Sebuah *color map* dapat disebut sebagai algoritma yang menghasilkan fungsi pemetaan atau algoritma yang mengubah warna gambar.

Pada penelitian ini untuk memvisualisasikan 3D rupa bumi menggunakan *colormap* MATLAB. Berikut adalah macam – macam *colormap* yang ada pada MATLAB ditunjukkan pada Gambar 2.9.



(Sumber: MATLAB)

Gambar 2.9 *Colormap* MATLAB.

Colormap yang digunakan adalah *colormap Jet* seperti yang tampak ada gambar diatas. *Colormap* tersebut memiliki *range* warna mulai dari biru tua hingga merah tua. *Colormap* memberikan warna pada visualisasi dengan data elevasi terendah dengan warna biru tua sampai dengan tertinggi dengan merah tua. Masalah yang muncul adalah ketika ketinggian lautan dengan tepi daratan berbeda tipis dengan beberapa meter saja. Sehingga warna antara lautan dimana

memiliki ketinggian 0 meter dengan daratan berkisar antara biru tua dengan biru yang bagi sebagian orang secara relatif melihatnya tidak jelas antara batas tersebut. Oleh karena itu diperlukan penyesuaian *colormap* agar batas antara daratan dan lautan terlihat jelas. Penyesuaian tersebut dilakukan dengan cara mengganti nilai terendah *colormap jet* yang semula berwarna biru tua dengan warna putih jika poros $z \leq 0$ artinya jika ketinggian terendah adalah lautan. Sehingga jika poros z ada yang bernilai ≤ 0 maka *colormap* akan berubah seperti yang diilustrasikan pada Gambar 2.10 berikut.



(Sumber: MATLAB)

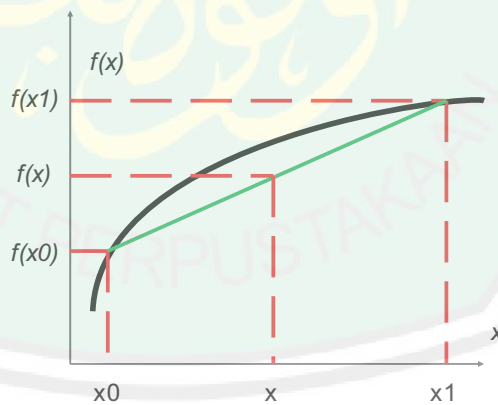
Gambar 2.10 Ilustrasi perubahan *colormap*.

2.11 Interpolasi Linier

Menurut Pujiyanta (2007) interpolasi digunakan untuk memperkirakan nilai (tengah) di antara titik-titik dari satu set nilai yang sudah diketahui. Dalam arti yang lebih luas, interpolasi merupakan upaya mendefinisikan suatu fungsi dekatkan suatu fungsi analitik yang tidak diketahui atau pengganti fungsi rumit yang tak mungkin diperoleh persamaan analitiknya. Dalam penelitian ini interpolasi digunakan untuk memperhalus kontur tanah karena data yang sebelumnya memiliki interval antar titik 30 meter menjadi 15 meter dengan penambahan titik elevasi ditengah-tengah titik elevasi yang ada sehingga intervalnya menjadi semakin pendek mengakibatkan tingkat kepresisiannya meningkat. Misalnya menempatkan objek berupa pohon pada area visualisasi, untuk mencari titik

terdekat posisi pohon akan lebih presisi pada interval 15 meter daripada 30 meter. Interpolasi dapat ditingkatkan lagi menjadi 7.5 meter, 3.75 meter dan seterusnya namun dengan konsekuensi data yang akan diproses semakin besar dan memerlukan sumber daya komputer yang lebih besar sehingga mengakibatkan waktu untuk memproses data lebih lama, oleh karena itu pada penelitian ini dibatasi interpolasi hanya sampai 15 meter mengingat sumber daya komputer yang terbatas.

Metode interpolasi ada beberapa macam yaitu interpolasi linier, interpolasi kuadrat, interpolasi polinomial, interpolasi rasional, dan lain sebagainya. Pada penelitian ini digunakan salah satu metode interpolasi yaitu interpolasi linier. Interpolasi linier untuk meramal kurva nilai $y = f(\bar{x})$ dapat dilakukan dengan menganggap bahwa y_k dan y_{k+1} dihubungkan oleh suatu garis lurus, seperti yang terlihat pada Gambar 2.11 berikut.



(Sumber : Komputasi Numerik dengan Matlab)

Gambar 2.11 Interpolasi linier untuk meramal kurva nilai $y = f(\bar{x})$.

Secara *geometric* peramalan grafik gambar 2.9 di atas yang menghubungkan titik (x_k, y_k) dengan titik (x_{k+1}, y_{k+1}) dapat dinyatakan oleh Rumus 2.7 berikut.

$$\bar{y} = y_k + \frac{y_{k+1} - y_k}{x_{k+1} - x_k} (\bar{x} - x_k) \quad (2.7)$$

Dengan demikian hasil yang diperoleh akan benar (eksak) memang merupakan fungsi linier. Persamaan tersebut dapat disederhanakan dalam interpolasi linier matriks menggunakan nilai rata-rata karena \bar{y} yang dicari nilainya merupakan nilai pada titik tengah matriks yang berada diantara titik-titik matriks.

$$A = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \rightarrow B = \begin{bmatrix} a & \frac{a+b}{2} & b \\ \frac{a+c}{2} & \frac{a+b+c+d}{4} & \frac{b+d}{2} \\ c & \frac{c+d}{2} & d \end{bmatrix} \quad (2.8)$$

Perhatikan Rumus 2.8 diatas misal sebuah matriks A berordo 2x2 akan diinterpolasi dengan menyisipkan 1 nilai diantara titik-titik yang ada maka persamaannya ditunjukkan pada Rumus 2.9 berikut.

$$B = A_{(m*2-1) \times (n*2-1)} \quad (2.9)$$

Matriks B menjadi berordo 3x3, jika matriks B diinterpolasi lagi bisa menjadi berordo 5x5 dan seterusnya. Untuk membuktikan bahwa nilai rata-rata merupakan interpolasi linier maka bisa dibuktikan dengan perhitungan yang ditunjukkan pada Gambar 2.12 berikut.

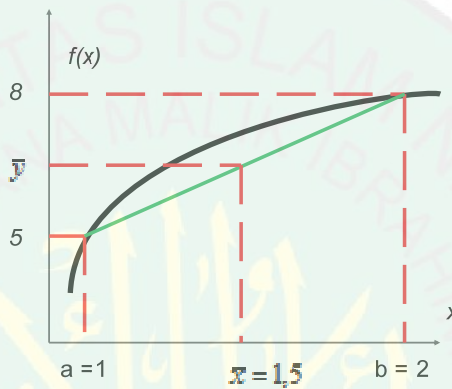
Misal

titik a = 5, titik pertama

titik b = 8, titik kedua

titik \bar{x} adalah titik tengah antara a dan b

berapakah nilai \bar{y} ?



Gambar 2.12 Contoh interpolasi linier.

$$\begin{aligned}\bar{y} &= y_k + \frac{y_{k+1} - y_k}{x_{k+1} - x_k} (\bar{x} - x_k) \\ \bar{y} &= 5 + \frac{8-5}{2-1} (1,5 - 1) \\ \bar{y} &= 5 + \frac{3}{1} (0,5) \\ \bar{y} &= 5 + 1,5 \\ \bar{y} &= 6,5\end{aligned}$$

Bila menggunakan nilai rata-rata adalah sebagai berikut.

$$\begin{aligned}\bar{y} &= \frac{a+b}{2} \\ \bar{y} &= \frac{5+8}{2} \\ \bar{y} &= 6,5\end{aligned}$$

Dengan perhitungan tersebut dapat disimpulkan bahwa hasilnya sama, sehingga dalam interpolasi linier matriks menggunakan nilai rata – rata untuk penyederhanaan proses perhitungannya.

2.12 Polygonal Modeling

Menurut Crys dian (2011) pendekatan *polygonal modeling* adalah metode sederhana namun efektif untuk membangun model kota menggunakan visualisasi 3D. Dengan menggunakan pendekatan ini berbagai fitur kota seperti bangunan, vegetasi, dan jalan dapat dikembangkan dengan cara yang efisien berdasarkan bentuk tepi objek. Informasi spasial yang kompleks biasanya meliputi perkembangan model 3D kota dapat dikelola secara sederhana. Penelitian ini akan mengimplementasikan *polygonal modeling* yang telah diteliti sebelumnya, objek yang akan divisualisasikan adalah pohon. Tujuan implementasi visualisasi pohon ini adalah membuktikan bagaimana *interpolasi* memberikan dampak visualisasi menjadi semakin detail dan akurat. Perbedaan akan terlihat ketika objek pohon divisualisasikan sebelum dan setelah interpolasi. Objek pohon ditunjukkan pada Gambar 2.13 berikut.



(Sumber: *Development of 3D City Model using Polygonal Modeling Approach*)

Gambar 2.13 Objek pohon.

BAB III ANALISIS DAN PERANCANGAN

3.1 Kebutuhan Sistem

Kebutuhan sistem diperlukan sebagai tahapan dasar analisis sistem yang dibangun, yaitu meliputi sumber data, kebutuhan *platform*, deskripsi sistem.

3.1.1 Sumber Data

Sumber data yang digunakan adalah data GDEM ASTER V2 yang merupakan produk dari METI dan NASA dirilis tanggal 17 Oktober 2011 dapat diunduh secara gratis di <http://gdex.cr.usgs.gov/gdex/>, data ini adalah *digital elevation model* yang berisi matriks nilai ketinggian tanah atau kontur bumi yang memiliki ketelitian hingga 30 meter. Jika dikemudian hari terdapat *update* versi GDEM ASTER maka *download* data tersebut, proses *update* data dilakukan oleh *administrator server* dengan cara meletakkan fail data pada folder *c:/maps/* dan tidak terdapat fasilitas *update* data yang dapat dilakukan oleh pengguna.

Data yang digunakan sebagai pembandingan diperoleh dari Google Earth dan GPS. Data pembandingan yang digunakan didapat dengan akuisisi data langsung menggunakan GPS Garmin 80 Mil, dan GPS yang ada pada *smartphone* Samsung Galaxy Young, akuisisi data bertempat pada Gunung Panderman Batu pada tanggal 10 Februari 2013. Dari pendataan tersebut diperoleh 22 data titik koordinat dan ketinggian pada GPS Garmin 80 Mil dan 17 data titik koordinat dan ketinggian pada GPS Samsung Galaxy Young. Pengambilan data antara kedua GPS tersebut dilakukan pada titik yang sama, data pada GPS Samsung Galaxy Young lebih sedikit yaitu 17 karena pada saat akuisisi data baterai *smartphone*

tersebut habis terlebih dahulu. Kemudian data pada masing-masing titik tersebut dibandingkan dengan ketinggian yang ada pada Google Earth dan aplikasi yang dibuat.

3.1.2 Kebutuhan Platform

Analisis kebutuhan merupakan analisis terhadap komponen-komponen yang digunakan untuk membuat sistem. Analisis kebutuhan ini terbagi menjadi dua macam, yaitu komponen perangkat lunak dan perangkat keras sebagai berikut.

1. Perangkat Lunak

Perangkat lunak adalah istilah umum untuk data yang diformat dan disimpan secara digital, termasuk program komputer, dokumentasinya, dan berbagai informasi yang bisa dibaca dan ditulis oleh komputer. Perangkat lunak yang dibutuhkan dalam pembuatan aplikasi ini antara lain:

- a. Sistem Operasi Windows 7.
- b. Web server Internet Information Service (IIS) 7.5.
- c. Database Management System (DBMS) Microsoft SQL Server 2008 Express Edition.
- d. Matlab R2012b.
- e. Microsoft Visual Studio 2010.
- f. Web browser Mozilla Firefox 18.0.1.

2. Perangkat Keras

Perangkat keras komputer adalah semua bagian fisik komputer. Perangkat keras yang dibutuhkan dalam pembuatan aplikasi ini antara lain:

- a. PC AMD Phenom XII 3.5 GHz, RAM DDR3 3GB.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Saat ini informasi mengenai data spasial (referensi ruang kebumihan / *georeference*) banyak dibutuhkan di berbagai bidang. Sebagai contoh informasi data spasial dibutuhkan dalam perencanaan pembangunan, pemanfaatan sumber daya alam, militer, dan strategi penanggulangan bencana. Salah satu contoh adalah strategi penanggulangan bencana pesawat Sukhoi Super Jet 100 yang menabrak tebing Gunung Salak pada tanggal 9 Mei 2012, informasi mengenai kontur Gunung Salak diperlukan oleh Tim SAR guna perencanaan evakuasi korban pesawat Sukhoi Super Jet 100 di tebing gunung yang terjatuh. Lokasi pesawat tersebut jatuh dan bagaimana perencanaan strategi evakuasi korban ditunjukkan pada Gambar 1.1 berikut.



(Sumber: <http://beritabogor.com/>)

Gambar 1.1 (a) Lokasi pesawat Shukoi Super Jet 100 jatuh,
(b) tim SAR sedang merencanakan strategi evakuasi.

Karena pentingnya penggunaan data spasial guna perencanaan pembangunan nasional maka Pemerintah Republik Indonesia melalui Peraturan Presiden Republik Indonesia Nomor 85 Tahun 2007 membuat sebuah Peraturan Presiden yang mengelola data spasial yaitu Jaringan Data Spasial Nasional (JDSN). Informasi data spasial yang tersaji diharapkan dapat dicerna dengan mudah oleh pengguna, informasi tersebut biasanya ditampilkan dalam peta 2D seperti yang terlihat pada Gambar 1.1 diatas, untuk menyajikan data berupa ketinggian tanah digunakan peta topografi yang memiliki garis kontur yang menyatakan ketinggian tanah pada garis yang sama. Namun sejalan dengan perkembangan teknologi, visual 3D dapat memberikan informasi yang lebih mudah dicerna oleh pengguna. Informasi rupa bumi haruslah sesuai atau paling tidak mendekati bentuk rupa bumi yang sebenarnya, oleh karena itu dibutuhkan sebuah informasi yang konsisten untuk mendukung berbagai kebutuhan yang telah dipaparkan.

Setelah dikaji dalam Al-Qur'an ternyata Allah SWT. telah menyinggung mengenai rupa bumi sebagaimana Firman Allah dalam Surah Qaaf ayat 7-8 sebagai berikut.

وَالْأَرْضَ مَدَدْنَاهَا وَأَلْقَيْنَا فِيهَا رَوَاسِيَ وَأَنْبَتْنَا فِيهَا مِنْ كُلِّ زَوْجٍ بَهِيجٍ
تَبْصِرَةً وَذِكْرًا لِكُلِّ عَبْدٍ مُنِيبٍ ﴿٧﴾

Artinya :

Dan Kami hamparkan bumi itu dan Kami letakkan padanya gunung-gunung yang kokoh dan Kami tumbuhkan padanya segala macam tanaman yang indah

dipandang mata, untuk menjadi pelajaran dan peringatan bagi tiap-tiap hamba yang kembali (mengingat Allah).

Bagaimana mengenai bumi yang dihamparkan dan diletakkan gunung-gunung yang kokoh dan ditumbuhkan segala macam tanaman yang indah telah diisyaratkan dalam Al-Qur'an untuk menjadi pelajaran dan peringatan bagi tiap-tiap hamba yang kembali (mengingat Allah).

Ternyata dijelaskan lagi pada Surah Al Anbiyaa' ayat 31 sebagai berikut.

وَجَعَلْنَا فِي الْأَرْضِ رَوَاسِيَ أَنْ تَمِيدَ بِهِمْ وَجَعَلْنَا فِيهَا فِجَاجًا
سُبُلًا لَّعَلَّهُمْ يَهْتَدُونَ ﴿٣١﴾

Artinya :

Dan telah Kami jadikan di bumi ini gunung-gunung yang kokoh supaya bumi itu (tidak) goncang bersama mereka dan telah Kami jadikan (pula) di bumi itu jalan-jalan yang luas, agar mereka mendapat petunjuk.

Ternyata gunung-gunung yang kokoh itu bertujuan agar bumi tidak goncang bersama manusia, bagaimana fungsi gunung yang dibahas disini adalah sebagai penyeimbang bumi dan disebutkan pula dijadikan jalan-jalan yang luas di bumi agar manusia mendapatkan petunjuk. Fungsi gunung yang kokoh sebagai pasak bumi dijelaskan lagi pada Surah An Naba' ayat 6-7 sebagai berikut.

أَلَمْ نَجْعَلِ الْأَرْضَ مِهْدًا ﴿٦﴾ وَالْجِبَالَ أَوْتَادًا ﴿٧﴾

Artinya :

Bukankah Kami telah menjadikan bumi itu sebagai hamparan?, dan gunung-gunung sebagai pasak?

Sekali lagi pada ayat ini disinggung kembali bahwa Allah menjadikan bumi itu sebagai hampan dan gunung-gunung sebagai pasak. Bagaimana gunung berfungsi sbagai pasak bumi yang menjadikan bumi tidak bergoncang.

Apa pelajaran dan peringatan mengenai bumi yang dihamparkan dan diletakkannya gunung-gunung yang kokoh ? Bagaimana bumi itu dijadikan jalan-jalan yang luas agar manusia mendapat petunjuk ? Inilah yang menjadi pembuka kajian ilmiah penelitian visualisasi rupa bumi dilakukan agar dapat mengetahui pelajaran dan peringatan apa yang terkandung dalam surah-surah tersebut. Penelitian ini ingin mengungkap bagaimana Allah memberikan pelajaran dan peringatan bagi manusia melalui visualisasi rupa bumi pada umumnya dan visualisasi gunung pada khususnya.

Rupa bumi dibentuk oleh elevasi (ketinggian) tanah, sebuah wilayah tentu memiliki elevasi tanah diatas permukaan laut (dpl) yang berbeda-beda, data elevasi tersebut bisa dimodelkan dalam bentuk visual 3D. Salah satu penyedia layanan visualisasi rupa bumi adalah Google Earth, layanan ini memungkinkan pengguna melihat kontur bumi secara 3D. Data yang digunakan pada layanan Google Earth adalah data *Global Digital Elevation Model (GDEM) Shuttle Radar Topography Mission (SRTM)* yang memiliki tingkat ketelitian 90 m (30 meter untuk wilayah Amerika Serikat).

Data dengan ketelitian tersebut dinilai kurang detail karena tingkat ketelitiannya rendah, namun pada tahun 2009 telah dirilis data *GDEM ASTER (Advanced Spaceborne Thermal Emission and Reflection Radiometer)* yang memiliki ketelitian 30 meter. Dan pada Oktober 2011 telah dirilis *GDEM ASTER*

versi 2 yang telah diperbaiki meliputi data kosong diisi dan banyak artefak dihapus. Aplikasi yang akan dibangun menggunakan data *GDEM ASTER* diharapkan lebih baik dan visual yang dihasilkan lebih detail daripada aplikasi yang telah ada sebelumnya dimana menggunakan data *GDEM SRTM*.

Sebuah fail data *GDEM ASTER* yang memiliki ukuran 3601 x 3601 dimana jarak antar titik nya adalah 30 meter memiliki panjang 108.000 meter, dan memiliki luas 11.664.000.000 m², masalah yang timbul untuk memvisualisasikan rupa bumi dengan satu file utuh adalah :

- Area terlalu luas sehingga kurang fokus ketika area yang lebih kecil yang diinginkan.
- Sumber daya sistem (processor, memory) akan terpakai sangat tinggi untuk memvisualisasikan 3D rupa bumi pada satu fail penuh sehingga dapat menyebabkan sistem *crash*.
- Ketika suatu wilayah terbagi dalam lebih dari satu fail data maka wilayah tersebut tidak bisa untuk divisualisasikan seluruhnya.

Sehingga untuk menanggulangi masalah-masalah yang timbul tersebut diperlukan beberapa metode. *Euclidean distance* merupakan metode yang dipakai untuk menghitung jarak antar dua titik. Untuk menentukan jarak terdekatnya menggunakan perbandingan antar 2 titik terdekat masing-masing poros (x dan y) menggunakan algoritma *round half down* dan *brute force closest-pair*, kedua algoritma tersebut dibandingkan untuk mengetahui mana yang lebih cepat. Sedangkan masalah ketika suatu wilayah terbagi lebih dari satu file data menggunakan metode *forward chaining* untuk mencari 4 file wilayah terdekat.

Setelah disusun kemudian data tersebut diseleksi untuk memvisualisasikan wilayah yang diinginkan saja. Untuk meningkatkan ketelitian digunakan interpolasi linier sehingga ketelitian yang sebelumnya 30 meter dapat lebih tinggi yaitu 15 meter. Karena ketelitian meningkat diharapkan gambar jauh lebih detail dan dapat memberikan informasi yang lebih akurat.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dikemukakan, maka rumusan masalah yang akan dijawab dalam penelitian ini adalah sebagai berikut:

1. Apakah algoritma *round-half-down* dan *brute force closest-pair* dapat mencari pasangan titik terdekat dengan tepat ? dan manakah yang lebih cepat ?
2. Apakah metode *euclidean distance* dan *forward chaining* dapat mendukung visualisasi 3D rupa bumi ?
3. Apakah metode interpolasi linier dapat meningkatkan ketelitian dalam visualisasi 3D rupa bumi ?

1.3 Batasan Masalah

Mengingat luasnya wilayah rupa bumi yang ada, maka dalam pembahasan visualisasi 3D rupa bumi dibatasi pada :

1. Data yang akan disediakan meliputi peta Pulau Jawa dan Bali, sedangkan objek akan difokuskan pada visualisasi 3D gunung. Namun jika ingin wilayah lain yang akan divisualisasikan cukup menambah file *GDEM ASTER* nya saja tanpa merubah aplikasi.
2. Mengingat keterbatasan sistem perangkat keras yang ada maka visualisasi maksimal dibatasi 501 x 501 titik dengan interval asli 30 meter sehingga luas

maksimal adalah $15.000 \text{ m} \times 15.000 \text{ m} = 225.000.000 \text{ m}^2$. Namun jika dikemudian hari sistem perangkat keras yang digunakan lebih canggih maka batasan visualisasi bisa ditambahkan bahkan dihilangkan.

3. Sistem koordinat yang digunakan adalah koordinat lintang dan bujur, satuan koordinat yang digunakan adalah *decimal degree*.
4. Menggunakan interpolasi linier untuk mendapatkan resolusi yang lebih tinggi dengan interval maksimal 15 meter.

1.4 Tujuan Penelitian

Tujuan dilakukannya penelitian ini adalah sebagai berikut.

1. Menerapkan algoritma *round-half-down* dan *brute force closest-pair* dalam mencari titik tengah terdekat kemudian membandingkan mana yang lebih cepat.
2. Menerapkan metode *euclidean distance* dan *forward chaining* untuk mendukung visualisasi 3D rupa bumi.
3. Menerapkan metode interpolasi linier untuk memperhalus tampilan dan membuktikan bahwa dengan interpolasi tersebut tingkat ketelitian visualisasi meningkat.

1.5 Manfaat Penelitian

Hasil dari penelitian ini diharapkan akan memberikan manfaat terhadap pengembangan data spasial antara lain :

1. Memudahkan pihak-pihak yang membutuhkan informasi mengenai visual rupa bumi mendapatkan visual secara cepat dan mudah.
2. Memberikan informasi mengenai visual rupa bumi dengan tingkat keakuratan yang tinggi.

3. Menjadi dasar pengembangan Sistem Informasi Geografis 3D yang membutuhkan data elevasi bumi.

1.6 Sistematika Penulisan

Penulisan skripsi ini tersusun dalam lima bab dengan sistematika penulisan sebagai berikut :

- BAB I Pendahuluan

Bab ini berisi pembahasan tentang latar belakang masalah, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, dan sistematika penulisan.

- BAB II Tinjauan Pustaka

Bab ini berisi pembahasan tentang visualisasi 3D rupa bumi : *state of the art, digital elevation model, model data raster, data ASTER, remapping point, euclidean distance, algoritma rounding, algoritma brute force closest-pair, forward chaining, koreksi tampilan, interpolasi linier, dan polygonal modeling.*

- BAB III Analisis dan Perancangan

Bab ini menjelaskan tentang perancangan dan implementasi visualisasi 3D rupa bumi berbasis data *GDEM ASTER 30 meter* yang meliputi kebutuhan sistem, perancangan dan implementasi algoritma, antarmuka, dan *database.*

- BAB IV Hasil dan Pembahasan

Bab ini berisi pembahasan tentang pengujian aplikasi meliputi pengujian ketepatan dan kecepatan pencarian titik tengah, pengujian ketelitian ketinggian titik tengah, pengujian ketepatan penyusunan empat ubin data terdekat, pengujian

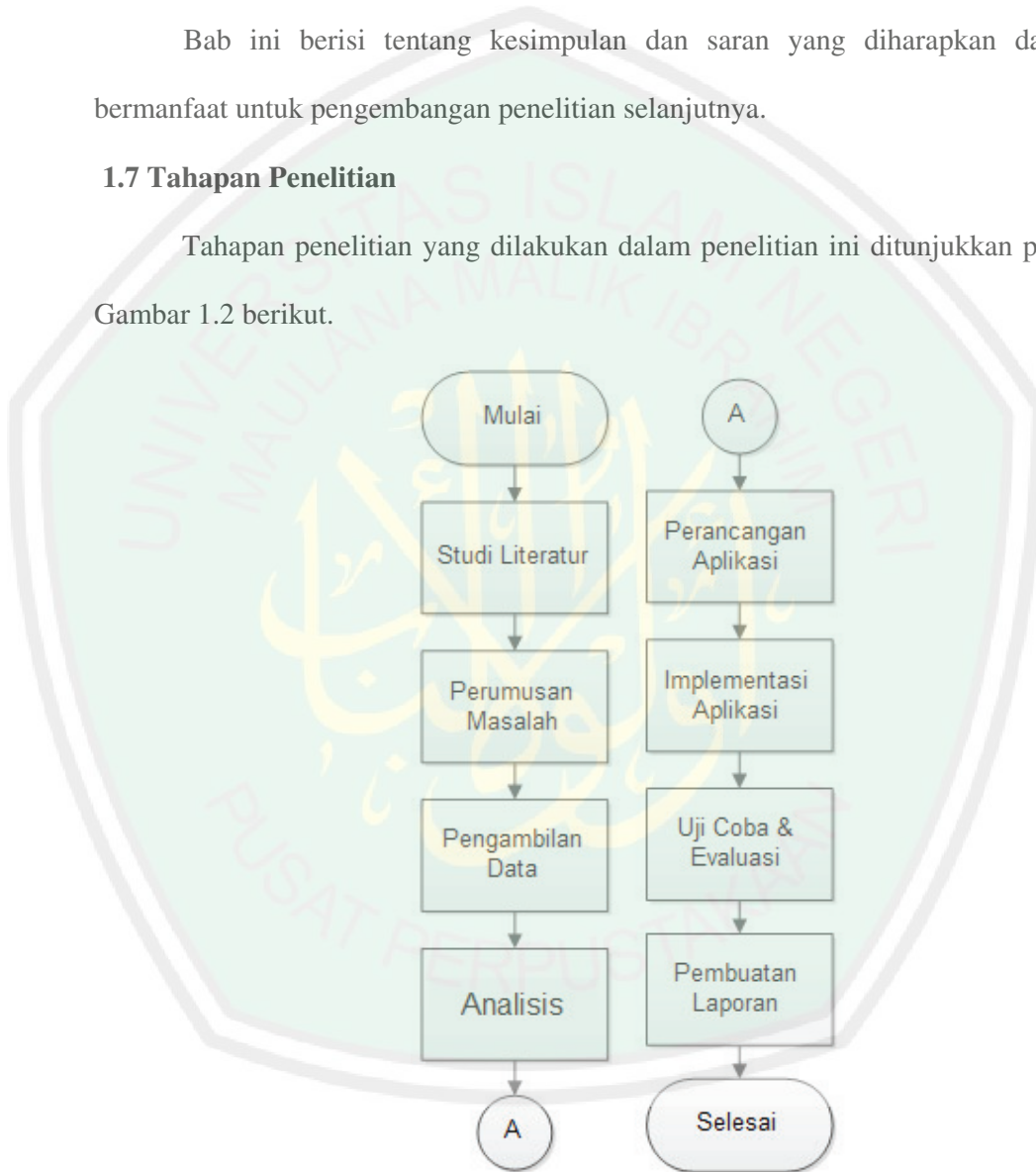
interpolasi linier, pengujian kompatibilitas pada perangkat, dan integrasi visualisasi 3D rupa bumi dengan Islam.

- BAB V Penutup

Bab ini berisi tentang kesimpulan dan saran yang diharapkan dapat bermanfaat untuk pengembangan penelitian selanjutnya.

1.7 Tahapan Penelitian

Tahapan penelitian yang dilakukan dalam penelitian ini ditunjukkan pada Gambar 1.2 berikut.



Gambar 1.2 Tahapan penelitian.

BAB II

TINJAUAN PUSTAKA

2.1 Landasan Peraturan

Penggunaan data spasial sangat penting dalam memberikan informasi untuk berbagai aspek sehingga Pemerintah Republik Indonesia menerbitkan peraturan yang menjadi payung hukum dalam pengelolaan data spasial. Data spasial adalah data hasil pengukuran, pencatatan, dan pencitraan terhadap suatu unsur keruangan yang berada di bawah, pada, atau diatas permukaan bumi dengan posisi keberadaannya mengacu pada sistem koordinat nasional. Ruang lingkup penelitian ini berada dalam wilayah data spasial maka dapat disimpulkan bahwa penelitian ini dibutuhkan, berikut beberapa landasan peraturan tentang data spasial.

- **Peraturan Presiden Republik Indonesia Nomor 85 Tahun 2007 Tentang Jaringan Data Spasial Nasional**

Data spasial memiliki peranan penting dalam memberikan informasi untuk menunjang pembangunan suatu negara. Informasi yang didapatkan diharapkan dapat menjadi salah satu sumber informasi dalam melakukan perencanaan pembangunan nasional, oleh karena pentingnya data spasial tersebut maka Pemerintah Republik Indonesia melalui Peraturan Presiden Republik Indonesia Nomor 85 Tahun 2007 membuat sebuah Peraturan Presiden yang mengelola data spasial yaitu Jaringan Data Spasial Nasional (JDSN).

- **Instruksi Presiden Republik Indonesia Nomor 6 Tahun 2012 Tentang Penyediaan, Penggunaan, Pengendalian Kualitas, Pengolahan dan Distribusi Data Satelit Penginderaan Jauh Resolusi Tinggi**

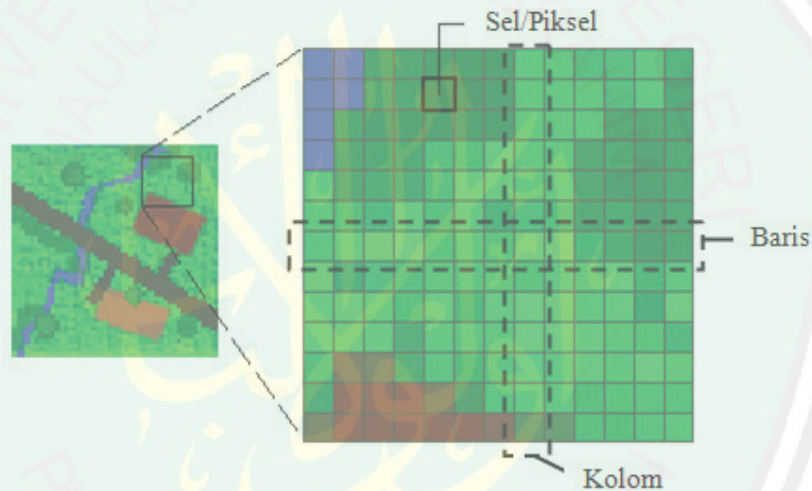
Dalam rangka pemanfaatan data satelit penginderaan jauh resolusi tinggi secara cepat untuk meningkatkan efisiensi penggunaan anggaran negara dan menghindari duplikasi alokasi anggaran dalam pengadaan data satelit penginderaan jauh resolusi tinggi, dengan ini menginstruksikan menggunakan citra tegak satelit penginderaan jauh resolusi tinggi yang disediakan oleh Badan Informasi Geospasial berdasarkan data satelit penginderaan jauh resolusi tinggi yang disediakan oleh Lembaga Penerbangan dan Antariksa Nasional. Data *GDEM ASTER* merupakan data satelit penginderaan jauh. Dimana Indonesia termasuk dalam cakupan data *GDEM ASTER*.

2.2 Digital Elevation Model

Sebuah model elevasi digital adalah model digital atau representasi 3D dari permukaan bumi yang dibuat dari data elevasi tanah. Menurut Jacobsen (2003) *Digital Elevation Model* (DEM) berdasarkan jumlah titik dengan x , y , dan z koordinat yang menggambarkan tanah gundul. DEM mungkin disusun dalam bentuk raster atau bentuk acak. Poros x dan y merupakan poros koordinat bumi, sedangkan poros z merupakan data ketinggian. DEM diperlukan untuk beberapa keperluan seperti generasi *ortoimages*, perencanaan penanggulangan banjir, pengendalian erosi, pertanian dan lain sebagainya.

2.3 Model Data Raster

Dalam bentuk paling sederhana, raster terdiri dari matiks sel / piksel yang diatur dalam baris dan kolom (atau grid), di mana setiap sel berisi nilai yang mewakili informasi seperti suhu, ketinggian, dan lain sebagainya, seperti yang ditunjukkan dalam Gambar 2.1 di bawah. Data Raster dapat diperoleh melalui foto udara, citra satelit, gambar digita, *remote sensing* atau peta yang dipindai (ESRI, 2009).

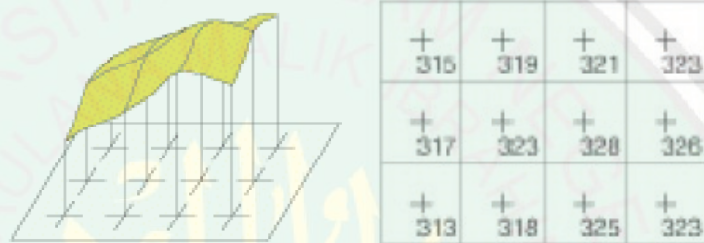


(Sumber: *Understanding Raster Data*)

Gambar 2.1 Struktur model data raster .

Pada Gambar 2.1 merupakan gambar struktur sel data raster, dalam set data raster setiap sel (yang juga dikenal sebagai piksel) memiliki nilai. Nilai-nilai sel merupakan fenomena digambarkan oleh data raster ditetapkan seperti kategori, magnitude, ketinggian, atau nilai spektral. Nilai data elevasi dapat disimpan dalam model data raster, dimana ketinggian tiap titik direpresentasikan dalam bentuk

nilai numerik. Untuk beberapa jenis data, nilai sel mewakili nilai yang diukur pada titik tengah sel. Sel tersebut memiliki posisi dan dikonversikan menjadi posisi baris dan kolom. Sel – sel tersebut disusun dalam suatu urutan yang membentuk data raster. Dalam penelitian ini tiap sel tersebut menyimpan data DEM. Contohnya adalah raster elevasi yang diilustrasikan melalui Gambar 2.2 berikut.



(Sumber: *Understanding Raster Data*)

Gambar 2.2 Contoh raster elevasi

Pada Gambar 2.2 sebuah data raster yang menyimpan nilai elevasi tanah. Posisi sel direpresentasikan dalam baris dan kolom, nilai sel merupakan data elevasi. Kumpulan sel tersebut membentuk matriks elevasi, dimana *longitude* direpresentasikan menjadi kolom matriks, dan *latitude* direpresentasikan menjadi baris matriks.

2.4 Data ASTER

Salah satu data elevasi tanah dengan model data raster adalah data *GDEM ASTER*. Untuk melakukan visualisasi rupa bumi dibutuhkan data DEM, data *GDEM ASTER* merupakan data *Global Digital Elevation Model (GDEM)* berbasis model data raster. *The Ministry of Economy, Trade, and Industry (METI) of Japan* dan *the United States National Aeronautics and Space Administration (NASA)* bersama-sama mengumumkan rilis dari *Advanced Spaceborne Thermal Emission and Reflection Radiometer (ASTER) Global Digital Elevation Model (GDEM)* versi 2 pada 17 Oktober 2011 (LP DAAC, 2011). Cakupan *GDEM ASTER* membentang dari 83° lintang utara hingga 83° lintang selatan, meliputi 99 persen dari daratan bumi (Melebihi cakupan SRTM dari 60° lintang utara hingga 56° lintang selatan). Karakteristik data *GDEM ASTER* ditunjukkan pada tabel 2.1 berikut.

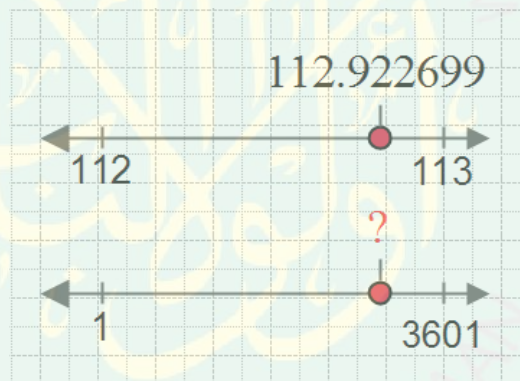
Tabel 2.1: Karakteristik *GDEM ASTER*

Ukuran ubin	3601 x 3601 (1° x 1°)
Ukuran piksel	1 <i>arc-second</i>
Sistem Koordinat Geografis	Geografis lintang dan bujur
Format <i>Output DEM</i>	GeoTIFF, signed 16-bit, vertikal dalam satuan meter Direferensikan ke the WGS84/EGM96 geoid
Nilai spesial <i>Digital Number</i>	-9999 untuk piksel kosong dan 0 untuk permukaan laut
Cakupan	Utara 83° sampai selatan 83°, 22.702 ubin

(Sumber : ASTER GDEM 2 README)

2.5 Remapping Point

Untuk mengkonversikan titik koordinat bumi lintang dan bujur menjadi titik baris dan kolom digunakan *remapping point*. Konsep *remapping* adalah mengubah nilai pada suatu *range* nilai menjadi nilai dengan *range* nilai yang lain, seperti konversi suhu °Celcius ke °Reamur atau sebaliknya. Misal sebuah titik lokasi di puncak Gunung Semeru memiliki koordinat lintang -8.111658° dan bujur 112.922699° . Pada file data ASTER satu file memiliki panjang 1° , dimana 1° terdapat 3601 titik artinya antara 112° sampai 113° memiliki 3601 titik. Untuk mengetahui posisi titik pada matriks dari bujur 112.922699° dapat dilakukan dengan cara *remapping point* seperti pada Gambar 2.3 berikut.



Gambar 2.3 Ilustrasi *remapping point*.

Mekanisme *remapping point* (disadur dari <http://arduino.cc/en/Reference/map>) ditunjukkan pada Rumus 2.1 berikut.

$$titikOut = lowOut + (valueIn - lowIn) * \frac{highOut - lowOut}{highIn - lowIn} \quad (2.1)$$

dimana :

titikOut = titik hasil map

lowOut = titik batas bawah hasil konversi yaitu 1

highOut = titik batas atas hasil konversi yaitu 3601

lowIn = titik batas bawah titik asal dalam contoh ini adalah 112

highIn = titik batas atas titik asal dalam contoh ini adalah 113

valueIn = titik asal yang akan dikonversi

sehingga :

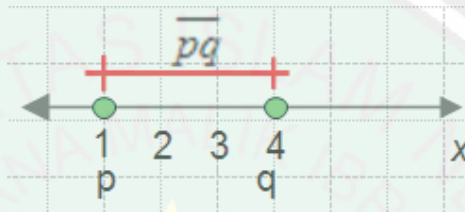
$$titikOut = 1 + (112.922699 - 112) * \frac{3601 - 1}{113 - 112}$$

$$titikOut = 1 + 0.922699 * \frac{3600}{1}$$

$$titikOut = 3322.7164$$

2.6 Euclidean Distance

Euclidean distance adalah sebuah metode untuk mengukur panjang antara dua titik, *euclidean distance* antara p dan q adalah panjang ruas garis yang menghubungkan mereka (\overline{pq}) (Wikipedia). Perhitungan *euclidean distance* satu dimensi diilustrasikan pada Gambar 2.4 berikut.



Gambar 2.4 Ilustrasi *euclidean distance* satu dimensi

Perhitungan jarak titik menghitung jarak tiap poros x dan y satu persatu maka menggunakan rumus *euclidean distance* satu dimensi yang ditunjukkan pada Rumus 2.2 berikut.

$$\begin{aligned} dx &= |x1 - x2| \\ dy &= |y1 - y2| \end{aligned} \quad (2.2)$$

dimana: $dx = distance$ / jarak $x1$ ke $x2$

$dy = distance$ / jarak $y1$ ke $y2$

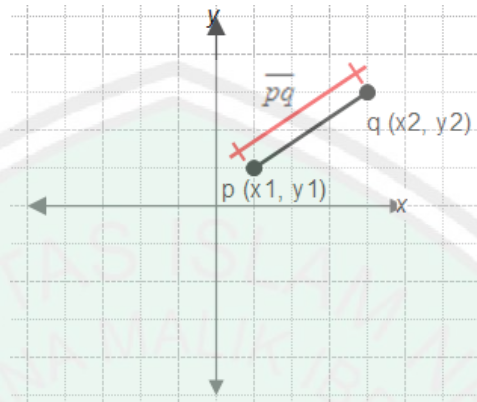
$x1 =$ titik koordinat $x1$ pada poros x

$x2 =$ titik koordinat $x2$ pada poros x

$y1 =$ titik koordinat $y1$ pada poros y

$y2 =$ titik koordinat $y2$ pada poros y

Perhitungan *euclidean distance* dua dimensi diilustrasikan pada Gambar 2.5 berikut.



Gambar 2.5 Ilustrasi *euclidian distance* dua dimensi.

Untuk perhitungan jarak titik dua dimensi dimana menghitung kedua titik pada poros x dan y secara bersamaan maka menggunakan rumus *euclidian distance* yang ditunjukkan pada Rumus 2.3 berikut.

$$d(p, q) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \quad (2.3)$$

dimana : $d(p, q) = \text{distance} / \text{jarak}$

x_1 = titik koordinat **p** pada poros x

x_2 = titik koordinat **q** pada poros x

y_1 = titik koordinat **p** pada poros y

y_2 = titik koordinat **q** pada poros y

2.7 Algoritma Rounding

Menurut Maxfield (2006) bahwa fakta paling mendasar terkait dengan pembulatan adalah mengubah beberapa kuantitas dari presisi yang lebih tinggi ke presisi yang lebih rendah. Misal pengukuran panjang pensil menggunakan jangka sorong mendapat hasil 10,967 cm bisa dibulatkan menjadi 11 cm. Ada beberapa jenis algoritma pembulatan ini, dapat dipilih yang sesuai untuk meminimalkan efek kehilangan presisi. Algoritma tersebut antara lain *round-up*, *round-down*, *round-toward-nearest*, *arithmetic rounding*, *round-half-up*, *round-half-down*, *round-half-even*, *round-half-odd*, *round-toward-zero*, *round-away-from-zero*, *round-ceiling*, *round-floor*, *truncation*, *round-alternate*, dan *round-random*. Dalam penelitian ini hanya menggunakan 3 algoritma pembulatan yaitu *round-ceiling*, *round-floor*, dan *round-half-down*, yang dapat dijelaskan sebagai berikut.

- *Round-ceiling* → Algoritma ini mengacu pada pembulatan ke arah positif tak terhingga. Dalam kasus angka positif, hasilnya akan tetap tidak berubah jika angka yang akan dibuang adalah nol, jika tidak maka akan dibulatkan. Contoh, 3.0 akan dibulatkan menjadi 3, tapi 3.1, 3.2, 3.3, 3.4, 3.5, 3.6, 3.7, 3.8, dan 3.9 semua akan dibulatkan menjadi 4. Sebagai perbandingan, dalam kasus angka negatif, angka yang tidak diinginkan dibuang begitu saja. Misalnya, -3.0, -3.1, -3.2, -3.3, -3.4, -3.5, -3.6, -3.7, -3.8, -3.9 dan semua akan dibulatkan ke -3.

Dalam MATLAB algoritma ini diimplementasikan menggunakan fungsi *ceil*.

- *Round-floor* → Lawan dari *round-ceiling*, algoritma ini mengacu pada pembulatan menuju tak terhingga negatif. Dalam kasus angka positif, angka yang tidak diinginkan dibuang begitu saja. Misalnya, 3.0, 3.1, 3.2, 3.3, 3.4, 3.5, 3.6, 3.7, 3.8, dan 3.9 semua akan dibulatkan menjadi 3. Dalam kasus angka negatif, hasilnya akan tetap tidak berubah jika angka yang akan dibuang adalah nol, jika tidak maka akan dibulatkan ke bawah. Misalnya, -3.0 akan dibulatkan ke -3, tapi -3.1, -3.2, -3.3, -3.4, -3.5, -3.6, -3.7, -3.8, -3.9 dan semua akan dibulatkan ke -4. Dalam MATLAB algoritma ini diimplementasikan menggunakan fungsi *floor*.
- *Round-half-down* → Algoritma ini mengacu pada pembulatan ke nilai terdekat, sedang nilai tengah akan dibulatkan ke bawah. Misal 3.0, 3.1, 3.2, 3.3, 3.4 akan dibulatkan ke 3, sedangkan 3.6, 3.7, 3.8, 3.9, 4.0 akan dibulatkan ke 4, dan 3.5 dimana nilai tersebut adalah nilai tengah maka karena *half-down* maka pembulatannya mulai nilai tengah kebawah yaitu 3. Algoritma ini nantinya akan digunakan untuk mencari titik terdekat diantara titik bilangan bulat pada baris dan kolom matriks. Untuk membandingkan jarak titik yang diperoleh dari hasil *remapping point* dengan titik terdekat pada batas atas dan batas bawah titik tersebut digunakan *euclidean distance* satu dimensi.

2.8 Algoritma Brute Force Closest-Pair

Menurut Levitin (2012) algoritma *brute force closest-pair* adalah menghitung jarak antara setiap pasang titik yang berbeda dan menemukan sepasang dengan jarak terkecil, dan tidak menghitung pasangan yang sama dua kali, untuk menghindarinya hanya pasangan titik (p_i, p_j) yang $i < j$. Diasumsikan bahwa titik-titik tersebut ditetapkan secara standar oleh titik (x, y) pada koordinat kartesius dan bahwa jarak antara dua titik $p_i(x_i, y_i)$ dan $p_j(x_j, y_j)$ adalah *euclidean distance* dua dimensi standar seperti pada Rumus 2.4 dibawah ini.

$$d(p_i, p_j) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \quad (2.4)$$

Pseudocode dari algoritma *brute force closest-pair* adalah ditunjukkan pada Gambar 2.6 berikut.

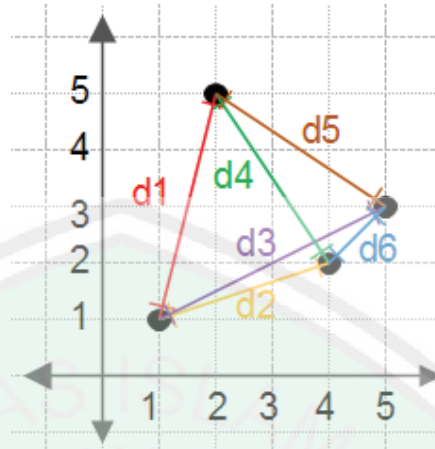
```

algorithm BruteForceClosestPair(P)
//Finds distance between two closest points
  in the plane by brute force
//Input: A list P of n (n ≥ 2) points p1(x1, y1),
  . . . , pn(xn, yn)
//Output: The distance between the closest pair of points

d ← ∞
for i ← 1 to n-1 do
  for j ← i + 1 to n do
    d ← min(d, sqrt((xi - xj)2 + (yi - yj)2))
return d

```

Gambar 2.6 *Pseudocode* algoritma *brute force closest-pair*.



Gambar 2.7 Ilustrasi *brute force closest-pair*.

Seperti pada Gambar 2.7 misal terdapat 4 buah titik yaitu (1,1), (2,5), (4,2), dan (5,3), masing-masing titik tersebut dihitung *euclidean distance* sebagai berikut.

$$(1,1) \rightarrow (2,5) = d1,$$

$$(1,1) \rightarrow (4,2) = d2,$$

$$(1,1) \rightarrow (5,3) = d3,$$

$$(2,5) \rightarrow (4,2) = d4,$$

$$(2,5) \rightarrow (5,3) = d5,$$

$$(4,2) \rightarrow (5,3) = d6.$$

Masing-masing dihitung *euclidean distance* pada tiap titik anggota yang didefinisikan sebanyak satu kali karena $(1,1) \rightarrow (2,5)$ atau $(2,5) \rightarrow (1,1)$ adalah titik yang sama dengan jarak yang sama. Jarak tersebut dibandingkan satu sama lain dan diseleksi mana yang paling kecil. Pasangan titik dengan jarak terkecil adalah hasil dari pencarian menggunakan algoritma ini.

2.9 Forward Chaining

Menurut Russell (2003) *forward chaining* adalah salah satu metode penalaran saat menggunakan *inference rule* dan dapat dijelaskan secara logis sebagai aplikasi modus ponens. *Forward chaining* dimulai dengan data yang tersedia dan menggunakan aturan inferensi untuk mengekstrak lebih banyak data sampai tujuan tercapai. Sebuah mesin inferensi yang menggunakan *forward chaining* diterapkan untuk orde pertama klausa dan dapat diimplementasikan secara efisien. Klausa seperti situasi → respon (jika situasi a maka respon b) berguna untuk sistem yang membuat kesimpulan untuk menanggapi informasi yang diberikan. Inferensi dimulai dari keadaan awal yaitu informasi atau fakta yang ada kemudian dicoba untuk dicocokkan dengan basis pengetahuan kemudian diperoleh tujuan yang diharapkan.

Forward chaining dapat diterapkan secara luas dalam berbagai bidang, salah satu contoh adalah sistem pakar untuk mendiagnosa suatu penyakit berdasarkan gejala yang timbul. Basis pengetahuan dapat diperoleh dari para ahli dalam hal ini dokter, petugas kesehatan, atau pihak-pihak yang memiliki pengetahuan tentang suatu penyakit dan gejala-gejala yang ditimbulkannya. Gejala penyakit digunakan sebagai input kemudian data tersebut diekstraksi menggunakan mesin inferensi berdasarkan basis pengetahuan yang diperoleh sebelumnya, hasil keluaran merupakan beberapa kemungkinan penyakit yang timbul dengan gejala yang diinputkan. Sedemikian luas penggunaan *forward chaining* dalam mengekstraksi data sehingga memungkinkan untuk digunakan dalam penelitian ini.

2.10 Koreksi Tampilan

- **Penyamaan skala poros x , y , dan z**

Menurut Crys dian (2009) ketika sebuah data *DEM* divisualisasikan terdapat masalah yaitu satuan pada poros x dan y berbeda dengan skala pada poros z . Korelasi tidak seimbang antara koordinat horizontal yang direpresentasikan oleh x - y ke vertikal koordinat yang direpresentasikan oleh z . Tampak bahwa nilai vertikal mendominasi nilai horizontal karena sumbu poros tersebut tidak sama satuannya. Pada data *GDEM ASTER* interval tiap titik tanpa interpolasi pada poros x dan y adalah 30 meter, sedangkan interval tiap titik pada poros z adalah 1 meter. Matriks data tersebut dikonversikan baris menjadi poros x , kolom menjadi poros y , isi tiap sel matriks menjadi poros z .

Untuk melihat visualisasi secara 3D sudut pandang dirumuskan seperti yang terlihat pada Rumus 2.5 berikut.

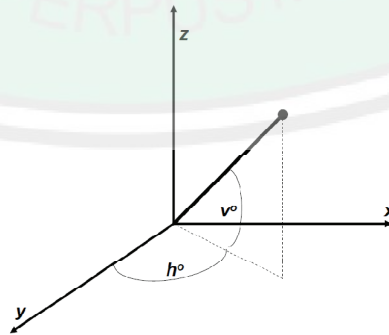
$$V = (h, v) \quad (2.5)$$

dimana : V = sudut pandang dari visualisasi 3D

h = ketinggian horizontal dari poros y

v = ketinggian vertikal dari bidang $x - y$

Berikut ilustrasi visualisasi 3D ditunjukkan pada Gambar 2.8 berikut.



(Sumber : 3D Visualization of Spatial Objects using Elevation Model)

Gambar 2.8 Sudut pandang 3D.

Untuk mengatasi masalah di atas, mekanisme untuk memaksa grid yang sama dibentuk untuk skala kedua sumbu vertikal dan horizontal dikembangkan dengan menggunakan pendekatan *thresholding*. Untuk menyamakan skala interval pada poros x dan y dengan poros z maka diambil poros x atau y terpanjang, Panjang dari poros tersebut dikalikan dengan interval kemudian dijadikan batas atas dari poros z (z_{max}) sedangkan batas bawah dari poros z (z_{min}) adalah 0 karena nilai 0 adalah permukaan laut. Setelah itu tampilan dikunci dengan mengunci rasio tampilan 3D menggunakan “*axis vis3d*” di MATLAB agar tampilan tidak berubah-ubah ketika rotasi objek 3D. Kemudian agar poros z yang kosong tidak perlu ditampilkan maka dipotong lagi z_{min} = nilai matriks minimal dan z_{max} = nilai matriks maksimal jika $z_{max} > z_{min}$.

Formula dari penyamaan satuan interval ditunjukkan pada Rumus 2.6 berikut.

$$z_{max} = \begin{cases} interval * (x_{max} - x_{min}) - 1 & \leftrightarrow x_{max} \geq y_{max} \\ interval * (y_{max} - y_{min}) - 1 & \leftrightarrow x_{max} < y_{max} \end{cases} \quad (2.6)$$

dimana:

x_{min} = 1 (index pertama matriks pada MATLAB adalah 1).

y_{min} = 1 (index pertama matriks pada MATLAB adalah 1).

x_{max} = panjang baris matriks.

y_{max} = panjang kolom matriks.

-1 = karena visualisasi dimulai pada ketinggian 0 pada poros z .

- **Penyesuaian Color Map**

Color Map adalah fungsi yang memetakan / mengubah warna dari satu (sumber) gambar untuk warna gambar lain (target) (Wikipedia) . Sebuah *color map* dapat disebut sebagai algoritma yang menghasilkan fungsi pemetaan atau algoritma yang mengubah warna gambar.

Pada penelitian ini untuk memvisualisasikan 3D rupa bumi menggunakan *colormap* MATLAB. Berikut adalah macam – macam *colormap* yang ada pada MATLAB ditunjukkan pada Gambar 2.9.



(Sumber: MATLAB)

Gambar 2.9 *Colormap* MATLAB.

Colormap yang digunakan adalah *colormap Jet* seperti yang tampak ada gambar diatas. *Colormap* tersebut memiliki *range* warna mulai dari biru tua hingga merah tua. *Colormap* memberikan warna pada visualisasi dengan data elevasi terendah dengan warna biru tua sampai dengan tertinggi dengan merah tua. Masalah yang muncul adalah ketika ketinggian lautan dengan tepi daratan berbeda tipis dengan beberapa meter saja. Sehingga warna antara lautan dimana

memiliki ketinggian 0 meter dengan daratan berkisar antara biru tua dengan biru yang bagi sebagian orang secara relatif melihatnya tidak jelas antara batas tersebut. Oleh karena itu diperlukan penyesuaian *colormap* agar batas antara daratan dan lautan terlihat jelas. Penyesuaian tersebut dilakukan dengan cara mengganti nilai terendah *colormap jet* yang semula berwarna biru tua dengan warna putih jika poros $z \leq 0$ artinya jika ketinggian terendah adalah lautan. Sehingga jika poros z ada yang bernilai ≤ 0 maka *colormap* akan berubah seperti yang diilustrasikan pada Gambar 2.10 berikut.



(Sumber: MATLAB)

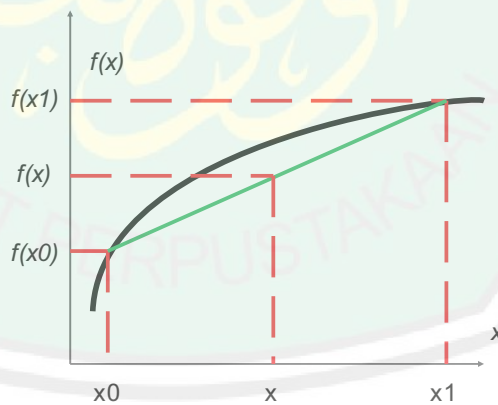
Gambar 2.10 Ilustrasi perubahan *colormap*.

2.11 Interpolasi Linier

Menurut Pujiyanta (2007) interpolasi digunakan untuk memperkirakan nilai (tengah) di antara titik-titik dari satu set nilai yang sudah diketahui. Dalam arti yang lebih luas, interpolasi merupakan upaya mendefinisikan suatu fungsi dekatkan suatu fungsi analitik yang tidak diketahui atau pengganti fungsi rumit yang tak mungkin diperoleh persamaan analitiknya. Dalam penelitian ini interpolasi digunakan untuk memperhalus kontur tanah karena data yang sebelumnya memiliki interval antar titik 30 meter menjadi 15 meter dengan penambahan titik elevasi ditengah-tengah titik elevasi yang ada sehingga intervalnya menjadi semakin pendek mengakibatkan tingkat kepresisiannya meningkat. Misalnya menempatkan objek berupa pohon pada area visualisasi, untuk mencari titik

terdekat posisi pohon akan lebih presisi pada interval 15 meter daripada 30 meter. Interpolasi dapat ditingkatkan lagi menjadi 7.5 meter, 3.75 meter dan seterusnya namun dengan konsekuensi data yang akan diproses semakin besar dan memerlukan sumber daya komputer yang lebih besar sehingga mengakibatkan waktu untuk memproses data lebih lama, oleh karena itu pada penelitian ini dibatasi interpolasi hanya sampai 15 meter mengingat sumber daya komputer yang terbatas.

Metode interpolasi ada beberapa macam yaitu interpolasi linier, interpolasi kuadrat, interpolasi polinomial, interpolasi rasional, dan lain sebagainya. Pada penelitian ini digunakan salah satu metode interpolasi yaitu interpolasi linier. Interpolasi linier untuk meramal kurva nilai $y = f(\bar{x})$ dapat dilakukan dengan menganggap bahwa y_k dan y_{k+1} dihubungkan oleh suatu garis lurus, seperti yang terlihat pada Gambar 2.11 berikut.



(Sumber : Komputasi Numerik dengan Matlab)

Gambar 2.11 Interpolasi linier untuk meramal kurva nilai $y = f(\bar{x})$.

Secara *geometric* peramalan grafik gambar 2.9 di atas yang menghubungkan titik (x_k, y_k) dengan titik (x_{k+1}, y_{k+1}) dapat dinyatakan oleh Rumus 2.7 berikut.

$$\bar{y} = y_k + \frac{y_{k+1} - y_k}{x_{k+1} - x_k} (\bar{x} - x_k) \quad (2.7)$$

Dengan demikian hasil yang diperoleh akan benar (eksak) memang merupakan fungsi linier. Persamaan tersebut dapat disederhanakan dalam interpolasi linier matriks menggunakan nilai rata-rata karena \bar{y} yang dicari nilainya merupakan nilai pada titik tengah matriks yang berada diantara titik-titik matriks.

$$A = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \rightarrow B = \begin{bmatrix} a & \frac{a+b}{2} & b \\ \frac{a+c}{2} & \frac{a+b+c+d}{4} & \frac{b+d}{2} \\ c & \frac{c+d}{2} & d \end{bmatrix} \quad (2.8)$$

Perhatikan Rumus 2.8 diatas misal sebuah matriks A berordo 2x2 akan diinterpolasi dengan menyisipkan 1 nilai diantara titik-titik yang ada maka persamaannya ditunjukkan pada Rumus 2.9 berikut.

$$B = A_{(m*2-1) \times (n*2-1)} \quad (2.9)$$

Matriks B menjadi berordo 3x3, jika matriks B diinterpolasi lagi bisa menjadi berordo 5x5 dan seterusnya. Untuk membuktikan bahwa nilai rata-rata merupakan interpolasi linier maka bisa dibuktikan dengan perhitungan yang ditunjukkan pada Gambar 2.12 berikut.

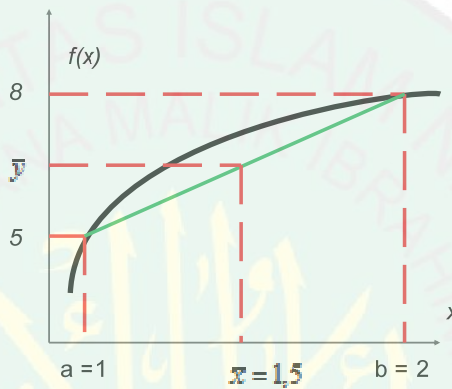
Misal

titik a = 5, titik pertama

titik b = 8, titik kedua

titik \bar{x} adalah titik tengah antara a dan b

berapakah nilai \bar{y} ?



Gambar 2.12 Contoh interpolasi linier.

$$\begin{aligned}\bar{y} &= y_k + \frac{y_{k+1} - y_k}{x_{k+1} - x_k} (\bar{x} - x_k) \\ \bar{y} &= 5 + \frac{8-5}{2-1} (1,5 - 1) \\ \bar{y} &= 5 + \frac{3}{1} (0,5) \\ \bar{y} &= 5 + 1,5 \\ \bar{y} &= 6,5\end{aligned}$$

Bila menggunakan nilai rata-rata adalah sebagai berikut.

$$\begin{aligned}\bar{y} &= \frac{a+b}{2} \\ \bar{y} &= \frac{5+8}{2} \\ \bar{y} &= 6,5\end{aligned}$$

Dengan perhitungan tersebut dapat disimpulkan bahwa hasilnya sama, sehingga dalam interpolasi linier matriks menggunakan nilai rata – rata untuk penyederhanaan proses perhitungannya.

2.12 Polygonal Modeling

Menurut Crys dian (2011) pendekatan *polygonal modeling* adalah metode sederhana namun efektif untuk membangun model kota menggunakan visualisasi 3D. Dengan menggunakan pendekatan ini berbagai fitur kota seperti bangunan, vegetasi, dan jalan dapat dikembangkan dengan cara yang efisien berdasarkan bentuk tepi objek. Informasi spasial yang kompleks biasanya meliputi perkembangan model 3D kota dapat dikelola secara sederhana. Penelitian ini akan mengimplementasikan *polygonal modeling* yang telah diteliti sebelumnya, objek yang akan divisualisasikan adalah pohon. Tujuan implementasi visualisasi pohon ini adalah membuktikan bagaimana *interpolasi* memberikan dampak visualisasi menjadi semakin detail dan akurat. Perbedaan akan terlihat ketika objek pohon divisualisasikan sebelum dan setelah interpolasi. Objek pohon ditunjukkan pada Gambar 2.13 berikut.



(Sumber: *Development of 3D City Model using Polygonal Modeling Approach*)

Gambar 2.13 Objek pohon.

BAB IV

HASIL DAN PEMBAHASAN

Pada bab ini membahas tentang pengujian dan pembahasan terhadap hasil pengujian aplikasi visualisasi.

4.1 Langkah – langkah pengujian

Langkah – langkah pengujian menjelaskan bagaimana aplikasi diuji. Terdapat beberapa parameter yang akan diuji, antara lain sebagai berikut:

1. Langkah pengujian ketepatan dan kecepatan pencarian titik tengah

Pengujian ini membandingkan implementasi antara algoritma *round-half-down* dan algoritma *brute force closest-pair* untuk mencari titik *latitude-longitude* dengan titik terdekat baris-kolom matriks yang menjadi titik tengah wilayah visualisasi. Parameter yang dibandingkan adalah ketepatan dan kecepatan pencarian titik terdekat *latitude-longitude* dengan titik baris-kolom matriks 3601 x 3601. Pengujian ini menggunakan komputer dengan spesifikasi sebagai berikut.

Prosesor	: AMD Phenom x2 3.2 GHz
Memory	: DDR3 3GB
VGA	: Ati Radeon 4920
OS	: Windows 7

Pengujian dilakukan dengan melakukan visualisasi 5 buah titik lokasi dengan tiap titik menggunakan dua algoritma yaitu *round-half-down (rhd)* dan *brute force closest-pair (bf)*. Nilai baris-kolom asli akan dibandingkan dengan nilai baris-kolom terdekat pada masing-masing algoritma diuji apakah tepat mencari titik terdekatnya. Setelah itu diuji berapa lama waktu yang

dihabiskan untuk proses pencarian titik terdekat tersebut menggunakan kedua algoritma tersebut. Karena terdapat dua metode input yaitu titik tengah dan titik tepi yang sebenarnya sama hanya saja pada titik tepi harus dinormalisasikan dahulu maka digunakan titik tengah untuk mewakili pengujian ini.

2. Langkah pengujian ketelitian ketinggian titik tengah

Pengujian ini membandingkan data ketinggian pada titik tengah wilayah visualisasi. Titik tengah ini memiliki nilai ketinggian yang akan dibandingkan dengan ketinggian GPS Garmin 80 Mil dan Google Earth pada titik yang sama, sedangkan GPS HP Android hanya digunakan sebagai data referensi saja untuk membuktikan bahwa pada lokasi yang sama sekalipun hasil pengukuran ketinggian menggunakan GPS memiliki hasil yang berbeda tergantung kualitas dan tingkat ketelitian masing-masing GPS. Setelah data tersebut diperoleh maka diperoleh nilai ketinggian dari masing-masing sumber tersebut kemudian dibandingkan satu sama lain dan persentase perbedaan ketinggian absolut tersebut dihitung kemudian dirata-rata, akan terlihat berapa persen tingkat ketelitian ketinggian. Dengan adanya pengujian ini maka dapat diketahui berapa persenkah *error* ketelitian aplikasi visualisasi yang dibuat.

3. Langkah pengujian ketepatan penyusunan empat ubin data terdekat

Pengujian ketepatan penyusunan empat ubin data terdekat menguji apakah penggunaan metode *forward chaining* berhasil untuk menyusun empat ubin terdekat tersebut kemudian divisualisasikan. Cara mengujinya dengan memvisualisasikan titik tepi dimana masing-masing titik tersebut berada pada ubin data yang berbeda.

4. Langkah pengujian interpolasi linier

Untuk menguji apakah interpolasi linier memberikan dampak meningkatnya ketelitian visualisasi maka dilakukan pengujian visualisasi pohon tanpa interpolasi dan dengan interpolasi, akan terlihat bagaimana ketelitian meningkat dengan interpolasi.

5. Langkah pengujian kompatibilitas pada perangkat

Pengujian kompatibilitas pada perangkat menguji apakah aplikasi visualisasi dapat diakses oleh banyak perangkat yang berbeda. Karena keterbatasan perangkat yang ada maka pengujian ini hanya menguji pada perangkat desktop Windows 7, Linux Mint, Tablet Android, dan Smartphone Android.

4.2 Hasil Pengujian

Setelah langkah – langkah pengujian dilakukan maka didapatkan data hasil pengujian. Data tersebut disajikan dalam bentuk tabel dan grafik. Hasil pengujian antara lain sebagai berikut:

4.2.1 Pengujian Ketepatan dan Kecepatan Pencarian Titik Tengah

Hasil pengujian ditunjukkan pada tabel 4.1 dan 4.2 berikut.

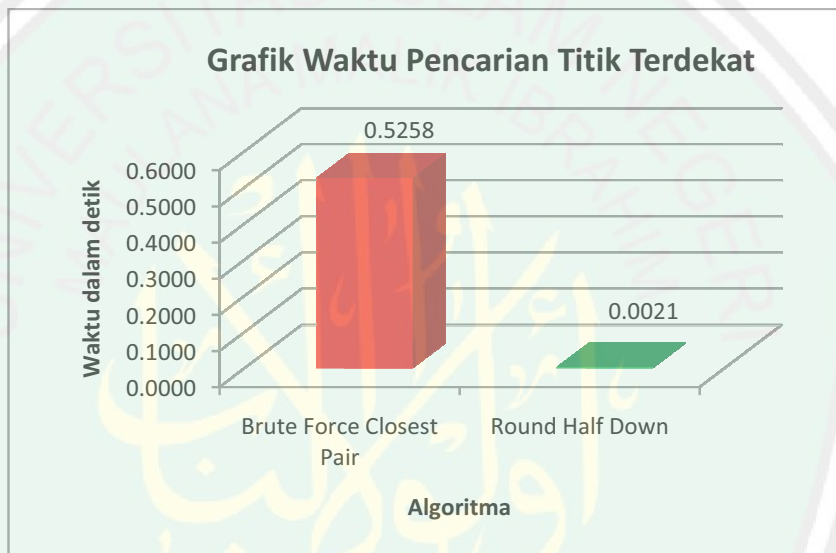
Tabel 4.1: Pengujian algoritma *brute force closest-pair*

No	Lokasi	Latitude	Longitude	Radius	Baris	Baris terdekat	Kolom	Kolom terdekat	Waktu Pencarian Brute Force Closest Pair
1	Gunung Arjuna	-7.765040	112.5897	150	2755.144	2755	2123.916	2124	0.6721
2	Gunung Pandoman	-7.904006	112.4965	35	3255.422	3255	1788.393	1788	0.4939
3	Pulau Sempu	-8.438431	112.6903	150	1579.351	1579	2486.080	2486	0.5186
4	Tambang Grasberg	-4.056800	137.1130	50	205.480	205	407.824	408	0.4718
5	Anak Krakatau	-6.101361	105.4245	40	365.899	366	1529.192	1529	0.4727
Waktu rata - rata dalam detik (s)									
									0.5258

Tabel 4.2: Pengujian ketepatan algoritma *round-half-down*.

No	Lokasi	Latitude	Longitude	Radius	Baris	Baris terdekat	Kolom	Kolom terdekat	Waktu Pencarian Round Half Down
1	Gunung Arjuna	-7.765040	112.5897	150	2755.144	2755	2123.916	2124	0.0019
2	Gunung Pandoman	-7.904006	112.4965	35	3255.422	3255	1788.393	1788	0.0020
3	Pulau Sempu	-8.438431	112.6903	150	1579.351	1579	2486.080	2486	0.0022
4	Tambang Grasberg	-4.056800	137.1130	50	205.480	205	407.824	408	0.0019
5	Anak Krakatau	-6.101361	105.4245	40	365.899	366	1529.192	1529	0.0024
Waktu rata - rata dalam detik (s)									
									0.0021

Dari tabel 4.1 dan 4.2 diatas dapat diketahui bahwa baris-kolom terdekat hasil pencarian kedua algoritma menghasilkan nilai yang tepat sama. Artinya menggunakan kedua algoritma tersebut akan menghasilkan gambar visual yang sama karena titik tengahnya sama. Sedangkan yang berbeda adalah waktu pencarian titik tengah terdekat tersebut. Perbedaan waktu tersebut dapat dilihat lebih jelas dalam Gambar 4.1 berikut.



Gambar 4.1 Grafik waktu pencarian titik terdekat.

Seperti yang terlihat pada gambar 4.1 Grafik waktu pencarian titik terdekat diatas bahwa bila dibandingkan maka perbandingan kedua algoritma tersebut adalah $\frac{\text{waktu BF}}{\text{waktu RHD}} = \frac{0.5258}{0.0021} = 250.38$, dapat disimpulkan bahwa algoritma *round-half-down* lebih cepat **250 x** daripada algoritma *brute force closest-pair*. Tentu saja waktu tersebut tergantung dari sumber daya komputer, semakin cepat komputer maka akan semakin cepat pula waktu pencarian titik, begitu pula sebaliknya.

4.2.2 Pengujian Ketelitian Ketinggian Titik Tengah

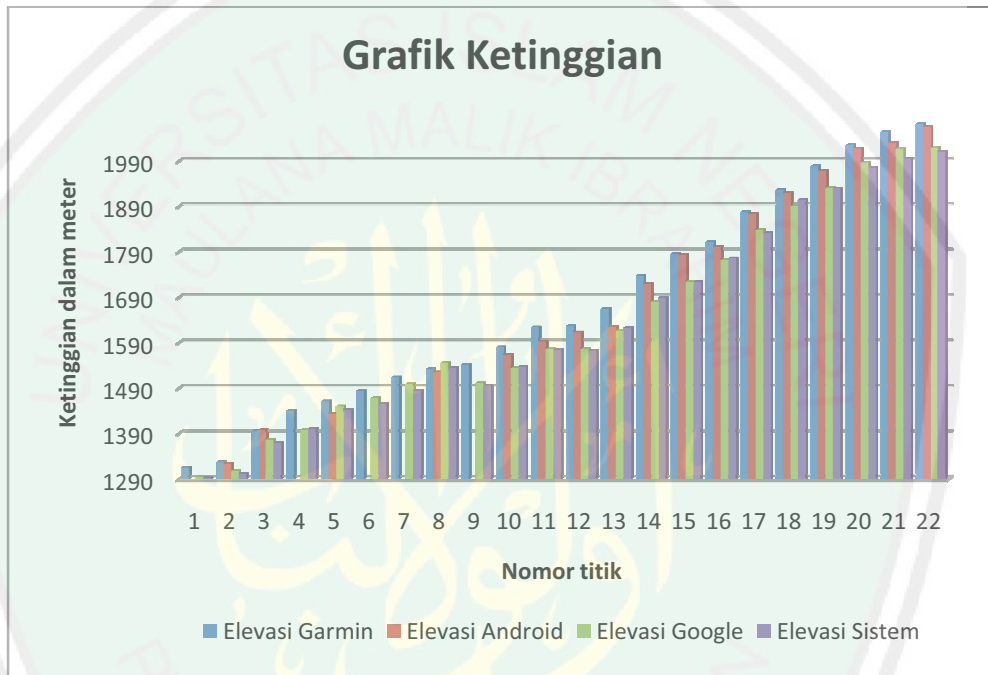
Hasil pengujian ditunjukkan pada tabel 4.3 berikut.



Tabel 4.3: Pengujian ketelitian ketinggian titik tengah

No	Waypoint GPS	Latitude GPS	Longitude GPS	Elevasi Garmin	Elevasi Android	Elevasi Google	Elevasi Aplikasi	Perbedaan Garmin / Google	Perbedaan Aplikasi / Google
1	56	-7.88725	112.49844	1315		1294	1293	1.62%	0.08%
2	57	-7.88785	112.49760	1327	1323	1308	1302	1.45%	0.46%
3	58	-7.89038	112.49552	1396	1398	1377	1369	1.38%	0.58%
4	81	-7.89009	112.49411	1440		1399	1401	2.93%	0.14%
5	59	-7.89229	112.49470	1461	1433	1450	1442	0.76%	0.55%
6	80	-7.89120	112.49275	1484		1468	1455	1.09%	0.89%
7	79	-7.89227	112.49263	1514		1500	1484	0.93%	1.07%
8	60	-7.89452	112.49351	1532	1525	1546	1534	0.91%	0.78%
9	78	-7.89227	112.49225	1542		1502	1496	2.66%	0.40%
10	77	-7.89318	112.49200	1580	1563	1534	1537	3.00%	0.20%
11	76	-7.89412	112.49216	1624	1593	1576	1575	3.05%	0.06%
12	61	-7.89436	112.49224	1627	1613	1576	1572	3.24%	0.25%
13	62	-7.89586	112.46213	1665	1625	1617	1623	2.97%	0.37%
14	63	-7.89721	112.46220	1738	1720	1681	1689	3.39%	0.48%
15	64	-7.89827	112.49248	1786	1784	1725	1724	3.54%	0.06%
16	65	-7.89992	112.49318	1812	1802	1772	1776	2.26%	0.23%
17	66	-7.90113	112.49415	1878	1873	1838	1832	2.18%	0.33%
18	67	-7.90184	112.49465	1927	1920	1891	1905	1.90%	0.74%
19	68	-7.90231	112.49515	1979	1968	1931	1930	2.49%	0.05%
20	69	-7.90291	112.49586	2025	2018	1987	1976	1.91%	0.55%
21	70	-7.90262	112.49737	2054	2031	2017	1995	1.83%	1.09%
22	71	-7.90263	112.49781	2072	2065	2020	2010	2.57%	0.50%
Perbedaan rata - rata								2.18%	0.45%

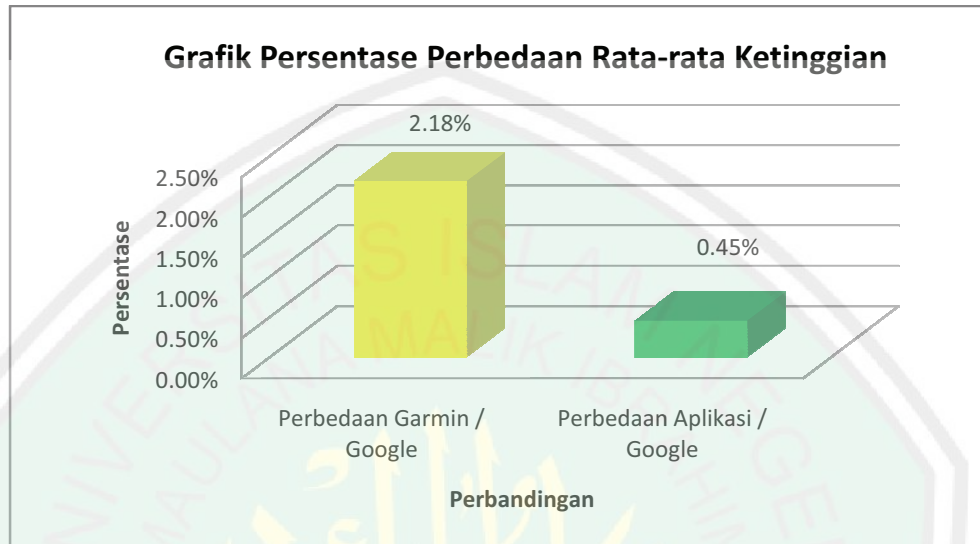
Dari data pada tabel 4.3 diatas didapatkan grafik ketinggian seperti yang terlihat pada Gambar 4.2, terlihat bahwa perbedaan ketinggian antara elevasi tanah pada Google Earth tidak terlalu berbeda jauh dengan aplikasi visualisasi yang dibuat, sedangkan bila dibandingkan dengan elevasi GPS Garmin maupun Android memiliki beda ketinggian yang cukup signifikan.



Gambar 4.2 Grafik ketinggian.

Dari grafik ketinggian pada Gambar 4.2 diatas akan dibandingkan perbedaan absolut dari ketinggian masing-masing sumber data. Rata-rata persentase perbedaan elevasi GPS Garmin dibanding Google Earth adalah 2,18 %, sedangkan perbedaan elevasi aplikasi visualisasi dibandingkan dengan Google Earth adalah 0,45 %. Hal ini membuktikan bahwa aplikasi yang dibuat lebih

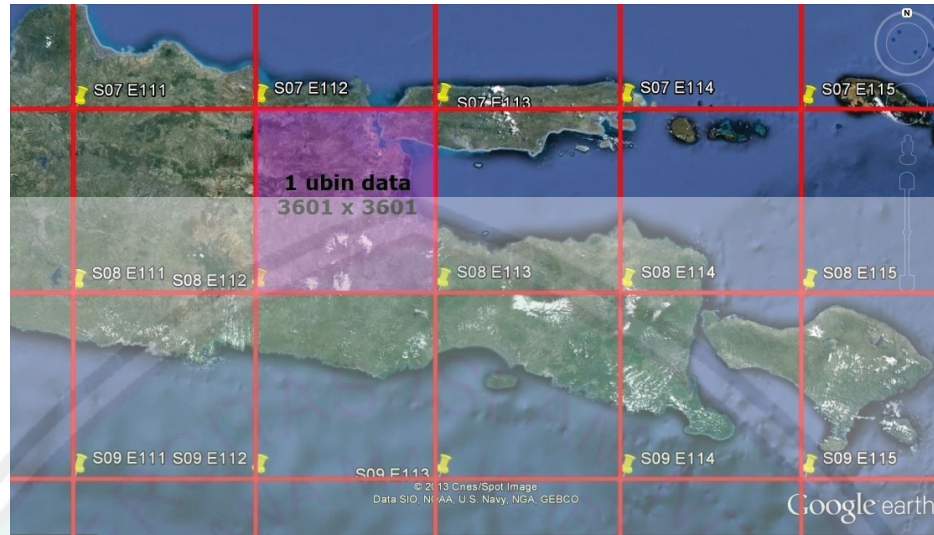
akurat daripada GPS. Grafik persentase perbedaan ketinggian rata-rata ditunjukkan pada Gambar 4.3 berikut.



Gambar 4.3 Grafik persentase perbedaan ketinggian rata-rata.

4.2.3 Pengujian Ketepatan Penyusunan Empat Ubin Data Terdekat

Pengujian ini dengan cara memvisualisasikan wilayah dimana wilayah tersebut berada pada ubin data yang berbeda. Ubin data *GDEM ASTER* tersusun seperti yang terlihat pada Gambar 4.4 berikut.



(Sumber: Google Earth)

Gambar 4.4 Susunan ubin data *GDEM ASTER*.

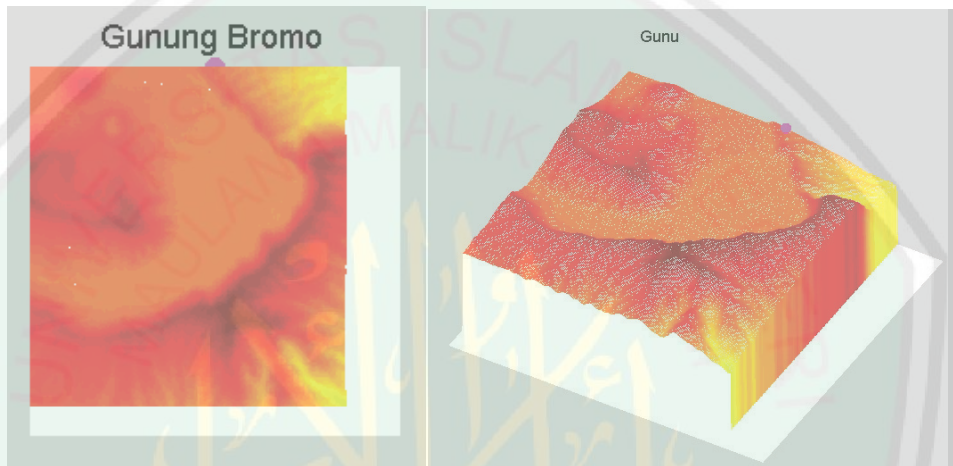
Sebagai contoh pengujian ini akan memvisualisasikan wilayah di Gunung Bromo dengan titik tepi 1 koordinat -7.931576, 112.9363 dan titik tepi 2 koordinat -8.005824, 113.0109. Koordinat tersebut dapat diilustrasikan pada Gambar 4.5 berikut.



(Sumber: Google Earth)

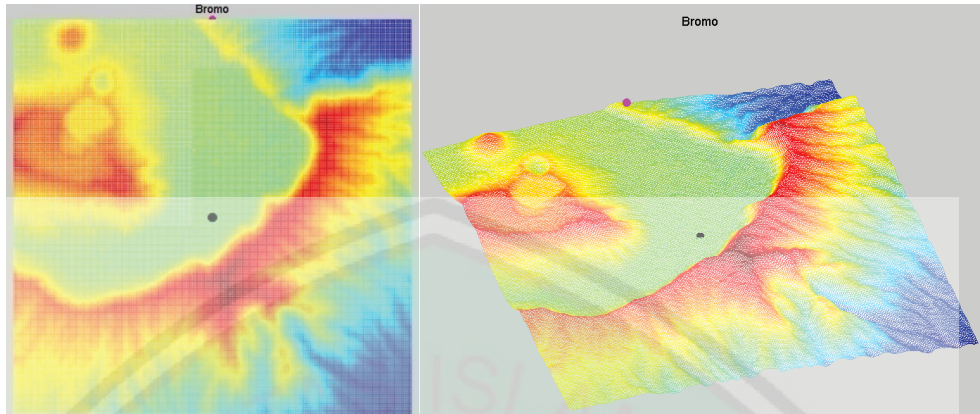
Gambar 4.5 Ilustrasi titik visualisasi.

Dari Gambar 4.5 ilustrasi titik visualisasi diatas terlihat bahwa wilayah visualisasi berada pada keempat bagian ubin data A1, A2, A3, dan A4. Jika tidak menggunakan *forward chaining* untuk menyusun empat ubin data yang berdekatan maka hasil visualisasi akan terpotong seperti yang terlihat pada Gambar 4.6 berikut.



Gambar 4.5 Hasil visualisasi tanpa penyusunan empat ubin data.

Jika menggunakan *forward chaining* untuk menyusun empat ubin data yang berdekatan, hasil visualisasi terlihat seperti yang ditunjukkan pada Gambar 4.7 dibawah ini, gambar yang dihasilkan sesuai dengan kontur tanah yang semestinya. Artinya pada wilayah perpotongan data dari keempat ubin data tidak terdapat keganjilan kontur tanah, sehingga dapat disimpulkan bahwa metode *forward chaining* berhasil menyusun keempat ubin data terdekat.



Gambar 4.7 Hasil visualisasi dengan penyusunan empat ubin data.

4.2.4 Pengujian Interpolasi Linier

Pengujian interpolasi linier memvisualisasikan pohon dengan lokasi *latitude-longitude* sebenarnya, contoh kasus yang diambil adalah visualisasi pohon beringin yang berlokasi di depan gedung Rektorat UIN Maulana Malik Ibrahim Malang seperti yang terlihat pada Gambar 4.8 berikut.



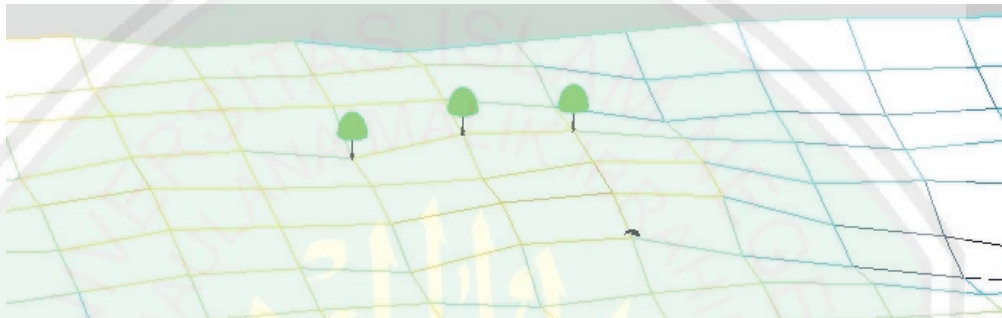
Gambar 4.8 Pohon yang akan divisualisasikan.

Keempat pohon seperti yang terlihat pada Gambar 4.8 tersebut didapatkan titik koordinat *latitude-longitudenya* menggunakan GPS Garmin seperti yang terlihat pada tabel 4.4 berikut.

Tabel 4.4: Titik koordinat pohon

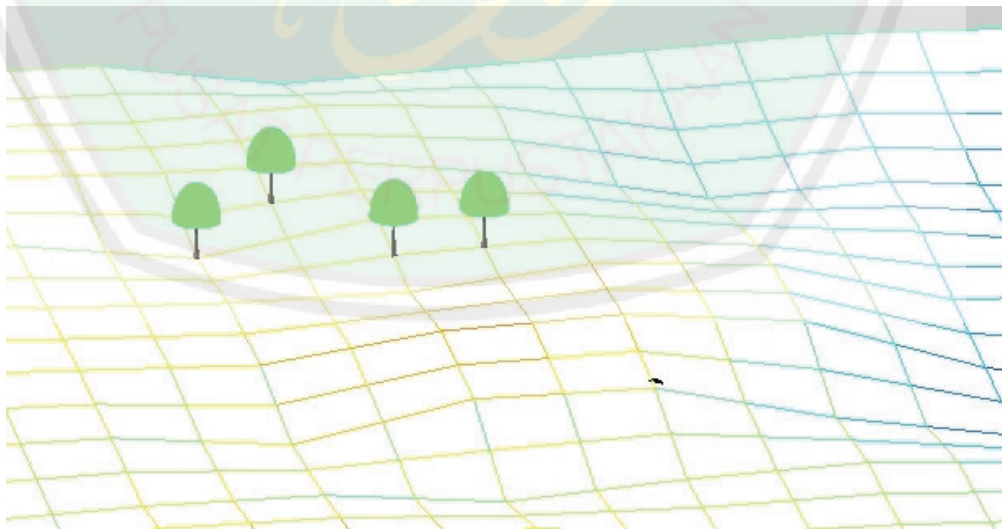
No Pohon	Latitude	Longitude
1	-7.95057	112.6086
2	-7.95076	112.6087
3	-7.95086	112.6085
4	-7.95104	112.6085

Data dari tabel 4.4 titik koordinat pohon diatas disimpan dalam sebuah fail, aplikasi akan membaca data tersebut dan memvisualisasikan pohon sesuai dengan koordinat tersebut, seperti yang telah disinggung sebelumnya bahwa titik koordinat tersebut akan dicari titik elevasi terdekatnya. Hasil dari visualisasi tanpa interpolasi linier terlihat seperti pada Gambar 4.9 berikut.



Gambar 4.9 Visualisasi pohon tanpa interpolasi.

Hasil dari visualisasi dengan interpolasi linier terlihat seperti pada Gambar 4.10 berikut.

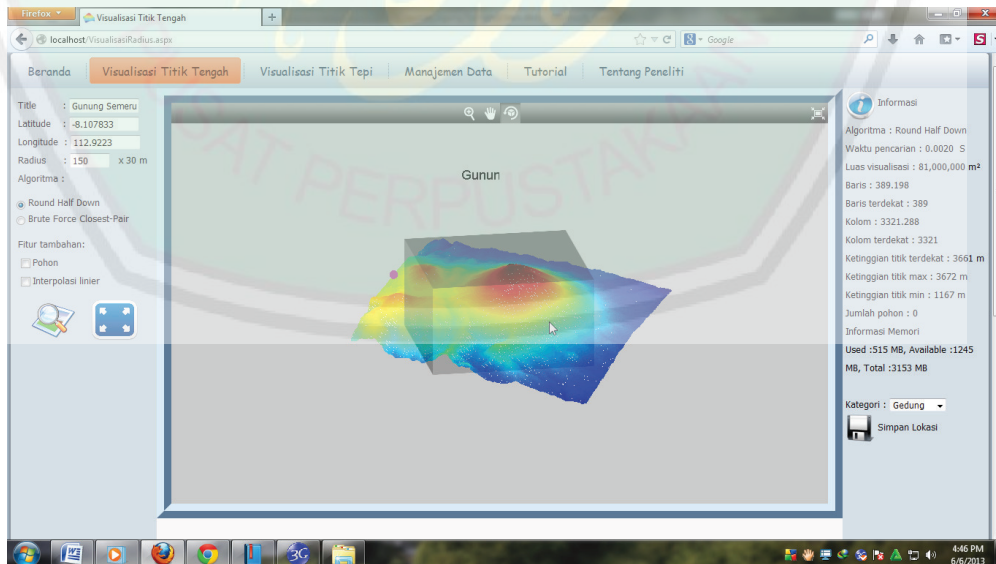


Gambar 4.10 Visualisasi pohon dengan interpolasi.

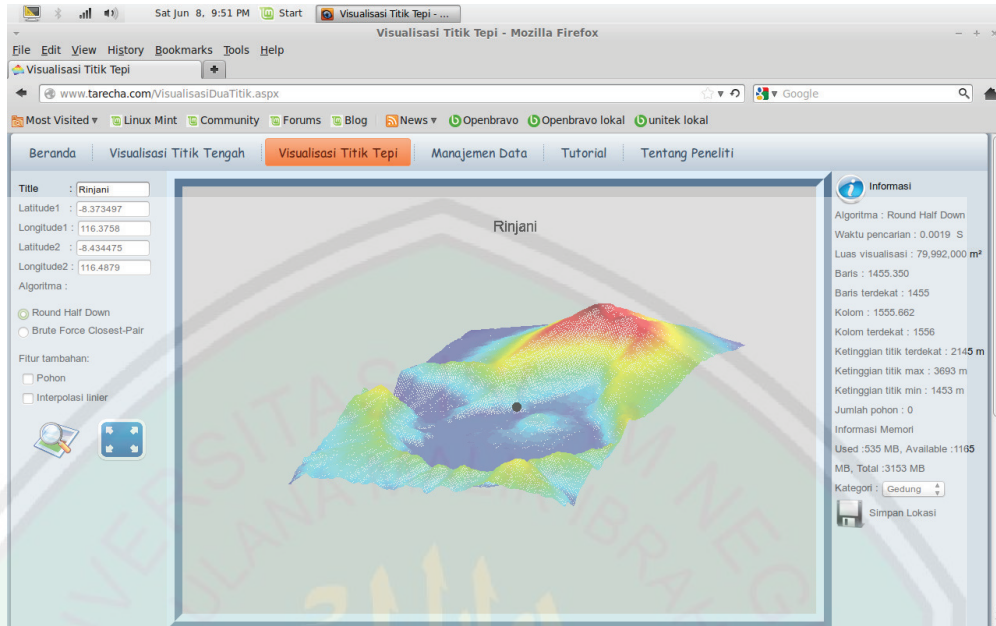
Jumlah pohon yang divisualisasikan diatas kontur tanah sebanyak empat buah namun terdapat perbedaan gambar yang dihasilkan, Gambar 4.9 visualisasi pohon tanpa interpolasi terlihat hanya tiga buah pohon saja, hal ini disebabkan karena jarak antar kedua pohon terlalu dekat sehingga ketika dicari titik elevasi terdekat dengan pohon menunjukkan titik yang sama dan gambar pohon tersebut tertumpuk pada titik tersebut. Jumlah pohon yang terlihat pada gambar 4.10 visualisasi pohon dengan interpolasi terlihat sebanyak empat buah dan letak pohon tidak segaris lurus tapi letak tersebut lebih mirip dengan letak yang sebenarnya seperti yang terlihat pada Gambar 4.8. Dari perbedaan gambar yang dihasilkan tersebut dapat disimpulkan bahwa dengan interpolasi ketelitian visualisasi meningkat.

4.2.5 Pengujian Kompatibilitas pada Perangkat

Pengujian ini menguji apakah aplikasi dapat berjalan pada berbagai perangkat yang berbeda, hasil pengujian ditunjukkan pada gambar-gambar berikut.



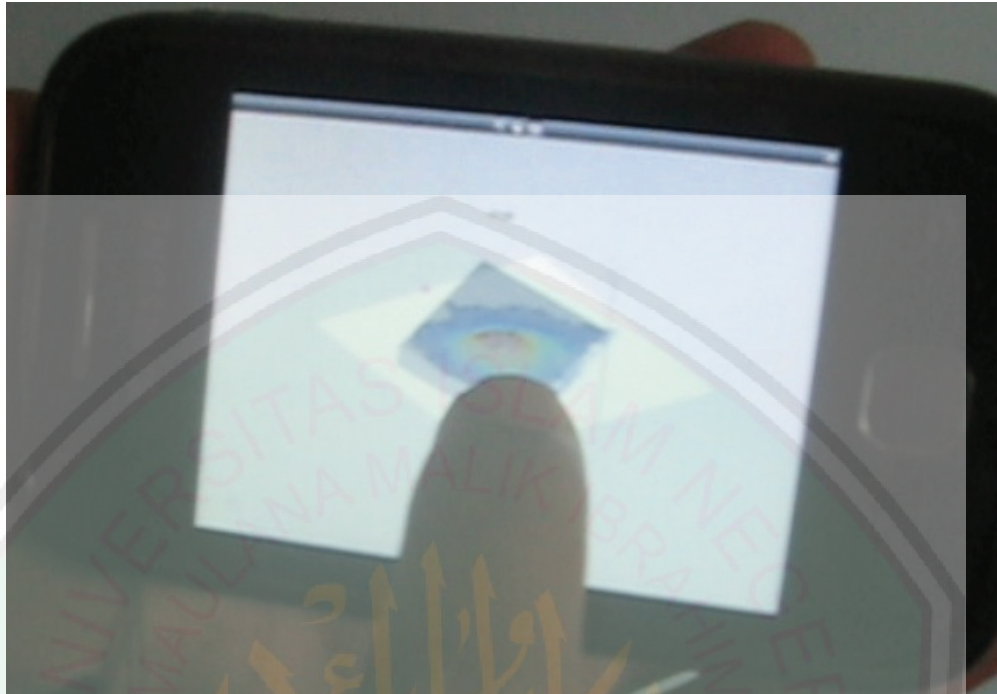
Gambar 4.11 Pengujian pada Windows 7.



Gambar 4.12 Pengujian pada Linux Mint.



Gambar 4.13 Pengujian pada Tablet Android.



Gambar 4.14 Pengujian pada Smartphone Android.

Dari pengujian yang telah dilakukan dapat disimpulkan bahwa aplikasi visualisasi dapat berjalan di berbagai perangkat bagi yang menggunakan *mouse* ataupun *touch screen*, tentu saja menggunakan *web browser* yang direkomendasikan yaitu *Mozilla Firefox*.

4.3 Pembahasan Pengujian

Setelah didapatkan hasil pengujian kemudian hasil pengujian tersebut dibahas agar didapatkan pengetahuan dan ilmu baru berdasarkan hasil penelitian.

Pembahasan hasil pengujian antara lain sebagai berikut:

1. Pembahasan hasil pengujian ketepatan dan kecepatan pencarian titik tengah

Algoritma *round-half-down* dan *brute force closest-pair* terbukti mampu untuk menemukan pasangan titik terdekat dengan tepat yang sama, namun perbedaan kedua algoritma tersebut adalah waktu proses pencarian dimana algoritma *round-half-down* 250 x lebih cepat daripada algoritma *brute force closest-pair*. Hal ini disebabkan karena algoritma *round-half-down* hanya menyeleksi titik yang sudah diprediksi sebagai pasangan terdekatnya yaitu batas bawah dan batas atas, sedangkan algoritma *brute force closest-pair* menyeleksi seluruh titik yang ada dan menghitung *euclidean distance* satu persatu. Kelemahan algoritma *round-half-down* dalam mencari titik terdekat adalah interval tiap titik tersebut harus = 1, algoritma tersebut tidak dapat digunakan untuk mencari titik terdekat bila interval antar titik tidak beraturan, sedangkan algoritma *brute force closest-pair* menawarkan fleksibilitas dalam pencarian pasangan titik terdekat karena membandingkan *euclidean distance* terhadap seluruh titik yang akan dicari pasangan terdekatnya tanpa terpengaruh berapapun interval antar titiknya.

2. Pembahasan hasil pengujian ketelitian ketinggian titik tengah

Ketinggian yang dihasilkan pada titik yang sama antara aplikasi visualisasi dan Google Earth memiliki tingkat *error* yang kecil yaitu $< 0,5 \%$ bila dibandingkan dengan antara GPS Garmin dan Google Earth sebesar 2,18%.

Hal ini membuktikan bahwa data satelit dalam hal ini *GDEM ASTER* yang digunakan pada aplikasi visualisasi dan *GDEM SRTM* yang digunakan pada Google Earth memiliki perbedaan yang relatif kecil bila dibandingkan menggunakan GPS Garmin. Keuntungan menggunakan data *GDEM ASTER* dibandingkan *GDEM SRTM* adalah resolusi *GDEM ASTER* lebih detail dan cakupan data lebih luas bila dibandingkan dengan *GDEM SRTM*. Beda antara GPS Garmin dan Google Earth cukup signifikan hal ini disebabkan oleh ketelitian GPS Garmin tersebut, jangankan dengan Google Earth dengan GPS Android pun memiliki perbedaan ketinggian meski pada titik yang sama, hal ini dipengaruhi kualitas perangkat tersebut namun hal ini masih bisa dianggap wajar karena tiap perangkat pasti memiliki toleransi dalam pengukuran.

3. Pembahasan hasil pengujian ketepatan penyusunan empat ubin data terdekat

Metode *forward chaining* terbukti mampu untuk menyusun empat ubin data terdekat dengan cara yang sederhana, hal ini dapat dilakukan karena format penamaan fail data *GDEM ASTER* sudah disusun sedemikian rupa sehingga mudah untuk menemukan fail terdekatnya berdasarkan nama fail. Wilayah visualisasi yang tersebar pada fail data yang berbeda mampu untuk disusun dengan tepat kemudian divisualisasikan.

4. Pembahasan hasil pengujian interpolasi linier

Interpolasi linier terbukti mampu untuk meningkatkan ketelitian visualisasi, semakin detail interpolasi maka ketelitian penempatan objek pohon akan semakin sesuai dengan lokasi aslinya. Semakin banyak titik baru yang dihasilkan interpolasi maka data yang akan diproses semakin besar, hal ini dapat menyebabkan proses untuk membangkitkan visual semakin lama. Nilai titik baru hasil interpolasi adalah hasil perhitungan matematis, bukan pengukuran sebenarnya dari ketinggian tanah, oleh karena itu interpolasi dapat digunakan seperlunya dengan mempertimbangkan besarnya data yang akan diproses dan sumber daya yang tersedia.

5. Pembahasan hasil pengujian kompatibilitas pada perangkat

Aplikasi berhasil dijalankan pada berbagai perangkat yang berbeda menggunakan *web browser* yang direkomendasikan yaitu *Mozilla Firefox*. Hasil yang didapat menggunakan *web browser* tersebut sama pada berbagai perangkat, namun hasil yang berbeda didapatkan bila menggunakan *web browser* lain seperti tampilan *full screen* yang berbeda-beda tiap *web browser*. Alasan dipilihnya *web browser Mozilla Firefox* karena *web browser* tersebut tersedia dalam berbagai *platform* sehingga pengalaman yang diberikan kepada pengguna cenderung sama.

4.4 Integrasi Visualisasi 3D Rupa Bumi dengan Islam

Seperti yang telah dikemukakan pada latar belakang, penelitian ini ingin mengungkap apa pelajaran dari bumi yang dihamparkan ?, petunjuk apa yang terkandung di dalamnya ? melalui visualisasi 3D rupa bumi ini. Pembahasan mengenai integrasi visualisasi 3D rupa bumi dengan Islam penulis kutip dari sebuah artikel dengan penulis anonim (<http://www.islamquest.net/id/archive/question/fa4232>) mengenai Bagaimana pandangan al-Qur'an sehubungan dengan konsep yang menyatakan bahwa bumi itu bulat? yang juga disadur dari beberapa kitab tafsir sebagai berikut.

Dalam Al-Qur'an sudah Al-Hijr ayat 19 berikut ini

وَالْأَرْضَ مَدَدْنَاهَا وَأَلْقَيْنَا فِيهَا رَوَاسِيَ وَأَنْبَتْنَا فِيهَا مِنْ كُلِّ شَيْءٍ مَوْزُونٍ

Artinya :

Dan Kami telah menghamparkan bumi dan menjadikan padanya gunung-gunung dan Kami tumbuhkan padanya segala sesuatu menurut ukuran.

“*madd al-ardh*” bermakna penghamparan panjang dan lebarnya. Dan sekiranya Tuhan tidak menghamparkan bumi dan bumi dijejali dengan deretan gunung-gunung maka bumi tidak pantas untuk dihuni, tidak dapat dijadikan tempat untuk bercocok tanam dan makhluk hidup tidak akan mendapatkan kesempurnaan hidup di dalamnya. Yang dimaksud dengan “*madda*” dan menghampar tidak bermaksud bahwa bumi tidak berbentuk bulat melainkan maksudnya adalah hamparan bumi tidak diciptakan menonjol secara utuh dan berbentuk rata. Apabila bumi diciptakan menonjol dan berbentuk rata (tidak bulat) maka manusia

tidak dapat hidup dengan baik di muka bumi, mengadakan perkebunan dan pertanian. Pendeknya bumi tidak akan menjadi tempat yang menyenangkan bagi manusia sebagaimana irama ayat-ayat setelahnya berada pada tataran menghitung pelbagai karunia yang menyenangkan bagi manusia yang juga merupakan penegas masalah ini.

Dalam Al-Qur'an sudah An-Naba' ayat 6-7 berikut ini:

أَلَمْ نَجْعَلِ الْأَرْضَ مِهْدًا ﴿٦﴾ وَالْجِبَالَ أَوْتَادًا ﴿٧﴾

Artinya :

Bukankah Kami telah menjadikan bumi itu sebagai hamparan?, dan gunung-gunung sebagai pasak?

Redaksi “*mihâd*” pada ayat ini bermakna tempat yang siap pakai, menghampar dan teratur. Derivatnya dari kata “*ma-h-d*” yang bermakna sebuah tempat untuk istirahat anak kecil (buaian atau tempat tidur). Karena itu, buaian yang disiapkan untuk anak kecil disebut sebagai *mahd*. Dengan demikian, *mahd* dan *mihad* adalah sebuah tempat yang seperti buaian yang siap untuk membina dan membangun sebagaimana yang diungkapkan dalam Al-Qur'an surah Thaha ayat 53 berikut ini:

الَّذِي جَعَلَ لَكُمُ الْأَرْضَ مِهْدًا وَسَلَكَ لَكُمْ فِيهَا سُبُلًا وَأَنْزَلَ مِنَ السَّمَاءِ

مَاءً فَأَخْرَجْنَا بِهِ أَزْوَاجًا مِّنْ نَّبَاتٍ شَتَّى ﴿٥٣﴾

Artinya :

Yang telah menjadikan bagimu bumi sebagai hamparan dan Yang telah menjadikan bagimu di bumi itu jalan-jalan, dan menurunkan dari langit air hujan. Maka Kami tumbuhkan dengan air hujan itu berjenis-jenis dari tumbuh-tumbuhan yang bermacam-macam.

SubhanaAllah setelah mengkaji dalam Al-Qur'an bagaimana bumi dihamparkan dan setelah melihat hasil visualisasi 3D hamparan bumi dapat diketahui bahwa hamparan bumi yang luas itu adalah karunia Allah SWT yang menyenangkan bagi manusia. Hamparan bumi tersebut dapat diamati melalui visualisasi 3D rupa bumi, saat pengambilan data ketinggian pada Gunung Panderman yang telah dilakukan penelenti aplikasi ini mampu untuk memberikan visual Gunung Panderman dengan baik, detail kontur Gunung Panderman terlihat dengan jelas. Diharapkan aplikasi ini dapat dimanfaatkan oleh pihak-pihak yang membutuhkan informasi visual 3D rupa bumi untuk berbagai kepentingan seperti yang telah dipaparkan pada latar belakang.

BAB V

PENUTUP

5.1 Kesimpulan

Kesimpulan yang diperoleh dari penelitian visualisasi 3D rupa bumi berbasis data *GDEM ASTER* adalah sebagai berikut:

1. Algoritma *round-half-down* dan *brute force closest-pair* dapat menemukan pasangan titik terdekat dengan tepat. Algoritma *round-half-down* 250 x lebih cepat daripada algoritma *brute force closest-pair* dalam mencari pasangan titik terdekat. Algoritma *round-half-down* hanya dapat digunakan untuk mencari titik terdekat jika interval masing-masing titik sebesar 1.
2. Metode *euclidean distance* dan *forward chaining* dapat digunakan untuk mendukung visualisasi dan memberikan hasil yang sesuai.
3. Perbedaan ketelitian elevasi tanah antara Google Earth yang menggunakan data *GDEM SRTM* 90 meter dan aplikasi visualisasi yang menggunakan data *GDEM ASTER* 30 meter memiliki perbedaan yang sangat kecil yaitu $< 0,5 \%$ sehingga dapat disimpulkan bahwa aplikasi visualisasi lebih akurat bila dibandingkan dengan GPS Garmin 80 Mil sebesar 2.18 %.
4. Interpolasi terbukti mampu meningkatkan ketelitian visualisasi dibuktikan dengan peletakan objek berupa pohon. Dengan menggunakan antarmuka berupa halaman *web* dan dapat diakses melalui internet maka aplikasi visualisasi dapat digunakan secara luas dan relatif tanpa terkendala pada berbagai perangkat.

5.2 Saran

Dalam penelitian ini masih terdapat beberapa kekurangan. Untuk pengembangan lebih lanjut terdapat saran-saran sebagai berikut ini.

1. Metode interpolasi dapat diperbanyak lagi seperti interpolasi *Nearest Neighbor*, interpolasi *Cubic Spline*, interpolasi kuadrat, dan lain sebagainya.
2. Visualisasi objek dapat diperbanyak lagi seperti visualisasi gedung, jalan, sungai, dan lain sebagainya.
3. Penelitian selanjutnya dapat ditambahkan simulasi bencana yang berkaitan dengan rupa bumi seperti gunung meletus, wilayah kebakaran hutan, dan lain sebagainya.

BAB IV

HASIL DAN PEMBAHASAN

Pada bab ini membahas tentang pengujian dan pembahasan terhadap hasil pengujian aplikasi visualisasi.

4.1 Langkah – langkah pengujian

Langkah – langkah pengujian menjelaskan bagaimana aplikasi diuji. Terdapat beberapa parameter yang akan diuji, antara lain sebagai berikut:

1. Langkah pengujian ketepatan dan kecepatan pencarian titik tengah

Pengujian ini membandingkan implementasi antara algoritma *round-half-down* dan algoritma *brute force closest-pair* untuk mencari titik *latitude-longitude* dengan titik terdekat baris-kolom matriks yang menjadi titik tengah wilayah visualisasi. Parameter yang dibandingkan adalah ketepatan dan kecepatan pencarian titik terdekat *latitude-longitude* dengan titik baris-kolom matriks 3601 x 3601. Pengujian ini menggunakan komputer dengan spesifikasi sebagai berikut.

Prosesor	: AMD Phenom x2 3.2 GHz
Memory	: DDR3 3GB
VGA	: Ati Radeon 4920
OS	: Windows 7

Pengujian dilakukan dengan melakukan visualisasi 5 buah titik lokasi dengan tiap titik menggunakan dua algoritma yaitu *round-half-down (rhd)* dan *brute force closest-pair (bf)*. Nilai baris-kolom asli akan dibandingkan dengan nilai baris-kolom terdekat pada masing-masing algoritma diuji apakah tepat mencari titik terdekatnya. Setelah itu diuji berapa lama waktu yang

dihabiskan untuk proses pencarian titik terdekat tersebut menggunakan kedua algoritma tersebut. Karena terdapat dua metode input yaitu titik tengah dan titik tepi yang sebenarnya sama hanya saja pada titik tepi harus dinormalisasikan dahulu maka digunakan titik tengah untuk mewakili pengujian ini.

2. Langkah pengujian ketelitian ketinggian titik tengah

Pengujian ini membandingkan data ketinggian pada titik tengah wilayah visualisasi. Titik tengah ini memiliki nilai ketinggian yang akan dibandingkan dengan ketinggian GPS Garmin 80 Mil dan Google Earth pada titik yang sama, sedangkan GPS HP Android hanya digunakan sebagai data referensi saja untuk membuktikan bahwa pada lokasi yang sama sekalipun hasil pengukuran ketinggian menggunakan GPS memiliki hasil yang berbeda tergantung kualitas dan tingkat ketelitian masing-masing GPS. Setelah data tersebut diperoleh maka diperoleh nilai ketinggian dari masing-masing sumber tersebut kemudian dibandingkan satu sama lain dan persentase perbedaan ketinggian absolut tersebut dihitung kemudian dirata-rata, akan terlihat berapa persen tingkat ketelitian ketinggian. Dengan adanya pengujian ini maka dapat diketahui berapa persenkah *error* ketelitian aplikasi visualisasi yang dibuat.

3. Langkah pengujian ketepatan penyusunan empat ubin data terdekat

Pengujian ketepatan penyusunan empat ubin data terdekat menguji apakah penggunaan metode *forward chaining* berhasil untuk menyusun empat ubin terdekat tersebut kemudian divisualisasikan. Cara mengujinya dengan memvisualisasikan titik tepi dimana masing-masing titik tersebut berada pada ubin data yang berbeda.

4. Langkah pengujian interpolasi linier

Untuk menguji apakah interpolasi linier memberikan dampak meningkatnya ketelitian visualisasi maka dilakukan pengujian visualisasi pohon tanpa interpolasi dan dengan interpolasi, akan terlihat bagaimana ketelitian meningkat dengan interpolasi.

5. Langkah pengujian kompatibilitas pada perangkat

Pengujian kompatibilitas pada perangkat menguji apakah aplikasi visualisasi dapat diakses oleh banyak perangkat yang berbeda. Karena keterbatasan perangkat yang ada maka pengujian ini hanya menguji pada perangkat desktop Windows 7, Linux Mint, Tablet Android, dan Smartphone Android.

4.2 Hasil Pengujian

Setelah langkah – langkah pengujian dilakukan maka didapatkan data hasil pengujian. Data tersebut disajikan dalam bentuk tabel dan grafik. Hasil pengujian antara lain sebagai berikut:

4.2.1 Pengujian Ketepatan dan Kecepatan Pencarian Titik Tengah

Hasil pengujian ditunjukkan pada tabel 4.1 dan 4.2 berikut.

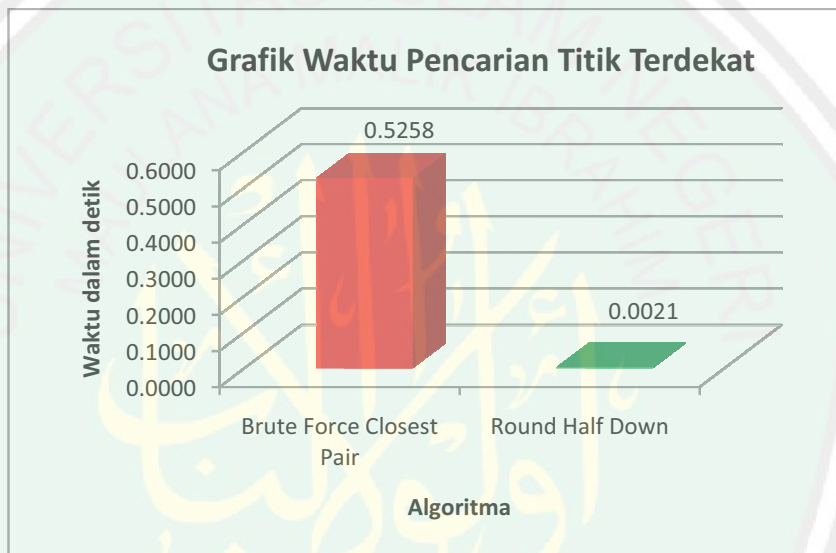
Tabel 4.1: Pengujian algoritma *brute force closest-pair*

No	Lokasi	Latitude	Longitude	Radius	Baris	Baris terdekat	Kolom	Kolom terdekat	Waktu Pencarian Brute Force Closest Pair
1	Gunung Arjuna	-7.765040	112.5897	150	2755.144	2755	2123.916	2124	0.6721
2	Gunung Pandoman	-7.904006	112.4965	35	3255.422	3255	1788.393	1788	0.4939
3	Pulau Sempu	-8.438431	112.6903	150	1579.351	1579	2486.080	2486	0.5186
4	Tambang Grasberg	-4.056800	137.1130	50	205.480	205	407.824	408	0.4718
5	Anak Krakatau	-6.101361	105.4245	40	365.899	366	1529.192	1529	0.4727
Waktu rata - rata dalam detik (s)									
0.5258									

Tabel 4.2: Pengujian ketepatan algoritma *round-half-down*.

No	Lokasi	Latitude	Longitude	Radius	Baris	Baris terdekat	Kolom	Kolom terdekat	Waktu Pencarian Round Half Down
1	Gunung Arjuna	-7.765040	112.5897	150	2755.144	2755	2123.916	2124	0.0019
2	Gunung Pandoman	-7.904006	112.4965	35	3255.422	3255	1788.393	1788	0.0020
3	Pulau Sempu	-8.438431	112.6903	150	1579.351	1579	2486.080	2486	0.0022
4	Tambang Grasberg	-4.056800	137.1130	50	205.480	205	407.824	408	0.0019
5	Anak Krakatau	-6.101361	105.4245	40	365.899	366	1529.192	1529	0.0024
Waktu rata - rata dalam detik (s)									
0.0021									

Dari tabel 4.1 dan 4.2 diatas dapat diketahui bahwa baris-kolom terdekat hasil pencarian kedua algoritma menghasilkan nilai yang tepat sama. Artinya menggunakan kedua algoritma tersebut akan menghasilkan gambar visual yang sama karena titik tengahnya sama. Sedangkan yang berbeda adalah waktu pencarian titik tengah terdekat tersebut. Perbedaan waktu tersebut dapat dilihat lebih jelas dalam Gambar 4.1 berikut.



Gambar 4.1 Grafik waktu pencarian titik terdekat.

Seperti yang terlihat pada gambar 4.1 Grafik waktu pencarian titik terdekat diatas bahwa bila dibandingkan maka perbandingan kedua algoritma tersebut adalah $\frac{\text{waktu BF}}{\text{waktu RHD}} = \frac{0.5258}{0.0021} = 250.38$, dapat disimpulkan bahwa algoritma *round-half-down* lebih cepat **250 x** daripada algoritma *brute force closest-pair*. Tentu saja waktu tersebut tergantung dari sumber daya komputer, semakin cepat komputer maka akan semakin cepat pula waktu pencarian titik, begitu pula sebaliknya.

4.2.2 Pengujian Ketelitian Ketinggian Titik Tengah

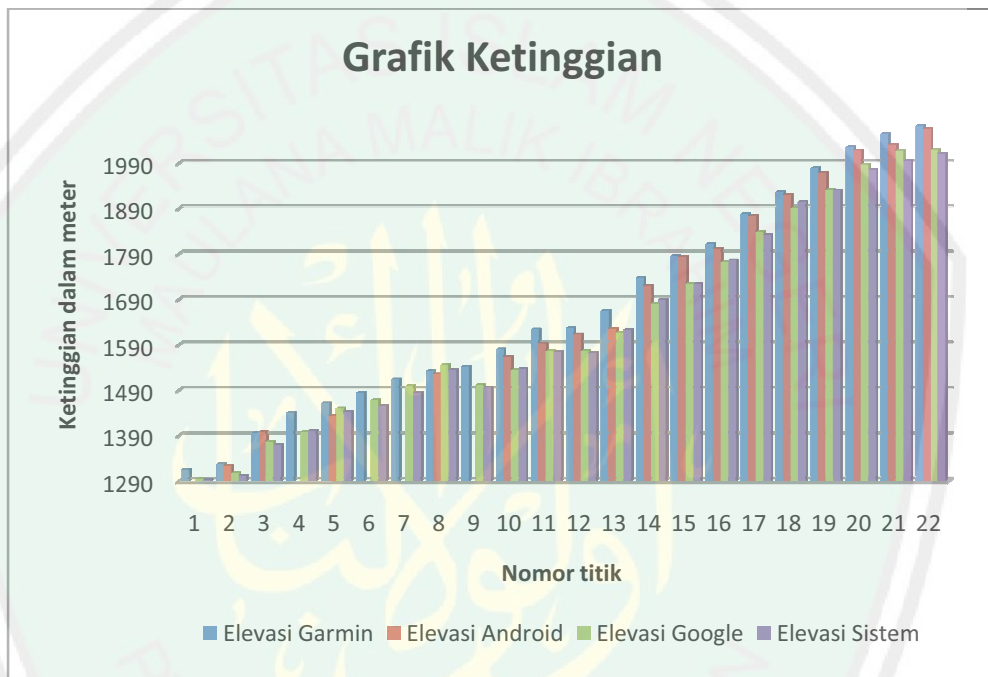
Hasil pengujian ditunjukkan pada tabel 4.3 berikut.



Tabel 4.3: Pengujian ketelitian ketinggian titik tengah

No	Waypoint GPS	Latitude GPS	Longitude GPS	Elevasi Garmin	Elevasi Android	Elevasi Google	Elevasi Aplikasi	Perbedaan Garmin / Google	Perbedaan Aplikasi / Google
1	56	-7.88725	112.49844	1315		1294	1293	1.62%	0.08%
2	57	-7.88785	112.49760	1327	1323	1308	1302	1.45%	0.46%
3	58	-7.89038	112.49552	1396	1398	1377	1369	1.38%	0.58%
4	81	-7.89009	112.49411	1440		1399	1401	2.93%	0.14%
5	59	-7.89229	112.49470	1461	1433	1450	1442	0.76%	0.55%
6	80	-7.89120	112.49275	1484		1468	1455	1.09%	0.89%
7	79	-7.89227	112.49263	1514		1500	1484	0.93%	1.07%
8	60	-7.89452	112.49351	1532	1525	1546	1534	0.91%	0.78%
9	78	-7.89227	112.49225	1542		1502	1496	2.66%	0.40%
10	77	-7.89318	112.49200	1580	1563	1534	1537	3.00%	0.20%
11	76	-7.89412	112.49216	1624	1593	1576	1575	3.05%	0.06%
12	61	-7.89436	112.49224	1627	1613	1576	1572	3.24%	0.25%
13	62	-7.89586	112.46213	1665	1625	1617	1623	2.97%	0.37%
14	63	-7.89721	112.46220	1738	1720	1681	1689	3.39%	0.48%
15	64	-7.89827	112.49248	1786	1784	1725	1724	3.54%	0.06%
16	65	-7.89992	112.49318	1812	1802	1772	1776	2.26%	0.23%
17	66	-7.90113	112.49415	1878	1873	1838	1832	2.18%	0.33%
18	67	-7.90184	112.49465	1927	1920	1891	1905	1.90%	0.74%
19	68	-7.90231	112.49515	1979	1968	1931	1930	2.49%	0.05%
20	69	-7.90291	112.49586	2025	2018	1987	1976	1.91%	0.55%
21	70	-7.90262	112.49737	2054	2031	2017	1995	1.83%	1.09%
22	71	-7.90263	112.49781	2072	2065	2020	2010	2.57%	0.50%
Perbedaan rata - rata								2.18%	0.45%

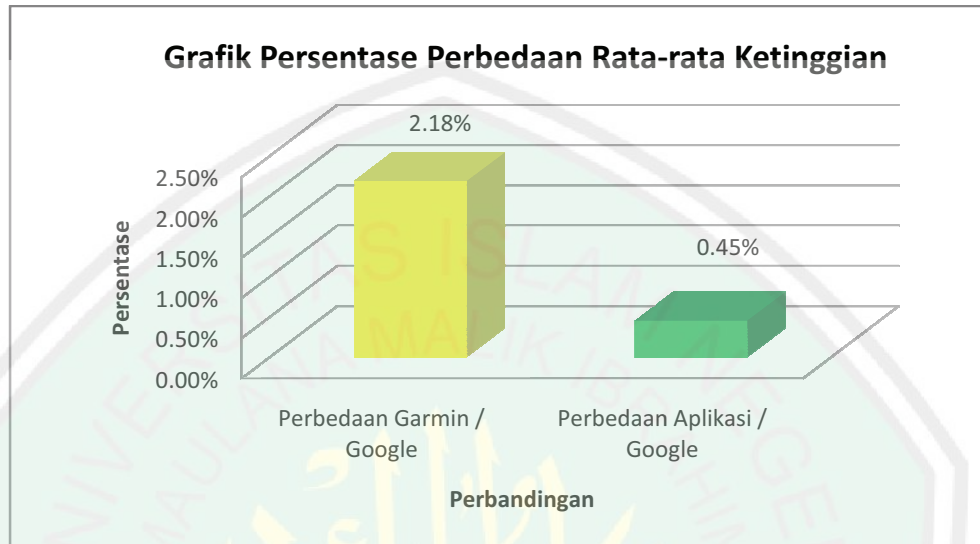
Dari data pada tabel 4.3 diatas didapatkan grafik ketinggian seperti yang terlihat pada Gambar 4.2, terlihat bahwa perbedaan ketinggian antara elevasi tanah pada Google Earth tidak terlalu berbeda jauh dengan aplikasi visualisasi yang dibuat, sedangkan bila dibandingkan dengan elevasi GPS Garmin maupun Android memiliki beda ketinggian yang cukup signifikan.



Gambar 4.2 Grafik ketinggian.

Dari grafik ketinggian pada Gambar 4.2 diatas akan dibandingkan perbedaan absolut dari ketinggian masing-masing sumber data. Rata-rata persentase perbedaan elevasi GPS Garmin dibanding Google Earth adalah 2,18 %, sedangkan perbedaan elevasi aplikasi visualisasi dibandingkan dengan Google Earth adalah 0,45 %. Hal ini membuktikan bahwa aplikasi yang dibuat lebih

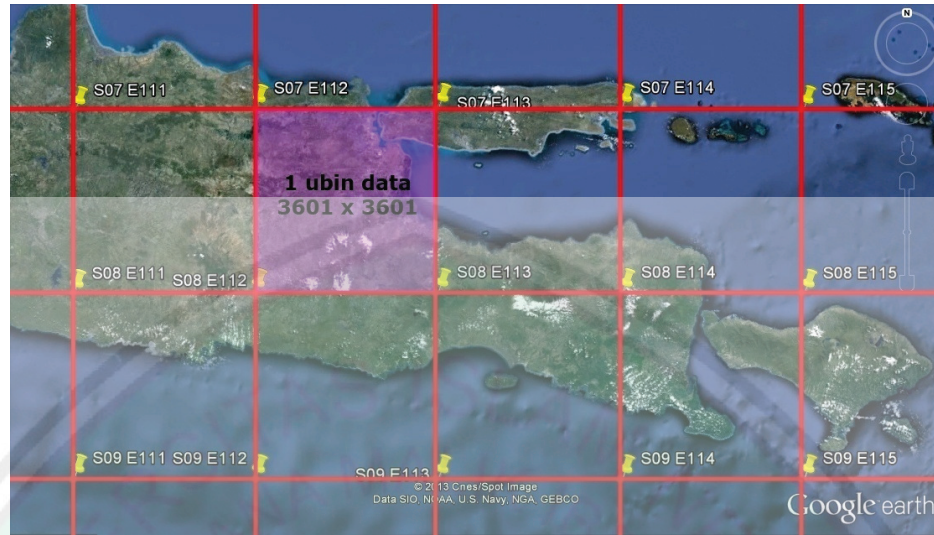
akurat daripada GPS. Grafik perbandingan perbezaan ketinggian rata-rata ditunjukkan pada Gambar 4.3 berikut.



Gambar 4.3 Grafik perbandingan perbezaan ketinggian rata-rata.

4.2.3 Pengujian Ketepatan Penyusunan Empat Ubin Data Terdekat

Pengujian ini dengan cara memvisualisasikan wilayah dimana wilayah tersebut berada pada ubin data yang berbeza. Ubin data *GDEM ASTER* tersusun seperti yang terlihat pada Gambar 4.4 berikut.



(Sumber: Google Earth)

Gambar 4.4 Susunan ubin data *GDEM ASTER*.

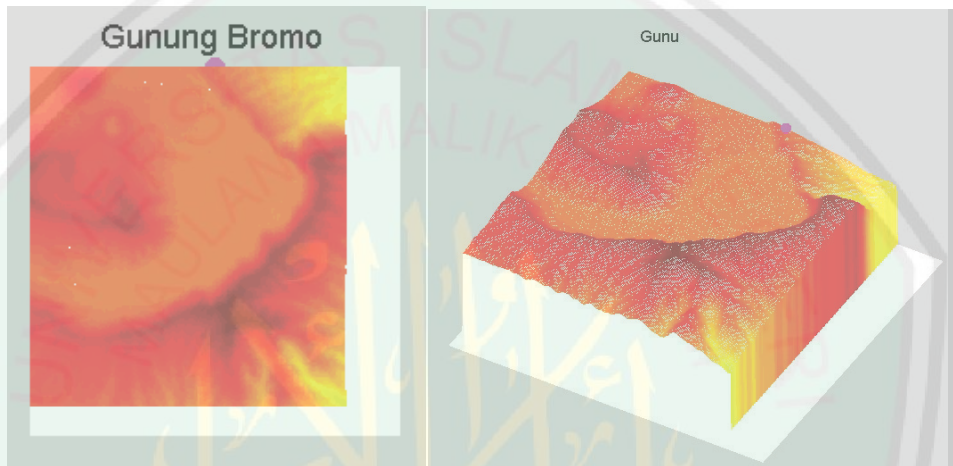
Sebagai contoh pengujian ini akan memvisualisasikan wilayah di Gunung Bromo dengan titik tepi 1 koordinat -7.931576, 112.9363 dan titik tepi 2 koordinat -8.005824, 113.0109. Koordinat tersebut dapat diilustrasikan pada Gambar 4.5 berikut.



(Sumber: Google Earth)

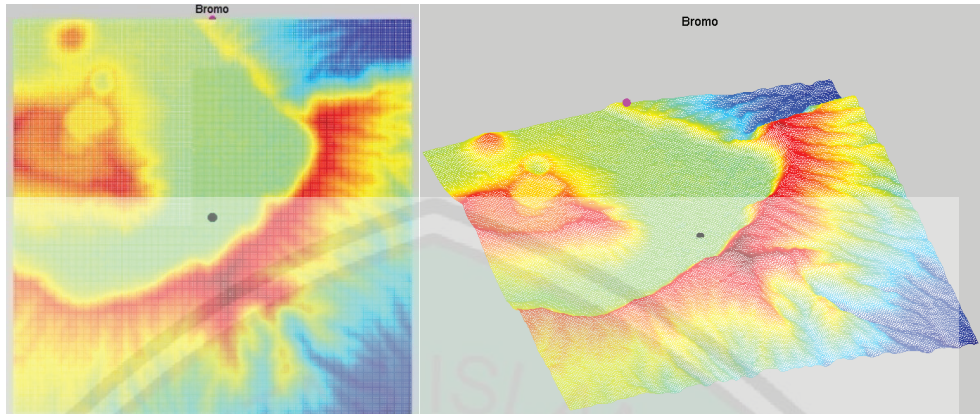
Gambar 4.5 Ilustrasi titik visualisasi.

Dari Gambar 4.5 ilustrasi titik visualisasi diatas terlihat bahwa wilayah visualisasi berada pada keempat bagian ubin data A1, A2, A3, dan A4. Jika tidak menggunakan *forward chaining* untuk menyusun empat ubin data yang berdekatan maka hasil visualisasi akan terpotong seperti yang terlihat pada Gambar 4.6 berikut.



Gambar 4.5 Hasil visualisasi tanpa penyusunan empat ubin data.

Jika menggunakan *forward chaining* untuk menyusun empat ubin data yang berdekatan, hasil visualisasi terlihat seperti yang ditunjukkan pada Gambar 4.7 dibawah ini, gambar yang dihasilkan sesuai dengan kontur tanah yang semestinya. Artinya pada wilayah perpotongan data dari keempat ubin data tidak terdapat keganjilan kontur tanah, sehingga dapat disimpulkan bahwa metode *forward chaining* berhasil menyusun keempat ubin data terdekat.



Gambar 4.7 Hasil visualisasi dengan penyusunan empat ubin data.

4.2.4 Pengujian Interpolasi Linier

Pengujian interpolasi linier memvisualisasikan pohon dengan lokasi *latitude-longitude* sebenarnya, contoh kasus yang diambil adalah visualisasi pohon beringin yang berlokasi di depan gedung Rektorat UIN Maulana Malik Ibrahim Malang seperti yang terlihat pada Gambar 4.8 berikut.



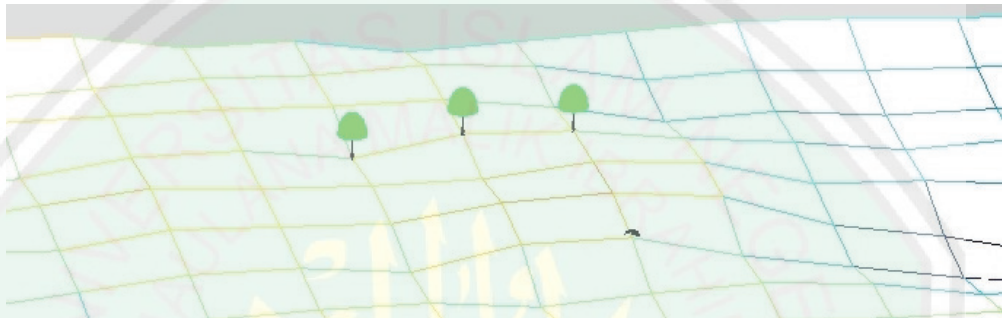
Gambar 4.8 Pohon yang akan divisualisasikan.

Keempat pohon seperti yang terlihat pada Gambar 4.8 tersebut didapatkan titik koordinat *latitude-longitudenya* menggunakan GPS Garmin seperti yang terlihat pada tabel 4.4 berikut.

Tabel 4.4: Titik koordinat pohon

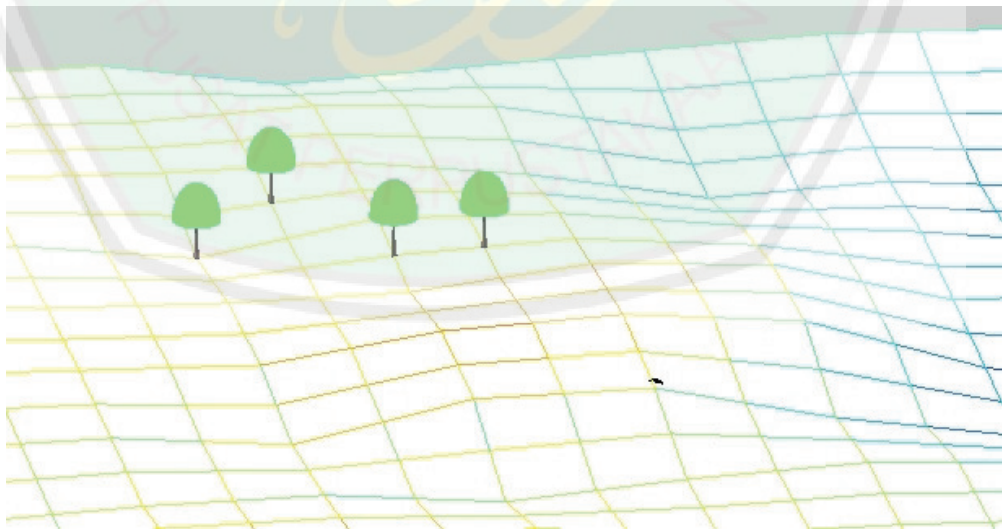
No Pohon	Latitude	Longitude
1	-7.95057	112.6086
2	-7.95076	112.6087
3	-7.95086	112.6085
4	-7.95104	112.6085

Data dari tabel 4.4 titik koordinat pohon diatas disimpan dalam sebuah fail, aplikasi akan membaca data tersebut dan memvisualisasikan pohon sesuai dengan koordinat tersebut, seperti yang telah disinggung sebelumnya bahwa titik koordinat tersebut akan dicari titik elevasi terdekatnya. Hasil dari visualisasi tanpa interpolasi linier terlihat seperti pada Gambar 4.9 berikut.



Gambar 4.9 Visualisasi pohon tanpa interpolasi.

Hasil dari visualisasi dengan interpolasi linier terlihat seperti pada Gambar 4.10 berikut.

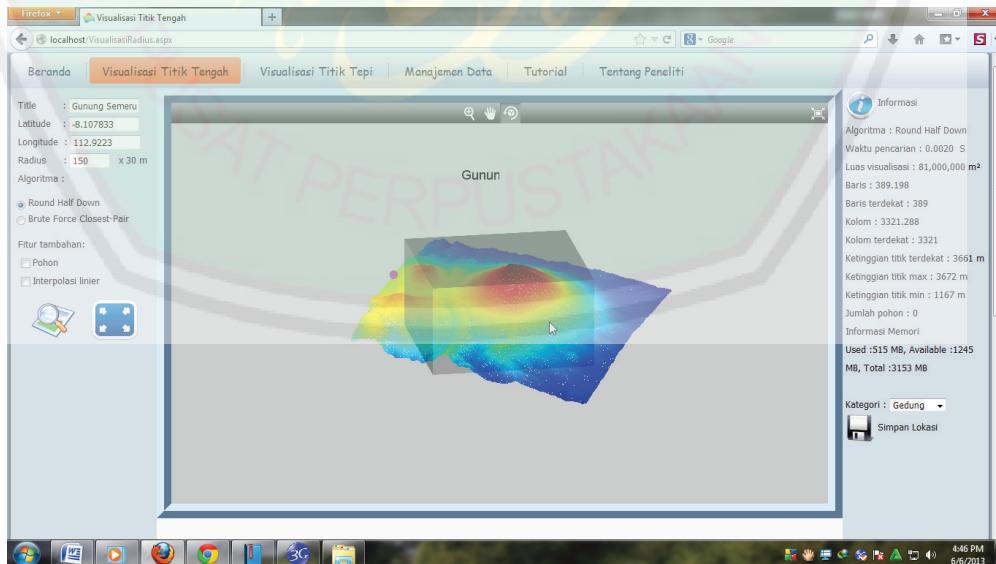


Gambar 4.10 Visualisasi pohon dengan interpolasi.

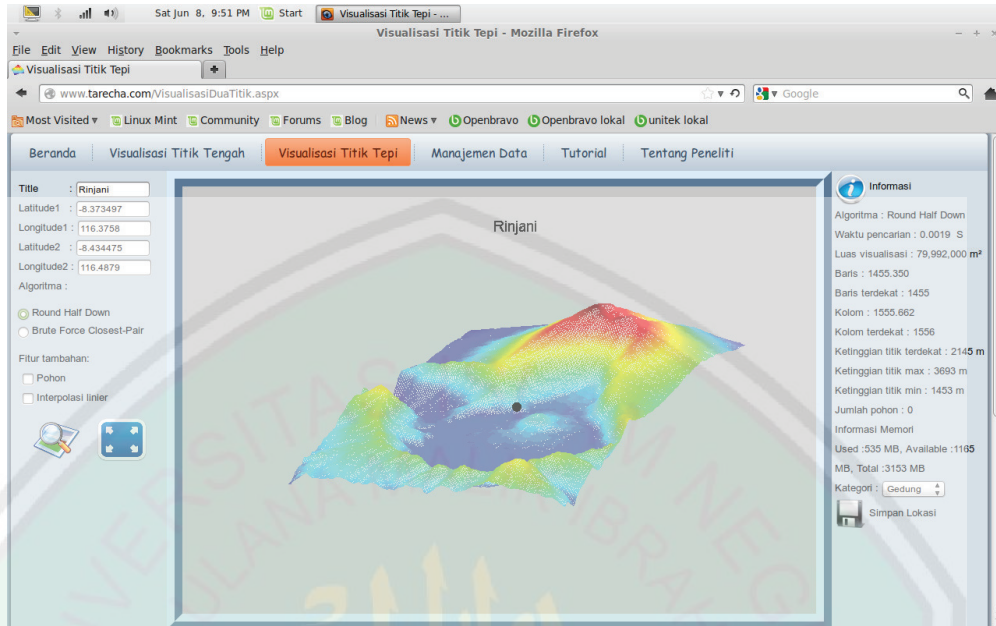
Jumlah pohon yang divisualisasikan diatas kontur tanah sebanyak empat buah namun terdapat perbedaan gambar yang dihasilkan, Gambar 4.9 visualisasi pohon tanpa interpolasi terlihat hanya tiga buah pohon saja, hal ini disebabkan karena jarak antar kedua pohon terlalu dekat sehingga ketika dicari titik elevasi terdekat dengan pohon menunjukkan titik yang sama dan gambar pohon tersebut tertumpuk pada titik tersebut. Jumlah pohon yang terlihat pada gambar 4.10 visualisasi pohon dengan interpolasi terlihat sebanyak empat buah dan letak pohon tidak segaris lurus tapi letak tersebut lebih mirip dengan letak yang sebenarnya seperti yang terlihat pada Gambar 4.8. Dari perbedaan gambar yang dihasilkan tersebut dapat disimpulkan bahwa dengan interpolasi ketelitian visualisasi meningkat.

4.2.5 Pengujian Kompatibilitas pada Perangkat

Pengujian ini menguji apakah aplikasi dapat berjalan pada berbagai perangkat yang berbeda, hasil pengujian ditunjukkan pada gambar-gambar berikut.



Gambar 4.11 Pengujian pada Windows 7.



Gambar 4.12 Pengujian pada Linux Mint.



Gambar 4.13 Pengujian pada Tablet Android.



Gambar 4.14 Pengujian pada Smartphone Android.

Dari pengujian yang telah dilakukan dapat disimpulkan bahwa aplikasi visualisasi dapat berjalan di berbagai perangkat bagi yang menggunakan *mouse* ataupun *touch screen*, tentu saja menggunakan *web browser* yang direkomendasikan yaitu *Mozilla Firefox*.

4.3 Pembahasan Pengujian

Setelah didapatkan hasil pengujian kemudian hasil pengujian tersebut dibahas agar didapatkan pengetahuan dan ilmu baru berdasarkan hasil penelitian.

Pembahasan hasil pengujian antara lain sebagai berikut:

1. Pembahasan hasil pengujian ketepatan dan kecepatan pencarian titik tengah

Algoritma *round-half-down* dan *brute force closest-pair* terbukti mampu untuk menemukan pasangan titik terdekat dengan tepat yang sama, namun perbedaan kedua algoritma tersebut adalah waktu proses pencarian dimana algoritma *round-half-down* 250 x lebih cepat daripada algoritma *brute force closest-pair*. Hal ini disebabkan karena algoritma *round-half-down* hanya menyeleksi titik yang sudah diprediksi sebagai pasangan terdekatnya yaitu batas bawah dan batas atas, sedangkan algoritma *brute force closest-pair* menyeleksi seluruh titik yang ada dan menghitung *euclidean distance* satu persatu. Kelemahan algoritma *round-half-down* dalam mencari titik terdekat adalah interval tiap titik tersebut harus = 1, algoritma tersebut tidak dapat digunakan untuk mencari titik terdekat bila interval antar titik tidak beraturan, sedangkan algoritma *brute force closest-pair* menawarkan fleksibilitas dalam pencarian pasangan titik terdekat karena membandingkan *euclidean distance* terhadap seluruh titik yang akan dicari pasangan terdekatnya tanpa terpengaruh berapapun interval antar titiknya.

2. Pembahasan hasil pengujian ketelitian ketinggian titik tengah

Ketinggian yang dihasilkan pada titik yang sama antara aplikasi visualisasi dan Google Earth memiliki tingkat *error* yang kecil yaitu $< 0,5 \%$ bila dibandingkan dengan antara GPS Garmin dan Google Earth sebesar 2,18%.

Hal ini membuktikan bahwa data satelit dalam hal ini *GDEM ASTER* yang digunakan pada aplikasi visualisasi dan *GDEM SRTM* yang digunakan pada Google Earth memiliki perbedaan yang relatif kecil bila dibandingkan menggunakan GPS Garmin. Keuntungan menggunakan data *GDEM ASTER* dibandingkan *GDEM SRTM* adalah resolusi *GDEM ASTER* lebih detail dan cakupan data lebih luas bila dibandingkan dengan *GDEM SRTM*. Beda antara GPS Garmin dan Google Earth cukup signifikan hal ini disebabkan oleh ketelitian GPS Garmin tersebut, jangankan dengan Google Earth dengan GPS Android pun memiliki perbedaan ketinggian meski pada titik yang sama, hal ini dipengaruhi kualitas perangkat tersebut namun hal ini masih bisa dianggap wajar karena tiap perangkat pasti memiliki toleransi dalam pengukuran.

3. Pembahasan hasil pengujian ketepatan penyusunan empat ubin data terdekat

Metode *forward chaining* terbukti mampu untuk menyusun empat ubin data terdekat dengan cara yang sederhana, hal ini dapat dilakukan karena format penamaan fail data *GDEM ASTER* sudah disusun sedemikian rupa sehingga mudah untuk menemukan fail terdekatnya berdasarkan nama fail. Wilayah visualisasi yang tersebar pada fail data yang berbeda mampu untuk disusun dengan tepat kemudian divisualisasikan.

4. Pembahasan hasil pengujian interpolasi linier

Interpolasi linier terbukti mampu untuk meningkatkan ketelitian visualisasi, semakin detail interpolasi maka ketelitian penempatan objek pohon akan semakin sesuai dengan lokasi aslinya. Semakin banyak titik baru yang dihasilkan interpolasi maka data yang akan diproses semakin besar, hal ini dapat menyebabkan proses untuk membangkitkan visual semakin lama. Nilai titik baru hasil interpolasi adalah hasil perhitungan matematis, bukan pengukuran sebenarnya dari ketinggian tanah, oleh karena itu interpolasi dapat digunakan seperlunya dengan mempertimbangkan besarnya data yang akan diproses dan sumber daya yang tersedia.

5. Pembahasan hasil pengujian kompatibilitas pada perangkat

Aplikasi berhasil dijalankan pada berbagai perangkat yang berbeda menggunakan *web browser* yang direkomendasikan yaitu *Mozilla Firefox*. Hasil yang didapat menggunakan *web browser* tersebut sama pada berbagai perangkat, namun hasil yang berbeda didapatkan bila menggunakan *web browser* lain seperti tampilan *full screen* yang berbeda-beda tiap *web browser*. Alasan dipilihnya *web browser Mozilla Firefox* karena *web browser* tersebut tersedia dalam berbagai *platform* sehingga pengalaman yang diberikan kepada pengguna cenderung sama.

4.4 Integrasi Visualisasi 3D Rupa Bumi dengan Islam

Seperti yang telah dikemukakan pada latar belakang, penelitian ini ingin mengungkap apa pelajaran dari bumi yang dihamparkan ?, petunjuk apa yang terkandung di dalamnya ? melalui visualisasi 3D rupa bumi ini. Pembahasan mengenai integrasi visualisasi 3D rupa bumi dengan Islam penulis kutip dari sebuah artikel dengan penulis anonim (<http://www.islamquest.net/id/archive/question/fa4232>) mengenai Bagaimana pandangan al-Qur'an sehubungan dengan konsep yang menyatakan bahwa bumi itu bulat? yang juga disadur dari beberapa kitab tafsir sebagai berikut.

Dalam Al-Qur'an sudah Al-Hijr ayat 19 berikut ini

وَالْأَرْضَ مَدَدْنَاهَا وَأَلْقَيْنَا فِيهَا رَوَاسِيَ وَأَنْبَتْنَا فِيهَا مِنْ كُلِّ شَيْءٍ مَّوْزُونٍ

Artinya :

Dan Kami telah menghamparkan bumi dan menjadikan padanya gunung-gunung dan Kami tumbuhkan padanya segala sesuatu menurut ukuran.

“*madd al-ardh*” bermakna penghamparan panjang dan lebarnya. Dan sekiranya Tuhan tidak menghamparkan bumi dan bumi dijejali dengan deretan gunung-gunung maka bumi tidak pantas untuk dihuni, tidak dapat dijadikan tempat untuk bercocok tanam dan makhluk hidup tidak akan mendapatkan kesempurnaan hidup di dalamnya. Yang dimaksud dengan “*madda*” dan menghampar tidak bermaksud bahwa bumi tidak berbentuk bulat melainkan maksudnya adalah hamparan bumi tidak diciptakan menonjol secara utuh dan berbentuk rata. Apabila bumi diciptakan menonjol dan berbentuk rata (tidak bulat) maka manusia

tidak dapat hidup dengan baik di muka bumi, mengadakan perkebunan dan pertanian. Pendeknya bumi tidak akan menjadi tempat yang menyenangkan bagi manusia sebagaimana irama ayat-ayat setelahnya berada pada tataran menghitung pelbagai karunia yang menyenangkan bagi manusia yang juga merupakan penegas masalah ini.

Dalam Al-Qur'an sudah An-Naba' ayat 6-7 berikut ini:

أَلَمْ نَجْعَلِ الْأَرْضَ مِهْدًا ﴿٦﴾ وَالْجِبَالَ أَوْتَادًا ﴿٧﴾

Artinya :

Bukankah Kami telah menjadikan bumi itu sebagai hamparan?, dan gunung-gunung sebagai pasak?

Redaksi “*mihâd*” pada ayat ini bermakna tempat yang siap pakai, menghampar dan teratur. Derivatnya dari kata “*ma-h-d*” yang bermakna sebuah tempat untuk istirahat anak kecil (buaian atau tempat tidur). Karena itu, buaian yang disiapkan untuk anak kecil disebut sebagai *mahd*. Dengan demikian, *mahd* dan *mihad* adalah sebuah tempat yang seperti buaian yang siap untuk membina dan membangun sebagaimana yang diungkapkan dalam Al-Qur'an surah Thaha ayat 53 berikut ini:

الَّذِي جَعَلَ لَكُمُ الْأَرْضَ مِهْدًا وَسَلَكَ لَكُمْ فِيهَا سُبُلًا وَأَنْزَلَ مِنَ السَّمَاءِ

مَاءً فَأَخْرَجْنَا بِهِ أَزْوَاجًا مِّنْ نَّبَاتٍ شَتَّى ﴿٥٣﴾

Artinya :

Yang telah menjadikan bagimu bumi sebagai hamparan dan Yang telah menjadikan bagimu di bumi itu jalan-jalan, dan menurunkan dari langit air hujan. Maka Kami tumbuhkan dengan air hujan itu berjenis-jenis dari tumbuh-tumbuhan yang bermacam-macam.

SubhanaAllah setelah mengkaji dalam Al-Qur'an bagaimana bumi dihamparkan dan setelah melihat hasil visualisasi 3D hamparan bumi dapat diketahui bahwa hamparan bumi yang luas itu adalah karunia Allah SWT yang menyenangkan bagi manusia. Hamparan bumi tersebut dapat diamati melalui visualisasi 3D rupa bumi, saat pengambilan data ketinggian pada Gunung Panderman yang telah dilakukan penelenti aplikasi ini mampu untuk memberikan visual Gunung Panderman dengan baik, detail kontur Gunung Panderman terlihat dengan jelas. Diharapkan aplikasi ini dapat dimanfaatkan oleh pihak-pihak yang membutuhkan informasi visual 3D rupa bumi untuk berbagai kepentingan seperti yang telah dipaparkan pada latar belakang.

BAB V

PENUTUP

5.1 Kesimpulan

Kesimpulan yang diperoleh dari penelitian visualisasi 3D rupa bumi berbasis data *GDEM ASTER* adalah sebagai berikut:

1. Algoritma *round-half-down* dan *brute force closest-pair* dapat menemukan pasangan titik terdekat dengan tepat. Algoritma *round-half-down* 250 x lebih cepat daripada algoritma *brute force closest-pair* dalam mencari pasangan titik terdekat. Algoritma *round-half-down* hanya dapat digunakan untuk mencari titik terdekat jika interval masing-masing titik sebesar 1.
2. Metode *euclidean distance* dan *forward chaining* dapat digunakan untuk mendukung visualisasi dan memberikan hasil yang sesuai.
3. Perbedaan ketelitian elevasi tanah antara Google Earth yang menggunakan data *GDEM SRTM* 90 meter dan aplikasi visualisasi yang menggunakan data *GDEM ASTER* 30 meter memiliki perbedaan yang sangat kecil yaitu $< 0,5 \%$ sehingga dapat disimpulkan bahwa aplikasi visualisasi lebih akurat bila dibandingkan dengan GPS Garmin 80 Mil sebesar 2.18 %.
4. Interpolasi terbukti mampu meningkatkan ketelitian visualisasi dibuktikan dengan peletakan objek berupa pohon. Dengan menggunakan antarmuka berupa halaman *web* dan dapat diakses melalui internet maka aplikasi visualisasi dapat digunakan secara luas dan relatif tanpa terkendala pada berbagai perangkat.

5.2 Saran

Dalam penelitian ini masih terdapat beberapa kekurangan. Untuk pengembangan lebih lanjut terdapat saran-saran sebagai berikut ini.

1. Metode interpolasi dapat diperbanyak lagi seperti interpolasi *Nearest Neighbor*, interpolasi *Cubic Spline*, interpolasi kuadrat, dan lain sebagainya.
2. Visualisasi objek dapat diperbanyak lagi seperti visualisasi gedung, jalan, sungai, dan lain sebagainya.
3. Penelitian selanjutnya dapat ditambahkan simulasi bencana yang berkaitan dengan rupa bumi seperti gunung meletus, wilayah kebakaran hutan, dan lain sebagainya.

DAFTAR PUSTAKA

- Al – Qur'an Digital versi 2.1 (<http://www.alquran-digital.com>)
- Anonim. 2009. *Understanding Raster Data*.
New York: ESRI.
- Anonim. 2011. *ASTER GDEM 2 Readme*.
United States of America: LP DAAC.
- Crysdian, Cahyo. 2009. *3D Visualization of Spatial Objects using Elevation Model*. Surabaya: seminar Intelligent Technology and It's Applications.
- Crysdian, Cahyo. 2011. *Development of 3D City Model using Polygonal Modeling Approach*. Yogyakarta : Conference on Information and Electrical Engineering (CITEE).
- Jacobsen, K. 2003. *DEM generation from satellite data*.
Germany: University of Hannover.
- Levitin, Anany. 2012. *Introduction to The Design & Analysis of Algorithms 3rd Edition*. New Jersey: Pearson.
- Maxfield, Clive. 2006. *An introduction to differencing algorithms*.
(<http://www.eetimes.com/design/signal-processing-dsp/4004643/An-introduction-to-differencing-algorithms>, diunduh pada tanggal 28 Februari 2013)
- Presiden Republik Indonesia. 2007. *Peraturan Presiden Republik Indonesia Nomor 85 Tahun 2007 Tentang Jaringan Data Spasial Nasional*. Jakarta : Deputi Sekertaris Kabinet.
- Presiden Republik Indonesia. 2012. *Instruksi Presiden Republik Indonesia Nomor 6 Tahun 2012 Tentang Penyediaan, Penggunaan, Pengendalian Kualitas, Pengolahan Dan Distribusi Data Satelit Penginderaan Jauh Resolusi Tinggi*. Jakarta : Deputi Sekertaris Kabinet.
- Pujiyanta, Ardi. 2007. *Komputasi Numerik dengan Matlab*.
Yogyakarta: Graha Ilmu.
- Russell, Stuart J. dan Peter Norvig. 2003. *Artificial Intelligence A Modern Approach Second Edition*. New Jersey: Pearson.

http://en.wikipedia.org/wiki/Color_mapping (diunduh pada tanggal 21 Januari 2013).

<http://www.islamquest.net/id/archive/question/fa4232> (diunduh pada tanggal 9 Juni 2013).

<http://arduino.cc/en/Reference/map> (diunduh pada tanggal 4 Juni 2013).

http://en.wikipedia.org/wiki/Wire-frame_model (diunduh pada tanggal 20 April 2013)

http://en.wikipedia.org/wiki/Advanced_Spaceborne_Thermal_Emission_and_Reflection_Radiometer (diunduh pada tanggal 21 Januari 2013).

http://en.wikipedia.org/wiki/Shuttle_Radar_Topography_Mission (diunduh pada tanggal 13 Januari 2013).

https://en.wikipedia.org/wiki/Google_Earth (diunduh pada tanggal 13 Januari 2013).

<http://www.beritabogor.com/2012/05/sukhoi-superjet-100-jatuh-di-gunung.html> (diunduh pada tanggal 12 Agustus 2012).



LAMPIRAN

Foto akuisisi data ketinggian di Gunung Panderman





Video hasil penelitian dapat dilihat pada link berikut
<http://www.youtube.com/watch?v=y0bAyARm3G8>

