

# **SIMULATOR VISUAL 2 DIMENSI PREDIKSI LINTASAN SATELIT ORBIT RENDAH BERBASIS MODEL SGP4 DALAM PROYEKSI MERCATOR**

## **(LOW ORBIT SATELLITE TRACK PREDICTION VISUAL 2 DIMENSION SIMULATOR BASE ON SGP4 IN MERCATOR PROJECTION)**

**Bina Pratomo<sup>1</sup>, Hasan Mayditia**

Pusat Teknologi Satelit, Lembaga Penerbangan dan Antariksa Nasional (LAPAN)

<sup>1</sup>e-mail: [bina.pratomo@lapan.go.id](mailto:bina.pratomo@lapan.go.id)

Diterima : 03 Juli 2019; Direvisi : 20 Agustus 2019; Disetujui : 30 September 2019

### **ABSTRAK**

*Software* prediksi posisi lintasan satelit ada yang gratis, berbayar, dan berlisensi serta ada yang mengembangkannya menggunakan *software* khusus simulasi satelit dengan kebutuhan spesifikasi komputer cukup besar. Dalam karya tulis ini *file* DJTSatLib.dll yang berisi algoritma model SGP4 dan *file* gdi32.dll yang berisi proses pengolah *graphic* teknik proyeksi Mercator dapat memvisualisasikan prediksi lintasan satelit dalam tampilan 2 dimensi dengan 100 titik posisi dalam 100 menit sebelum waktu lintasan jangkauan stasiun bumi. Hasilnya dapat dijalankan menggunakan komputer berspesifikasi relatif rendah dengan 15 MB pemakaian RAM dan 8% pemakaian CPU.

Kata kunci : *Satelit, SGP4, Mercator, Visualiasi*

### **ABSTRACT**

There are satellite position prediction *softwares* free, paid, and licensed and some peoples develop it by using special satellite simulation *software* that need quite high specification. This paper by using DJTSatLib.dll *file* contain SGP4 model's algorithm and gdi32.dll *graphic* processor *file* for Mercator projection technique able to visualize satellite track prediction in 2 dimension with 100 point position in last 100 minute from in coverage *ground station* pass ing time. The results can run using low spec computer with 15 MB of RAM usage and 8% CPU usage.

Keywords : *Satellite, SGP4, Mercator, Visualization*

## 1 Pendahuluan

Pertumbuhan akan pembuatan satelit di dunia meningkat setiap tahunnya sebesar 47% dengan didominasi satelit (<1200 Kg) pada ketinggian LEO (Bryce, 2017).

Untuk dapat mengakses satelit LEO diperlukan suatu alat yang dapat memprediksi posisi satelit pada waktu tertentu, sehingga perangkat pengakses satelit dapat diarahkan ke posisi satelit guna mendapatkan sinyal yang kuat dan data yang akurat.

Alat yang diperlukan untuk mengetahui posisi satelit saat ini yang banyak digunakan adalah *software* prediksi posisi satelit, dengan *software* ini bisa didapatkan informasi posisi satelit saat ini hingga beberapa waktu kedepan.

*Software* tersebut ada yang berbasis *web* seperti N2YO dengan alamat URL <https://www.n2yo.com>, *WEB-GIS Based Satellite tracking* yang dikembangkan dengan memanfaatkan *database* sebagai penyimpanan datanya (Maiti, 2018), *SatView* (<http://www.satView.org/>).

(Madina, 2015) menggunakan *software* berlisensi yang berbasis windows yaitu *Satellite Tool Kit* (STK). (Alam, 2018) menggunakan *software* orbitron untuk pengembangan *Automated satellite tracking system* untuk antena *directional*.

(Hao, 2012) menggunakan *application peripheral interface* (API) ArcGis dan OpenGL memvisualisasikan posisi satelit secara 2 dimensi dan 3 dimensi yang memerlukan proses *render* yang cukup memakan sumberdaya komputer serta penerapannya dapat digunakan di bahasa pemrograman .NET, Java, C++.

(Mahdi, 2015) menggunakan *software open source* GMAT (*General mission Analysis Tool*) yang dikembangkan tim pengembang NASA

untuk memvisualisasikan *ground track* satelit dalam bentuk 2 dimensi dan 3 dimensi, proses visualisasi hanya bisa dijalankan didalam *software* GMAT.

*Software* berbasis *web* dan windows, serta beberapa karya tulis di atas tidak menjelaskan proses visualisasi posisi dan lintasan satelit tampil sedemikian rupa, tidak menjelaskan teknik dan cara membuatnya sehingga tidak bisa dimodifikasi untuk keperluan lain yang berkaitan dengan data dan informasi yang disajikan. Beberapa diantaranya membutuhkan spesifikasi komputer yang relatif tinggi dalam menampilkan hasil visualisasi.

Tujuan penelitian ini adalah ingin mengetahui penggabungan penggunaan *file* DJTSatLib.dll dan gdi32.dll untuk dapat menghasilkan aplikasi yang mampu memvisualisasikan prediksi posisi dan lintasan satelit.

Karya tulis ini menjelaskan penggunaan permodelan SGP4 (*Simplified General Perturbations Satellite Orbit Model 4*) yang dikemas dalam sebuah *file dynamic link library* (DLL). *File* DLL digunakan di berbagai bahasa pemrograman berbasis windows, di dalamnya terdapat subrutin dan fungsi yang dapat dipanggil untuk memproses data orbit satelit.

*File* DJTSatLib.dll dapat diunduh di situs <https://www.Satsignal.eu/software/wxtrack.htm> (diakses Juli 2019). *File* ini merupakan salahsatu *file* yang digunakan di dalam *software* WxTrack, yaitu suatu *tool* tak berbayar yang banyak digunakan oleh pengguna amatir dalam mengobservasi posisi *longitude*, *latitude*, *altitude*, dan minimum elevasi suatu objek satelit (Pisanu, 2017).

Teknik Mercator memanfaatkan *file* API gdi32.dll yang merupakan *file* bawaan microsoft windows. Teknik ini digunakan

untuk memproyeksikan gambar peta dunia serta koordinat *longitude latitude* ke dalam koordinat x, y dalam peta.

*Tools* bahasa pemrograman yang digunakan adalah visual basic 6 yang sampai saat ini masih digunakan, (Prabowo, 2019) menggunakan visual basic 6 untuk menganalisis daya dukung pondasi tiang dengan cukup menggunakan komputer intel *dualcore* 2.6GHz, RAM 2 GB, dan *Hardisk* 80 GB. Dengan menggunakan visual basic 6, pengembangan sistem visualisasi prediksi posisi dan lintasan satelit ini berpeluang dapat dimodifikasi untuk ditingkatkan ke pemanfaatan lainnya, salah satunya sistem pengendali sudut azhimut dan elevasi antenna untuk penjejak (*tracking*) satelit yang ada di referensi (Rahman, 2008).

Hasil *software simulator* ini diharapkan dapat dijalankan di komputer yang berspesifikasi relatif rendah sehingga bisa digunakan dengan biaya relatif murah.

### **Two Line Element**

Parameter masukan yang digunakan untuk mengaktifkan sistem visualisasi ini adalah data TLE (*Two Line Element*), parameter ini merupakan *element kaplerian orbital* pada titik tertentu terhadap waktu dari setiap objek di ruang angkasa (Riesing, 2015). TLE berbentuk untaian data teks yang terdiri atas dua baris mengandung data orbital dan 1 baris nama satelit. *Format* dari TLE ini tampak pada gambar 1-1 dengan rincian data sebagai berikut :

- Nama satelit (*Name of Satellite*) maksimal 11 karakter di gambar 1-1 dinyatakan dengan nama satelit NOAA 6
- Nomor satelit (*Satellite Number*) terdiri 5 digit nomor yang didaftarkan pada katalog USS STRATCOM
- *International Designator* pada data ini 2 digit pertama menandakan tahun peluncuran, di gambar dinyatakan dengan angka 84 yang berarti diluncurkan tahun 1984, sisa nomor lainnya menunjukkan jumlah objek yang telah diluncurkan sebelumnya pada tahun yang sama.
- Inklinasi (*inclination*) merupakan besar sudut dalam satuan derajat antara garis katulistiwa dan jalur orbit satelit tersebut (*orbital plane*).
- *Epoch Year and Julian Day Fraction* (JDF) adalah tanggal disaat data TLE ini diambil (the astronomical epoch). 2 digit pertama menandakan tahun, dan sisa nomor lainnya menandakan jumlah hari terhitung sejak 1 Januari tahun saat data TLE ini diterbitkan.
- *Right Ascension of the Ascending Node* (RAAN) merupakan besar sudut dalam satuan derajat antara *spring equinox* dan titik dimana orbit objek bersilang dengan jalur *equatorial*.
- *Eccentricity* merupakan nilai konstan antara 0 dan 1 yang menentukan bentuk orbit objek, nilai 0 menunjukkan *elliptical orbit* sedangkan nilai 1 menunjukkan *circular orbit*.
- *1<sup>st</sup> derivative of Mean Motion* (*Ballistic Coefficient*) adalah fraksi harian atas perubahan dalam jumlah revolusi yang dibuat objek. Di ekspresikan dalam bentuk *revolutions/day*.
- *Argument of Perigee* adalah besar sudut antara RAAN dan *perigee orbit objek*
- *2<sup>nd</sup> derivative of Mean Motion* merupakan *derivative* ke dua dari **daily mean motion object**,

- diekspresikan dalam bentuk  $revolutions/day^3$ .
- *Mean Anomaly* adalah nilai sudut yang diukur dari perigee dan diekspresikan dalam bentuk derajat dari posisi objek di orbit, didasarkan pada *circular orbit radius equal to the semi-major axis*.
- *Drag Term (Radiation Pressure Coefficient)* didefinisikan sebagai penarikan atmosfer pada pergerakan satelit, diekspresikan dalam bentuk  $Earth\ radius^{-1}$ .
- *Ephemeris Type* jumlah yang digunakan hanya di internal, biasanya bernilai nol. Ini merepresentasikan model orbit yang digunakan untuk menghasilkan data.
- *Element Number & Check Sum*, tiga digit pertama merepresentasikan jumlah TLE yang dihasilkan untuk objek bersangkutan oleh US STRATCOM. Counter meningkat hingga 999 dan akan diset 0 untuk selanjutnya, digit terakhir merupakan nilai *checksum* baris pertama

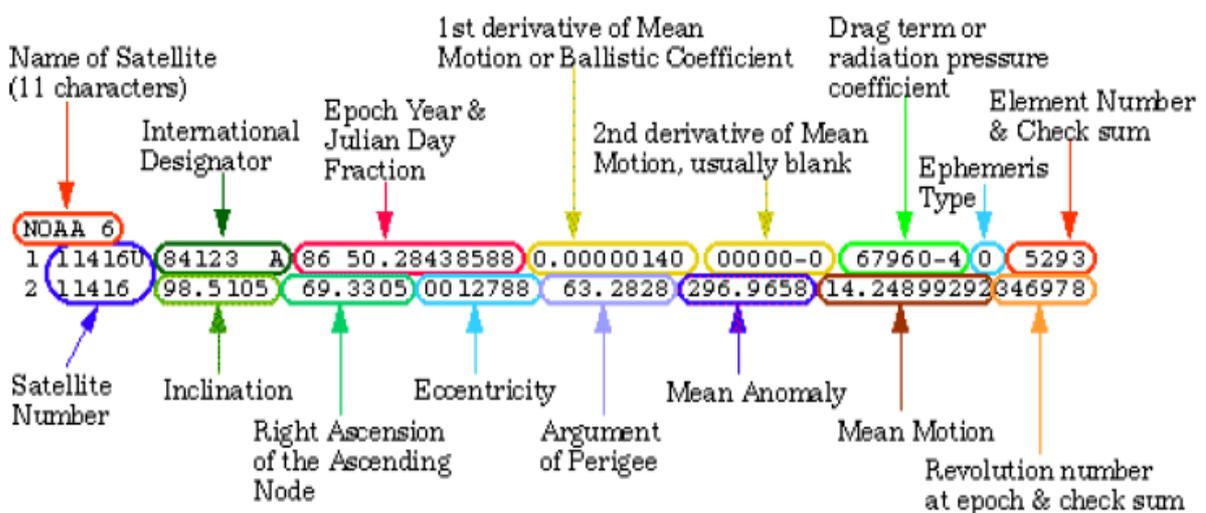
- *Revolution Number at Epoch & Check Sum* jumlah revolusi dari objek pada epoch tertentu. Digit terakhir merupakan nilai checksum baris ke 2.

Data TLE bisa didapat di situs [www.space-track.org](http://www.space-track.org). pada situs tersebut terdapat lebih dari 18.000 objek yang dapat diakses publik, dan diperbaharui perhari bahkan lebih sering lagi (Chen, 2017). Situs lainnya terdapat di <https://celestrak.com>.

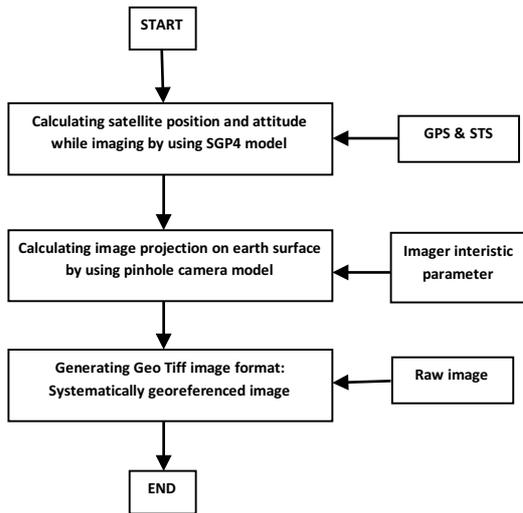
## 2 METODOLOGI

Penelitian ini menggunakan model SGP4, parameter masukannya berupa data TLE sebuah satelit orbit rendah dan data waktu yang diinginkan. Model SGP4 akan menghasilkan keluaran posisi satelit yaitu *longitude*, *latitude*, dan *altitude*.

Model SGP4 digunakan pada sebuah penelitian di pusat teknologi satelit, digunakan dalam pemrosesan multispectral images satelit LAPAN-A3/IPB (Hakim, 2018). Gambar 2-1 menampilkan proses penggunaan model SGP4 pada penelitian tersebut.



Gambar 1-1. Contoh dan *Format* teks data TLE (Muntoni, 2019)

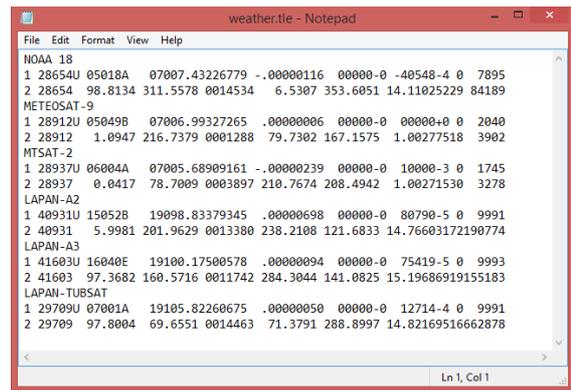


Gambar 2-1: Penggunaan algoritma SGP4 untuk pemrosesan multispectral images

Data posisi dari permodelan SGP4 ditambah dengan data posisi stasiun bumi (*ground station*) dimasukkan ke proses berikutnya untuk mendapatkan daftar estimasi waktu lintasan. Data estimasi berisi daftar waktu dan sudut elevasi satelit saat melintas disuatu daerah yang tergapai oleh *ground station*, terhitung dari mulai waktu yang diinginkan hingga pukul 23:59.

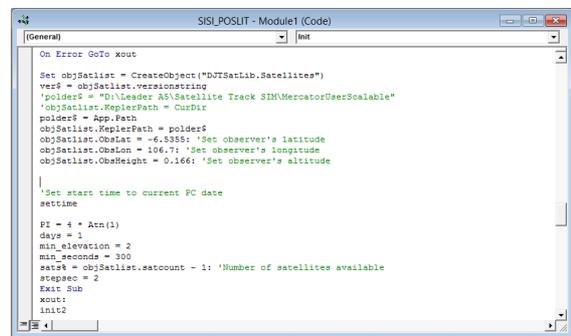
Daftar waktu lintasan digunakan sebagai acuan untuk mendapatkan perhitungan posisi satelit sebelum dan sesudah melintasi *ground station* pada satu periode waktu yang diinginkan. Data ini dapat membantu penggiat satelit untuk mempersiapkan peralatan akusisi data satelit. Waktu yang digunakan dalam aplikasi ini menggunakan Pewaktuan yang terdapat dalam sistem komputer.

Pemasukan data TLE pada penelitian ini ditempatkan dalam sebuah *file* teks yang diakses aplikasi dan pemutakhirannya dilakukan secara manual dengan mengganti baris data TLE yang diinginkan dengan yang baru, data tersebut berasal dari situs <https://celestrak.com>. Tampilan *file* teks tersebut tampak pada gambar 2-2.



Gambar 2-2: File teks berisi data TLE

*File* teks yang berisi data TLE tersebut diakses aplikasi yang dikembangkan dengan menggunakan bahasa pemrograman visual basic 6 melalui sebuah subrutin *script* *ysng* terdapat DJTSatLib.dll. Subrutin *script* ini mengekstrak data TLE sehingga didapat daftar satelit beserta parameter keplerian satelit tersebut. Gambar 2-3 menampilkan penggunaan subrutin inisialisasi dari *file* DJTSatLib.dll. Variabel global bertipe *object* dalam hal ini *objSatlist* di atur menjadi *object* DJTSatLib.Satellites, dengan penganuran ini seluruh fungsi dan atributnya dapat digunakan. Atribut KeplerPath diisi dengan nama folder dimana *file* teks TLE berada, sedangkan atribut *ObsLat*, *ObsLon*, dan *ObsHeight* merupakan attribute posisi *ground station*.



Gambar 2-3 Subrutin inisialisasi objek DJTSatLib

Setelah proses inialisasi dijalankan semua fungsi yang ada di dalam DJTSatLib dapat diakses. Hal yang pertama di proses adalah menampilkan daftar satelit yang terdapat dalam file teks TLE ke dalam *combo box*, sehingga pengguna dapat memilih satelit yang akan di visualisasikan posisinya.

Proses penampilan daftar waktu lintasan yang akan melintas di daerah *ground station* dilaksanakan setelah memilih sebuah nama satelit di dalam *combo box*, daftar waktu yang dihasilkan terhitung mulai waktu terkini sistem komputer hingga pukul 23:59 pada hari yang sama. *Event double click* pada daftar lintasan menghasilkan daftar posisi dan waktu 100 menit sebelum waktu lintasan yang dipilih, pada proses inilah fungsi algoritma model SGP4 digunakan.

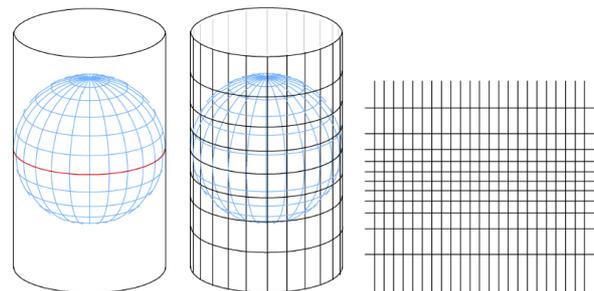
Pengambilan data 100 menit sebelum waktu lintasan dilakukan dengan *interval* 1 menit, sehingga akan didapat 100 data waktu dan posisi, data ini disimpan dalam daftar posisi satelit yang kemudian diproses untuk divisualisasikan dalam bentuk titik dan garis lintasan.

Ketika aplikasi dijalankan selain pengaturan *variable* objek sebagai objek DJTSatLib, rutin penampilan gambar peta dunia dengan teknik mercator projection juga dijalankan, sehingga sebelum daftar lintasan satelit yang dipilih belum tampil, gambar peta dunia sudah muncul di tampilan layar.

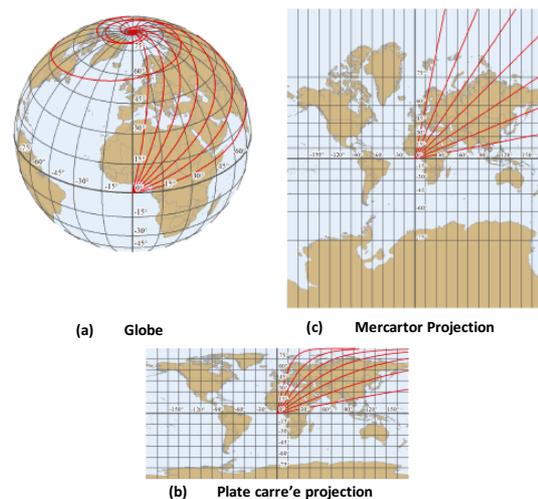
Mercator projection merupakan suatu teknik proyeksi yang sudah banyak diketahui dan digunakan dalam pemetaan seperti atlas dan *wall map*, dengan teknik ini bumi diproyeksikan dalam sebuah tabung dengan tanpa tutup yang kemudian di rentangkan dalam bentuk lembar datar seperti tampak pada gambar 2-4.

Lembar datar tersebut dibuatkan *grid* untuk memproyeksikan garis

*latitude* dan *longitude* dengan bereferensi garis titik (0,0) pada tabung disesuaikan dengan titik *latitude* 0 dan *longitude* 0, titik pertemuan ujung lembaran bersumbu pada sumbu x. Titik-titik koordinat pada sumbu x sesuai dengan titik-titik *longitude* pada *equator* dari bola bumi sehingga nilainya mulai dari -180° hingga 180°, dimana 180° dan -180° berhimpit seperti tampak pada gambar 2-5, sedangkan sumbu y memproyeksikan *latitude*.



Gambar 2-4: Tampilan proyeksi bumi ke bentuk lembar datar (Vis, 2018)



Gambar 2-5 Proyeksi koordinat (a) tampilan pada *globe*, (b) tampilan pada Mercator projection, (c) tampilan pada Plate carrée projection (Vis, 2018)

Dengan memproyeksikan sumbu x sebagai *longitude* dan sumbu y sebagai *latitude*, nilai – nilai posisi satelit yang dihasilkan SGP4 dapat disesuaikan letaknya melalui pemrograman visual

basic 6. untuk mendapatkan visualisasi koordinat *longitude* dan *latitude* sesuai dengan gambar peta dunia diperlukan langkah langkah yang terdiri dari beberapa proses seperti:

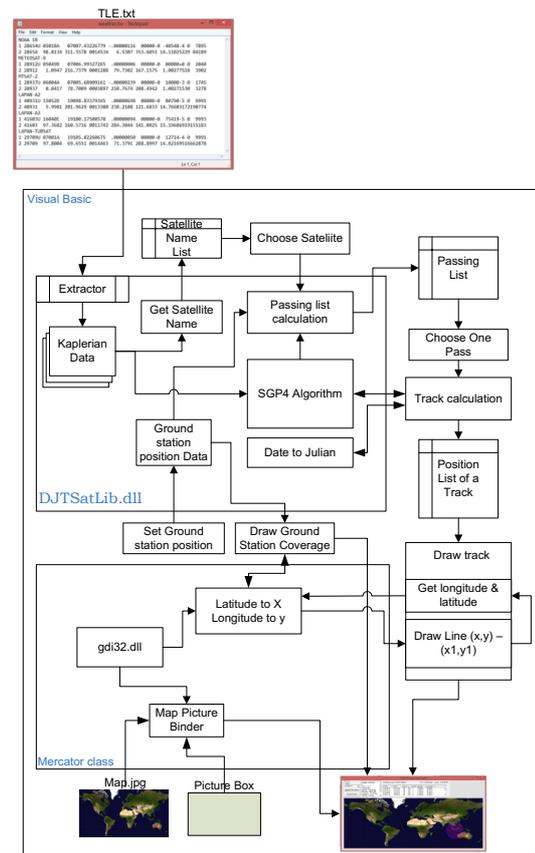
- Memproses gambar peta dunia 2 dimensi pada tampilan aplikasi
- Memproyeksikan *grid longitude* pada sumbu x gambar peta dunia
- Memproyeksikan *grid latitude* pada sumbu y gambar peta dunia

Visualisasi yang digunakan pada teknik mercator di komputer ber sistem operasi windows 32 bit ini dapat menggunakan *file graphic device interface (GDI) gdi32.dll*. *File gdi32.dll* memiliki banyak fungsi yang dapat mengolah gambar untuk ditampilkan di layar monitor. Salah satu fungsi dari *gdi32* adalah proses penyematan *file* gambar peta dunia ke komponen *picture box* visual basic secara penuh sesuai dengan ukuran *picture box* dan resolusi monitor pada Komputer. Dengan demikian didapatkan rasio *pixel* gambar sehingga satu *pixel* pada monitor merpresentasikan koordinat *longitude* dan *latitude*.

Posisi *pixel* dinyatakan dengan nilai x dan y, dengan fungsi khusus nilai x dan y dapat dikonversi ke nilai *longitude* dan *latitude*, demikian pula nilai *longitude* dan *latitude* dapat dikonversikan ke nilai x dan y yang merupakan koordinat (x,y) pada gambar peta dunia.

Fungsi konversi *longitude* dan *latitude* ke dalam nilai x dan y digunakan untuk mengubah nilai *longitude* dan *latitude* yang ada pada daftar posisi satelit dalam satu lintasan ke dalam nilai koordinat x dan y. Menghubungkan antar titik tersebut dengan garis dapat divisualisasikan menjadi garis lintasan satelit.

Alur metode di atas direpresentasikan kedalam diagram alur metode pada gambar 2-6 berikut.



Gambar 2-6: Blok Diagram alur metode yang digunakan

### 3 HASIL PEMBAHASAN

Metode yang dijabarkan di atas diimplementasikan ke dalam *software* visual basic 6, dengan memanfaatkan *file DJTSatLib.dll* untuk pengolahan data algoritma model SGP4 dan *file gdi32.dll* untuk pengolahan teknik mercator. Dari hasil pengembangan didapat empat buah *file* seperti yang tampak pada gambar 3-1 yaitu :

1. *Form1.frm*  
Berisi layout tampilan GUI (*Graphic User Interface*) yang merupakan *interface* untuk memudahkan pengguna berinteraksi dengan aplikasi
2. *djt.bas*  
Merupakan *file module* yang berisi *variable* dan subrutin pengolahan data algoritma model SGP4.
3. *cLabel.cls*

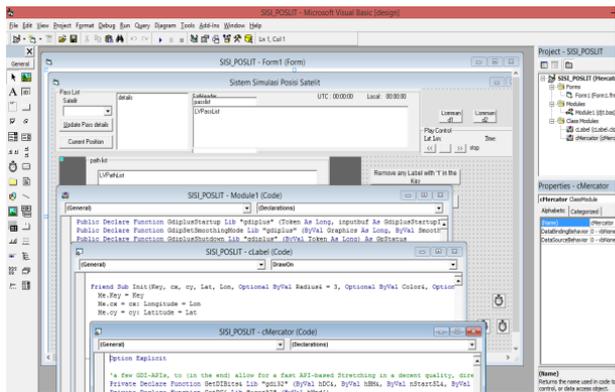
merupakan *file class* yang berisi deklarasi sebuah fungsi *file gdi32.dll* dan beberapa subrutin untuk penyesuaian gambar peta dunia kedalam *form*

4. *cMercator.cls*

merupakan *file class* yang berisi deklarasi beberapa fungsi *file gdi32.dll* dan beberapa subrutin seperti konversi *longitude latitude* ke bentuk koordinat *x* dan *y*.

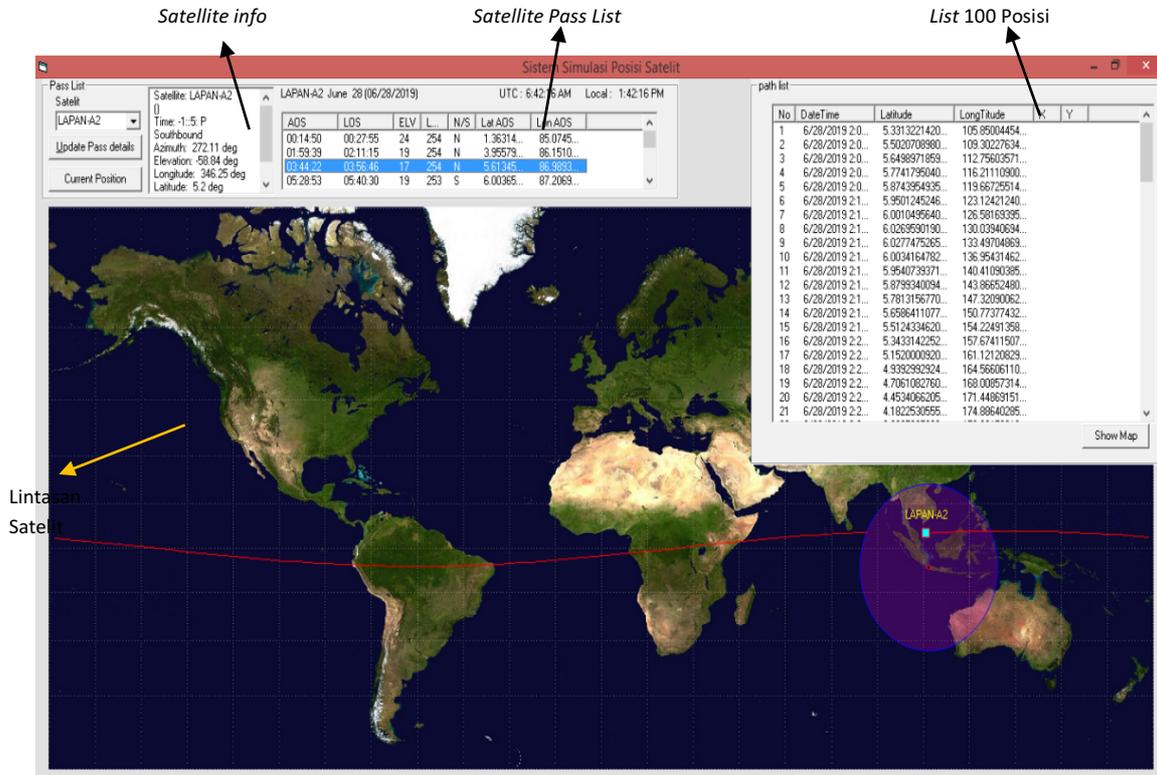
satelit LAPAN-A2 sebagai objeknya. Alur interaksi pengguna dengan aplikasi dijelaskan sebagai berikut :

1. Pilih nama satelit yang akan disimulasikan pada *combo box* satelit.
2. Setelah terpilih maka secara otomatis detil info yang berada pada *list* satelit info akan muncul, begitu juga daftar waktu kedatangan satelit yang terjangkau antenna stasiun bumi akan muncul pada *list* *satellite pass list*.
3. Pilih salah satu waktu kedatangan dengan cara mendouble *click row pass list* yang dikehendaki.
4. Setelah terpilih *pass list* yang dikehendaki, secara otomatis akan muncul garis merah pada gambar peta dunia yang merupakan jalur lintasan satelit dimulai dari posisi satelit 100 menit sebelum waktu kedatangan yang dipilih.



Gambar 3-1: Tampilan *file* yang dihasilkan

Hasil dari eksekusi *simulator* ini tampak pada gambar 3-2 dengan



Gambar 3-2. Tampilan aplikasi visualisasi posisi satelit orbit rendah 2 dimensi

Untuk memastikan kevalidan data posisi satelit yang dihasilkan rutin algoritma model SGP4 dari program yang dibuat ini, penulis melakukan perbandingan data posisi tersebut dengan data posisi yang didapatkan dari *software* STK. *Software* STK biasa digunakan oleh operator satelit di pusat teknologi satelit LAPAN.

Pembandingan dilakukan terhadap daftar 100 posisi satelit pada tanggal 28 Juni 2019 pukul 5:34:20 AM waktu UTC sampai dengan tanggal 28 Juni 2019 pukul 7:13:20 AM waktu UTC, dengan data TLE yang diambil dari situs celestrack pada tanggal 28 juni 2019 pukul 14:03 waktu Indonesia bagian barat. Tampilan hasil algoritma dari program simulasi tampak pada gambar 3-3, sedangkan hasil dari *software* STK tampak pada gambar 3-4.

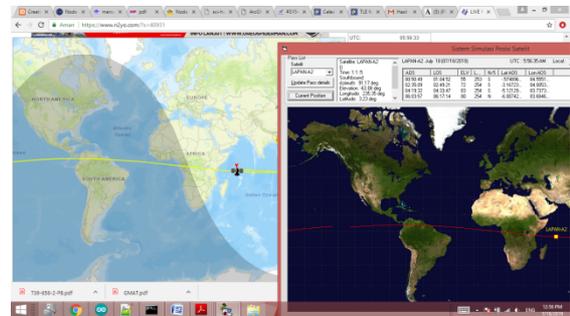
No	DateTime	Latitude	LongTude
1	6/28/2019 5:34:20 AM	5.20384932819521	105.506102939534
2	6/28/2019 5:35:20 AM	5.22509432707299	108.95341560631
3	6/28/2019 5:36:20 AM	5.01904858028493	112.398445400608
4	6/28/2019 5:37:20 AM	4.7921801537188	115.841278440169
5	6/28/2019 5:38:20 AM	4.54544100016095	119.261775949544
6	6/28/2019 5:39:20 AM	4.2796707547054	122.715911658227
7	6/28/2019 5:40:20 AM	3.9965651349717	126.156702821996
8	6/28/2019 5:41:20 AM	3.69671769714123	129.589207244908
9	6/28/2019 5:42:20 AM	3.36157013062461	133.020525110763
10	6/28/2019 5:43:20 AM	3.02423944590906	136.445798522006
11	6/28/2019 5:44:20 AM	2.71065946715761	139.877199691267
12	6/28/2019 5:45:20 AM	2.3576706761244	143.302948299638
13	6/28/2019 5:46:20 AM	1.98491957168102	146.727288309898
14	6/28/2019 5:47:20 AM	1.6239002187743	150.150496543169
15	6/28/2019 5:48:20 AM	1.24613998693379	153.572667332013
16	6/28/2019 5:49:20 AM	863192247428933	156.994721736018
17	6/28/2019 5:50:20 AM	476631515015601	160.416393481411
18	6/28/2019 5:51:20 AM	8.68473178021134E-02	163.838226126914
19	6/28/2019 5:52:20 AM	-3.00361720211312	167.260567765707
20	6/28/2019 5:53:20 AM	-669793629376367	170.683765629075

Gambar 3-3: Tampilan 20 pertama data posisi dari program yang dibuat

Time (UTC)	Sat (deg)	Alt (km)	Lat Rate (deg/min)	Alt Rate (deg/min)
28 Jun 2019 05:34:20.000	5.469	109.164	0.00087	-0.00163
28 Jun 2019 05:35:20.000	5.225	108.953	0.00236	-0.00201
28 Jun 2019 05:36:20.000	5.019	112.399	0.00127	-0.00249
28 Jun 2019 05:37:20.000	4.792	115.841	0.00349	-0.00272
28 Jun 2019 05:38:20.000	4.545	119.261	0.00472	-0.00280
28 Jun 2019 05:39:20.000	4.280	122.710	0.00477	-0.00249
28 Jun 2019 05:40:20.000	3.997	126.156	0.00483	-0.00207
28 Jun 2019 05:41:20.000	3.697	129.589	0.00497	-0.00182
28 Jun 2019 05:42:20.000	3.382	133.020	0.00512	-0.00160
28 Jun 2019 05:43:20.000	3.052	136.450	0.00530	-0.00139
28 Jun 2019 05:44:20.000	2.711	139.877	0.00549	-0.00120
28 Jun 2019 05:45:20.000	2.358	143.303	0.00569	-0.00105
28 Jun 2019 05:46:20.000	1.995	146.727	0.00590	-0.00093
28 Jun 2019 05:47:20.000	1.624	150.150	0.00612	-0.00084
28 Jun 2019 05:48:20.000	1.246	153.573	0.00635	-0.00076
28 Jun 2019 05:49:20.000	0.863	156.995	0.00659	-0.00068
28 Jun 2019 05:50:20.000	0.477	160.416	0.00684	-0.00060
28 Jun 2019 05:51:20.000	-0.089	163.838	0.00709	-0.00052
28 Jun 2019 05:52:20.000	-0.471	167.261	0.00734	-0.00044
28 Jun 2019 05:53:20.000	-0.899	170.684	0.00759	-0.00036
28 Jun 2019 05:54:20.000	-1.374	174.106	0.00784	-0.00028
28 Jun 2019 05:55:20.000	-1.899	177.528	0.00809	-0.00020
28 Jun 2019 05:56:20.000	-2.474	180.950	0.00834	-0.00012
28 Jun 2019 05:57:20.000	-3.099	184.372	0.00859	-0.00004
28 Jun 2019 05:58:20.000	-3.774	187.794	0.00884	-0.00000
28 Jun 2019 05:59:20.000	-4.499	191.216	0.00909	-0.00000
28 Jun 2019 06:00:20.000	-5.274	194.638	0.00934	-0.00000

Gambar 3-4: Tampilan 24 pertama data posisi dari *software* STK

Pada kedua sumber data tersebut tampak nilai *longitude* dan *latitude* sama besarnya hingga 3 digit dibelakang koma. Perbandingan visualisasi lintasan satelit LAPAN-A2 antara *simulator* dan aplikasi *online* n2yo.com pada waktu yang sama tampak pada gambar 3-5.



Gambar 3-5: Perbandingan tampilan lintasan hasil *simulator* dengan n2yo.com pada waktu yang sama.

Tampak pada gambar 3-5 dengan data TLE terkini (diunduh tanggal 18 Juli 2019), satelit LAPAN-A2 sebagai objeknya, dan tanggal simulasi 18 Juli 2019 pukul 5:56:35 AM waktu UTC, *simulator* menghasilkan garis lintasan yang serupa dengan tampilan hasil simulasi pada situs n2yo.com dengan objek dan waktu yang sama.

Satellite *pass list* (daftar waktu satelit melintas di sekitar stasiun bumi) dihasilkan oleh subrutin/fungsi yang memanfaatkan subrutin/fungsi yang berada di dalam *file* DJTSatLib.dll yaitu *Pass List PerPeriod* dengan contoh penggunaan :

```
pass es$ = objSatlist . Pass List ForPeriod (s$,
dt, min_seconds, days,
min_elevation)
```

dengan parameter sebagai berikut:

**s\$**=> Merupakan Nama satelit

**dt**=> Tanggal mulai pencarian *pass list*

**min\_seconds** => durasi minimum yang akan dicari *pass list* nya

**days** => jumlah hari yang akan di *list* sejak tanggal dt

**min\_elevation** => besar sudut elevasi minimal yang hendak di *list*

Hasil pencarian berupa data tipe string (*variable pass es\$*) yang perlu diparsing (dipecah) untuk dimasukkan ke dalam komponen *List View* pada GUI program. Daftar posisi yang ditampilkan di *List View* 100 menit posisi satelit sebelum waktu melintas di area stasiun bumi merupakan hasil dari algoritma SGP4 yang berada di fungsi

**objSatlist .Isvisible(s\$, julian, threshdeg, southbound, az, el, longit, Lat, range, rr, alt, phase)**

dengan parameter sebagai berikut:

**s\$** => Merupakan Nama satelit yang dipilih (*input*)

**julian** => Waktu dalam bentuk *Julian date* yang akan di ketahui posisi satelitnya (*input*)

**threshdeg** => Besar sudut elevasi yang ingin diketahui (*input*)

**southbound** => Indikator posisi berada di selatan *equator* atau tidak (*output*)

**az** => Besar sudut azimuth satelit terhadap stasiun bumi pada posisi yang dihasilkan (*output*)

**el** => Besar sudut elevasi satelit terhadap stasiun bumi pada posisi yang dihasilkan (*output*)

**longit** => nilai posisi longitudinal pada waktu yang dicari (*output*)

**Lat** => nilai posisi *latitude* pada waktu yang dicari (*output*)

**alt** => nilai *altitude* pada waktu yang dicari (*output*)

Dengan 3 parameter *input* didapatkan *output* posisi satelit, nilai *Julian date* merupakan hasil konversi waktu kalender menggunakan fungsi `objSatlist.DateTimetoJT (tgl)` dimana *tgl* berisi tanggal dengan *Format MM/dd/yyyy HH:mm:ss*. Teknik Mercator yang diimplementasikan ke dalam pengembangan program dengan visual basic 6 memanfaatkan *file API gdi32.dll* untuk memproses gambar peta

dunia yang sudah disesuaikan rasio dan skalanya untuk dapat fit dengan *formnya*, dan resolusinya disesuaikan.

Fungsi - fungsi yang digunakan untuk proses tersebut memecah *file gambar* menjadi 32-bit *array* dan menyesuaikan resolusi dengan menampilkan hasil olahan di dalam *array* ke dalam gambar kembali. Dengan demikian setiap *pixel* pada tampilan masih sesuai dengan rasio dan skalanya meskipun *form* diperbesar atau diperkecil ukurannya (*resize*).

Setelah ukuran gambar peta dunia disesuaikan dengan ukuran jendela aplikasi, nilai representasi x dan y sebagai koordinat *longitude* dan *latitude* juga didapatkan. Dengan demikian nilai posisi *longitude* dan *latitude* yang berada pada daftar *List View* 100 menit posisi satelit dapat dikonversikan ke nilai x dan y melalui fungsi :

**x = Mercator.LonToPxIX(nilaiLon)**

dan

**y = Mercator.LatToPxIY(nilaiLat)**

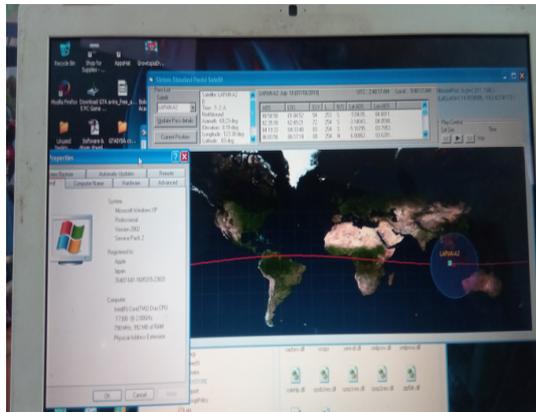
nilai x dan y yang dihasilkan ditampilkan di atas gambar peta dunia dengan

**picMercator.Line (yXa, xXa)-(yXb, xXb), vbRed**

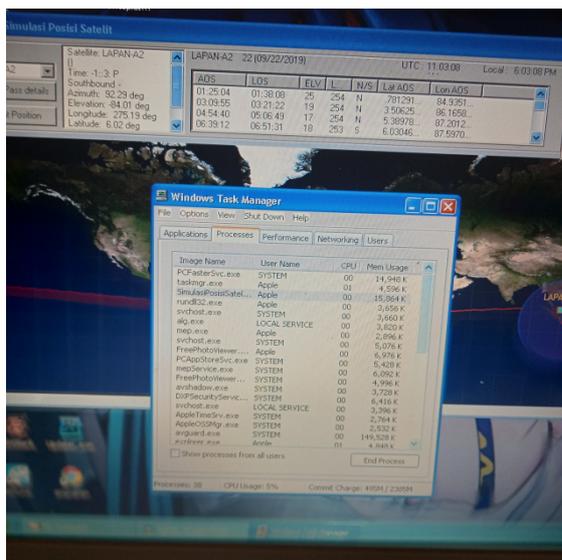
dimana (yXa, xXa) *variable* berisi nilai x dan y awal garis dan (yXb, xXb) *variable* yang berisi nilai x dan y akhir garis. *vbRed* merupakan nilai konstanta warna merah untuk memberikan mewarna garis.

Pemanfaatan *DJTSatLib.dll* dan *gdi32.dll* membuat aplikasi *simulator* ini relatif ringan saat dijalankan (*execute*) di Komputer, hal ini dibuktikan dengan memasang dan menjalankan aplikasi *simulator* ini pada komputer dengan CPU IntelCore 2 Duo, RAM 1 GB, *Hardisk* 80 GB dan sistem operasi windows xp, tampilannya tampak pada gambar 3-6. Jumlah *memory* yang digunakan untuk menjalankan aplikasi sebesar 15 Mega Byte, tampilannya tampak pada gambar

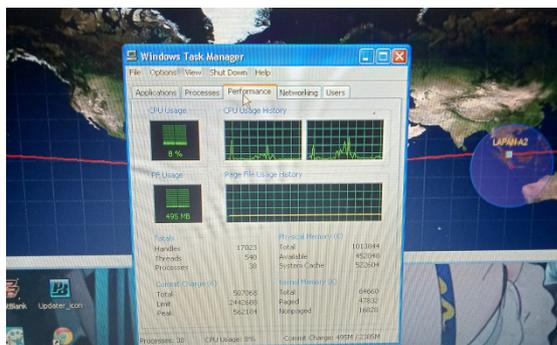
3-7. Gambar 3-8 menampilkan performance komputer saat menjalankan aplikasi, tampak pemakaian CPU sebesar 8%.



Gambar 3-6: Tampilan aplikasi simulator pada komputer spek rendah



Gambar 3-7: Tampilan jumlah penggunaan memory pada aplikasi.



Gambar 3-8: Tampilan performance windows saat menjalankan aplikasi.

#### 4 KESIMPULAN

Penggunaan algoritma model SGP4 yang berada di file DJTSatLib.dll dan file API gdi32.dll untuk teknik Mercator, dapat mensimulasikan visual posisi lintasan satelit orbit rendah secara dua dimensi dengan 100 titik posisi dalam 100 menit sebelum satelit masuk ke jangkauan stasiun bumi. Aplikasi simulator ini berjalan baik dengan menggunakan komputer berspesifikasi rendah, cukup menggunakan memory RAM 15 MB dan kinerja CPU 8%.

#### UCAPAN TERIMAKASIH

Penulis mengucapkan kepada pusat teknologi satelit yang telah memberikan sarana prasarana, Program RisetPro Kemenristek Dikti yang telah mensponsori pelatihan, Harry septanto yang membimbing penulisan, Ade putri S. yang menyediakan data pembanding STK, serta Tim redaksi dan reView er Jurnal Teknologi Dirgantara.

#### DAFTAR RUJUKAN

Bryce, *State of Satellite Industry Report*, October 2017, 20<sup>th</sup> Edition Satellite Industry association

Abhisek Maiti et.al, *AN OPEN SOURCE WEB-GIS BASED PRECISE SATELLITE TRACKING AND VISUALISATION TOOL USING TWO LINE ELEMENT DATA*, ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume IV-5, 2018

Md. Mahbubul Alam et. al 2018, *Fully Automated Satellite Tracking System for Directional Antenna*, Thesis Department of Electrical & Electronics Engineering (EEE), BRAC University.

Kathleen Riesing, *Orbit Datermination from Two Elements Sets of ISS-Deployed CubeSats*, 29<sup>th</sup> Annual AIAA/USU Conference on Small Satellites 2015.

- Giacomo Muntoni 2019, *Space Debris Observations With The Sardinia Radio Telescope*, Doctoral Thesis Università degli Studi di Cagliari
- Junyu Chen et. al 2017, *Improved orbit prediction of LEO objects with calibrated atmospheric mass density model*, Journal of Spatial Science
- Patria rahman Hakim et. al. 2018, *Development of Systematic Image Processing of LAPAN-A3/IPB Multispectral Images*, International journal OF Advanced Studies In Computer Science In Engineering IJASCSE Volume 7 Issue 10, 2018.
- T. Pisanu et. al. 2017, *Technical Report Space Debris Forecasting Campaign*, DOI: 10.13140/RG.2.2.18673.89446
- Marc Vis 2018, *History of Mercator Projection*, Bachelor thesis, Utrecht University
- Roza Madina, Amrullah Abdul Qadir, Satriya Utama 2015, *Penentuan Orbit Satelut LAPAN-A2*, Media Dirgantara Vol.10 No.4 Desember 2015
- Wu Hao, Li Zyang, Hu Jian, Tang Lingli, Li Chuanrong 2012, *Satellite Visualization Tool Based On the ArcGIS Engine and OpenGL*, Academy of Opto-electronics, Chinese Academy of Sciences, Beijing, China
- Mohammed Chessab Mahdi 2015, *TIGRISAT Orbital MotionSimulation and Analysis*, JCET Vol. 5 Iss. 1 Jan 2015 PP. 1-8 DOI: 10.14511/jcet.issue.050101
- Rio Prabowo, Andri Wahyono 2019, *Aplikasi Program Microsoft Visual Basic 6 Dalam Menganalisis Daya Dukung Pondasi Tiang Pancang*, Skripsi Program Studi Teknik Sipil Fakultas Teknik Universitas Katolik Soegijapranata Semarang.
- Abdul Rahman 2008, *System Tracking Stasiun Bumi Satelit orbit Rendah*,