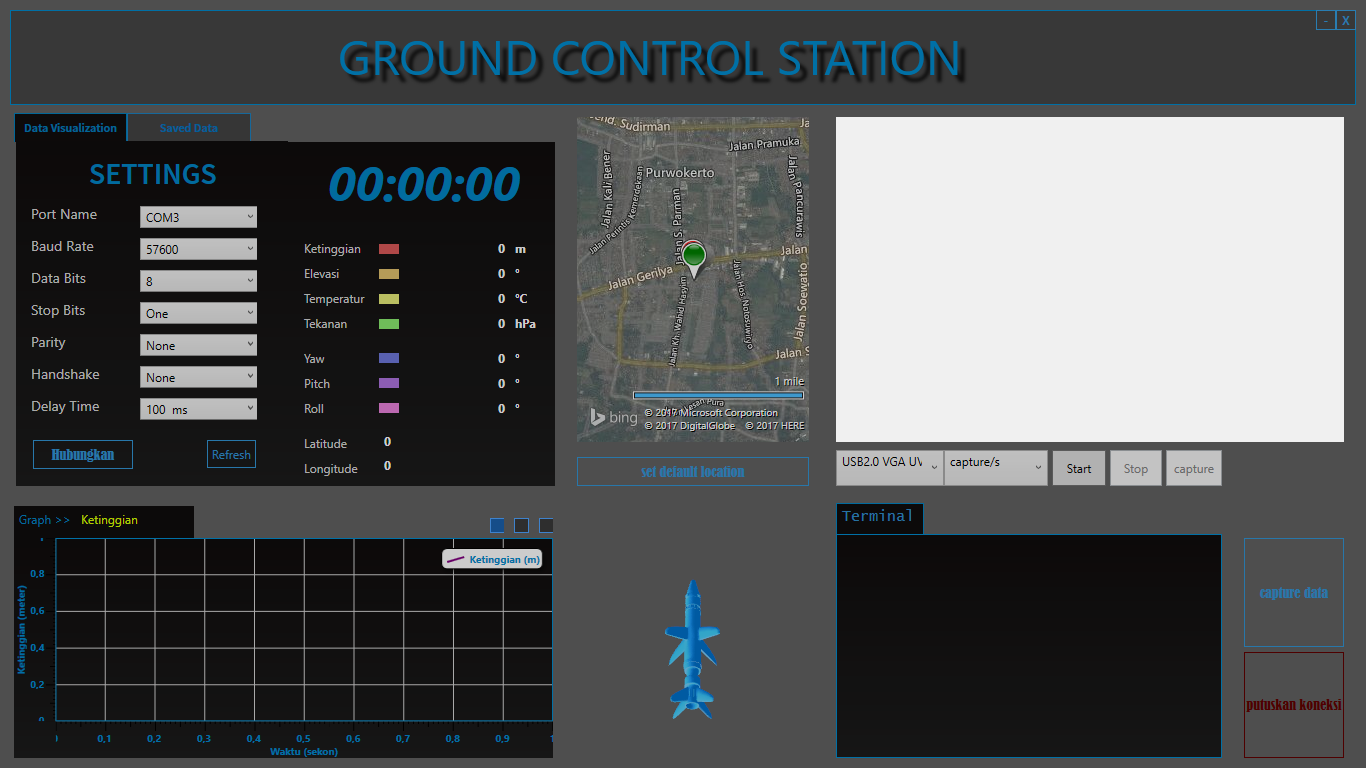
## Ground Control Station

Ground Control Station (GCS) merupakan perangkat lunak berbasis desktop yang berfungsi untuk menerima data yang dikirim dari UAV. GCS dikembangkan menggunakan aplikasi Visual Studio 2015 dengan *framework* .NET dan bahasa pemrograman C#.

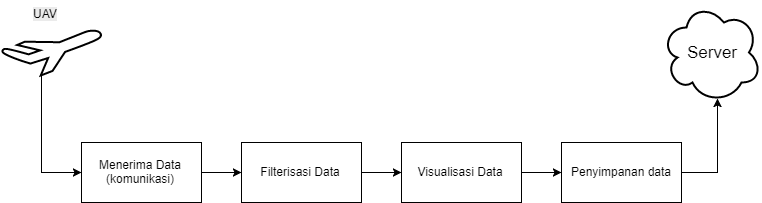


Gambar 1. Tampilan *Ground Control Station*

GCS memiliki beberapa fungsi utama, yaitu:

* Memonitor status dari *Unmanned Aerial Vehicle* (UAV)
* Membangun hubungan komunikasi dari dan ke UAV
* Pengendali UAV dalam menjalankan misi terbang

Alur kerja GCS dibangi menjadi 4 proses, yaitu: komunikasi, filterisasi, visualisasi, dan penyimpanan data.

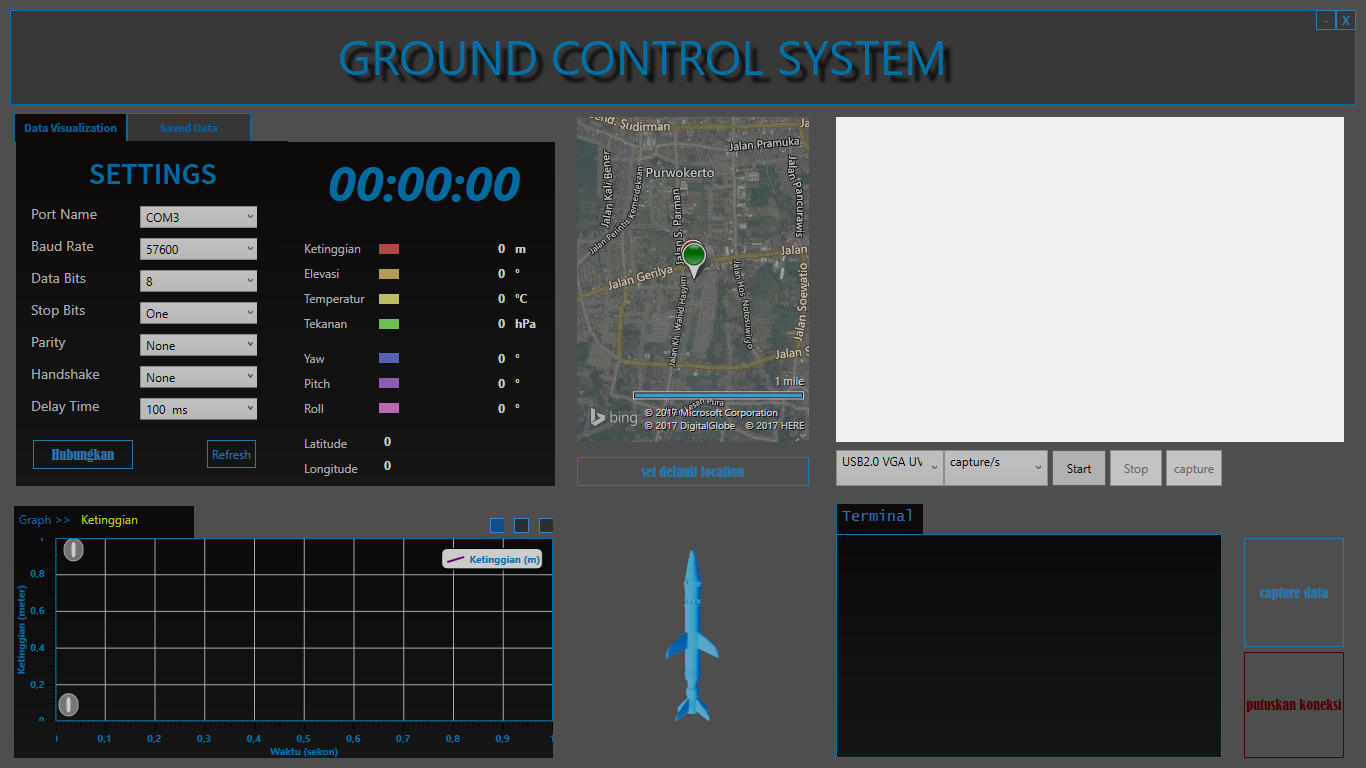


Gambar 2. Alur proses GCS

### Komunikasi

Komunikasi dari UAV ke GCS dilakukan dengan menggunakan 2 kanal. Kanal pertama digunakan untuk komunikasi data serial menggunakan frekuensi 433 MHz. Kanal kedua digunakan untuk mengirim data *Streaming* video dengan menggunakan frekuensi analog 2.4 GHz.

Untuk komunikasi serial, pengaturan komunikasi dapat disesuaikan melalui panel *settings.* Pengaturan yang dapat dilakukan adalah pengaturan port, baud rate, data bits, stop bits, parity, handshake, dan delay. Pengaturan dari GCS harus sesuai dengan pengaturan yang ada di UAV. Bila ada perbedaan pengaturan antara GCS dan UAV, maka data yang dikirim tidak akan diterima dengan sempurna dan akan mengalami galat.



Gambar 3. Panel pengaturan komunikasi serial pada GCS

Pada transmisi *streaming* video, digunakan kanal dan *port* lain yang terhubung dengan UAV. Untuk menghubungan GCS dengan UAV, maka pengguna cukup memilih *device* yang terhubung pada *port* USB di laptop.

### 

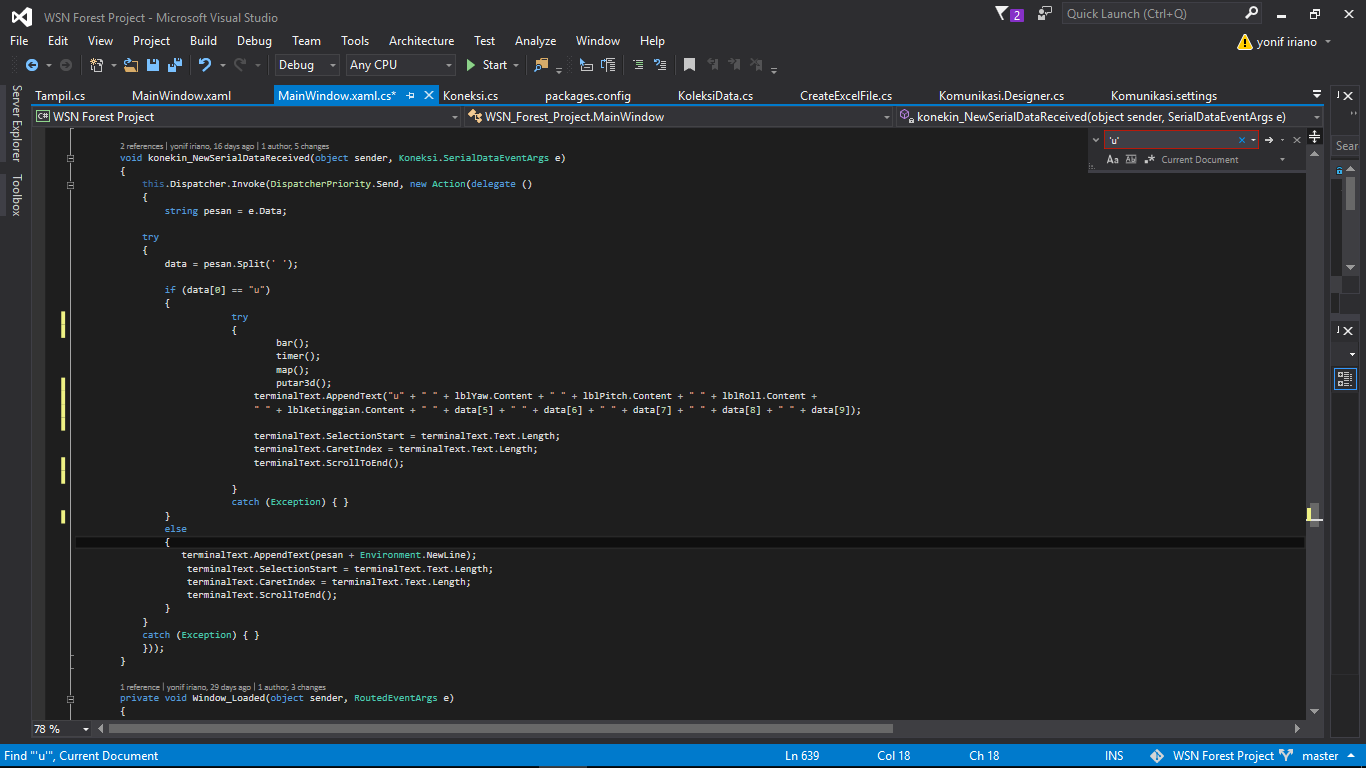
Gambar 4. Pengaturan untuk komunikasi *transmisi* streaming video

### Filterisasi data

Data yang dikirim dari UAV ke GCS tidak selalu terkirim dengan baik. Data yang diterima oleh GCS kadang tidak sesuai dengan data yang dikirim oleh UAV. Hal ini disebut dengan galat / *error.* Data yang mengalami galat tidak akan bisa dibaca sehingga tidak dapat dilakukan proses selanjutnya. Untuk mencegah data galat agar tidak dilanjutkan ke proses selanjutnya, maka diperlukan proses filterisasi.

Proses filterisasi dilakukan dengan memberikan cara memberikan *flag* pada data yang dikirim. Setiap data yang dikirim memiliki *header* berupa huruf ‘u’ dengan tipe data *string* lalu diikuti oleh data-data sensor yang masing-masing data dipisahkan oleh spasi.

Setiap data yang diterima, dilakukan pemisahan (*split*) dengan spasi sebagai pemisah. Data yang telah dipisah disimpan didalam *array* dengan nama variable “data”. Flag berupa string ‘u’ akan dialokasikan pada *array* ke-0 (data[0]). Dari array ke-0 inilah, dilakukan proses filterisasi. Bila data[0] berisi karakter ‘u’, maka satu baris data tidak mengalami galat sehingga nilainya valid dan bisa dilakukan proses selanjutnya. Sebaliknya, bila nilai dari data[0] tidak sama dengan ‘u’, maka data kemungkinan mengalami galat selama proses transmisi sehingga integritas data diragukan. Maka data tersebut akan diabaikan dan tidak dilanjutkan ke proses selanjutnya.

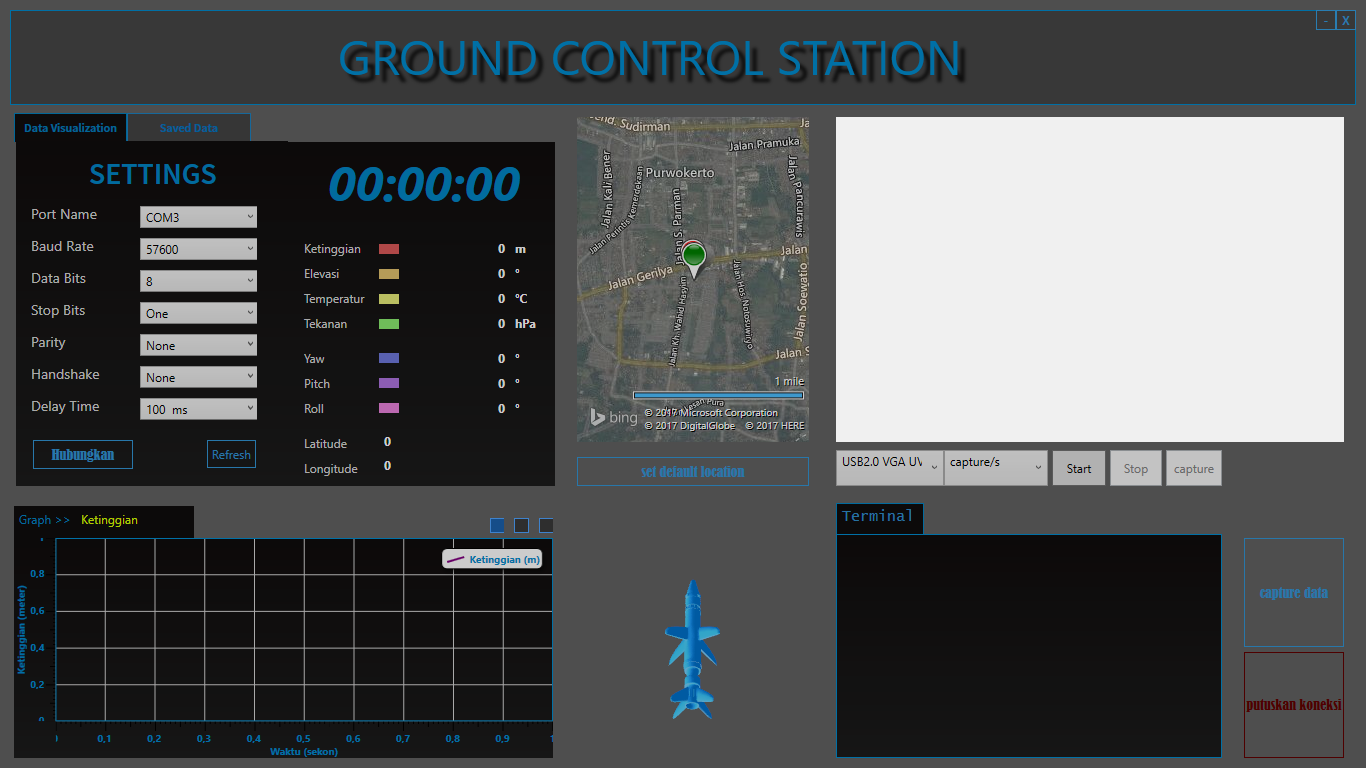


Gambar 4. Baris kode untuk filterisasi data

### Visualisasi data

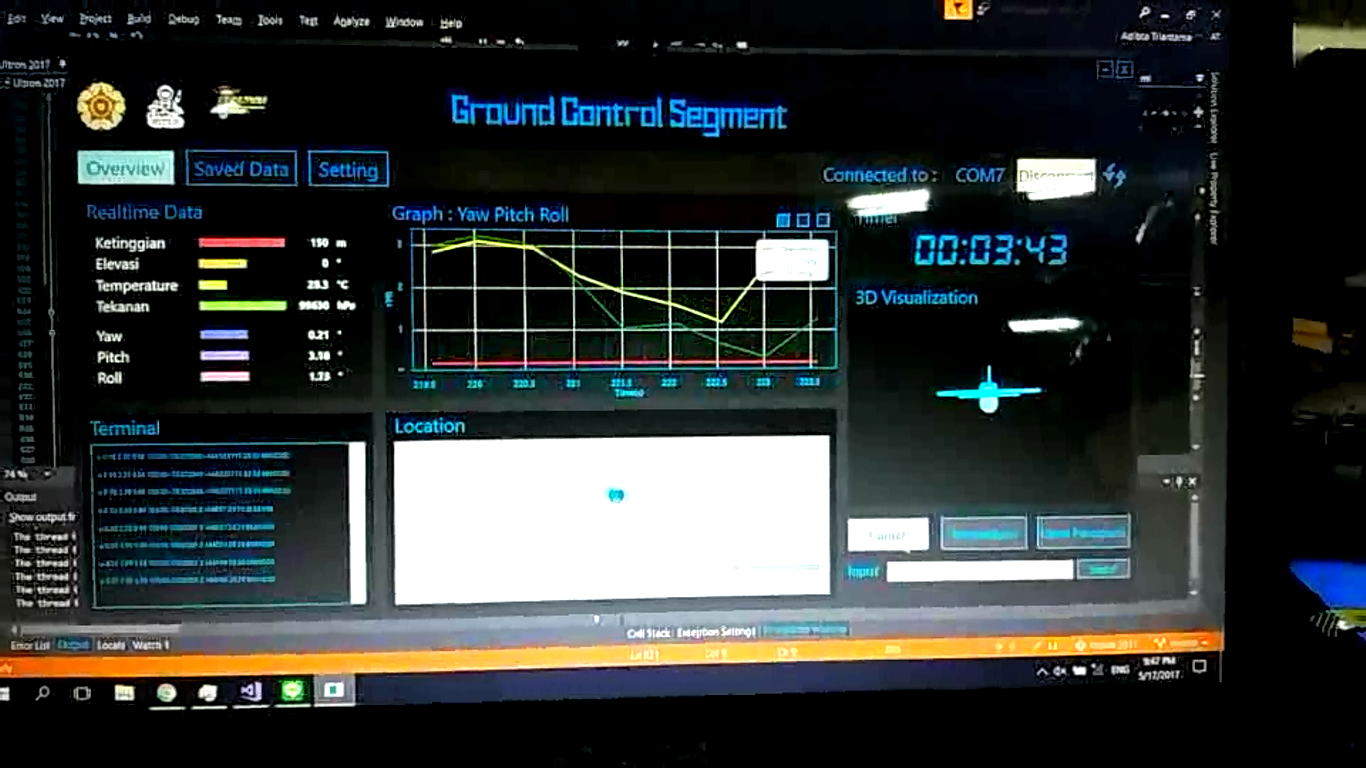
Data yang telah lolos proses filtersisasi, selanjutnya akan divisualisasikan. Antarmuka visualisasi pada GCS terdiri dari 5 ragam interaksi, yaitu: bar, grafik, visualisasi 3D, peta, dan *streaming* video.

Untuk ragam interaksi bar, dilakukan visualisasi untuk seluruh data. Ragam interaksi bar berguna untuk mempermudah pengguna membaca data yang masuk secara *real-time*.



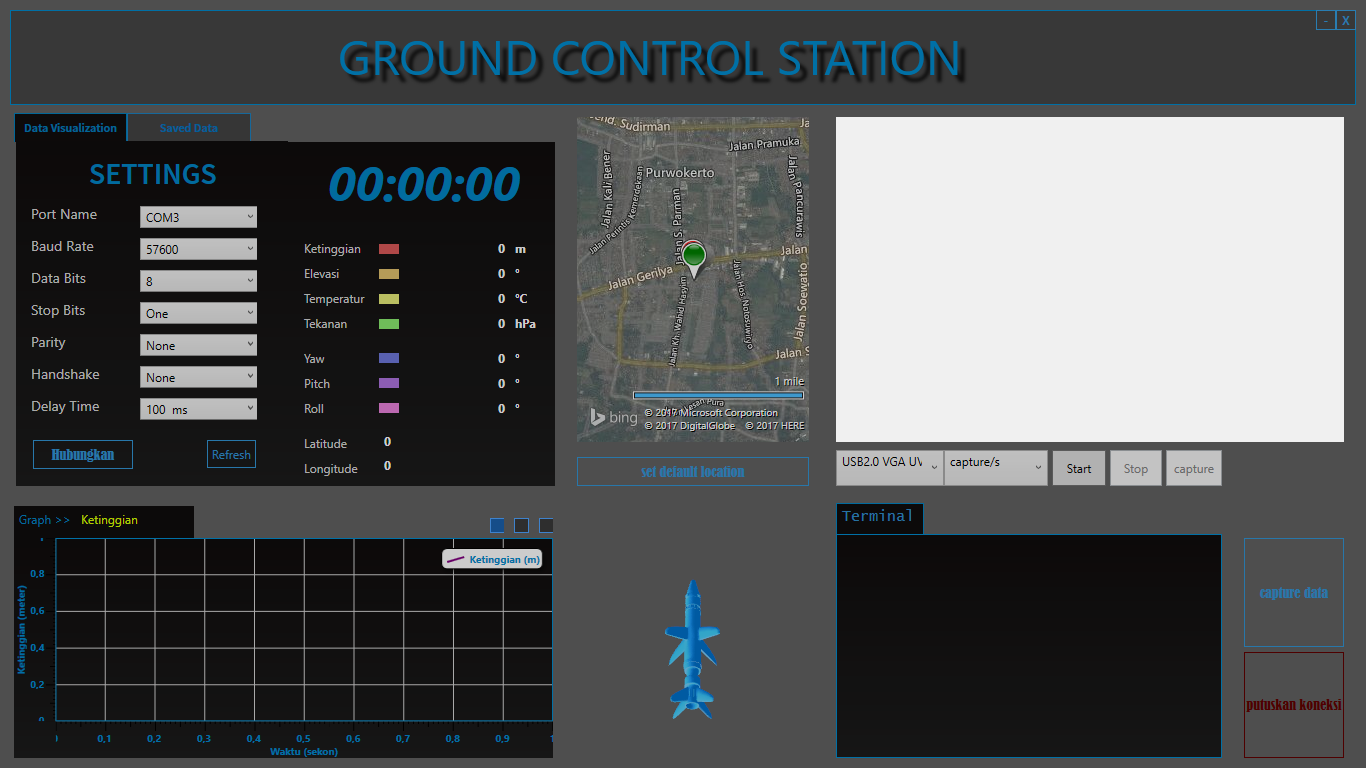
Gambar 5. Antarmuka bar

Ragam interaksi selanjutnya adalah grafik. Antarmuka grafik berguna untuk melihat perubahan data setiap satuan waktu. Setiap *history* perubahan data dapat dilihat melalui antarmuka grafik. Data yang ditampilkan melalui antarmuka grafik adalah data *yaw-pitch-roll*, ketinggian, dan elevasi.



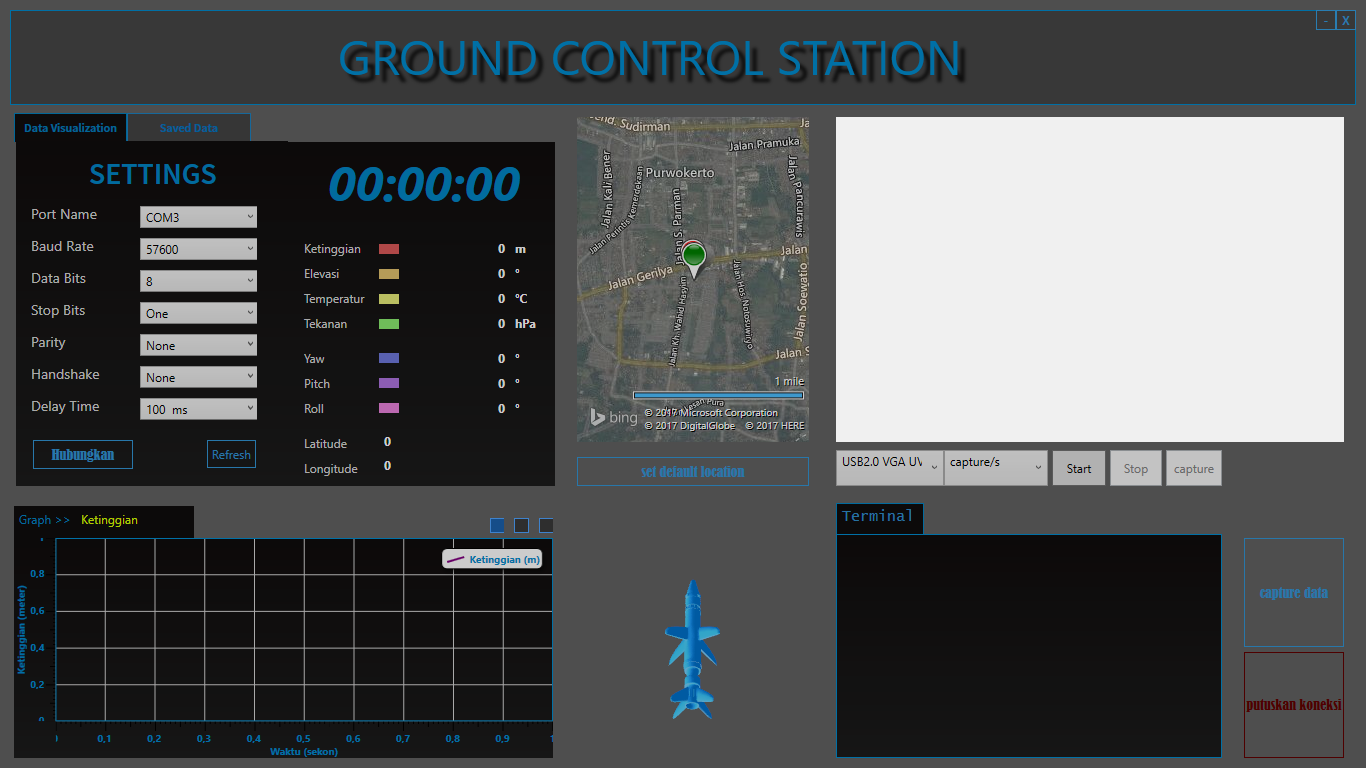
Gambar 6. Antarmuka Grafik

Ragam interaksi yang ketiga adalah visualisasi 3D. Visualisasi ini berfungsi untuk menampilkan *state* posisi dari pesawat berdasarkan data yaw-pitch-roll yang diterima oleh GCS.



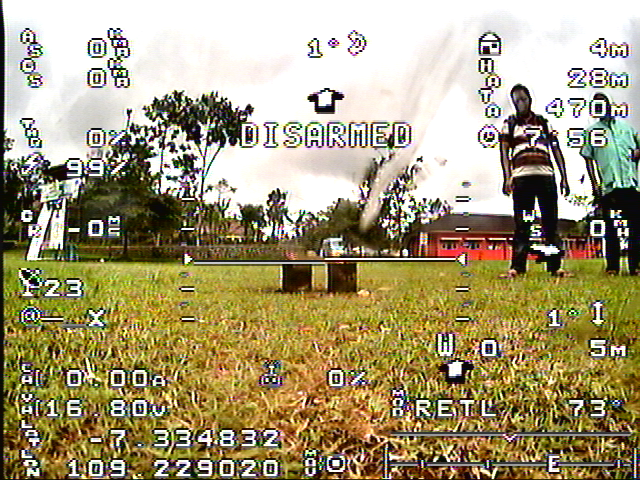
Gambar 7. Visualisasi 3D

Ragam interaksi selanjutnya adalah peta. Peta digunakan untuk menampilkan lokasi terkini dari UAV. Data koordinat didapatkan dari *longitude* dan *latitude* yang dikirimkan oleh UAV. Peta yang digunakan merupakan peta dari Bing Map dari Microsoft.



Gambar 8. Antarmuka peta

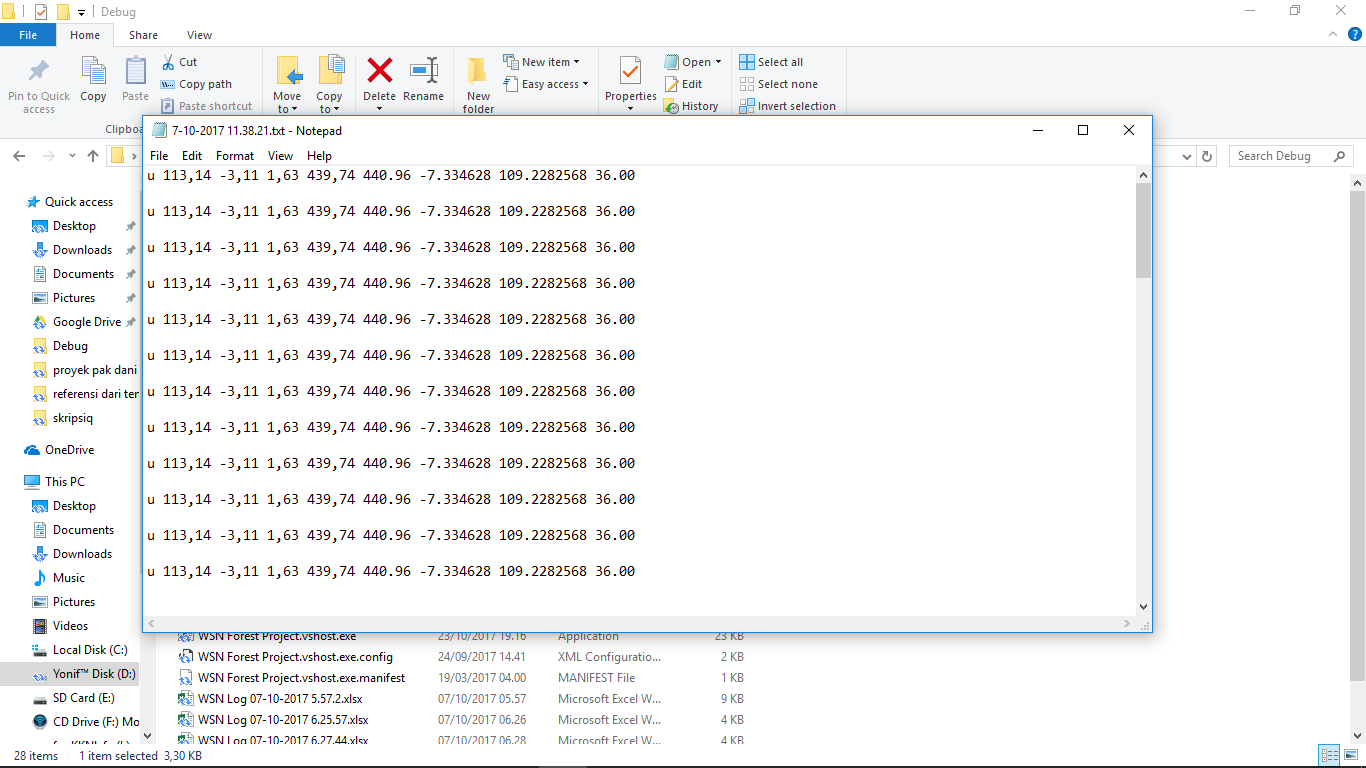
Ragam interaksi yang kelima adalah streaming video. Komunikasi ini terpisah dari data serial dan menggunakan kanal yang berbeda. Video ditampilkan secara real-time dan dapat dilakukan *capture* bila pengguna ingin melakukannya.



Gambar 9. Hasil *capture* dari *streaming* video

### Penyimpanan data

Penyimpanan data dilakukan secara luring (*offline)* dan disimpan didalam memori perangkat komputer pengguna*.* Hal ini dilakukan mengingat peluncuran pesawat dilakukan di tempat yang minim sinyal dan tidak dimungkinkan untuk menghubungkan aplikasi ke server secara *online*.



Gambar 10. Contoh hasil penyimpanan data

Data disimpan kedalam 2 format. Yaitu format Microsoft Excel (.xlsx) dan format teks (.txt). Format Excel digunakan karena lebih mudah dibaca oleh pengguna karena berwujud tabel, sedangkan format .txt digunakan untuk diupload dan diolah di *server* ketika perangkat sudah terhubung ke internet. Format .txt dipilih karena ukurannya kecil dan relatif ringan.