Sightline Control Basics for Geo-Pointing and Locating - Part 2

by

Peter J. Kennedy

Introduction: This part of the course will apply the sightline control (SLC) fundamentals described in Part 1 to the geo-pointing and location problem. Initially Section 1.0 Part 1 is reviewed, particularly pointing performance requirements which directly impact geo-pointing and location performance. The errors associated with the pointing solution for geo-location also provide a basis for examining geo-location techniques using image geo-registration to improve performance, described in Section 4.0 of the course. As this is effectively a technology in itself, just the salient aspects of the process are reviewed but should provide a source for further study and investigation if of interest. Regardless of the geo-location technique used, however, geo-pointing will generally be part of the solution. If not the solution, it will provide coarse location estimates for the geo-registration process. A substantial amount of image spatial processing is required to obtain an accurate solution to an image geo-registered location. Processing is described at a functional to capture the overall design process. Part 3 of the course will describe sensor geometry and characteristics critical to image collection for geo-location performance.

1.0 Line of Sight Definition and Performance Overview [1,2, 3]

The line of sight is defined as a vector between points on an observation platform and an observed target location in a North-East-Down (NED) reference coordinate frame as defined in Part 1 of the course. Figure 1.0 illustrates a typical pointing scenario.

https://www.pexels.com/photo/quadcopter-flying-on-the-skey-1034812/ archived on 20 May 2018 at the Wayback Machine

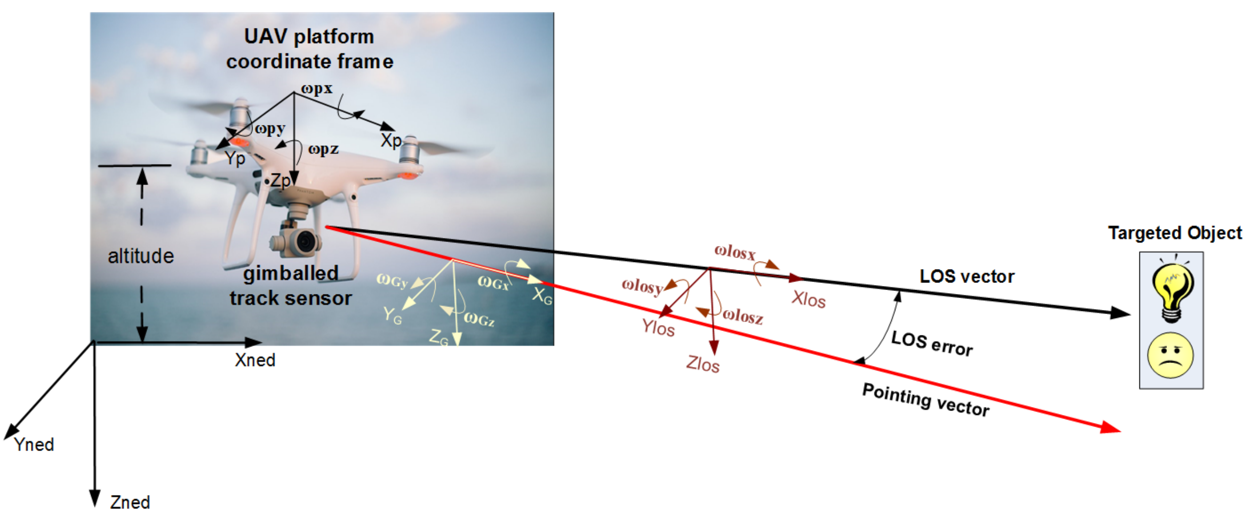


Figure 1.0 LOS Pointing Geometry

The sensor is required to maintain a target track providing its relative location based on a performance accuracy requirement. Sensors are aligned to the inertial stabilization sensor reference establishing the sightline or bore-sight within a gimbal structure capable of rotating about multiple axes. The gimbal is mounted to a stationary or moving platform. Sightline control (SLC) can be considered a two part problem:

- (1) LOS Point/Track: LOS pointing based on sensor performance over a specified the LOS platform to target kinematic envelope. It must meet pointing accuracy required for tracking and accurate location of a target
- (2) LOS Stabilization: LOS stabilization isolates the sensor from platform motion to stabilize the operating environment; rejecting disturbances due to platform motion to achieve the desired track/pointing accuracy. The control system isolating the sensor LOS vector from angular motion is: Stabilizing the LOS

Pointing error is the difference between the actual LOS orientation and the sensor pointing vector to the target. Error is a function of inherent sensor pointing error, platform motion/vibration residual error, and often the atmosphere. Error sources are characterized in the 'error budget' with platform motion ('own ship motion') often driving performance.

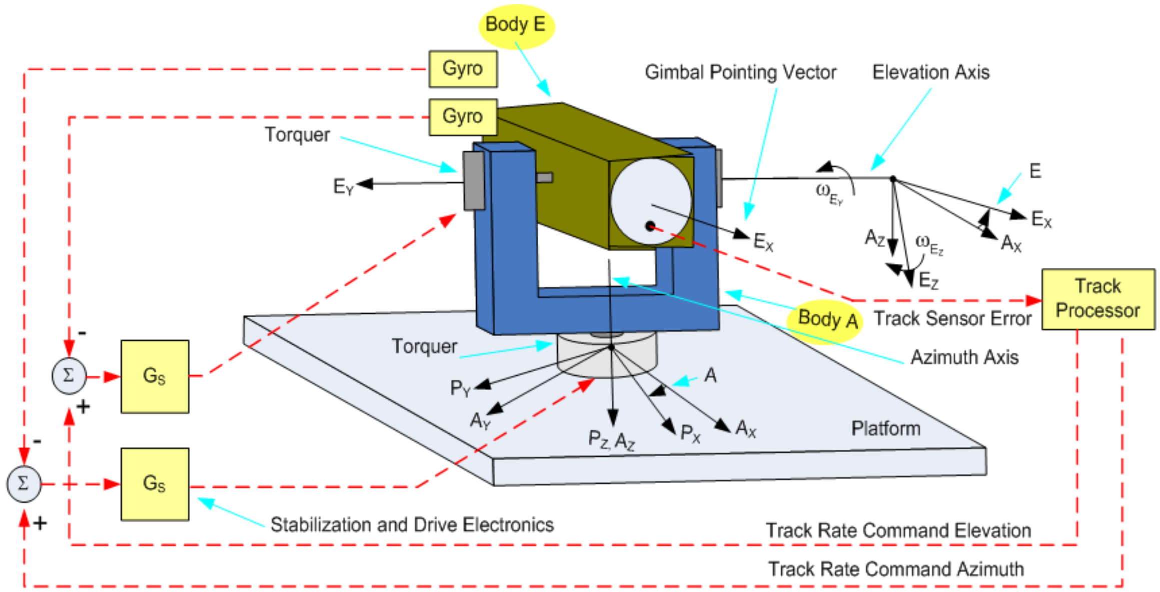


Figure 2.0 Simple 2-axis Elevation over Azimuth Gimbaled Track Sensor Geometry

Figure 2.0 illustrates a simple two axis elevation over azimuth (el/az) gimbal design. The outer body rotates in a horizontal plane relative to the base; termed azimuth rotation. The inner body rotates in a vertical plane relative to azimuth; termed elevation rotation. A coordinate frame is attached to each rotating axis body; as shown the inner el coordinate frame x-axis is the pointing axis. As shown, sensor and gyros are mounted directly on Body E with LOS motion defined by the inner el axes. Direct drive motors are used to control axis rotation; they will decouple axes inertia from base motion except for friction. The LOS and elevation inner body coordinate frames are coincident; gimbal elevation is LOS elevation; LOS cross elevation (XEL) is the rotation axis orthogonal to the LOS elevation axis (y) and the LOS pointing axis (x). So it is important to note that the rotation axes are azimuth outer and elevation inner while the LOS axes are cross elevation and elevation. Elevation LOS and rotation are the same but cross elevation ~ azimuth*cos(E). At E=0° azimuth and cross elevation are equal; At E=90°, all 2-axis gimbal designs have a NADIR condition (Body E in Figure 2.0 points straight up) meaning from the definition of XEL, there is a singularity or division by zero and the azimuth axis control demand grows unbounded. Physically the XEL rotation axis is lost resulting in the condition termed 'gimbal lock'. Figure 3.0 illustrates a high level SLC control system architecture and the interaction between key elements.

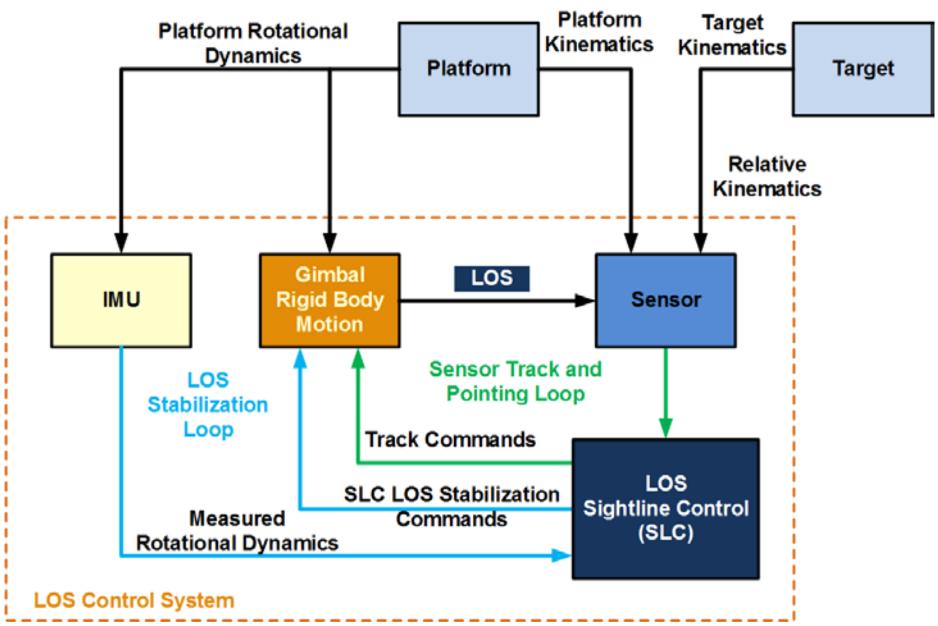


Figure 3.0 High Level SLC System Architecture

The primary input motion drivers are the target and platform. The track sensor provides the information for the track and pointing control loop based on the target to platform kinematics (green). The gimbal is the payload LOS pointing device and its motion relative to the LOS is characterized by the gimbal 'Rigid Body' dynamics which is effectively the mass to control and stabilize; often termed the plant in control terminology. The inertial measurement unit (IMU) measures the angular rate motion of the platform to drive the LOS stabilization control loop (blue). An IMU consists of 3 gyros measuring angular rates about 3 orthogonal axes (i.e. x, y, z) axes defined by the IMU and 3 accelerometers measuring linear acceleration along each axis. The track loop is the outer low bandwidth servo loop while the stabilization loop is the inner high bandwidth servo loop controlling gimbal motion.

Many 2-axis designs use mirrors with the sensor package located below the gimbal base. This reduces payload size, weight, and power (SWaP); a significant benefit if geo-locating from a small UAV, but also adds complexities unique to mirrors described in Part 1. Multi-Axis Gimbal Geometries > 2 offer the potential for improved pointing performance benefitting from:

- Limited travel also limits disturbance geometry dependence to small angles
- All inner LOS axes protected from environment; no environmental seals/seal friction required.

A 3-axis design with an inner axis mounted on the elevation body rotating in XEL can solve the NADIR issue. A general control configuration for a multi-axis gimbal design has the inner axes driven by track sensor error; stabilize about inner axes; outer axes follow inner axes. A generic block diagram for a three-axis design is shown in Figure 4.0. The inner axes are: (1) cross elevation (XEL), (2) elevation (E) and the outer axis: azimuth (AZ). A stabilized steering mirror (SSM) is mounted on cross elevation. The sensor drives inner axes track loops while the inner axes rate loops use LOSR feedback from the IMU mounted on elevation axis. Cross elevation mirror angle error drives the azimuth axis in following servo slaved to cross elevation

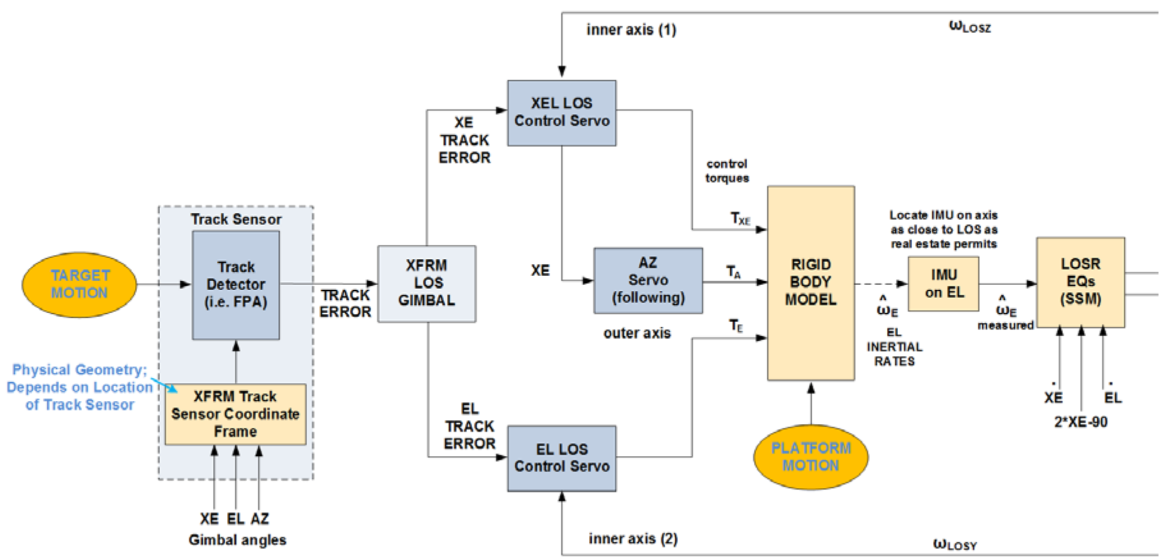


Figure 4.0 Typical Three Axis Gimbal Control Architecture

A consideration for using a multi-axis mirror gimbal design is as payload size increases; size and weight become prohibitive for direct mount on an inner gimbal. Payloads will require waveguides or periscopic optical path and steering mirrors mounted to inner axes. When using a long optical path; it is best to integrate the final LOS control precision beam steering components as the last pointing elements in the gimbal optical path; otherwise they can:

- impact diameter of optical path, vignette sensor field of view unless beam motion offset is accounted for
- Compromise overall LOS pointing as they will drive inner axes after the precision beam steering elements; resulting in LOS axis coupling

Regardless of the axis configuration, controlling the relative angular motion, direct or induced, between the platform and each LOS axis is critical to stabilization. Disturbance attenuation is primarily an inverse function of loop gain or bandwidth. LOS and gimbal motion, characterized by the 'Rigid Body' gimbal dynamics (as opposed to structural flexure which also needs to be addressed), must be modeled as part of the design process. In a simplified control loop block diagrams the Rigid Body is equivalent to 1/J; J being a characteristic rotating axis inertia. In an actual application, it is a set of vector equations that describe the motion of a rigid body in 3 dimensions; to be described later. The key to SLC is disturbance rejection which equates to LOS stabilization. A simple illustration mapping of disturbance rejection to elements of a SLC system is shown in Figure 5.0 using the simple 2-axis gimbal from Figure 2.0. These disturbances will produce geo-pointing errors that ultimately impact geo-location.

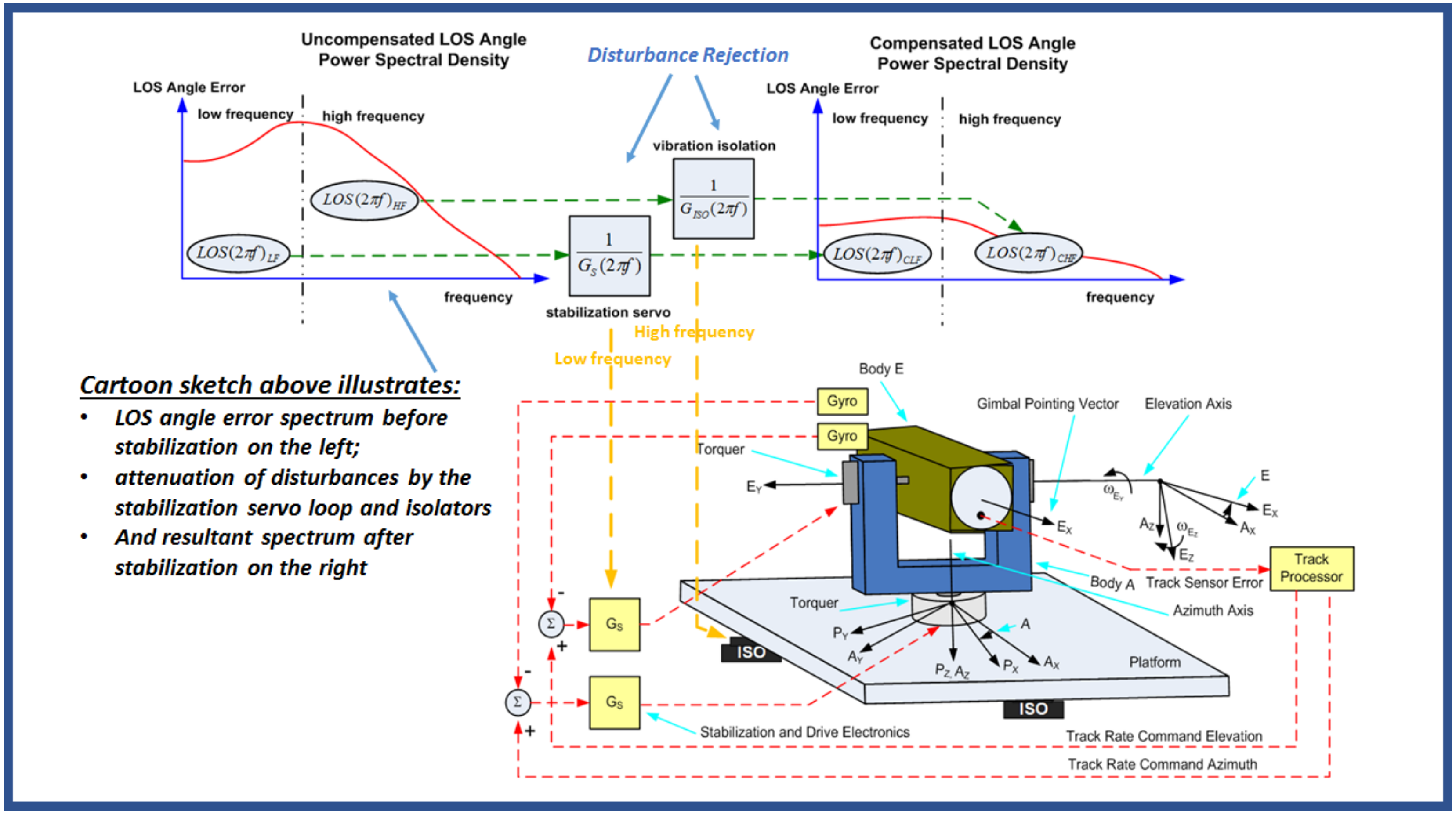


Figure 5.0 SLC System Disturbance Rejection

The cartoon sketch in the upper portion of the figure illustrates the disturbance rejection objective. The LOS angle error spectrum before stabilization is on the left; followed by attenuation of disturbances by the stabilization servo loop and isolators and finally the resultant spectrum after stabilization on the right. The stabilization servo loop handles lower frequency disturbances. Gain (i.e. G_S) equates to BW so higher bandwidth provides better attenuation limited by power, noise, and servo stability constraints. In the digital world higher bandwidth equates to a higher sample frequency probably > 1 KHz and 5 KHz typical. Latencies (i.e. serial links) are a potential source of significant error. High frequency disturbances usually require mechanical isolation if significant to performance; resonances are always there just where. Disturbance error is characterized in terms of jitter and bias or offset. Jitter is short term deviation about a zero mean. Bias is a longer term error, often due to mounting/component misalignment. Jitter will deviate about this bias or offset. For design, bias and jitter are quantified via error allocation (desired or predicted) distribution and a final error budget (final or allocation after complete critical design phase). Figure 6.0 illustrates the physical phenomena of jitter and bias error. The 2-axis gimbal is shown again with the sensor field of view (FOV) and/or divergence projected (solid red lines) as the large tan circular pattern in the figure. The red- cross is the pattern center or instantaneous aim-point. The target location within the sensor field of view is the blue-red square. Perfect tracking is shown in the upper left caption with the red-cross aim-point superimposed on the blue-red target square. The upper right shows the effect of a constant bias error; the aim-point being offset by a constant pointing error. The combination of pointing bias and jitter is then shown superimposed on the sensor FOV in the main figure. The bias offset, shown by the black cross, is the longer term offset error, often due to mounting/component misalignment. Jitter is a short term deviation about a zero mean and it generates a radial jitter envelope shown by the striped gray circle in the figure. The aim-point (red-cross) is offset to the black cross and can lie anywhere within jitter radial envelope. The impact of jitter on pointing depends a lot on the sensor characteristics; primarily instantaneous FOV (IFOV: FOV subtended by a pixel) and frame rate (FR). For geo-location, jitter will manifest itself as increased pointing error reducing geo-location accuracy. Geo-locating an image often uses an image-registration technique (described in section 4.0). This is a process that compares and aligns a sensed image with a geo-referenced stored image. Severe jitter can result in image blurring. This will reduce the effectiveness of the image processing used to align the sensed image with the referenced image, thereby reducing the overall location accuracy of the geo-registration. For example, frame rate or integration time essentially bounds the camera disturbance frequency spectrum at F_{DF} =max(FR, 1/(2*Ti)). The IFOV can be used to estimate a bound for the worst case angular rate jitter as approximately $|\omega_{dist}| < IFOV * (2 \pi * F_{DF})$.

The last topic addressed for the overview is performance. Given all the SLC considerations described how does one evaluate overall performance? Figure 7.0 is a somewhat highly condensed chart describing the overall performance evaluation process. Performance at a system level must flow down from system or mission requirements. Sensor components are chosen to meet or exceed those requirements. This process is illustrated by the dark brown square centered in the figure. The platform and system component jitter and bias disturbance environment is characterized in an error budget or allocation, conceptually illustrated in the upper right hand corner. Allocation is generally considered the desired distribution of errors while the budget is what one has to live with as the design evolves. The physical interpretation of the jitter and bias error, discussed previously, is again shown in the upper left corner. An algorithm is then required for evaluating a performance metric as a function of the jitter and bias error. This is often based upon an estimated statistical distributions of jitter and the bias. The example provided in the bottom right side of the figure is termed the Rician Distribution derived from a bi-normal pointing distribution and weighed by a beam shape profile. However the metric should be chosen that best fits the application. The metric shown may work well for LIDAR but geo-pointing with a passive sensor may only require the pointing distribution. Using a metric, the impact of jitter and bias error on performance can be predicted. Initially looking at an uncompensated design to establish the overall level of compensation required followed by introducing the required compensation controls and isolation. For example for the metric described pointing energy on target can be predicted as a function of the jitter and bias error. The plot in the bottom right are constant contours of EOT as a function of normalized jitter and bias error. It can be observed there is a range of jitter and bias values that can meet the desired EOT. Similar plots could be obtained for only pointing probability. Required EOT or pointing probability would be a system performance requirement to meet a mission objective.

Section 1.0 Key Points Summary

- ***SLC two part problem: (i) LOS Track/Point and (ii) LOS stabilization***
- ***Performance defined by geo-pointing sensor accuracy requirements and LOS stability for sensor to meet its requirements over the platform disturbance envelope***
- ***Disturbances to geo-pointing LOS characterized in terms of jitter and bias; quantified by disturbance type in an error budget***
- ***Pointing metric is required to evaluate pointing performance and design tradeoffs to meet performance***

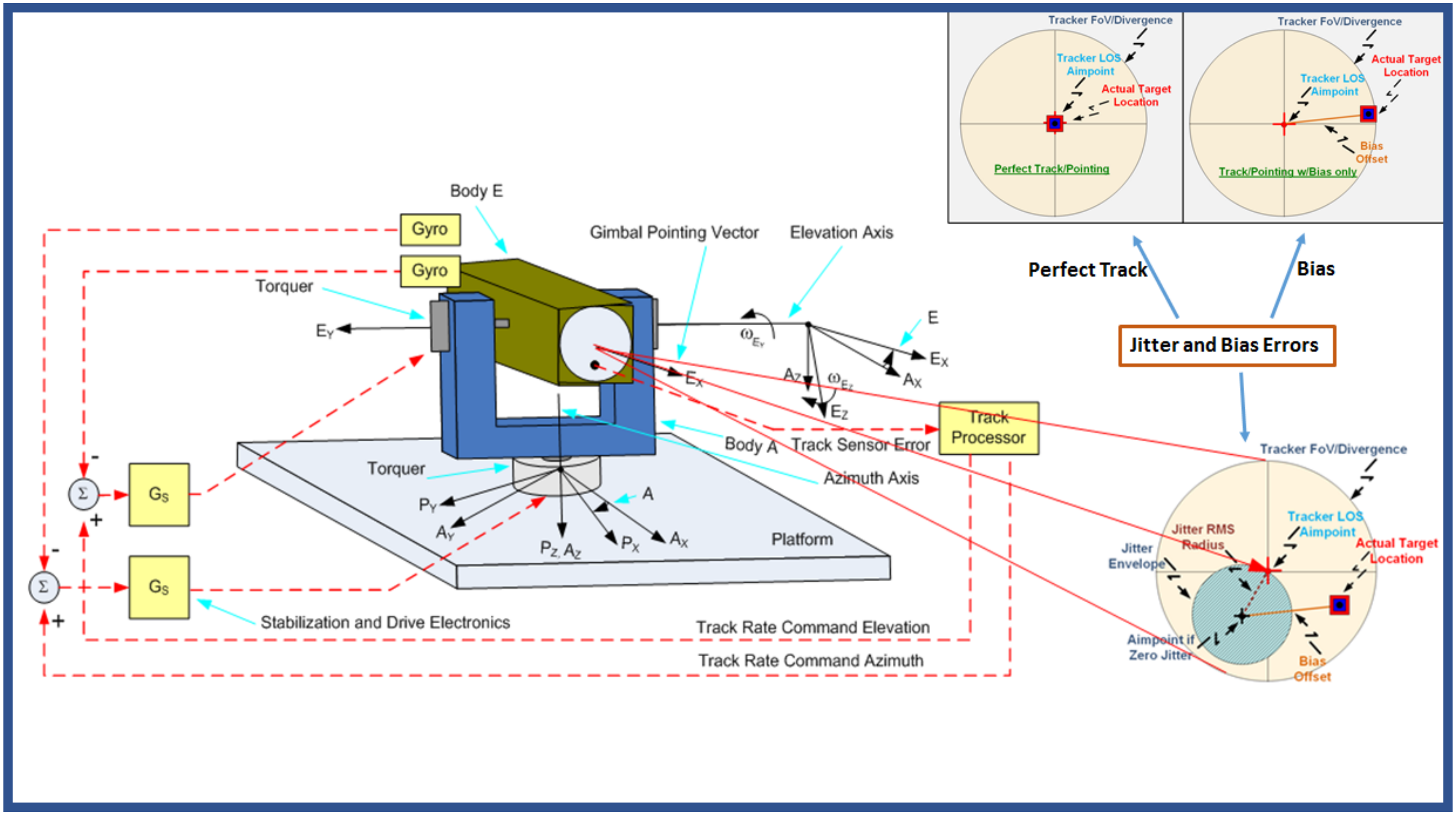


Figure 6.0 Illustration of Jitter and Bias Pointing Error

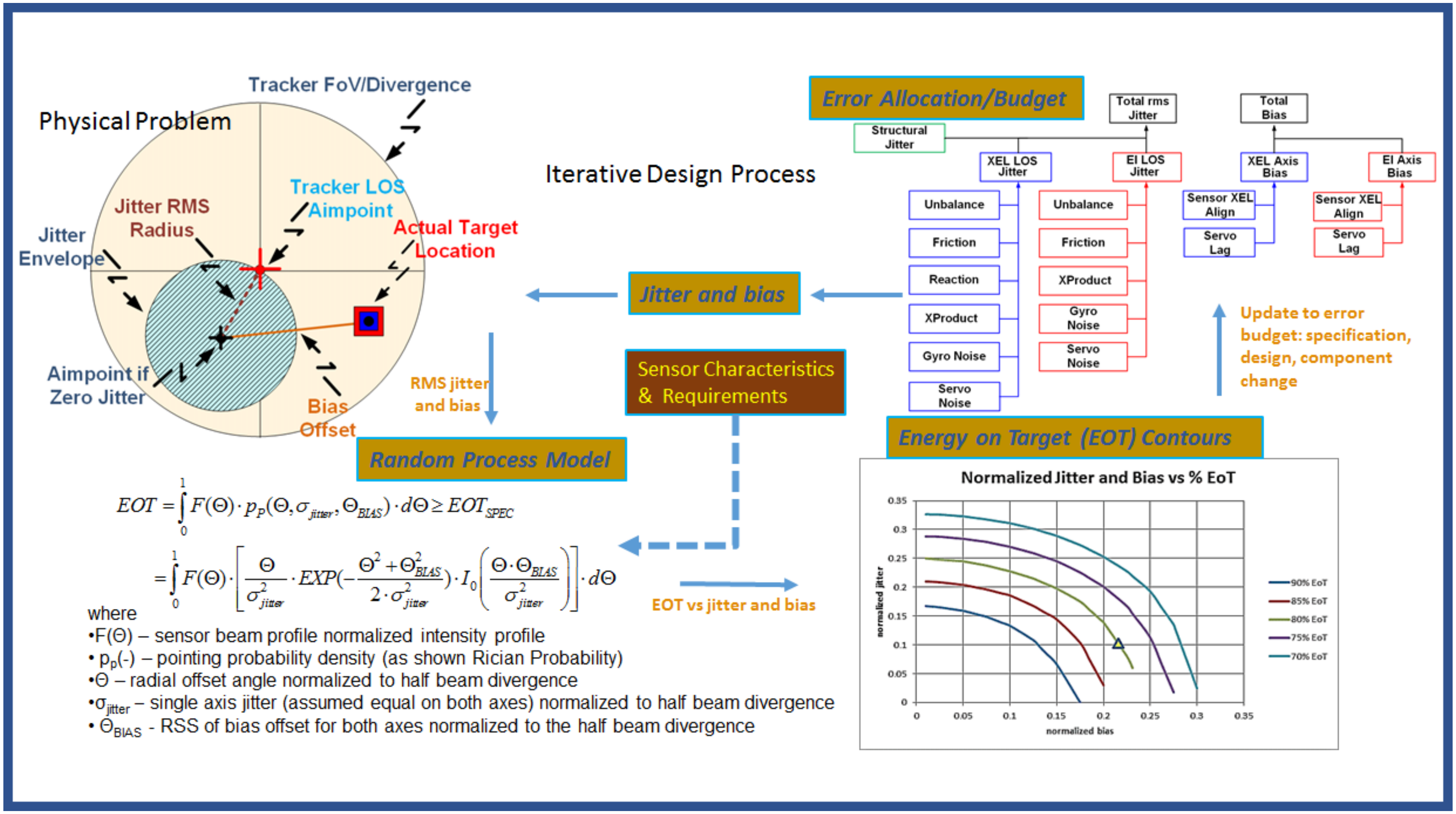


Figure 7.0 SLC Performance Evaluation Process

2.0 LOS Pointing Control Design Components

A very brief overview of key components required by a gimbal system for geo-pointing

- Inertial Angular Motion and Position Sensors

  Inertial motion sensors include Gyros; Inertial Measurement Unit (IMU), Inertial Navigation system (INS). Gyros and IMU measure angular rate. Performance ranges from commercial to tactical to navigation quality. Types include Spinning mass, fiber optic gyro (FOG), ring laser gyro (RLG), MEMS rate sensors, magnetic hydrodynamic rate sensor, and hemispherical resonator gyro. Types are summarized as:

  - gyros-measures inertial angular rate; configured in single axis, two-axis, and three-axis versions
  - inertial measurement unit (IMU) – 3 gyros mounted on three orthogonal axes (i.e. x, y, z) measure rate about each respective axis and 3 accelerometers mounted on each axis measure acceleration along each axis
  - Inertial navigation system (INS)-measures inertial angular orientation and with GPS, location. It is effectively an IMU with integrated outputs from a navigation algorithm (i.e. Attitude and Navigation Equations). A simplified block diagram of an INS is shown in Figure 8.0.

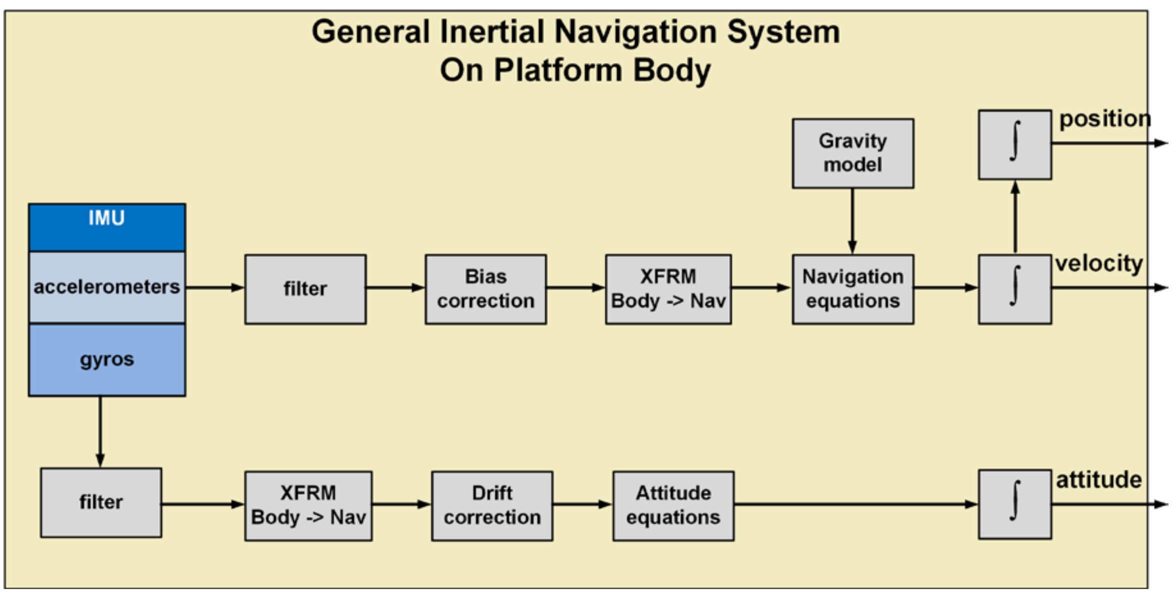


Figure 8.0 INS Simplified Block Diagram

- For many applications, especially requiring geo-positioning an INS is mounted to the platform body (possibly comes with the platform) and an IMU or set of gyros are mounted on gimbal or at its base. In general the gyro and IMU have much higher bandwidth and sampling rate than the INS as the INS needs time effectively process data. However if the INS and IMU are aligned; accurate inertial angle data from the INS can be blended with integrated rate data from the IMU to update the effective inertial angles between INS sample substantially improving geo-positioning performance. The alignment between INS and IMU is often termed transfer alignment and can be performed during operation at a time interval consistent with the sensor noise characteristics. An alignment algorithm is described in section 3.0.

- Gyro and IMU (gyro and accelerometers) error sources that can corrupt the geo-location estimate are drift and angular random walk (ARW). Drift is usually considered a bias, but may have short and long term components. ARW is considered a random disturbance error associated with the INS gyros and accelerometers. The magnitude of the random error vary significantly with the sensor quality. ARW is effectively the result of integrating the gyro white noise. Random walk can be described by the differential equation $dx/dt= \eta$ where η is a white noise term with PSD N. Drift is effectively a constant bias angular rate on the output of a gyro, independent of the input. It is an offset rate that will not change during a short run, but may vary from turn-on to turn-on or over longer periods. Another gyro error is the scale factor error which is in general a linear error that is proportional to the input signal; however it may exhibit some degree of non-linearity over the full scale sensor range.
- Inertial errors can vary significantly depending on the quality of the inertial sensors. Drift and bias in tactical grade sensors is usually high. Heading can be difficult to measure without an inertial grade INS/GPS. Tactical grade inertial sensors often use a magnetometer that provides heading relative to magnetic north which can be corrected for true north given latitude and longitude. However magnetometers measurements are corrupted by externally induced magnetic fields. Differential GPS is another method to obtain heading; accuracy being dependent on the separation distance between two GPS antennas. In addition, as the heart of an INS is an IMU, many of the IMU errors can be modeled by random white noise and Markov processes and included in a navigation error model implemented within an Extended Kalman Filter that significantly reduces their impact on the final solution.
- The performance of low quality inertial sensors can sometimes be improved if referenced to an inertial grade sensor co-located on the same structure. This s another advantage of transfer alignment mentioned above; using information from the reference system to align low quality inertial sensors on the platform. The algorithms use velocity or velocity integral matching to align accelerometers and gyro rate as will be shown in the next section.

- <u>Relative Angle Position Sensors</u>

  Two often used relative angle position sensors for measuring gimbal angles are resolvers and encoders.

- Resolvers: construction similar to motor, rotor and stator coils produce sine/cosine signals from whose phase angle is determined, absolute angle measurement, wide angle range 360°, use multi-speed for high accuracy down to ~10 urad
- Encoders:
  - Relative encoders: bar type pattern generates pulse sequence whose count provides angle relative to a reference, when power is lost lose count, must reset wide angle range 360° ,very accurate with interpolation algorithms
  - Absolute encoders: similar to resolver, maintains count even if power is lost , provide wide angle range 360°
- Other position sensor types; potentiometer, hall effect, capacitive, inductive most tend to have limited angle coverage, less accuracy

- <u>Motors</u>
  - Recommend direct drive for LOS stabilization applications to benefit from load inertia; also cog less with low ripple torque
  - Brushless DC (BLDC) direct drive motors need to commutate with hall effect sensors, resolver, or encoder
  - Brush DC direct drive, simpler to implement, downside brush friction, sparking concerns in explosive atmosphere and brush wear. For many applications gimbals do not run continuously at high RPM so wear may be minimal and friction often less than bearing friction
  - For small payload applications with low load inertia can consider motors or other motion control devices that provide very high BW response, compensating for low inertia.
    - Limited angle rotary torque motors
    - Friction drive piezoelectric
    - Fast Steering Mirrors (FSM) high BW limited travel
    - Voice coil actuators (VCA)

<u>Section 2.0 Key Points Summary</u>

- ***A brief description of key hardware components required to implement a SLC system for geo-positioning is provided***
- ***Inertial rate and position sensors: gyro, IMU, INS.***
- ***Angle measurement sensors: resolvers, encoders***
- ***Motors: direct drive and limited angle actuators***

3.0 Geo-Pointing

Accurate pointing, discussed previously, provides the means to geo-locate or estimate the geographic location of an objects position by determining its geographic coordinates. A geo-location is defined based upon the latitude and longitude coordinates of a particular location and may be enhanced further by cross referencing or mapping to another type of address information depending on the application. This section describes basic geo-location of a point-like object based upon the location and orientation of a pointing platform. The geo-pointing vector for location will require several coordinate frames of reference which may include sensor LOS, gimbal orientation, and the final geo-location reference frame. The actual coordinate frames will be application dependent. Defining the rotation matrices and the direction cosine matrices for each reference coordinate frame is key to determining a geo-pointing vector. Quaternions are another approach to representing the DCM. As the derivative of quaternion elements are a set of time varying linear differential equations, their values are easily obtained via integration. Quaternion rotation algorithms are readily available as self-contained CAD processing blocks; however quaternions will not be discussed further in the course.

For a geo-pointing application, one may assume the platform position is known and a targeted objects position at some known range needs to be determined. An optical or RF sensor mounted on the platform may be used to determine target angular position relative to the platform if not known. The sensor is mounted on a 2-axis gimbal, as described in previous sections of the course. The sensor could be part of a sensor network, surveillance or security, so that an objects position must be known in a coordinate frame common to the sensor network. The other major platform sensors are the gimbal angular position sensors, an IMU for stabilization, an INS for platform inertial orientation, and a GPS. The target location is initially measured in a sensor line of sight (LOS) frame, converted to a position vector, and rotated through a sequence of direction cosine matrices (DCM) to obtain its position in the common frame. For the example, a gimbal mounted sensor is secured to a vehicle and would have coordinate frames that define the: sensor, gimbal, platform body, local surface referenced north east down, and finally the geo-location frame. The most often used geo-location frames are earth centered earth fixed (ECEF) and earth centered inertial (ECI). The ECEF frame is more often used for representing position and velocity of terrestrial objects while ECI for satellite applications specifying celestial objects location. The remainder of this course will assume the use of ECEF coordinates and a set of coordinate frames, often used, illustrated in Figure 9.0 as follows:

https://www.pexels.com/photo/quadcopter-flying-on-the-skey-1034812/ archived on 20 May 2018 at the Wayback Machine

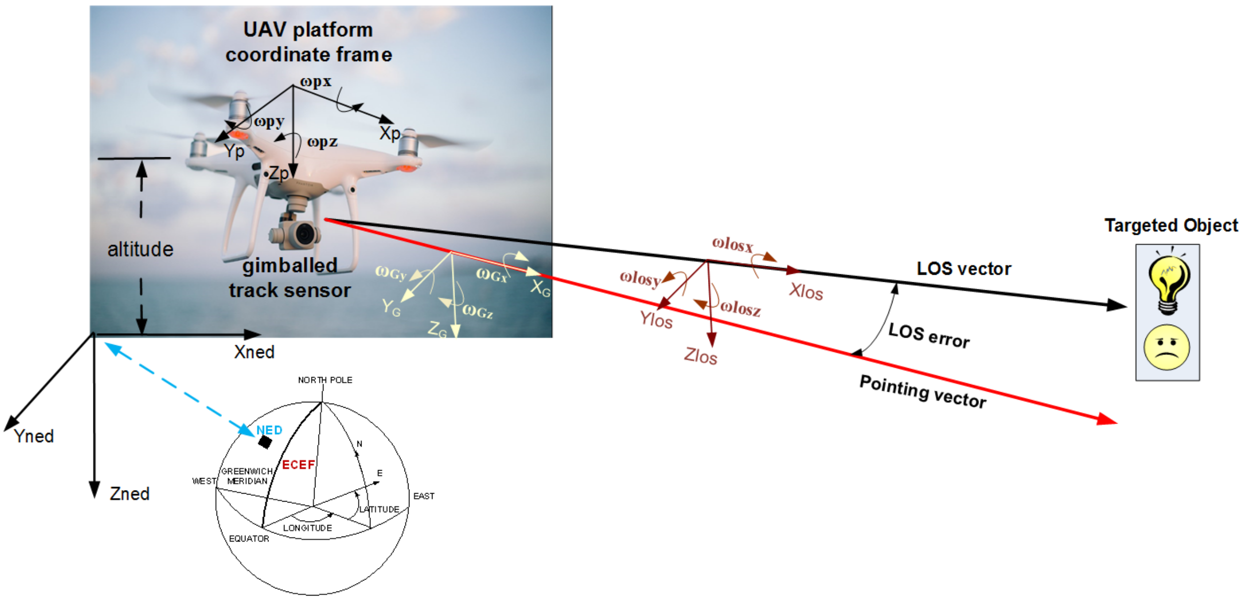


Figure 9.0 Typical coordinate frame geometry for geo-location

- Earth-Centered, Earth-Fixed (ECEF)
  - The ECEF coordinate frame is a spherical earth model with the origin at the center of the Earth (Earth-Centered) and rotates with the Earth (Earth-fixed)
  - The z-axis through the North Pole
  - The x axis through the Greenwich Meridian
  - The y axis completing a right-handed coordinate frame.
  - Latitude, longitude, and altitude location maps to ECEF position vector
- North East Down (NED)
  - Also termed inertial reference or local level
  - Platform/sensor pointing reference location
  - X axis points north orthogonal to gravity, Y axis points east orthogonal to gravity, and Z axis down co-linear with gravity
- Platform coordinate frame
  - Heading is about the NED Z axis (~down)
  - Pitch is about the new Y axis,
  - Roll is about the new X axis, with right hand rule for positive rotation
- Gimbaled coordinate frame (configuration dependent)

- Azimuth is about z-axis orthogonal to base mounting plane (~down)
- elevation is about the new Y axis,
- Roll is about the new X axis, with right hand rule for positive rotation

– Line-of-Sight (LOS) frame
  - X axis out the LOS, Y axis to the right when looking forward and Z axis down, with right hand rule for positive rotation
  - If a mirror is used, need to develop LOS and image kinematics

The angular rates shown in the figure are defined similar to those described in Part 1.0 of the course.

- Platform angular rates referenced to NED coordinates - ω_{PX}, ω_{PY}, ω_{PZ}
- Gimbal angular rates referenced to NED coordinates – gimbal dependent
- LOS angular rates referenced to NED coordinates - ω_{LOSX}, ω_{LOSY}, ω_{LOSZ}

For the two-axis gimbal example used throughout the course, the general gimbal rates are actually the azimuth [ω_{AX}, ω_{AY}, ω_{AZ}] and elevation [ω_{EX}, ω_{EY}, ω_{EZ}] axis rates described in Part 1.0 of the course. Using ECEF coordinates, an objects earth-centered location can be defined as a position in terms of latitude, longitude, and altitude relative to an ellipsoidal Earth model referenced to the World Geodetic System (WGS) 84 datum. The LOS coordinate frame defines the sensor bore sight relative to the LOS. With the sensor mounted on the inner gimbal structure, it is aligned to the gimbal axes. The gimbal frame defines the gimbal orientation relative to the platform body. The body coordinate frame is attached to the platform and often uses standard aircraft coordinates with the x-axis along the platform longitudinal axis, the z-axis pointing down orthogonal to the x-axis, and the y-axis oriented to complete a right handed coordinate system. The body coordinate frame is easily referenced to a north east down coordinate (NED) frame, sometimes referred to as a flat earth or local level model and in context of prior sections of this course was the inertial frame. An INS provides measurements relative to true or magnetic north (heading) and gravity (pitch and roll). The NED frame is then referenced to ECEF coordinates based upon latitude and longitude. For the example, with the gimbaled sensor configuration, the relative target position vector measured in the sensor LOS coordinate frame would rotate through the following sequence of 4 DCMs to obtain its orientation in the ECEF frame:

$$D\bar{C}M_{LOS}^{ECEF} = D\bar{C}M_{NED}^{ECEF}(\Phi_P, \Lambda_P) \cdot D\bar{C}M_{PLAT}^{NED}(\omega, \theta, \psi) \cdot D\bar{C}M_{GIMBAL}^{PLAT}(\alpha, \beta) \cdot D\bar{C}M_{LOS}^{GIMBAL}(\varepsilon x, \varepsilon y)$$

The notation uses a 'hat' to denote a matrix and a 'tilde' a vector. The subscript associated with a DCM denotes the source (from) coordinate frame while the superscript the destination (to) coordinate frame. Using this notation, the matrix inverse is denoted by changing the source to destination and destination to source (I.e. $\left(D\bar{C}M_A^B\right)^{-1} = D\bar{C}M_B^A$. Angles are defined as follows:

Φ_P: platform geodetic latitude (WGS84)
Λ_P: platform longitude (WGS84)
θ : platform pitch (platform INS)
ω : platform roll (platform INS)
ψ : platform heading (platform INS)
α : azimuth (gimbal)
β : elevation (gimbal)
Δx: measured sensor LOS x-target offset from center
εx = pixel $\Delta x * IFOV_{sensor}$ (for a camera)
Δy: measured sensor LOS y-target offset from center
εy = pixel $\Delta y * IFOV_{sensor}$ (for a camera)

Each DCM is the product of two or three rotations defined as follow. The LOS will be referenced to a sensor. The pointing error is defined in the LOS frame by the pointing error angle rotation matrix sequence product that defines the DCM from LOS to gimbal. The pointing vector rotated through this DCM into gimbal coordinates as:

$$D\bar{C}M_{LOS}^{GIMBAL}(\varepsilon x, \varepsilon y) = \begin{bmatrix} \cos(\varepsilon x) & -\sin(\varepsilon x) & 0 \\ \sin(\varepsilon x) & \cos(\varepsilon x) & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos(\varepsilon y) & 0 & \sin(\varepsilon y) \\ 0 & 1 & 0 \\ -\sin(\varepsilon y) & 0 & \cos(\varepsilon y) \end{bmatrix}$$

The pointing vector is then rotated through the gimbal DCM, defined for the simple two axis elevation over azimuth configuration, into platform coordinates. This DCM is the gimbal axes elevation and azimuth rotation matrix sequence product given by:

$$D\bar{C}M_{GIMBAL}^{PLAT}(\alpha, \beta) = \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) & 0 \\ \sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos(\beta) & 0 & \sin(\beta) \\ 0 & 1 & 0 \\ -\sin(\beta) & 0 & \cos(\beta) \end{bmatrix}$$

The platform motion relative to NED is defined by the platform axes yaw, pitch, and roll rotation matrix sequence product. The pointing vector rotates through this DCM into NED coordinates:

$$D\bar{C}M_{PLAT}^{NED}(\omega, \theta, \psi) = \begin{bmatrix} \cos(\psi) & -\sin(\psi) & 0 \\ \sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos(\theta) & 0 & \sin(\theta) \\ 0 & 1 & 0 \\ -\sin(\theta) & 0 & \cos(\beta) \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0) \\ 0 & \cos(\omega) & -\sin(\omega) \\ 0 & \sin(\omega) & \cos(\omega) \end{bmatrix}$$

The final pointing vector rotation is from NED to ECEF coordinates. This DCM is the longitude and latitude rotation matrix sequence product as:

$$D\bar{C}M_{NED}^{ECEF}(\Phi_P, \Lambda_P) = \begin{bmatrix} \cos(\Lambda_P) & -\sin(\Lambda_P) & 0 \\ \sin(\Lambda_P) & \cos(\Lambda_P) & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos(\Phi_P) & 0 & -\sin(\Phi_P) \\ 0 & 1 & 0 \\ \sin(\Phi_P) & 0 & \cos(\Phi_P) \end{bmatrix} \cdot \begin{bmatrix} 0 & 0 & -1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

The relationship between geodetic latitude measured by the GPS and geocentric latitude is:

$$\tan(\Phi_{geocentric}) = \left(\frac{R_{PL}}{R_{EQ}}\right)^2 \cdot \tan(\Phi)$$

The geo-pointing problem is locating a target vector in ECEF coordinates, given a target range measurement. The target vector in ECEF coordinates is the sum of the platform position in ECEF coordinates plus the measured relative target position vector, or:

$$\hat{P}_{TGT}^{ECEF} = \hat{P}_{PLAT}^{ECEF} + range \cdot D\bar{C}M_{LOS}^{ECEF} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

The DCM is determined based upon the measured gimbal angles, INS, and GPS data. Range equates to distance. For a short distance between platform and target (flat earth model applies), measurable with a rangefinder or triangulation, range is often used. For longer distances dependent on the earth curvature, models that account for curvature and obtain distance from differences in latitude and longitude are required; as will be discussed. Platform position in ECEF Cartesian coordinates is obtained from the vehicle geodetic latitude, longitude, and altitude as:

$$\hat{P}_{PLAT}^{ECEF} = \begin{bmatrix} \left(\dfrac{R_{EQ}}{\sqrt{1-e^2 \cdot \sin^2(\Phi)}} + \text{altitude}\right) \cdot \cos(\Phi_P) \cdot \cos(\Lambda_P) \\ \left(\dfrac{R_{EQ}}{\sqrt{1-e^2 \cdot \sin^2(\Phi)}} + \text{altitude}\right) \cdot \cos(\Phi_P) \cdot \sin(\Lambda_P) \\ \left(\left(\dfrac{R_{PL}}{R_{EQ}}\right)^2 \cdot \dfrac{R_{EQ}}{\sqrt{1-e^2 \cdot \sin^2(\Phi)}} + \text{altitude}\right) \cdot \sin(\Phi_P) \end{bmatrix}$$

Where the earth's semi-major axis R_{EQ} = 20, 925, 646 feet (earth's radius @ equator), R_{PL}=20, 850, 147.59 (earths radius @ pole) and the square of the earth's eccentricity e^2 = 0.00669438. To obtain the ECEF angular coordinates of the target, longitude can be obtained as:

$$\Lambda_T = \tan^{-1}\left(\frac{P_{TGTX}^{ECEF}}{P_{TGTY}^{ECEF}}\right)$$

There is not a direct solution for geodetic latitude but it can be determined iteratively with an initial estimate as:

$$\Phi_T \approx \tan^{-1}\left(\frac{P_{TGTZ}^{ECEF}}{\sqrt{(P_{TGTX}^{ECEF})^2 + (P_{TGTY}^{ECEF})^2}}\right)$$

There are also converters available on the internet that will do the x, y, z to latitude, longitude, and altitude conversion such as: https://www.mathworks.com/matlabcentral/fileexchange/7941-convert-cartesian-ecef-coordinates-to-lat-lon-alt. However as this is obtained from an iterative estimate there will be an associated error. Another application would be determining pointing angles for a targeted object position, given its GPS coordinates. The target location is:

$$\hat{P}_{TGT}^{ECEF} = \begin{bmatrix} \left(\dfrac{R_{EQ}}{\sqrt{1-e^2 \cdot \sin^2(\Phi_T)}} + \text{altitude tgt} \right) \cdot \cos(\Phi_T) \cdot \cos(\Lambda_T) \\ \left(\dfrac{R_{EQ}}{\sqrt{1-e^2 \cdot \sin^2(\Phi_T)}} + \text{altitude tgt} \right) \cdot \cos(\Phi_T) \cdot \sin(\Lambda_T) \\ \left(\left(\dfrac{R_{PL}}{R_{EQ}} \right)^2 \cdot \dfrac{R_{EQ}}{\sqrt{1-e^2 \cdot \sin^2(\Phi_T)}} + \text{altitude tgt} \right) \cdot \sin(\Phi_T) \end{bmatrix}$$

From the equation above for the target vector in ECEF coordinates, the unit target vector location in platform coordinates is obtained as:

$$\hat{P}_{TGT}^{PLAT} = D\bar{C}M_{ECEF}^{PLAT} \cdot \left(\frac{\hat{P}_{TGT}^{ECEF} - \hat{P}_{PLAT}^{ECEF}}{range} \right)$$

$$D\bar{C}M_{ECEF}^{PLAT} = D\bar{C}M_{NED}^{PLAT}(\omega, \theta, \psi) \cdot D\bar{C}M_{ECEF}^{NED}(\Phi_P, \Lambda_P)$$

Rotating this vector through gimbal coordinates; the desired perfect pointing vector is obtained by solving for the gimbal azimuth and elevation angles that satisfy the vector equality condition:

$$\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} = D\bar{C}M_{PLAT}^{GIMBAL}(\alpha, \beta) \cdot \hat{P}_{TGT}^{PLAT}$$

or

$$\hat{P}_{TGT}^{PLAT} = D\bar{C}M_{GIMBAL}^{PLAT}(\alpha, \beta) \cdot \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

Using the definition of the gimbal DCM for the simple two-axis El/Az gimbal configuration, the target vector must equate to:

$$D\bar{C}M_{GIMBAL}^{PLAT}(\alpha,\beta)\cdot\begin{bmatrix}1\\0\\0\end{bmatrix}=\begin{bmatrix}\cos(\alpha) & -\sin(\alpha) & 0\\ \sin(\alpha) & \cos(\alpha) & 0\\ 0 & 0 & 1\end{bmatrix}\cdot\begin{bmatrix}\cos(\beta) & 0 & \sin(\beta)\\ 0 & 1 & 0\\ -\sin(\beta) & 0 & \cos(\beta)\end{bmatrix}\cdot\begin{bmatrix}1\\0\\0\end{bmatrix}=\begin{bmatrix}\cos(\beta)\cdot\cos(\alpha)\\ \cos(\beta)\cdot\sin(\alpha)\\ -\sin(\beta)\end{bmatrix}$$

$$\therefore\quad P_{TGT}^{PLAT}=\begin{bmatrix}\cos(\beta)\cdot\cos(\alpha)\\ \cos(\beta)\cdot\sin(\alpha)\\ -\sin(\beta)\end{bmatrix}$$

The gimbal angles for perfect pointing are then obtained as:

$$\alpha=\tan^{-1}\left(\frac{P_{TGT,Y}^{PLAT}}{P_{TGT,X}^{PLAT}}\right)$$

$$\beta=\tan^{-1}\left(\frac{-P_{TGT,Z}^{PLAT}}{\sqrt{P_{TGT,X}^{PLAT\,2}+P_{TGT,Y}^{PLAT\,2}}}\right)$$

The algorithm for a geo-pointing solution then requires:

- Conversion of target and platform GPS locations to x, y, z vectors in the ECEF frame
- Measured range or calculated distance between target and platform GPS coordinates
- Calculating the vector difference to obtain the LOS pointing vector in the ECEF frame.
- Rotating the LOS ECEF pointing vector into a platform frame x, y, z vector.
- Solve for the azimuth and elevation angles that satisfy the perfect pointing equality.

Assuming the gimbal axes are aligned to the INS, these azimuth and elevation angles are the geo-pointing command angles. For a short range or distance between the targeted object and platform, the range can be assumed measurable. If this is not the case, then formulas that account for the earth's curvature must be used. The Haversine formula calculates the distance between two latitude, longitude coordinates as (4, 5):

$$\text{distance}(\Phi_P,\Lambda_P,\Phi_T,\Lambda_T)=$$
$$2\cdot a\cdot\sin^{-1}\left(\sqrt{\left(\sin^2\left(\frac{1}{2}\cdot(\Phi_T-\Phi_P)\right)+\cos\Phi_T\cdot\cos\Phi_P\cdot\sin^2\left(\frac{1}{2}\cdot(\Lambda_T-\Lambda_P)\right)\right)}\right)$$

The distance calculation is impacted by the shape of the earth impacts the distance calculation. A number of different distance approximation formulas exist for a flat surface; spherical surface; and ellipsoidal surface assumptions. The earth is fatter around the middle than poles so is ellipsoidal with the rotation about the minor axis. The WGS-84 spheroid reference datum is typically used for the purpose of navigation and WGS-84 altitude is used in calculating the ECEF vectors. The typical measurements required for geo-pointing are:

- GPS location of platform and of the target
  - GPS satellites (4 minimum)/antenna/receiver
- Platform orientation (heading, pitch, roll)
  - Inertial Navigation System (INS) or Attitude, Reference, Heading System (AHRS)
- Gimbal orientation of respective gimbal angles: encoders, resolvers, RVDT's, etc. (gimbal system needs to be aligned to INS)
- Sensor pointing error information

Geo-pointing Error: The measured target location will be corrupted by several sources of error. Referring back to the expression for the target location, this can be expressed symbolically by deviations (Δ-denotes error) to the nominal values as:

$$\hat{P}_{TGTcalc}^{ECEF} = (\hat{P}_{PLAT}^{ECEF} + \Delta\hat{P}_{PLAT}^{ECEF}) + (range + \Delta range)\cdot(D\bar{C}M_{LOS}^{ECEF} + \Delta D\bar{C}M_{LOS}^{ECEF})\cdot\begin{bmatrix}1\\0\\0\end{bmatrix}$$

or

$$\hat{E}_{TGT}^{ECEF} = \hat{P}_{TGTcalc}^{ECEF} - \hat{P}_{TGT}^{ECEF} = \Delta\hat{P}_{PLAT}^{ECEF} + (\Delta range\cdot D\bar{C}M_{LOS}^{ECEF} + range\cdot\Delta D\bar{C}M_{LOS}^{ECEF} + \Delta range\cdot\Delta D\bar{C}M_{LOS}^{ECEF})\cdot\begin{bmatrix}1\\0\\0\end{bmatrix}$$

The error vector equation can be expressed in a form that is a linear function of the deviations; ignoring the last second order error term, and assuming the delta DCM can be easily characterized in simple terms as:

$$\hat{E}_{TGT}^{ECEF} = \left[\bar{I}_{3x3}\ ,\ \left[D\bar{C}M_{LOS}^{ECEF}\cdot\begin{bmatrix}1\\0\\0\end{bmatrix}\right]_{3x1}\ ,\ range\cdot\bar{I}_{3x3}\right]\cdot\begin{bmatrix}\Delta\hat{P}_{PLAT}^{ECEF}\\ \Delta range\\ \left[\hat{\Delta}_{D\bar{C}M}\right]_{3x1}\end{bmatrix}\quad \text{(3x7 matrix : 7x1 vector)}$$

$$\text{with}\quad \left[\hat{\Delta}_{D\bar{C}M}\right]_{3x1} = \Delta D\bar{C}M_{LOS}^{ECEF}\cdot\begin{bmatrix}1\\0\\0\end{bmatrix}\ ;$$

The error measurement model can be used with a least squares estimator to determine a rough order of magnitude of the gross errors. Error evaluation at a component level would require expressing the model in terms of the actual component DCMs. In general, the geo-pointing errors or deviations can be categorized as those:

- associated with the measured angles in the DCM sequence due to the gimbal/platform sensors , and
- associated with the platform location in ECEF coordinates caused by GPS location accuracy relative to the pointing vector.

The errors relative to the DCM relate back to the pointing error budget and can be summarized as:

- Gimbal Pointing Error Budget (see section 1)
  - sensor noise and accuracy
  - gimbal alignment related to the track sensor
  - gimbal angle (encoder, resolver, etc.) noise and accuracy
  - gimbal rotation axis alignment, orthogonality
  - IMU accuracy and measurement noise (scaling, resolution, ARW, drift)
  - In general see section 1.0 pointing jitter and bias error budget
- INS accuracy and measurement noise
- GPS latitude and longitude angle errors
- INS to gimbal alignment (discussed below)
- INS to IMU alignment (discussed below**)**
- Quantization
- Measurement timing latencies, absolute or relative to each sensor or different sensor sample rates

The geo-pointing error will have bias and jitter components as with any pointing error budget and is a subset of the geo-location error substantiated in DCM deviations. In addition, the inertial sensors also exhibit random and bias components as described in section 2.0. The INS will have internal misalignment as well as misalignment relative to the gimbal. Internal misalignment is specified by the manufacturer for any tactical grade sensor or better and can be compensated for to some degree. But there is also a mechanical mounting misalignment between the sensors and gimbal which must be accounted for. This misalignment, however, is measurable to some tolerance and can be corrected by alignment processes that will be described shortly. Quantization produces a white noise on any output proportional to the magnitude of the quantization error. The errors that determine the accuracy of the platform ECEF vector depend on the accuracy and noise associated with the GPS estimates of latitude, longitude and altitude. This can also impact the distance calculation although for short range it could be a rangefinder integrated with the gimbaled sensor. These sources of error may require sophisticated compensation techniques to obtain the geo-pointing and geo-location accuracy desired. To minimize their impact on geo-location, sensor data can fused using an Extended Kalman Filter (EKF), a short description of which is provided in section 4.0. With the filter and known noise distributions, geo-location estimate errors can be significantly reduced. Most INS/GPS, now even of the MEMS tactical grade, contain EKF navigation model for obtaining the best estimates of platform angles, position, and velocity.

A metric is required to evaluate position accuracy. As error sources that have the greatest effect are predominantly random; a probability distribution describing the random process; as discussed in section 1.0 for gimbal pointing jitter and bias, is useful. A simple metric often used is the circular error probable (CEP) geo-location metric to quantify performance. This is a measure of a systems locating accuracy. It is defined as the radius of a circle, centered on a mean, whose boundary is defines the region in which a pointing estimate falls 50% of the time or effectively the median error radius. The mean offset relative to the actual targeted point is the bias. Note the similarity of this definition with pointing bias and jitter described in section 1.0. The concept of CEP is a metric used when measuring the accuracy of a position obtained by a GPS navigation system. There are other associated metrics such as the DRMS (distance root mean square), which is the square root of the average squared distance error, and R95, which is the radius of the circle where 95% of the values would fall.

Another technique to improve geo-location accuracy is using geo-registration of video sequences to a 3D terrestrial landscape geo-referenced image. This has been applied many applications as it can shift the burden of precision pointing from inertial angle and rate sensors to image registration between a sensed and geo-referenced image. Precision inertial pointing can require navigation grade inertial rate and orientation sensors which significantly impact SWaP and cost. Video geo-registration still requires these sensors for coarse pointing but with much less precision allowing for a lower performance grade inertial sensor. These lower grade inertial sensors require much less SWaP and cost significantly less. One obvious application for this approach is geo-location using a mini-UAV where the quality of the inertial sensors is poor due to space, weight, and power (SWaP) constraints. The video geo-registration approach to geo-location will be described in section 4.0. It is important to bear in mind however that extracting data from an image is no panacea when one considers the near infinite variation in image scene data due to terrain and illumination available on the planet.

Geo-Pointing Alignment Calibration: For nearly all geo-locating applications, stabilized accurate point is required. Some applications that allow for post-processing of the information, platform motion may be measured and then subtracted from the recorded geographical information. In any case, accurate alignment of the between system sensors and the sensor platform is necessary. With gimbaled pointing sensors, aligning the platform INS with the gimbal IMU is very important. The INS could be pre-installed in the platform vehicle and located remote from the gimbal IMU location. The procedure for doing this is often termed transfer alignment. Other applications may require direct alignment between the gimbal and INS. The key to any alignment procedure is to identify a common measurement between the two sensors, determine the measurement error between the two, and finally design an alignment algorithm that corrects for the errors. Alignments in general are a major element of any pointing system design. Two simple examples are provided; however for an actual system there will generally be several more alignment correction parameters and terms required; effectively one for every gimbal rotation axis, inertial sensing component, and the track sensor.

INS to IMU Transfer Alignment: The IMU provides only angular rates and linear accelerations. The IMU needs the vehicle's heading to North, pitch and roll measured by the vehicle INS transferred from the INS to IMU for inertial orientation reference. The alignment transfer, termed transfer alignment, is performed by measuring information common to both sensors, inertial rates, and using a least square fit to the data to obtain the alignment errors or displacements between the two components. These displacements are used to populate an alignment matrix that relates measurements in one coordinate frame to the other. A general algorithm for generating an alignment matrix relating the equivalent angles or angular rates measured by two sensors A and B can be obtained as (using angular rate, ω notation):

$$\hat{\omega}_B(t) = \bar{C}_{ALIGN} \cdot \hat{\omega}_A(t)$$

The equation can be expanded to provide three equations in terms of each row vector of the alignment matrix as:

$$\omega_{Bx} = c_{x1} \cdot \omega_{Ax} + c_{x2} \cdot \omega_{Ay} + c_{x3} \cdot \omega_{Az} = \hat{c}_x^T \cdot \hat{\omega}_A$$
$$\omega_{By} = c_{y1} \cdot \omega_{Ax} + c_{y2} \cdot \omega_{Ay} + c_{y3} \cdot \omega_{Az} = \hat{c}_y^T \cdot \hat{\omega}_A$$
$$\omega_{Bz} = c_{z1} \cdot \omega_{Ax} + c_{z2} \cdot \omega_{Ay} + c_{z3} \cdot \omega_{Az} = \hat{c}_z^T \cdot \hat{\omega}_A$$

The alignment vectors are independent so the each equation represents a linear relationship between the alignment matrix row vector and the INS rate data producing the equivalent IMU rate. There will be r-measurements performed so that for each equation one has the following relationship for the Ith measurement:

$$\omega_{Bx,i} = \hat{\omega}_{A,i} \cdot \hat{c}_x$$
$$\omega_{By,i} = \hat{\omega}_{A,i} \cdot \hat{c}_y$$
$$\omega_{Bz,i} = \hat{\omega}_{A,i} \cdot \hat{c}_z$$

This set of equations can be solved using a least square estimator to provide a best linear fit to the data to estimate the alignment matrix parameters. Define the following vectors:

$$\hat{V}_{Bx} = \begin{bmatrix} \omega_{Bx,1} \\ . \\ . \\ \omega_{Bx,r} \end{bmatrix}; \hat{V}_{By} = \begin{bmatrix} \omega_{By,1} \\ . \\ . \\ \omega_{By,r} \end{bmatrix}; \hat{V}_{Bz} = \begin{bmatrix} \omega_{Bz,1} \\ . \\ . \\ \omega_{Bz,r} \end{bmatrix}; \bar{U}_A = \begin{bmatrix} \hat{\omega}_{A,1}^T \\ . \\ . \\ \hat{\omega}_{A,r}^T \end{bmatrix}$$

The least square or regression solution for each parameter vector c_j (j=x,y,z) which defines a row in the alignment matrix is given by:

$$\hat{c}_x^* = [\bar{U}_A^T \cdot \bar{U}_A]^{-1} \cdot \bar{U}_A^T \cdot \hat{V}_{Bx}$$
$$\hat{c}_y^* = [\bar{U}_A^T \cdot \bar{U}_A]^{-1} \cdot \bar{U}_A^T \cdot \hat{V}_{By}$$
$$\hat{c}_z^* = [\bar{U}_A^T \cdot \bar{U}_A]^{-1} \cdot \bar{U}_A^T \cdot \hat{V}_{Bz}$$

This set of equations can then be expressed in a final matrix form as:

$$\bar{C}_{ALIGN} = \begin{bmatrix} \hat{c}_x^* \\ \hat{c}_y^* \\ \hat{c}_z^* \end{bmatrix} = [\bar{U}_A^T \cdot \bar{U}_A]^{-1} \cdot \bar{U}_A^T \cdot \bar{V}_B$$

$$where \;\; \bar{V}_B = \begin{bmatrix} \hat{V}_{Bx} & \hat{V}_{By} & \hat{V}_{Bz} \end{bmatrix}$$

This form of the least square parameter estimation equation is sometimes referred to as the batch least square estimator. The general alignment matrix described can be constrained to a perturbation format for axis alignment consistent with the nominal axis value and its perturbations orthogonal to that axis. The diagonal terms of the alignment matrix are set equal to one (assume scale factors for each axis equal) and the off-diagonal terms are the alignment perturbation errors resulting in a skew-symmetric matrix. Using this format and changing the general sensor notation such that A-INS and B-IMU the alignment matrix values are given by:

$$c_{x1}=c_{y2}=c_{z3}=1$$
$$c_{y3}=-c_{z2}= \delta x_{INS}$$
$$c_{x2}=-c_{y1}= \delta z_{INS}$$
$$c_{x3}=-c_{z1}= -\delta y_{INS}$$

The variables δx_{INS}, δy_{INS}, δz_{INS} are the perturbation alignment angle errors associated with each respective axis. The previous set of matrix equations can then be expressed as:

$$\omega_{IMUx} = \omega_{INSx} + \delta z_{INS} \cdot \omega_{INSy} - \delta y_{INS} \cdot \omega_{INSz}$$
$$\omega_{IMUy} = -\delta z_{INS} \cdot \omega_{INSx} + \omega_{IMUy} + \delta x_{INS} \cdot \omega_{INSz}$$
$$\omega_{IMUz} = \delta y_{INS} \cdot \omega_{INSx} - \delta x_{INS} \cdot \omega_{INSy} + \omega_{INSz}$$

or

$$\omega_{IMUx} - \omega_{INSx} = \delta z_{INS} \cdot \omega_{INSy} - \delta y_{INS} \cdot \omega_{INSz}$$
$$\omega_{IMUy} - \omega_{INSy} = -\delta z_{INS} \cdot \omega_{INSx} + \delta x_{INS} \cdot \omega_{INSz}$$
$$\omega_{IMUz} - \omega_{INSz} = \delta y_{INS} \cdot \omega_{INSx} - \delta x_{INS} \cdot \omega_{INSy}$$

finally

$$\begin{bmatrix} \omega_{IMUx} - \omega_{INSx} \\ \omega_{IMUy} - \omega_{INSy} \\ \omega_{IMUz} - \omega_{INSz} \end{bmatrix} = \begin{bmatrix} 0 & -\omega_{INSz} & \omega_{INSy} \\ \omega_{INSz} & 0 & -\omega_{INSx} \\ -\omega_{INSy} & \omega_{INSx} & 0 \end{bmatrix} \cdot \begin{bmatrix} \delta x_{INS} \\ \delta y_{INS} \\ \delta z_{INS} \end{bmatrix}$$

This expression is actually solvable since there are three equations and three unknowns. However a least square solution based upon several sets of measurements is still best given noisy measurements. The equation now has the general form:

$$\Delta\hat{\Omega} = \bar{W}_{INS} \cdot \hat{\delta}_{IMU:INS}$$

where

$$\Delta\hat{\Omega} = \begin{bmatrix} \omega_{IMUx} - \omega_{INSx} \\ \omega_{IMUy} - \omega_{INSy} \\ \omega_{IMUz} - \omega_{INSz} \end{bmatrix} \quad ; \quad \bar{W}_{INS} = \begin{bmatrix} 0 & -\omega_{INSz} & \omega_{INSy} \\ \omega_{INSz} & 0 & -\omega_{INSx} \\ -\omega_{INSy} & \omega_{INSx} & 0 \end{bmatrix} \quad ; \hat{\delta}_{IMU:INS} = \begin{bmatrix} \delta x_{INS} \\ \delta y_{INS} \\ \delta z_{INS} \end{bmatrix}$$

For r-measurements, each measurement set appends to the previous vector or matrix so using similar notation:

$$\hat{V}_{IMUr} = \begin{bmatrix} \Delta\hat{\Omega}_1 \\ \cdot \\ \cdot \\ \Delta\hat{\Omega}_r \end{bmatrix} ; \bar{U}_{INSr} = \begin{bmatrix} \bar{W}_{INS,1} \\ \cdot \\ \cdot \\ \bar{W}_{INS,r} \end{bmatrix}$$

$$\hat{V}_{IMUr} \text{ - 3r - element vector} \quad ; \quad \bar{U}_{INSr} \text{ - 3r x 3 element matrix}$$

The error parameter vector for the r-th measurement is then:

$$\hat{\delta}_{IMU:INSr} = [\bar{U}_{INSr}^T \cdot \bar{U}_{INSr}]^{-1} \cdot \bar{U}_{INSr}^T \cdot \hat{V}_{IMUr}$$

A quick dimensionality check helps to verify the solution. The inverse matrix is 3x3, the U^T_{INS} transpose is 3x3r and the V_{IMU} vector 3rx1 whose product results in a 3x1 vector, δ, as desired. In addition, the matrix $U_{INS}^T * U_{INS}$ is always a 3x3 matrix and its elements are a summation of measurement terms of the same form. Similarly the vector $U_{INS}^T * V_{IMU}$ is always a 3 element vector with its elements the summation of measurement terms of the same form. So matrix elements can be created as a running sum of the same element from all measurements with each new measurement added to the sum. To see this, the matrix $U_{INS}^T * U_{INS}$ and vector $U_{INS}^T * V_{IMU}$ can be expanded for two measurements with the sum observed in each term of the matrix expansion as:

$$\bar{U}_{INS2}^T \cdot \bar{U}_{INS2} =$$

$$\begin{bmatrix} -\left(\omega_{INSy1}{}^2 + \omega_{INSz1}{}^2 + \omega_{INSy2}{}^2 + \omega_{INSz2}{}^2\right) & \left(\omega_{INSx1} \cdot \omega_{INSy1} + \omega_{INSx2} \cdot \omega_{INSy2}\right) & \left(\omega_{INSx1} \cdot \omega_{INSz1} + \omega_{INSx2} \cdot \omega_{INSz2}\right) \\ \left(\omega_{INSx1} \cdot \omega_{INSy1} + \omega_{INSx2} \cdot \omega_{INSy2}\right) & -\left(\omega_{INSx1}{}^2 + \omega_{INSz1}{}^2 + \omega_{INSx2}{}^2 + \omega_{INSz2}{}^2\right) & \left(\omega_{INSz1} \cdot \omega_{INSy1} + \omega_{INSz2} \cdot \omega_{INSy2}\right) \\ \left(\omega_{INSx1} \cdot \omega_{INSz1} + \omega_{INSx2} \cdot \omega_{INSz2}\right) & \left(\omega_{INSz1} \cdot \omega_{INSy1} + \omega_{INSz2} \cdot \omega_{INSy2}\right) & -\left(\omega_{INSy1}{}^2 + \omega_{INSx1}{}^2 + \omega_{INSy2}{}^2 + \omega_{INSx2}{}^2\right) \end{bmatrix}$$

The vector expansion is:

$$\bar{U}_{INS2}^T \cdot \hat{V}_{IMU2} = \begin{bmatrix} -\omega_{INSz1} \cdot V_{IMUy1} + \omega_{INSy1} \cdot V_{IMUz1} - \omega_{INSz2} \cdot V_{IMUy2} + \omega_{INSy2} \cdot V_{IMUz2} \\ \omega_{INSz1} \cdot V_{IMUx1} - \omega_{INSx1} \cdot V_{IMUz1} + \omega_{INSz2} \cdot V_{IMUx2} - \omega_{INSx2} \cdot V_{IMUz2} \\ -\omega_{INSy1} \cdot V_{IMUx1} + \omega_{INSx1} \cdot V_{IMUy1} - \omega_{INSy2} \cdot V_{IMUx2} + \omega_{INSx2} \cdot V_{IMUy2} \end{bmatrix}$$

To obtain a valid solution, it is important that all rates are sufficiently excited otherwise there will not be sufficient information to determine the correct alignment errors. Measurements should be made from a moving platform or vehicle. Given a valid solution, then the δ errors are used to populate the misalignment matrix and the relationship of measurements made by the INS can be transferred between the INS and the IMU. INS heading, pitch, roll vector can be translated to the IMU coordinates and the IMU heading, pitch, and roll updated with the IMU rates and accelerations between alignment transfer updates. Alignment transfer updates should be performed periodically based upon IMU drift rate.

Platform to INS Alignment: A second alignment example is provided; this case being between an INS and gimbal mounted on the same platform vehicle. The common measurement point will be target located at a known location as defined by GPS data in NED coordinates and tracked by a sensor mounted on a gimbal. The gimbal configuration will be that assumed for prior examples; simple 2 axis el/az design. Referring to the previous geo-pointing examples, the expression for determining the pointing vector between two GPS measured locations can be written as:

$$\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} = D\bar{C}M_{PLAT}^{GIMBAL} \cdot D\bar{C}M_{NED}^{PLAT} \cdot D\bar{C}M_{ECEF}^{NED} \cdot \left(\frac{\hat{P}_{TGT}^{ECEF} - \hat{P}_{PLAT}^{ECEF}}{range} \right)$$

This condition holds, as derived earlier, for the azimuth and elevation angles obtained for perfect pointing based upon the measured platform angles. However a misalignment between the INS and platform results in a pointing error and this condition can be expressed as:

$$\begin{bmatrix} 1 \\ \varepsilon_Y \\ \varepsilon_Z \end{bmatrix} = D\bar{C}M_{PLAT}^{GIMBAL} \cdot \delta DCM_{P\,INS}^{P\,GIMBAL} \cdot D\bar{C}M_{NED}^{PLAT} \cdot D\bar{C}M_{ECEF}^{NED} \cdot \left(\frac{\hat{P}_{TGT}^{ECEF} - \hat{P}_{PLAT}^{ECEF}}{range} \right)$$

The misalignment matrix is given by:

$$\delta D\bar{C}M_{P\,INS}^{P\,GIMBAL} = \begin{bmatrix} 1 & -\delta z_P & \delta y_P \\ \delta z_P & 1 & -\delta x_P \\ -\delta y_P & \delta x_P & 1 \end{bmatrix} = \bar{I} + \Delta D\bar{C}M_{P\,INS}^{P\,GIMBAL}$$

where $\Delta D\bar{C}M_{P\,INS}^{P\,GIMBAL} = \begin{bmatrix} 0 & -\delta z_P & \delta y_P \\ \delta z_P & 0 & -\delta x_P \\ -\delta y_P & \delta x_P & 0 \end{bmatrix}$

The last three terms can be combined into an inertial or NED referenced pointing vector; known because we have measured INS angles (requires INS/platform motion) and the ECEF vector positions were known from GPS or:

$$\hat{D}P^P = D\bar{C}M_{NED}^{PLAT} \cdot D\bar{C}M_{ECEF}^{NED} \cdot \left(\frac{\hat{P}_{TGT}^{ECEF} - \hat{P}_{PLAT}^{ECEF}}{range} \right)$$

Substituting into the original expression:

$$\begin{bmatrix} 1 \\ \varepsilon_Y \\ \varepsilon_Z \end{bmatrix} = D\bar{C}M_{PLAT}^{GIMBAL} \cdot \left(\bar{I} + \Delta D\bar{C}M_{P\,INS}^{P\,GIMBAL} \right) \cdot \hat{D}P^{p}$$

The platform to gimbal DCM is populated by the azimuth and elevation gimbal command angles obtained for solving the perfect pointing condition without any assumed misalignment. An optical sensor mounted on the gimbal (i.e. as shown in section 1.0) measures the ε_Y ε_Z errors which are the aim point deviations relative to the target. This expression can then be written in the form defining a relationship between an input and output vector linearly related by the misalignment parameter error matrix as:

$$D\bar{C}M_{GIMBAL}^{PLAT} \cdot \begin{bmatrix} 1 \\ \varepsilon_Y \\ \varepsilon_Z \end{bmatrix} - \hat{D}P^{p} = \Delta D\bar{C}M_{P\,INS}^{P\,GIMBAL} \cdot \hat{D}P^{p}$$

$$\text{defining } \Delta\hat{\Omega} = D\bar{C}M_{GIMBAL}^{PLAT} \cdot \begin{bmatrix} 1 \\ \varepsilon_Y \\ \varepsilon_Z \end{bmatrix} - \hat{D}P^{p}$$

$$\therefore \qquad \Rightarrow \qquad \Delta\hat{\Omega} = \Delta D\bar{C}M_{P\,INS}^{P\,GIMBAL} \cdot \hat{D}P^{p} \qquad \Leftarrow$$

The output vector Ω, and the output vector DP are both obtained from measurement and calculation. The expression is now taking the form used to obtain a LSE solution for the INS to IMU transfer alignment. Expand the right side:

$$\Delta\hat{\Omega} = \begin{bmatrix} 0 & -\delta z_P & \delta y_P \\ \delta z_P & 0 & -\delta x_P \\ -\delta y_P & \delta x_P & 0 \end{bmatrix} \cdot \hat{D}P^{P} = \begin{bmatrix} \delta y_P \cdot DP_Z^P - \delta z_P \cdot DP_Y^P \\ \delta z_P \cdot DP_X^P - \delta x_P \cdot DP_Z^P \\ \delta x_P \cdot DP_Y^P - \delta y_P \cdot DP_X^P \end{bmatrix}$$

$$\Delta\hat{\Omega} = \begin{bmatrix} 0 & DP_Z^P & -DP_Y^P \\ -DP_Z^P & 0 & DP_X^P \\ DP_Y^P & -DP_X^P & 0 \end{bmatrix} \cdot \begin{bmatrix} \delta x_P \\ \delta y_P \\ \delta z_P \end{bmatrix}$$

$$\text{defining} \quad \hat{\delta}_{PLAT:INS} = \begin{bmatrix} \delta x_P \\ \delta y_P \\ \delta z_P \end{bmatrix} \quad ; \quad D\bar{P}M = \begin{bmatrix} 0 & DP_Z^P & -DP_Y^P \\ -DP_Z^P & 0 & DP_X^P \\ DP_Y^P & -DP_X^P & 0 \end{bmatrix}$$

finally results in an equation for the alignment error

$$\Delta\hat{\Omega} = D\bar{P}M \cdot \hat{\delta}_{PLAT:INS}$$

This is the same form of the expression used for the transfer alignment. Now define:

$$\hat{V}_{PLATr} = \begin{bmatrix} \Delta\hat{\Omega}_1 \\ . \\ . \\ \Delta\hat{\Omega}_r \end{bmatrix} ; \bar{U}_{INSr} = \begin{bmatrix} D\bar{P}M_1 \\ . \\ . \\ D\bar{P}M_r \end{bmatrix}$$

$\hat{V}_{PLATr}$ - 3r - element vector ; $\bar{U}_{INSr}$ - 3r x 3 element matrix

The LSE solution is then obtained as:

$$\hat{\delta}_{PLAT:INSr} = [\bar{U}^T_{INSr} \cdot \bar{U}_{INSr}]^{-1} \cdot \bar{U}^T_{INSr} \cdot \hat{V}_{PLATr}$$

The INS data needs to be obtained while on-the-move with the platform vehicle creating yaw, pitch, and roll motion. The errors are obtained by an optical sensor that measures actual aim point versus commanded aim point. Note that with an actual system there will be several other sources of misalignment. This is just a simple example to demonstrate the misalignment measurement process.

Summary Section 3.0 Key Points

- ***Definition of key coordinate frames used for geo-pointing are provided***
- ***The DCM sequence between ECEF coordinates and LOS coordinates is shown***
- ***The geo-pointing vector between a geo-referenced platform location and a target location is derived***
- ***Pointing angles between two geo-referenced locations, platform and target are determined***
- ***Disturbances are described; jitter and bias associated with pointing as discussed in section 1.0 plus the addition of inertial positioning errors***
- ***LOS pointing alignment calibration is discussed and methods to determine and compensate for misalignment***

4.0 Geo-location (6, 7, 8, 9, 10, 11)

Geo-location, like many technological areas, covers a wide range of subjects; many of which are technologies in themselves. The goal of this section is to review some applications, significant processing approaches, and technologies required to implement an application solution. In general, geo-location will always be dependent on geo-pointing, either directly or to initialize image registration. Geo-location was basically described in section 3.0 by the determination of a 'point' location in ECEF coordinates; latitude, longitude, and altitude. A point being a small geographic area relative to a region of interest. This could even be a mobile phone or iPhone whose IP address registers with a geographic location that maps to regional geographic coordinates. This section will focus on geo-locating in ECEF coordinates, and primarily in the context of an image as opposed to a point, requiring the alignment of the sensed image with one already geo-located. This process is termed geo-registration; the alignment of the sensed frame of image data with a geo-located (latitude, longitude, and elevation based) calibrated reference image. There are a multitude of applications for geo-registering an image. For example, aligning a sensed image of even a small region with a geo-referenced map may provide a more accurate geo-location than attainable with a geo-pointing system alone as described in section 3.0. This would depend on the application and inertial sensors used. The information within the image may also be of significant value so image quality beyond that required for registration may be critical. For example, use of image information within the transportation infrastructure for highway and bridge maintenance management, damage and structural deterioration assessment, traffic pattern analysis and control, etc. is a growing application. Similarly, monitoring trends in foliage and vegetation patterns, flood and drought conditions, deforestation status, fire prevention and hazard condition awareness, and soil erosion patterns all utilize sensed scene image information. Geo-registration will be described in more detail. Some acronyms and definitions relevant to geo-location and geo-registration in particular are provided in Table 1.0 with websites used for reference listed below each.

Table 1.0 Brief List of Geo-Locating Relevant Acronyms and Definitions

1	GIS:	Geographic Information System	all functions related to acquisition, reduction, processing, geographic image and location data
	https://en.wikipedia.org/wiki/Geographic_information_system		
2	GNSS	Global Navigation Satellite System	term for satellite navigation systems that provide autonomous geo-spatial positioning with global coverage; the GPS, GLONASS, Galileo, etc.
	https://www.semiconductorstore.com/blog/2015/What-is-the-Difference-Between-GNSS-and-GPS/1550/		
3	GPS	Global Positioning System	US based system with up to 32 medium Earth orbit satellites
	Reference same as table entry 2		
4	NA	Orthographic projection	Orthogonal projection of a 3D image to a 2D representation of the image
	https://en.wikipedia.org/wiki/Orthographic_projection		
5	NA	Geo-registration	alignment of an unreferenced GIS image with a geographically registered GIS image
	http://www.georeference.org/doc/georegistration.htm		

6	NA	Geodetic datum	A coordinate system with reference points to positions on earth; such as the World Geodetic System WGS 84
	https://en.wikipedia.org/wiki/Geodetic_datum		
7	ECEF	Earth-Centered, Earth-Fixed	Geographic Cartesian coordinate system with origin defined as the earth center of mass and coordinate axes fixed to the rotating earth. See section 3.0
	https://en.wikipedia.org/wiki/ECEF		
8		Geo-reference:	Relating a ground system of geographic coordinates with a local map coordinate system or aerial photo image.
	https://en.wikipedia.org/wiki/Georeferencing		
9		Geo-rectify	Project an image, mapping it to a known geo-referenced coordinate system.
	https://imageryspeaks.wordpress.com/2012/01/24/georeferencing-vs-georectification-vs-geocoding/		
10		Ortho-rectify:	Accurately adjusting image data to a geo-referenced coordinate system, with distortions due to topographic variation corrected.
	Reference same as table entry 10		
11		Ortho-image	A geometrically corrected aerial or satellite image such that the scale is uniform for a given map projection.
	https://en.wikipedia.org/wiki/Orthophoto		
12		Ortho-photographs	Used in GIS as an accurate geo-referenced map background image.
	Reference same as table entry 11		

Geo-location System Architecture and Requirements (6, 7):

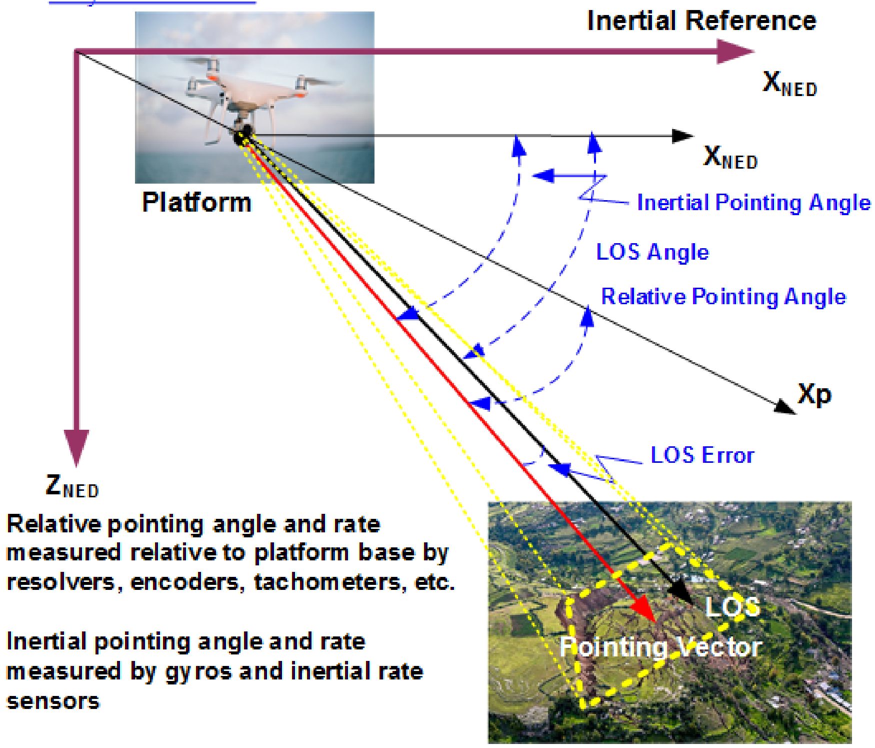
General Geo-location: Geo-location registration approaches are examined to provide an overview of the technology and the required algorithm processing requirements. There are two general geo-registration techniques; direct geo-registration and image geo-registration. Direct is effectively the geo-pointing problem described in section 3.0 requiring navigation grade inertial sensors. Image geo-registration compares a sensed image with a precision geo-referenced image of the same region. This method is a significant part of many geo-location algorithms so the discussion will focus on this method. The image processing requirements are generally intensive and can vary significantly depending on the amount of available a priori information which is often dependent on application. Processing usually consisting of several steps which in general fall into some form of feature and/or image correlation processing algorithm(s). With the advancements in artificial intelligence, image processing and registration algorithms available and in development are escalating quickly. The platform size can constrain the sensor and pointing architecture limiting data quality and quantity. Large aircraft or platforms, are less constrained, and can carry payloads with high precision inertial sensors, high resolution cameras and multi-axis pointing gimbals. A small UAV may have a payload capacity for only a small low resolution camera and tactical or less grade inertial sensors. Application requirements will drive the sensor choice which could include a variety of cameras with performance characteristics chosen based upon:

- Sensor type and performance characteristics
    - FOV
    - Resolution (defined by FOV and pixel or line density)
    - frame rate
    - spectral band
- Sensor Data Acquisition and Storage
- Sensor Data Rate and Processing
- Sensor LOS Control; pointing and stabilization (platform motion/vibration)
- Processing requirements

Specialty sensors such as hyper-spectral imaging sensor, high resolution mega-pixel cameras, synthetic aperture radar (SAR), and LIDAR are also used in geo-locating applications. Large airborne platforms can accommodate precision inertial and optical sensor suites and will generally use a fully stabilized gimbaled pointing system to direct the sensor field of view. However even with large platforms, as sensor suite payload SWaP grows, stabilized gimbal size can get quite large so mounting the sensors below the gimbal base on the platform or an outer axis and using mirrors to direct the optical path through the gimbal structure becomes a viable alternative. This was briefly described in Part 1,0 of the course. Image geo-registration is used in many applications [6] that require geo-referenced sensed imagery. As mentioned previously, surveillance, navigation, targeting and GIS related data collection use geo-located imagery. The military use of image geo-registration is primarily focused on geo-location for targeting, navigation, and surveillance. Generating high accuracy surveillance patterns, navigation routes as well as target location all require precise and fast geo-registration methods that integrate GPS and inertial measurements with high speed geo-registering processing methods to generate geo-registered images. Using video imagery and a GPS system, unmanned aircraft sensors can geo-register images to accurately locate features of interest. Using GIS related data, geo-registration can correlate data sets from many environmental, geological, and behavioral studies. Geo-referenced sensed imagery is an important by-product of image geo-registration.

Inertial scanning techniques, used with either geo-registration method, are also of critical importance especially when sensors have a narrow field of view (FOV) to providing high resolution but a wide angular coverage or field of regard (FOR) is desired for geo-location so scanning the wide FOR with a narrow FOV sensor is used to satisfy the requirement. For example, a narrow FOV can be inherent in a LIDAR system which may scan with a narrow laser beam over a wide receiver sensor FOV or scan both the laser narrow divergence and matching sensor FOV over the wide FOR. Scanning systems must provide patterns that insure uninterrupted scene coverage in the local or NED coordinates, no pattern gaps in the projected pattern in these coordinates. Minimizing lost data usually requires scan pattern overlap on the order of 20%. For this reason they need to know platform velocity and often coordinate the pattern for it. Scan patterns relative to stationary coordinates such as raster, rosette, and step-stare need to be adjusted once integrated with a moving platform; often using the platform forward motion to assist with pattern coverage. With these adjustments, the patterns are often referred to by their unique motion relative to the platform and are normally coordinated with platform velocity such as ‘racetrack’ scan pattern rectilinear swaths, whiskbroom, and push broom. For every data image taken, platform location and orientation and pointing gimbal orientation is required to map and stich sensed image frames to an image of the full field of regard. Figure 10.0 illustrates the general geo-location geo-registration operational concept.

https://www.pexels.com/photo/quadcopter-flying-on-the-skey-1034812/ archived on 20 May 2018 at the Wayback Machine



https://www.flickr.com/photos/ministeriodedefensaperu/39935939755/in/dateposted/

Figure 10.0 General Geo-location Operational Concept

Direct Geo-registration Geo-location [7]: Before delving into the details associated with image processing and registration, an approach termed direct geo-registration is discussed. In essence, this is the ground target geo-pointing problem based on pointing geometry described in section 3.0. Operation requires acquisition and track of a targeted object so the camera centers its FOV on the target object thereby establishing a LOS vector from the camera center to the ground object that defines the sensor LOS geometry. Given measured range or estimated from a terrain digital elevation map (altitude, pointing angle and flat earth assumption), the object position vector in ECEF Cartesian coordinates can be determined. The algorithm requires rotation of the LOS vector through the sequence of DCM transformations described in section 3.0; and the estimated ECEF target coordinates. The platform position is obtained from an on-board GPS receiver; platform INS attitude angles, gimbal orientation from angle measurement sensors (resolvers or encoders), and the camera sensor error, all of which are used for the DCM calculation. The problem with the approach is that the measurement of the platform position, attitude, gimbal and camera angles are affected by several error sources, as discussed in section 3.0 which can result in a ground target location error exceeding the desired location accuracy. However given these errors are compensated for; Direct Geo-registration is a viable approach. Effectively knowing an accurate geo-location of the center of the sensed image, a camera model can be used to align each image pixel to geo-reference. Geo-location implemented using direct geo-registration requires very accurate differential GPS position data, a navigation grade INS/IMU for platform orientation data, a precision time base, as well as GPS aided inertial navigation software [7]. A typical measurement processing architecture to obtain precise data geo-location estimates is shown in Figure 11.0. The system shown [7], does not use an INS, reducing cost and SWAP, but uses a high quality grade IMU and SW to implement the INS functionality. The key real time software elements are shown within the blue rectangle in Figure 11.0 [7]: The navigation model is effectively a software implementation of the block diagram shown in Figure 8.0. It generates a solution in real time to the navigation equations; integrating acceleration and angular rates sensed by the IMU. Initialization is required with known position and velocity from the GPS aligned with respect to the true vertical and true North. An error model is implemented by a Kalman Filter [7]; using the navigation model and IMU errors driving the navigation model performance. Differences between the position from the navigation model and the position from the GPS are processed in the Kalman filter to estimate position error from the navigation model. The error controller applies Kalman filter generated estimates of IMU sensor errors to the IMU-measured incremental angles and velocities prior to integration, which effectively calibrates the IMU sensors. The resulting navigation position and velocity solution, output in real time, is accuracy limited by the absolute accuracy of the GPS position and velocity. The Smoother and data controller are used for post-processing of the geo-location information, applying optimal estimates of the navigation model and IMU sensor errors to the recorded data; improving accuracy of the location estimates [7]. Purchasing an inertial grade INS will provide the same functionality with additional processing; however there is significantly greater cost and SWAP with the INS. Although strides are being made towards reducing high grade inertial sensors SWaP; those with navigation type performance are still fairly large limiting their use and that of a direct geo-registration approach to larger platforms. An advantage of the approach described, using the IMU, is it may be applied to smaller platform applications. Two companies that offer hardware and software applicable to this approach are https://www.applanix.com/ ; with complete packages for direct geo-location and https://www.novatel.com/ with products more specific to GNSS functions.

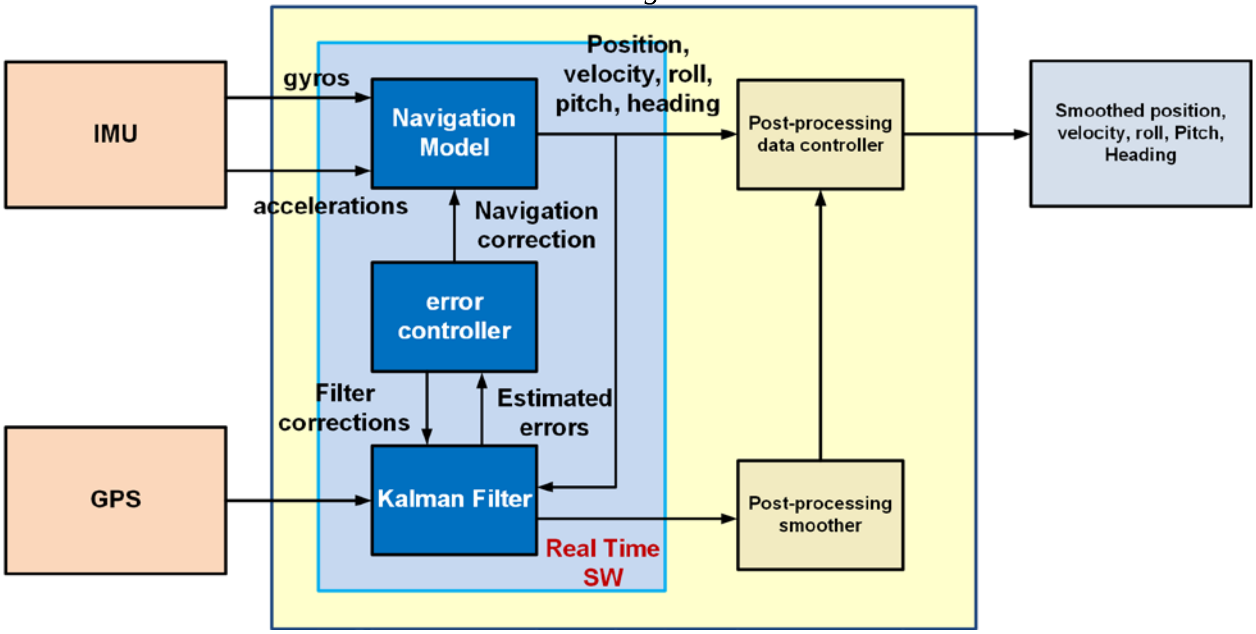


Figure 11.0 [7] High Accuracy Inertial Data for Direct Geo-registration

Image Geo-registration Geo-location [8, 9, 10, 11]: This technique for geo-location provides the potential for improved geo-location accuracy as well as application dependent information derived from the scene imagery. Geo-location using image geo-registration is based upon aligning and matching a camera image with a stored geo-referenced image such as from a satellite. Registration requires matching and aligning the images using image feature discriminants. Once the platform camera image can be registered to the geo-referenced satellite image, the geo-located coordinates of the ground object can be determined from the reference image. The availability of high resolution satellite or aerial images is growing rapidly making the image geo-registration approach very viable. There are two main image registration techniques; correlation based and pattern or feature matching. These may be used as stand-alone algorithms or in concert with each other. The geo-registration process requires feature matching and/or correlation to optimally align the two images to one common frame for processing. This process effectively overlays and aligns the two images after selecting and filtering features to accentuate attributes common to both. These features act as geometric points of reference for the same scene taken at different times, from different viewpoints and by different sensors. Camera image variation will occur due to sensor noise, lighting, atmospheric variations, blurred backgrounds and target and clutter spatial diversity. The referenced image or map of a geographic area provides a known latitude and longitude for selected feature points that are matched to similar points on the unreferenced image after accurately aligning the images using basic transformation operations that include rotation, translation, shearing, and scaling. A cross-correlation match criterion is often used as a metric to correlate images and determine the best image match. In general there are many techniques used for the geo-registration process. The approach chosen depends on; required performance, processing requirements, and application specific features. One common element required for most geo-registration processing techniques is a camera model. Often a pinhole camera model is used; parameterized by what are termed intrinsic and extrinsic characteristics. Intrinsic include camera optics focal length, images offsets and scaling, while extrinsic are the DCM required to rotate an object vector to the image plane as discussed in section 3.0 with any offsets. The model is generally of the form [8]:

$$\hat{w} = \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \bar{M}_{ECEF}^{\text{Image}} \cdot \hat{P}_{TGTPLAT}^{ECEF}$$

$$\text{where: } \hat{P}_{TGTPLAT}^{ECEF} = \begin{bmatrix} \dfrac{\hat{P}_{TGT}^{ECEF} - \hat{P}_{PLAT}^{ECEF}}{range} \\ 1 \end{bmatrix} \;;\; \bar{M}_{ECEF}^{\text{Image}} = \bar{M}_{LOS}^{\text{Image}} \cdot \bar{M}_{ECEF}^{LOS}$$

$$\text{intrinsic: } \bar{M}_{LOS}^{\text{Image}} = \begin{bmatrix} f_u & s & c_u \\ 0 & f_v & c_v \\ 0 & 0 & 1 \end{bmatrix} \;;\; \text{extrinsic: } \bar{M}_{ECEF}^{LOS} = \left[D\bar{C}M_{ECEF}^{LOS}, \hat{t} \right] \;;\; \hat{t} = \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix}$$

The variables w,v are positions on the image focal plane. The intrinsic rotation matrix maps a vector in the LOS frame to the image plane. It is parameterized by the optics focal length (f_u, f_v), any image plane reference offset (c_u, c_v) and an x, y scaling factor (s) correction. The extrinsic matrix is just the DCM rotating a vector from ECEF coordinates to the LOS. It is appended with an offset to account for any coordinate frame offset relative to the image plane. Image geo-registration is simply an extension of the image-registration to include a geographically based reference image and utilization of platform inertial and Global Positioning System (GPS) measurements in the geo-locating calculations for geo-registration. Geo-pointing data is still required for an initial coarse geo-location estimate; but this approach may provide relief in the quality of the inertial data, especially the INS. However as performance depends on image quality, a high grade IMU may still be required to reduce blur and distortion due to LOS disturbances as described in Section 1.0. In addition, processing terrestrial images can be challenging to begin with when one considers terrestrial texture variations in high vegetation, snow, desert, jungle, plains, mountains, and water under an almost infinite variation of lighting conditions.

A growing geo-location application is geo-registration of a targeted area from a small remote platform, such as a UAV. The geo-registration approach for the small platform is driven by the platform SWaP constraints which in turn limits the use of navigation grade inertial sensors for direct geo-registration. These limitations also apply to cameras, however small cameras can pack a lot of performance and combined with a high performance processing algorithm can make geo-registration from a small platform such as a UAV viable. As an example, typical sensor configuration might mount a small video camera below the airframe on a small two axis stabilized gimbal that stabilizes the camera image relative to platform motion. The camera pointing vector is maintained nominally oriented downwards roughly perpendicularly to the terrain. This reduces image deformation due to perspective improving the ability to match the video images directly to the ortho-rectified reference images. The ground target is tracked using a video tracker and application tuned algorithm; maintaining the sensor LOS on the desired object providing its pixel coordinate in the image frame. The tracking algorithm SW would likely reside on-board in a track video processor to reduce tracker update latency while the video stream for geo-location is transmitted to a ground station to perform the image processing tasks for image registration and determination of the ground target geo-location.

In general, image registration consists of a set of generic processes that select the image features, match the features, use image rotation, translation, and scaling to align the image features, and finally map the unregistered image to the geo-registered reference. This is performed with levels of image resolution that change from low to high, progressively increasing to their full value. The algorithmic implementation will vary but have similar objectives as follows: (i) Project the reference and sensed video images to a common coordinate frame using existing geo-pointing data (GPS, INS). This projection initializes image-registration alignment conditions (ii) Image contour or feature processing to provide a feature structure that captures both geometric and intensity image characteristics to support matching and alignment of sensed video to reference images. (iii) Generate a precision spatial correspondence between the images using global correlation matching techniques that effects a precise alignment of the two images. A key to this final alignment is the camera model to improve correspondence of the two images via a progressive model scaling and orientation refinements. The basic process is shown in Figure 12.0 and a brief description of each step follows [9, 10, 11].

(i) Pre-Processing and Image Projection to a Common Frame: The camera video image, orientation data, and a model of the image sensor are required for initial coarse alignment of the reference and video images. The reference image is projected to the sensor coordinate frame and course aligned using the available pointing platform/gimbal data; GPS, and INS. This initializes conditions for applying more accurate image-based alignment algorithms. The reference imagery is normally a wide area, high-resolution ortho-image stored in the image processor.

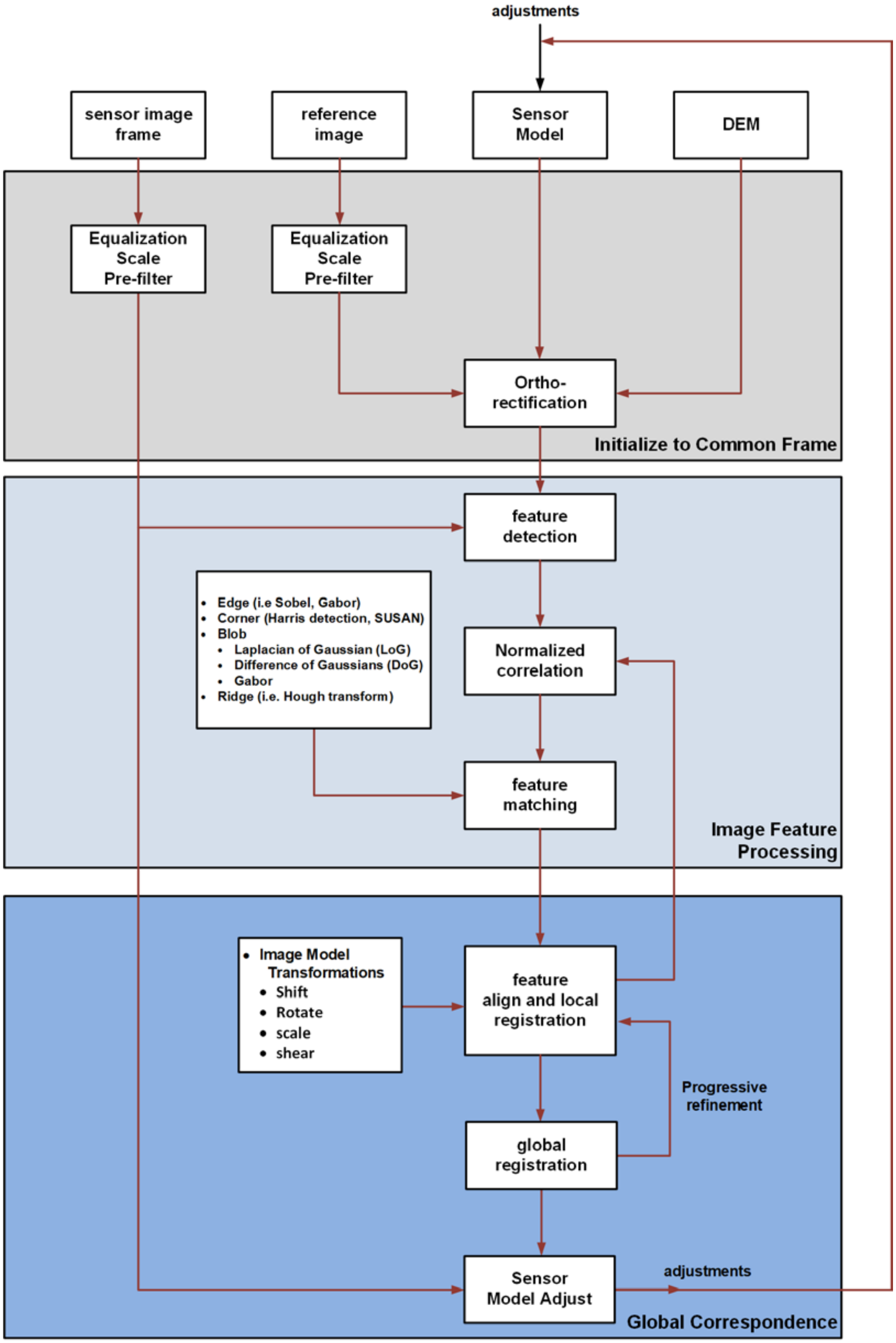


Figure 12 Typical coordinate frame geometry for geo-location

Each pixel in the reference image has a longitude, latitude and elevation co-registered with a Digital Elevation Map (DEM). The DEM is used to reduce distortion effects when the reference 3D image is converted to a 2D orthographic image projection or ortho-image. This process, referred to as ortho-rectification, is most often implemented using one of three algorithms; polynomial rectification, projective rectification, and differential rectification. This 2D registered image, projected into the sensed image video frame, is then used for subsequent feature matching. The texture structure of the reference ortho-image is used to identify coarse features with high spatial content that can be used for alignment on the sensed image plane, initialized using the geo-pointing data. This is followed by simple mapping orientation adjustments with the image sensor model that may require some pre-processing of the acquired sensed image. Often the image is converted to a gray scale or scaled consistent with the reference image and a method of image intensity equalization or normalization applied. The sensed image is also corrected for camera lens distortion and some form of filtering applied to reduce background clutter. A median filter is one approach to filtering small details visible in the sensed image but not the reference image while preserving edge contour acuity. This pre-processing results the reference ortho-image mapped to the camera model coordinate frame as provided by the platform geo-location data. The reference and sensed images are now prepared for further alignment and matching processing.

(ii) Image Processing [9, 10, 11]: The sensed image is preprocessed to enhance features that capture both the geometric alignment and intensity structure to support sensed image to reference matching. If not already applied, intensity equalization or normalization may be required to enhance feature selection by reducing contrast variation between images. Feature detection, matching and alignment are used to align frame-to-frame sensed video images, relating successive video frames to extend the spatial information beyond that of a single frame. This improves the probability of having sufficient structure to match the reference image. Often an affine transformation (linear mapping: rotation, translation, scaling, shear) provides a sufficient orientation adjustment for frame-to-frame alignment. Selecting a set of image intensity features to discriminate a common pattern structure and are invariant to the sensed and reference images is critical to alignment. Good features discriminate against background, exhibit high intensity to background contrast, and have a local dominant orientation or well defined point-like geometry. Features may vary significantly between rural, suburban, and city geographic structure scene and selection must adjust accordingly. Features used for matching will also be adjusted as processing progresses from a course to a fine alignment. For initial coarse alignment, the terrain imagery feature most significant might be corners, points, or small blobs with high spatial frequency content. Lines or edges, although providing feature structure may not work well since these features often repeat through an image. Once the image is coarse aligned, however, these could work well for finer alignment. In general, automatic feature detection requires spatial processing algorithms and there are many designed to detect different spatial properties. The Hough transform can be used to identify lines (edges, ridges). The Sobel and Gabor filters are often applied to detect edges. The LaPlacian of Gaussian (LoG) and Difference of Gaussian (DoG) are often applied for blob detection. The Gabor filter, termed a texture filter, can be tuned to act as a spatial band pass filter for regions with high spatial frequency content besides edges such as blobs. Some cases, such as rural imagery, may just not have a lot of high spatial content and networks of roads may be the most dominant feature. For this case edge detection and/or line transformations may provide the best feature extraction algorithm. For example, edge detection using the Sobel filter followed by post processing the filtered image through a Hough transform.

For evaluation, selected features need to be quantified so feature descriptors are identified to characterize a local image feature region which are captured in a vector containing this feature attribute information. Descriptors may include orientation and gradient magnitude for edge detection or the polarity and strength in blob detection. An image correlation matching algorithm is used to match features of the sensed and reference image. Small feature regions containing corresponding image features, generally have good statistical correlation. A normalized cross correlation match criterion is often used [10]. The image correlation matching algorithm will require some localized feature alignment using the set of affine transformations mentioned previously that include; translation, rotation, scaling, shearing and reflection; to find the orientation with the best matching results. Once features are matched and coarsely aligned, a mapping function transformation can be created for fine alignment.

(iii) Correspondence [9, 10, 11]: Correspondence is the final step in the geo-registration process. Detailed spatial correspondence between the sensed video image and projected reference imagery will result in precise geometric alignment. Integral to the alignment process is adjustment of the camera model image transformation matrices to better reflect the match of the video and reference as well as a resampling of the video image to match the pixel structure of the reference image. This often requires pixel interpolation to increase or decrease the number of image pixels, compensating for angular shifts or translations that limit accurate image correlation. A progressive refinement strategy is used to obtain accurate alignment using the feature points from the previous step for the feature-based registration process. Parameters of the mapping functions, estimated in the previous step, are used for initial fine alignment of the sensed image with the reference image. Early iterations are effectively improvements to the previous step, resulting in a frame by frame set of locally aligned sensed to reference images for processing during global matching. The region around each feature pixel of the video frame is correlated with a larger window from the reference image resulting in correlation regions with higher resolution. Cross correlation is performed to determine the feature alignment that maximizes the correlation function. Affine transformations continue to be applied, mapping and linking sets of sensed image points to reference image locations with high accuracy. The camera model is also updated; iteratively defining the optimum mapping function parameters for alignment. The optimal transformation for all correlations being that associated with the pixel coordinates with the highest peak on the correlation surface.

Once the reference Image and the sensed video image have been accurately aligned through feature-based registration, a direct correlation or pattern matching based hierarchical registration method (increasing resolution of correlated regions) is used to provide a final adjustment. Correlation based matching can be implemented efficiently in real time and this method positions the sensed image at every location within the reference image. A cross-correlation criteria determines the location providing the best match. Pattern matching is notionally similar to the feature matching descriptions and as discussed the matching criteria may include the correlation of image regions with many features. In general, there are several steps to the geo-registration process, as coarse estimates are refined to fine estimates, and both techniques may be applied at some point during the processing. Once image alignment is within the desired tolerance, the final algorithm maps the image to a physical location. The sensed image is registered to the reference image whose features are geo-registered to a known latitude and longitude; now mapped to the sensed image.

Due to the application diversity of the video imagery sensed, there is equivalent diversity in the image processing techniques. It is not a simple problem, video image diversity is near infinite when one considers cities, suburbia, rural, mountain, plains, water, desert, snow, jungle etc. all of which change with season and time. The process description provided intends to capture only a general flow.

Section 4.0 Key Points Summary

- ***Geo-location geo-registration can be performed via two methods i) direct geo-registration, ii) image geo-registration***
- ***Direct geo-registration is effectively a very accurate geo-pointing solution that can be applied to geo-reference data. It requires highly accurate inertial sensors and processing software to implement***
- ***Image geo-registration is a geo-referenced application of image registration requiring geo-referenced image data***
- ***Image processing architectures vary but generally follow a 3-step sequence: (i) projecting images into a common frame, (ii) image feature selection, matching and alignment and (iii) global image correspondence with a precision correlation algorithm.***
- ***Feature extraction , matching, alignment algorithms and correlation processing are techniques used in most image geo-registration architectures; algorithms vary***
- ***A camera model is a key element of most image processing architectures.***

5.0 Course Summary

Part 2 of the course reviewed details associated with the geo-pointing and location problem. Section 1 was effectively a review from Part 1.0 of the course, but critical to understanding the geo-pointing architecture. It covered LOS definition, methodology to characterize disturbances in terms of bias and jitter and performance metrics to evaluate the disturbance rejection requirements. Section 2 A was very brief discussion of key LOS control components; again discussed in Part 1.0 but these inertial sensing components are critical to the geo-location process so included for reference. Section 3 Described Geo-pointing; repeating some of the discussion from Part 1.0 but then delving into pointing errors unique to geo-pointing and brief overview of necessary alignment procedures. Section 4 described geo-location details; limitations of direct geo-registration and subsequent focus on image geo-registration processes.

References

1. Rue, A.K. "Stabilization of Precision Electro Optical Pointing and Tracking Systems", IEEE Transactions on Aerospace and Electronic Systems, AES-5 No. 5, 805-819, September 1969

2. Rue, A.K. "Precision Stabilization Systems", IEEE Transactions on Aerospace and Electronic Systems, AES-10 No. 1, 34-42, January 1974

3. Kennedy, P., Kennedy, R., Stabilizing the Line of Sight , ISBN: 978-0-9988539-7-0; 978-1-7320721-0-7, Laurin Publishing/Photonics Media Press, April 6 2018 (www.lineofsightstabilization.com)

4. https://en.wikipedia.org/wiki/Haversine_formula

5. https://en.wikipedia.org/wiki/Geographical_distance

6. Stuart Ness ,"Geo-registration of Remotely Sensed Imagery", Department of Computer Science and Engineering University of Minnesota, https://www.semanticscholar.org/paper/Georegistration-of-Remotely-Sensed-Imagery-Ness/800318a3618004a37820c041fedea42ad5208599

7. Mohamed M.R. Mostafa, Joseph Hutton "DIRECT POSITIONING AND ORIENTATION SYSTEMS HOW DO THEY WORK? WHAT IS THE ATTAINABLE ACCURACY? ",Proceedings, American Society of Photogrammetry and Remote Sensing Annual Meeting St. Louis, MO, USA, April 24-27, 2001 https://pdfs.semanticscholar.org/3415/6321a265ca863185edb5c01fedcb1c88be7b.pdf

8. Aleksandar Milosavljevic´ , Dejan Rancic', Aleksandar Dimitrijevic',

Bratislav Predic' and Vladan Mihajlovic', "A Method for Estimating Surveillance Video Geo-references" ISPRS Int. J. Geo-Inf. 2017, 6, 211; https://www.semanticscholar.org/paper/A-Method-for-Estimating-Surveillance-Video-Milosavljevic-Rancic/4b7ccf454bc3b86e6a839ca704aac2a6ee10ddc6

9. Richard P. Wildes, David J. Hirvonen, S.C. Hsu, R. Kumar, W.B. Lehman, B. Matel, Wenyi Zhao "Video geo-registration algorithm and quantitative evaluation". Published in Proceedings Eighth IEEE International Conference 2001 https://www.semanticscholar.org/paper/Video-georegistration%3A-algorithm-and-quantitative-Wildes-Hirvonen/683436559caea84bcbb41670e489beef6da6804d

10. Yaser Sheikh1, Sohaib Khan2 and Mubarak Shah1, "Feature-based Geo-registration of Aerial Images" 1 Computer Vision Lab, School of Computer Science, University of Central Florida, Orlando, FL 32816-2362, USA 2 Department of Computer Science and Computer Engineering, Lahore University of Management Sciences, Lahore, Pakistan https://www.semanticscholar.org/paper/Georegistration-of-Remotely-Sensed-Imagery-Ness/800318a3618004a37820c041fedea42ad5208599

11. Gianpaolo Conte, Maria Hempel, Piotr Rudol, David Lundström, Simone Duranti, Mariusz Wzorek, "High Accuracy Ground Target Geo-location Using Autonomous Micro Aerial Vehicle Platforms", Patrick Doherty, Dept. of Computer and Information Science, Linköping University, Sweden American Institute of Aeronautics and Astronautics https://pdfs.semanticscholar.org/6698/c08db821af246dded25ad146c1825e3700cb.pdf?_ga=2.249655744.731699222.1562267270-1260025231.1540647625

