satellite photogrammetry spring 2014 homework 2
due monday 3 feb.

using the results of homework 1 (hopefully with correct result), make the matlab functions described in the following pages. verify consistency of direct and inverse functions as described in step 5 on page 3/4 following.

Satellite Photogrammetry HW2

Build a suite of matlab functions + a "main" calling program as follows:

(0) main program reads .att and .eph files into a global variable which is then accessed by the FI2G function.

(1) function $\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = FI2G(l, s, h, \Delta q)$ This implements the projection done in HW 1. Add the Δq argument. (see later)

(2) function $\begin{bmatrix} \phi \\ \lambda \end{bmatrix} = FI2G\_PL(l, s, h, \Delta q)$ this is just a wrapper of FI2G(1) + a conversion from cartesian to geographic coordinates

(3) function $\begin{bmatrix} d\phi \\ d\lambda \end{bmatrix} = FI2G\_PL\_\phi(l, s, h, \phi, \lambda, \Delta q)$

This is just a wrapper of FI2G_PL (2) + a subtraction. The output vector is just a misclosure between input ϕ, λ and the ϕ, λ produced by l, s, h. In other words it is just:

$$\begin{bmatrix} d\phi \\ d\lambda \end{bmatrix} = \underset{\text{input}}{\begin{bmatrix} \phi \\ \lambda \end{bmatrix}} - \underset{\text{computed from } (l,s,h)}{FI2G\_PL(l,s,h,\Delta q)}$$

in other words l, s, h and ϕ, λ are not consistent necessarily.

(4) using the output of FI2G_PL_ϕ (3) : $\begin{bmatrix} F_\phi \\ F_\lambda \end{bmatrix} = \begin{bmatrix} d\phi \\ d\lambda \end{bmatrix}$

make function $\begin{bmatrix} l \\ s \end{bmatrix}$ = FG2I$(\phi, \lambda, h, \Delta q)$ as follows

start with $l^0, s^0 = (0,0)$ & input $\phi, \lambda, h, \Delta q$

ITERATE 3 TIMES (confirm Δs small)

(a) compute $\begin{bmatrix} F_\phi \\ F_\lambda \end{bmatrix}$ using FI2G_PL_ϕ *

(b) compute partials $\frac{\partial F_\phi}{\partial l}, \frac{\partial F_\phi}{\partial s}, \frac{\partial F_\lambda}{\partial l}, \frac{\partial F_\lambda}{\partial s}$ by:

all same do not have to compute again.

$$\begin{bmatrix} \frac{\partial F_\phi}{\partial l} \\ \frac{\partial F_\lambda}{\partial l} \end{bmatrix} = \frac{\text{FI2G\_PL\_}\phi(l^0+\Delta l, s^0, h, \phi, \lambda, \Delta q) - \text{FI2G\_PL\_}\phi(l^0, s^0, h, \phi, \lambda, \Delta q)^{*}}{\Delta l}$$

$$\begin{bmatrix} \frac{\partial F_\phi}{\partial s} \\ \frac{\partial F_\lambda}{\partial s} \end{bmatrix} = \frac{\text{FI2G\_PL\_}\phi(l^0, s^0+\Delta s, h, \phi, \lambda, \Delta q) - \text{FI2G\_PL\_}\phi(l^0, s^0, h, \phi, \lambda, \Delta q)^{*}}{\Delta s}$$

(c) solve for corrections to l^0, s^0 :

$$\begin{bmatrix} \frac{\partial F_\phi}{\partial l} & \frac{\partial F_\phi}{\partial s} \\ \frac{\partial F_\lambda}{\partial l} & \frac{\partial F_\lambda}{\partial s} \end{bmatrix} \begin{bmatrix} \Delta l \\ \Delta s \end{bmatrix} = \begin{bmatrix} -F_\phi \\ -F_\lambda \end{bmatrix}$$

$$J \quad \Delta = -F$$

$$\Delta = -J^{-1} F$$

(d) update l^0, s^0 :

$$\begin{pmatrix} l^0_{new} \\ s^0_{new} \end{pmatrix} = \begin{pmatrix} l^0_{old} + \Delta l \\ s^0_{old} + \Delta s \end{pmatrix}$$

after you are done iterating, the refined l, s are consistent with input ϕ, λ, h and you have <u>solved</u> the ground to image projection problem:

$$\begin{pmatrix} l \\ s \end{pmatrix} = FG2I(\phi, \lambda, h, \Delta q) \checkmark$$

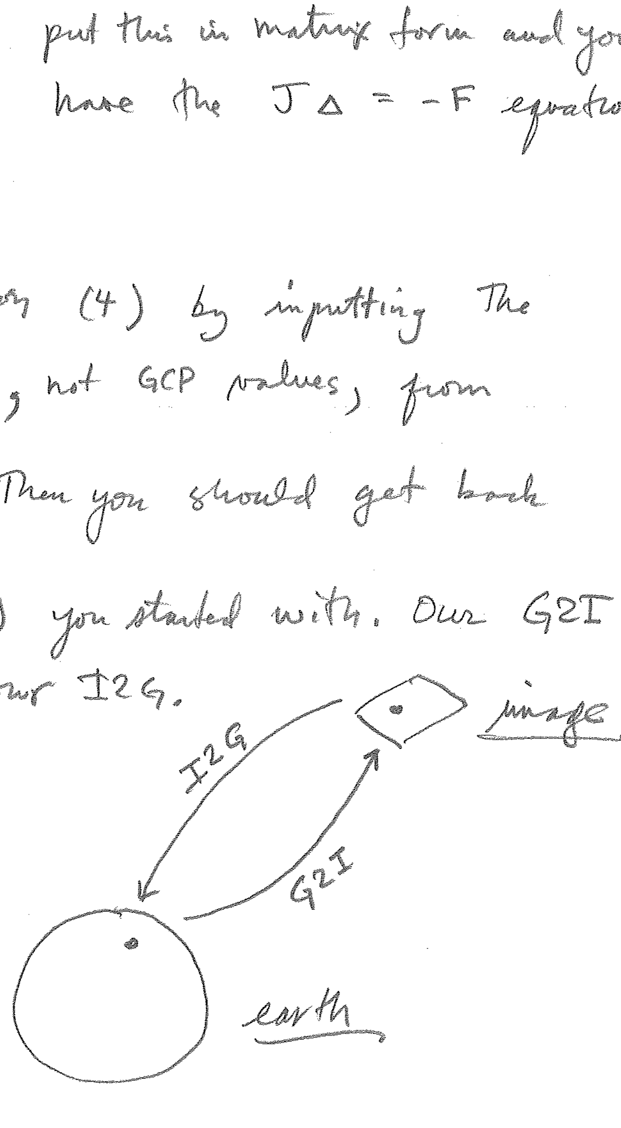
equations in (c) are from Taylor Series Linearization

$$F_\phi(l,s) = 0 \ , \quad F_\phi \approx F_\phi(l^\circ, s^\circ) + \frac{\partial F_\phi}{\partial l}\Delta l + \frac{\partial F_\phi}{\partial s}\Delta s = 0$$

$$F_\lambda(l,s) = 0 \ , \quad F_\lambda \approx \underbrace{F_\lambda(l^\circ, s^\circ) + \frac{\partial F_\lambda}{\partial l}\Delta l + \frac{\partial F_\lambda}{\partial s}\Delta s = 0}_{}$$

put this in matrix form and you have the $J\Delta = -F$ equation

(5) verify equation (4) by inputting the <u>computed</u> (ϕ, λ, h), not GCP values, from $\begin{pmatrix} l \\ s \end{pmatrix} = \begin{pmatrix} l_{meas} \\ s_{meas} \end{pmatrix}$. Then you should get back exactly the (l,s) you started with. Our G2I is just inverting our I2G.



Add a Δq argument to all functions. It is really only used in the FI2G function as follows:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = FI2G(l, s, h, \Delta q) \quad , \quad \Delta q = \begin{bmatrix} \Delta q_i \\ \Delta q_j \\ \Delta q_k \end{bmatrix}$$

- interpolate q_i, q_j, q_k, q_s based on line number and time.

- normalize q_i, q_j, q_k, q_s by computing and dividing by magnitude

- revise $q_i = q_i + \Delta q_i$
  $q_j = q_j + \Delta q_j$
  $q_k = q_k + \Delta q_k$

- normalize q_i, q_j, q_k, q_s by computing and dividing by magnitude

Note if you use $\Delta q = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$ then everything works just as before. If Δq is nonzero, then you are really modifying the physical math model.