Interview Question:

Sign Location Computation (Camera Calibration)

Introduction

Our research focuses on "***Smart Cities: Intelligent Transportation Infrastructure Health and Safety Condition Assessment and Management Using Sensing Technologies and Artificial Intelligence***". It is required for our research to automatically 1) detect roadway assets, including signs, lightpoles, guardrails, potholes, crackings, etc., 2) compute their location/position (coordinates), and 3) measure their geometry (e.g. length, height, and area). In these cases, you are required to compute the location (coordinates) of a roadway asset (e.g. sign) using a) basic camera calibration based on two consecutive images with their corresponding GPS coordinates, 2) SLAM/SFM.

This exercise is to prepare and evaluate your capability of camera calibration and location measurement. You are required 1) to review and develop mathematical formulations based on different hardware configurations, 2) to develop codes for implementing these mathematical formulations, 3) to validate the accuracy of your computed outcomes by comparing with the actual coordinates of signs, and 4) to discuss the reasons of discrepancy and to recommend methods for improving the accuracy of location computation. The following describes the problem formulations in the interview questions:

Three parts of questions you need to answer.

Part 1: Literature Review & Dataset

You will need to validate and refine the mathematical formulations. Please take a look at the mathematical formulations provided and the relevant references. Validate them and confirm yourself. Make necessary refinement if it is needed. You will need to prepare a formal report in Part 1. Be sure to provide REFERENCES from papers, reports and web sites you cited in your final report.

Case 1: Use 2D images and the correspond coordinates along with known camera intrinsic parameters to determine the physical location of a sign. The location computation is based on the case that 2D images with the corresponding GPS coordinates and the constant interval (e.g. 5 meters) between two consecutive image frames and the intrinsic camera parameters are also provided.

<<Anirban, Provide an illustration. The Figure A below shows the vehicle trajectory of the images taken and the location of a sign. The image is taken at a constant 5-meter interval. The image coordinates and the interval between

two images (5 meters) are given. What we want to determine is the location of a sign using the two consecutive images with their corresponding coordinates with the intrinsic camera parameters. Figure B also lists the intrinsic camera parameters.>>

Case 2*: images with the corresponding GPS coordinates and the distance between two consecutive image frames is NOT constant*. The distance between two points are calculated based on the distance between two GPS points (This is our Smart Phone case using Allgather application)

Part 2: Coding and Developing Applications

Based on the mathematical formulations you have finalized, you will need to develop *a tool/application that enable a user to click the same sign on two consecutive images. The interval of these two consecutive images is 5 meters (or set up a variable, called Interval with the default value of 5 meters).*

Application 1: *Users click at the same sign position from two consecutive images, we can then compute the x, y coordinate of a sign. The intrinsic camera parameters are given and the corresponding location of each image is also given.*

Application 2: *Users can click a point to get x, y coordinates, to measure distance and height, and to measure areas using* vanish point concept to form a mathematical formulation. You will need to establish a mathematical formulation and logic first. The users can then measure.

1) the length (depth) of a pavement marking, and guardrail distance,
2) the height of a sign post and a light pole,
3) an area, pothole area or patched area,
4) x, y coordinates of a point (by pointing to a location)

Part 3: Dataset and Validation, and Envision

Anirban, please prepare them for me. <<A test and validation data sets are provided below. The first test data set include: 1) a series of images taken using a GT sensing vehicles with a constant interval (5 meters) along with the corresponding GPS coordinates for each image, and 2) intrinsic camera parameters for the sensing vehicle. The second test data set include 1) a series of images taken using a smart phone (allgather) WITHOUT a constant interval along with the corresponding GPS coordinates for each image, and 2) intrinsic camera parameters for the Smart phone.>>

<<The validation dataset is the actual sign locations of 10 signs (collected using 3D lidar technology).>>

- How are your sign location computation outcomes (derived from two methods with two test data sets) compared to the ground truth (actual sign locations)? Please prepare a table listing them and also plot them on a map to discuss their discrepancies.
- Based on the discrepancies, please recommend the methods to further improve location computation accuracy.

[Part 4: Coding and Developing Applications

Case 3: Similar situation as Case 2. You need to use **SLAM and SFM** to conduct the location computation.]

Reports

For each part you will send us (james.tsai@ce.gatech.edu, a.chatterjee@gatech.edu and nicolas.six@gatech.edu) a report of around 2 or 3 pages in pdf format with your analysis and comments. You will also send us them in your code for parts 1 and 2 by giving us the link to your github / gitlab repository. Please include in the repository the dataset an easy instruction on how to set it up (like a bash script) as well as the trained weight of your neural networks (if they are too large you can use GT's Dropbox or any file sharing service you want).

We highly recommend you to follow the above instructions carefully.

Deadlines:

- Part 1: Sunday February 17, 2019 11:59 PM
- Part 2: Sunday March 17, 2019 11:59 PM
- Part 3: Sunday March 31, 2019 11:59 PM

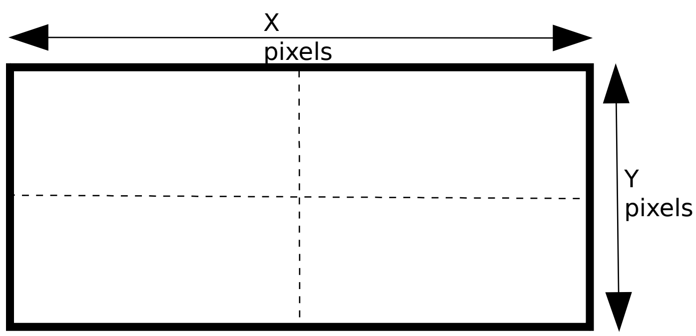
You are obviously welcome to start the latter parts before the deadline of the previous one.

Appendix A: Formulation of a sign location computation using

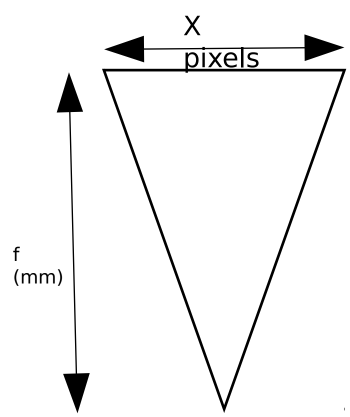
Estimation of traffic sign position using two consecutive images

Assuming tangential and barrel correction has been performed, we have a pinhole model:

Front View:



From top View:

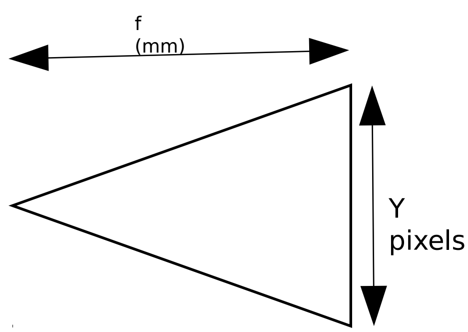


$$\frac{X}{2}\, pixels = f\,tan\left(\frac{\theta}{2}\right)\ mm$$

$$1\ \ pixel\ \ in\ \ x\ \ direction = \frac{2 f\,tan\left(\frac{\theta_x}{2}\right)}{X}\ \ mm$$

Distance from center in a projection f mm away.

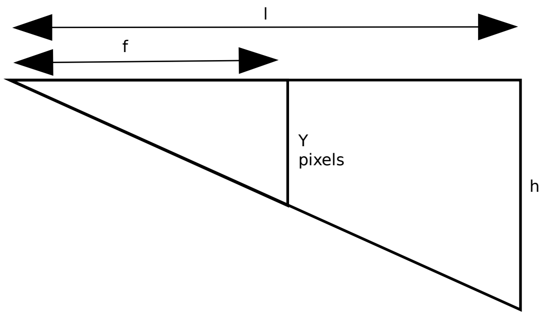
From side view:



$1\ pixel\ in\ y\ direction = \frac{2f\tan\left(\frac{\theta_y}{2}\right)}{Y}\ mm$ Distance from center in a projection f mm away.

Using 1 frame:

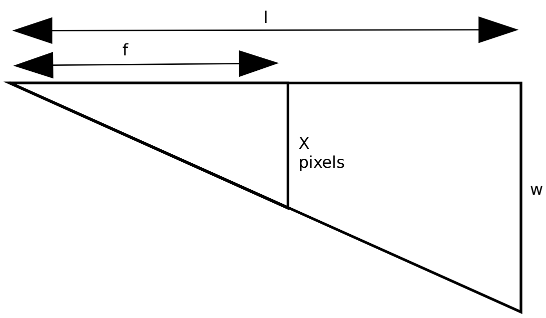
Side view:



$$\frac{y\left(\frac{2f\tan\left(\frac{\theta_y}{2}\right)}{Y}\right)}{f} = \frac{h}{l}$$

$$l = \frac{Yh}{2y\tan\left(\frac{\theta_y}{2}\right)}$$

Top view:

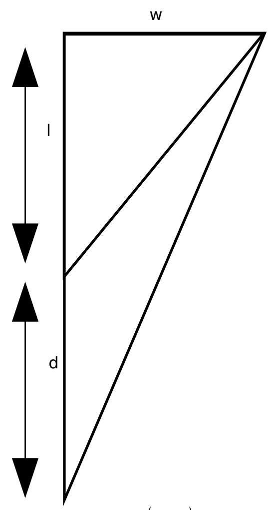


$$\frac{x\left(\frac{2f\tan\left(\frac{\theta_x}{2}\right)}{X}\right)}{f} = \frac{w}{l}$$

$$w = \frac{2xl\tan\left(\frac{\theta_x}{2}\right)}{X}$$

$$w=\frac{2x\ \ tan\left(\frac{\theta_x}{2}\right)}{X}\quad\frac{Y\,h}{2\,y\,tan\left(\frac{\theta_y}{2}\right)}=\frac{x\,Y\,tan\left(\frac{\theta_x}{2}\right)h}{y\,X\,tan\left(\frac{\theta_y}{2}\right)}$$

Using 2 frames:



$$tan(\theta_1)=\frac{x_1 2f\,tan\left(\frac{\theta_x}{2}\right)}{Xf}=\frac{2x_1 tan\left(\frac{\theta_x}{2}\right)}{X}$$

$$tan(\theta_2)=\frac{2x_2 tan\left(\frac{\theta_x}{2}\right)}{X}$$

$$using:\qquad l\,tan(\theta_2)=w=(l+d)\,tan\theta_1$$

$$we\ \ have\ :\ l=d\frac{x_1}{x_2-x_1}\qquad and\qquad w=l\,tan(\theta_2)=\ d\frac{x_1}{x_2-x_1}\ \frac{2x_2 tan\left(\frac{\theta_x}{2}\right)}{X}$$

Remarks (EL HAFIDI, Badr):

- some intrinsic parameters are missing (skew, camera center, aspect ratio)
- we should use epipolar geometry
- 3 frames should be specified: world frame (3D), camera frame(3D), image frame(2D)
- 2 transformations should be identified:
  - world frame (3D) to camera frame(3D)
  - camera frame(3D) to image frame(2D)
- the vehicle path was assumed to be a straight line perpendicular to the image frame which is usually not the case

- we should formulate the problem using matrices for the following reasons:
    - In computer vision, mathematical problems formulated with matrices are easy to understand, because we use matrices that have a specific meaning (matrix of intrinsic parameters, Project matrix, rotation matrix, translation matrix, essential matrix, fundamental matrix)
    - we can use all the linear algebra properties
- It easy to solve the problem when it is formulated using matrices, a good solution can be estimated using least square or Single Value Decomposition.