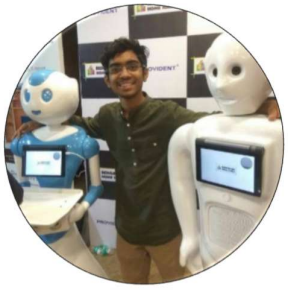
Submission
Preparation for Summer 2019



Sai Siddartha Maram
Former Intern Invento Robotics, India
Thapar Institute of Engineering and Technology

Abstract
Mapping of transportation infrastructure and its health is a critical area when we discuss the bringing up smart-cities. Monocular images are easy images which can be collected though pinhole cameras. Photometry deals with extracting geometrical information from these images. The report discusses methods, validation, applications and insights generated form the dataset.

Keywords: Birds Eye View, Aspect ratio, GPS, UTM N16, Camera Matrix, Distortion

Understanding the Dataset
Understanding the dataset is a critical element before beginning programming, the dataset provided has been obtained from data collected from GeorgiaTech Sensing Vehicle (GTSV). The GTSV consists of a camera which faces straight ahead and collects images after a particular interval. The sign annotations of each expressway, contains the coordinates through which you can extract the center of the sign located in the image. Converting the json file into a csv file which allows easier parsing of it from the programming perspective. We have the sign annotations as follow in csv format for various expressways.

frame_name	top_x	top_y	width	height	class
0018625.jpg	668	592	33	25	1

frame_name	top_x	top_y	width	height	class
0017765.jpg	864	605	22	56	1

Similarly, we have, csv files which indicate the real-world coordinates of cameras while taking the image. Other important file is the ground truth of the sign in the real world. This will never be used as input in the algorithm/application but will only be used for validation and evaluating our algorithm.

Understanding the images and camera
Images taken from cameras have certain extend of distortion associated it. This distortion can be identified and eliminated using the intrinsic parameters of the camera. The config.yaml, contains all the particular details and intrinsic parameters specific to the camera. Here we have two kinds of cameras which are in use, the camera associated with GTSV which takes images of the dimensions 2248 x 2048 pixels and then the AllGather application from the smartphone camera which takes images of the dimension 1140 x 1080 pixels. Through camera calibration we can develop two very critical matrices which will be used across multiple programming sections. The camera matrix and distortion matrix of the different cameras are as follows:

Camera Matrix of 2248 * 2048 :

2468.6668434782608	0	1228.876620888020
0	2468.6668434782608	1012.976060035710
0	0	1

Distortion Matrix of 2248 * 2048

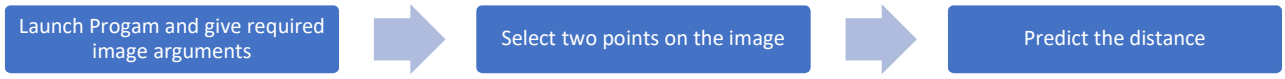
0.00125859	0	-0.00010658	0

In the same manner we have matrices for the smart phone application also.

Please do look into submission 1.0, where the approach in terms of calculation has been clearly elaborated

Task 1.0

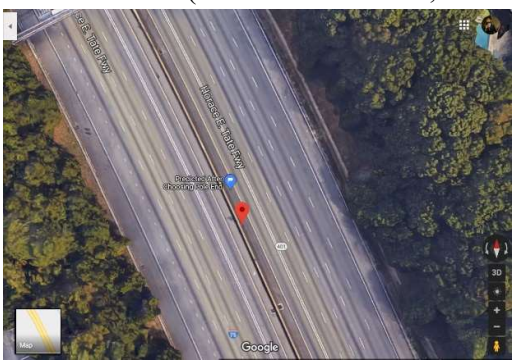
In this particular section we will be trying to develop a user interface where, two images are bought in front of the user, the user is allowed to select two points, one on both of the images. The application keeps track of these points and then based on the algorithm provides the corresponding UTM N16 coordinates, these UTM N16 coordinates can be easily converted into GPS coordinates which can be mapped as required.



Details and instruction to set up and run the program have been clearly explained in the video submission. The application takes in two input arguments which are images on a particular expressway. The rest is left to the program to track user cursor coordinates and on double click store image coordinates. Once two points are selected one from each image, the user can quit the application and note down the coordinates. The user is free to use the coordinates as required.

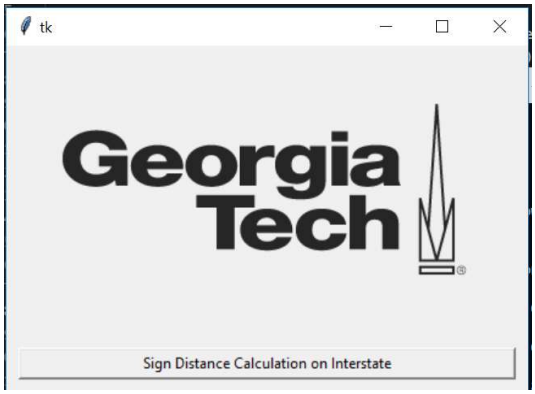
Note the user marked sign, with the mouse cursor, the user marked sign is indicated by a blue marker and its points coordinates are stored within the application. Once the user has finished selecting his points, the algorithm outputs the predicted location
Predicted: (736463.9917377126, 3750779.4410056357)

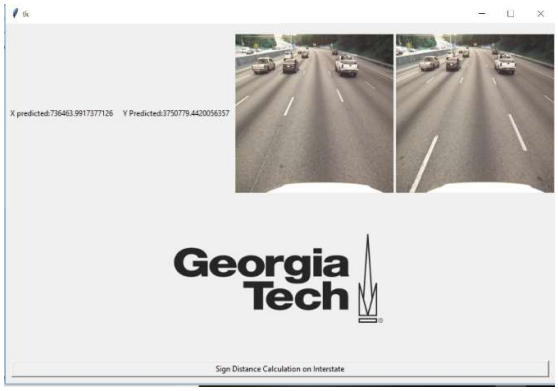
Ground Truth: (736465.737487371, 3750769.21175686)



The corresponding GPS coordinates can be found and marked on mapping applications. To the left marked in a blue pin is the predicted and the pin marked in red indicates the actual location of the sign.

To remove the burden of selection of signs from the user, we can achieve the location of the center of the sign using the given annotations. For the selection of images we can use the developed GUI. This GUI prompts the user to select two images, which have the sign.

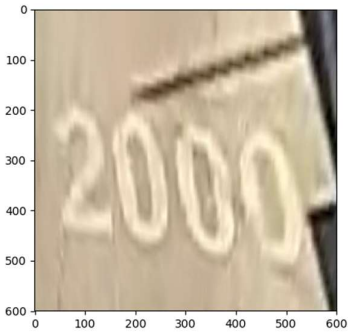


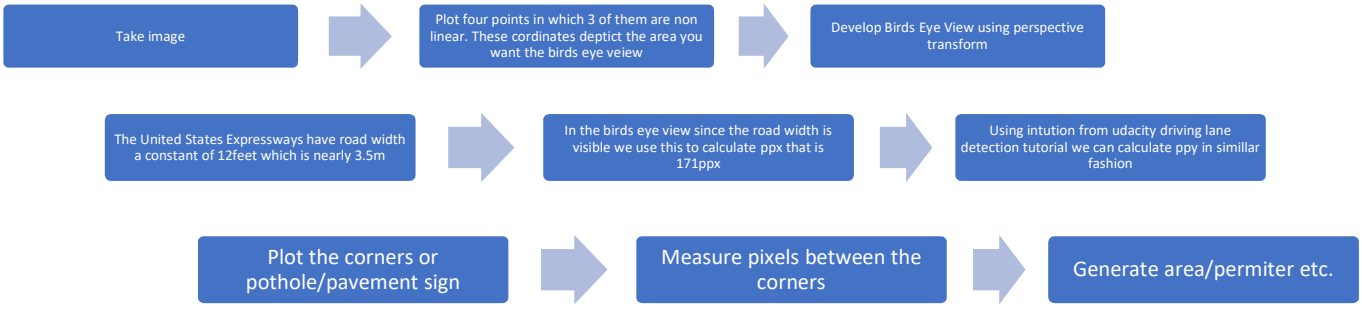


The algorithm then performs the concerned calculations and displays the result on the screen shown above.

Dimensions of Pavement markings and Potholes: (Choose the smartphone images since they have pavement signs)

Consider the following image, here we see the pavement marking 2000 written across the lane. Whenever we look to obtain the area of an object on planar surfaces on an image, our intuition says pick the four corners and then calculate the dimensions. But on repeated testing what has been found is this is only valid if and only if the camera is placed exactly vertical about the object. So we have to figure out a way to develop a birds eye perspective to generate the true dimensions of the pavement marking or pothole.





Here in this case we clearly see that the birds-eye view we have is a 600 x 600 image, now the lane width which is 12ft nearly 3.5meters corresponds to 600pixels width. From here we have

3.5 meters → 600pixels

1 meter → 171 pixels

After exploiting the camera matrix and the perspective transform matrix which we used to get the birds eye view, we have pixels per meter along the y direction as 58pixels.

Now marking the four coordinates of the pavement sign we get a rectangle which is 164pixels along the y_axis and 418pixels along the x_axis.

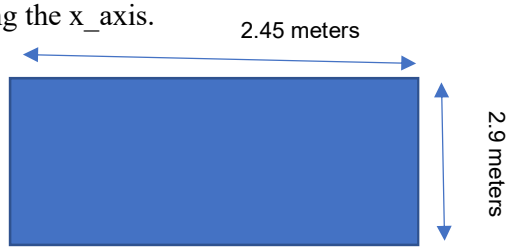
171 pixels → 1meter

418 pixels → 2.45m along the x-axis

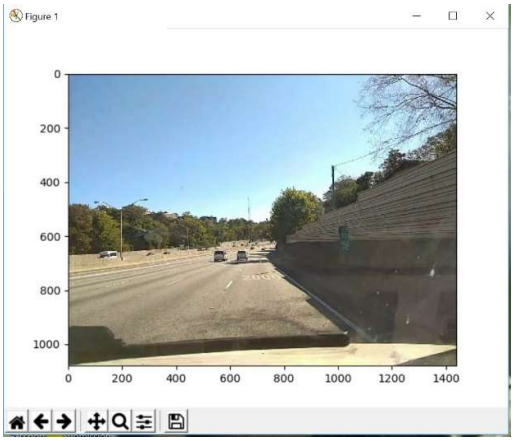
Similarly,

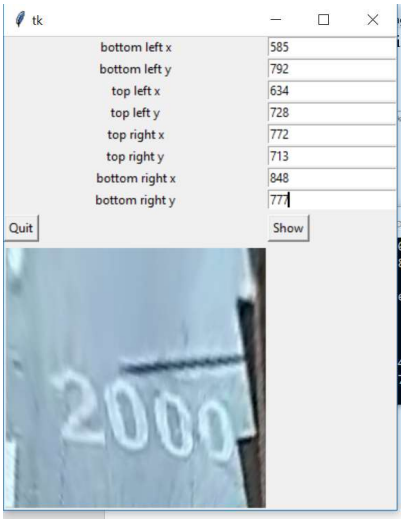
58 pixels → 1 meter

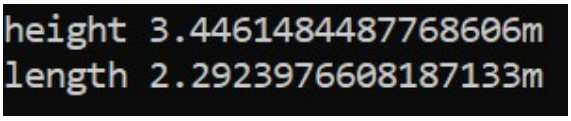
164pixels → 2.9 m

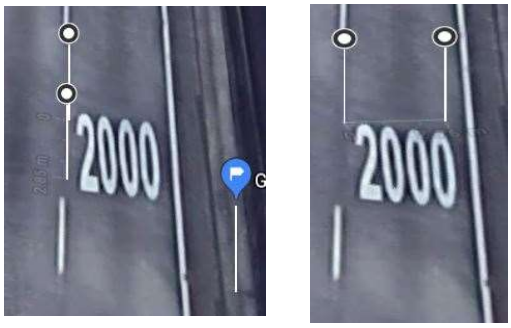


These are screenshots from google satellite view which gives an the width and height of the pavement sign, we see that the sign dimensions we have measured is similar.







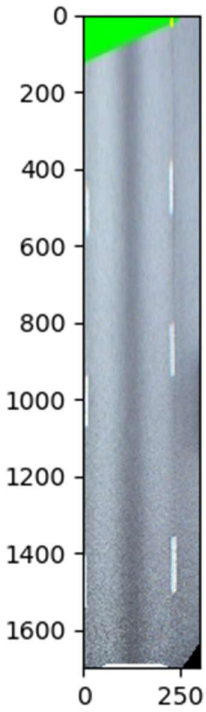


Error Analysis and Possible different methods for distance calculations on interstate expressways

Apart from the method mentioned in the appendix, one possible way of obtaining the distance I looked to explore was the Birdeye view perspective. There is no specific application as such developed for this but the algorithm has been developed.

Consider the image

Consider the image where we need to find how ahead the green sign on the left is. Now project this bottom of sign using our draw on image application onto the road. To get the true perspective of the distance we will need the top view. We generate this through perspective transform matrix.



Now we have generated the birds view of the road where we can get the true dimension of the road. The Birds view is 300pixels wide and contains one lane of the road which is 12 feet of 3.5meters. This indicates, we have 83pixels per meter. Using the perspective transform matrix we can calculate the pixels per meter along the y axis which is 55 pixels per meter along y-axis. Now the we have 1700 pixels from the car to the green mark, which leads to the conclusion 1700/55 meters which is nearly 31meters. Which is nearly 15 meters off the actual distance. This error raises because of two major issues.

GTSV i75 Error Log

Image 1	Image 2	Predicted X	Predicted Y	Actual X	Actual Y	class	Difference in X	Difference in Y m
0002875.jpg	0002876.jpg	736488.8028	3750779.906	736465.7375	3750769.212	2	23.06536182	10.69464182
0002876.jpg	0002877.jpg	736485.7616	3750782.744	736465.7375	3750769.212	2	20.02408894	13.53180086
0002877.jpg	0002878.jpg	736483.7692	3750781.695	736465.7375	3750769.212	2	18.03171565	12.48372681
0002878.jpg	0002879.jpg	736482.0172	3750779.945	736465.7375	3750769.212	2	16.27972664	10.7336885
0002879.jpg	0002880.jpg	736479.746	3750779.732	736465.7375	3750769.212	2	14.00854395	10.52012991
0002890.jpg	0002891.jpg	736455.5178	3750831.205			1		
0002891.jpg	0002892.jpg	736452.8499	3750831.948			1		

GTSV i285 Error Log

Image 1	Image 2	Predicted X	Predicted Y	Actual X	Actual Y	Difference in X m	Difference in Y m
0000750.jpg	0000751.jpg	732070.6627	3738153.675	732063.6162	3738027.003	7.046581531	126.6726392
0000751.jpg	0000752.jpg	732072.147	3738131.624	732063.6162	3738027.003	8.530848115	104.6218328
0000752.jpg	0000753.jpg	732071.6096	3738118.855	732063.6162	3738027.003	7.99348252	91.85208799
0000753.jpg	0000754.jpg	732069.4018	3738112.641	732063.6162	3738027.003	5.785629946	85.63839285
0000754.jpg	0000755.jpg	732070.2648	3738095.411	732063.6162	3738027.003	6.648683129	68.40877923
0000755.jpg	0000756.jpg	732069.1946	3738084.981	732063.6162	3738027.003	5.578400377	57.97811824
0000756.jpg	0000757.jpg	732068.6186	3738073.246	732063.6162	3738027.003	5.00247573	46.24310731

GTSV StateWay

Image 1	Image 2	Predicted X	Predicted Y	Actual X	Actual Y	Difference in X m	Difference in Y m
0000750.jpg	0000751.jpg	732070.6627	3738153.675	732063.6162	3738027.003	7.046581531	126.6726392
0000751.jpg	0000752.jpg	732072.147	3738131.624	732063.6162	3738027.003	8.530848115	104.6218328
0000752.jpg	0000753.jpg	732071.6096	3738118.855	732063.6162	3738027.003	7.99348252	91.85208799
0000753.jpg	0000754.jpg	732069.4018	3738112.641	732063.6162	3738027.003	5.785629946	85.63839285
0000754.jpg	0000755.jpg	732070.2648	3738095.411	732063.6162	3738027.003	6.648683129	68.40877923
0000755.jpg	0000756.jpg	732069.1946	3738084.981	732063.6162	3738027.003	5.578400377	57.97811824
0000756.jpg	0000757.jpg	732068.6186	3738073.246	732063.6162	3738027.003	5.00247573	46.24310731

SmartPhone 0

image_1	image_2	predicted_x	predicted_y	actual_x	actual_y
0017765.jpg	0017766.jpg	740633.8213	3743164.072	740662.332	3743127.976
0017766.jpg	0017767.jpg	740635.0039	3743160.361	740662.332	3743127.976
0017767.jpg	0017768.jpg	740635.5923	3743161.055	740662.332	3743127.976
0017768.jpg	0017769.jpg	740634.4771	3743173.608	740662.332	3743127.976
0017769.jpg	0017770.jpg	740637.7149	3743156.099	740662.332	3743127.976
0017770.jpg	0017771.jpg	740636.5326	3743168.21	740662.332	3743127.976
0017771.jpg	0017772.jpg	740639.2739	3743155	740662.332	3743127.976
0017772.jpg	0017773.jpg	740638.9614	3743160.9	740662.332	3743127.976
0017773.jpg	0017774.jpg	740640.4148	3743156.204	740662.332	3743127.976
0017774.jpg	0017775.jpg	740641.2867	3743154.933	740662.332	3743127.976
0017775.jpg	0017776.jpg	740642.1092	3743153.941	740662.332	3743127.976
0017776.jpg	0017777.jpg	740641.8367	3743158.559	740662.332	3743127.976

SmartPhone 1

image_1	image_2	Predicted X	Predicted Y	Actual X	Actual Y
0018625.jpg	0018626.jpg	741100.0089	3742737.73	741114.0434	3742737.581
0018626.jpg	0018627.jpg	741099.718	3742756.64	741114.0434	3742737.581
0018627.jpg	0018628.jpg	741099.5611	3742772.493	741114.0434	3742737.581
0018628.jpg	0018629.jpg	741100.9992	3742734.633	741114.0434	3742737.581
0018629.jpg	0018630.jpg	741100.6823	3742754.382	741114.0434	3742737.581
0018630.jpg	0018631.jpg	741100.5635	3742768.879	741114.0434	3742737.581
0018631.jpg	0018632.jpg	741101.2267	3742756.478	741114.0434	3742737.581
0018632.jpg	0018633.jpg	741101.1699	3742768.717	741114.0434	3742737.581
0018633.jpg	0018634.jpg	741102.1406	3742746.02	741114.0434	3742737.581
0018634.jpg	0018635.jpg	741101.5791	3742774.786	741114.0434	3742737.581
0018635.jpg	0018636.jpg	741102.7841	3742744.504	741114.0434	3742737.581