Applying Structure-from-Motion technique for visual odometry

V. Potó and Á. Barsi
Budapest University of Technology and Economics, Budapest, Hungary
poto.vivien@epito.bme.hu, barsi.arpad@epito.bme.hu

***Abstract* — Autonomous vehicles have several sensors, that enable sensing their environment. Positioning with GNSS systems has limits regarding the availability/accessibility and accuracy. The positioning accuracy of these vehicles can be increased using the environmental sensors. Karlsruhe Institute of Technology (KIT) and Toyota Technological Institute at Chicago have created a test data set, named KITTI containing color and grayscale camera image series, lidar measurements and GPS/INS data. We study the suitability of these image data for positioning purposes. In our paper we calculate the positioning by visual odometry.**

I. Introduction

As well as the world is developing so fast, people's needs grow also. There was a huge development in the last 200 years in the evolution of cars. The first car was designed by François Isaac de Rivaz in 1808. It worked with internal combustion engine and with hydrogen. In 1870 the first four-cycle, gasoline powered combustion engine came out, that was made by Siegfried Marcus. Nikolaus Otto invented the four-stroke petrol internal combustion engine and Rudolf Diesel made the four-stroke diesel engine. The beginning of battery electric car was bounded to Ányos Jedlik and Gaston Planté. [1]

In the last years a new era in the evolution of cars can be distinguished: the development of autonomous vehicles. According to Wikipedia, "An autonomous car is a vehicle that is capable of sensing its environment and navigating without human input." [2]

This new milestone brings new problems and questions that the world should solve. One of them is the localization and navigation. Nowadays the drivers use GNSS systems for positioning, but for example in urban environment with the tall buildings the satellites are invisible and measuring less than four satellites, the positioning fails. Furthermore, even if the number of satellites is enough, the car has problems with the accuracy. There are two choices: GNSS systems must be improved to have higher accuracy, or other additional methods have to be involved. One of them is the use of highly accurate and detailed map, which contains the road and its surroundings. Vehicles of nowadays are equipped with different sensors (cameras, lidars and radar-based sensors) to capture their environment constantly. Comparing the sensed data with the content of the map, the self-driving car can specify its location and plan its further path.

There are numerous methods to solve the localization problem. We have applied visual odometry computed on the bases of Structure-from-Motion (SfM) technique, so the position has been derived from the captured images of the onboard cameras. For this experiment we have used the test dataset from the KITTI Vision Benchmark Suite.

II. Data

KITTI is an exciting project in Karlsruhe, Germany, created and managed by the Karlsruhe Institute of Technology (KIT) and Toyota Technological Institute at Chicago. A Volkswagen Passat B6 was equipped with the following instruments:

- 1 Inertial Navigation System (GPS/IMU): OXTS RT 3003,
- 1 Laserscanner: Velodyne HDL-64E,
- 2 Grayscale cameras, 1.4 Megapixels: Point Grey Flea 2 (FL2-14S3M-C),
- 2 Color cameras, 1.4 Megapixels: Point Grey Flea 2 (FL2-14S3C-C),
- 4 Varifocal lenses, 4-8 mm: Edmund Optics NT59-917.

The system configuration is in Figure 1 and Figure 2.

There were dozens of urban, rural ways as well as highways captured by the probe vehicle in Karlsruhe and the surroundings. An eight core i7 computer with a RAID system, running Ubuntu Linux and real-time database was used to record the obtained data. All cameras were directed forward.

The freely available dataset can be used for several purposes: stereo image processing, optical flow, visual odometry, 3D object detection and 3D tracking. The researchers are supported by evaluation metric, so new, improved methods can be validated. [3]

We have used the raw data and the odometry dataset. The raw data has been classified into the following categories: city, residential, road, campus, person and calibration. The zipped datasets contain four data packet: unsynced + unrectified data, synced + rectified data, calibration files and tracklets. The difference between the unsynced + unrectified data and synced + rectified data is the level of processing. In the second data group all images are already undistorted and rectified, and synchronized with all sensor observations via time stamps. The first dataset contains the raw data, obtained directly from the instruments. Tracklets are elementary labelled objects (e.g. car, truck, tram, pedestrian) along a track.

We have chosen the synchronized and rectified dataset for our research work considering different environmental types. These datasets are presented with some features in Table 1.

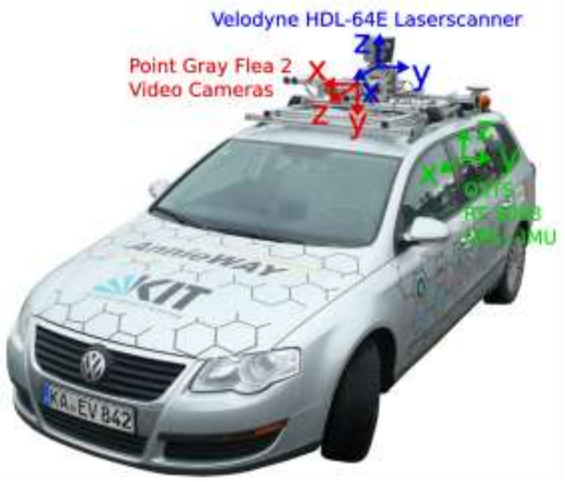
Velodyne HDL-64E Laserscanner
Point Gray Flea 2 Video Cameras
OXTS RT 3003 GPS / IMU
x
y
z

Figure 1. Fully equipped probe vehicle [3]

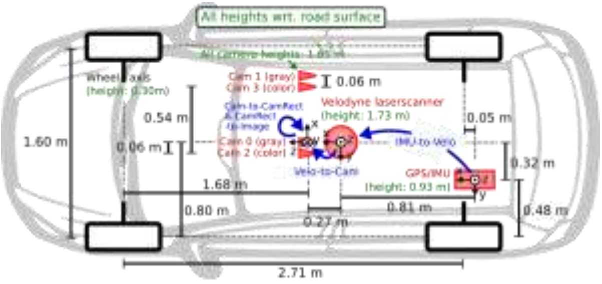

Figure 2. Fully equipped probe vehicle – top view [3]

Each dataset is labelled with a short dataset number. The full dataset identifier consists of the date of measuring, the word “drive” and the sequence number, e.g. 2011_09_26_drive_0020, which is the 20th dataset recorded on 26th September 2011. All zipped dataset stores the following file structure in 6 directories: image_00, image_01, image_02, image_03, oxts and velodyne_points. The first four folders are for the 1392 × 512 pixel sized camera images in png format and a text file with timestamps; the first two directories contains the grayscale images, then the color images, respectively. The oxts folder contains the GPS and IMU data in txt format and a description of the data format and timestamps. The Velodyne_points folder contains the lidar data points in binary format and 3 timestamp files. We have used solely the color images of the folder “image_02” and the oxts directory for validation.

A development kit with Matlab and C++ codes is also available to help people in the use of the data sets. [4, 5]

III. Method

The term odometry has the meaning of „route measurement” coming from composing two Greek words. In robotics, it is meant as the estimation technique of positioning of a wheeled robot relative to a starting location. It has been realized mostly by the measurement of wheel rotation (The so-called rotary encoders obtain the information e.g. in cars about the distance travelled). If the diameter or radius of the wheel is known, the distance can be calculated by multiplying the perimeter by the number of rotations. Beside this technique, odometry is continuously studied and visual odometry has also been invented. Visual odometry is per definition „the process of determining the position and orientation of a robot by analyzing the associated camera images” [6]

TABLE I.
Main parameters of the evaluated Datasets

Dataset number	Short dataset number	Category	Number of photos	Shape
2011_ 09_26_ drive_ 0020	20	residential	420	arched
2011_ 09_26_ drive_ 0032	32	road	390	hooked
2011_ 09_26_ drive_ 0039	39	residential	395	straight
2011_ 09_26_ drive_ 0070	70	road	1104	arched
2011_ 09_26_ drive_ 0093	93	city	433	broken
2011_ 09_26_ drive_ 0095	95	city	268	arched
2011_ 09_26_ drive_ 0104	104	city	312	straight
2011_ 09_26_ drive_ 0117	117	city	660	hooked
2011_ 09_28_ drive_ 0001	1	city	106	arched
2011_ 09_29_ drive_ 0004	4	road	339	straight
Odometry dataset	Odo	city	531	broken

Visual odometry requires therefore camera images being suitable for computing the position of the vehicle. Because cameras are important components of the future’s vehicle, big efforts have been taken to fix cameras on vehicles, to capture images and to evaluate them in order to support the vehicle control, mostly to detect obstacles, pedestrians and other vehicles on the road. The basic idea with visual odometry was the usage of these captured images also for deriving the position. Among the wide spectrum of possible solutions, our approach was focused on a mature technique applied in digital photogrammetry and image analysis [7].

The Structure-from-Motion technology is based on point features, which can be detected on the images and have exact identifiable position. There are interest operators, e.g. Förstner, Harris, Moravec, SIFT, SURF etc., which are standard tools in computer vision and extract points in images being in corners, intensity jumps, line ends. If the so determined points can be found in multiple images, these points give the possibility to couple the images in space (Figure 3). The more points are exactly identified and found in several images, the better merge can be achieved. The merging step is done pairwise and at the end a bundle adjustment can "fine tune" the whole system. After aligning all obtained images, the result is the relative orientation elements of the image projection centers. It means that the first image defines the coordinate system, in which the second image is coupled to the first one, then the third to the second one and so on. The coordinates of the image projection centers are showing exactly the movement of the camera carrier platform, i.e. the trajectory of the vehicle in a local reference system.

The described image alignment is an elementary part of an object reconstruction software package, like Agisoft Photoscan, Pix4D, VisualSFM. The basic idea of our paper was to test the ability of such software in the solution of the visual odometry problem related of vehicular camera images.

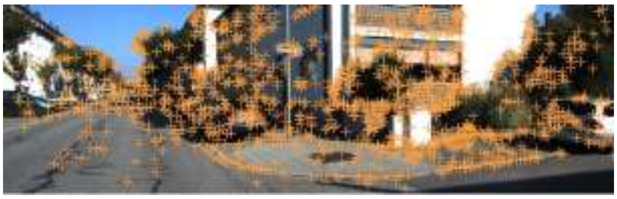

Figure 3. A Points obtained by interest operator being at least on 13 images

IV. Results

The positioning calculation of visual odometry has been conducted by Pix4D Mapper run in the cloud. The Amazon hosted virtual machine had two Intel(R) Xeon(R) CPU E5-2666 v3 @ 2.90GHz processors, 36 threads, 60 GB available RAM and Linux 3.13.0-91-generic x86_64 operating system (There was no need for GPU power).

The resulting image projection centers in green can be seen with some spatially determined tie points in Figure 4. All of the projection centers visualized forms a sequence, where the probe vehicle has moved. This sequence is therefore corresponding to the trajectory of the vehicle. In the test set the vehicle's navigation data is also available: the GPS and inertial measurements were fused and are also available to analyze the vehicle's movement. In our project this GPS-based data was applied for validating and

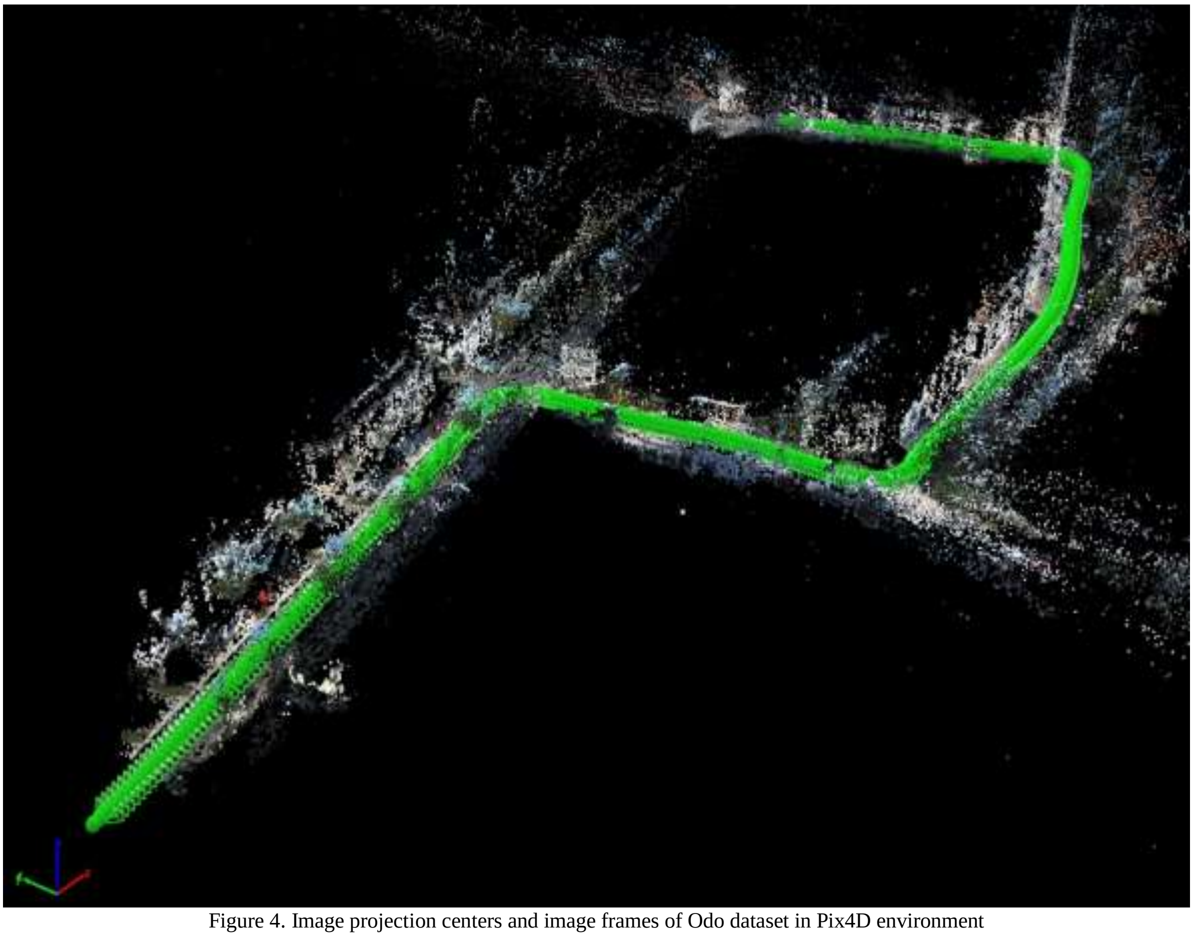

Figure 4. Image projection centers and image frames of Odo dataset in Pix4D environment

checking the quality of the obtained odometry measures. Mathworks Matlab was used for all further analysis steps.

Odometry produces the projection center coordinates in a local coordinate system. The center points have to be transformed into WGS84 geographic coordinate system, where the GPS/IMU positions (stored in oxts file) are given. Using common points of both coordinate systems the necessary transformation parameters can be achieved.

In Figure 5 we can see the oxts dataset (reference data) in green, and the transformed odometry data in blue. Figure axes represents the longitude and latitude coordinates in decimal degrees.

Some differences between the two data sequences can be noticed. There are breakings in the sequences, where the continuation of the sequences could not be achieved in odometry, i.e. there were some problems with merging the corresponding images. Of course, similar gap can be detected also in the reference measurements, meaning that the GPS signal receive had also troubles.

One can calculate an estimation for transforming the geographic coordinates into planar metric system. Then the two sequences can be compared; not only visually, but numerically, too. The point-by-point computation of differences between the odometry and reference data can be seen in Table II. The first column of the table shows the dataset short identifiers as numbers, in the next columns are some statistic data for the differences: the average

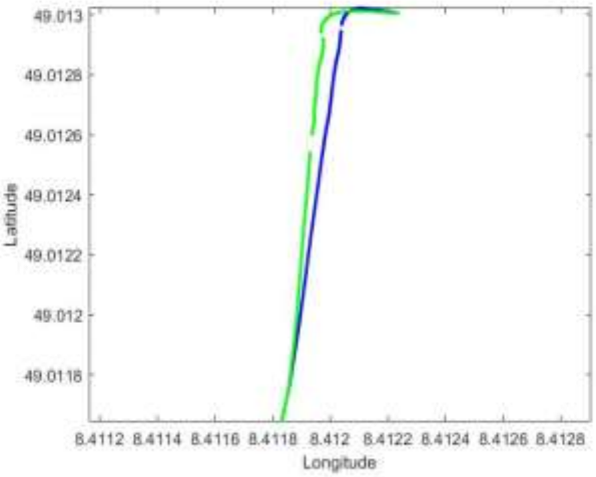

Figure 8. The calculated trajectory (Dataset 117). (Blue – result of the visual odometry, green – GPS/IMU reference

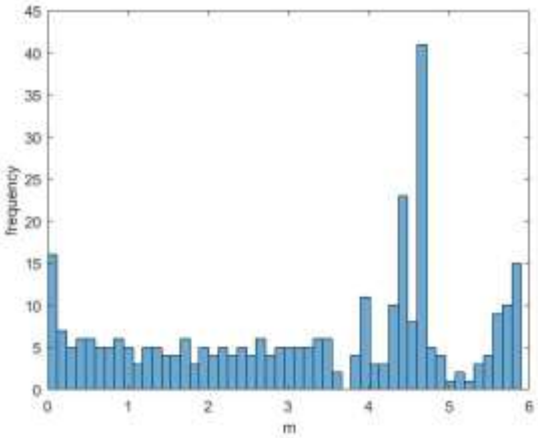

Figure 9. Histogram of differences for Dataset 117

Table II.
Statistics of results of the evaluated datasets in meter

Short number	Average difference	Maximal difference	Difference's median
1	2.6591	4.9502	2.7608
4	18.9735	24.782	20.4966
20	-	-	-
32	58.8058	93.2657	64.207
39	4.9704	17.4699	4.3955
70	2.6282	3.4853	3.1438
93	69.2613	196.2815	43.505
95	21.1573	49.1437	17.5812
104	1.4138	2.7006	1.3488
117	3.2015	5.8808	3.5387
Odo	0.6710	1.0749	0.6611

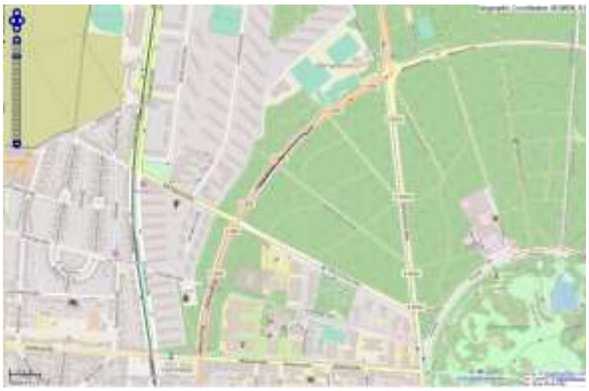

Figure 5. The result of Dataset 4 with black line in Matlab Webmap environment

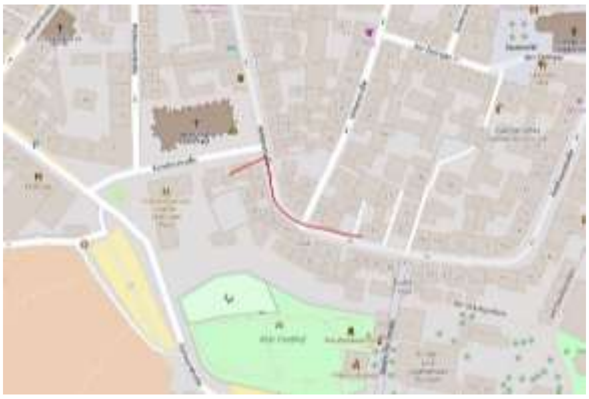

Figure 6. Dataset 93 in Matlab Webmap environment

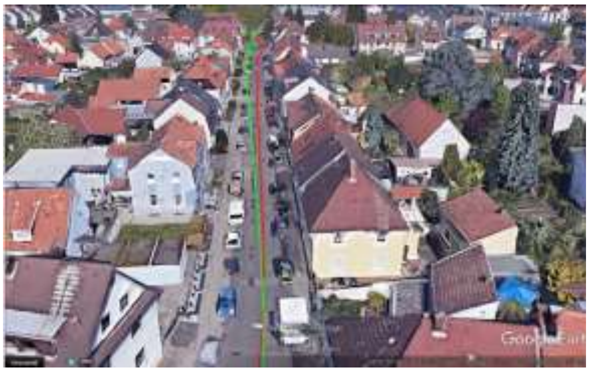

Figure 7. Dataset 39 in Google Earth environment

difference, the maximal difference and the difference's median. All given features are in meter. There was some unacceptably high difference, they are marked in red and are removed from the further analyses. [8]

For Dataset 20 the is no statistics, because there was a mistake in the dataset, so it has been dropped.

There are some extremities in the differences in the case of Dataset 93. With this dataset the GPS measurements were fully chaotic, some strange errors occurred. The odometry has resulted a smooth, believable trajectory, that can be seen in Figure 8, but still this dataset was also dropped.

With Dataset 32 – straight trajectory in urban environment – the obtained differences are mostly acceptable, but approaching the midpoint of the track, this difference reach almost 100 m. After the midpoint the computed trajectory goes back to the normal state (arched form differences). Supposedly in this case there were some point identification problem and systematic error in merging.

Considering Dataset 95 there is a similar effect in image coordinates. This dataset's shape is also arched. Figure 6 presents a histogram of differences for Dataset 117. On the x axis there are the differences in meter, on the y axis, the frequencies, respectively. The distribution of the differences is almost equal, only two spikes are visible between 4 and 5 m. This figure illustrates that the SfM based visual odometry can result stabile solutions.

We have used also another methods for displaying and verifying. The transformed odometry datasets can be shown with OpenStreetMap background, so the interpretation of the trajectories is easier. An example can be found on Figure 7 for Dataset 4 with black line on the beltway. Some noise can be detected at the upper end of the black line.

Another presentation way is with Google Earth environment. In Figure 9 there is the Dataset 39 in green with the verifying oxts data in red.

V. Summary

The 3-dimensional object reconstruction has been solved by the Structure-from-Motion technique. This algorithm has a processing step, when the projection centers of the taken images are calculated. The result is a coordinate tuple in a local reference system.

Because the development of the future autonomous vehicles is strongly based on camera images, numerous equipment is developed to collect imagery. The combination of the image processing of the automotive (onboard) cameras and the photogrammetric object reconstruction by the Structure-from-Motion technique was the basic idea of our research. The results have shown that the projection center calculation is a possible way for the visual odometry solution.

The test can point on that the captured images of photogrammetrically lower geometric resolution are suitable to execute this task. The image set collected during a test drive has many images, but the content change is quite low between consecutive images, so the similarity enables their coupling and the relative position could be derived.

In the test suite several environmental types (residential, city and road) were chosen. The applied technique can be evaluated in this aspect, too. The prior expectation was that city has the highest variability, has the highest amount of image patterns, the most features can be detected there and so the best solutions will be there. Our most interesting trajectory is also in a city (Odo). This hypothesis was not checked directly but the city test data were mostly successfully evaluated. Of course, if city or residential roads are used and the cameras captures trees or similar objects along the trip, enough patterns and features are present to be able to solve the odometry challenge.

Furthermore, if the odometry is solved, the spatial positions of the cameras are known, then using two or more corresponding synchronized cameras opens the way to apply stereo object positioning and reconstruction. This achievement is essential in environmental sensing and the vehicle control, when other vehicles, pedestrians or any objects on the road (like dropped cargo) must be detected and accidents can be avoided.

References

[1] Wikipedia – History of the automobile https://en.wikipedia.org/wiki/History_of_the_automobile

[2] Wikipedia – Autonomous car https://en.wikipedia.org/wiki/Autonomous_car

[3] KITTI webpage http://www.cvlibs.net/datasets/kitti/index.php

[4] Geiger, A., Lenz, P., Stiller, C., & Urtasun, R. (n.d.). "Vision meets Robotics: The KITTI Dataset" http://www.cvlibs.net/publications/Geiger2013IJRR.pdf

[5] Geiger, A., Lenz, P., & Urtasun, R. (n.d.). "Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite." http://www.cvlibs.net/publications/Geiger2012CVPR.pdf

[6] Wikipedia – Visual odometry https://en.wikipedia.org/wiki/Visual_odometry

[7] Somogyi Á - Barsi Á Pixel-based 3D Object Reconstruction, In: Orosz Gábor Tamás (Ed.) 11th International Symposium on Applied Informatics and Related Areas (AIS 2016), Székesfehérvár, 2016.11.17, pp. 60-63

[8] G. Mélykúti, "Topography – Basic terms in mapping" in Hungarian 2010 http://www.tankonyvtar.hu/hu/tartalom/tamop425/0027_TOP1/ch01s04.html