

Monocular Height Estimation Method with 3 Degree-Of-Freedom Compensation of Road Unevennesses

Alex Masuo Kaneko^{*}, Kenjiro Yamamoto

Hitachi, Ltd. Research & Development Group. Center for Technology Innovation - Mechanical Engineering. Robotics Research Department. Japan.

ARTICLE INFO

Article history:

Received: 08 June, 2017

Accepted: 26 July 2017

Online: 10 August, 2017

Keywords:

height estimation

autonomous robots

monocular camera

3 DOF compensation

ABSTRACT

Height estimation of objects is a valuable information for locomotion of autonomous robots and vehicles. Even though several sensors such as stereo cameras have been applied in these systems, cost and processing time have been motivating solutions with monocular cameras. This research proposes two new methods: i) height estimation of objects using only a monocular camera based on flat surface constraints and ii) 3 degree-of-freedom compensation of errors caused by roll, pitch and yaw variations of the camera when applying the Flat Surface Model. Experiments outdoors with the KITTI benchmark data (4997 frames and 436 objects) resulted in improved accuracy of the estimated heights from a maximum error of 1.51 m to 1.12 m and reduced number of estimation failures by 4 times, proving the validity and effectiveness of the proposed method.

1 Introduction

This paper is an extension of work originally presented in the 2016 IEEE/SICE International Symposium on System Integration [1]. This previous work proposed a monocular height estimation method by chronological correction of road unevenness, which basic experiments in laboratory and asphalt correcting camera pitch variations proved the validity of the method. However, since only pitch variations were considered and experiments were conducted on asphalt in one environment, several items remained as future work. Therefore, we extended our work by the following:

- Analysis and 3 degree-of-freedom (3DOF) compensation of errors caused by roll, pitch and yaw variations.
- Extended experiments on asphalt with several conditions of road and objects, permitting further analysis of external disturbances.

Cameras have been applied in many fields such as robotics and autonomous driving for localization and object recognition [2]-[7]. Even though stereo cameras can provide the depth to obstacles, their higher

cost and required processing have motivated several studies to estimate depth or height with monocular cameras. Height estimation permits the robot to detect and avoid potential obstacles on the road, becoming a valuable information of the surrounding environment. Studies [8] and [9] estimate height of objects with a steady camera. While the first relies on a previous calculation of the vanishing point, the latter relies on a known object height in the scene. On the other hand, other studies focus on height estimation using a moving camera. Study [10] estimates height by computing the focus of expansion in the scene and segmenting ground and plane by sinusoidal model fitting in reciprocal-polar space. The method proposed in [11] estimates height of objects on the road by obstacle segmentation and known camera displacement from odometric measurements, refining the measurements with several frames. Although there are many promising height estimation methods, they still strongly rely on extra extraction of information from the scene or external sensors. Moreover, the presented methods assume that the ground is flat and no discussions or analyses of eventual pose variations of the camera were mentioned. In this context, we propose in this work a height estimation method that requires no previous information of the scene, nor information from

^{*}Corresponding author: Alex Masuo Kaneko. Hitachi, Ltd. Research & Development Group. Center for Technology Innovation - Mechanical Engineering. Robotics Research Department. 832-2, Horiguchi, Hitachinaka, Ibaraki, Japan.
E-mail: alexmasuo.kaneko.bs@hitachi.com

external sensors. Furthermore, even though we assume flat surface, we analyze and compensate the effects of roll, pitch and yaw variations of the camera.

The rest of this paper is organized as follows. Section 2 introduces the existing issues of monocular cameras due to depth ambiguity and roll, pitch and yaw variations. Section 3 explains the proposed monocular height estimation method and compensations of roll, pitch and yaw variations. The conducted experiments are detailed in section 4. The obtained results are discussed in section 5. Lastly, the conclusions and future work are provided in section 6.

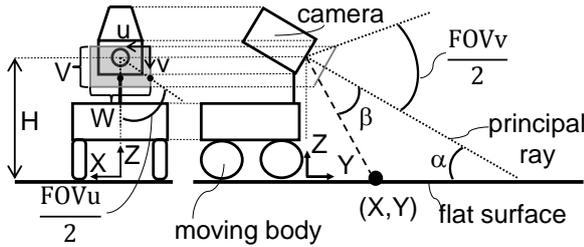


Figure 1: Conversion between pixels and meters using the Flat Surface Model.

2 Issues of Monocular Cameras

In this section, two main issues of monocular cameras are introduced: A) depth ambiguity and B) effects of roll, pitch and yaw variations.

2.1 Depth Ambiguity

In order to explain this limitation, we first briefly introduce the Flat Surface Model (FSM), a technique that permits easy relation of pixels (in camera coordinates) and meters (in real world coordinates), and is commonly applied with monocular cameras [12]-[14]. Figure 1 shows a moving body, a camera and flat surface. The camera is attached to the moving body at a known height H and angle α in relation to the flat surface. The line connecting the camera center and the flat surface with angle α is called principal ray. The coordinate system of the camera is defined by pixels (u, v) , with a known and fixed vertical length V and a horizontal length W . The maximum angle seen by the camera (field of view) in both u and v directions are fixed and defined as angles $FOVu$ and $FOVv$. The coordinate system defined by (X, Y, Z) is fixed on the moving body. If we assume that the surface is flat, the relation between a pixel (u, v) and its real position (X, Y, Z) in meters is given by (1) to (3), where β is the angle in relation to the principal ray.

$$\beta = \text{atan}\left(\frac{(2v - V)}{V} \tan\left(\frac{FOVv}{2}\right)\right) \quad (1)$$

$$Y = \frac{H}{\tan(\alpha + \beta)} \quad (2)$$

$$X = Y \frac{\cos(\beta)}{\cos(\alpha + \beta)} \frac{(2u - W)}{W} \tan\left(\frac{FOVu}{2}\right) \quad (3)$$

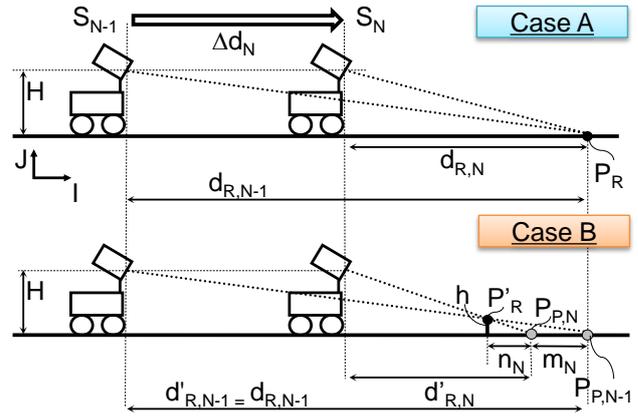


Figure 2: Example of application (A) and limitation (B) when applying the Flat Surface Model.

One common application of the FSM is to estimate the camera displacement by Visual Odometry (VO) [12]-[14], which is briefly explained in “Case A” of Figure 2. First, consider the moving body and camera described in Figure 1 running on a flat surface. In an initial position S_{N-1} , the camera takes a frame and shoots a point P_R on the ground, computing $Y_{N-1} = d_{R,N-1}$. Next, consider that the moving body moves Δd_N in direction I of a coordinate system (I, J) fixed on the ground, reaching position S_N . In this new position, it takes another frame, tracks point P_R and $Y_N = d_{R,N}$ is obtained by the FSM. Using the computed information Y_{N-1} and Y_N from the two positions S_{N-1} and S_N , the real camera displacement ΔCam_N can be correctly estimated by (4). By repeating the previous steps, the displacement of the moving body can be estimated on the following positions. Notice that we showed a simple case with only one point P_R and direction I , but many points and directions can be considered in VO.

$$\Delta Cam_N = d_{R,N-1} - d_{R,N} = \Delta d_N \quad (4)$$

The depth ambiguity can be visualized in “Case B” of Figure 2, which contains an object of height h on the ground. Let’s assume that in position S_{N-1} the camera shoots a point P'_R on the top of the object, and the projection on the flat surface is point $P_{P,N-1}$, exactly on the same location as point P_R in “Case A”. Although points P_R and P'_R belong to different (X, Y, Z) in the world, the FSM can’t distinguish this ambiguity and computes $Y_{N-1} = d'_{R,N-1} = d_{R,N-1}$. However, when the camera moves Δd_N in direction I , it tracks point P'_R and consequently, computes $Y_N = d'_{R,N}$ at position S_N . Finally, the displacement $\Delta Cam'_N$ seen by the camera becomes as (5), what shows that the real displacement Δd_N is not correctly estimated due to the presence of the object.

$$\Delta Cam'_N = d'_{R,N-1} - d'_{R,N} \neq \Delta d_N \quad (5)$$

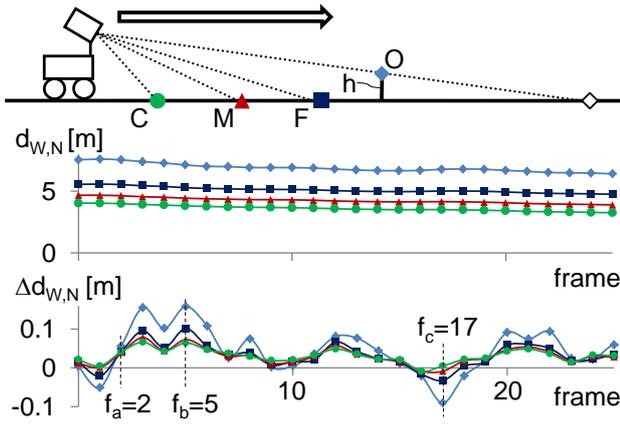


Figure 3: Influences of pose variation on the Flat Surface Model.

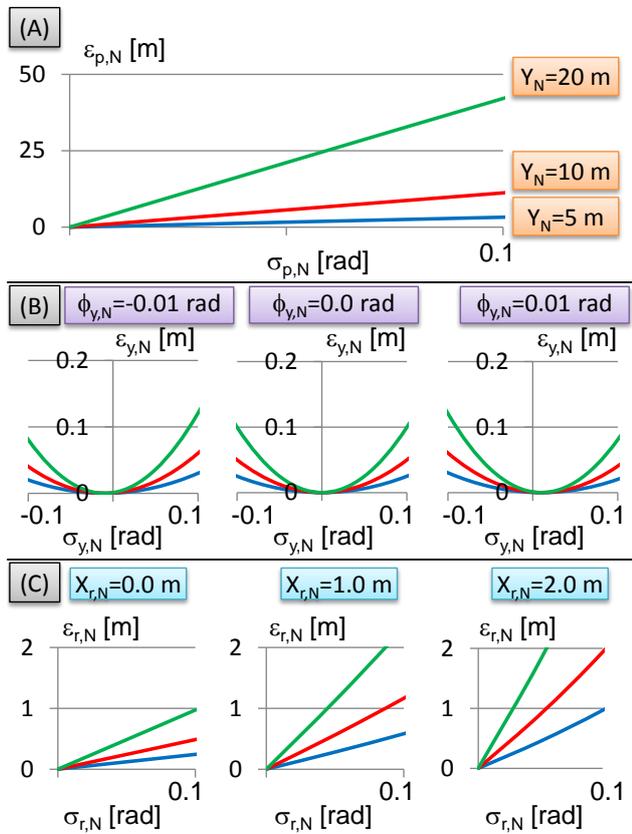


Figure 4: Expected errors of the calculated distances by the FSM in function of pitch (A), yaw (B) and roll (C).

2.2 Effects of Roll, Pitch and Yaw Variations

Even though the FSM assumes that the surface is perfectly flat, in fact small unevennesses exist. Such influences are detailed in Figure 3, with a simple analysis done on a real asphalt. A camera was fixed on a moving body and moved tracking four points: i) on a static object with height h (point O), ii) on the ground far from the camera (point F), iii) on the ground close to the camera (point C) and iv) on the ground with a distance between F and C (point M). In each frame, the distances $Y_N = d_{W,N}$ obtained by the FSM were

computed. In order to better visualize the influences of unevennesses, the corresponding displacements $\Delta d_{W,N} = d_{W,N} - d_{W,N-1}$ are also displayed in the figure. If the surface of the asphalt was really flat, then the displacements $\Delta d_{W,N}$ of each point C , M and F were expected to be the same in each frame. However, while this happened in frame $f_a = 2$ for example (suggesting that the pose of the camera was exactly the one expected by the FSM), in frames $f_b = 5$ and $f_c = 17$ the displacements were different (suggesting that the pose of the camera was different from the one expected by the FSM). Moreover, Figure 3 shows another important pattern, which points closer to the camera have smaller magnitudes of $\Delta d_{W,N}$: the magnitude of $\Delta d_{W,N}$ calculated with C in a determined frame is smaller than the one calculated with M , which is smaller than the one calculated with F , which is smaller than the one calculated with O in the same frame. This example clearly shows the effect of unevennesses on the FSM. A further analysis according to roll, pitch and yaw variations is presented in Table 1 and Figure 4.

2.2.1 Pitch

Figure (a) of Table 1 shows the influence of pitch variations on the FSM. Consider that in a certain frame N a moving body that is shooting a point O_R on a static object on the surface has its pitch changed (represented by $\sigma_{p,N}$) by an unevenness. This variation shifts height H to $H_{p,N}$ and therefore the camera computes the distances by the FSM in relation to a new wrong flat surface, $FS_{p,N}$. In this wrong surface, the projection of point O_R becomes $P_{p,N}$ and consequently, $Y_N = d_{p,N}$ is calculated. However, the correct distance in such configuration is the one computed with G_R , the projection of O_R on the real surface on the ground, resulting in $Y_N = d_{R,N}$. The relation between the wrong ($d_{p,N}$) and correct ($d_{R,N}$) distances is shown by (6), (7) and (8), where l_p is the axis of rotation and $\gamma_{p,N}$ is the angle between the camera height H and l_p . It is important to notice that the presented equations don't explicitly contain the object height h . It happens because the computed $Y_N = d_{p,N}$ itself contains this information, since it is function of O_R , $P_{p,N}$ and h . We define the error caused by pitch variation $\sigma_{p,N}$ as $\varepsilon_{p,N}$, which is the difference between the wrong estimated distance ($d_{p,N}$) and correct one ($d_{R,N}$), according to (9). Figure 4 (A) quantifies this error $\varepsilon_{p,N}$ by adopting as example $H = 1.65$ m, $l_p = 1.03$ m, several camera pitch variations ($\sigma_{p,N}$) and apparent distances to a point by the FSM ($Y_N = d_{p,N}$). The adopted values of the constant parameters H and l_p are the same as the ones used in the experiments section for easier analysis of the obtained results. We can observe that $\varepsilon_{p,N}$ increases with the increase of $\sigma_{p,N}$ and Y_N . Even for a pitch variation of $\sigma_{p,N} = 0.1$ rad, the error $\varepsilon_{p,N}$ can reach nearly 50 m when $Y_N = 20$ m, what shows that $\varepsilon_{p,N}$ is very sensitive even to small $\sigma_{p,N}$.

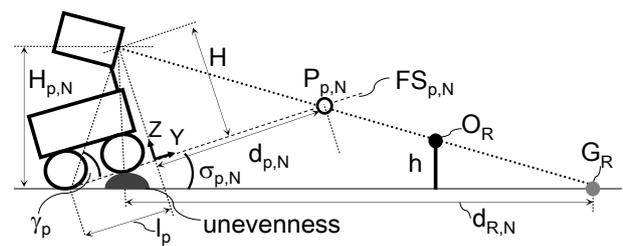
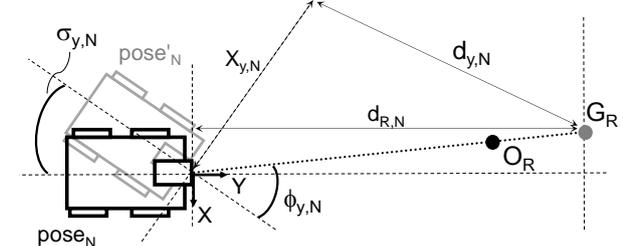
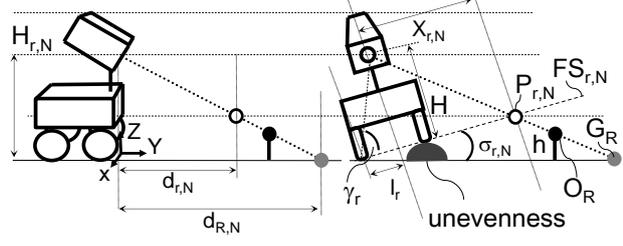
Variation	Compensation
 <p>(a) pitch</p>	$\gamma_p = \arctan\left(\frac{H}{l_p}\right) \quad (6)$ $H_{p,N} = \sqrt{l_p^2 + H^2} \sin(\sigma_{p,N} + \gamma_p) \quad (7)$ $d_{R,N} = \frac{H_{p,N}[(d_{p,N} + l_p)\cos(\sigma_{p,N}) - \sqrt{l_p^2 + H^2}\cos(\sigma_{p,N} + \gamma_p)]}{H_{p,N} - (d_{p,N} + l_p)\sin(\sigma_{p,N})} \quad (8)$ $\varepsilon_{p,N} = d_{R,N} - d_{p,N} \quad (9)$
 <p>(b) yaw</p>	$\phi_{y,N} = \text{atan}\left(\frac{X_{y,N}}{d_{y,N}}\right) \quad (10)$ $d_{R,N} = \sqrt{X_{y,N}^2 + d_{y,N}^2} \cos(\sigma_{y,N} - \phi_{y,N}) \quad (11)$ $\varepsilon_{y,N} = d_{R,N} - d_{y,N} \quad (12)$
 <p>(c) roll</p>	$\gamma_r = \arctan\left(\frac{H}{l_r}\right) \quad (13)$ $H_{r,N} = \sqrt{l_r^2 + H^2} \sin(\sigma_{r,N} + \gamma_r) \quad (14)$ $d_{R,N} = \frac{H_{r,N}d_{r,N}}{H_{r,N} - (l_r + X_r)\sin(\sigma_{r,N})} \quad (15)$ $\varepsilon_{r,N} = d_{R,N} - d_{r,N} \quad (16)$

Table 1: Error compensation of the calculated distances by the FSM caused by variations of roll, pitch and yaw.

2.2.2 Yaw

Figure (b) of Table 1 describes the influence of yaw variations on the FSM. Consider that in a certain frame N , a moving body that is shooting a point O_R on a static object on the surface has its yaw changed ($\sigma_{y,N}$). In the former pose ($pose'_N$), the camera computes the distance to point G_R as $Y_N = d_{y,N}$ and $X_N = X_{y,N}$. However, the correct distance in Y direction in the new pose ($pose_N$) is $d_{R,N}$. Equations (10) and (11) show the relation between the wrong and correct distances, where $\phi_{y,N}$ is the angle in which the camera sees the object in the former pose. The error $\varepsilon_{y,N}$ caused by $\sigma_{y,N}$ is defined as (12). Figure 4 (B) shows error $\varepsilon_{y,N}$ in function of yaw variation ($\sigma_{y,N}$), $\phi_{y,N}$ and apparent distance to the point ($Y_N = d_{y,N}$). First, when $\phi_{y,N} = 0$ rad, we can observe that $\varepsilon_{y,N}$ increases with the increase of Y_N and absolute value of $\sigma_{y,N}$. For similar conditions of Y_N and $\sigma_{y,N}$, when $\phi_{y,N} \neq 0$ the errors are shifted to the left or right according to the value of $\phi_{y,N}$. From this analysis we can observe that small yaw variations such as $\sigma_{y,N} = 0.1$ rad results in

$\varepsilon_{y,N} > 0.1$ m for the adopted range of the parameters in the example, what shows that yaw variations cause smaller errors in the distances computed by the FSM comparing to the pitch variations.

2.2.3 Roll

Finally, Figure (c) of Table 1 shows the influence of roll variations on the FSM. Consider a camera tracking a point O_R on a static object of height h on the surface in a certain frame N . In the same frame, the moving body is affected by an unevenness on the flat surface, varying its roll ($\sigma_{r,N}$). This variation changes height H to $H_{r,N}$ and the camera computes the distances by the FSM in relation to a wrong flat surface, $FS_{r,N}$. In this wrong surface, the projection of point O_R becomes $P_{r,N}$ and $Y_N = d_{r,N}$ is calculated. However, the correct distance is the one computed with G_R , the projection of O_R on the real surface on the ground, resulting in $Y_N = d_{R,N}$. The relation between the wrong ($d_{r,N}$) and correct ($d_{R,N}$) distances is shown by (13), (14) and (15), where l_r is the axis of rotation

and $\gamma_{r,N}$ is the angle between the camera height H and l_r . We define the error caused by roll variation $\sigma_{p,N}$ as $\varepsilon_{p,N}$, which is the difference between wrongly estimated distance ($d_{r,N}$) and correct one ($d_{R,N}$), as shown in (16). Figure 4 (C) shows error $\varepsilon_{r,N}$ in function of camera roll variation ($\sigma_{r,N}$), X_r and apparent distance to the point ($Y_N = d_{r,N}$). We can observe that $\varepsilon_{r,N}$ increases with the increase of $\sigma_{r,N}$, X_r and Y_N . For roll variations of $\sigma_{r,N} = 0.1$ rad, $\varepsilon_{r,N} > 2.0$ m can occur for the adopted range of the parameters in the example, but those errors are still smaller than the ones caused by pitch variation.

3 Proposed Method

This section is divided into three parts: A) proposed height estimation method, B) compensation of roll, pitch and yaw variations and C) proposed algorithm.

3.1 Proposed Height Estimation

Although the FSM has the ambiguity limitation when computing VO, in fact, the difference of the obtained displacements caused by the object ($\Delta Cam_N, \Delta Cam'_N$) contains useful information. First, the presence of objects in the scene influence the apparent displacements in pixels seen from the camera. Such difference in apparent displacements is explored in [12] to find irregularities in the optic flow and detect precipices. On the other hand, no further information can be extracted by existing techniques. Here, our method is based on the principle that since the FSM assumes known H and α , then we can assume that the resulting projections are also function of these dimensions. If we further observe the geometrical relations caused by the FSM and triangulation in two positions (Case B in Figure 2), we can in fact obtain geometrical relations in function of H and α , as shown in (17), (18) and (19). From these equations and (4), we obtain (20) to (22). Several relations can be observed from the equations. First, the object causes an extra amount of apparent displacement, defined as m_N , and it is proportional to the object height h and camera height H . Second, the object height is function of the correct camera displacement ΔCam_N and wrong apparent displacement $\Delta Cam'_N$. As afore mentioned, ΔCam_N can be estimated by traditional VO and even though this technique will be applied, it is not the focus of this work.

$$\frac{H}{d_{R,N-1}} = \frac{h}{m_N + n_N} \quad (17)$$

$$\frac{H}{d'_{R,N}} = \frac{h}{n_N} \quad (18)$$

$$d'_{R,N} = d_{R,N} - m_N \quad (19)$$

$$\Delta Cam'_N = d_{R,N-1} - d_{R,N} + m_N = \Delta Cam_N + m_N \quad (20)$$

$$m_N = \frac{h(d_{R,N-1} - d'_{R,N})}{H} = \frac{h(d'_{R,N-1} - d'_{R,N})}{H} \quad (21)$$

$$h = H \left(1 - \frac{\Delta Cam_N}{\Delta Cam'_N}\right) \quad (22)$$

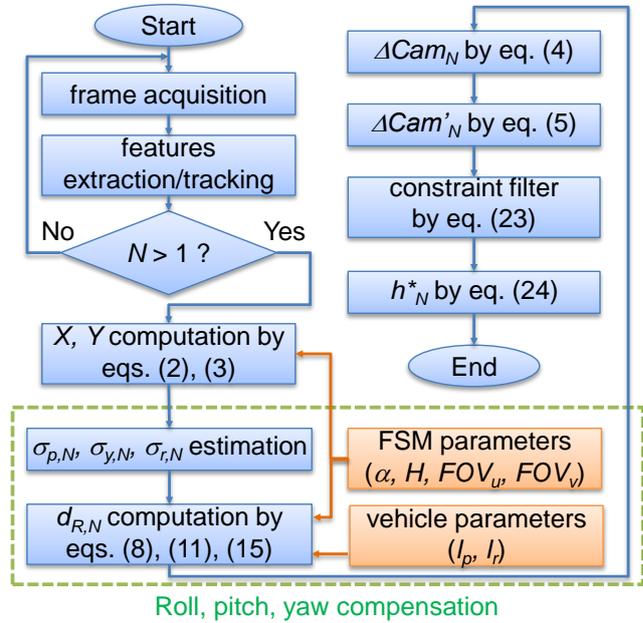


Figure 5: Flowchart of the proposed height estimation method and compensations.

3.2 Compensation of Roll, Pitch and Yaw Variations

The equations displayed in Table 1 show that the compensation of the influences caused by roll, pitch and yaw are straightforward and can be easily done if $\sigma_{p,N}$, $\sigma_{y,N}$ are $\sigma_{r,N}$ are known. Thus, for each acquired frame by the camera, we compute these variations and substitute them with the constant and known parameters of the vehicle (l_p, l_r) and the FSM (α, H, FOV_u, FOV_v) in (8), (11) and (15), obtaining the compensated value $d_{R,N}$. The main steps of the proposed compensations are illustrated in the bottom part of Figure 5.

3.3 Proposed Algorithm

Figure 5 summarizes the flow of the proposed algorithm. Frames are acquired, features are extracted and tracked in two frames. Next, the corresponding values of X and Y are computed by the FSM. Roll, pitch and yaw variations ($\sigma_{p,N}$, $\sigma_{y,N}$, $\sigma_{r,N}$) are estimated by VO and we compensate their influences using (8), (11) and (15), as explained in the previous section. The camera displacement (ΔCam_N) and compensated apparent displacement of the object ($\Delta Cam'_N$) are calculated by the FSM. Here, we propose to apply two filters. First, a filter based on displacement constraints to verify the applied compensations. Even though

the compensations strongly rely on the estimated roll, pitch and yaw, wrong estimations on the contrary lead to high errors. The filter works according to the observed in Figure 3. If a tracked point belongs to an object above the surface, then its apparent displacement must be bigger than the camera displacement ($\Delta Cam_N < \Delta Cam'_N$). Therefore, we check this condition by calculating the apparent displacement of the object with ($\Delta Cam'^c_N$) and without ($\Delta Cam'^{mc}_N$) compensations. If one of them satisfies the condition, then we adopt this displacement as $\Delta Cam'_N$. If both or none of them satisfies the condition, then we use the average of the two displacements to estimate the height, as show in (23). Finally, the median (h^*_N) of the previous estimations ($h_1, h_2, \dots, h_{N-1}, h_N$) is also applied to filter eventual noises, as (24).

$$\Delta Cam'_N = \begin{cases} \Delta Cam'^c_N & \text{if } \Delta Cam'^c_N > \Delta Cam_N > \Delta Cam'^{mc}_N \\ \Delta Cam'^{mc}_N & \text{if } \Delta Cam'^c_N < \Delta Cam_N < \Delta Cam'^{mc}_N \\ \frac{\Delta Cam'^c_N + \Delta Cam'^{mc}_N}{2} & \text{otherwise} \end{cases} \quad (23)$$

$$h^*_N = \text{median}(h_1, h_2, \dots, h_{N-1}, h_N) \quad (24)$$

4 Experiments

The proposed method was evaluated with data from the KITTI Vision Benchmark Suite (called hereon as "KITTI") [15] and processed with a computer Intel(R) Core(TM) i7-4600, 2.10 GHz, operating system Ubuntu (TM) 14.04, Eclipse (TM) development environment and OpenCV libraries [16]. Here, we want to verify the validity and effectiveness of the proposed height estimation and the proposed compensations of errors caused by roll, pitch and yaw variations. Thus, all estimated heights were done with two methods for later comparison: i) with roll, pitch and yaw compensation and ii) without compensation.

4.1 Applied VO

In order to estimate the camera displacement ΔCam_N in each frame, we adopted a simple VO with rotation estimation by Nister's 5-point algorithm [17] and translation by the FSM using features in a ground region close to the camera (details in the Appendix section), similarly to [12]. The parameters (FOV_u, FOV_v, W, V, H , etc) necessary for the experiments were adopted according to the provided by the KITTI. The camera inclination in relation to the ground was not directly provided, but we estimated that $\alpha = 1.1^\circ$ using the provided velodyne data. The feature extractor applied was FAST [18]. The features were automatically extracted when their number was bellow 1500 features. In order to evaluate the proposed monocular height estimation method, only the left images of the grayscale camera of the KITTI were used.

4.2 Evaluation Criteria

The evaluation was based on the error (ϵ_N) between the ground truth ($h_{GT,N}$) and the estimated height (h^*_N) in each frame N , according to (25). The ground truth adopted was mainly the height provided by the velodyne, available in the KITTI. Further details can be found in the Appendix section.

$$\epsilon_N = |h_{GT,N} - h^*_N| \quad (25)$$

We also adopted a criterion to consider if the height estimation in a frame failed or not. For example, since we considered only objects above the ground surface, the estimated heights must be higher than 0 m (i.e., $h_{min} = 0$). Furthermore, since the FSM considers objects below the horizon line, all objects used in the experiment should have maximum height equal to the camera height ($h_{max} = H$). All estimated heights outside this maximum and minimum were considered as failure, as detailed in (26).

$$\text{height estimation} = \begin{cases} \text{success} & h_{min} \leq h^*_N \leq h_{max} \\ \text{failure} & \text{otherwise} \end{cases} \quad (26)$$

4.3 Results

Experiments were conducted with 10 video sequences of the KITTI, which contained many static objects (parked cars, poles, fences, houses, people, boxes, etc) in the scene. In total, height was estimated 4561 times with 436 objects. Objects with height $0 \leq h \leq H$ and distance from 4 to 31 m from the camera were used. The obtained results are summarized in Table 2, Table 3 and Figure 6.

Data	Video sequence (KITTI)	Valid frames	Valid objects
1	00	1103	85
2	05	384	35
3	07	229	18
4	08	373	29
5	09	228	23
6	13	347	43
7	15	471	32
8	16	361	35
9	18	407	32
10	19	1094	104
total	-	4997	436

Table 2: Summary of the conducted experiments.

	without comp.	with comp.
comp. parameter	-	pitch, yaw, roll
average ϵ_N [m]	0.23	0.20
maximum ϵ_N [m]	1.51	1.12
number of failures	283	65

Table 3: Comparison of the obtained errors with and without the proposed compensations.

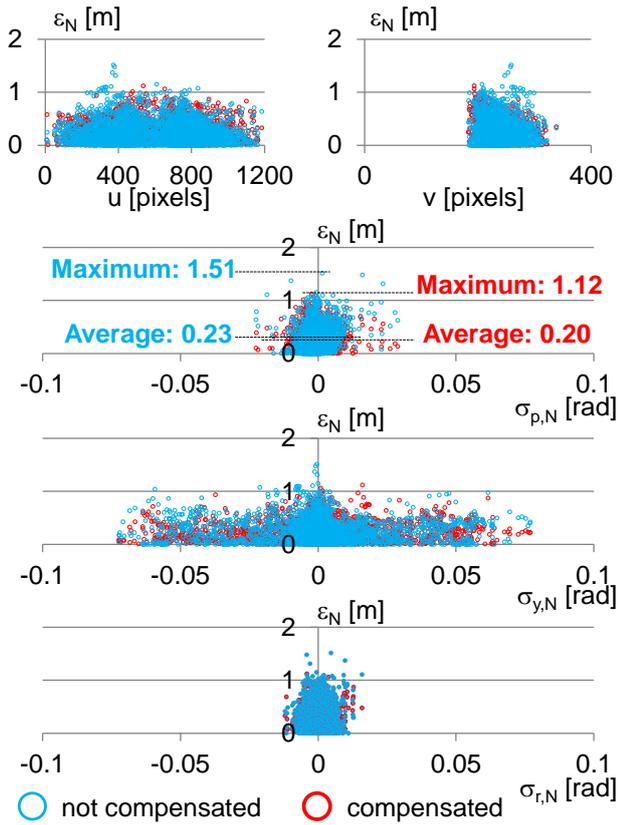


Figure 6: Obtained errors of the estimated heights of all data in the experiments. The results are displayed according to pixels (u, v) and unevennesses ($\sigma_{p,N}$, $\sigma_{y,N}$, $\sigma_{r,N}$).

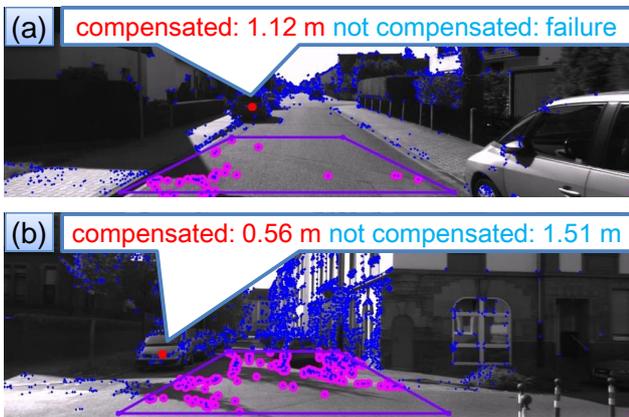


Figure 7: Maximum error cases with and without compensation.

5 Discussions

Table 2 shows the 10 data and corresponding video sequence belonging to the KITTI. The number of valid frames and objects used in each data are also displayed. The estimated heights of all objects chosen within this data are presented in relation to the pixel positions (u, v) in Figure 6. We can observe that the objects were well distributed along pixel u direction, covering many possible positions during the experiments. On the other hand, due to the geometrical

limitations of the FSM only pixels below the horizon line (i. e., $v > 200$ pixels) were used, but we can also observe objects distributed along this interval. Next, the used data is displayed in relation to $\sigma_{p,N}$, $\sigma_{y,N}$ and $\sigma_{r,N}$. Variations of roll ($\sigma_{r,N}$) and pitch ($\sigma_{p,N}$) in the used data were smaller than those of yaw ($\sigma_{y,N}$): while the magnitude of roll and pitch variations were within 0.03 rad, the magnitude of the yaw variations were over 0.05 rad. The average error of all used data resulted in 0.20 m with the 3DOF compensations and in 0.23 m when no compensations were done. The maximum error resulted in 1.12 m for the compensated case and 1.51 m for the non-compensated one. Since both average and maximum errors were improved with the compensations, we can affirm that the proposed method is valid and effective. These cases of maximum errors are illustrated in Figure 7. In (a), the case of maximum error with compensations is displayed. The point was chosen too close to the horizon line, becoming very sensitive to noises and causing the high error. However, for the same case, the method without compensation failed to estimate the height. In (b), the case of maximum error without compensation occurred when the camera estimated the object height while climbing a slope. As presented in the previous sections, pitch variations cause higher errors comparing to yaw and roll, and a high error of 1.51 m was expected. Nevertheless, when the proposed compensations were applied in the same data, the estimation error dropped to 0.56 m. Furthermore, the distribution of $\sigma_{p,N}$, $\sigma_{y,N}$ and $\sigma_{r,N}$ in the experiments (Figure 8) shows that even though the data was taken on public roads, most of the pose variations were below 0.01 rad: 98.8% of the $\sigma_{p,N}$, 76.6% of the $\sigma_{y,N}$ and 99.7% of the $\sigma_{r,N}$. We can observe that the errors were higher for higher values of pitch variation $\sigma_{p,N}$ when no compensation was done, as expected by Figure 4. Even though in average the compensated and non-compensated methods had similar errors (0.20 and 0.23 m), the difference of both errors was higher for higher variations of $\sigma_{p,N}$, $\sigma_{y,N}$ and $\sigma_{r,N}$. During the experiments objects with distances further than 20 m were used, what was enough to cause more than 5 m error according to Figure 4. Since the obtained errors were below this expected ones, we can affirm that the obtained results were satisfactory. Examples of cases with higher yaw variations are shown in the Appendix section. We can observe a significant difference of the proposed method in terms of number of successful height estimations. During the experiments, the compensated method failed to estimate height 65 times, while the non-compensated one failed 283 times (4 times more). Such failures require further analysis in future work, but the effectiveness of the proposed compensations became clear.

Although this work relied on VO, it didn't focus on improving its accuracy. However, we estimated that the applied VO had around 13% error per frame and influenced directly the results, what means that the proposed method becomes more accurate with the improvement of VO itself. Such VO errors led to wrong

estimation of camera pose, generating higher height estimation errors and examples are shown in the Appendix section. Even though the proposed method improved the average and maximum error of the estimated heights, some limitations still exist. The small camera pose variation per frame suggests that further evaluation with higher variations is necessary, by for example, increasing the moving body's velocity and analyzing the relation between camera frame rate and obtained height estimation errors. Finally, the experiments made evident another necessary correction: we considered angular variations (rotation) during the proposed compensations, however translations in Y also occurred. According to the FSM, such translations change the camera height H and must be considered when computing the distance to objects. We believe that this consideration can further increase the accuracy of our height estimation method.

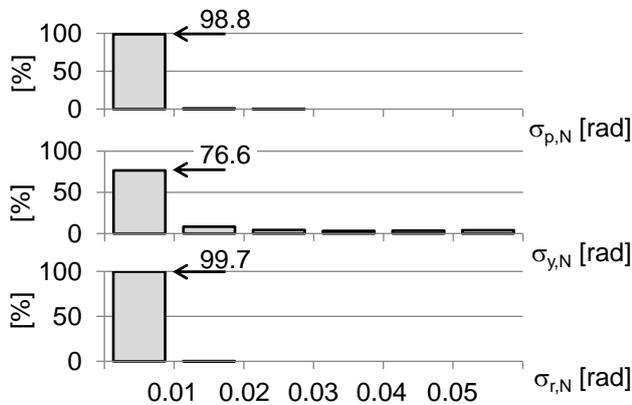


Figure 8: Distribution of the pose variation detected in the used data.

6 Conclusions and Future Work

A novel method of monocular height estimation with 3DOF compensation of roll, pitch and yaw variations was proposed. The method can estimate height of objects with only two frames of a monocular camera. Experiments outdoors with the KITTI benchmark data (4997 frames and 436 objects) resulted in improved accuracy of the estimated heights from a maximum error of 1.51 m to 1.12 m and reduced number of estimation failures by 4 times, proving the validity and effectiveness of the proposed method. This method can be enhanced by improving monocular visual odometry techniques and considering translational variations of the camera during height estimation. Further investigation about influences of frame rate, moving velocity and robustness are planned in the future.

Conflict of Interest The authors declare no conflict of interest.

References

[1] A. M. Kaneko, K. Yamamoto. Monocular Height Estimation by Chronological Correction of Road Uneven-

ness. 2016 IEEE/SICE International Symposium on System Integration (SII2016). Sapporo, Japan. 2016. <https://doi.org/10.1109/SII.2016.7843971>

[2] A. M. Kaneko, K. Yamamoto. Landmark Recognition Based on Image Characterization by Segmentation Points for Autonomous Driving. SICE Multi-Symposium on Control Systems 2016 (MSCS2016). Nagoya, Japan. 7-10 March, 2016. <https://doi.org/10.1109/SICEISCS.2016.7470160>

[3] M. Aly. Real Time Detection of Lane Markers in Urban Streets. IEEE Intelligent Vehicles Symposium. Eindhoven. 4-6 June, 2008. <https://doi.org/10.1109/IVS.2008.4621152>

[4] T. Wu, A. Ranganathan. A Practical System for Road Marking Detection and Recognition. IEEE Intelligent Vehicles Symposium (IV). Alcalá de Henares. 3-7 June, 2012. <https://doi.org/10.1109/IVS.2012.6232144>

[5] P. Foucher, Y. Sebsadji and J. P. Tarrel, P. Nicolle. Detection and Recognition of Urban Road Markings Using Images. IEEE International Conference on Intelligent Transportation Systems (ITSC). Washington DC, USA. 2011. <https://doi.org/10.1109/ITSC.2011.6082840>

[6] D. Scaramuzza. 1-Point-RANSAC Structure from Motion for Vehicle-Mounted Cameras by Exploiting Non-holonomic Constraints. International Journal of Computer Vision, Volume 95, issue 1, 2011. <https://doi.org/10.1007/s11263-011-0441-3>

[7] M. W. Maimone, Y. Cheng, L. Matthies. Two years of Visual Odometry on the Mars Exploration Rovers. J. Field Robotics 24(3): 169-186 (2007). <https://doi.org/10.1002/rob.v24:3>

[8] M. K. Momeni C. H. S. Diamantas, F. Ruggiero, and B. Siciliano. Height Estimation from a Single Camera View. In Proceedings of the International Conference on Computer Vision Theory and Applications (VISIGRAPP 2012) ISBN 978-989-8565-03-7, pages 358-364. DOI: 10.5220/0003866203580364

[9] P. Viswanath, I.A. Kakadiaris and S. K. Shah. A Simplified Error Model for Height Estimation using Single Camera. 2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops. 2009. <https://doi.org/10.1109/ICCVW.2009.5457466>

[10] N. Pears, B. Liang, Z. Chen. Mobile Robot Visual Navigation Using Multiple Features. EURASIP J. Adv. Sig. Proc. 2005(14): 2250-2259. 2005. <https://doi.org/10.1155/ASP.2005.2250>

[11] A. Wedel, T. Schoenemann, T. Brox, D. Cremers. WarpCut - Fast Object Segmentation in Monocular Video. 29th DAGM Symposium, Heidelberg, Germany. Proceedings, pp. 264-273. 2007. http://doi.org/10.1007/978-3-540-74936-3_27

[12] J. Campbell, R. Sukthankar et al. A Robust Odometry and Precipice Detection System Using Consumer-grade Monocular Vision. Proceedings of the 2005 IEEE International Conference on Robotics and Automation, 2005. ICRA 2005. Pp 3421-3427. 2005. <https://doi.org/10.1109/ROBOT.2005.1570639>

[13] S. Lovegrove, A. J. Davison, J. I. Guzman. Accurate Visual Odometry from a Rear Parking Camera. IEEE Intelligent Vehicles Symposium (IV). Pp 788-793. 2011. <https://doi.org/10.1109/IVS.2011.5940546>

[14] N. Nourani-Vatani, et al. Practical Visual Odometry for Car-like Vehicles. IEEE International Conference on Robotics and Automation (ICRA). Kobe, Japan. 2009. <https://doi.org/10.1109/ROBOT.2009.5152403>

[15] A. Geiger, P. Lenz and R. Urtasun. Are We Ready for Autonomous Driving? The KITTI Vision Benchmark Suite. Conference on Computer Vision and Pattern Recognition (CVPR). 2012. <http://doi.org/10.1109/CVPR.2012.6248074>

[16] Bradski, G. Opencv. Dr. Dobbs Journal of Software Tools. 2000.

- [17] D.Nister. An Efficient Solution to the Five-Point Relative Pose Problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 26, No. 6. 2004. <https://doi.org/10.1109/TPAMI.2004.17>
- [18] E. Rosten, R. Porter and T. Drummond. FASTER and Better: A Machine Learning Approach to Corner Detection. *IEEE Trans. Pattern Analysis and Machine Intelligence*. 2010. <https://doi.org/10.1109/TPAMI.2008.275>
- [19] H. Hirschmuller. Accurate and Efficient Stereo Processing by Semi-Global Matching and Mutual Information. *International Conference on Computer Vision and Pattern Recognition*, 2005. <https://doi.org/10.1109/CVPR.2005.56>

Appendices

In this section, we provide further details of the experimental conditions and examples of compensations.

A Experimental Conditions

We further explain the experimental conditions adopted in the paper. The experiments were conducted according to Figure 9. The algorithm executed VO automatically for each frame (a), which were shifted manually by a human operator. Among all the extracted feature points, the operator chose with the mouse any point on a desired object (b). Here, since the FSM has geometrical restrictions, only points below the horizon (approximately in $v = \frac{V}{2}$) were chosen. The chosen point (in red) was tracked over the frames until the last frame possible (c), and its height and ground truth were computed and stored in each frame. The estimated position, stored data and extracted points of the current object were reseted after the heights were estimated (d) and the process repeated for all used data.

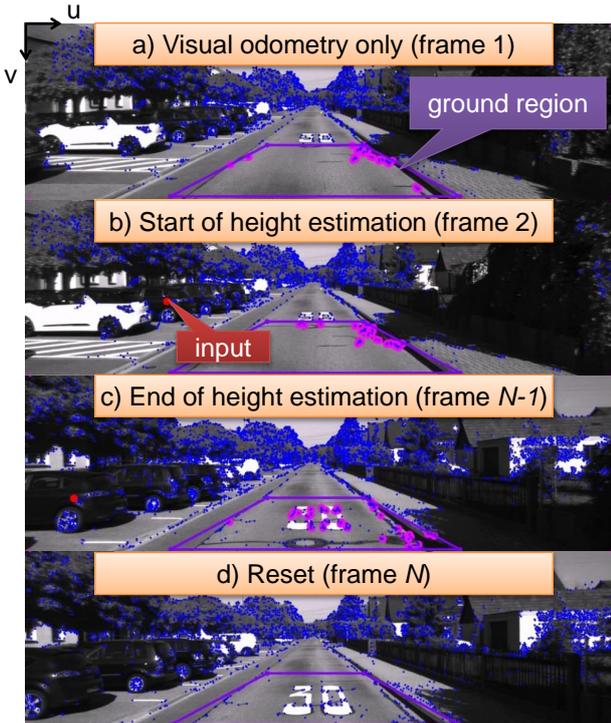


Figure 9: Adopted sequence to estimate the height of an object during the experiments.

First, all estimated heights and relevant data were saved and later verified for validity. Frames whose ground region contained few or no features (due to light conditions for example) or had a moving object were considered invalid (Figure 10). Tracking was also verified (Figure 11). Although we needed only two frames per estimation, we focused on features consistent in many frames (a minimum of 3 estimations per object). Thus, in case the tracking failed before the fourth frame, that object and estimations were considered invalid. If the tracking failed after the fourth frame, only the estimations before that were considered valid.

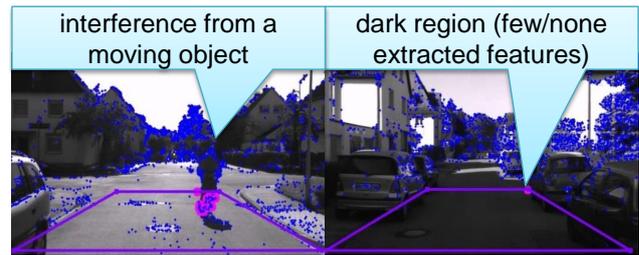


Figure 10: Examples of frames considered invalid: dark regions (right) and moving object (left).

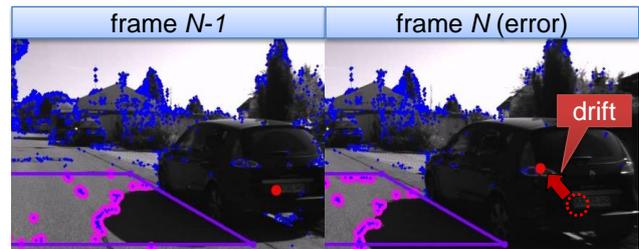


Figure 11: Example of tracked feature considered invalid.

We adopted as ground truth the height from the velodyne. The necessary information for converting the camera-velodyne coordinates were also provided by the KITTI. Due to limitations of the sensor resolution, we chose as ground truth the closest pixel with available velodyne data. This search was also automatically done by the algorithm in each frame. However, since the velodyne fails in some situations, we also computed the height by the disparity from the left and right images of the camera, using Semi Global Block Matching [19] in OpenCV. For each point chosen by the operator, its corresponding velodyne data was recorded. Since the camera was moving forward and approaching the object, we considered that the depth from the velodyne to the object on the next frame should become smaller than the one in the previous frame. In case it wasn't, we considered that the height estimated by the velodyne was also not consistent, and the one calculated by the stereo camera was used instead (Figure 12). In case both depths from the sensors were not smaller than the previous one, then the current estimation and frame were considered invalid. Since we considered the initial frame of each object as reference, variations in Y of the camera itself were also compensated when computing the ground

truth.

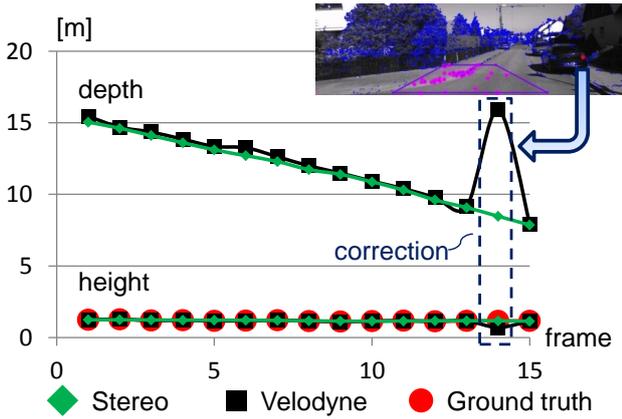


Figure 12: Example of adopted ground truth and correction. Data from the stereo camera was used when the velodyne data was considered inconsistent.

B Further Examples of Compensations

Even though there were few occurrences of yaw variations higher than 0.01 rad per frame during the experiments, we can further observe the benefits of the compensations in the example in Figure 13, which exemplifies the estimated heights during a curve. In the first frames, the camera was mostly moving forward so that ϵ_N was small for both compensated and not compensated cases. When the yaw variation started to increase the errors also increased. However, in frame 10 we can see ϵ_N decreasing to a minimum due to the decrease of $\phi_{v,N}$, increasing again when $\phi_{v,N}$ started increasing, as foreseen by Figure 4.

Figure 14 presents other examples of compensation. The cases in Figure 14 (iii) to (vi) had smaller average errors per frame after the compensations. On the other hand, even though the proposed method im-

proved the average and maximum error of the estimated heights, some limitations as in Figure 14 (i) and (ii) still exist. We believe that such errors were caused by the errors of the adopted VO itself (13 % error) and improvements will be further investigated in future work.

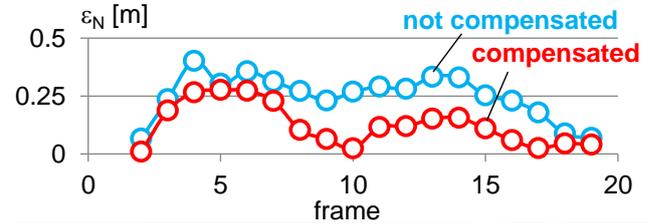


Figure 13: Example of yaw compensation.

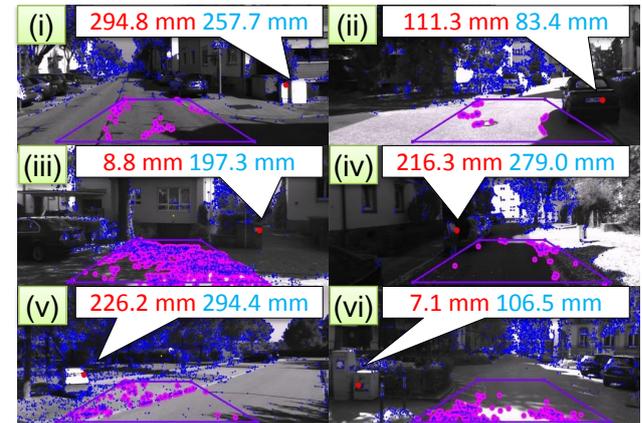


Figure 14: Examples of objects and average ϵ_N per frame. The numbers in blue (right) are the non-compensated cases and the numbers in red (left) are the compensated ones.