An Embedded Multi-Sensor Data Fusion Design for Vehicle Perception Tasks

Mokhtar Bouain[1,2], Karim M. A. Ali[1], Denis Berdjag[1], Nizar Fakhfakh[2] and Rabie Ben Atitallah[1]
[1] Univ. Valenciennes, CNRS, UMR 8201- LAMIH, F-59313 Valenciennes, France
[2] Navya Company, Paris, France
Email: {mokhtar.bouain, karim.ali, denis.berdjag, rabie.benatitallah}@univ-valenciennes.fr
Email: {mokhtar.bouain, nizar.fakhfakh}@navya.tech

Abstract —Nowadays, multi-sensor architectures are popular to provide a better understanding of environment perception for intelligent vehicles. Using multiple sensors to deal with perception tasks in a rich environment is a natural solution. Most of the research works have focused on PC-based implementations for perception tasks and very few concerns have been addressed for customized embedded designs. In this paper, we propose a Multi-Sensor Data Fusion (MSDF) embedded design for vehicle perception tasks using stereo camera and Light Detection and Ranging (LIDAR) sensors. A modular and scalable architecture based on Zynq-7000 SoC was designed.

Index Terms—Sensor Fusion, Embedded Systems, FPGA, Intelligent Vehicle.

I. Introduction

Intelligent Transportation System (ITS) applications are widely involved in our daily life. Among these applications, we can mention intelligent vehicles, Advanced Driver Assistance System (ADAS) for lane detection, parking assist, tracking [1], cross traffic alert, pedestrian detection... [2] [3] [4].

Building ITS systems requires the use of different types of sensors to improve the traffic safety, to ensure the reliability of navigation tasks and efficient perception. The main goal to use multi-sensors architecture is to achieve tasks that cannot be performed with a single sensor. In fact, a single sensor is limited in the amount of details that can be captured when used to measure a physical quantity. This limitation arises because one single sensor generally suffers from many problems: (1) Field Of View limited coverage (FOV). (2) limited temporal coverage due to the limited rate of sensor acquisition. (3) The breakdown and dysfunction of sensor affect the system reliability. (4) The measurements from individual sensors are limited to the precision of the employed sensing element. (5) The measured data is uncertain when some features are missed (e.g. occluded objects). Therefore, using multiple sensors is a potential solution to overcome the problems as mentioned above.

Combining information from multi-sensor system introduces new challenges [5]. One of the important challenges is a spatio-temporal task: the spatial part is the alignment of frame sensors while the second is handling the update rates of sensors. The alignment process consists of finding the relation between the coordinates of sensor frames to ensure the transformation from one frame into another. The second challenge is the operational timing in the case of homogeneous or heterogeneous sensors. The operation frequencies of the sensors are different. Consequently, a well-designed data fusion method should incorporate multiple time scales in order to deal with such timing variations in data [5]. We mention others challenges such as data association and the architecture that will be used to perform data fusion either centralized or decentralized.

Many classifications are proposed for data fusion techniques in the literature [6]. Among these classifications is based on the abstraction level of the employed data. This class is subdivided into three levels: (1) Low level fusion where the raw data will be directly fused. (2) Medium level, in this case, the characteristics (e.g. shape, texture, and position) are combined to obtain the features that could be employed to perform specific tasks. (3) High level fusion where the fusion is carried-out at the level of decision.

Most of the research works have focused on PC-based implementations for perception tasks [7] and very few concerns have been addressed for customized embedded designs. FPGA reconfigurable circuits are considered as a preferable choice to implement perception tasks for different reasons: (1) FPGA offers high performance computing power at lower operating frequencies. (2) They can realize massively parallel architectures by profiting from the huge amount of programmable logics available on a single chip. (3) FPGAs are good candidates for building energy efficient systems due to their low power consumption. (4) Sensors are modular devices that use standard communication ports like CAN bus, Ethernet, FPGA Mezzanine Card (FMC)... Thus, FPGA can play a role as a communication centric platform between the connected sensors [8].

In this paper, we present a Multi-Sensor Data Fusion (MSDF) design for vehicle perception tasks for embedded systems. This is mainly based on Zynq-7000 heterogeneous System-on-Chip (SOC) platform.

Corresponding author email: mokhtar.bouain.@univ-valenciennes.fr

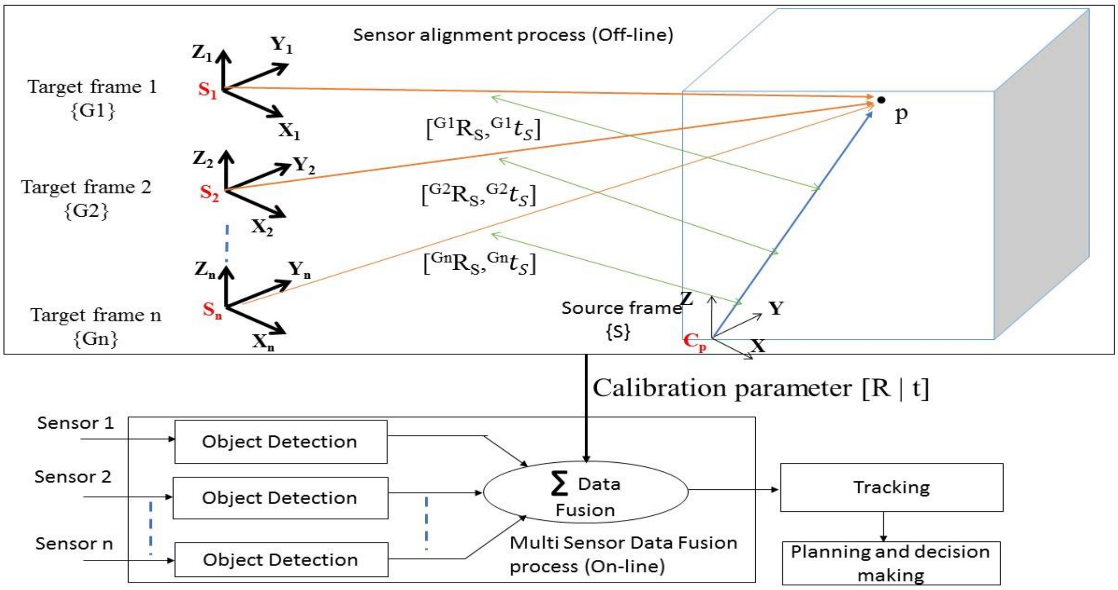

Fig. 1: Multi sensor data fusion framework.

Our proposed design represents a modular architecture based on FMC I/O interface for automotive applications.

The rest of the paper is organized as follows. Section II describes the MSDF framework. In section III, we describe the proposed design on the embedded platform. Section V presents the results of the functions that already implemented. Finally, future works are discussed in Section VI.

II. MSDF FRAMEWORK

Based on the described problem and solution in [7], we aim to represent the position uncertainty of a detected object as a 2D Gaussian distribution. We extend the proposed approach to *n* sensors and we append the alignment process.

These sensors are homogeneous and/or heterogeneous and their task is to measure the position of the detected obstacles. Fig. 1 shows the structure of MSDF generic framework for *n* given sensors. It consists of two main processes: **(1) Sensor alignment process (off-line)**. The inputs of this process are sensors data while the outputs are the calibration parameters (rotation matrix and translation vector). This process is an extrinsic calibration between different sensors (source and targets) allows estimating the relative position of point *p* in a common frame. **(2) Object detection process (on-line)**. In this step, there are *n* processing chains each of them provides a list of the detected objects. With the conjunction of the calibration parameters obtained from sensor alignment process, we are able to fuse the data together to better detect the objects in the surroundings. In this work, we are interested in LIDAR and stereo camera sensors. We aim to use Bayesian fusion technique on objects of these two lists provided by the two sensors to get a new list of fused objects. In the following subsections, we will detail each task.

A. Frames alignement

To efficiently perform sensor fusion, the sensors should be calibrated. The calibration process is an alignment procedure of a given sensor frames. That is to say, find the relation between the coordinates of sensor frames to ensure the transformation from a frame into another.

To carry-out this process we used the method described in [8] based on least squares analytical solution to find the unknown 6 Degree of Freedom (DOF) transformation between the two sensor frames. This method provides an analytical solution using a white board with three black lines in the middle. It is based on point-normal vector correspondences and it is using the detected lines from the camera frame and the end-points from LIDAR sensor to establish the closed-form solution. Fig. 2 shows the main steps to perform the sensor alignment process.

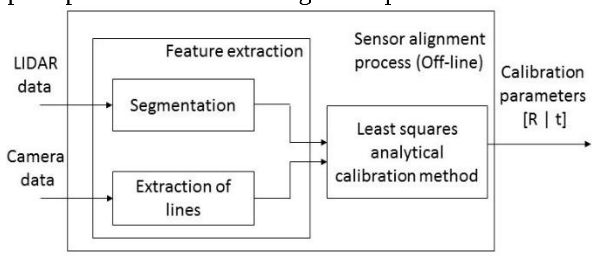

Fig. 2: Calibration process.

The first step consists to extract the co-features for both types of sensors: line detections for cameras and segmentation process for LIDAR sensors to make fully automated feature acquisition. To detect the lines, the Hough transform is used and to detect the end-points, we use the segmentation process described in the next section**.** The second step is to establish and solve the closed-form to obtain the rotation matrix and translation vector allowing alignment of sensor frames.

To summarize, the goal is to find the rigid transformation $[{}^{C}R_{L} \mid {}^{C}t_{L}]$, where ${}^{C}R_{L}$ is the rotation matrix and ${}^{C}t_{L}$ is the translation vector allow to make the transformation from LIDAR to camera sensor frame. It consists to determine the correspondence of a given LIDAR point represented as $p_L = [x_L; y_L; z_L]^T$ located into the frame of the LIDAR sensor *{L}*, in the frame of the camera *{C}*. Let $p_C = [x_C; y_C; z_C]^T$ be the correspondence of p_L, so we write the transformation between the camera and LIDAR frames as follow:

$$p_c = {}^{C}R_L\, p_L + {}^{C}t_L \quad (1)$$

Figure 3 illustrates the relevant transformations between the different frames for LIDAR and stereo vision system. We distinguish four geometric transformations:

- $[{}^{L}R_w \mid {}^{L}t_w]$: Transformation between real world and LIDAR.
- $[{}^{LC}R_w \mid {}^{LC}t_w]$: Transformation between real world and camera (Left).
- $[{}^{LC}R_L \mid {}^{LC}t_L]$: LIDAR to left camera transformation.
- $[{}^{LC}R_{RC} \mid {}^{LC}t_{RC}]$: Right to left camera transformation.

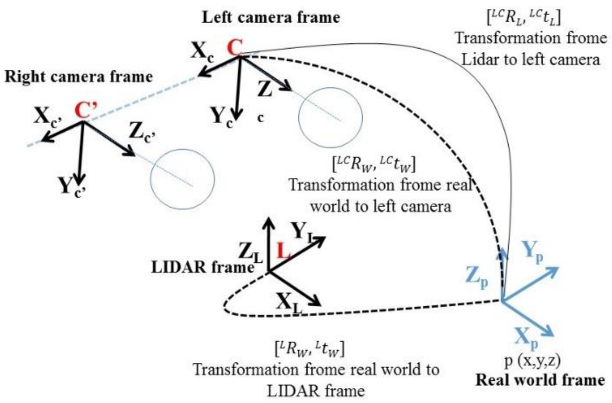

Fig. 3: Transformations between different frames.

B. *Exraction of LIDAR points*

To extract the projected points of the source sensor on the calibration board, the automatic extraction approach by differentiation of the measurements and background data in static environments is often used. However, in this work, this task is carried out by using a segmentation process. Each segment (cluster) is defined as a set of points and is composed of a minimum number of points distant according to a threshold distance denoted *Thr*.

Therefore, if $dist(p_i;\ p_{i+1}) < Thr$ then a segment is defined with C_i as its centroid. Where p_i is the impact point of the LIDAR sensor, $dist(p_i\ ,\ p_{i+1})$ is the Euclidean distance between two adjacent points and *Thr* is the required threshold. The coordinate of each centroid C_i is calculated as follows: $\left(\sum \frac{p_{xi}}{n}, \sum \frac{p_{yi}}{n}\right)$ where n is the number of points. We can add another parameter to fix the minimum number of points that make a segment.

C. *Sum of Absolute Difference stereo matching algorithm*

Figure 4(a) shows how the depth of objects is determined in stereo matching problem. For every pixel x_R in the right image, we try to find its best matching pixel x_L in the left image at the same image line. Assuming two cameras of focal length *(f)* at the same horizontal level, separated from each other by a distance baseline *(b)*. Pixel *(p)* in the space will be located at point (x_R) and point (x_L) in the right and left image respectively. The difference between the two points on the image plane is defined as disparity *(d)* as depicted in Fig 4(b). Therefore, the depth of pixel *(p)* from the two cameras can be calculated using the following equation:

$$deph = \frac{\text{baseline} * \text{focal length}}{\text{disparity}} = \frac{b * f}{(x_R - x_L)} \quad (2)$$

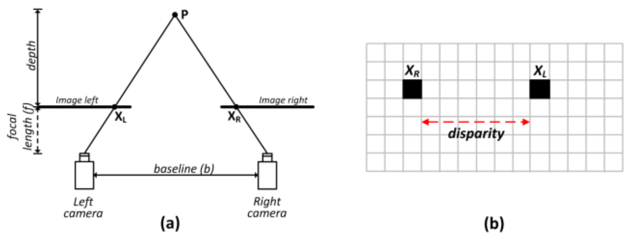

Fig. 4: Calculating the depth of an object in stereo matching problem.

There are different algorithms in the literature that are used to solve the stereo matching problem. In this work, Multiwindow Sum of Absolute Difference (Multi-window SAD) was used with 5-window configuration. Figure 5(a) shows that pixel *(p)* lies in the middle of window (E) while it is surrounded by another four windows named (A, B, C and D).

The four windows are partially overlapped at the border pixel. The dimension of each of them is equal to (winH+1 x winV+1) while the size of the window (E) is equal to (2*cwinH+1 x 2*cwinV+1). A score of a window is equal to the aggregation of its pixels. In 5- window SAD, the correlation score at pixel *(p)* is equal to the score value of window (E) in addition to the best minimum two score values of the other four windows (A, B, C and D). The score is calculated at different disparities such that the best matching between candidates is the one of the minimum scores as shown in Fig. 5(b).

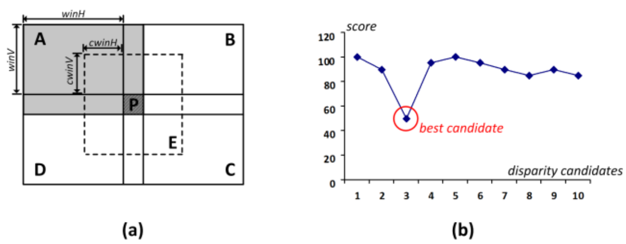

Fig. 5: (a) 5-window SAD configuration (b) Disparity of minimum score is considered as the best matching.

D. MSDF using Bayesian approach

The authors of [9] used the Bayesian Fusion technique to fuse the positions acquired by two sensors in the context of environment perception of autonomous vehicles.

The sensors are employed to detect the positions of obstacles. Position uncertainty is represented using 2D Gaussian distribution for both objects. Therefore, if X is the true position of the detected object, by using the Bayesian fusion, the probability of fused position P_F $[x_F\ y_F]^T$ by the two sensors is given as:

$$P_{prob}(P|X) = \frac{e^{\frac{-(P-X)^T R^{-1}(P-X)}{2}}}{2\pi\sqrt{|R|}} \quad (3)$$

where P is the fused position and R is the covariance matrix are given as:

$$P = \frac{\frac{P_1}{R_1} + \frac{P_2}{R_2}}{\frac{1}{R_1} + \frac{1}{R_2}} \; and \; \frac{1}{R} = \frac{1}{R_1} + \frac{1}{R_2}$$

Where P_1 and R_1 are respectively the position and covariance matrix of sensor 1 and P_2 and R_2 are that of sensor 2. So, we will represent the position uncertainty as 2D Gaussian distribution. To explain how the Bayesian approach works, we simulated the behavior of two sensors which detect an object. Figure 6 presents the results of the simulation.

We explain this approach according to three cases: **Case (1)** Figure 6(a) presents the results when the two sensors have a similar covariance matrix i.e. the fused position (the black crosses) will be in the middle of the tow provided positions by sensors 1 and 2. **Case (2)** (Figure 6(b)) When Sensor 2 has a covariance matrix greater than that of sensor 1, in this case, the fused positions (the black crosses) follows the positions provided by the sensor 1 (blue curve). **Case (3)** (Figure 6(c)) When the sensor 1 has a covariance matrix greater than that of sensor 2, contrary to the previous case the fused positions (the black crosses) follows the positions provided by the sensor 2 (red curve). It is clear that the outcome is a combination of the two measurements weighted by their noise covariances matrices. To summary, the fused results using Bayesian approach follow the measurements provided by the sensor which has the smallest covariance matrix and gives more *trust* to it.

III. EMBEDDED PLATFORM

Figure 7 shows the design architecture of MSDF based on Zynq-7000 SoC. This SoC integrates dual-core ARM Cortex- A9 based processing system (PS) and programmable logic (PL) in a single device. The sensors are coupled to the platform through FMC interface. The image frames coming from the camera are buffered in the DDR memory through AXIDMA communication.

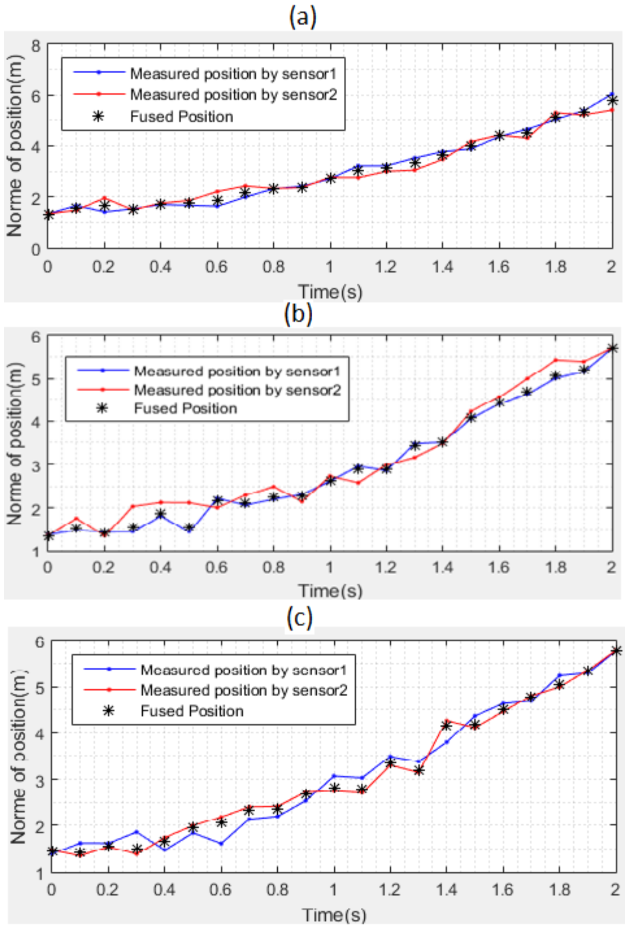

Fig. 6: Fusion of positions using Bayesian approach :(a) The two sensors have similar covariance matrices (b) Sensor 2 has a covariance matrix greater than that of sensor 1 (c) Sensor 1 has a covariance matrix greater than that of sensor 2.

The stored left/right frames are sent sequentially to *Stereo Processing* block for stereo matching processing. Two main steps are executed: disparity calculation to obtain the disparity of detected objects followed by depth calculation to know how far are the objects from the vehicle. On the other hand, the data from LIDAR sensor are processed by LIDAR processing IP.

The inputs for Data Fusion IP are the vectors containing the positions of the detected objects. The output of this IP is a vector containing a fused list of the detected objects. Taking into consideration that the input sensors data must be synchronized according to their time stamps.

The implemented algorithms running on this platform are developed using High-Level Synthesis (HLS) tools. These tools allow compiling C/C++ code into RTL design. Time-to-market is a crucial constraint in the automotive industry; therefore, using HLS tools permit the designers to rapidly test different algorithms and their implementation alternatives within short design cycle. In order to obtain an efficient hardware implementation, the high-level code is subjected to a set of HLS optimization steps [11].

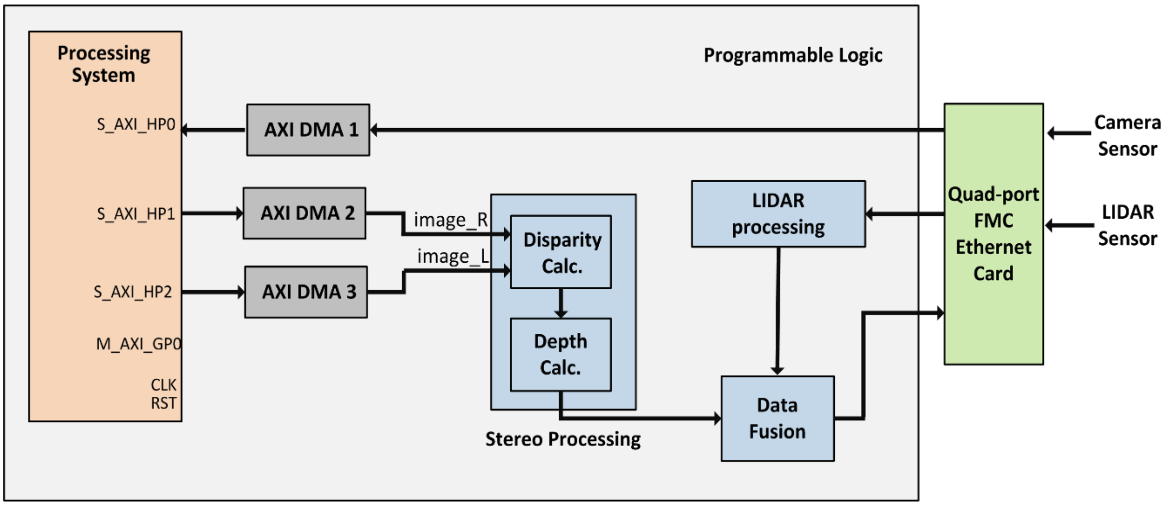

Fig. 7: MSDF architecture based on Zynq-7000 SoC

IV. EXPERIMENTS

A. *Frames alignment*

This process is performed by using a white board with three black lines in the middle. The used sensors are a color camera with 640x480 resolution and a line scan LIDAR with an angular resolution 0.25 degree. The camera is modeled by the standard pinhole model. The calibration board was moved in a 3m to 9m distance range. Figure 8 shows the correspondence between two frames: images and LIDAR sensors. It presents the results using two approaches: **(1)** The original approach using a white pattern contains one black line **(2)** Multi-lines approach using a white pattern contains three black lines. For more details, the interested reader is referred to [10]. Note that, the alignment process task is carried-out on off-line for a specific number of poses taken by both of sensors. So, this step allows finding the rigid transformation between different sensor frames to project all the object detections in the same frame to carry-out the fusion process.

B. *Stereo matching algorithm*

The C-code for Multi-window SAD algorithm was implemented into hardware design by using Vivado HLS 2015.2. During our experiments, we used Vivado 2015.2 design suite to implement our system over Zynq ZC706 FPGA evaluation board (XC7Z045-FFG900) with input grey images of size 640x480. The system was configured for 5-window SAD with the following parameters: winH =23, winV =7, cwinH =7, cwinV =3 and maximum disparity=64. Table I shows the synthesis results for some design alternatives for stereo matching algorithm at different frame rate. The final choice is constrained by how much hardware resources are available after system integration or at what frame rate the system will operate. For example, Design #9 utilized 71% of the available hardware resources at frame rate of 87 frame/s operating at 200 MHz. But we can decrease the frame rate to 73 or 59 frame/s in order to save between 12%-25% of the available HW resources for accelerating the other functionalities in the system.

TABLE I: DIFFERENT DESIGN ALTERNATIVES FOR STEREO MATCHING ALGORITHM

Design alternatives	Slice (54650)	FF (437200)	LUT (218600)	BRAM36K (545)	HW utilization (%)	Frequency (MHz)	Frame rate (frame/s)	Power (in Watt)
#1	26548	69268	80493	177.5	48.5	100	32.58	1.2354
#2	26041	64792	80512	177.5	47.6	150	46.94	1.411
#3	25143	72259	80567	177.5	46	200	59.55	1.6437
#4	33640	105228	89444	233.5	61.55	100	40.55	1.4157
#5	34944	83480	105341	233.5	63.9	150	58.2	1.6478
#6	32616	93435	105423	233.5	59.6	200	73.47	1.9121
#7	40326	109624	130054	289.5	73.79	100	48.64	1.5119
#8	41208	102174	130168	289.5	75.4	150	69.59	1.7952
#9	38980	114617	130097	289.5	71.32	200	87.41	2.109

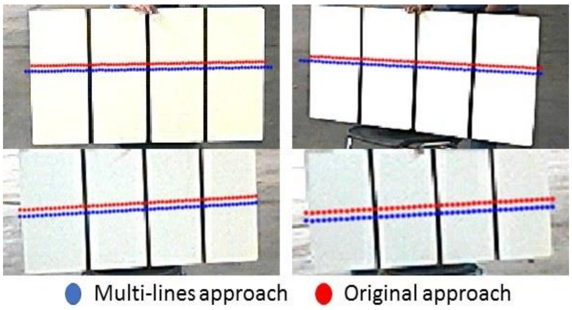

Fig. 8. Calibration results

C. *Power consumption*

Table I also presents the power consumption in Watt for Zynq-7000 SOC at different frame rates. We could notice that the power consumption increases as the frequency increases for the respective values: 100, 150 and 200 MHz. Also, the maximum power consumption in all possible designs is about 2 watts for the design #9 which is still lower than the power consumed by the other potential industrial solutions such as PCs [12].

V. CONCLUSIONS

In these works, we presented an embedded design for MSDF based on stereoscopic camera and LIDAR sensors for vehicle perception tasks. We detailed the main steps to build the MSDF design. The first step is about the sensor calibration i.e. the alignment camera and LIDAR sensor frames. This process allows finding the relation between the coordinates of sensor frames to ensure the transformation from a frame into another. Since we aim to fuse the detected obstacles provided by each sensor, the idea is to represent the position uncertainty as 2D Gaussian distribution. So, we used the Bayesian approach to combine the detected objects. According to the simulation results, we saw that the outcome is a combination of the sensor measurements weighted by their noise covariance's matrices. Otherwise, since the generation of clusters by the LIDAR sensor is faster than the stereo camera, using FPGA solution is a potential solution to improve the power processing. Hence, we can reduce the necessary time to provide the detected object by cameras. For stereo matching, we could obtain frame rate ranges between 32-87 frame/s according to the chosen implementation. Our future work will be focused to implement the second module of stereo processing IP which is the depth calculation in order to generate the clusters that will be fused to that of LIDAR. In addition, we aim to add the RADAR sensor to our platform to improve the accuracy of the obstacle detection tasks.

REFERENCES

[1] S.-F. Hsiao, G.-F. Yeh, and J.-C. Chen, "Design and implementation of multiple-vehicle detection and tracking systems with machine learning," in Digital System Design (DSD), 2014 17th Euromicro Conference on. IEEE, 2014, pp. 551–558.

[2] R. Bishop, "A survey of intelligent vehicle applications worldwide," in Proceedings of the IEEE Intelligent Vehicles Symposium 2000 (Cat. No.00TH8511), 2000, pp. 25–30.

[3] J. Horgan, C. Hughes, J. McDonald, and S. Yogamani, "Vision-based driver assistance systems: Survey, taxonomy and advances," in 2015 IEEE 18th International Conference on Intelligent Transportation Systems, Sept 2015, pp. 2032–2039.

[4] H. Liu, S. Niar, Y. El-Hillali, and A. Rivenq, "Embedded architecture with hardware accelerator for target recognition in driver assistance system," ACM SIGARCH Computer Architecture News, vol. 39, no. 4, pp. 56–59, 2011.

[5] B. Khaleghi, A. Khamis, F. O. Karray, and S. N. Razavi, "Multisensor data fusion: A review of the state-of-the-art," Information Fusion, vol. 14, no. 1, pp. 28 – 44, 2013.

[6] F. Castanedo, "A review of data fusion techniques," The Scientific World Journal, vol. 2013, 2013.

[7] C. Li, "Multi-lane detection based on rmfp for autonomous vehicle navigation in urban environments," Revista de la Facultad de Ingenier´ıa, vol. 31, no. 5, 2016.

[8] M. Bouain, V. Viswanathan, R. B. Atitallah, and J.-L. Dekeyser, "Communication-centric design for fmc based i/o system," in Reconfigurable and Communication-Centric Systems-on-Chip (ReCoSoC), 2014 9th International Symposium on. IEEE, 2014, pp. 1–8.

[9] Q. Baig, A. O., V. Trung-Dung, and F. Thierry, "Fusion Between Laser and Stereo Vision Data For Moving Objects Tracking In Intersection Like Scenario," in IV'2011 - IEEE Intelligent Vehicles Symposium, Baden-Baden, Germany, Jun. 2011, pp. 362–367.

[10] Bouain M., Berdjag D., Fakhfakh N. and Ben Atitallah R. (2017). An Extrinsic Sensor Calibration Framework for Sensor-fusion based Autonomous Vehicle Perception.In Proceedings of the 14th International Conference on Informatics in Control, Automation and Robotics - Volume 1: ICINCO, ISBN 978-989-758-263-9, pages 505-512. DOI: 10.5220/0006438105050512.

[11] K. M. Ali, R. B. Atitallah, N. Fakhfakh, and J.-L. Dekeyser, "Exploring hls optimizations for efficient stereo matching hardware implementation," in International Symposium on Applied Reconfigurable Computing. Springer, 2017, pp. 168–176.

[12] Chakraborty, S., Lukasiewycz, M., Buckl, C., Fahmy, S., Chang, N., Park, S., ... & Adlkofer, H. (2012, March). Embedded systems and software challenges in electric vehicles. In Proceedings of the conference on design, automation and test in Europe (pp. 424-429). EDA Consortium.

Mokhtar Bouain received his Master's degree in Automatic and Industrial Computing from the National Institute of Applied Sciences and Technology (INSAT), Tunisia. He is currently R&D software engineer at NAVYA company and Ph.D. student in LAMIH laboratory, University of Valenciennes, France. His research interests include embedded systems and intelligent transportation systems.

Karim M. A. Ali received his B.Sc. in computer engineering from University of Ain Shams, Cairo, Egypt and his M.Sc. in embedded systems design from ALaRI, University of Lugano, Switzerland. He is currently a PhD student in LAMIH laboratory, University of Valenciennes, France. His research interests are oriented toward reconfigurable platforms (FPGA), video processing applications and high-level synthesis tools.

Denis Berdjag has obtained the engineering degree from the University of Sétif (Algeria) in 1999, the Master's degree and the Ph.D. degree in automatic control, respectively, in 2003 and 2007 from the University of Science and Technology of Lille (France). He is currently an associate professor of automatic control at Valenciennes University (France) in the LAMIH CNRS UMR 8201 (Laboratoire d'Automatique, de Mécanique et d'Informatique industrielles et Humaines) research group. His research addresses fault detection, isolation and diagnosis of systems (nonlinear, hybrid or human machine systems) and multisensor data fusion.

Nizar Fakhfakh received his Master's degree in Computer Science, Engineering Information and Systems from the National Institute of Applied Sciences, Rouen. In June 2011, he received his PhD degree on Automatic, Information Engineering, Image and signal processing in the French national institute for Transport and Safety Research (INRETS-French) and the Swiss Federal Institute of Technology in Lausanne (EPFL-Switzerland). He is currently Project Manager at NAVYA company. His research interests include: computer vision, autonomous driving, image processing and stereo-vision.

Rabie Ben Atitallah received his PhD degree in Computer Science from the University of Lille 1 in March 2008. In September 2009, Rabie Ben Atitallah obtained a full-time position as an Associate Professor in Computer Science at the UVHC and a member of LAMIH laboratory (UMR CNRS/UVHC 8201) within the DIM (Decision, Interaction, and Mobility) team. In December 2014, he received the Habilitation degree in Computer Science from the UVHC. In addition, he had the status of Associated Researcher at INRIA Lille Nord Europe within the DaRT and DreamPal team-projects until the end of 2016. Currently, Rabie Ben Atitallah is the scientific animator of the theme OPTIMOB (stands for Optimization and Mobility). Rabie Ben Atitallah was the task manager of INRIA Lille in the frame of the ANR OpenPeople (2009 - 2013). Also, he was managing several industrial collaborations (more than 10 projects) with Airbus Group Innovation, Airbus Helicopters, Navya, Nolam Embedded Systems, etc. He co-authored more than 80 publications (patents, journals, and international conferences). Due to his significant scientific contributions, he holds the scientific award of excellence and he is nominated as a member of the National University Committee (CNU) in computer science. His research domain covers embedded system design for intelligent transportation, reconfigurable computing, low power-aware design, high level synthesis tools and design space exploration.