Object Detection Based on Binocular Vision with Convolutional Neural Network

Zekun Luo
Tongji University
School of Electronics and Information Engineering, Shanghai, P. R. China
1631514@tongji.edu.cn

Xia Wu
Tongji University
School of Electronics and Information Engineering, Shanghai, P. R. China
wuxia@tongji.edu.cn

Qingquan Zou
SAIC Motor Corporation Limited
Research & Advanced Technology Department, Shanghai, P. R. China
Zouqingquan@saicmotor.com

Xiao Xiao
SAIC Motor Corporation Limited
Research & Advanced Technology Department, Shanghai, P. R. China
xiaoxiao@saicmotor.com

ABSTRACT
Autonomous vehicles are widely accepted as one of the most potential technologies in alleviating traffic problems. In most existing autonomous vehicles for object detection and distance measurement, compared with radar or LIDAR which obviously increases the cost, camera combined with Convolutional Neural Network (CNN) has advantage in accuracy and low cost. However, most object detection methods applied on camera cannot perform distance measurement. In this paper, we simultaneously carry out real-time object detection and distance measurement (DDM) in one system by utilizing CNN on a binocular camera. Firstly, a binocular camera is used to acquire disparity maps. Secondly, a set of high-quality region proposals is generated by those disparity maps and the number of region proposals is reduced. Thirdly, CNN is utilized to classify those region proposals and get the bounding box of detected objects. Consequently, those reduced region proposals generated by disparity maps lead to improved computational efficiency. Finally, the object distance is measured by the disparity map and the bounding box. The experiment results show that the proposed method can achieve an accuracy of 87.2% on KITTI dataset and an accuracy of 68% in the real environment for object detection. The average relative error of the distance measurement is 0.85% within 10 meters in real environment. The operation time of the whole DDM system is less than 80 ms.

CCS Concepts
• **Computing methodologies→ Object detection**

Keywords
Object detection; Distance measurement; CNN; Binocular camera;

SPML '18, November 28–30, 2018, Shanghai, China

DOI: https://doi.org/10.1145/3297067.3297081

1. INTRODUCTION
Vehicle surrounding environment perception (object detection and distance measurement) is significant for autonomous vehicles and driver assistant systems. Millimeter wave radar and LIDAR used to occupy pivotal position in environment perception. However, as one of the widely used perception devices, camera performs better than millimeter wave radar and LIDAR due to the lower cost and the ability to acquiring more texture information from the objects regardless of their material and shape. With the development of the deep learning algorithm such as Convolutional Neural Network (CNN), these methods become even more prevalent. CNN has been demonstrated to be significantly effective in tremendous object detection methods [1].

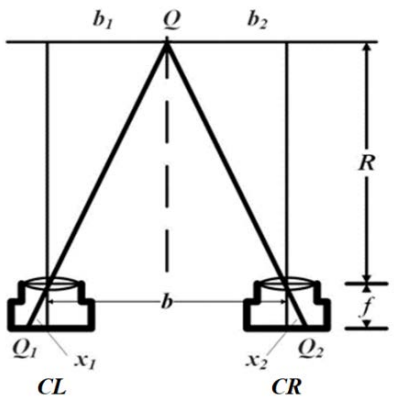


Figure 1. The binocular model.

The object detection methods can be divided into three categories (i.e., RGB image based, RGB-D image based and video based [2]) according to basic information types.

RGB image-based detection uses texture and color information for object detection. Two kinds of detectors are defined in RGB image-based detection methods, namely one-stage (End-to-End) detector and two-stage (region proposals) detector [3]. The two-stage detector typically includes two steps: 1) generate a series of region proposals that potentially contain objects. 2) use a neural network to extract features from these region proposals. Those features are used to detect objects and predict their border position contained in these region proposals. R-CNN (Region with CNN

features) [4], Fast-RCNN [5] and Faster-RCNN [6] are proposed as two-stage detectors and their region proposals are generated according to texture and color information of RGB image, e.g., R-CNN uses "Selective Search" algorithm for region proposal generation, CNN for feature extraction and SVM (Support Vector Machine) for classification. The one-stage detector obtains detection results by directly loading raw image, without generating region proposals. For example, the YOLO (You Only Look Once) [7] method divides the picture into a fixed number of grids and each grid predicts one bounding box and some class probabilities. Generally, the one-stage detector is faster than two-stage detector since the region proposal step is ignored, but prediction precision is reduced.

RGB-D image-based method takes advantage of depth information. For example, [8] uses sliding windows in depth images for object detection. [9] extends the DPM (Deformable Part Model) method by using color images and disparity maps, the detection accuracy on the KITTI dataset has been proved to be improved. 3DOP (3D Object Proposals) uses depth information to detect the distance between the ground and the objects. High accuracy is achieved by minimizing energy functions [10].

However, those object detection methods cannot be directly applied to autonomous vehicles and driver assistant systems. Though most of those methods perform better on PASCAL VOC dataset than real driving environment with different objects varying in different sizes. In addition, those methods pay little attention to distances measurement, while the distances between objects and vehicles are also crucial for autonomous vehicles and driver assistant systems.

Vision-based distance measurement systems receive attention since they are more flexible and have lower cost. Those systems can be divided into two categories, namely monocular vision distance measurement system and binocular vision distance measurement system.

The monocular vision distance measurement system focuses on calculating the mapping rule between the pixel coordinate of image and actual coordinate of real world, and then the distance between the objects and the cameras can be obtained according to pixel range and the mapping rules. A mapping rule which requires a set of parameters easy to be attained is mentioned in [11]. [12] establishes the mapping rules by applying the ray angles. In [13], the distance between two vehicles is estimated based on the size of the license plate. [14] focus on the similarity between the license plate and the image and runs a measurement application on Android operating system.

However, due to the shaking of vehicle, the mapping rule between the image pixels coordinate system and the real-world coordinate system may change, and thus the measurement results of the distance may be inaccurate and insufficient to meet the requirement of the autonomous vehicles and driver assistant applications.

The binocular vision (composed of two cameras equipped on vehicles sides–by-sides) distance measurement system is supposed to have the potential for eliminating the errors caused by vehicle's shaking. According to the principle of triangulation, the distance between the vehicles and objects varies inversely with disparity, and the disparity can be calculated from the disparity maps generated from left and right cameras, e.g., a stereo vision system designed for object detection and distance measurement is presented in [15], and a 3D camera combined with template matching technique is used in [16] to obtain the disparity and measure the distance.

In this paper, we firstly obtain disparity map generated by binocular vision equipment. And then, we extract depth information from the disparity map to generate an RGB-D image. That depth information in RGB-D image can be simultaneously used for the distance measurement and region proposal required by object detector. It is demonstrated that the proposed binocular vision and CNN based DDM system can achieve high object detection accuracy on KITTI dataset, acceptable object detection accuracy and high distance measurement accuracy in real driving environment. The computational delay is within 80 ms. The proposed DDM system provides a low-cost equipment scheme for autonomous vehicle and driver assistant system, and makes it possible for them to obtain high accuracy and low delay environment perception (object detection and distance measurement) with only a binocular camera.

The paper is organized as follows. Section 2 gives a detail description of the principles and implementation of the proposed system. Section 3 describes the evaluation of two functions (object detection and distance measurement) implemented in the system and the analysis the results. Finally, section 4 concludes this paper.

2. SYSTEM IMPLEMENTATION

The implementation process of DDM system can be separated into four steps: (1) Depth information acquisition: A binocular camera is used to acquire disparity maps from the surrounding environment, and then depth information of pixels is extracted from those disparity maps; (2) Region proposals: A set of high-quality region proposals is generated from those disparity maps. In this way, the number of region proposals is greatly reduced compared with those directly generated from the background images, and this can dramatically increase the detection speed; (3) CNN detector: CNN is used to detect the objects and obtain their bounding boxes; (4) Distance measurement: Measuring the distance of object based on depth information (acquired in (2)) and bounding boxes (acquired in (3)).

2.1 Depth Information Acquisition

The depth information (the distance between a pixel captured by the camera and its real position) plays a key role in region proposals and distance measurement [17], and depth information is always contained in disparity (coordinate differences between the corresponding points) map. The corresponding points are referred to the same object appeared on two pictures generated by two cameras located in different positions, the two cameras with the same performance are exactly composed a binocular stereo vision system. There are many algorithms published to calculate disparity map. Hereby, the Block Matching (BM) stereo correspondence algorithm is utilized.

Based on the disparity map, the depth information of each pixel can be calculated as follows: as is sketched in Figure 1, suppose two cameras CL and CR have exactly the same optical properties, e.g., the same focal length f. And the distance between central points of the two cameras is b. Q is part of an object within the coverage of the two cameras, it appears as two pixels Q_1 and Q_2 on camera CL and CR. The depth information R is the radial distance between the central point of two cameras and the object Q. According to algorithm from [18], we can get:

$$\frac{b_1}{R} = \frac{-x_1}{f} \tag{1}$$

$$\frac{b_2}{R} = \frac{x_2}{f} \tag{2}$$

Depth R can be calculated according to (1) and (2):

$$R = \frac{bf}{x_1 - x_2} \tag{3}$$

$x_1 - x_2$ denotes the disparity between the corresponding points Q_1 and Q_2.

2.2 Region Proposals

Due to the fact that pixels of the same object have continuous depth, and saltation always appears on the edges of different objects. By performing edge detection and counter detector algorithm on depth maps, we can find out contours of objects without being affected by the color and texture. Those circumscribed rectangles of the contours can then be selected as region proposals.

In the case of autonomous driving, the depth of the road pixels changes continuously with no explicit contour, so the whole area can be treated as background and discarded. In addition, some of the region proposals can also be discarded according to α (the ratio between pixel size and disparity), i.e., by taking α as a measurement criterion, those region proposals with α out of range of a certain threshold will be discarded. According to the imaging principle, α of each region proposal can be calculated as:

$$\frac{Size_w}{R} = \frac{Size_c}{f} \tag{4}$$

$$\alpha = \frac{Size_c}{x_1 - x_2} = \frac{Size_w}{b} \tag{5}$$

where $Size_w$ and $Size_c$ are physical size and pixel size of the region proposal, respectively.

In this paper, the threshold of α is resulted from its statistical character. Finally, we choose [0.8, 3.5] as threshold. If a region proposal has α within [0.8, 3.5], it can be used for detection in next step, otherwise, it should be discarded. In this way, we only keep a small number of region proposals (about 10). Due to the fact that less region proposal means less detection time, the computational speed of the proposed system can be significantly improved.

2.3 CNN Detector

The region proposals generated in the second step will be loaded into a CNN detector for object detection. Similar to YOLOv2, the structure of CNN used in this paper also has 16 convolutional layers and two fully connected layers. As is shown in Table 1, the CNN uses ReLU (Rectified Linear Unit) as activation function, and BN (batch normalization) in each convolution layer for data normalization. The output consists of the network consist of 7 predictions, 2 for object classes predicting, 4 for bounding box predicting, and one for confidence (IOU Intersection over Union) predicting. The training process of CNN will be elaborated as follows.

Firstly, pre-training the network on the ImageNet64x64 dataset. Since the number of training samples is limited and the amount of network parameters is enormous, the CNN detector always tends to be under-fitting. To solve this problem, we pre-train convolutional layers of the network on the ImageNet64x64 dataset (ImageNet64x64 dataset contains 1281167 training images and 50000 testing images with the size of 64x64 pixels, and those images can only be used for object classification) [19]. Two fully connected layers are re-connected to output 1000 class probabilities and the loss function is defined as cross-entropy loss.

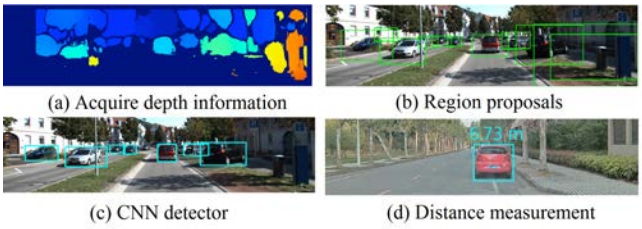
Table 1. CNN Network Structure

Type	Filters	Size/stride	Output
Convolutional	16	3×3	128×64
Max pool	\	2×2/2	64×32
Convolutional	32	3×3	64×32
Max pool	\	2×2/2	32×16
Convolutional	16	1×1	32×16
Convolutional	128	3×3	32×16
Convolutional	16	1×1	32×16
Convolutional	128	3×3	32×16
Max pool	\	3×3	16×8
Convolutional	32	1×1	16×8
Convolutional	256	3×3	16×8
Convolutional	32	1×1	16×8
Convolutional	256	3×3	16×8
Max pool	\	2×2/2	8×4
Convolutional	64	1×1	8×4
Convolutional	512	3×3	8×4
Convolutional	64	1×1	8×4
Convolutional	512	3×3	8×4
Convolutional	128	1×1	8×4
Convolutional	32	1×1	8×4
Fully Connected	\	\	256
Fully Connected	\	\	7

Secondly, re-training the network on the KITTI dataset. The parameters trained in the first step are treated as initial value of the convolutional layer parameters, and the fully connected layers are re-set to output object classes, bounding box and confidence. The whole network is trained with KITTI dataset. KITTI dataset images contains 7481 images with labels [20]. The first 6000 training images are selected for training, and the remaining are reserved for testing. Similar to the loss function of YOLO, we defined the loss function of the detection network as a multi-part loss:

(a) Left camera image (b) right camera image

Figure 2. The camera images.



(a) Acquire depth information (b) Region proposals

(c) CNN detector (d) Distance measurement

Figure 3. The process of the proposed DDM system.

$$loss = -\sum_{i=1}^{n} p \log p'(x) + \lambda \sum_{i=1}^{n} ((x - x')^2 + (y - y')^2) + \lambda \sum_{i=1}^{n} ((\sqrt{w} - \sqrt{w'})^2 + (\sqrt{h} - \sqrt{h'})^2) + \sum_{i=1}^{n} (iou - iou')^2 \quad (6)$$

in which, (x, y) is the center of the object in the pixel coordinate system, w and h are the width and height of the object. The p is the actual class, and iou is the actual IOU value. x', y', w', h', p' and iou' are the corresponding output of x, y, w, h, p and iou, respectively. The coefficient λ (λ =2) is used to control the loss weight of each part. To avoid overfitting, Dropout (with a probability of 0.5) is applied before the first fully connected layer to prevent co-adaptation between the layers [21]. In addition, data enhancement was carried out by flipping and zooming the images randomly.

2.4 Distance Measurement

Based on bounding boxes obtained in section 2.3, we can find the corresponding pixels contained in those boxes in disparity map, and calculate their depth. The distance between the object and the camera can be calculated as the average depth value of all pixels belonging to the object.

The whole system implementation can be intuitively observed in Figure 2 and Figure 3. Figure 2 is the original images captured by the left and right cameras. Figure 4 visualizes the four steps: In Figure 3 (a), we obtain depth map according the disparity map generated by Figure 2(a) and Figure 2 (b). These region proposals shown in Figure 3 (b) are generated by the depth map. In Figure 3(c), region proposals are utilized as input data of CNN detector to obtain detection result (category, bounding box and confidence). Finally, the distance between the object and camera is measured and shown in Figure 3 (d).

3. SYSTEM EVALUATION AND RESULTS ANALYSIS

The proposed DDM system can be simultaneously used for object detection and distance measurement, but the two functions (object detection function and distance measurement function) cannot be evaluated in KITTI synchronously, because images contained in KITTI are only labeled with categories, bounding boxes and confidence, and they can only be used to evaluate object detection function. In this paper, we firstly evaluate object detection function on KITTI, and then evaluate object detection and distance measurement function in real environment.

3.1 Evaluate Object Detection Function on KITTI

After training CNN detector with the first 6000 training images, the remained 1481 images contained in the KITTI dataset are loaded to the proposed system to evaluated the effectiveness of its object detection function. Due to the fact that different object detection tasks have different detection difficulties, the objects contained in the dataset are divided into 3 (Easy, Moderate, and Hard) levels. The detection result of an object is marked as true if the overlap between the detected bounding box and ground truth bounding box (IOU) is larger than 70%.

The precision/recall (PR) curve is shown in Figure 4. To get a fair comparison among different object detection methods, we take AP (average precision) and running time as criteria (see in Table 2). It is obvious that our system performs better for the easy task, because the value of AP is up to 87.2% and the running time can almost meet the object detection requirement of autonomous vehicles. In the case of the moderate and hard tasks, detecting accuracy of the proposed DDM system are remained to be improved compared to Faster R-CNN and YOLOv2. Objects within the easy tasks are not occluded by any obstacles and they

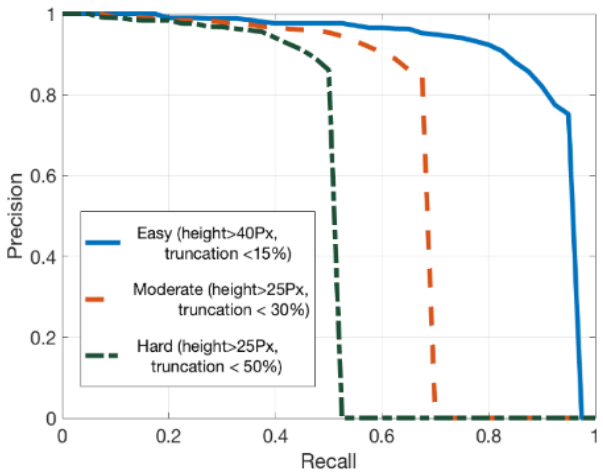


Figure 4. precision/recall (PR) curve

are close to cameras, so accurate region proposals of these objects can be easily obtained from disparity maps. For objects within the moderate and hard tasks, they cannot be completely perceived in disparity maps, so accurate regional recommendations of these object cannot be easily obtained, which resulting in reduced detection performance for moderate and hard tasks.

Table 2. AP (%) of the KITTI dataset

Method	Easy	Moderate	Hard	Running time
YOLOv2	86.40%	69.01%	59.57%	0.03 s
Faster R-CNN	87.90%	79.11%	70.19%	2 s
MV-RGBD-RF [22]	76.49 %	69.92%	57.47 %	4 s
DPM-C8B1[9]	74.95 %	60.99 %	47.16 %	15 s
Vote3D [23]	56.66 %	48.05 %	42.64 %	0.5 s
The proposed DDM system	87.2%	61.60%	52.20%	0.08 s

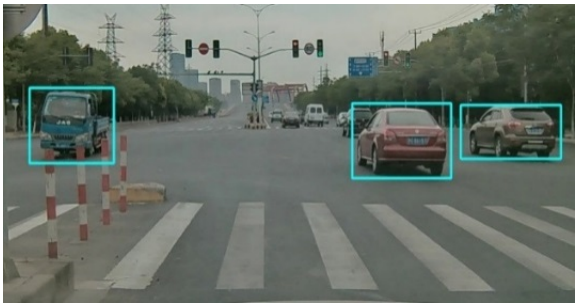


Figure 5. Picture captured in real driving environment

3.2 Evaluate Object Detection and Distance Measurement Function in Real Environment

To evaluate object detection function in real environment, we captured 785 images on Caoan highway in Jiading Dist., Shanghai (see in Figure 5). Objects contained in images were also classified into 3 (Easy, Moderate, and Hard) levels. The proposed DDM system could reach an AP of 68% for the easy object detection task. Then, to evaluate distance measurement function in real environment, we measured the distance (range from 5 meters to 30 meters) of a car driving in campus of Tongji University. We compare distance measurement performance between the proposed DDM system and other methods (D. Deshmukh. et al [12], Sasaki. et al [14] and Nedevschi. et al [15]). The final result is shown in Table 3, the error rate of the proposed DDM system lies below 5% within the range of 30 meters, it monotonically increasing with distance. It also illustrates that the proposed DDM system is appropriate for autonomous vehicles because the objects around the vehicle are more critical for driving safety.

Table 3. Comparison of performance between different methods.

Method	10 m	20 m	30 m
D. Deshmukh. et al [12]	-	2.25%	-
Sasaki. et al [14]	5%	-	-
Nedevschi. et al [15]	1%	-	1.5%
The proposed DDM system	0.85%	2.00%	4.83%

4. CONCLUSIONS

In this paper, we proposed a binocular vision and convolutional neural network based DDM system to carry object detection and distance measurement synchronously. The proposed system was proved to perform well in KITTI dataset and real driving environment, and it has the potential to supply autonomous driving vehicles/driver assistant system with a low-cost equipment scheme to achieve high accuracy, low-delay environment perception (objection detection and distance measurement).

5. ACKNOWLEDGMENTS

This work was supported by the Shanghai Natural Science Foundation (Grant No. 16ZR1446300), the Natural Science Foundation (Grant No. 61401314) and the Fundamental Research Funds for the Central Universities with (Grant No. 1600219316).

6. REFERENCES

[1] Dai, J., Li, Y., He, K. and Sun, J. 2016. R-FCN: Object detection via region-based fully convolutional networks. *Advances in neural information processing systems* (2016), 379–387.

[2] Fragkiadaki, K., Arbeláez, P., Felsen, P. and Malik, J. 2015. Learning to segment moving objects in videos. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (Jun. 2015), 4083–4090.

[3] Lin, T. et al. 2017. Focal Loss for Dense Object Detection. *international conference on computer vision*. (2017), 2999–3007.

[4] Girshick, R., Donahue, J., Darrell, T. and Malik, J. 2014. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. *2014 IEEE Conference on Computer Vision and Pattern Recognition* (Jun. 2014), 580–587.

[5] Girshick, R. 2015. Fast R-CNN. *Proceedings of the IEEE international conference on computer vision* (2015), 1440–1448.

[6] Ren, S., He, K., Girshick, R. and Sun, J. 2017. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 39, 6 (Jun. 2017), 1137–1149.

[7] Redmon, J., Divvala, S., Girshick, R. and Farhadi, A. 2016. You Only Look Once: Unified, Real-Time Object Detection. *2016 IEEE Conference on Computer Vision and Pattern Recognition* (CVPR) (Jun. 2016), 779–788.

[8] Song, S. and Xiao, J. 2014. Sliding Shapes for 3D Object Detection in Depth Images. *European Conference on Computer Vision* (2014), 634–651.

[9] Yebes, J.J., Bergasa, L.M. and García-Garrido, M. 2015. Visual object recognition with 3D-aware features in KITTI urban scenes. *Sensors*. 15, 4 (2015), 9228–9250.

[10] Chen, X., Kundu, K., Zhu, Y., Berneshawi, A.G., Ma, H., Fidler, S. and Urtasun, R. 2015. 3d object proposals for accurate object class detection. *Advances in Neural Information Processing Systems* (2015), 424–432.

[11] Bucher, T. 2000. Measurement of distance and height in images based on easy attainable calibration parameters. *Proceedings of the IEEE Intelligent Vehicles Symposium 2000* (Cat. No.00TH8511) (2000), 314–319.

[12] D. Deshmukh, P. 2012. Analysis of Distance Measurement System of Leading Vehicle. *International Journal of Instrumentation and Control Systems.* 2, (2012), 11–23.

[13] Wang, W. et al. 2015. A rough vehicle distance measurement method using monocular vision and license plate. 2015 IEEE *International Conference on Cyber Technology in Automation, Control, and Intelligent Systems* (CYBER) (Jun. 2015), 426–430.

[14] Sasaki, N. et al. 2017. Development of inter-vehicle distance measurement system using camera-equipped portable device. *2017 17th International Conference on Control, Automation and Systems (ICCAS)* (2017), 994–997.

[15] Nedevschi, S. et al. 2004. High accuracy stereo vision system for far distance obstacle detection. *IEEE Intelligent Vehicles Symposium*, 2004 (Jun. 2004), 292–297.

[16] Zivingy, M. 2013. Object distance measurement by stereo vision. *International Journal of Science and Applied Information Technology (IJSAIT)*. 2, (2013), 05–08.

[17] Brown, M.Z., Burschka, D. and Hager, G.D. 2003. Advances in Computational Stereo. *IEEE Transactions on Pattern Analysis & Machine Intelligence.* 25, 8 (2003), 993–1008.

[18] Mahammed, M.A., Melhum, A.I. and Kochery, F.A. 2013. Object distance measurement by stereo vision. *International Journal of Science and Applied Information Technology (IJSAIT).* 2, 2 (2013), 05–08.

[19] Chrabaszcz, P., Loshchilov, I. and Hutter, F. 2017. *A Downsampled Variant of ImageNet as an Alternative to the CIFAR datasets.* (2017)

[20] Geiger, A., Lenz, P. and Urtasun, R. 2012. Are we ready for autonomous driving? The KITTI vision benchmark suite. *2012 IEEE Conference on Computer Vision and Pattern Recognition* (Jun. 2012), 3354–3361.

[21] Hinton, G.E., Srivastava, N., Krizhevsky, A., Sutskever, I. and Salakhutdinov, R.R. 2012. Improving neural networks by preventing co-adaptation of feature detectors. *Computer Science.* 3, 4 (2012), 212-223.

[22] González, A., Vázquez, D., López, A.M. and Amores, J. 2017. On-Board Object Detection: Multicue, Multimodal, and Multiview Random Forest of Local Experts. *IEEE Transactions on Cybernetics.* 47, 11 (Nov. 2017), 3980–3990.

[23] Wang, D.Z. and Posner, I. 2015. Voting for Voting in Online Point Cloud Object Detection. *Robotics: Science and Systems* (Jul. 2015).