 DOI: 10.3217/978-3-85125-663-5-08

Machine Vision for Embedded Devices: from Synthetic Object Detection to Pyramidal Stereo Matching

Daniele Evangelista[1], Marco Imperoli[2], Emanuele Menegatti[1] and Alberto Pretto[3]

***Abstract*— In this work we present an embedded and all-in-one system for machine vision in industrial settings. This system enhances the capabilities of an industrial robot providing vision and perception, e.g. deep learning based object detection and 3D reconstruction by mean of efficient and highly scalable stereo matching. To this purpose we implemented and tested innovative solutions for object detection based on synthetically trained deep networks and a novel approach for depth estimation that embeds traditional 3D stereo matching within a pyramidal framework in order to reduce the computation time. Both object detection and 3D stereo matching have been efficiently implemented on the embedded device. Results and performance of the implementations are given for publicly available datasets, in particular the T-Less dataset for texture-less object detection, Kitti Stereo and Middlebury Stereo datasets for depth estimation.**

I. INTRODUCTION

State-of-the-art industrial machine vision systems currently works with 3D sensors, sometimes coupled with a color or a gray-level camera. Traditionally, the 3D information has been acquired using *passive* stereo systems, i.e. systems composed by two or more cameras. The depth map is recovered by means of a correspondence problem: matched points projections are triangulated between pairs of sensors. Unfortunately, these systems often fail to provide an accurate 3D reconstruction for large portions of the framed scene, due to the absence of salient visual features. To overcome this limitation, *active* stereo systems have been introduced. Active vision sensors use light emitters that project a specific pattern (*Active Stereo and Structured Light* sensors) or a light with a specific wavelength (*Time-of-Flight* sensors): all these sensors modify in some way the surrounding environment (i.e., they illuminate the scene). In the first cases, the correspondence problem is solved in different ways: by performing a traditional stereo matching algorithm using visual features synthetically created by the light projector for the active stereo sensors; by searching the known pattern in the camera image (so called *pattern decoding*) for the structured light sensors.

In this work we propose an embedded and all-in-one device that integrates both active and passive stereo matching technologies. In particular, two high resolution color cameras provide the system with passive stereo capabilities, and a random pattern projector mounted at the center of the cameras baseline provides active stereo capabilities by means of structured illumination of the scene (see Fig.2 for more details). The proposed sensor also integrates a CPU and a powerful Graphical Processing Unit (GPU) specifically designed to run expansive Machine Learning algorithms (e.g. Deep Learning) and a complete Unix based Operating System. This design enables the possibility to mount the system directly on top of a robotic cell and being connected bidirectionally with the robot system without the need of any external unit. This high level of flexibility makes the system appropriate for bin picking applications where a robotic manipulator needs to be driven by a vision system to detect and accurately manipulate highly cluttered objects.

Fig. 1: The proposed sensor: renders (left column) and its realization with a really functioning prototype (right column).

To be able to perceive and accurately detect objects, vision systems rely on 2D and 3D information at the same time. For this reason we propose an efficient depth estimation method that embeds traditional 3D stereo matching techniques within a pyramidal framework in order to reduce the computation time. Moreover, on the system, we also implemented deep neural network based object detectors that were trained using synthetically generated data. This process drastically decreases the time needed for collecting data, and does not require any human intervention for annotating the data. The aforementioned perception pipeline has been tested on highly challenging task, namely texture-less objects, a very common situation in industrial settings where objects quite often do not offer any, or very poor, texture detail.

This work was supported by the European Commission under 601116-ECHORD++ (FlexSight experiment).

[1] authors are with the Department of Information Engineering (DEI), University of Padova, Italy, (evangelista, emg)@dei.unipd.it

[2] author is with the Department of Computer, Control, and Management Engineering, Sapienza University of Rome, Italy imperoli@diag.uniroma1.it

[3] author is the FlexSight s.r.l. company, Padua, Italy alberto.pretto@flexsight.eu

II. Related Works

A. Texture-less Object Detection

Object detection in images has been approached mainly in two ways: methods based on sliding window as Deformable Part Model from [1]; classification of region proposals produced with region proposal algorithms as the well known Selective Search from [2]. Thanks to the enormous increase in the research on Convolutional Neural Networks (CNNs), methods on region proposals have become prominent. R-CNN from [3] has been the first deep neural network trained for extracting features from region proposals using convolutional networks. This approach has been further improved in Faster R-CNN from [4] where the selective search region proposal algorithm is replaced with a Region Proposal Network (RPN, first time introduced with [5] and [6]) and the complete deep network is trained end-to-end for extracting the proposals and performing classification on the object's bounding box extracted using regression.

In this work we used a object detector called Single Shot Detector, from [7] that improve the quality and speed of the detection w.r.t. Faster R-CNN object detector by simultaneously producing a score for each object category in each predicted box and then classifying them. In this way the deep network is easier to train, faster, and ready to be integrated into other tasks.

B. Stereo Matching

Depth estimation from stereo is one of the most active topics in computer vision of the last 30 years. Given two rectified images, the problem is to find for each pixel in the reference image the corresponding point in the second image. Rectification reduces the correspondences' search along the same scanline. As described in [8], the main steps of stereo algorithms are: matching cost computation, cost aggregation, disparity optimization followed by a disparity refinement step. Methods can be categorized in local [9] [10] [11] [12], global [13] [14] [15] [16] or semiglobal [17] [18], depending on the techniques used to solve each step of the pipeline.

Recent works exploit the framework of PatchMatch Stereo [11] [14]. These methods exploit alternatively a random depth generation procedure and the propagation of depth, resulting in a total runtime cost of $O(W \log L)$, where W is the window size used to compute the matching cost between patches and L the number of searched disparities. The method proposed in [12], instead, strongly relies on superpixels, removing the linear dependency on on both the window size and label space. However, the superpixels's estimation requires a high computational time.

The active stereo problem has been recently addressed by exploiting efficient learning-based solutions [19] [20] [21] [22].

Recent deep learning based methods, among the others [23] [24] [25], provide very accurate results. However, these techniques usually don't generalize well to different contexts and require a fine-tuning of the CNN. Others [26] [27] [28] try to predict depth from a single image, but in practice are limited to very specific scenes.

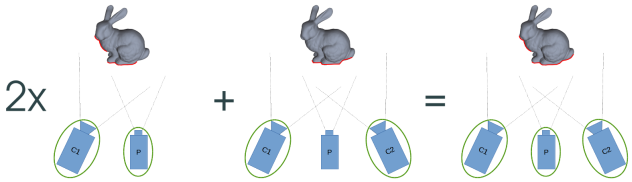


Fig. 2: The proposed system embeds multiple types of stereo vision technologies: 2 active stereo systems (C_1+P and C_2+P) and 1 passive stereo system (C_1+C_2).

III. Perception

The proposed system has been studied for industrial robotics applications where perception capabilities have a key role. Object detection and depth estimation are two of the main important tasks in this field, in the following we present more in detail the prototype we built and our custom solutions for the two applications, namely *texture-less object detection* based on deep transfer learning from synthetic data and *depth estimation* by mean of 3D Stereo matching within a pyramidal framework.

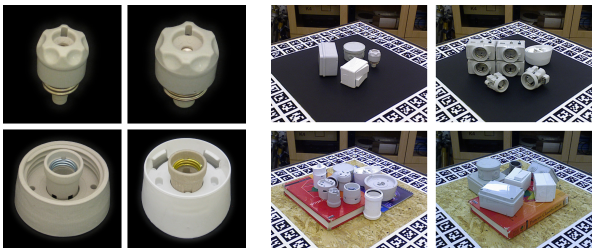
A. The Embedded Device

The proposed system (coded with the name FlexSight C1 and depicted in Fig.1) integrates both active and passive stereo matching technologies. In particular, two high resolution color cameras provide the system with passive stereo capabilities, while a random pattern projector mounted at the center of the cameras baseline provides active stereo capabilities by means of structured illumination of the scene (see Fig.2 for more details). It also integrates a CPU and a powerful Graphical Processing Unit (GPU) specifically designed to run expansive Machine Learning algorithms (e.g. Deep Learning) and a complete Unix based Operating System. This design enables the possibility to mount the system directly on top of a robotic cell and being connected bidirectionally with the robot system without the need of any external unit. This high level of flexibility makes the system appropriate for bin picking applications where a robotic manipulator needs to be driven by a vision system to detect and accurately manipulate highly cluttered objects.

B. Deep Learning Texture-less Object Detection

Data driven methods demonstrated to be very effective in detecting common textured and complex objects [30] [31], on the contrary, that is not the case with texture-less objects, e.g. mechanical parts in industrial bin picking applications. Texture-less objects do not expose so many features that a deep neural network can learn, and most often, having no texture highlights object symmetries and similarities making difficult the generalization of the task, in this way both classification and detection accuracy fall down rapidly.

The key aspect of every data driven task is the nature of the data itself, how the information encoded in the data is



(a) (b)

Fig. 3: (a) Example object classes (b) and test scenes from the *T-Less* dataset [29].

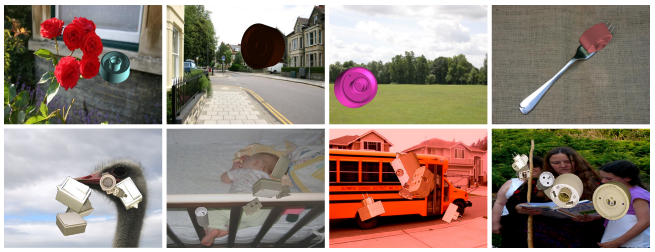


Fig. 4: Some examples from the synthetic generated dataset. Upper raw shows full synthetic example images, lower row shows semi-synthetic example images.

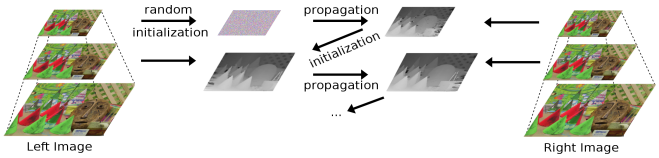


Fig. 5: Hierarchical architecture with propagation from top to bottom.

well exposed and how this can enhance the deep network capability in detect and emphasize highly generalized and heterogeneous features. Data collection is a fundamental aspect within the entire learning process and most of the time this task is done manually. Human intervention is often needed in collecting and then labeling the huge amount of data necessary for feeding the networks with enough information in order to avoid problems such as overfitting the input data. To limit, and somehow overcome, human intervention in data preparation we used synthetic data. Synthetic data is automatically generated by means of projection of the 3D object models onto random and highly generalized backgrounds. This process allows fast and accurate data collection. Without the need of manual intervention the data is generated directly ready to be used for the training of the detection model. Moreover, given the potentially infinite amount of data that can be generated, we are able to create well generalized datasets making the texture-less object detection training process more focused in learning more general features such as object shape, edges, occlusions and symmetries rather than color and appearance.

We exploited multiple CNNs architectures capable of running in inference mode on our embedded system, from accurate and efficient implementation of fully convolutional neural networks with region-based detector [32], to more fast and compact CNNs architectures such as [33] [34] [35] [7]. During training, for all our deep models, the layers responsible for feature extraction have been freezed to generic layers pre-trained on real images, and only the remaining layers are trained with our fully synthetic data. This process is also called *transfer learning* and it demonstrated to be very effective when training large and complex deep convolutional networks with pure synthetic images [36].

The system has been tested with some of the objects presented in the *T-Less Dataset* from [29]. Some examples of objects from T-Less can be seen in Fig.3.

C. Pyramidal Stereo Matching

In local stereo matching, a support window is centered on a pixel of the reference frame. In order to find the correspondence, this support window is displaced in the second image to find the point of lowest dissimilarity. Here is the implicit assumption that the pixels within the support region have a constant disparity. This does not apply to slanted surfaces, which are then reconstructed as compositions of frontal-parallel surfaces. The PatchMatch Stereo algorithm [11] overcomes this problem by estimating a 3D plane at each pixel onto which the support window is projected. As shown in [37], this technique provides very accurate disparities but it is also very slow, i. e. it is not suitable for real-time computing.

Inspired by [38] and [39], we propose to embed the PatchMatch Stereo algorithm [11] in a pyramidal framework (see Fig. 5) in order to reduce the matching time, while sensibly increasing the accuracy of the estimated disparities.

The disparity estimation of the upper levels of the pyramids (lower image resolution) is propagated on the lower levels (higher image resolution), enabling *i)* a considerable speed up of the random search step and *ii)* a reduction of the size of the support window in the lower pyramid levels.

IV. Experiments

A. Object Detection

As already anticipated, we overcome the problem of data acquisition and manual labeling by mean of synthetic data generation. In particular, starting from the 3D CAD model representation of our object, we project it onto random natural images as background, positioning the object in completely random position and orientation in the camera reference frame. Moreover, the objects are rendered using

Training Data	Obj_5	Obj_8	Obj_9	Obj_10	**Average**
Full Synthetic	0.3732	0.288	0.3179	0.2725	0.3129
Semi-Synthetic	0.5283	0.468	0.4956	0.477	**0.49225**

TABLE I: Performance on 4 of the objects' classes in the T-Less *test primesense* data. Results are given in terms of mAP@0.5 (mean Average Precision with 0.5 Intersection Over Union threshold).

random colors and illumination conditions (e.g. light intensity and position). In Fig.4 some example of the synthetic data are given. With this set of data we are able to train very deep networks for object detection, e.g. [32] [33] [34] [35] [7]. We will focus on *Single Shot Detector (SSD)* deep network as it has been the fastest in training time while achieving almost the same accuracy among all the tested networks.

A set of 10000 samples have been generated using the aforementioned procedure with random background extracted from the Microsoft Research Cambridge Object Recognition Image Database[1]. The deep model has been implemented with the TensorFlow Object Detection API[2] and trained on a machine equipped with a Nvidia GTX 1060 GPU Board.

Table I shows some quantitative results obtained using the synthetically generated data for 4 different classes of the dataset. The poor performance of this model reflects how it actually does not generalize well the task. An effective increase of performance has been obtained by training the model with *semi-synthetic* data: real images of real objects (as the ones in Fig.3 (a)) have been used instead of CAD renderings. This approach makes the transfer learning task easier, because real object images actually have more visual features, and make the network easily learn to detect and accurately distinguish among different objects on the test data.

Fig.6 shows some qualitative results obtained with the semi-synthetic approach. In particular, Fig.6 (a, b) show good detections in a dense scene, where the desired object is very similar to some other in the scene. Class similarity still remains a problem for the network, and it can be seen in the detection examples given in Fig.6 (c, d) where the desired object class is often confused with a similar one.

B. Stereo Matching

The proposed algorithm has been tested and evaluated on two popular benchmark data: Middlebury Stereo 2014 [40] and Kitti Stereo 2012 [41]. The evaluation has been performed on a i7-5700HQ CPU, 2.70GHz, and then implemented also on the embedded device, which is equipped with a ARM Cortex-A57 (quad-core), 1.73GHz, with an increase of runtime of 20%. The results in Tab. II refer to downscaled ($0.5Mpx$) version of the Middlebury training images. The evaluation on the Kitti dataset (see Tab. III), instead, has been performed using the original resolution (1242x375px) colored images. The state-of-the-art deep learning based methods [23] and [25] have been tested on a Nvidia GTX 1060 GPU using the pretrained models on Kitti-Stereo 2012 training set. As reported in Tab. III, [23] and [25] show superior performance when using fine-tuned models on the specific benchmark[3]. However, the degraded results in Tab. II show the difficulty of these techniques to generalize to completely different scenarios. The proposed method, instead, is able to generalize (the same set of parameters has been used in both evaluations), providing comparable results, in terms of bad pixel rate[4], in both benchmarks and outperforming other state-of-the-art algorithms. More specifically, the proposed method, compared to [11], is able to decrease the computational time up to 60%, while the accuracy of the disparities is improved up to 20%, demonstrating the effectiveness of the pyramidal framework.

Algorithm	bad 0.5	bad 1.0	bad 2.0	bad 4.0	Runtime
PSMNet [23]	89.4%	76.5%	57.1%	35.9%	0.7 s (GPU)
MC-CNN [25]	67.9%	40.2%	26.7%	13.9%	101 s (GPU)
ELAS [42]	67.3%	38.6%	25.9%	13.5%	0.3 s
[11]	47.2%	27.5%	15.8%	6.2%	22.3 s
Pyramidal Matching	46.3%	25.8%	12.9%	5.5%	8.7 s

TABLE II: Average performance on Middlebury training dataset [40].

Algorithm	bad 2.0	bad 3.0	Runtime
PSMNet [23]	2.4%	1.5%	0.4 s (GPU)
MC-CNN [25]	3.9%	2.4%	67 s (GPU)
ELAS [42]	10.8%	8.2%	0.2 s
[11]	8.1%	5.3%	13.1 s
Pyramidal Matching	7.4%	4.5%	5.6 s

TABLE III: Average performance on Kitti-Stereo 2012 dataset [41].

V. Conclusions and Future Works

In this work we presented an embedded system developed for industrial robots, where texture-less object detection and stereo matching are two important tasks. The system is meant for working without the need of any external computational unit, moreover it embeds vision techniques that minimize, and to some extent cancel, the human intervention in the loop. In particular, synthetic data demonstrated to have a huge potential in limiting the manual intervention in data acquisition and data annotation. The proposed synthetic pipeline is tested on a very challenging dataset, which contains low variability among the different classes of objects, most of them reflect high similarity and symmetries making the learning process difficult to generalize to unknown test data. Synthetic data may overcome the problem of generality by introducing high variability, both in terms of visual and geometric features. Further investigations must be done in order to increase the performance of pure synthetically generated data, so that to drive deep models to learn not mainly relying on the visual features, e.g. object textures, but focusing the learning process on more geometric features such as object shape and edges, which are independent from

[1] https://www.microsoft.com/en-us/download/details.aspx?id=52644

[2] https://github.com/tensorflow/models/tree/master/research/object_detection

[3] Results for [23] and [25] in Tab. III are taken from the Kitti stereo evaluation website (http://www.cvlibs.net/datasets/kitti/eval_stereo_flow.php?benchmark=stereo).

[4] The "bad N" metric, used in Tab. II and III, refers to the percentage of pixels whose disparity error is grater than N.

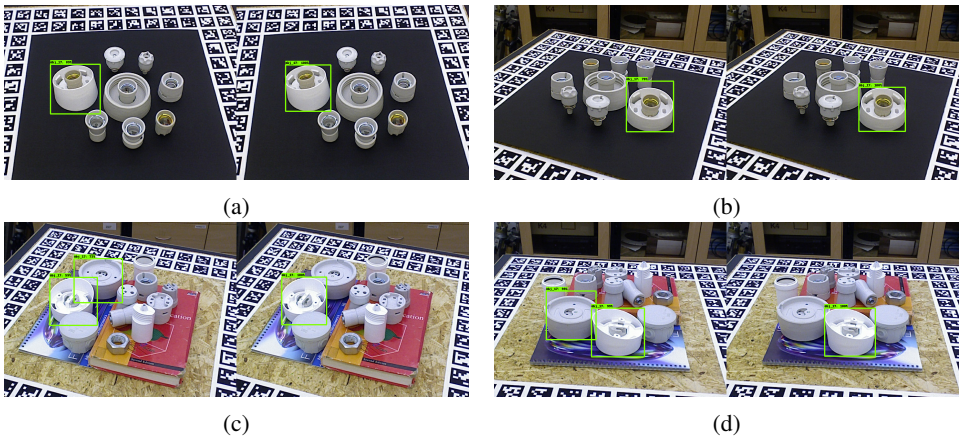


Fig. 6: Qualitative results of the texture-less object detection system. Each couple of images report the network detection *(left)* and the ground truth detection *(right)*. (a) and (b) represents positive examples of detection, with high score in terms of accuracy in the detection and *IoU (Intersection over Union)* between network detection and ground truth. On the contrary (c) and (d) depict some examples of failures.

the visual aspect of an object and more suitable for texture-less object detection tasks.

A possible solution is to include the depth information in the learning process. In this direction, we proposed a pyramidal stereo matching framework that provides accurate depth estimation that could be used in the detection pipeline. Although we showed an improvement of runtime performance compared to [11], the proposed stereo matching algorithm is still not suitable for real-time computing. In future work, we will exploit the massive parallelization capabilities of modern architectures by providing a GPU implementation that might lead to real-time performance.

REFERENCES

[1] P. F. Felzenszwalb, R. B. Girshick, and D. McAllester, "Discriminatively trained deformable part models, release 4," http://people.cs.uchicago.edu/ pff/latent-release4/.

[2] J. R. R. Uijlings, K. E. A. van de Sande, T. Gevers, and A. W. M. Smeulders, "Selective search for object recognition," *International Journal of Computer Vision*, vol. 104, no. 2, pp. 154–171, 2013.

[3] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, June 2014, pp. 580–587.

[4] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in Neural Information Processing Systems 28*, C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, Eds. Curran Associates, Inc., 2015, pp. 91–99.

[5] D. Erhan, C. Szegedy, A. Toshev, and D. Anguelov, "Scalable object detection using deep neural networks," *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2155–2162, 2014.

[6] C. Szegedy, S. E. Reed, D. Erhan, and D. Anguelov, "Scalable, high-quality object detection," *CoRR*, vol. abs/1412.1441, 2014. [Online]. Available: http://arxiv.org/abs/1412.1441

[7] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. E. Reed, C. Fu, and A. C. Berg, "SSD: single shot multibox detector," *CoRR*, vol. abs/1512.02325, 2015. [Online]. Available: http://arxiv.org/abs/1512.02325

[8] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *International Journal of Computer Vision*, vol. 47, no. 1, pp. 7–42, Apr 2002. [Online]. Available: https://doi.org/10.1023/A:1014573219977

[9] K.-J. Yoon and I.-S. Kweon, "Locally adaptive support-weight approach for visual correspondence search," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 2, June 2005, pp. 924–931 vol. 2.

[10] C. Rhemann, A. Hosni, M. Bleyer, C. Rother, and M. Gelautz, "Fast cost-volume filtering for visual correspondence and beyond," in *IEEE Computer Vision and Pattern Recognition*, 2011.

[11] C. R. Michael Bleyer and C. Rother, "Patchmatch stereo - stereo matching with slanted support windows," in *Proceedings of the British Machine Vision Conference*. BMVA Press, 2011, pp. 14.1–14.11, http://dx.doi.org/10.5244/C.25.14.

[12] J. Lu, H. Yang, D. Min, and M. N. Do, "Patch match filter: Efficient edge-aware filtering meets randomized search for fast correspondence field estimation," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2013.

[13] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient belief propagation for early vision," *International Journal of Computer Vision*, vol. 70, no. 1, pp. 41–54, Oct 2006. [Online]. Available: https://doi.org/10.1007/s11263-006-7899-4

[14] F. Besse, C. Rother, A. Fitzgibbon, and J. Kautz, "Pmbp: Patchmatch belief propagation for correspondence field estimation," *International Journal of Computer Vision*, vol. 110, 10 2013.

[15] P. Krähenbühl and V. Koltun, "Efficient inference in fully connected crfs with gaussian edge potentials," in *Advances in Neural Information Processing Systems 24*, J. Shawe-Taylor, R. S. Zemel, P. L. Bartlett, F. Pereira, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2011, pp. 109–117.

[16] Y. Li, D. Min, M. Brown, M. Do, and J. Lu, "Spm-bp: Sped-up patchmatch belief propagation for continuous mrfs," in *2015 International Conference on Computer Vision, ICCV 2015*. United States: Institute of Electrical and Electronics Engineers Inc., 2 2015, pp. 4006–4014.

[17] H. Hirschmuller, "Stereo processing by semiglobal matching and mutual information," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 2, pp. 328–341, Feb 2008.

[18] M. Bleyer and M. Gelautz, "Simple but effective tree structures for dynamic programming-based stereo matching," in *International Conference on Computer Vision Theory and Applications (VISAPP)*, 2008.

[19] Y. Zhang, S. Khamis, C. Rhemann, J. Valentin, A. Kowdle, V. Tankovich, S. Izadi, T. Funkhouser, and S. Fanello, "Activestereonet: End-to-end self-supervised learning for active stereo systems," 09 2018.

[20] S. R. Fanello, J. Valentin, C. Rhemann, A. Kowdle, V. Tankovich, P. Davidson, and S. Izadi, "UltraStereo: Efficient learning-based matching for active stereo systems," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, jul 2017.

[21] S. R. Fanello, J. Valentin, A. Kowdle, C. Rhemann, V. Tankovich, C. Ciliberto, P. Davidson, and S. Izadi, "Low compute and fully

parallel computer vision with hashmatch," in *2017 IEEE International Conference on Computer Vision (ICCV)*, Oct 2017, pp. 3894–3903.

[22] S. R. Fanello, C. Rhemann, V. Tankovich, A. Kowdle, S. O. Escolano, D. Kim, and S. Izadi, "Hyperdepth: Learning depth from structured light without matching," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016, pp. 5441–5450.

[23] J.-R. Chang and Y.-S. Chen, "Pyramid stereo matching network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5410–5418.

[24] Z. Liang, Y. Feng, Y. Guo, H. Liu, W. Chen, L. Qiao, L. Zhou, and J. Zhang, "Learning for disparity estimation through feature constancy," 2018.

[25] J. Zbontar and Y. LeCun, "Stereo matching by training a convolutional neural network to compare image patches," *Journal of Machine Learning Research*, vol. 17, pp. 1–32, 2016.

[26] D. Eigen, C. Puhrsch, and R. Fergus, "Depth map prediction from a single image using a multi-scale deep network," in *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, ser. NIPS'14. Cambridge, MA, USA: MIT Press, 2014, pp. 2366–2374.

[27] A. Saxena, M. Sun, and A. Y. Ng, "Make3d: Learning 3d scene structure from a single still image," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 5, pp. 824–840, May 2009.

[28] C. Godard, O. Mac Aodha, and G. J. Brostow, "Unsupervised monocular depth estimation with left-right consistency," in *CVPR*, 2017.

[29] T. Hodaň, P. Haluza, Š. Obdržálek, J. Matas, M. Lourakis, and X. Zabulis, "T-LESS: An RGB-D dataset for 6D pose estimation of texture-less objects," *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2017.

[30] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision (IJCV)*, vol. 115, no. 3, pp. 211–252, 2015.

[31] T. Lin, M. Maire, S. J. Belongie, L. D. Bourdev, R. B. Girshick, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: common objects in context," *CoRR*, vol. abs/1405.0312, 2014. [Online]. Available: http://arxiv.org/abs/1405.0312

[32] J. Dai, Y. Li, K. He, and J. Sun, "R-FCN: object detection via region-based fully convolutional networks," *CoRR*, vol. abs/1605.06409, 2016. [Online]. Available: http://arxiv.org/abs/1605.06409

[33] J. Redmon, S. K. Divvala, R. B. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," *CoRR*, vol. abs/1506.02640, 2015. [Online]. Available: http://arxiv.org/abs/1506.02640

[34] J. Redmon and A. Farhadi, "YOLO9000: better, faster, stronger," *CoRR*, vol. abs/1612.08242, 2016. [Online]. Available: http://arxiv.org/abs/1612.08242

[35] ——, "Yolov3: An incremental improvement," *CoRR*, vol. abs/1804.02767, 2018. [Online]. Available: http://arxiv.org/abs/1804.02767

[36] S. Hinterstoisser, V. Lepetit, P. Wohlhart, and K. Konolige, "On pre-trained image features and synthetic images for deep learning," *CoRR*, vol. abs/1710.10710, 2017. [Online]. Available: http://arxiv.org/abs/1710.10710

[37] B. Tippetts, D. J. Lee, K. Lillywhite, and J. Archibald, "Review of stereo vision algorithms and their suitability for resource-limited systems," *Journal of Real-Time Image Processing*, vol. 11, no. 1, pp. 5–25, Jan 2016.

[38] T. Xu, P. Cockshott, and S. Oehler, "Acceleration of stereo-matching on multi-core cpu and gpu," in *2014 IEEE Intl Conf on High Performance Computing and Communications, 2014 IEEE 6th Intl Symp on Cyberspace Safety and Security, 2014 IEEE 11th Intl Conf on Embedded Software and Syst (HPCC,CSS,ICESS)*, Aug 2014, pp. 108–115.

[39] Y. Hu, R. Song, and Y. Li, "Efficient coarse-to-fine patch match for large displacement optical flow," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016, pp. 5704–5712.

[40] D. Scharstein, H. Hirschmller, Y. Kitajima, G. Krathwohl, N. Nesic, X. Wang, and P. Westling, "High-resolution stereo datasets with subpixel-accurate ground truth." in *GCPR*, ser. Lecture Notes in Computer Science, X. Jiang, J. Hornegger, and R. Koch, Eds., vol. 8753. Springer, 2014, pp. 31–42.

[41] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[42] A. Geiger, M. Roser, and R. Urtasun, "Efficient large-scale stereo matching," in *Asian Conference on Computer Vision*, Queenstown, New Zealand, November 2010.