Semantic Scene Segmentation for Indoor Robot Navigation via Deep Learning

Yao Yeboah
Guangdong University of Technology
School of Automation
Guangzhou 510006, Guangdong
P.R.China
yeboahyao@gdut.edu.cn

Cai Yanguang
Guangdong University of Technology
School of Automation
Guangzhou 510006, Guangdong
P.R.China
caiyg99@163.com

Wei Wu
South China University of Technology
School of Automation Science and Engineering
Guangzhou 510630, Guangdong
P.R.China
weiwu@scut.edu.cn

Zeyad Farisi
South China University of Technology
School of Automation Science and Engineering
Guangzhou 510630, Guangdong
P.R.China
z_doo@hotmail.com

ABSTRACT

This paper presents a vision-based framework for indoor robot navigation which exploits semantic segmentation and deep learning towards accurate and efficient indoor scene mapping and collision-free navigation for hardware constrained robotics. Firstly, a scheme for accurate and efficient path extraction using deep convolutional neural networks (DCNNs) and transfer learning for semantic pixel-wise segmentation is put forward. Secondly, multiple DCNN architectures and semantic segmentation techniques are explored to highlight the challenges associated with implementation as well as the trade-offs between accuracy and efficiency associated with the state-of-the-art. Finally, the achieved models are deployed and experimentally validated. Experimental results highlight promising potential with good segmentation accuracies and real-time feasibility. Results further highlight significant accuracy-efficiency trade-offs which are strongly driven by model decoder sub-network design.

CCS Concepts

• **Computing methodologies→Vision for robotics** • **Computer systems organization→Robotic autonomy**.

Keywords

Semantic Segmentation; Convolutional Neural Network (CNN); Transfer Learning; Indoor Robot Navigation

ICRCA'18, August 11–13, 2018, Chengdu, China.

ACM ISBN 978-1-4503-6530-7/18/08…$15.00
https://doi.org/10.1145/3265639.3265671

1. INTRODUCTION

The ability to navigate autonomously is a crucial requirement in robotics, allowing robots to interface with the real world in an intuitive manner for satisfying design goals in several applications including security, robot assisted living (RAL), human computer interaction (HCI), but to mention a few.

By exploiting Time of Flight (ToF) techniques, some of the earliest approaches achieved a means of mapping complex indoor environments for the task of navigation [1]. While such techniques have achieved good results [2-4], the existing shortcomings of ToF sensors including multi-path reflections and depth holes remain challenging. Stereo vision techniques which offer an alternative approach for depth estimation and scene mapping have also been researched [5, 6]. In stereo vision, a 3D scene is reconstructed by relying upon camera pairs in capturing different perspectives of the same feature points within the scene. While robust to the fore-mentioned challenges of ToF approaches, they over-rely on feature consistencies across image pairs, a condition that cannot always be satisfied in indoor environments due to textural inhomogeneity, illumination variation, but to mention a few. Perhaps the greatest breakthrough in addressing indoor navigation for robots has been achieved by the Simultaneous Localization and Mapping (SLAM) category of approaches. The effectiveness of SLAM as an indoor navigation solution for robotics has been highlighted by several state-of-the-art [7-9]. Despite its superior performance, SLAM remains highly dependent on temporal continuity and does not offer a means of constructing learnable representations of the environment. Furthermore, it is challenging to incorporate the carefully crafted SLAM components into end-to-end trainable frameworks.

For robot platforms with significantly constrained power and computational budgets, there is the need for solutions that are efficient in terms of hardware and software. For such platforms, vision-based navigation remains a promising solution. Vision-based navigation has been researched for more than a decade and addresses navigation by relying solely on visual input from the scene, often from a single camera; significantly cutting down hardware complexity. Some of the earliest works in this area

adopted traditional image processing and machine learning techniques in achieving scene understanding [10, 11]. In such early works, manual feature selection and engineering have been relied upon for extracting features of interest such as lanes and obstacles. Due to the drawbacks of manual feature selection and extraction, such approaches fail to effectively generalize and sustain performance for natural unconstrained environments.

Since the conception of the AlexNet [12], convolutional neural networks (CNNs) and deep convolutional neural networks (DCNNs) have become the de-facto standard for addressing the vast majority of computer vision tasks including classification [12], detection [13, 14], localization [15], and understanding [16]. Semantic segmentation has become a growing research focus in addressing problems such as biomedical imaging and autonomous navigation due to the capability to achieve pixel level understanding using only visual input in the form of images or videos. Semantic segmentation offers a promising and efficient approach towards addressing the indoor navigation problem for robotics.

In this paper, multiple CNN architectures and approaches for semantic segmentation are explored and verified for the purpose of indoor navigation. The achieved models are implemented and verified through simulation and experimental analyses using an experimental robot platform. The achieved results show that the models achieve high accuracies in path extraction for supporting the robot navigation task. Furthermore, good generalization ability is demonstrated by the models in natural unconstrained indoor environments.

The remainder of the paper is organized as follows. In Section 2 the research background and related literature are covered. The Section 3 presents the CNN models along with model training and post-processing techniques. Experimental results, analyses and discussions are covered in the Section 4. The paper concludes in the Section 5 with an outlook on future work.

2. RESEARCH BACKGROUND

Semantic pixel-wise segmentation is a fast-growing research field with a broad application spectrum including driver assistance technology [17], biomedical imaging [18], robotic manipulation [19], but to mention a few. This strong research interest is motivated by the capability of semantic segmentation algorithms to build useful and meaningful cues from visual data in the form of images or videos. While not an entirely new research area, the field has seen significant leaps in the achieved performance since the emergence of deep learning. Early approaches strongly relied on manual feature crafting and engineering which drew upon conventional machine learning techniques such as Support Vector Machines (SVM) [20], Boosting [21] and Random Forests [22]. The performances of these approaches have been greatly surpassed by deep learning-based alternatives which continue to dominate the state of the art [23-27]. The adoption of deep learning for the task of semantic segmentation has been fueled by the ground-breaking achievements that have been made by CNNs and DCNNs in other areas of visual information processing such as classification [12], detection [13] and localization [15].

A review of the state-of-the-art CNN and DCNN-based semantic segmentation approaches shows that irrespective of the algorithm exploited, most approaches essentially adopt an *encoder-decoder* structure. In the encoder network, high resolution input images are transformed into low resolution activation maps via consecutive convolutional and pooling operations which refine and down-sample feature maps. A majority of the state-of-the-art have based their encoder networks on the VGG-16 architecture [28] which is made up of 13 convolutional and 3 fully connected layers. Conversely, the decoder network is designed to recover a multi-dimensional representation of the feature maps produced by the encoder network for pixel-wise classification. The decoder network design has been the core variation amongst the various state-of-the-art models. The design of the decoder network has further implications for the overall performance of the model in terms of accuracy, efficiency and end-to-end trainability. It is worth highlighting that the state-of-the-art continue to trade-off between model efficiency (memory usage and prediction speed), prediction accuracy and training time [23]. The U-Net architecture [25] which is famous in the medical imaging community has been shown to heavily trade-off efficiency and training time for accuracy gains by propagating entire feature maps from the encoder network into the decoder network. This design allows for good segmentation accuracy to be achieved but significantly increases the total number of model parameters as well as the physical memory required for storing the model at training and test time. In the Deep-Lab architecture [27], *atrous convolutions* (dilated convolutions) are a key feature in the decoder network for successfully up-sampling low resolution activation maps for final pixel-wise classification. The algorithm further adopts a Conditional Random Field (CRF) post-processing scheme in achieving smooth and competitive segmentation results.

Over the years, the Fully Convolutional Network (FCN) [24] has been one of the most widely adopted semantic segmentation algorithms. In the FCN architecture, the decoder network is trained to achieve feature map up-sampling by fusing activation maps within each decoder block with the corresponding encoder output, and using this fused activation map as input to the next decoder block. The algorithm has been shown to achieve good segmentation results with a comparatively smaller decoder size. For robot platforms with low computational and hardware budgets, efficiency is a core requirement to be considered in the design and adoption of deep learning-based semantic scene segmentation algorithms. This should however be achieved in a manner that does not significantly trade-off scene segmentation accuracy. The SegNet architecture [23] has perhaps made the greatest achievement in exploring and refining this trade-off. In the SegNet architecture, instead of transferring entire activation maps from the encoder to the decoder blocks, the so-called *pooling indices* (the maximum location of the encoder feature map) are rather stored and transferred to the corresponding decoder block. This technique has been shown to yield promising and competitive segmentation results in a manner that optimizes the model efficiency in terms of size and training time.

3. INDOOR NAVIGATION VIA SEMANTIC SCENE SEGMENTATION

At a high level, semantic segmentation allows the robot platform to detect and extract the usable path from the rest of the scene (eg. walls, people, doors etc.) in a manner that further allows for the implementation of collision-free navigation.

3.1 Deep Network Architecture

In this work we exploit DCNNs in semantically segmenting the scene captured by the on-board camera for extracting the usable path towards autonomous navigation. Based on the successes of DCNNs in addressing semantic segmentation, we base our models on the approaches in [23] and [24]. The models are however fine-tuned using a transfer learning technique described in the Section

3.2 in order to make them feasible for the task of indoor robot navigation.

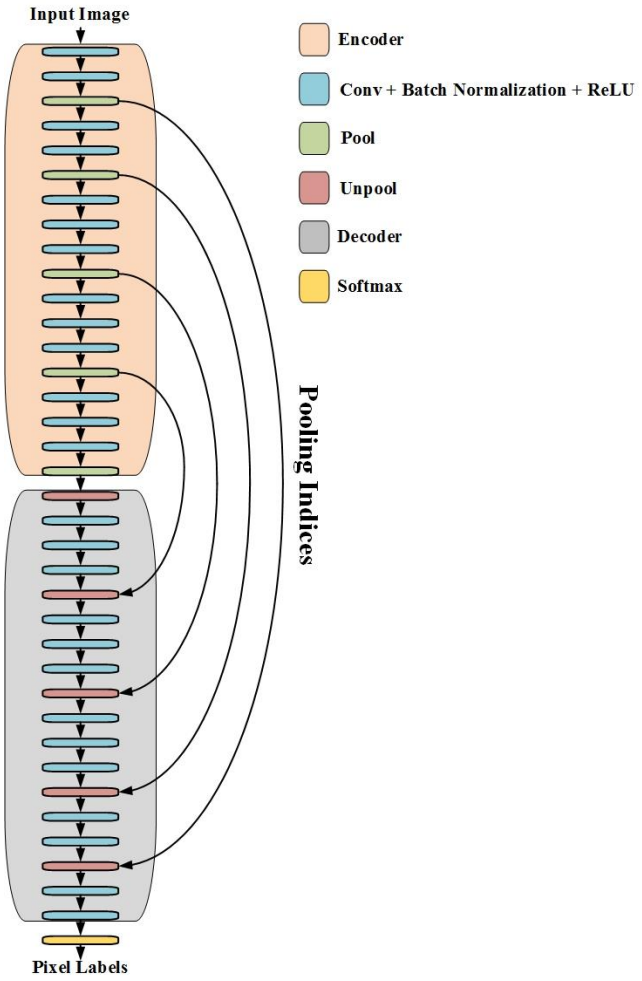

Figure 1. An illustration of the model-1 architecture.

The models are graphically depicted in the Figure 1 and 2. The model-1, based on the approach in [23] which is depicted in the Figure 1 is made up of two main sub networks. The encoder subnetwork which encodes the input image into feature maps is made up of 13 convolutional layers. The convolutions are combined with batch normalization and element-wise Rectified Linear non-linearity layers (ReLU). The batch normalization helps to introduce larger variances into the activations during training time. The subnetwork further exploits 5 *max* pooling layers for down-sampling the feature maps in order to achieve more fine-grained feature representations. The pooling layers adopt a window with a stride size of 2. The window dimensions and stride size ensure that the sliding window does not overlap. We highlight that this yields translation invariance over small spatial shifts. Therefore, by stacking multiple pooling operations in the encoder subnetwork, the translation and subsampling effect is compounded, boosting classification robustness. However, contrary to CNNs designed for the purpose of object classification, in semantic scene segmentation task, there is the need to preserve spatial resolution within the feature maps.

Ideally, the output resolution of the model needs to match the input resolution and the degradation of the activation map resolution which is caused by the pooling operations in the

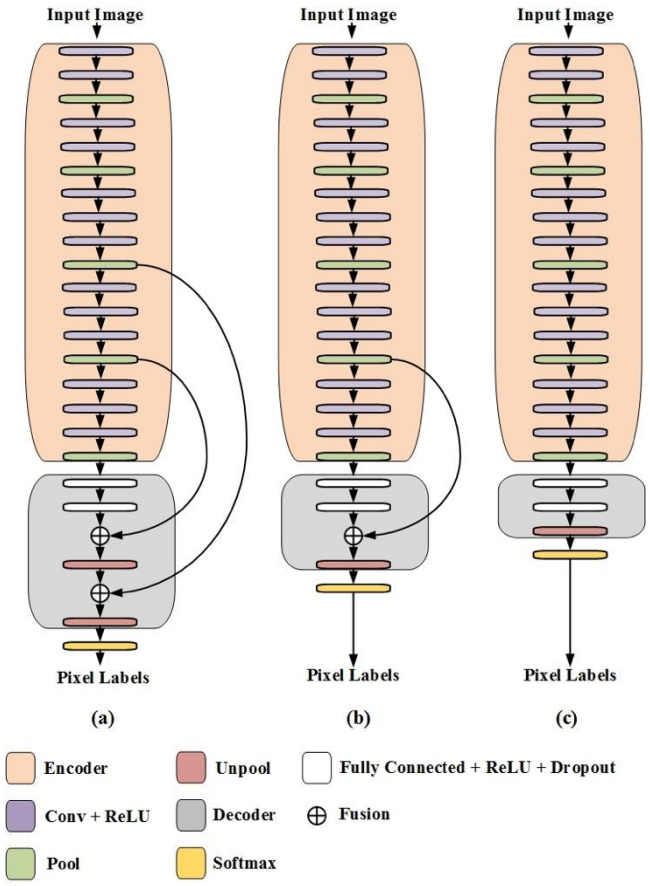

Figure 2. An illustration of the model-2 architecture and its variants.

encoder sub-network needs to be appropriately addressed. This is especially true for navigation applications where the scale and proportions of objects in the scene should not be degraded in the output semantic map. In this model therefore, the pooling indices which represent the maximum feature locations within each pooling window are recorded before pooling is performed. The indices are then stored and transferred to the corresponding *unpooling* layers within the decoder subnetwork where they facilitate consecutive upsampling until the activation maps are restored to full resolution. The decoder subnetwork combines the unpooling layers with additional convolutions, batch normalization and ReLU. After the 13th convolutional layer in the decoder, a Softmax layer is attached for pixel-wise classification. For the task of indoor navigation, since it is our interest to extract the usable path from the rest of the scene, we define two labels of interest as: (1) Usable path and (2) Obstacles.

The second model which is based on [24] is depicted in the Figure 2. As depicted in the Figure 2, the model-2 also consists of encoder and decoder subnetworks. The encoder subnetwork is similar in design to the model-1 in that they both apply 13 convolutional layers and 5 *max* pooling layers. This similarity in encoder structure makes it feasible to quickly implement the encoder subnetworks using the first 13 layers of the VGG-16. We highlight that a slight variation however exists in the fact that in the model-2, there are no batch normalization operations. The key difference between the model-1 and 2 lies in the decoder subnetwork as has been highlighted in the literature review. While the model-1 is symmetrical between the encoder and decoder

subnetworks, the model-2 adopts a much smaller decoder structure. Additionally, a different approach is taken to addressing the activation map upsampling problem. Here in the model-2, the activation maps from the pooling layers are copied, stored and transferred to the decoder for upsampling, contrary to model-1 where only the indices are stored as 2 bits for each window. In this paper, we explore 3 different variants of the model-2 as depicted in the Figure 2. In the model-2-v1 depicted as Figure 2 (a), two activation maps are copied from the encoder pooling layers 3 and 4. These activations are fused at two points within the decoder for feature map upsampling. This allows for more fine-grained activations (8x) to be recovered at a higher cost (memory and computation). The model-2-v2 (Figure 2 (b)) only transfers activation maps from the 4th encoder pooling layer for upsampling at a rate of 16x while in the model-2-v3 (Figure 2 (c)), no feature maps are transferred from the encoder, leading to coarse activation maps (upsampled at 32x). We explore and discuss the impacts of the different upsampling techniques adopted in the v1, v2 and v3 and highlight how they affect the overall model training time, hardware memory usage, segmentation accuracy, path extraction performance and the overall robot navigation robustness.

3.2 Training Approach

The training of deep convolutional neural networks such as the models in this paper requires large training datasets that can satisfy the large entropical capacities of the networks and help them generalize well. However, the manual labeling of datasets is a time-consuming effort and the indoor navigation research area has comparatively smaller public semantic segmentation datasets compared with other applications such as outdoor navigation and Augmented Reality (AR). *Transfer learning* is therefore a critical tool for successfully training our deep models. The transfer learning technique used in this paper is achieved in two steps. (1) *Pretraining*: At this stage, the model is trained on the ImageNet dataset [29] which is made up of about a million images belonging to 20 thousand categories. While this dataset is not explicitly designed for navigation, it allows the models to learn low and mid-level features such as edges and shapes which are useful for a majority of computer vision tasks. (2) *Finetuning*: After the models are pretrained over several epochs until no more gains in accuracy can be achieved, they are finetuned using our own dataset of about 2000 images which are manually collected and labelled for semantic segmentation. Our dataset is highly useful for indoor robot navigation because it is collected using a mounted camera on the robot while a human pilot navigates an indoor scene. This allows the dataset to capture a large variance of the true conditions to be encountered during deployment. The dataset is first recorded in video format and split into multiple images at a frequency of 1 Hertz. Sampled scenes from the dataset and their manual labels are shared in the Figure 3.

Figure 3. Samples from our dataset and accompanying labels used for model fine-tuning.

3.3 Post-Processing

Finally, after semantic scene segmentation has been achieved via the DCNN models, post-processing is applied in refining the predictions and highlighting visual image features (eg. boundaries) for efficient navigation. The morphological *closing* technique is used to improve boundary strength and connect disjoint pixels within the predicted patches. It then becomes possible to detect visual key-points and track them across frames for realizing navigation. In this paper, we exploit the Oriented FAST and Rotated BRIEF (ORB) algorithm in achieving this goal.

4. EXPERIMENTAL RESULTS AND DISCUSSION

In achieving implementation of the models and the respective variants, CUDA v8 and cuDNN v5 libraries are exploited for parallel computation acceleration. Model training is performed on a workstation equipped with two Nvidia GTX 1080 Ti cards. At inference time, the models are run on the robot platform equipped with a single Nvidia GTX 1080 card and an Intel i7 processor. The experimental robot platform is depicted in the Figure 4.

In order to cut down overall training time, the model weights are first initialized using weights from the first corresponding 13 layers of a VGG-16 model pretrained on the ImageNet. This is possible because the first 13 layers of the models are similar to those of the VGG-16 network. This allows for the pretraining time to be significantly cut down. The finetuning of all the models is performed using the Stochastic Gradient Descent (SGD) with momentum. The parameters we have used in successfully training all the models to convergence are shared in the Table 1 below. All models are trained with a batch size of 8 and all batches are shuffled at every epoch.

The models are then tested on novel scenes which are not contained within our training dataset. The semantic segmentation

Figure 4. Experimental robot platform.

results achieved by the models at test time are presented in the Figure 5. The results also highlight the error rate by overlapping the ground truth labels with the predicted labels.

Table 1. Model training parameters

Models	Learning Rate	Momentum	Epochs
Model-1	1e-3	0.9	100
Model-2-v1	1e-3	0.9	50
Model-2-v2	1e-3	0.9	50
Model-2-v3	1e-3	1.0	100

In the Figure 5, the errors are color-coded such that green represents false negatives while light magenta represents false positives. True positives are labeled in black and magenta for scenes 1 and 2 respectively. The results show significantly higher false positive rates in the model-1 (first row) which is based on the approach proposed in [23]. The approach, while comparatively more efficient in terms of memory and inference time, is more unstable and trades off significant accuracy for this efficiency as further supported by the Table 2. On the other hand, the models based on the segmentation approach in [24] achieve higher accuracies by trading off a significant amount of efficiency as shown in the Table 3.

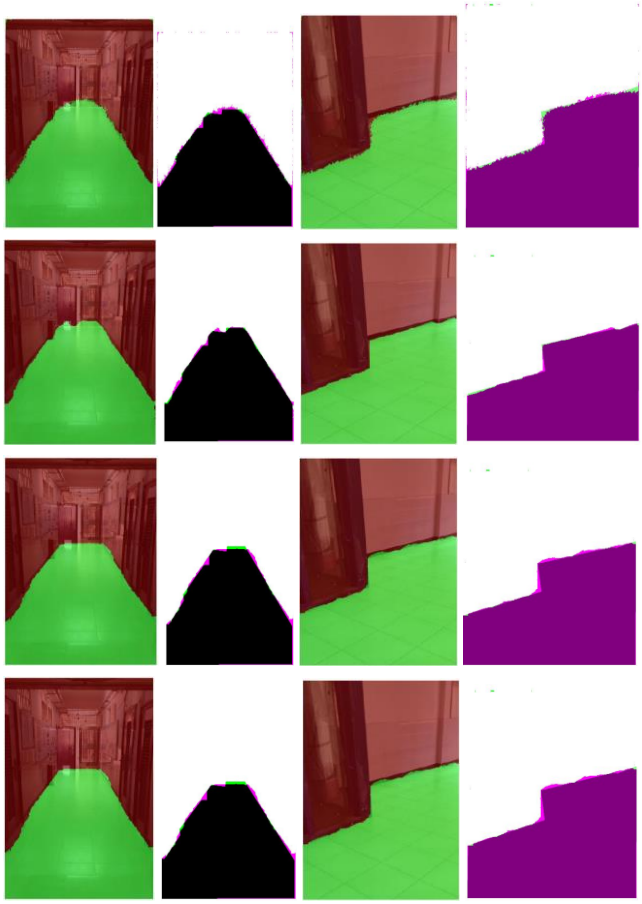

Figure 5. Semantic segmentation results achieved at test time with corresponding error rates.

The models use up more of the robot platform physical memory are more latent. The model-2-v1 (second row) achieves the most superior accuracy followed by model-2-v2 (third row) and then model-2-v3 (fourth row). We see that the model-2-v3 which does not transfer any activation maps to the decoder only slightly outperforms the model-1.

Table 2. Quantitative model performances (mean values)

Models	Accuracy	IoU	BFScore
Model-1	0.9835	0.9669	0.8444
Model-2-v1	0.9905	0.9802	0.9293
Model-2-v2	0.9882	0.9756	0.9036
Model-2-v3	0.9840	0.9664	0.8477

We further share efficiency evaluation results for the models in the Table 3. These performances are measured at test time and represent physical memory consumpiton on the robot platform.

Table 3. Model efficiency evaluation

Models	Memory (MB)	Load Time (s)	Inference Time (s)
Model-1	111.1	0.8	0.18
Model-2-v1	501.2	3.2	0.23
Model-2-v2	501.2	3.1	0.23
Model-2-v3	501.3	3.1	0.23

Finally, the trade-off between efficiency and model accuracy, a core factor to be considered when developing semantic segmenation techniques for autonomous robots with resource contraints is highlighted in the Figure 6.

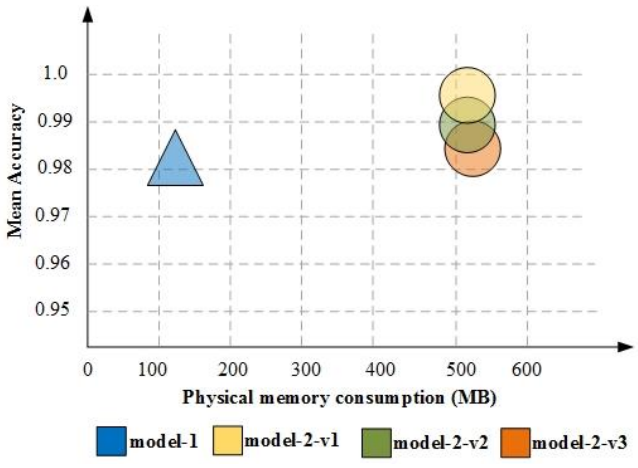

Figure 6. Highlighting the trade-off between model accuracy and efficiency.

5. CONCLUSIONS

This paper proposes a vision-based navigation approach for indoor robot navigation which exploits semantic scene segmentation via deep learning. The main contribution achieved is the exploitation of DCNN and transfer learning in achieving

efficient semantic scene mapping and path extraction for indoor robot navigation. The paper experiments with multiple semantic segmentation approaches and explores the trade-off between efficiency and segmentation accuracy in an experimental manner. The achieved results highlight promising potential for achieving feasible and efficient solutions for indoor navigation for robots with small payloads. Furthermore, by exploring the trade-off between accuracy and efficiency encountered when selecting network architectures and approaches for semantic segmentation, this paper aids the research field by serving as a reference.

Future work will focus on the control strategy that will build directly upon the semantic scene segmentation output (the extracted usable path) and achieve accurate and efficient navigation without the need for additional sensors.

6. ACKNOWLEDGEMENTS

This work is supported by the Science and Technology Program of Guangdong Province under grant No. 2016A050502060 and No. 2016A040403012, and the Science and Technology Program of Guangzhou under grant No. 201604016055, No. 201604046015 and No. HD14ZD001.

7. REFERENCES

[1] Bostelman, R. V., Hong, T. H. and Madhavan, R. 2005. Towards AGV safety and navigation advancement obstacle detection using a TOF range camera. In *Proceedings of IEEE International Conference on Advanced Robotics* (Seattle, WA, USA, July 18-20, 2005). 460-467.

[2] Jalobeanu, M., Shirakyan, G., Parent, G., Kikkeri, H., Peaseley, B. and Feniello, A. 2015. Reliable Kinect-based navigation in large indoor environments. In *Proceedings of IEEE International Conference on Robotics and Automation* (Seattle, WA, USA, May 26-30, 2015). 495-502.

[3] Zhou, Y., Jiang, G., Xu, G., Wu, X. and Krundel, L. 2014. Kinect depth image based door detection for autonomous indoor navigation. In *Proceedings of IEEE International Symposium on Robot and Human Interactive Communication* (Edinburgh, UK, Aug. 25-29, 2014). 147-152.

[4] Correa, D.S.O., Sciotti, D.F., Prado, M.G., Sales, D.O., Wolf, D.F. and Osorio, F.S. 2012. Mobile Robots Navigation in Indoor Environments Using Kinect Sensor. In *Proceedings of Second Brazilian Conference on Critical Embedded Systems* (Campinas, Brazil, May 20-25, 2012). 36-41.

[5] Cucchiara, R., Perini, E. and Pistoni, G. 2007. Efficient Stereo Vision for Obstacle Detection and AGV Navigation. In *Proceedings of 14th International Conference on Image Analysis and Processing* (Modena, Italy, Sept. 10-14, 2007). 291-296.

[6] Solak, S. and Bolat, E.D. 2015. Distance estimation using stereo vision for indoor mobile robot applications. In *Proceedings of 9th International Conference on Electrical and Electronics Engineering* (Bursa, Turkey, Nov. 26-28, 2015). 685-688.

[7] Yuan, W., Li, Z. and Su, C.Y. 2016. RGB-D sensor-based visual SLAM for localization and navigation of indoor mobile robot. In *Proceedings of International Conference on Advanced Robotics and Mechatronics* (Macau, China, Oct. 18-20, 2016). 82-87.

[8] Xi, W., Ou, Y., Peng, J. and Yu, G. 2017. A new method for indoor low-cost mobile robot SLAM. In *Proceedings of IEEE International Conference on Information and Automation* (Macau, China, July 18 – 20, 2017). 1012-1017.

[9] Lee, T., Kim, C. and Cho, D. 2018. A Monocular Vision Sensor-Based Efficient SLAM Method for Indoor Service Robots. *IEEE Transactions on Industrial Electronics*. (April 2018), 1–1.

[10] Kluge, K. and Lakshmanan, S. 1995. A deformable-template approach to lane detection. In *Proceedings of the Intelligent Vehicles Symposium* (Detroit, MI, USA, Sept. 25-26, 1995). 54-59.

[11] Gonzalez, J. P. and Ozguner, U. 2000. Lane detection using histogram-based segmentation and decision trees. In *Proceedings of IEEE Intelligent Transportation Systems* (Dearborn, MI, USA, Oct. 1-3, 2000). 346-351.

[12] Krizhevsky, A., Sutskever, I. and Hinton, G.E. 2012. ImageNet Classification with Deep Convolutional Neural Networks. In *Advances in Neural Information Processing Systems*, 25, 1097-1105.

[13] Ren, S., He, K., Girshick, R. and Sun, J. 2015. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In *Advances in Neural Information Processing Systems*, 28, 91-99.

[14] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S.E., Fu, C. and Berg, A.C. 2015. SSD: Single Shot MultiBox Detector. *CoRR*, abs/1512.02325.

[15] Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., Torralba, A. 2016. Learning Deep Features for Discriminative Localization. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition* (Las Vegas, NV, USA, June 27-30, 2016). 2921-2929.

[16] Karpathy, A. and Fei-Fei L. 2017. Deep Visual-Semantic Alignments for Generating Image Descriptions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 39, 4 (April 2017), 664-676.

[17] Nurhadiyatna, A. and Lončarić, S. 2017.Semantic image segmentation for pedestrian detection. In *Proceedings of the 10th International Symposium on Image and Signal Processing and Analysis* (Ljubljana, Slovenia, Sept. 18-20, 2017). 153-158.

[18] Tai, L., Ye, H., Ye, Q. and Liu, M. 2017. PCA-aided fully convolutional networks for semantic segmentation of multi-channel fMRI. In *Proceedings of International Conference on Advanced Robotics* (Hong Kong, China, July 10-12, 2017). 124-130.

[19] Llopart, A., Ravn, O., Andersen, N. A. and Kim, J. H. 2017. Generalized framework for the parallel semantic segmentation of multiple objects and posterior manipulation. In *Proceedings IEEE International Conference on Robotics and Biomimetics* (Macau, China, Dec. 5-8, 2017). 561-568.

[20] Su, P., Xue, Z., Chi, L., Yang, J. and Wong, S. T. 2012. Support vector machine (SVM) active learning for automated Glioblastoma segmentation. In *Proceedings IEEE International Symposium on Biomedical Imaging* (Barcelona, Spain, May 2-5, 2017). 598-601.

[21] Sturgess, P., Alahari, K., Ladicky, L. and Torr, P. H.S. 2009. Combining appearance and structure from motion features for road scene understanding. In *Proceedings of British Machine Vision Conference* (London, U.K, Sept., 2009).

[22] Shotton, J., Johnson, M. and Cipolla, R. Semantic texton forests for image categorization and segmentation. In *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition* (Anchorage, Ak, USA, June 23-28, 2008). 1-8.

[23] Badrinarayanan, V., Kendall, A. and Cipolla, R. 2017. SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 39, 12 (Dec. 2017), 2481-249.

[24] Long, J., Shelhamer, E. and Darrell, T. 2015. Fully convolutional networks for semantic segmentation. In *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition* (Boston, MA, USA, June 7-12, 2015). 3431-3440.

[25] Ronneberger, O., Fischer, P. and Brox., T. 2015. U-net: Convolutional networks for biomedical image segmentation. I*n Proceedings of Internaitonal Conference on Medical Image Computing and Computer-Assisted Intervention* (Munich, Germany, Oct. 5-9, 2015). 234–241.

[26] Paszke, A., Chaurasia, A., Kim, S. and Culurciello, E. 2016. ENet: A Deep Neural Network Architecture for Real- Time Semantic Segmentation. *arXiv preprint arXiv*: 1606.02147v1.

[27] Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K. and Yuille, A.L. 2018. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 40, 4 (April 2017), 834-848.

[28] Simonyan, K. and Zisserman, A. 2014. Very Deep Convolutional Networks for Large-Scale Image Recognition. CoRR, abs/1409.1556.

[29] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A.l, Bernstein, M., Berg, A.C. and Fei-Fei, L. 2015. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, 115, 211-252.