Journal of Universal Computer Science, vol. 24, no. 4 (2018), 475-487
submitted: 18/10/17, accepted: 15/4/18, appeared: 28/4/18

Real Time Path Finding for Assisted Living Using Deep Learning

Ugnius Malūkas
(Rubedo sistemos, Kaunas, Lithuania
ugnius@malukas.lt)

Rytis Maskeliūnas
(Kaunas University of Technology, Kaunas, Lithuania
rytis.maskeliunas@ktu.lt)

Robertas Damaševičius
(Kaunas University of Technology, Kaunas, Lithuania
robertas.damasevicius@ktu.lt)

Marcin Woźniak
(Silesian University of Technology, Gliwice, Poland
Marcin.Wozniak@polsl.pl)

Abstract: The paper presents a computer vision based system, which performs real time path finding for visually impaired or blind people. The semantic segmentation of camera images is performed using deep convolutional neural network (CNN), which able to recognize patterns across image feature space. Out of three different CNN architectures (AlexNet, GoogLeNet and VGG) analysed, the fully connected VGG16 neural network is shown to perform best in the semantic segmentation task. The algorithm for extracting and finding paths, obstacles and path boundaries is presented. The experiments performed using own dataset (300 images extracted from two hours of video recording walking in outdoors environment) show that the developed system is able to find paths, path objects and path boundaries with an accuracy of 96.1 ± 2.6%.

Keywords: Semantic segmentation, path finding, object recognition, Image processing, Deep Learning, Neural networks, outdoor navigation; Assisted Living.
Categories: I.2.10, I.4.6, I.5.4

1 Introduction

The 2014 World Health Organization (WHO) Fact Sheet of Visual impairment and blindness [WHO 2014] estimates that 39 million people are blind and 246 million have low vision worldwide. The total productivity loss due to visual impairments for 2010 was $ 167 billion and is expected to rise in 2020 up to $ 176 billion [AMD 2010]. This is caused by reduce of hourly work, absenteeism for hospitalisation and premature mortality associated with low vision and blindness, e.g., 50% of blind people have head-related accidents happening once or more a year [Manduchi and Kurniawan 2011]. Assistive technologies [Hersch and Johnson 2010] such as autonomous navigation is very important for secure living of people suffering from visual impairment. Autonomy provides ability to move freely around and live normal daily life, while without freedom of movement, visually impaired people depend on the assistance of other people or even guide dogs. Recognition of activities can provide valuable information on health, wellbeing, and fitness of monitored persons outside a hospital setting [Damasevicius et al. 2016]. As visually impaired people are prone to a higher number of accidents in daily life, because of difficulties of walking outdoors, finding the path, avoiding obstacles and pedestrians, is important. Current research on obstacle avoidance aids for the visually impaired is mostly based on ultrasound sensors (see, e.g., [Shin and Lim 2007]). The use of such systems are often hindered by the need of carrying cumbersome equipment and the quality of recognition is degraded from noise interference and mirror effects.

Using Global Positioning System (GPS) for outdoor navigation assistance suffers from low accuracy in city environment, signal loss due to multi-path effect, blocking from to the presence of buildings or trees [Rodrigues et al. 2012]. As the accuracy of commercial GPS is limited to about 20 m, visually impaired users can face danger when walking in unfamiliar urban environment [Loomis et al. 2001].

Radio-frequency identification (RFID) help blind people to walk without being accompanied while using touching clues such as smart canes [Nassih et al. 2012]. While RFID based recognition is unaffected by poor lighting or other constraints that hinder the efficiency of other systems, there are other problems related to dependence upon battery lifetime, slow reaction time and floor-level detection.

Computer vision based systems usually use a monocular or stereo camera or smartphone attached to the chest, shoulder, or mounted on a head [Martınez and Ruiz, 2008; Pradeep et al 2010; Tapu et al. 2013]. Such systems aim to provide an accurate position of a person with respect to his/her environment and other valuable information of the environment such as 3D scene understanding [Geige et al. 2011], and environment mapping [Saez et al. 2005]. Systems requiring minimal or none intervention of an user or a leading persons are preferred for blind people such as a system that estimated depth from a single image based on local depth hypothesis [Praveen and Praily 2013].

More recently, the Kinect sensor has been used for indoor navigation and obstacle avoidance, however, it is not reliable outdoors due to sun illumination that hinders correct depth acquisition [Li et al. 2015].

Multimodal feedback systems provide several output channels (audio, haptic) [see, e.g., [Dakoupolous and Bourbakis 2009]. Communication channels may include range and GPS sensors, 2D vibration vest, and an ear speaker with a speech synthesizer. Information about obstacles in an image in front of a blind person can be transformed (sonified) into auditory feedback to guide a blind user around obstacles [Wörtwein et al. 2016; Sekhar et al. 2016].

In this paper, we present a computer vision based system, which performs real time path finding for visually impaired or blind people. The aim is to investigate and apply a semantic segmentation algorithm capable of recognizing, walk paths, routs, and various objects in the visual material. The rest of the paper is organized as follows. Section 2 reviews the state-of-the-art methods of semantic segmentation based on using Convolutional Neural Networks. Section 3 describes the proposed method for image segmentation. Section 4 presents the experimental results considering challenging real-world road environments with independent moving objects. Finally, Section 5 concludes the article and discusses future work.

2 State-of-the-Art of semantic segmentation using Convolutional Neural Networks

Semantic segmentation seeks to find meaningful regions in the processed images, and then assign them to a specific class based on the characteristics of each pixel [Long et al. 2015]. The image recognition system uses a problem-solving algorithm based on the use of a deep neural network (NN) that allows to store information about the features inherent to the image, to process it, compare it with each other, and to use collected information in a similar situation. The neural network discovers common features and features of the objects in an image. The use of a neural network to segment objects in pictures and to examine how features discovered by objects travel through various neuronal network architectures is a prerequisite for the transfer of complex human perception processes into the computer space and the ability to develop smart applications for assisted living environments such as, e.g., for inferring user behaviours and intents [Fornaia et al. 2015].

The extraction and classification of image features has always been important for computer vision [Gabryel and Damaševičius, 2017]. Currently, Convolutional Neural Networks (CNN) [LeCun and Bengio 1995] are considered as top selection for classification architecture in semantic segmentation systems. CNNs are neural networks specifically designed to accept 2-dimensional input data such as images. In at least one layer of the network, the 2D convolution of a 2-dimensional image with a 2-dimensional kernel, given by the following equation, must be present:

$$C(i,j) = (I * K)(i,j) = \sum_m \sum_n I(m,n) K(i-m, j-n) \quad (1)$$

A typical convolutional neural network is mainly composed of input layer, convolution layer, downsampled layer (pool layer), all connection layer and output layer. An important part of the layered architecture is a filter (or kernel), i.e., an operator applied to the image to obtain activation maps, i.e., regions where features specific to the kernel have been detected in the input. The output of the convolution operation is usually run through a nonlinear activation function and then further modified by means of a pooling function, which replaces the output in a certain location with a value obtained from nearby outputs. This pooling function helps make the representation learned invariant to small translations of the input and performs subsampling of the input data. Common CNN architecture alternates convolution and subsampling layers. Deep CNN reduce the dimensionality of image by increasing the number of hidden layers (convolutional layers and sampling layers), and extracts the sparse image features in a low-dimensional space.

Most known examples of CNNs are AlexNet [Krizhevsky et al. 2012] and GoogLeNet [Szegedy et al. 2015]. Alexnet has only eight layers, while the first five layers are convolutional layers followed by fully connected layers. GoogLeNet is more complex and has 22 layers and an inception module. The inception module consists of a Network in Network (NiN) layer, a pooling operation, a large-sized convolution layer, and small-sized convolution layer, computed in parallel and followed by 1×1 convolution operations to reduce dimensionality.

VGGNet [Simonyan and Zisserman 2014] consists of up to 19 layers, but a number of parameters is prevented from being increased drastically by using small receptive fields (3x3 and 1x1 filters). Currently, the VGG model is often used because the neural network is simple and works very well in classifying and localizing tasks.

ResNet [He et al. 2016] is a 152 layer ultra-deep network architecture. It is based on the residual module (ResNet-blocks), which introduce skip or shortcut connections, and make it easy for network layers to represent the identity mapping. Rather than learning an output function, the residual block only learns the residual, thus allowing NN models to be trained effectively.

SegNet [Badrinarayanan et al. 2015] architecture consists of several stages. The decoder stage of SegNet has a set of upsampling and convolution layers which are at last followed by a softmax classifier to predict pixel-wise labels for an output. Each upsampling layer in the decoder stage corresponds to a max-pooling one in the encoder part. Those layers upsample feature maps use max-pooling indices from their corresponding feature maps in the encoder phase. The upsampled maps are then convolved with a set of trainable filter banks to produce feature maps. When the feature maps have been restored to the original resolution, they are fed to the softmax classifier to produce the final segmentation

Region-based convolutional neural networks (R-CNNs) [Girschick et al. 2014] share convolutional layers with object detection networks by adding additional layers to evaluate the objectness scores at each location on a regular grid. Selective Search generates regions with the highest probability of containing an object. Then these regions are submitted to a trained CNN that extracts a feature vector for each region, which is further used for classification.

Fully Convolutional Network (FCN) [Long et al. 2015] is trained by using end-to-end training (data input directly related to data output) and the "pixel by pixel" principle, semantic segmentation is a modern technology that does not require further processing of data. Existing neural networks, which are fully interconnected and, at the same time, convolutional, are able to find densities that are of the same dimensions as inputs. Training and getting results are performed simultaneously with the use of Feedforward principle when data is transmitted to the neural network and backpropagation when errors are calculated and neural network weights are updated. The upsampling of pixels opens up the ability to get a "pixel by pixel". Fully Convolutional Networks extend the ordinary conventional convolutional neural networks, since instead of a particular result (for example, the classification result) it is possible to obtain the output result as same dimensions as the input image.

3 Method

3.1 Model and architecture of neural network

VGG16 neural model (Figure 1) have been modified to FCN-VGG16 neural network. The VGG16 model is characterized by its simplicity, since it only goes deep by using convolutional operations and does not have a network in network as the GoogLeNet neural model. The fourth (pool4) and fifth (pool5) max pooling layers were extracted from the VGG16 Neural Network for the segment result of this project. The fifth max pooling layer and the eighth fully-connected layer (score_fr) were combined in order to obtain an intermediate result that contains a small-dimensional segmentized view. After that, the result is doubled to match the size of the pool4 layer. The modified pool4 layer, in turn, also has a certain intermediate segmental image, which is then spun so that the resulting result would not be angled (pixeled). Finally, pool4 and pool5 layers with segmented views are summed.

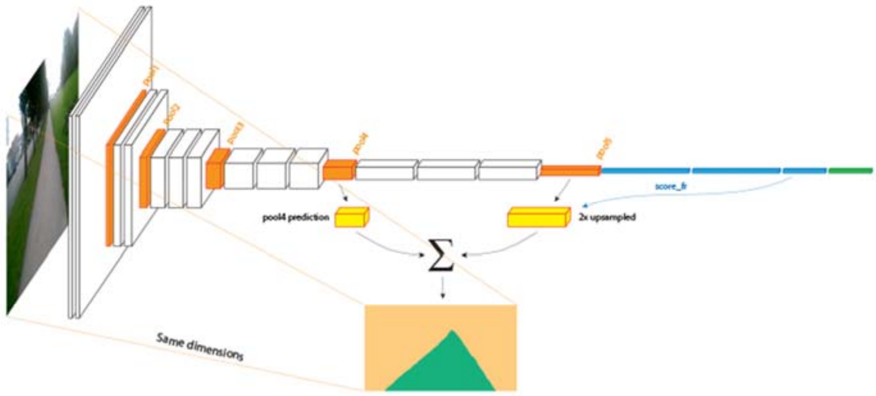

Figure 1: Adapted VGG16 for semantic segmentation scheme

3.2 Teaching

By doing training of neural network, it is necessary to choose a proper loss minimization function which helps to achieve result how good the neural model training was. Here we use cross entropy as loss function:

$$L = \frac{1}{N}\sum_{i=1}^{N} E(\sigma_i, g_i) \tag{2}$$

here L – loss minimization function, which is the mean of cross entropies, N – class number, and g_i – input, and E is cross entropy:

$$E(\sigma, g) = -\sum_{i=1}^{K} g_i \ln(\sigma(z_i)_k) \tag{3}$$

here z_i – intermediate result, g_i – input, and σ_i – softmax activation function:

$$\sigma(z)_j = \frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}} \tag{3}$$

here $\sigma(z)_j$ – softmax activation function, z_j – neuron weight and input product, e – mathematical constant. Softmax activation function helps to categorize inputs into probabilistic categories.

Cross entropy works like statistical distance measure: the lower the value of entropy, the closer the predicted class is to the correct class, and the greater the entropy value, the further the statistical distance is to the incorrect class.

$$z_j = w_j^{\mathsf{T}} \cdot x \tag{4}$$

here z_j – weight and neuron product, w_j^{T} – neuron weights, x – input.

The optimization method helps minimize the loss function. We use the Adam optimization algorithm, which has been demonstrated to perform well in solving various problems and is very versatile in the formation of neuronal models [Kingma].

4 Experiments

4.1 Dataset

We use the VGG16 neural network trained on the ImageNet data set [Deng]. ImageNet is a platform that contains a large database of photos with their descriptions. ImageNet's dataset consists of over 14 million text-wrapped photos that are categorized into more than 22,000 categories.

Figure 2: Examples of images in additional dataset

VGG16 neural network has been trained with additional data collected specifically to address obstacle and path finding challenges. Dataset consists of visual material obtained in different seasons, with varying lighting conditions and different pavements. The additional data set consists of 300 different 320x180 sized photos. Objects in each picture can be divided into three classes: 1) path, 2) obstacle, 3) everything else that is neither a road nor an obstacle (road boundaries). Obstacles were those objects that are on the road or touching it, or lying across the road. A sample of images from the additional dataset is given in Figure 2.

4.2 Experimental setting

The training was performed on a computer with an Intel Core i7-6700 CPU @ 3.40GHz processor, 16 GB of RAM. The video card has no effect on training because it has not been used. The single image processing speed of a neuron model is 105 ms or 9.5 frames per second.

4.3 Generated filters

During the training, it is very important to pay attention to how neural network can store the features itself. In order to get good results, it is useful to consider the information stored in the filters. Intermediate layers contain information on how well certain neural network operations work. Figure 3 demonstrates how inputs effects the first convolution layer filters. We can clearly see the results of feature detection, and that ReLU activation function worked well.

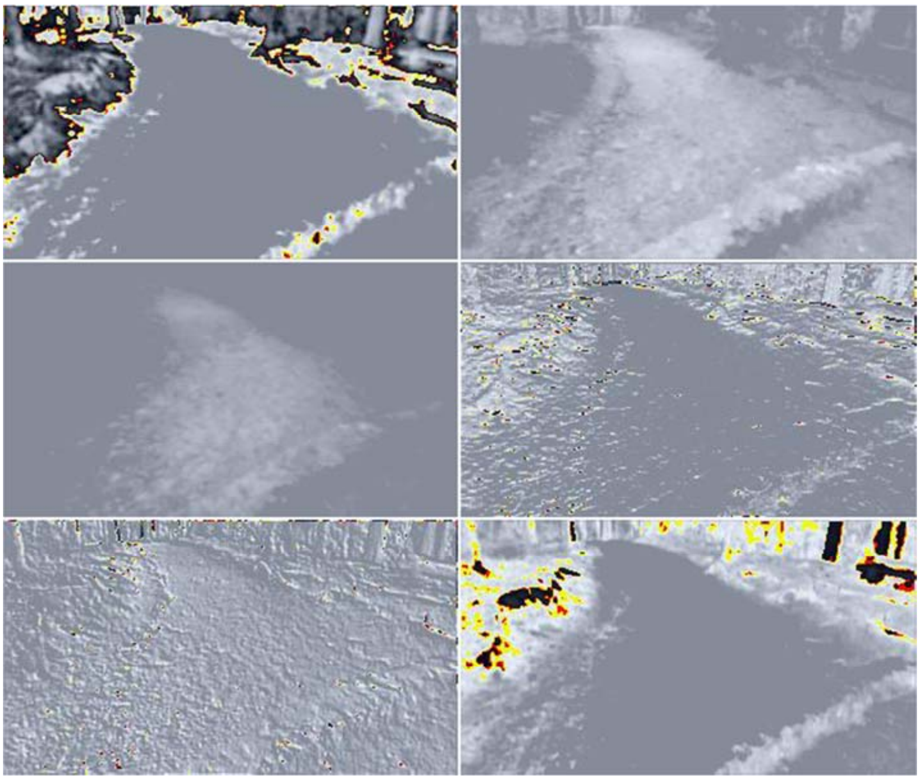

Figure 3: Examples of the 1st convolutional layer filters.

Note that images of 11th convolution layer outputs (filters) are scaled for clearer visualization (see Figure 4). In reality, the dimensions of these filters dimension are much smaller.

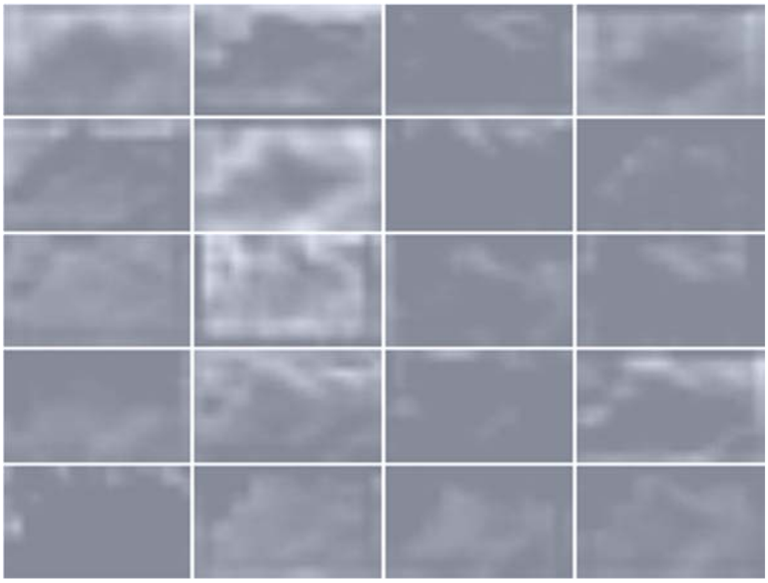

Figure 4: Examples of the 11th convolutional layer filters.

4.4 Evaluation

We have evaluated the accuracy of the model calculated as follows:

$$A = \frac{1}{N}\sum_{i=1}^{N} D\left(p_i, r_i\right) \tag{5}$$

here A is the accuracy of the instantiated and actual values of the dataset being compared, N is the number of units of the dataset to be compared, D is the difference between the true value and the% function, p_i is the result obtained, and r_i is the actual input, and D is the ratio between the received and the true value, p is the result obtained, r is the actual input as follows:

$$D\left(p, r\right) = \frac{p}{r} \cdot 100\% \tag{6}$$

For evaluation, a dataset was divided into a training set, a validation set, and testing set.

The training set is constructed from 80% of the dataset, validation – from 10% and testing – from the remaining 10%. Validation and testing sets were not used in the training process. The validation set is used to adjust the model in such a way that the intermediate results are as good as possible. Meanwhile, the testing set is only used to train the model itself, but it is not used either in the training process or in the modification of the model.

4.5 Results

The standard deviation of the test data set is 2.6%. The standard deviation diagram of the test data set (see Figure 5) shows the accuracy of the images and the standard deviation limits. The accuracy varies between 90.4% and 99.5%.

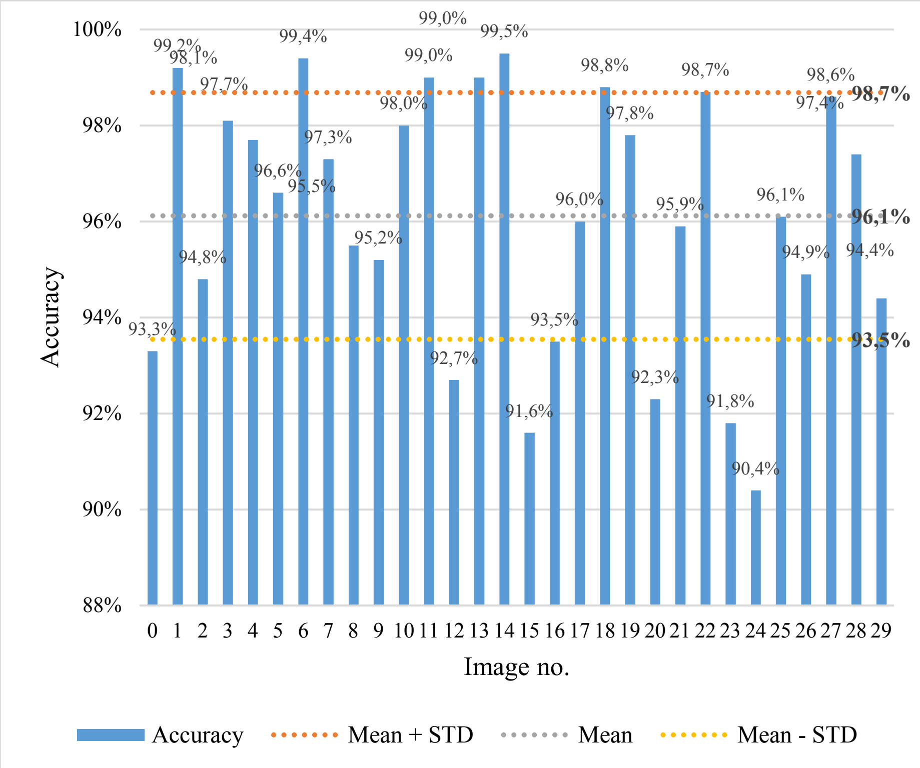

Figure 5: Standard deviation diagram of the test data set.

In Figure 6, the actual entries images used and the results obtained are presented. In parentheses, the segmentation accuracy compared to the actual input is presented. The real-time results from the implementation of the method is presented in the supplement video, see: https://youtu.be/jTj_mzeoVm0.

The analysis of the results show that the trained neural model works well (93.3% accuracy) under the following conditions: tile-based path, bright daytime, and cloudy. The obstacles on the sides of the roads have not been found. The results in the daylight environment during an asphalt road in a forest environment are excellent (99.2% accuracy). However, the edges of the grass are found only approximately. During the day, the paved road is found to be 94.9% accuracy, while in the twilight winter conditions the method achieves 97.4% accuracy on an asphalt road. The fallen leaves visible on the photo do not affect the way the track is found. The neural model is able to find people at a bright daytime (98.8% accuracy). The algorithm runs 99.0% accuracy under winter conditions, fog and, and when the road is made of tiles.

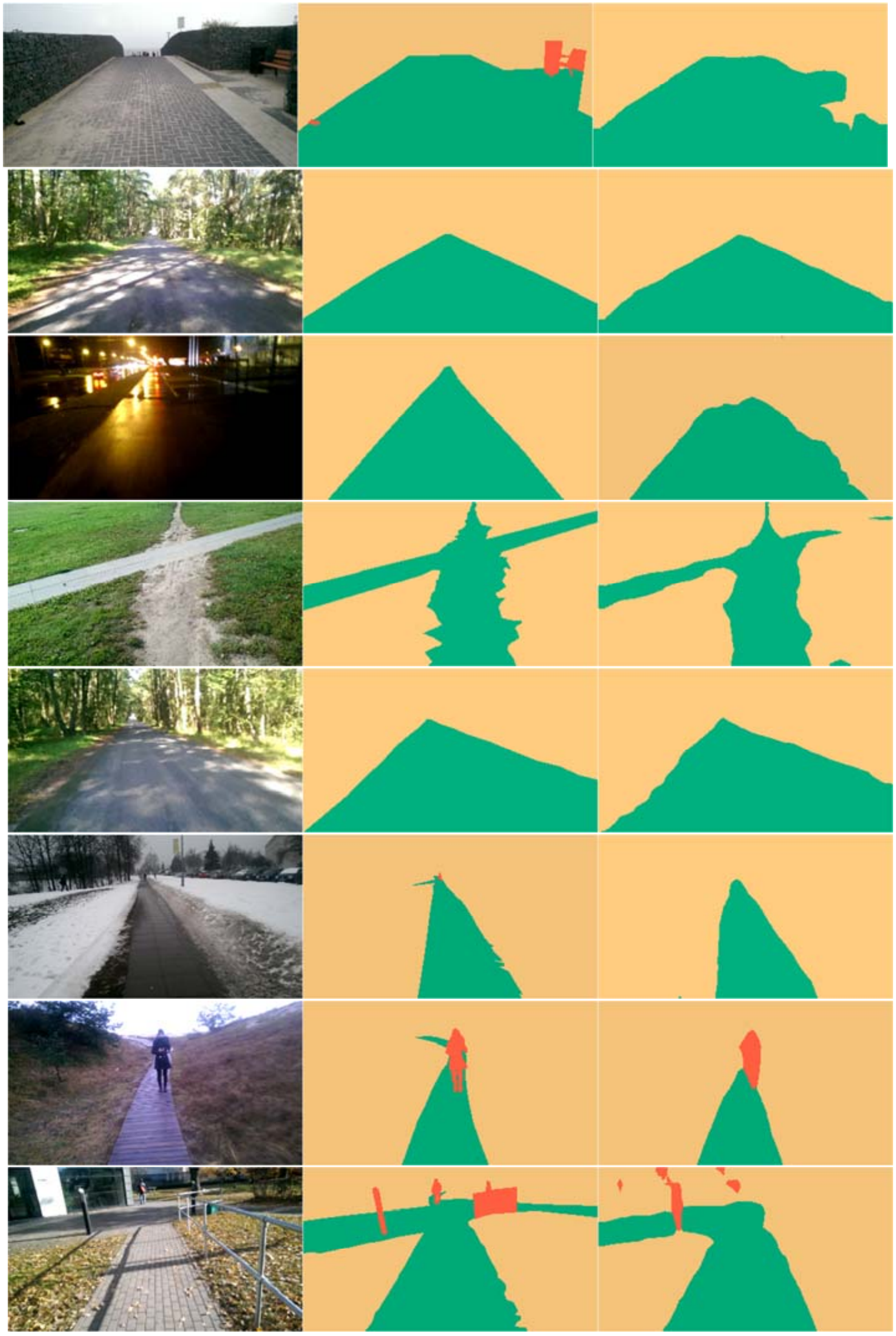

Figure 6: Examples of results: image (left), ground truth (center) and result (right)

5 Conclusions

The developed computer vision based system for real-time path finding based on fully convolutional neural network VGG16 (FCN-VGG16) presented here has an accuracy at of 96.1±2.6%. The average performance of a single-image model for a model is 105 ms or 9.5 frames per second. Comparing the results of the images, we observed that the system works with similar accuracy under different ambient conditions (lighting, seasonality of the year). The average accuracy using good daylight images is 96.2 ± 2.6%, and the use of photos in the twilight is 95 ± 2.1%. The results indicate that the algorithm works best (97.5 ± 2.4% accuracy) when the image has a clear boundary of the way: the edges of the asphalt road, the snow-covered sideways. The algorithm finds objects poorly when these objects are in the distance (not detecting 20 or more meters away from objects located far away from the camera), are within the boundaries of the channel, or the boundaries are unclear (the trail and its edges are measured). The system is able to distinguish between different paths (tile-based - 96.1 ± 2.8%, wood-based - 95.3 ± 3.5%, asphalted - 97.9 ± 1.5%, trampled by foot - 95.1 ± 2.2%) under different seasonality (snow - 98.4 ± 0.7%, leaves dropped by trees - 95.1 ± 2.6%).

The developed system can be used as a digital assistant to ensure higher safety of visually impaired people living and autonomous life.

References

[AMD 2010] AMD Alliance International, The Global Economic Cost of Visual Impairment, March 2010.

[Badrinarayanan et al. 2015] V. Badrinarayanan, A. Kendall, and R. Cipolla, „SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation“, Cvpr 2015, p. 5, 2015.

[Dakopoulos and Bourbakis, 2009] D. Dakopoulos and N. Bourbakis, "Towards a 2D tactile vocabulary for navigation of blind and visually impaired," 2009 IEEE International Conference on Systems, Man and Cybernetics, San Antonio, TX, 2009, pp. 45-51.

[Damasevicius et al. 2016] R. Damasevicius, M. Vasiljevas, J. Salkevicius, and M. Wozniak. Human Activity Recognition in AAL Environments Using Random Projections. Comp. Math. Methods in Medicine 2016: 4073584:1-4073584:17 (2016)

[Deng et al. 2009] J. Deng, W. Dong, R. Socher, L.-J. Li, F.-F. Kai, ImageNet: A Large-Scale Hierarchical Image Database, Conference on Computer Vision and Pattern Recognition, CVPR 2009, 248-255.

[Fornaia et al. 2015] A. Fornaia, C. Napoli, G. Pappalardo, and E. Tramontana. Using AOP Neural Networks to Infer User Behaviours and Interests. In Proceedings of the 16th Workshop "From Objects to Agents", Naples, Italy, June 17-19, 2015. CEUR Workshop Proceedings 1382, 46-52.

[Gabryel and Damaševičius 2017] Gabryel M., Damaševičius R. (2017) The Image Classification with Different Types of Image Features. In: International Conference on Artificial Intelligence and Soft Computing ICAISC 2017. LNCS, vol 10245. Springer, Cham., 497-506.

[Geiger et al. 2011] Geiger, A.; Lauer, M.; Urtasun, R. A Generative Model for 3D Urban Scene Understanding from Movable Platforms. In IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, 20–25 June 2011; pp. 1945–1952.

[Girshick et al. 2014] R. Girshick, J. Donahue, T. Darrell, and J. Malik, Rich feature hierarchies for accurate object detection and semantic segmentation, in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014, 580-587.

[He et al. 2016] K. He, X. Zhang, S. Ren, and J. Sun, Deep residual learning for image recognition, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016, 770-778.

[Hersh and Johnson 2010] Hersh, M., Johnson, M.A.: Assistive Technology for Visually Impaired and Blind People. Springer, Berlin (2010)

[Kingma and Ba, 2014] D. Kingma and J. Ba, Adam: A Method for Stochastic Optimization, Int. Conf. Learn. Represent., p. 1–13, 2014.

[Krizhevsky et al. 2012] A. Krizhevsky, I. Sutskever, ir H. Geoffrey E., „ImageNet Classification with Deep Convolutional Neural Networks“, Adv. Neural Inf. Process. Syst. 25, p. 1–9, 2012.

[LeCun and Bengio 1995] Y. LeCun and Y. Bengio: Convolutional Networks for Images, Speech, and Time-Series, in Arbib, M. A. (Eds), The Handbook of Brain Theory and Neural Networks, MIT Press, 1995.

[Li et al. 2015] B. Li, X. Zhang, J. P. Munoz, J. Xiao, X. Rong and Y. Tian, "Assisting blind people to avoid obstacles: An wearable obstacle stereo feedback system based on 3D detection," 2015 IEEE International Conference on Robotics and Biomimetics (ROBIO), Zhuhai, 2015, pp. 2307-2311.

[Long et al. 2015] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3431-3440 (2015).

[Loomis et al. 2001] Loomis, J.; Golledge, R.G.; Klatzky, R.L. GPS-Based Navigation Systems for the Visually Impaired. In Barfield, W., Caudell, T. Eds.; Fundamentals of Wearable Computers and Augmented Reality; Lawrence Erlbaum Associates Publishers, 2001.

[Manduchi and Kurniawan, 2010] R. Manduchi and S. Kurniawan. Mobility-related accidents experienced by people with visual impairment. Research and Practice in Visual Impairment and Blindness, 4(2):44–54, 2011.

[Martinez and Ruiz 2008] J. M. S. Martinez and F. E. Ruiz. Stereo-based aerial obstacle detection for the visually impaired. In Workshop on Computer Vision Applications for the Visually Impaired, 2008.

[Nassih et al. 2012] Nassih, M., Cherradi, I., Maghous, Y., Ouriaghli, B., Salih-Alj, Y.: Obstacles recognition system for the blind people using RFID. In: 6th International Conference on Next Generation Mobile Applications, Services and Technologies, 2012, pp. 60–63 (2012).

[Pradeep et al. 2010] V. Pradeep, G. Medioni, and J.Weiland. Robot vision for the visually impaired. In 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 15–22, 2010.

[Praveen and Paily 2013] Praveen, R.G., Paily, R.P.: Blind navigation assistance for visually impaired based on local depth hypothesis from a single image. Procedia Eng. 64, 351–360 (2013).

[Rodrıguez et al. 2012] A. Rodrıguez, J.J. Yebes, P.F. Alcantarilla, L.M. Bergasa, J. Almazan and A. Cela. Assisting the Visually Impaired: Obstacle Detection and Warning System by Acoustic Feedback. Sensors 2012, 12, 17476-17496.

[Saez et al. 2005] Saez, J.M.; Escolano, F.; Penalver, A. First Steps towards Stereo-Based 6DOF SLAM for the Visually Impaired. In IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops, San Diego, CA, USA, 25 June 2005; p. 23.

[Sekhar et al. 2016] V.C. Sekhar, S. Bora, M. Das, P. K. Manchi, S. Josephine and R. Paily, "Design and Implementation of Blind Assistance System Using Real Time Stereo Vision Algorithms," 2016 29th International Conference on VLSI Design and 2016 15th International Conference on Embedded Systems (VLSID), Kolkata, 2016, pp. 421-426.

[Shin and Lim 2007] B.-S. Shin and C.-S. Lim. Obstacle detection and avoidance system for visually impaired people. In Haptic and Audio Interaction Design, 78–85. Springer, 2007.

[Simonyan and Zisserman 2014] K. Simonyan and A. Zisserman, „Very Deep Convolutional Networks for Large-Scale Image Recognition“, ImageNet Chall., p. 1–10, 2014.

[Szegedy et al. 2015] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich„ Going deeper with convolutions, In the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2015, p. 1–9.

[Tapu et al. 2013] R. Tapu, B. Mocanu, A. Bursuc, and T. Zaharia. A smartphone-based obstacle detection and classification system for assisting visually impaired people. In 2013 IEEE International Conference on, Computer Vision Workshops (ICCVW), 444–451, 2013.

[WHO 2014] WHO Visual impairment and blindness World Health Organization (WHO), Fact Sheet N°282, Updated August 2014.

[Wörtwein et al. 2016] Wörtwein T., Schauerte B., Müller K., Stiefelhagen R. (2016) Mobile Interactive Image Sonification for the Blind. In: Miesenberger K., Bühler C., Penaz P. (eds) Computers Helping People with Special Needs. ICCHP 2016. Lecture Notes in Computer Science, vol 9758. Springer, 2016, 212-219.

[Yeon and Kwangsu, 2013] K. S. Yeon and C. Kwangsu, Usability and Design Guidelines of Smart Canes for Users with Visual Impairments , International Journal of Design, 7(3), 2013, 99-110.