EVALUATION OF DEPTH MODALITY IN CONVOLUTIONAL NEURAL NETWORK CLASSIFICATION OF RGB-D IMAGES

Michal VARGA, Ján JADLOVSKÝ
Department of Cybernetics and Artificial Intelligence, Faculty of Electrical Engineering and Informatics,
Technical University of Košice, Letná 9, 042 00 Košice, Slovak Republic,
E-mail: michal.varga@tuke.sk, jan.jadlovsky@tuke.sk

ABSTRACT

This paper investigates the value of depth modality in object classification in RGB-D images. We use a simple model based on a multi-layered convolutional neural network which we train on a dataset of segmented RGB-D images of household and office objects. We evaluate and quantify the benefit of additional depth modality and its effect on classification accuracy on this dataset. Also, we compare the benefit of depth channel against the addition of color to grayscale image. Our experimental results support a conclusion, that for these categories of objects the depth modality provides a significant benefit to classification, which also outweighs the benefit of color information. Similar supporting evidence found in recent research is shown in comparison along with the resulting quantified benefit of depth modality.

Keywords: *3D imaging, computer vision, convolutional neural network, deep learning, object classification*

1. INTRODUCTION

In the last few years, 3D scanners and cameras have become increasingly affordable and commonplace, especially in the segment of mobile robotics. With products like Microsoft® Kinect™, the cost of equipping a robotic platform with a good quality 3D sensor is lower than ever. In robotics, the fusion of 2D and 3D information is widely utilized by various navigation, SLAM (Simultaneous localization and mapping) and object detection and classification algorithms to name a few. The 3D information is essential especially for mapping and navigation algorithms, however, object detection and classification solutions can also utilize this modality. While its benefit is intuitive, we decided to investigate the object classification domain and quantify the value of additional image dimension with regards to RGB-D (RGB + depth channel) image classifier performance.

Currently there are many approaches to 3D image classification. The choice of technique is affected by many factors, one of them being the format of classified data. While the various representations are usually interconvertible, not all conversions are bi-directional and lossless. This is due to the fact that not all formats can contain arbitrary amount of data. This is the case with RGB-D images, which only represent a single "view" of the object and therefore usually miss those parts that are occluded or facing away from the camera. However, they are well suited for our experiment, because the color or intensity information is easily separable from the spatial information (depth), and therefore the modalities can be compared with little difficulties.

As mentioned, single RGB-D frames provide little information about the complete shape of the object. Sometimes, a single view is enough for reliable classification but there are cases when the view angle limits the available information. In robotics, the usual way to deal with this problem is to acquire several images of the object from multiple points of view. These RGB-D images or point clouds are then fused together to create a single 3D representation of the object. This form of information then requires a slightly different approach to classification, for example a 2D classifier fed with multiple 2D projections of the object or a fully 3D convolutional network.

2. RELATED WORK

Many different approaches to 3D classification have been published. The dataset we chose to use (CIN-DB) was published alongside a paper [1] describing authors' approach to classification of RGB-D images. In this paper the authors propose a classification technique which utilizes both 2D (SURF, Pyramids of histograms of HOG (Histogram of oriented gradients), Self Similarity Features and color histograms) and 3D local descriptors (3D Shape Context, Depth Buffer, Shape-Index Histograms and MD2 Shape Distributions). Feature vectors are finally classified by an ensemble of SVM (Support vector machine) classifiers and an MLP (Multilayer perceptron).

2.1. Classification of RGB-D data

Classification of RGB-D images is often achieved by utilizing convolutional neural networks (CNNs). [2] presents a novel technique for RGB-D image classification and object pose estimation using deep convolutional neural networks. Authors preprocessed the images by masking the color and colorizing the depth channel. Several SVM's were used for final classification and for pose estimation authors designed a two-step regressor (SVM + RBF (Radial basis function) kernel support vector regressor). Authors managed to improve the state-of-the-art in classification and pose accuracy using this technique.

Another technique presented in [3] also utilizes a CNN in a two-stream architecture. Outputs from two convolutional networks (for color and depth channel) converge in one fully connected layer and a softmax classifier. Authors improved their accuracy by augmenting the dataset by corrupting the samples with realistic noise patterns.

Another multimodal technique has been presented in [4]. Authors utilize a depth-coloring technique to convert the depth channel into two additional RGB channels. One is created by color-jet technique, the other contains RGB-encoded surface normals. The proposed model architecture contains three convolutional streams (RGB, color-jet, surface normals) and their outputs are concatenated and fed into a softmax layer. Authors used reference implementations of GoogLeNet [5] and CaffeNet [6] for RGB and depth channels respectively.

High accuracy achieved using CNNs has been reported in a paper [7] comparing several deep learning models for classification of RGB-D images and videos. Published results show that CNN-based models outperform DBNs (Deep Belief Networks), SDAEs (Stacked Denoising Autoencoders) and also LSTMs (Long Short Term Memory networks) on video datasets.

2.2. Classification of 3D data

2D convolutional networks may also be used for 3D-only datasets. Some approaches are based on projecting the 3D data to 2D images and then using those 2D views to classify the 3D object. One of the recent research papers in this area documents a technique which uses 2D projection in multi-view CNN network in combination with saliency-based boosting [8]. The proposed method outperformed all 2D view based CNNs and several 3D classifiers, such as 3D ShapeNets [9] and VoxNet [10].

Some models, like VGG3D [11] try to utilize depth information by generating RGB voxel grids from RGB-D images. These are then fed into a modified VGGnet model which is pre-trained on a large 2D RGB dataset. Only the input layer is modified by replicating the pre-trained convolution kernels in the third dimension. The resulting architecture is then fine-tuned on 3D (RGB-D) data. In this publication, authors also present a comparison between RGB, depth-only and RGB-D input modalities with results that support our hypothesis.

The main benefit of projection based techniques over the 3D convolution is computational performance. While the convolution is not a very expensive operation, introduction of additional dimension significantly increases the number of required operations. However, the classification performance of 3D convolutional techniques is superior, which has been proven by several model architectures, like the VRN Ensemble [12]. Since the hardware keeps getting cheaper and more powerful, we consider the 3D convolution a very perspective approach to 3D classification.

3. DATASET

To train the model and evaluate our objectives, we decided to use a publicly available dataset of 2D color and depth images of common household and office objects published by Browatzki et al. [1]. This dataset contains labeled images of objects from 18 categories, each containing 3 to 14 exemplars. Each object was recorded 36 times on a turntable, rotating 10° between each recording. One view consists of a high resolution 2D color image and a spatial scan from a time-of-flight camera. The views are already segmented so they only contain the scanned object while the background is set to zero in both color and shape image (see Fig. 1). [1]

3.1. Data preprocessing

To be able to feed the images into the convolutional neural network, we had to resize each view to a common square shape. Since most views were cropped, and therefore rectangular, both color and depth images were padded as necessary to preserve their aspect ratio during resizing.

One final preparation step was to randomly split the dataset into training and testing subset. We decided to keep 10% of views from each class for testing and to use the rest for training the model. Browatzki et al. [1] used a different strategy, omitting whole instances (objects) when creating the test set. This allowed them to evaluate their model's ability to learn the intra-class variance. This is beyond the scope of this paper, since our goal is solely to evaluate the benefit of spatial (depth) modality.

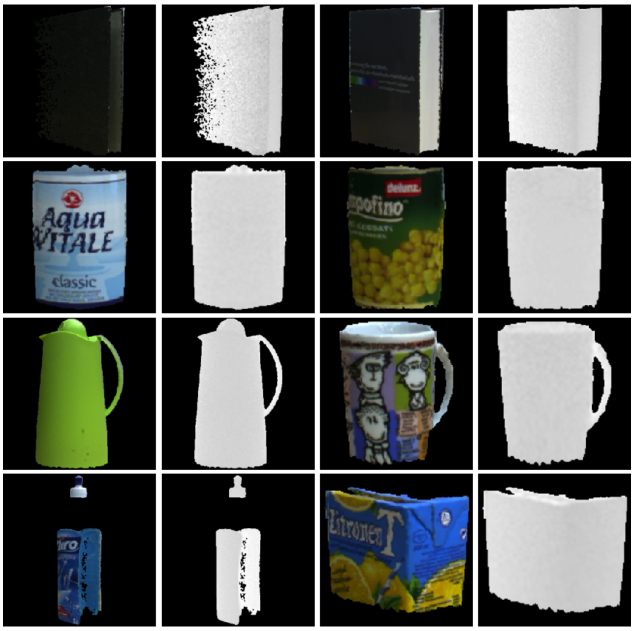

Fig. 1 Sample RGB and depth images from CIN-DB [1] dataset (8 out of 18 classes). First pair of columns top to bottom: binder, bottle, coffee pot, dish liquid. Second pair of columns: book, cans, cup, drink carton.

4. CLASSIFICATION

Our goal was to create a classifier to assign one of 18 classes to each image (view) from the testing dataset. Views of every class instance (object) are available in both training and testing sets.

For this experiment, we constructed a very simple model based on a convolutional neural network. The model consists of two convolutional and max pooling layer pairs followed by a fully connected hidden layer with ReLU (Rectified Linear Unit) nonlinearity and dropout regularization and another fully connected softmax output layer. The only part of the model that changes between the experiments is the first convolutional layer, where the convolution kernel shape must be compatible with the number of channels of the input image. Therefore, the kernel shape and, hence, the number of model parameters is adjusted depending on the format of input data. The architecture is displayed in Fig. 2. We used several regularization techniques (L2, local response normalization, dropout), however they've had minimal effect on such simple architecture. The final form of the loss function (1) consist of softmax cross entropy and λ-weighted L2 regularization factor, where N is the size of a batch and M is the number of classes.

$$L(w) = -\sum_{i=1}^{N}\sum_{k=1}^{M} y_k^{(i)} \log(\hat{y}_k^{(i)}) + \lambda ||w||_2^2 \qquad (1)$$

The model was intentionally designed to be as simple as possible to avoid overfitting for all types of input data. Failure to do so would lead to distortion of results when comparing various modalities. While the depth channel introduces additional complexity to the input data, we found the simple model is capable enough to utilize it and improve its classification accuracy.

The input resolution for the network has been set to 48x48 pixels. The first convolutional layer contains 32 filters with a size of 5x5, the second contains 64 with the same size. The fully connected layer contains 1024 hidden neurons connected to 18 neurons in the output layer. Both max pooling layers use 2x2 non-overlapping windows.

We trained the model with batch training algorithm and using Adam gradient descent optimizer. The weights and biases were initialized with normal random distribution. The dataset has been shuffled once during the preprocessing stage and the model receives the data in the same order in each epoch. This approach has simplified the tuning process, however it might be beneficial (could improve training speed) to reshuffle the dataset for each epoch separately. We kept the number of epochs low (under 100) since the small dataset size has lead to overfitting of the model even though the architecture was very simple. The model was implemented in Python using the TensorFlow framework.

5. EVALUATION

We evaluated the classification accuracy of the model by performing 5 training runs and averaging their accuracy on the testing dataset. Before each run the model's parameters were randomly initialized. In the resulting average accuracies formula (2), r stands for training run, i for testing sample index and δ is the Kronecker delta function used to compare one-hot encoded prediction and target vectors. The results for different input types are displayed in Table 1. In formula (2) N stands for training set size and $\delta_{\mathbf{y}^{(r,i)}\hat{\mathbf{y}}^{(r,i)}}$ is equal to 1 only when the true label $\mathbf{y}$ is equal to prediction $\hat{\mathbf{y}}$.

$$A = \frac{1}{5}\sum_{r=1}^{5}\frac{1}{N}\sum_{i=1}^{N}\delta_{\mathbf{y}^{(r,i)}\hat{\mathbf{y}}^{(r,i)}} \cdot 100\% \qquad (2)$$

In the confusion matrix comparison (Fig. 3) we can see that the depth modality was equally beneficial for all classes. The most notable confusions where the depth channel was beneficial were knife-fork and phone-mouse. The pairs that remained difficult to distinguish were book-binder, fork-spoon and pen-scissors. The first pair is obviously difficult to tell apart, even for a human if the resolution is low. The latter two pairs might be confused mainly due to low resolution and aggressive cropping (in some views, significant parts of the object are missing or the angle makes the object indistinguishable). Some examples of misclassified samples are shown in Fig. 4.

Another problem with the dataset, visible in Fig. 4, is that there is no way to determine the real scale of the object in the frame. This would not be a big issue if the scale was normalized, however, each image have been cropped to the bounds of the part that passed the depth segmentation. For some classes and view angles this causes the object (or a smaller part of it) seem disproportionately larger than it actually is, confusing the classifier.

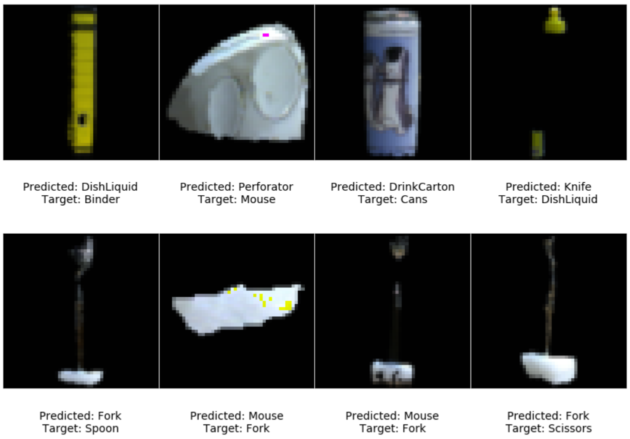

Fig. 4 Examples of misclassified samples. Utensils are the most problematic due to large parts of their shape being discarded which is caused by low resolution and noise of the depth sensor.

We speculate that the accuracy boost provided by the depth information could be even higher if different method of classification was used. While convolutional networks work very well for identifying basic shapes, textures and spatial relations, they are not very well suited for depth images. This is due to the fact that the local nature of the convolution does not take into account the spatial discontinuity of the depth images (what is close together on the depth image may be very distant in 3D). Some depth-preprocessing techniques like color-jet and surface normal coloring of the depth image [4] have been proposed and they have been proven to improve the classification accuracy. It has also been shown, however, that the fully 3D convolutional architectures are even better suited for this task, which we would like to investigate in our future work.

6. TIMING AND PERFORMANCE

Since the proposed model architecture is very simple and both the dataset size and image resolution relatively low, the training took a very short time. 30 epochs long training of the most complex version of the network (for RGB-D input) took around 1 minute, RGB and depth-only variations took slightly less time. The

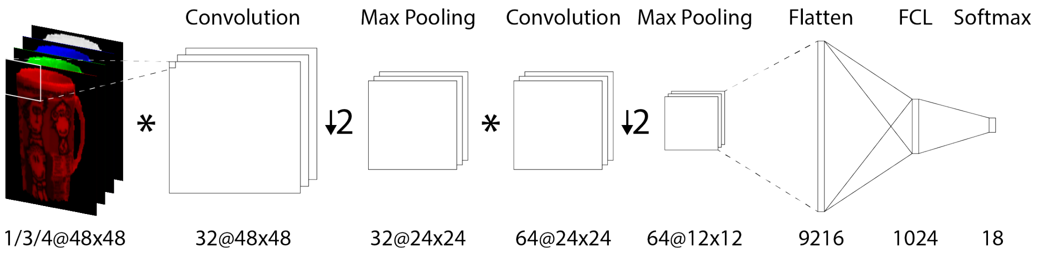

Fig. 2 Architecture of the convolutional neural network used for image classification. The network consists of two convolution-pooling pairs followed by a fully connected layer (FCL) and a softmax classifier. The input layer is parametrized can accommodate images consisting of arbitrary amount of channels (RGB, grayscale/depth, RGB-D).

inference speed in case RGB-D classifier was roughly 3300 samples per second. The resolution of input images for all modalities was 48x48 pixels. The training was performed on a workstation with Intel i7-6700K CPU and Nvidia GeForce GTX 1080 GPU.

7. CONCLUSION

The results of this study provide a reference point for evaluation of different modalities for the RGB-D image classification problem. The quantitative comparison shows significant improvements of classification accuracy over plain RGB data. Compared deep models demonstrated relative improvements in range of 3% to 10% (see Table 1). The evaluation has been performed using our CNN-based classifier along with results published in [7], [11] and [4] acquired using several different architectures. The comparison shows that the benefit of depth modality is comparable across multiple different classification methods.

The classification accuracy of our model has improved by 44.3% (grayscale to grayscale + depth) and 10.45% (RGB to RGB-D) respectively (see Table 1). ROC curves for our classifiers using various input configurations are compared in Fig. 5. These results were obtained on a dataset of common office and household objects.

Our classifier achieved comparable accuracy using depth-only and RGB data on this dataset. Some classes, especially utensils, were difficult to distinguish by depth image alone due to low resolution, noise and aggressive cropping. The performance of our model shows that the depth modality provides more significant improvement to classification accuracy than color and if the depth channel is available along with grayscale image, addition of color information does not further improve the accuracy.

Even though in our future research we are planning to focus on others forms of 3D data representation, like point clouds and voxel grids, this experiment has provided us with valuable results. It demonstrated that the 3D modality bears valuable information for classification and at the same time, the color channel provides little benefit when 3D data is available (applies to limited domain of office and household objects, which is our area of research).

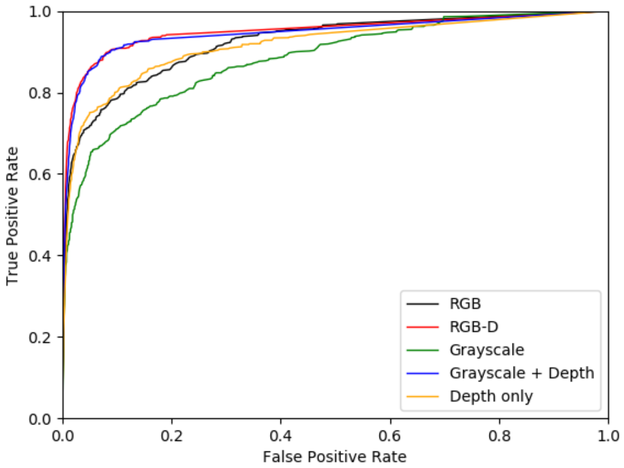

Fig. 5 ROC curves for each input format. The addition of depth data causes significant improvement of classification accuracy.

Table 1 Categorization performance of a trained model. Values for our model (bottom row) represent average accuracies of 5 runs, each consisting of random parameter initialization, training and testing. The results from [7] are retrieved on the same dataset (CIN-DB [1]) as our model, the VGG3D [11] has been evaluated on Washington RGB-D dataset [13]. Only the best performing variations from [7] were picked for the comparison.

Method	Depth	Grayscale (GS)	GS + Depth	Improvement (GS ->GS-D)	RGB	RGB-D	Improvement (RGB ->RGB-D)
DBN [7]	78.60	-	-	**-**	74.50	82.30	**10.47**
CNN [7]	83.50	-	-	**-**	79.10	84.60	**6.95**
SDAE [7]	75.60	-	-	**-**	74.50	79.30	**6.44**
VGG3D [11]	78.43	-	-	**-**	88.96	91.84	**3.24**
Multimodal DCNN [4]	85.20	-	-	**-**	87.90	92.20	**4.89**
Our CNN	63.84	53.11	76.64	**44.30**	69.55	76.82	**10.45**

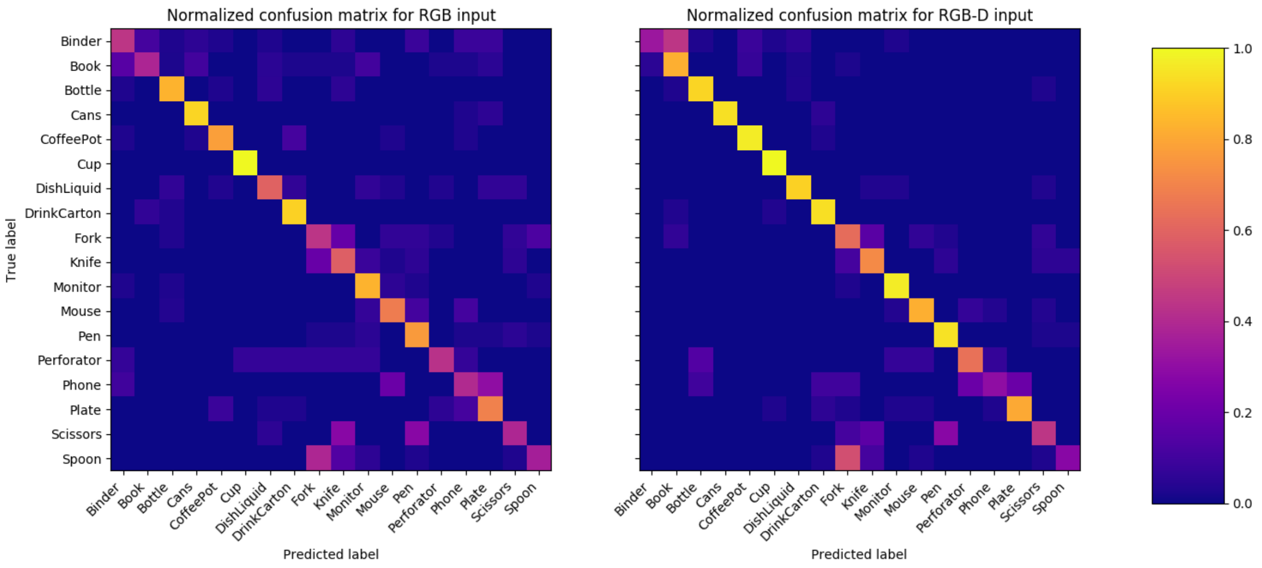

Fig. 3 Confusion matrix comparison of RGB and RGB-D input data format.

ACKNOWLEDGEMENT

This publication is the result of the Project implementation: University Science Park TECHNICOM for Innovation Applications Supported by Knowledge Technology - II. phase, ITMS: 313011D232 (40%) and ”Centre of Information and Communication Technologies for Knowledge Systems”, ITMS: 26220120020 (20%), both supported by the Operational Programme Research & Development funded by the ERDF, grant TUKE FEI-2015-33: Research Laboratory of Nonlinear Underactuated Systems (30%) and grant KEGA 001TUKE-4/2015 (10%).

REFERENCES

[1] B. BROWATZKI, J. FISCHER, B. GRAF, H. H. BÜLTHOFF, and C. WALLRAVEN, “Going into depth: Evaluating 2d and 3d cues for object classification on a new, large-scale object dataset,” in *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*. IEEE, 2011, pp. 1189–1195.

[2] M. SCHWARZ, H. SCHULZ, and S. BEHNKE, “Rgb-d object recognition and pose estimation based on pre-trained convolutional neural network features,” in *Robotics and Automation (ICRA), 2015 IEEE International Conference on*. IEEE, 2015, pp. 1329–1335.

[3] A. EITEL, J. T. SPRINGENBERG, L. SPINELLO, M. RIEDMILLER, and W. BURGARD, “Multimodal deep learning for robust rgb-d object recognition,” in *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*. IEEE, 2015, pp. 681–687.

[4] M. M. RAHMAN, Y. TAN, J. XUE, and K. LU, “Rgb-d object recognition with multimodal deep convolutional neural networks,” in *Multimedia and Expo (ICME), 2017 IEEE International Conference on*. IEEE, 2017, pp. 991–996.

[5] C. SZEGEDY, W. LIU, Y. JIA, P. SERMANET, S. REED, D. ANGUELOV, D. ERHAN, V. VANHOUCKE, A. RABINOVICH *et al.*, “Going deeper with convolutions.” Cvpr, 2015.

[6] J. DONAHUE, “Caffenet,” https://github.com/BVLC/caffe/tree/master/models/bvlc_reference_caffenet, 2016.

[7] L. SHAO, Z. CAI, L. LIU, and K. LU, “Performance evaluation of deep feature learning for rgb-d image/video classification,” *Information Sciences*, vol. 385, p. 266–283, 2017.

[8] Y. MA, B. ZHENG, Y. GUO, Y. LEI, and J. ZHANG, “Boosting multi-view convolutional neural networks for 3d object recognition via view saliency,” in *The 12th Conference on Application of Image and Graphics Technology (IGTA)*, 2017.

[9] Z. WU, S. SONG, A. KHOSLA, F. YU, L. ZHANG, X. TANG, and J. XIAO, “3d shapenets: A deep representation for volumetric shapes,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1912–1920.

[10] D. MATURANA and S. SCHERER, “Voxnet: A 3d convolutional neural network for real-time object recognition,” in *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*. IEEE, 2015, pp. 922–928.

[11] S. ZIA, B. YUKSEL, D. YURET, and Y. YEMEZ, “Rgb-d object recognition using deep convolutional neural networks,” in *2017 IEEE International Conference on Computer Vision Workshop (ICCVW)*. IEEE, 2017, pp. 887–894.

[12] A. BROCK, T. LIM, J. RITCHIE, and N. WESTON, “Generative and discriminative voxel mod-

eling with convolutional neural networks," *arXiv preprint arXiv:1608.04236*, 2016.

[13] K. LAI, L. BO, X. REN, and D. FOX, "A large-scale hierarchical multi-view rgb-d object dataset," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*. IEEE, 2011, pp. 1817–1824.

Received February 19, 2018, accepted November 15, 2018

BIOGRAPHIES

Michal Varga was born on 24. 7. 1992. In 2015 he graduated (MSc) with distinction at the Department of Cybernetics and Computers and Artificial Intelligence of the Faculty of Electrical Engineering and Informatics at Technical University in Košice. He is an external PhD candidate at the Department of Cybernetics and Computers and Artificial Intelligence. His scientific research is focused on computer vision, generative modelling and deep neural networks. In addition, he works as a software developer in healthcare industry and is specialized in development of ultrasound imaging systems.

Ján Jadlovský works at the Department of Cybernetics and Artificial intelligence of Technical University of Košice as a Assoc prof. He is a graduate of Technical University of Košice, Faculty of Electrical Engineering. In terms of pedagogy he focuses on the issues of proposal and implementation of distributed systems that control production processes. In his science-research based activities he is oriented towards distributed control systems, image recognition, complex functional diagnostics of single purpose regulators, diagnostics of production control systems, creation of information and control systems with application of the latest information technology. He is a chief executive of company KYBERNETIKA, s.r.o., Košice, that is oriented towards design engineering, implementation and operation of production and diagnostic systems in the electrical engineering, mechanical and metallurgical production.