Scene Understanding and 3D Imagination: A Comparison between Machine Learning and Human Cognition

Michael Schoosleitner[1] and Torsten Ullrich[1,2]
[1]*Institute of Computer Graphics and Knowledge Visualization, Graz University of Technology, Austria*
[2]*Fraunhofer Austria Research GmbH, Visual Computing, Austria*

Keywords: 3D Imagination, Scene Understanding, Assistance System, Computer-aided Design, Machine Learning, Computer-aided Manufacturing, Artificial Intelligence, Human Cognition.

Abstract: Spatial perception and three-dimensional imagination are important characteristics for many construction tasks in civil engineering. In order to support people in these tasks, worldwide research is being carried out on assistance systems based on machine learning and augmented reality.
In this paper, we examine the machine learning component and compare it to human performance. The test scenario is to recognize a partly-assembled model, identify its current status, i.e. the current instruction step, and to return the next step. Thus, we created a database of 2D images containing the complete set of instruction steps of the corresponding 3D model. Afterwards, we trained the deep neural network *RotationNet* with these images. Usually, the machine learning approaches are compared to each other; our contribution evaluates the machine learning results with human performance tested in a survey: in a clean-room setting the survey and RotationNet results are comparable and neither is significantly better. The real-world results show that the machine learning approaches need further improvements.

1 INTRODUCTION

Assistance systems find a multitude of applications in practically all areas of everyday life. In the context of cyber-physical systems (Tao et al., 2019), they can be used to support technicians: technical details, construction plans, manuals and other kinds of information can be displayed at the right time in their field of vision; i.e., they can have the next step of a repair directly displayed in their view. Exactly this application scenario is examined in this evaluation – a machine learning, computer-vision system shall recognize a partly-assembled model and the last, completed instruction step.

In detail, the long-term goal of this assistance system is to support a technician during the assembly of a complex device, which is manufactured in a small series or is even unique. In this setting, an augmented reality (AR) system might observe the assembly via video camera, determine the current state of the work piece (i.e., which construction step was carried out last) and display the next step of the construction manual. Consequently, the main components of the new support system are an augmented reality, head-mounted display (Kress and Cummings, 2017) with an integrated camera system (Evans et al., 2017), a database with the construction plans of a computer-aided design (CAD) model, and a machine learning component to analyze the images taken by the camera returning the current construction status.

In this paper we focus on the computer vision, machine learning component that has been trained with the assembly of a CAD model using the corresponding construction and assembly plans. The results of such a component are then compared to human performances.

2 RELATED WORK

The problem to identify the current status of an assembly (by identifying the correct step number) can be approached using 3D techniques based on depth images and reconstruction algorithms (Häne et al., 2017), or using 2D image-based methods. This distinction is blurred because 3D depth information can not only be extracted from several 2D images (Hartley and Zisserman, 2004), but can also be learned from a single 2D image (Saxena et al., 2006), (Kuznietsov

Schoosleitner, M. and Ullrich, T.
Scene Understanding and 3D Imagination: A Comparison between Machine Learning and Human Cognition.
DOI: 10.5220/0009350002310238
In *Proceedings of the 15th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2020) - Volume 2: HUCAPP*, pages 231-238
ISBN: 978-989-758-402-2; ISSN: 2184-4321

et al., 2017), (Mahjourian et al., 2018). Since machine learning approaches (if necessary) can implicitly learn the depth information, the explicit handling of depth information is not necessary with image-based approaches on a machine-learning basis.

If each intermediate step represents a separate class, the problem of visual recognition of the current state can be considered as an image classification problem (Deng et al., 2009). An overview of the state-of-the-art in machine learning in general and image classification in particular can be found in "A State-of-the-Art Survey on Deep Learning Theory and Architectures" (Alom et al., 2019).

According to the long-term goal, the new assistance system will be used in the field of mechanical engineering; due to intellectual property protection regulations, a LEGO Technic™ model of comparable complexity is used instead of real CAD data: the "Airport Rescue Vehicle" (no. 42068) consists of 1094 parts and measures over 42cm high, 45cm long and 15cm wide. The corresponding CAD model has a sequence of 137 construction steps and has been authored by PHILIPPE HURBAIN. It is published under the license CCAL VERSION 2.0 at LDRAW.ORG. Figure 1 (left) shows a semi-transparent rendering of the model including its inner parts.

In order to learn the construction of this CAD model, a multi-view convolutional neural network (MVCNN) is used. The method used for the practical implementation is called RotationNet (Kanezaki et al., 2018). It is inspired by the concepts of MVCNN (Su et al., 2015), a multi-view convolutional neural network to classify 3D objects from 2D images, and by the pose estimation technique of "Convolutional Models for Joint Object Categorization and Pose Estimation" (Elhoseiny et al., 2016). These two concepts are combined and extended by RotationNet: MVCNN uses different view points and camera positions distributed over a sphere to get 2D images of a 3D object. These images are trained with the network structure of AlexNet (Krizhevsky et al., 2012). The MVCNN approach demands for each class that all camera positions are available as images; i.e. for every pre-defined view point all 3D objects must be captured. This is, however, hard to realize in a real-world scenario, where view positions are often limited and not precise. Therefore, RotationNet removes this limitation of MVCNN by repositioning the pooling layer and combining it with the method of (Elhoseiny et al., 2016). They propose to use object prediction and pose estimation with one 2D image as input for the classification process.

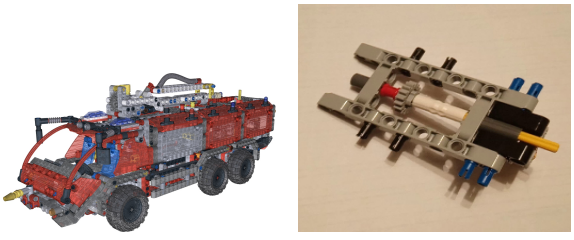

Figure 1: The test set of the new assistance system is a LEGO Technic™ model consisting of 1094 parts that are assembled in 137 construction steps. The assembly of the "Airport Rescue Vehicle" has been video recorded. While the image on the left hand side shows the complete model, the image on the right hand side shows the CAD model at an early stage.

To estimate unknown poses of a 3D object during the training process, an unsupervised pose estimation is used, which is influenced by (Zhou et al., 2017). This is a "meta" task which is conducted in every training step. Another benefit is that it is possible to predict a class with a specific set of images captured from one view position region. This is important for estimating new positions and simultaneously classifying objects. To get a low error rate, this step is of significant importance.

As mentioned above, the camera positions used in RotationNet are highly influenced by MVCNN. RotationNet uses a camera distribution on a dodecahedron for the image benchmark data sets.

3 PROPOSED METHOD

The ModelNet benchmark consists of two databases, namely ModelNet10 and ModelNet40 (Wu et al., 2015). These two databases comprehend CAD models with 10 and 40 classes. This benchmark (see `modelnet.cs.princeton.edu`) provide researchers in computer vision, computer graphics, robotics and cognitive science, with a comprehensive clean collection. Furthermore, the benchmark lists the classification accuracy of state-of-the-art algorithms. Table 1 shows a comparison of selected methods (as of September 2019). Based on this benchmark, we choose the RotationNet framework to solve the CAD classification problem; respectively, the assembly assistance task.

Due to the fact that the CAD model in the assembly assistance scenario is not similar to the existing dataset used by RotationNet at the ImageNet Large Scale Visual Recognition Competition (ILSVRC) in 2012 (Russakovsky et al., 2015), we created a new database using the LEGO Technic™ model mentioned before. The LEGO model data format (LDRAW) stores the model data as instruction steps which fit to the proposed use case. Every in-

Table 1: Selection of methods compared with the ModelNet Benchmark as of Sep. 2019. RotationNet has the best accuracy on both datasets.

Algorithm	ModelNet40	ModelNet10
RotationNet (Kanezaki et al., 2018)	97.37%	98.46%
PANORAMA-ENN (Sfikas et al., 2018)	95.56%	96.85%
MVCNN-MultiRes (Qi et al., 2016)	91.40%	–
VRN-Ensemble (Brock et al., 2016)	95.54%	97.14%

struction step corresponds to one object class in RotationNet. In total, there are 137 steps; in every class 20 different 2D images are rendered from spherical distributed camera positions on a dodecahedron as shown in Figure 2. The RGB color space is used for the images without any additional textured background. In this way, our test differs from (Kanezaki et al., 2018) as they use gray-scaled 3D models without any background from the ModelNet database.

Furthermore, RotationNet adds a so-called *incorrect view* class for a better stability of the unsupervised pose estimation. Together with the 137 instruction step classes, the total number of classes sums up to 138.

According to the use case of an AR system the

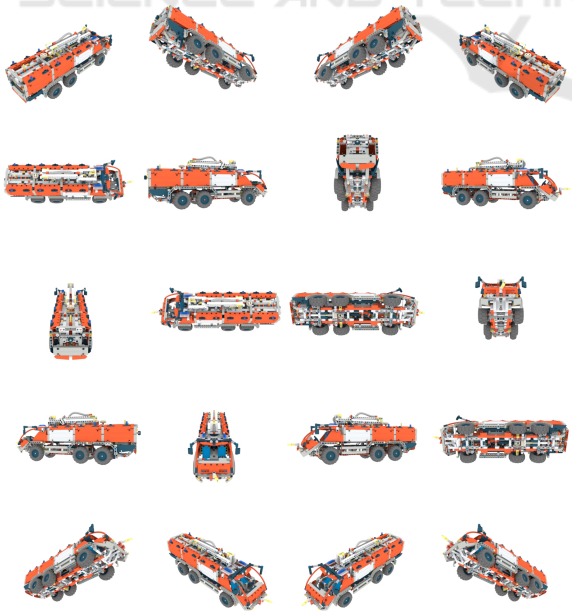

Figure 2: For each of the 137 instruction steps of the "Airport Rescue Vehicle", 20 views are rendered. The shown images depict the complete model; i.e. the result of the last instruction step.

network structure of the machine learning framework has been adopted: the input layer is extended to fit our model database, which consists of 137 classes and one additional class for the *incorrect view*. The output layer supports all numbers of classes and 20 views per class which results in 2 760 output parameters. The hidden layers and the order remain. (Kanezaki et al., 2018) compare different CNN structures on their accuracy using the ModelNet database. The main results are listed in Table 2. The best compromise between accuracy, memory size and number of parameters is AlexNet (Krizhevsky et al., 2012). Its accuracy is 1% lower than the best one but uses fewer parameters and has less memory consumption.

Table 2: Comparison of four different algorithms, which are used by RotationNet on ModelNet10 and ModelNet40 database.

Algorithm	Parameters	Memory	Accuracy
AlexNet (Krizhevsky et al., 2012)	60.2M	1.8GB	96.4%
VGG-M (Chatfield et al., 2014)	102.2M	5.3GB	97.4%
ResNet-50 (He et al., 2016)	24.2M	7.1GB	96.9%
ResNet-18 (He et al., 2016)	11.6M	2.5GB	96.0%

The learning rate and momentum for the training process in our setting are the same as published in RotationNet. Without using any GPUs the training process with our database takes about two days using 54 Intel Xeon® CPUs at 2.60GHz with batch size 52.

The network output layer values are the probabilities for every class, including the *incorrect view* and all pose estimations per class. The best view position is chosen by taking the highest probability value of a class without the *incorrect view*; i.e., inter-class probabilities are taken into account. This means not only one specific class is used to predict the pose, but more than one can be used for one prediction result.

The decision, which class matches best is calculated by the maximum value of the probability product of the views and classes. The product of all view probabilities gives the prediction for the class and its viewpoint; i.e., the prediction of the class and pose is a probability maximization over a set of images.

4 EVALUATION

In order to evaluate the machine learning system, three experiments have been designed using different

test sets; i.e. with different views on the "Airport Rescue Vehicle" CAD model. In each test, a set of images form the input for the classification task. Every set merely contains images from one class, one instruction step respectively, with different, known and unknown view positions.

4.1 Training Positions

In the first test series, all the images that have already been used to train the system are reused to test the system. In detail, for each class (137 in total) the 20 images of the pre-defined dodecahedron camera positions (see Figure 2) are used, and the system should return the correct class (main objective) and the correct pose (secondary objective).

4.2 Unknown Positions

The second test series uses the same CAD model with new camera positions. 24 view positions, which are not included in the training set, are distributed equally on a sphere according to the distribution suggested by (Schinko et al., 2011). In other words, the test set is disjunct to the training set.

4.3 Real Images

The final test series consists of real-life captured images. Using a video camera a sequence of images is captured of one instruction step and the sequential frames are used as input for the trained network. Each image has been converted to RGB color space, cropped to aspect ratio $1:1$, and re-sampled to 256×256 pixels to meet the requirements of the system for input images. Figure 1 (right) shows an example frame, which has been extracted from a video at an early stage of the assembly.

4.4 Results

For every experiment the success rates for the correctly identified construction step is referred to as Top-1; the correct construction step within the classification set with the five highest probabilities is referred to as Top-5. The results of our experiments are listed in Table 3.

The Top-1 success rate of the test series using images already used to train the system is 8.03%. The Top-5 success rate of this test series is 27,74%, which is a rise compared to the Top-1 rate by a factor of 3.45. If the test series comprehends newly rendered images exclusively used for testing purposes with unknown positions the success rates drop to 4.38% for Top-1 and to 7.30% for Top-5, respectively. In the real-world scenario, the success rates drop to 0.73% in both categories, Top-1 and Top-5. A detailed analysis of these results reveals several challenges.

Table 3: Top-1 and Top-5 accuracy rates of the three test series with (1) rendered images already used for training, with (2) newly rendered images exclusively used for testing purposes, and with (3) real images captured using a video camera.

Test Set	Top-1 Accuracy	Top-5 Accuracy
Training Positions (see Section 4.1)	8.03%	27.74%
Unknown Positions (see Section 4.2)	4.38%	7.30%
Real Images (see Section 4.3)	0.73%	0.73%

The first problem is the strong reduction of the success rate due to external influences. The real video sequences do not only show the CAD model, but sometimes also not yet assembled parts, packaging material, and everyday objects that happen to be in the video. These things have not been learned and therefore lead to false classification results. A reduction of external influences significantly improves the result; nevertheless, deviations in the video image from the trained data remain: light settings, shadows, distortion, etc.

The second problem is the homogeneous data space. The homogeneity can be illustrated by two Figures: Figure 3 shows the ImageNet benchmark and Figure 4 shows the instructions to assemble the CAD model.

The differences between any two images of the ImageNet benchmark (Deng et al., 2009) are much greater than the differences in the construction manual; there, only a few pixels change per class, and depending on the view point even no pixels may change at all. This degree of homogeneity is also a challenge for humans.

Furthermore, this problem is intensified by the used metric. From an application point of view, the subdivision into 137 classes may seem reasonable due

Figure 3: A random selection of two root-to-leaf branches of the ImageNet benchmark. *Image source: (Deng et al., 2009).*

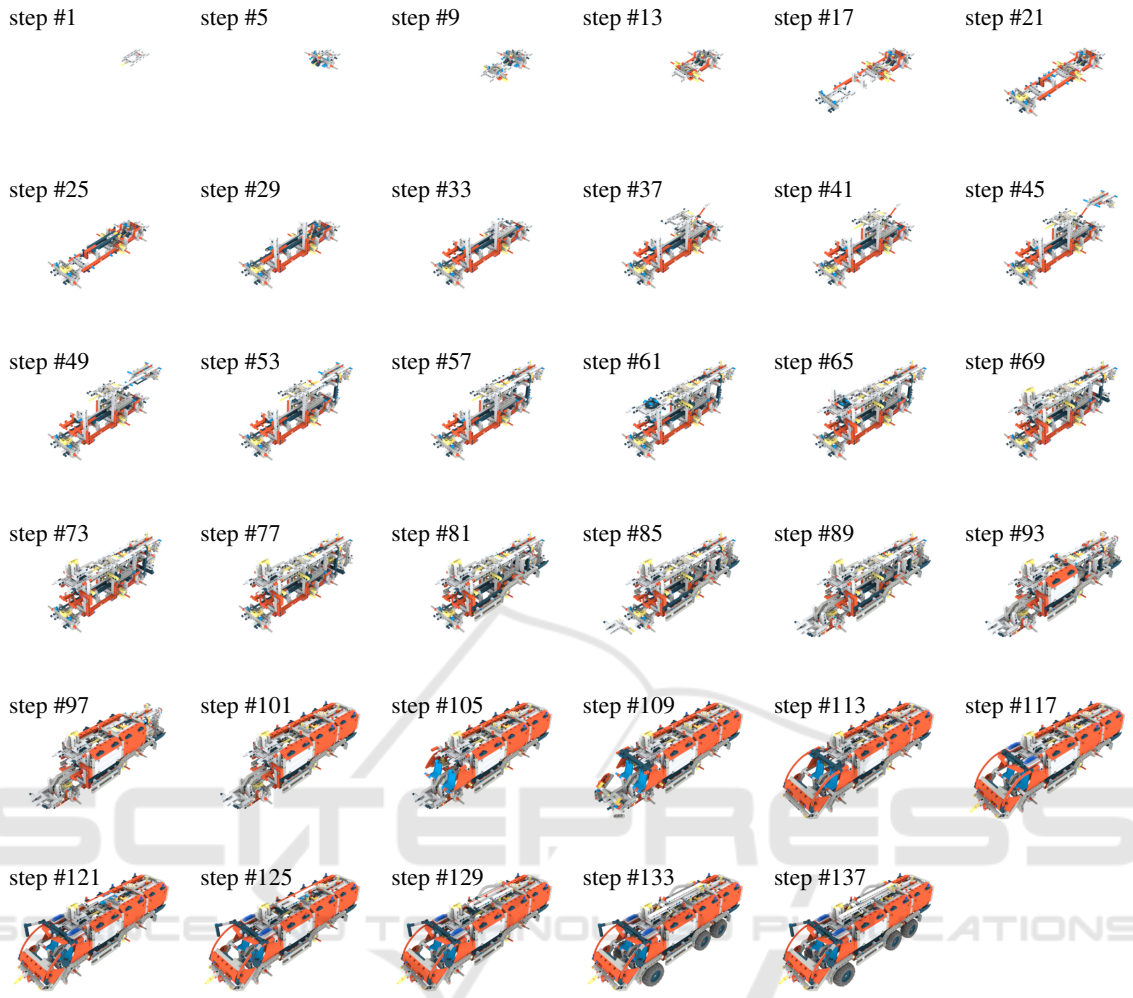

Figure 4: The construction of the "Airport Rescue Vehicle" comprehends 137 instruction steps. This overview shows the result after every fourth instruction step. All views are rendered with the same camera perspective. Differences between individual steps are not always apparent from all views.

to the corresponding number of construction steps; from a machine learning point of view, this large number of classes with marginal differences makes little sense. For example, the construction steps #69 to #77 show only minimal visual changes; even if the image classification returns all Top-5 hits within this interval, in 4/9 (44%) of the cases the correct result is not present.

5 SURVEY

In order to interpret the gained results appropriately, a survey has been performed to compare its performance with humans. The intention is to have a reference success rate on how well humans estimate the presented instruction step. The survey results are then compared with the experiments described in the Sections 4.1 and 4.2. Due to the metric problem mentioned above, we measure the distance (number of steps) between the predicted result and the ground truth.

5.1 Set-up

The survey consists of a simple questionnaire containing two parts. The first part is a simplified construction manual similar to Figure 4. It comprehends $M = 40$ different 2D images rendered from one specific camera position. The camera position is the same for all 40 images. As in Figure 4, the views were selected at equal distances between the construction steps. The second part is the answer sheet with $Q = 16$ different 2D images at different assembly stages with

Table 4: This overview lists the results of the machine learning approach based on RotationNet compared to test persons as assessed in the survey. It shows the error distribution measured as deviation between the correct instruction step and the estimated instruction step by the machine learning system resp. the guessed instruction step in the survey.

Test Set	Minimum	First Quartile	Median	Third Quartile	Maximum	Mean	Standard Deviation
Training Positions (test size $n = 2740$)	0	0	1	4	24	3.219	4.828
Unknown Positions (test size $n = 2740$)	0	3	8	16	36	10.175	8.875
Survey (test size $n = 1056$)	0	0	1	5	56	4.329	7.108

varying camera positions. The stages and the camera view points have been chosen randomly.

All participants were asked to find the best correspondence between the 16 question images to the 40 construction images. A single choice had to be made for each question, whereas the same answer could be given multiple times on different questions. There has been no time limit to answer the questionnaire.

5.2 Evaluation

The survey was printed on paper in high quality and in high resolution. They have been completed by employees and members of the institute as well as by master students. $N = 69$ attendees took part in the survey; from these, $A = 66$ completed and fully filled out questionnaire forms were returned. Three questionnaires were not completed in full or did not return at all. As a consequence, the return rate is $r = 95.7\%$.

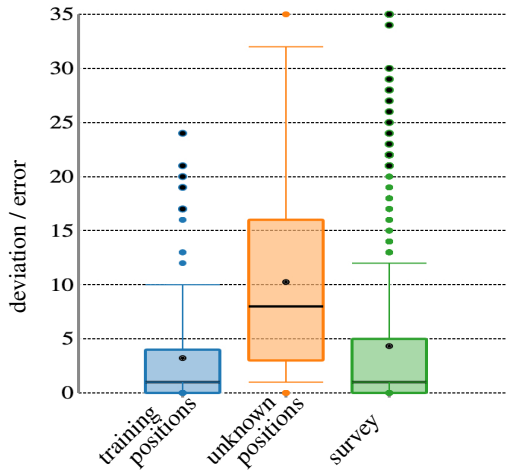

Figure 5: The error distribution as listed in Table 4 as Box-and-Whisker plot. The many outliers in the survey show that some people have clear problems with the 3D imagination task.

The results of the machine-learning-based approach using RotationNet and the survey are compared to each other: the results of the configurations described in Section 4.1 and in Section 4.2 are referred to as *Training Positions* and *Unknown Positions*, respectively; the survey results are referred to as *Survey*. All results are listed in Table 4 and visualized in a Box-and-Whisker plot in Figure 5.

With real images the RotationNet algorithm has an accuracy on the scale of a random process. Using rendered images, the accuracy improves significantly (see Table 3). As a consequence, only the tests with rendered images are analyzed further: testing the machine learning system with images already used during the training phase, the system achieves an error of 3.219 on average; i.e. the prediction of the instruction step is on average 3.219 steps off. In case of new images, which are unknown to the trained system, the error rises to a difference of 10.175 steps on average. The average human error measured by the survey is 4.329 steps; i.e. the human performance is clearly better than the machine learning system with untrained images. With trained images, the machine learning system is slightly better. However, the improvement is not significant: since both data sets do not follow a normal distribution, and no common distribution can be assumed either (according to Kolmogorov-Smirnov tests), we determine the confidence intervals of the expected values according to (Oliphant, 2006). The confidence intervals of the expected errors remain disjunct up to $p = 91.9\%$ – a value, which is usually considered to be non-significant.

6 CONCLUSIONS

The aim of the paper is to test machine learning approaches represented by the best algorithm according to the ModelNet benchmark (as Sep. 2019) in a prac-

tical application in order to find out its real-life usability. For this use-case we generated a database of an instruction step-based 3D CAD model and used it to train the instructions using RotationNet. The objective is to predict the current instruction step based on a simple image of the current assembly stage. As a second means for the test, the prediction success rate of humans is tested in a survey. The combined interpretation of machine learning results and survey results reveals limited applicability of RotationNet for real-life purposes.

6.1 Lessons Learned

We have identified some challenges that RotationNet faces. A vast amount of difficulties have to do with the image resolution used. The input image resolution of RotationNet is limited to a size of 256×256 pixels. This leads to the fact that important details are hardly visible in many CAD renderings in this resolution. Furthermore, common cameras are featured with a much higher resolution and their captured images have to be scaled down for RotationNet. The down-scaling process affects the details in the target image negatively.

The viewpoints of the training data are equally distributed but the amount of the overall viewpoints is not very high. Considering the fact that arbitrary viewpoints of CAD models can be rendered with limited effort (compared to taking photographs), we suggest a higher resolution of viewpoints on a sphere. We expect that the training data per class rises and that the number of unknown positions will be reduced. This might lead to a better recognition of minimal details of the object model.

Another issue is the invisibility of certain assembly steps. The result of neighboring instruction steps look almost identical when the model is near to completion. This effect occurs when added bricks are occluded by others in the actual view.

Furthermore, when applying the machine learning system to a real-life scenario, there is always a background behind the object model. The background information includes background noise and indirect model information such as local or global illumination and shading. The used training images, however, do not contain any background information or noise at all. Currently, it is not clear how to train a network to handle background noise without having to manually capture many different backgrounds at unacceptably high costs.

Finally, the survey shows that it is difficult for the human eye to distinguish the instruction steps from each other as well and to assign them to the right image. The viewpoints have a strong influence on the detection of the right step and the low resolution makes it difficult to identify the right images.

6.2 Improvements

Using a higher resolution for the images is the most important step when improving RotationNet in order to get more detailed information of the model per image. This improvement effects the granularity of details and the distinction between the instruction steps; this means that the differences between the single instruction steps raises. In a real-world scenario the images captured by a camera have a higher resolution and must be down-scaled, which has negative influences on the preservation of details. To improve RotationNet, the input size and the parameters must be adapted to a higher resolution which needs, however, higher computational power.

The resolution of the viewpoints on the sphere is equally distributed on a dodecahedron but the amount of viewpoints is not sufficient for real-world use. A higher resolution would lead to more training data which can be learned by the system which would result in a better prediction. This improvement needs higher computational power only during training and not in the prediction phase which is a benefit for the application of the system.

An overall challenge of all methods is the influence of the background on the recognition of the model. The recognition and reduction of background information may improve the prediction. At the training it is hardly possible to know in advance how the background will look like in the productive application. The captured scene can be indoor or outdoor, with various illumination configurations. One option may be a pre-segmentation of an image and to pass only the extracted foreground to the prediction system. An advantage of this approach is to leave the network input image size untouched and crop the interesting image parts to this size. A downside is the difficulty to find the right segment within the image.

6.3 Contribution and Benefit

The presentation of the problems and the lessons learned are an important contribution. Furthermore, the provision of the LEGO^{TM} CAD data set and its renderings is a valuable benefit to the community. The CAD model is converted to a sequence of 3D models in Alias Wavefront OBJ format, and all sequence steps are available in commonly used image formats. They will be available at: `https://github.com/FhA-VC`.

ACKNOWLEDGEMENTS

The authors acknowledge the generous support of the Carinthian Government and the City of Klagenfurt within the innovation center *KI4Life*.

REFERENCES

Alom, M. Z., Taha, T. M., Yakopcic, C., Westberg, S., Sidike, P., Nasrin, M. S., Hasan, M., Van Essen, B. C., Awwal, A. A. S., and Asari, V. K. (2019). A State-of-the-art Survey on Deep Learning Theory and Architectures. *Electronics*, 8:292ff.

Brock, A., Lim, T., Ritchie, J. M., and Weston, N. (2016). Generative and Discriminative Voxel Modeling with Convolutional Neural Networks. *International Conference on Neural Information Processing Systems / 3D Deep Learning Workshop*, 30:1–9.

Chatfield, K., Simonyan, K., Vedaldi, A., and Zisserman, A. (2014). Return of the Devil in the Details: Delving Deep into Convolutional Nets. *British Machine Vision Conference*, 6:1–12.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). Imagenet: A Large-scale Hierarchical Image Database. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 12:248–255.

Elhoseiny, M., El-Gaaly, T., Bakry, A., and Elgammal, A. M. (2016). A Comparative Analysis and Study of Multiview CNN Models for Joint Object Categorization and Pose Estimation. *International Conference on Machine Learning (ICML)*, 33:888–897.

Evans, G., Miller, J., Pena, M. I., MacAllister, A., and Winer, E. (2017). Evaluating the Microsoft Hololens through an Augmented Reality Assembly Application. *Degraded Environments: Sensing, Processing, and Display (Proceedings of SPIE Defense and Security)*, 10197:1–16.

Häne, C., Zach, C., Cohen, A., and Pollefeys, M. (2017). Dense Semantic 3d Reconstruction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39:1730–1743.

Hartley, R. and Zisserman, A. (2004). *Multiple View Geometry in Computer Vision*. Cambridge University Press.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep Residual Learning for Image Recognition. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 19:770–778.

Kanezaki, A., Matsushita, Y., and Nishida, Y. (2018). RotationNet: Joint Object Categorization and Pose Estimation Using Multiviews from Unsupervised Viewpoints. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 21:5010–5019.

Kress, B. C. and Cummings, W. J. (2017). Towards the Ultimate Mixed Reality Experience: Hololens Display Architecture Choices. *Digest of Technical Papers, Society for Information Display*, 48:127–131.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). ImageNet Classification with Deep Convolutional Neural Networks. *International Conference on Neural Information Processing Systems*, 25:1097–1105.

Kuznietsov, Y., Stückler, J., and Leibe, B. (2017). Semi-supervised Deep Learning for Monocular Depth Map Prediction. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 20:2215–2223.

Mahjourian, R., Wicke, M., and Angelova, A. (2018). Unsupervised Learning of Depth and Ego-motion from Monocular Video Using 3d Geometric Constraints. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 21:5667–5675.

Oliphant, T. E. (2006). A Bayesian perspective on estimating mean, variance, and standard-deviation from data. *Brigham Young University (BYU) Faculty Publications*, 1877-438:http://hdl.lib.byu.edu/1877/438.

Qi, C. R., Su, H., Nießner, M., Dai, A., Yan, M., and Guibas, L. (2016). Volumetric and Multi-view CNNs for Object Classification on 3D Data. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 19:5648–5656.

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

Saxena, A., Chung, S. H., and Ng, A. Y. (2006). Learning Depth from Single Monocular Images. *Advances in Neural Information Processing Systems*, 18:1161–1168.

Schinko, C., Ullrich, T., and Fellner, D. W. (2011). Simple and Efficient Normal Encoding with Error Bounds. *Theory and Practice of Computer Graphics*, 29:63–66.

Sfikas, K., Pratikakis, I., and Theoharis, T. (2018). Ensemble of PANORAMA-based convolutional neural networks for 3D model classification and retrieval. *Computers & Graphics*, 71:208–218.

Su, H., Maji, S., Kalogerakis, E., and Learned-Miller, E. (2015). Multi-view Convolutional Neural Networks for 3d Shape Recognition. *IEEE International Conference on Computer Vision (ICCV)*, 11:945–953.

Tao, F., Zhang, M., and Nee, A. Y. C. (2019). *Digital Twin Driven Smart Manufacturing*. Academic Press.

Wu, Z., Song, S., Khosla, A., Yu, F., Zhang, L., Tang, X., and Xiao, J. (2015). 3d ShapeNets: A Deep Representation for Volumetric Shapes. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 18:1912–1920.

Zhou, T., Brown, M., Snavely, N., and Lowe, D. G. (2017). Unsupervised Learning of Depth and Ego-Motion from Video. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 20:1851–1860.