A REVIEW REGARDING DEEP LEARNING TECHNOLOGY IN MOBILE ROBOTS

Mihalca, V.O.
Department of Mechatronics, Faculty of Managerial and Technological Engineering, University of Oradea
ovidiu@vmihalca.ro

Birouaș, F.
Department of Mechatronics, Faculty of Managerial and Technological Engineering, University of Oradea
fbirouas@uoradea.ro

Avram, F.
Department of Mechatronics, Faculty of Managerial and Technological Engineering, University of Oradea
favram@uoradea.ro

Nilgesz, A.
Department of Mechatronics, Faculty of Managerial and Technological Engineering, University of Oradea
arnoldnilgesz@gmail.com

***Abstract* - Deep Learning usage is spread across many fields of application. This paper presents details from a selected variety of works published in recent years to illustrate the versatility of the Deep Learning techniques, their potential in current and future research and industry applications as well as their state-of-the-art status in vision tasks, where their efficiency is experimentally proven to near 100% accuracy. The presented applications range from navigation to localization, object recognition and more advanced interactions such as grasping.**

Keywords: deep learning, neural network, RNN, CNN, mobile robots

I. INTROTUCTION

Neural network systems have proven to give results in many types of applications in the last few years, becoming an essential technology in areas involving computer or machine vision. Technological advances which include cheaper hardware, better hardware performance and possibilites, as well as the rise of parallel processing on GPUs allow neural network-based architectures to be employed more commonly in both industry and IT projects, thus to be studied, optimized and refined as a current emerging technology and future base for many products and services.

Deep learning refers to the use of neural networks for learning desired features on labeled sample data, in order to be used later for identifying those features in new data. The types of neural networks usually involved have many hidden layers, being called *deep*, hence the term *Deep learning*.

II. NAVIGATION

Mobile robots, as well as industrial robots, have seen succesful use of deep learning in recent years. In [2] the authors explored an architecture with separate neural networks for perception and control. It was used for navigation purposes and tested on a mobile robot with two wheel differential drive. The model was inspired by a DQN (Deep Q-Network), but split into separate networks to create a simplified model.

The perception network is a CNN with three convolutional layers. It receives depth information as input and outputs feature representations which are fed into the control network. The latter is built using three fully-connected layers. The authors of the paper motivated the use of a second network for control in order for the robot to rapidly adapt to new environments without pre-training in the new environment. A simulated experimental setup was used involving the use of Gazebo for the 3D environment and robot, as well as a CNN-based reinforcement learning control framework.

In order to train the robot, it was made to explore the environment while being remote controlled, thus labelling the depth image acquired from the sensor with the control commands. The trained model was then used to extract feature maps in real-time from sensor data, which were the output of the last ReLU layer. Then the control network was used for estimating the Q-value like in a DQN.

The depth image in this case could be regarded as the state s_t and it would be memorized along with its action a_t, reward r and next state s_{t+1} which is a new set of feature maps extracted from the image captured after activating the action a_t.

III. LOCALIZATION

A localization problem was tackled in the works of Turan, Shabbir, Araujo, Konukoglu and Sitti[1] in a paper coming from medical uses of deep learning. The work presented an architecture based on CNN-RNN for endoscopic capsule robots, which involves a fusion of frame data and magnetic data. The conclusion of the paper emphasizes the advantage of not having to tune the parameteres of the system, only the hyperparameters.

The technique presented begins with preprocessing the captured frames to enchance vessels in the tissue, followed by keyframe selection. The selection is necessary because many similar frames are generated from the slow motion of the capsule robot inside the gastrointestinal tract.

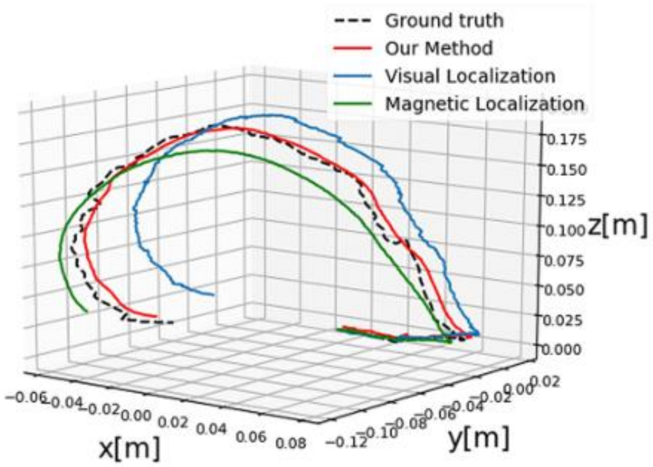
In contrast with typical Deep Learning applications, the method presented in the paper applies optical flow to consecutive keyframes, thus focusing the CNN on the dynamics between the frames instead of quasi-static data. There are several optical flow algorithms taken into account, but Farneback [3] is stated by the paper to perform best. The resultant optical flow vector needs to be quantized such thta the CNN input information is discrete. The paper presents the method of quantification, with the maximum value the diagonal length of the image resolution and the minimum value zero, followign with dividing the quantification range into 1024 intervals and concatenating the resulting x and y values for each pixel.

The capsule robots are magnetically controlled and also use a magnetic Hall sensor array for localization. The localization technique is briefly described, with a reference to [4] where it is presented in more detail.

The architecture presented in the paper is based on convolutional layers with the purpose of extracting features from quantized optical flow vector and LSTM units for fusion of flattened feature vector and magnetic localization information.

The implementation is based on Theano, using a Keras front-end which is a Python wrapper over various machine learning backends. The network was trained on an Amazon GPU compute instance.

The main evaluation system for the proposed architecture is trajectory estimation. For dataset preparation, four digital endoscopic cameras were used, along with Optitrack motion tracking system composed of eight Prime-13 cameras. The proposed system is compared to several other approaches and the trajectory close to ground-truth data demonstrates increased accuracy, as shown in the chartes presented in the paper:



IV. DETECTION & RECOGNITION

A classical problem in Machine (Computer) Vision is the Traffic Sign Recognition (TSR). Deep Learning, particularly convolutional neural nets, have become state-of-the-art in solving object detection tasks. In [6] , the authors have described an architecture which combines a convolutional neural network used for classification with other algorithms.

The paper describes the phases involved in the task: detection and localization, followed by the classification. Before the first phase, there is some image preprocessing done. Red and blue pixels are extracted from the image and this procedure generates a type of point-like noise in the images. To solve the issue, the authors propose the algorithm in paper [7] with an efficient CUDA implementation. According to the paper, when used on a NVIDIA Jetson TK1 GPU, the performance reaches 7-10 ms of preprocessing time for a videoframe, sufficient for real-time requirements.

For the detection and localization phase a modified Generalized Hough Transform (GHT) was used. The algorithm has been presented in [8] . The paper claims the algorithm is effective on the preprocessed images and tracking is improved by using the vehicle’s current speed value.

In the final phase, classification is executed using a convolutional neural network. Briefly the paper introduces the concepts of artificial neurons, artificial neural networks and the principles of the architecture of ANNs. Noteworthy the statement that “in classification problems, the most commonly used cost function is the cross entropy”:

$$H(p,q) = -\sum_{i} Y(i) log y(i)$$

Also, for the activation function, the ELU function is used to avoid the vanishing gradient problem:

$$ELU(x) = \begin{cases} exp(x) - 1, x \leq 0 \\ x, x > 0 \end{cases}$$

The implementation proposed by the paper makes use of the TensorFlow library[5] . German Traffic Sign Recognition Benchmark was used for testing purposes, as well as training the network. The paper claims that network design is heuristic in nature and there are no clear strategical recipes for creating such architectures.

Therefore, in accordance with the heuristic nature of designing CNNs, an iterative method was used, involving different layer strategies which were tested for accuracy. As an advice hailing from the paper authors, there must be an adequate correlation between network depth and the volume of data. The model gets overfitted when using a deep network and there’s little data for it, and if using a network with a small number of layers on lots of data it manifests poor accuracy.

The first attempt at a network architecture was as follows:

TABLE 1. INITIAL NN ARCHITECTURE

Layer
Convolutional, stride 2, kernel 7x7x4
Convolutional, stride 2, kernel 5x5x8
Convolutional, stride 2, kernel 3x3x16
Convolutional, stride 2, kernel 3x3x32
Convolutional, stride 1, kernel 2x2x16
Convolutional, stride 1, kernel 2x2x8
Convolutional, stride 1, kernel 2x2x4
Fully connected-64
Fully connected-16
Softmax

In the last softmax layer the output is normalized and the classification occurs (its output are probabilities for each of the classes involved). Following this architecture was a second, simplified one:

TABLE 2. SECOND NN ARCHITECTURE

Layer
Convolutional, stride 2, kernel 3x3x16
Fully connected-512
Softmax

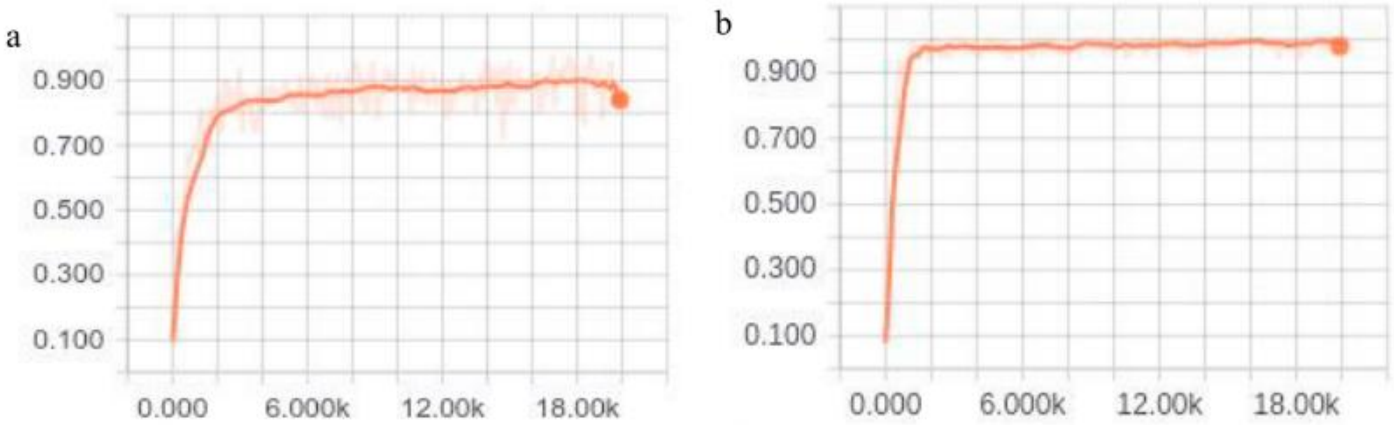
The accuracy of the single-layer architecture does not approach the previous one, therefore an architecture with additional layers is devised as the third and final one:

TABLE 3. THIRD (FINAL) NN ARCHITECTURE

Layer
Convolutional, stride 2, kernel 3x3x16

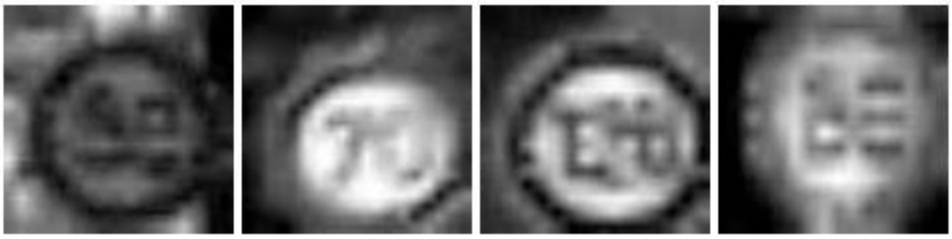
Convolutional, stride 2, kernel 3x3x32
Convolutional, stride 2, kernel 3x3x64
Fully connected-512
Softmax

Model training was done with 80% of the dataset. The rest of 20% was used for testing. While training, every iteration consisted of 50 processed images in a single batch. After 100 iterations, the accuracy would be computed using a batch of 50 images from the test dataset. A final measurement of accuracy was conducted after the training was complete, with all test images. The following figure illustrates the evolution of accuracy:



The paper tests the accuracy of the proposed solution in all phases using the GTSDB and the GTSRB datasets. It claims an accuracy of 99.94% in detecting and localizing traffic signs out of the 9987 images used. Results and efficiency are compared with other publications as well, so that the authors conclude among the top techniques for TSR is using CNNs for both detection and classification, with an accuracy of 99.89% for detection and 99.55% for classification. The method is described in [9] .

The paper presented here also shows an image with unsuccesful classification because of low quality of images:



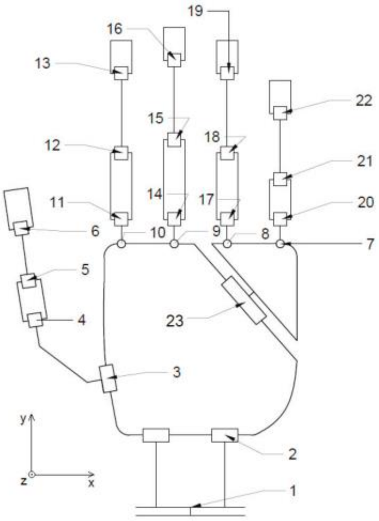
It underlines that recognition rate of success is greatly influenced by the quality of the inputs, across all algorithms that were studied.

Finally, in the conclusion of the paper it states again the achieved accuracy of 99.94% correctly classified images, as well as its trait of being capable of real-time video processing and recognition of trained-for traffic signs. As a future direction for the paper, it proposes using a CNN for both detection and classification and to extend the existing CNN to recognize more traffic sign classes.

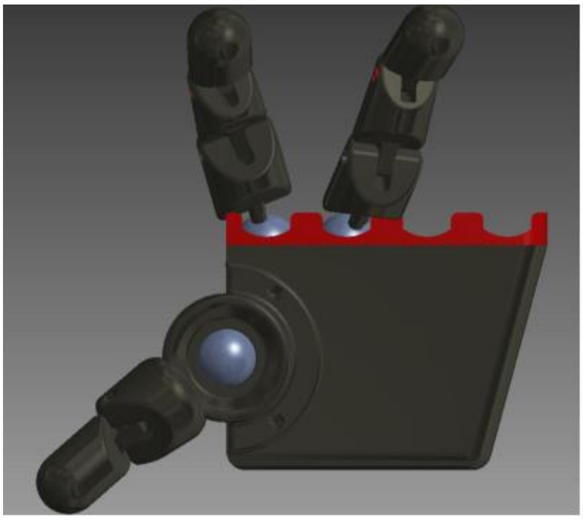
V. GRASPING & HCI

Hailing from the domain of robotics involved with creating agents that can one day operate among humans as naturally as possible, the intention of the paper is to present a model of hand that employs deep learning for grasping functionality.

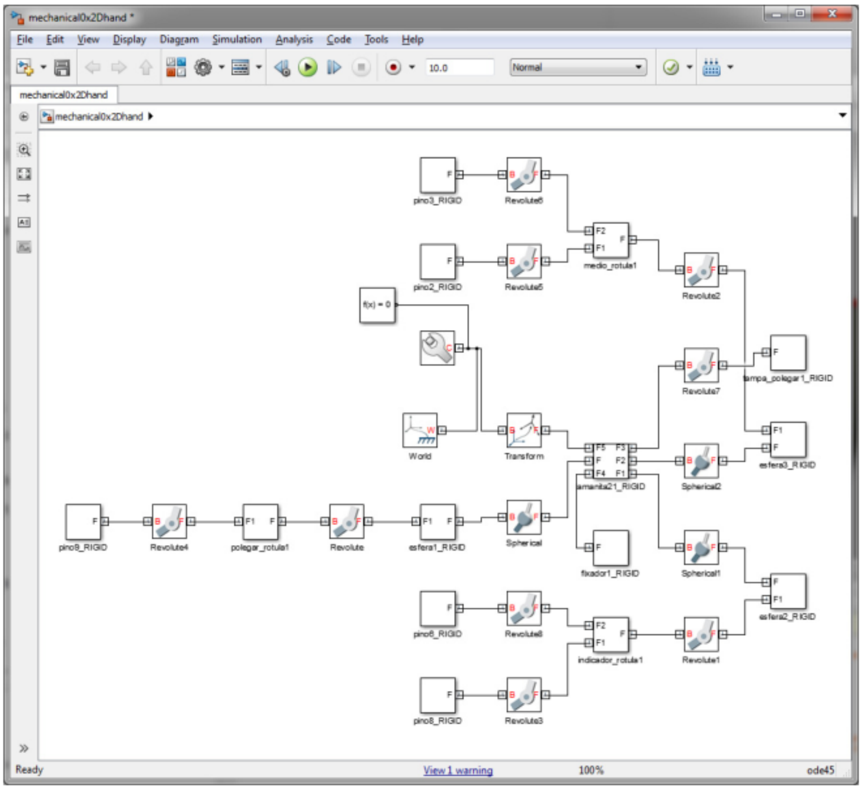
Therefore, Bezak, Bozek and Nikitin propose in [10] a kinematics model for a robot hand with 23 degrees of freedom. The schematic for the model is as follows:



The kinematic analysis of the model follows. Afterwards, the paper proposes a method of implementation based on using a software pipeline consisting of CAD and Matlab with Simulink. A simplified, three-fingered model is created in Autodesk Inventor, with the model then exported to Simulink through the use of the tool smlink_linkinv. The CAD model is presented in the following image from the paper:



The Simulink model obtained needs to be adjusted in order to represent the desired functionality. Following is a screenshot of the Simulink model as presented in the paper:



It is asserted in the paper that for as higher accuracy of a manipulation task as possible it is necessary to maintain a strict control of the interaction between the system and the target object. Therefore, for an efficient grasping, the system requires visual object detection. An introductory section on ANNs follows, as well as Deep Learning concepts, laying the ground for them as proposed solution to the vision problem.

For demonstration of funcitonality, a simulation environment consisting of MATLAB SimMechanics is used. The scene contains the model of the robotic hand along with objects and simulated camera. The objects are recognized by the vision system, which is based upon the MATLAB Computer Vision Toolbox having an implementation of a CNN.

VI. CONCLUSION

In this paper, multiple selected publications were presented, coming from diverse fields of applications of Deep Learning such as navigation, localization, object detection & recognition. Its purpose was to prove the versatility of Deep Learning and how it lends to solving labelling or characterization tasks of input data.

Their generic nature together with the ability to create a model by iterating over data samples make them an efficient tool used both standalone as well as in conjuction with other algorithms to enable key functionality in robots of the future. Deep Learning applications have already become commonplace and the recent experiments (such as those presented in this paper) prove that neural network-based architectures are a de-facto solution to vision tasks and many vital and expected features of mobile systems: navigation, localization, detection and advanced interactions.

Acknowledgements

Writing of this paper made possible by the University of Oradea’s Department of Mechatronics support for research regarding mobile robot systems and Artificial Intelligence together with Machine Vision methods and techniques applied to such mechatronic systems.

References

[1] Turan, M.; Shabbir, J.; Araujo, H.; Konukoglu, E. and Sitti, M. (2017). *A deep learning based fusion of RGB camera information and magnetic localization information for endoscopic capsule robots*, International Journal of Intelligent Robotics and Applications 1 : 442-450.
[2] Tai, L. and Liu, M. (2016). *Mobile robots exploration through cnn-based reinforcement learning*, Robotics and Biomimetics 3.
[3] Farnebäck, G. (2003). *Two-frame motion estimation based on polynomial expansion*, Image Analysis. Lecture Notes in Computer Science .
[4] Son, D.; Yim, S. and Sitti, M. (2016). *A 5-d localization method for a magnetically manipulated untethered robot using a 2-d array of hall-effect sensors.*, IEEE/ASME Trans. Mechatron. .
[5] Abadi, M.; Agarwal, A.; Barham, P.; Brevdo, E.; Chen, Z.; Citro, C.; Corrado, G. S.; Davis, A.; Dean, J.; Devin, M.; Ghemawat, S.; Goodfellow, I.; Harp, A.; Irving, G.; Isard, M.; Jia, Y.; Jozefowicz, R.; Kaiser, L.; Kudlur, M.; Levenberg, J.; Mané, D.; Monga, R.; Moore, S.; Murray, D.; Olah, C.; Schuster, M.; Shlens, J.; Steiner, B.; Sutskever, I.; Talwar, K.; Tucker, P.; Vanhoucke, V.; Vasudevan, V.; Viégas, F.; Vinyals, O.; Warden, P.; Wattenberg, M.; Wicke, M.; Yu, Y. and Zheng, X. (2015). *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*, .
[6] Shustanov, A. and Yakimov, P. (2017). *CNN Design for Real-Time Traffic Sign Recognition*, Procedia Engineering 201 : 718-725.
[7] Fursov, V.; Bibkov, S. and Yakimov, P. (2013). *Localization of objects contours with different scales in images using Hough transform [in Russian]*, Computer Optics. .
[8] Yakimov, P. (2015). *Tracking traffic signs in video sequences based on a vehicle velocity [in Russian]*, Computer Optics. .
[9] Aghdam, H.; Heravi, E. and Puig, D. (2016). *A practical approach for detection and classification of traffic signs using Convolutional Neural Networks*, Robotics and Autonomous Systems .
[10] Bezak, P.; Bozek, P. and Nikitin, Y. (2014). *Advanced Robotic Grasping System Using Deep Learning*, Procedia Engineering 96 : 10-20.