Implementation of Artificial Intelligence Methods for Virtual Reality Solutions: a Review of the Literature

Rytis Augustauskas
Department of Automation
Kaunas University of Technology,
Kaunas, Lithuania
rytis.augustauskas@ktu.edu

Aurimas Kudarauskas
Department of Automation
Kaunas University of Technology,
Kaunas, Lithuania
aurimas.kudarauskas@ktu.edu

Cenker Canbulut
Department of Multimedia Engineering
Kaunas University of Technology,
Kaunas, Lithuania
cenker.canbulut@ktu.edu

***Abstract*—Today, Artificial Intelligence (AI) used widely in data science and computer vision. It has proven to be state of the art algorithm for classification tasks. One of the tasks that Virtual reality often solves can be specified as object recognition or classification. These types of tasks benefit from automatic feature detector provided by convolutional neural networks (CNN). This article investigates and provides a practical guide on implementing AI methods for object recognition and skeleton recognition to show practical solutions on the given tasks for Virtual Reality.**

Keywords—CNN; Neural network; VR; AI; Image processing, object recognition.

I. INTRODUCTION

Nowadays deep learning is new and hot topic. Researches done in AI field can provide satisfactory solutions for object detection, image classification, natural language processing and many other areas corresponding the use of AI. One of the biggest field of deep learning utilization is computer vision. Advanced artificial intelligence methods can detect object, understand person movement, interpret gestures or behavior using RGB and depth data. Modern sensors, such as *Microsoft Kinect, Leap Motion, Intel Real Sense* or any other unmentioned here, can help to extract visual information about the scene context. Machine could be made to "understand" the scene without including other sensors, but only with a visual data (RGB and depth map). It is also more natural way of interaction in case of understanding gestures and pose, because no other input device, such as, joystick is involved.

In this paper, we are making an overview of the newest researches for deep learning utilization in Virtual Reality field by mentioning data preprocessing, gestures recognition, pose estimation methods based on neural networks. We used articles from IEEE database due to high article acceptance requirements to the database. AI theme is very popular, so we included only the articles written over the last two years except for few written in last three years. The exception was made due to the impact that the articles made to the industry.

II. OVERVIEW

The following overview of literature is organized in sections. Sections II-A to II-C overviews generic problems related to application of neural networks for problem solving. Part II-D describes latest methods on object detection related to person tracking. Section II-E covers state of art methods of pose and hand keypoints estimation and gestures recognition done by using deep neural networks.

A. *Training dataset*

When you are working with neural networks training data set is a must since you can rely the solution in given standardization. This task can be very labor intensive. The best tradeoff between information provided to algorithm and time needed for marking is bounding box method [1]. Also, it is possible to minimize labor time by utilizing internet generated data, but which also must be filtered well [2]. When problem does not have strict classes for objects, it is possible to use automate class generation algorithm to remove time needed for database preparation [3]. If you are working with specific objects and there is no large dataset, the neural network can be pretrained on training dataset of similar nature [4]. Also, it is worth noting, that only larger networks will benefit from more detail input data [5].

B. *Data preprocessing*

With wide variety of sensors used for collecting data it is hard to have normalized data. Also, different sensors require different filtering. If working with different image sensors there are great median filter [4], or filters based on neural network [5][6]. When working with moving depth sensors you can get ghosting effect in data clouds. Inaccurate data can be filtered by utilizing segmentation of data cloud with convolutional neural network. [7]

When images are used as data, you must compensate the differences of object sizes. Usually, dedicated neural networks are used to generate regions of picture that are most likely to contain object [8][9]. When you already have regions of interests (ROI) you can make few iterations with different scales over the ROI, so you could extract small objects [10].

Other way is to use few neural networks optimized for different scales [11]. Also, it is important to note that shallow CNN perform better with small objects then deep ones due to the information lost in convolution layers [12]. The extraction of most distinctive features can be improved by the regularizations of a spatial transformation branch and a Fisher encoding based multi modal fusion branch. [13]. Other great approach to solve small scale object detection problem is usage of atrous convolutions, these convolutions adapt to different input sizes and have constant output size [14].

C. Optimization of CNN

Usually training of convolutional neural network (CNN) can take a lot of computational power. This can be reduced by restructuring layer of CNN [15]. Also, it is possible to reduce computational needs of algorithm by removing background information from input data [16]. When the CNN algorithm is optimized towards execution speed you should reduce parallel operations and use larger feature maps or combine feature maps of two different convolutional layers [17][18][19]. If working with ROI, you can increase algorithm speed by implementing cascade filtering algorithm [20]. Also, the search of ROI can be improved by combining convolutional layer map with edge map of the same image [7]. Computing the same algorithm with different image scales takes a lot of time. It is possible to use one scale feature map and calculate the feature maps of different scales. This improves the ability to detect small objects and reduce required computation time [21].

It is possible to minimize selection of CNN architecture time by utilizing performance index calculation method [22].

D. Object detection using CNN

Approach using Convolutional Neural Network for object recognition where, it can be human, hand or any other object to define, gives another alternative on problem solving of object detection. Research made in Madrid proposes deep learning-based approach using CNN with the combination of Long Short-Term Memory (LSTM) method to recognize skeleton-based human activity and hand gesture [23]. There are many approaches to problem solution based on recognition of human skeleton [24]. However, success of deep learning techniques started around 2012. Proposed research relies on CNN with combination of LSTM. As we know CNNs are structured to explore high spatial local correlation patterns in images. In this approach, CNN focuses on position patterns of skeleton joints in 3D space and after LSTM recurrent network is used to capture spatiotemporal patterns related to the time evolution of 3D coordinates of the skeleton joints. Proposed approach has input data structure arranged in three-dimensional block where each dimension of this block matches with the number of skeleton joints, J.

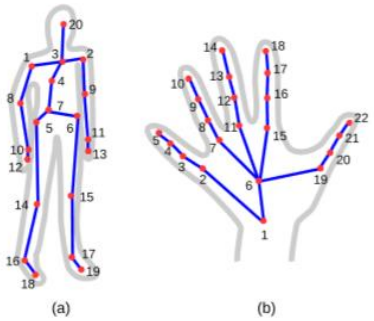

Figure 1 - Example of a full-body skeleton (20 joints) and a hand skeleton (22 points) [23].

In the case of **Figure 1**, for full-body skeleton input data will be composed as 20 joints with T time steps and three spatial coordinates. In hand skeleton case it is 22 joints composed to T time and three spatial coordinates. At each time step block is shifted one frame releasing the oldest and including new one in overlapping mode. In this approach, CNN is performed on 3D information of data and some temporal dimension (T time steps) to generate the features detected in the input block. Later, LSTM is used to integrate features detected in the consecutive overlapping blocks which allows system to maintain information beyond the last T time steps. More information about the LSTM can be found in [25]. Structured combined CNN and LMO for training is shown in Figure 2.

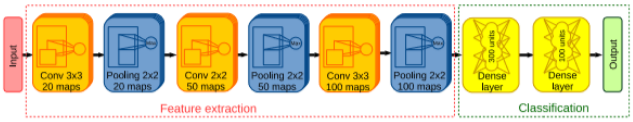

Figure 2 - The structure of the network during the pre-training stage consists of a CNN attached to a LSTM [23].

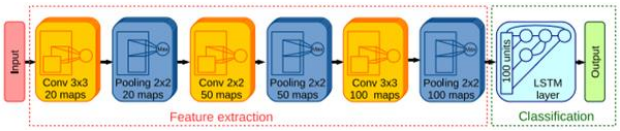

Figure 3 - The structure of the network during the final stage consists of a CNN attached to a LSTM [23].

During the experimental work there are 5 different datasets are used as follows MSRDailtActivity3D dataset, UIKinect-Action3D dataset, NTU RGB+D dataset, Montalbano V2 datasets, Dynamic hand gesture (DHG-14/28) dataset. Each dataset come with different activities to perform as some has human-object interactions as well. As each dataset can act different on the proposed method, it is necessary to keep recent and most used datasets to test the given method for validity. The capability of skeleton tracking is also depending on the hardware architecture where the proposed method is performed. Said that, for smooth recognition and overall tracking performance, the hardware system of the computer should be taken into account. As this method may require data augmentation since the data taken

from the datasets might be very small or different in terms of color aspects, it gives good results in terms of human gesture recognition desired to be tracked or captured. As a result, it might be very good alternative to recognize full human body skeleton and hand gestures using CNN with LSTM combination.

Object detection problem can be solved by iterative CNN. The image is split in equal boxes that has the class to be search assigned to each box. Step by step the bounding boxes are moved toward the candidate of the class the box should be bounding. After some iterations box ends up bounding the object that it was searching. This method can increase detection speed by five times compared to Fast R-CNN. [26]

CNN performs well on image classification problem, although object detection problem requires to extract special information of object. By introducing feature maps that also includes the spacing of the feature it is possible to increase both speed and accuracy of object detection compared to Fast R-CNN. [27]

Typically, object recognition is performed on 2D objects. It is possible to perform object detection from 3D points cloud. You can achieve great accuracy by making three projections of object and utilizing three CNN networks for classification. [9]

Some objects have wide variety of features depending on viewing angle. Monolitick neural networks has problems to detect objects of widely diverse categories. Introduction of subcategories (S-CNN) improves the performance of object recognition in such situations. [28]

Combination of image sensors feature map and depth sensors feature map can introduce impressive results. Obtained feature maps can be classified by support vector machine or mondrian forest algorithms. [10]

Experiments performed to test the effectiveness of the proposed approach in terms of detecting a specific gesture of the user. The training was performed on high computational power PC with specifications of GPU using an Intel Xeon E5-1620v3 server clocked at 3.5 GHz with 16 GB RAM and a 2015 NVIDIA GeForce GTX TITAN Black GPU with 6 GB of GDDR5 memory and 2880CUDA cores.

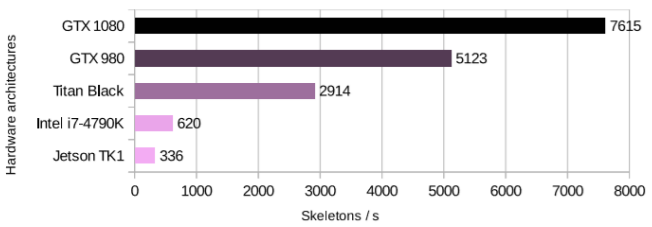

Figure 4 - Skeleton count per second of each GPU processor

Execution of experiment methodology performed by creating action subsets and each subset contain group of similar hand gesture motion. This way, authors compare accuracy of proposed approach in terms of gesture and body recognition. In this review paper, we will show overall accuracy of Action Subsets that are created under specific criteria by the authors. Depending on that facts, we can see average accuracy measurement in overall value corresponding to each action subsets with amount of training.

Table 1 - Accuracy obtained from action subsets given by T training amount.

T	AS1	AS2	AS3	Average
10	92.39	94.65	93.7	93.58
20	93.3	94.6	99.1	95.66
30	93.81	85.72	93.7	87.74

From the **Table 1** we can find mean value of each average to define consistency of the proposed approach by finding mean value of function (Equation 1).

$$n_y = y_1 + y_2 + y_3 + \cdots + y_n \quad (1)$$

By calculating mean value of given averages, we get 92.3 which shows us solidarity and consistency of the proposed approach by considering the fact of Training times that are performed to output result. Authors also show extended data of each Action Subset where each subset contains proposed hand gesture types that are similar. In the context, you will also see type of proposed gesture and what error has been occurred during the execution and detection of the same gesture. It means if the proposed gesture was pinching and error occurred during the execution of this gesture it will be recorded what gesture has been detected instead of the proposed gesture. This data can be very useful to identify what type of difficulties the proposed approach is struggling when training the dataset of MSRDailyActivity3D Dataset. The article also provides secondary protocol where this time body attributes are also recognized but it only contains Kinect dataset compared with the MRSDailyActivity3D dataset. This protocol is containing 11 body and hand gesture tracking mixture. As we believe the research scope was explained better with the precise conditions during the first protocol we will keep second protocol out of our scope but it can be investigated within the article itself [23].

Table 2 - Accuracy results for MSR Action3D Dataset with other methodologies

Method	Accuracy			
	AS1	AS2	AS3	Average
Wang et al., 2016 [29]	-	-	-	97.4
CNN + LSTM [23]	93.3	94.6	99.1	95.7
Chen et al., 2015 [30]	98.1	92.0	94.6	94.9
Du et al., 2015 [31]	93.3	94.6	95.5	94.5

Tao & Vidal, 2015 [32]	89.8	93.6	97.0	93.5
Lillo et al., 2016 [33]	-	-	-	93.0
Vemulapalli et al., 2014 [34]	95.3	83.9	98.2	92.5
Ben Amor et al., 2016 [35]	-	-	-	89.0
Li et al., 2010 [36]	72.9	71.9	79.2	74.7

As **Table 2** shows, the proposed approach has significant accuracy on the proposed action subsets that are trained using CNN+LSTM. Some methodologies that are identified on table do not contain same action subsets as this research, but they also rely on the similar hand gesture types with only difference as the approach to measure accuracy is executed differently. We can conclude that, article relies on absolute facts by using known datasets to evaluate their approach as well as the presentation of paper shows the relevancy to the outputted data. Personally, the given approach can be very good alternative to implement CNN+LSTM on recognizing body and hand gestures. We highly recommend researches to see given approach on the original article to observe scale of the desired methodology. The approach can be implemented in today's games or engines to improve effectiveness of the gesture recognitions to develop further applications.

E. Pose estimation and gesture recognition

One of the problems in Virtual and Augmented reality applications is person pose estimation and hand gesture recognition. It can be challenging task, especially when environment is complex. Due to difficulty, it can even be divided in several parts: person detection, joints extraction and merge to the skeleton (pose estimation). Furthermore, hand gestures can be interpreted, if needed. Few years ago, pose estimation task had already been possible to perform with Microsoft Kinect SDK [37]. It uses RGB and depth camera data to extract skeleton and estimate its position in 3D space. In this case, point cloud data can be very useful in distinguishing person from background.

Nowadays, there are even more modern approaches to solve pose estimation task. With a help of deep neural networks, it is even possible to extract person and detect joints with only RGB camera data. Wei et al [38] and Cao et al [39] proposed interesting methodology to detect 2D pose. Convolutional neural network is utilized to detect joints of person. This [38] is a state-of-art method on LSP [39] and FLIC [40] datasets.

Figure 5 - Convolutional pose machine joints detection example[1]

Next year (2017), further technique by researchers was introduced [39]. Same as the method mentioned previously, it used convolutional neural network to detect joints from RGB image. Algorithm is capable to detect more than one person in image. It is state of art method in performance and efficiency on MPII [41] Multi-Person dataset, scoring 75.6% accuracy on whole testing set. On laptop with Nvidia GeForce GTX-1080 GPU algorithm achieves 8.8fps on a frame 1080x1920 resized to 368x654 with 19 people in it.

Figure 6 - "Realtime Multi-Person 2D Human Pose Estimation using Part Affinity Fields" pose estimation.

Another part of even deeper person behavior understanding is gesture recognition. It is a big part in VR application, because hand gestures enable more natural interaction that does not requires any input equipment. Control can be done by interpreting visual information only.

Gesture recognition in Virtual reality application have been already enabled by companies, such as, Leap Motion [42] or Softkinetic [43]. Mentioned companies are solving problem with depth cameras **Figure** 7.

a) b)

Figure 7 - a) DepthSense DS325[2] and b) Leap Motion sensor[3].

[1] https://www.youtube.com/watch?v=EOKfMpGLGdY

Both of mentioned sensors can be attached to headset or work in separate devices **Figure** 8.

a) b)

Figure 8- a) standalone sensor mode[4], b) sensor attached to headset[5].

For hands and gestures recognition, algorithms are using depth data. Because of this technology, sensors are depended on the object distance from camera and they have short working range, i.e., SoftKinetic DS325 working range is 0.15-1m.

However, hand detection and gesture recognition can be done with different techniques and data. Recently, different methods [44], [45] have been introduced. In one research [44], algorithm to detect hand joints from RGB data is proposed. It uses convolutional neural networks for hand pose detection. Method can run real-time with GPU and its accuracy is as high as other methods that uses depth sensor for the task. Furthermore, from different viewpoints, it can produce 3D hand pose estimation by triangulating feeds from different cameras (Fig. 9).

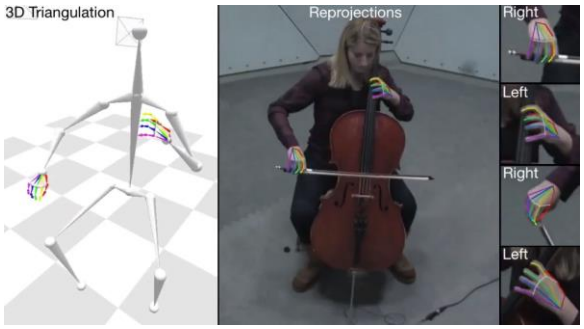

Figure 9 - Hand pose estimation generated by RGB data from different viewpoints[6]

Another proposed method [45] not only detects hand, but also, recognizes hand gestures. Research uses Danish and New Zealand sign language data from RWTH-PHOENIX-Weather 2014 [46] dataset for CNN training. Can achieve more than 100fps on single Nvidia Geforce 980 GTX.

Figure 10 - Small part of RWTH-PHOENIX-Weather 2014 signs language dataset

From reviewed methods, real implementation can be found. OpenPose project [47], utilizes pose estimation [38], [39] and hand and fingers joints detection [44].

Figure 11 - OpenPose project demonstration [47].

III. Conclusions

Large interest in Artificial intelligence keeps the scientific society in high research pace. We have overviewed latest achievements in the field of AI application for virtual reality technologies and provided systematic classification of research papers and their contributions to the field. Also, we have separated general advancements of CNN algorithm and advancements directly related to VR to provide search tool for relevant literature. This type of training set is optimal between time needed for preparation and performance for training. Also, if you have small data set we would recommend using pre-training technique. Depending on your technical capabilities we highly recommend implementing one of techniques addressing different object scales.

For pose estimation and gestures recognition, several approaches using only RGB or depth camera data were introduced. Due to the necessity of having a depth camera, the inherent noise associated with direct lightning conditions and the depth camera measuring range limitations, AI methods that uses only RGB data, might be one of the best approaches. From the given examples, it can be seen that AI based techniques [38][39][44][46] perform accurately. The drawback of the mentioned methods is their necessity of a high-end GPU to produce more frames per seconds.

[2] https://www.leapmotion.com/

[3] https://www.leapmotion.com/

[4] https://www.rockpapershotgun.com/tag/leap-motion-3d-jam/

[5] https://www.vrheads.com/how-use-leap-motion-your-oculus-rift

[6] https://www.youtube.com/watch?v=q4xbmEQp3VE

References

[1] S. Hong, S. Kwak, and B. Han, “Weakly Supervised Learning with Deep Convolutional Neural Networks for Semantic Segmentation: Understanding Semantic Layout of Images with Minimum Human

Supervision," *IEEE Signal Process. Mag.*, vol. 34, no. 6, pp. 39–49, 2017.
[2] G. Zheng, M. Tan, J. Yu, Q. Wu, and J. Fan, "Fine-grained image recognition via weakly supervised click data guided bilinear CNN model," *Proc. - IEEE Int. Conf. Multimed. Expo*, no. July, pp. 661–666, 2017.
[3] C. Hsu, C. Lin, and S. Member, "CNN - Based Joint Clustering and Representation Learning with Feature Drift Compensation for Large - Scale Image Data," vol. 20, no. 2, pp. 421–429, 2017.
[4] L. Jin and H. Liang, "Deep learning for underwater image recognition in small sample size situations," *Ocean. 2017 - Aberdeen*, no. 61379007, pp. 1–4, 2017.
[5] M. Valdenegro-Toro, "Best Practices in Convolutional Networks for Forward-Looking Sonar Image Recognition," 2017.
[6] K. Zhang, W. Zuo, S. Gu, and L. Zhang, "Learning Deep CNN Denoiser Prior for Image Restoration," pp. 3929–3938, 2017.
[7] F. Gomez-Donoso, A. Garcia-Garcia, J. Garcia-Rodriguez, S. Orts-Escolano, and M. Cazorla, "LonchaNet: A sliced-based CNN architecture for real-time 3D object recognition," *Proc. Int. Jt. Conf. Neural Networks*, vol. 2017–May, pp. 412–418, 2017.
[8] F. Beritelli, G. Capizzi, G. Lo Sciuto, C. Napoli, and F. Scaglione, "Automatic heart activity diagnosis based on gram polynomials and probabilistic neural networks". Biomedical Engineering Letters, vol. 8, issue 1, pp. 77-85, 2018.
[9] Q. Lu, C. Liu, Z. Jiang, A. Men, and B. Yang, "G-CNN: Object Detection via Grid Convolutional Neural Network," *IEEE Access*, vol. 5, pp. 24023–24031, 2017.
[10] T. Chen, S. Lu, and J. Fan, "S-CNN: Subcategory-aware convolutional networks for object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 8828, no. c, pp. 1–8, 2017.
[11] Y. Gao *et al.*, "Scale optimization for full-image-CNN vehicle detection," *IEEE Intell. Veh. Symp. Proc.*, pp. 785–791, 2017.
[12] B. Nagy and C. Benedek, "3D CNN based phantom object removing from mobile laser scanning data," *Proc. Int. Jt. Conf. Neural Networks*, vol. 2017–May, pp. 4429–4435, 2017.
[13] C. Eggert, S. Brehm, A. Winschel, D. Zecha, and R. Lienhart, "A closer look: Small object detection in faster R-CNN," *Proc. - IEEE Int. Conf. Multimed. Expo*, vol. 0, pp. 421–426, 2017.
[14] Y. Lou, G. Fu, Z. Jiang, A. Men, and Y. Zhou, "PBG-Net: Object detection with a multi-feature and iterative CNN model," *2017 IEEE Int. Conf. Multimed. Expo Work. ICMEW 2017*, no. July, pp. 453–458, 2017.
[15] and L. G.-D. Le-Le, Wang, "Research on Relief Effect of Image Based on the 5 Dimension CNN.," 2017, pp. 416–418.
[16] C. Termritthikun and S. Kanprachar, "Accuracy improvement of Thai food image recognition using deep convolutional neural networks," *2017 Int. Electr. Eng. Congr.*, pp. 1–4, 2017.
[17] C. Bentes, D. Velotto, and B. Tings, "Ship Classification in TerraSAR-X Images With Convolutional Neural Networks," *IEEE J. Ocean. Eng.*, vol. 43, no. 1, pp. 258–266, 2017.
[18] U. Asif, M. Bennamoun, and F. Sohel, "A Multi-modal, Discriminative and Spatially Invariant CNN for RGB-D Object Labeling," *IEEE T. Patt. Anal. Mach. Intell.*, vol. 8828, no. c, 2017.
[19] H. Li, Y. Huang, and Z. Zhang, "An improved faster R-CNN for same object retrieval," *IEEE Access*, vol. 5, no. 8, pp. 13665–13676, 2017.
[20] T. Guan and H. Zhu, "Atrous Faster R-CNN for Small Scale Object Detection," *2017 2nd Int. Conf. Multimed. Image Process.*, pp. 16–21, 2017.
[21] M. A. Waris, A. Iosifidis, and M. Gabbouj, "CNN-based edge filtering for object proposals," *Neurocomputing*, vol. 266, pp. 631–640, 2017.
[22] D. Anisimov and T. Khanova, "Towards lightweight convolutional neural networks for object detection," *2017 14th IEEE Int. Conf. Adv. Video Signal Based Surveill.*, no. August, pp. 1–8, 2017.
[23] J. C. Núñez, R. Cabido, J. J. Pantrigo, A. S. Montemayor, and J. F. Vélez, "Convolutional Neural Networks and Long Short-Term Memory for skeleton-based human activity and hand gesture recognition," *Pattern Recognit.*, vol. 76, pp. 80–94, 2018.
[24] L. Lo Presti and M. La Cascia, "3D skeleton-based human action classification: A survey," *Pattern Recognit.*, vol. 53, pp. 130–147, 2016.
[25] F. Bonanno, G. Capizzi, S. Coco, C. Napoli, A. Laudani, and G. Lo Sciuto, "Optimal thicknesses determination in a multilayer structure to improve the SPP efficiency for photovoltaic devices by an hybrid FEM-cascade neural network based approach". in International Symposium on Power Electronics, Electrical Drives, Automation and Motion (SPEEDAM), pp. 355-362, 2014.
[26] F. Yang, W. Choi, and Y. Lin, "Exploit All the Layers: Fast and Accurate CNN Object Detector with Scale Dependent Pooling and Cascaded Rejection Classifiers," *2016 IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. 2129–2137, 2016.
[27] M. Najibi, M. Rastegari, and L. S. Davis, "G-CNN: an Iterative Grid Based Object Detector," 2015.
[28] Y. Liu, H. Li, J. Yan, F. Wei, X. Wang, and X. Tang, "Recurrent Scale Approximation for Object Detection in CNN," vol. 1, pp. 571–579, 2017.
[29] C. Wang, Y. Wang, and A. L. Yuille, "Mining 3D Key-Pose-Motifs for Action Recognition," *2016 IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. 2639–2647, 2016.
[30] C. Chen, R. Jafari, and N. Kehtarnavaz, "Action recognition from depth sequences using depth motion maps-based local binary patterns," *Proc. - 2015 IEEE Winter Conf. Appl. Comput. Vision, WACV 2015*, pp. 1092–1099, 2015.
[31] Y. Du, W. Wang, and L. Wang, "Hierarchial Recurrent Neural Network for Skeleton Based Action Recognition," *IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. 1110–1118, 2015.
[32] L. Tao and R. Vidal, "Moving Poselets: A Discriminative and Interpretable Skeletal Motion Representation for Action Recognition," *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 2016–February, pp. 303–311, 2016.
[33] I. Lillo, J. C. Niebles, and A. Soto, "A Hierarchical Pose-Based Approach to Complex Action Understanding Using Dictionaries of Actionlets and Motion Poselets," pp. 1981–1990, 2016.
[34] R. Vemulapalli, F. Arrate, and R. Chellappa, "Human action recognition by representing 3D skeletons as points in a lie group," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 588–595, 2014.
[35] B. Ben Amor, J. Su, and A. Srivastava, "Action Recognition Using Rate-Invariant Analysis of Skeletal Shape Trajectories," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 1, pp. 1–13, 2016.
[36] W. Li, Z. Zhang, and Z. Liu, "Action Recognition Based on A Bag of 3D Points.pdf," *Comput. Vis. Pattern Recognit. Work. (CVPRW), 2010 IEEE Comput. Soc. Conf.*, pp. 9–14, 2010.
[37] "Microsoft Kinect SDK," 2018. [Online]. Available: https://www.microsoft.com/en-us/download/details.aspx?id=44561 .
[38] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh, "Convolutional Pose Machines."
[39] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields," 2016.
[40] B. Sapp and B. Taskar, "MODEC: c" *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 3674–3681, 2013.
[41] M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele, "2D human pose estimation: New benchmark and state of the art analysis," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 3686–3693, 2014.
[42] "Leap Motion product website." [Online]. Available: https://www.leapmotion.com/
[43] "Depthsense company website." [Online]. Available: https://www.sony-depthsensing.com.
[44] T. Simon, H. Joo, I. Matthews, and Y. Sheikh, "Hand Keypoint Detection in Single Images using Multiview Bootstrapping," pp. 1145–1153, 2017.
[45] O. Koller, H. Ney, and R. Bowden, "Deep Hand: How to Train a CNN on 1 Million Hand Images When Your Data is Continuous and Weakly Labelled," *2016 IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. 3793–3802, 2016.
[46] O. Koller, J. Forster, and H. Ney, "Continuous sign language recognition: Towards large vocabulary statistical recognition systems handling multiple signers," *Comput. Vis. Image Underst.*, vol. 141, pp. 108–125, 2015.
[47] "OpenPose: Real-time multi-person keypoint detection library for body, face, and hands estimation." [Online]. Available: https://github.com/CMU-Perceptual-Computing-Lab/openpose.