FirstPiano: A New Egocentric Hand Action Dataset Oriented Towards Augmented Reality Applications

Théo Voillemin, Hazem Wannous, Jean-Philippe Vandeborre

FirstPiano: A New Egocentric Hand Action Dataset Oriented Towards Augmented Reality Applications

Théo Voillemin, Hazem Wannous(✉), and Jean-Philippe Vandeborre

IMT Nord Europe, Univ. Lille, CNRS, UMR 9189 - CRIStAL, 59000 Lille, France
{theo.voillemin,hazem.wannous,
jean-philippe.vandeborre}@imt-nord-europe.fr

Abstract. Research on hand action recognition has achieved very interesting performance in recent years, notably thanks to deep learning methods. With those improvements, we can see new visions towards real applications of new Human-Machine interfaces (HMI) using this recognition. Such new interactions and interfaces need data to develop the best user experience iteratively. However, current datasets for hand action recognition in an egocentric view, even if perfectly useful for these problems of recognition, they generally lack of a limited but coherent context for the proposed actions. Indeed, these datasets tend to provide a wide range of actions, more or less in relation to each other, which does not help to create an interesting context for HMI application purposes. Thereby, we present in this paper a new dataset, FirstPiano, for hand action recognition in an egocentric view, in the context of piano training. FirstPiano provides a total of 672 video sequences directly extracted from the sensors of the Microsoft HoloLens Augmented Reality device. Each sequence is provided in depth, infrared and grayscale data, with 4 different points of view for the last one, for a total of 6 streams for each video. We also present the first benchmark of experiments using a Capsule Network over different classification problems and different stream combinations. Our dataset and experiments can therefore be interesting for research communities of action recognition and human-machine interface.

Keywords: Action recognition · Human machine interaction · Hand action dataset

1 Introduction

Augmented reality (AR) and virtual reality (VR) technologies have reached levels that were fantasized a few decades ago. However, the promises of new and natural interactions associated with the manipulation of a virtual world and the portability of these devices are still far from being fulfilled. Indeed, if VR devices have so far favored the use of joysticks, AR technologies, on the other hand, have mainly decided to use basic hand gesture recognition to permit the user to manipulate their interfaces. In both cases, hand action recognition has

yet been fully considered and implemented. Still, hand action is certainly the principal activity in everyday of a human being since we interact with the world around us with our hands. Hence, it would be interesting for AR and VR applications to draw all potential of hand action recognition features.

For such applications, algorithms for hand action recognition are needed, and for such algorithms, datasets for training them are needed. However, most of these datasets focus their efforts on providing a diversity of actions like object and environment manipulations and also interactions with people instead of focusing on a specific task. Hence, it is difficult to train such an algorithm that can be used into a real AR or VR application since these datasets do not provide a clear and useful context.

Thereby, we propose in this paper a new hand action dataset, FirstPiano, which provides data for action recognition purposes, but also for training some algorithms that can be used for useful AR applications and so helping HMI research field. The acquisition of the data of FirstPiano was thus motivated to capture an egocentric point of view of hand action within a strict and focus context for such AR purposes. This also motivated us to obtain the data from an easily reproducible setup. Hence, FirstPiano is a dataset capturing a context of piano learning acquired directly from the integrated sensors of the Microsoft HoloLens device. It contains 672 video sequences spread over up to 16 labels. Each sequence is provided with 6 different video streams, depth, infrared and grayscale images. A first benchmark using a recently developed neural network architecture is also described to propose a first estimation of results on our dataset on different data inputs and classification problem configurations.

2 Related Work

Hand Action Datasets: With the emergence and greater accessibility of RGB or depth sensors at the beginning of the 2010s, many researchers have increasingly focused on the problems of hand action recognition. We can then find a lot of datasets that offer a common base to compare different proposed algorithms. All hand action datasets are in an egocentric point of view, but we can find two different contexts. The first one is the context of only grasping an object [2,3,25], where it is interesting to analyze the exact position of the hand depending on the nature of the grasped object, such as by the fingertips when grasping a pencil, or pliers shape for holding a can. These datasets most focus on analyzing the shape of the hand, and the object grasped rather than the temporal evolution of the gesture of the hand. The second context is that of daily activities [1,11–13,17,19,22,28]. In this case, the object grasped is manipulated into fine action, such as pouring milk or cleaning glasses. First Person Hand Action dataset, presented by Hernando *et al.* [12], is certainly the current most elaborate dataset in this context with 1,175 sequences over 45 labels, but with RGB and depth data for each one of its sequences and also the exact position of the joints of the hand doing the action. EGTEA, presented by Li *et al.* [17], the evolution of GTEA [11] contains 28 h of daily activities with RGB videos,

and also the gaze information and the mask of the hand. Still, these datasets, even if very interesting for hand action recognition training, lack a concrete, precise, and focused context by providing a lot of different actions, not necessarily linked to each other and whose recognition would not be of interest for an AR application. However, from a third person point of view, we can find interesting hand gesture datasets that offer precise context for applications. For example, Molchanov *et al.*, with NVGesture [20], proposed a dataset of hand gestures for designing touchless interfaces while using a car so the user can focus on driving. De Smedt *et al.* proposed DHG 14/28 [5] with a coarse and fine gesture that is executed both with the all hand opens or with only the index that can be used for interacting with computer interfaces.

Hand Action Recognition Algorithms: The last few years, deep learning algorithms have become one the most efficient method in many fields especially hand action recognition. Recently, even handcrafted approaches finally use neural network by extracting manually very precise and fine features before passing them to a network [31] or by using a neural network in parallel of handcrafted features then by merging them together [15,30]. Specialized algorithms for hand action recognition are quite rare to find in the state of the art in contrast to hand gesture recognition, where we can find neural networks that are specially designed to hand understanding by using hand skeleton with RNN architecture [4] or graph convolutional network [16]. Hand action recognition solutions are generally the same as those for human body activities [7,14,24] since both can consider a skeleton based representation as a sequence of joints [6], or 2D/3D CNN when this representation is not rich enough to capture the movement and analyze the manipulated object. Lin *et al.* [18] implemented a temporal shift module to share temporal information from a frame to the next inside a simple 2D CNN. Duarte *et al.* [8] proposed a first implementation of a 3D capsule network [26] for video understanding of human activities.

Human-Machine Interfaces on AR Device: Just before the arrival of augmented reality devices to the general public, such as the Microsoft HoloLens, some research groups have already developed their own smart glasses with hand action recognition solution integrated. For example, Schröder *et al.* [27] proposed smart glasses with RGB and depth sensors connected to an external computer to process an action made by the user to then display contextual information over the glasses. Essig *et al.* [9] proposed a similar project, but with a long-term vision to be fully personalized to the user by progressively constructing a mental representation of the user to display precise feedback over his glasses.

3 FirstPiano: Egocentric Dataset of Piano Interaction

3.1 Dataset Overview

FirstPiano is a hand action dataset focused on hand action recognition and to be used for AR applications. For this purpose, it contains a set of 672 actions videos with 6 different modalities for each of them for a total of 4,032 videos and 473,892 frames. We propose a context of piano training where a subject is asked to play a major scale in a right way or a wrong way for a total of up to 16 labels.

To understand how we get to 16 labels, we need to explain how a major scale is constructed in the musical field. The western musical language has 7 different notes (C, D, E, F, G, A, B), which are the white keys on a piano and 2 variations for each of them, flat or sharp, which are the black keys. Two musical notes are separated by what is called an interval, the smallest one being the half interval which can be observed on a piano between two consecutive keys (for example, between white A and black $B\flat$ or between white B and white C). A full interval is a succession of two half intervals (for example, between white C and white D, between black $G\sharp$ and black $B\flat$ or between white E and black $F\sharp$). To construct a major scale, one need to take a base note, which give the name of the scale, and to take the 7 next notes by following this series of intervals: 1, 1, 1/2, 1, 1, 1, 1/2. We propose examples for C and F major scales on Fig. 1.

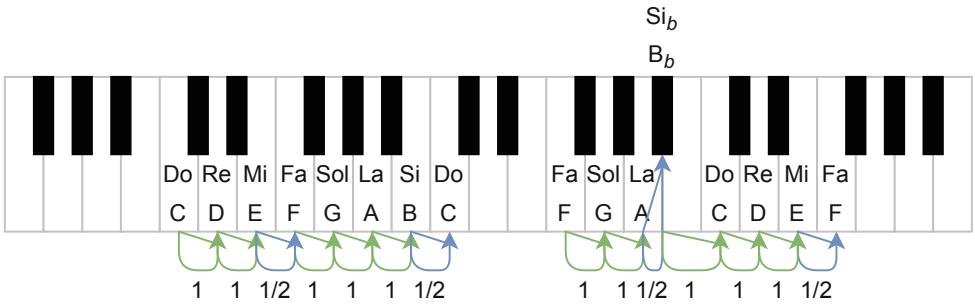

Fig. 1. Example of C and F major scale construction on a piano

Moreover, to be considered as rightly-played, we impose a precise gesture. Let's consider the fingers on a hand as 1 for the thumb, 2 for the index, 3 for the middle, 4 for the ring finger and 5 for the little finger. When played with the right hand, the 8 notes of a major scale must be played in ascending way on the following order: 1, 2, 3, 1, 2, 3, 4, 5. When played with the left hand the order is different: 5, 4, 3, 2, 1, 3, 2, 1. For descending way, the order for both hands is reversed. An example of this precise gesture with the left hand on a D major can be seen on Fig. 2.

Hence, each major scale is constructed from a start note is unique and need to have a precise gesture during its execution. We decided to limit the dataset to the

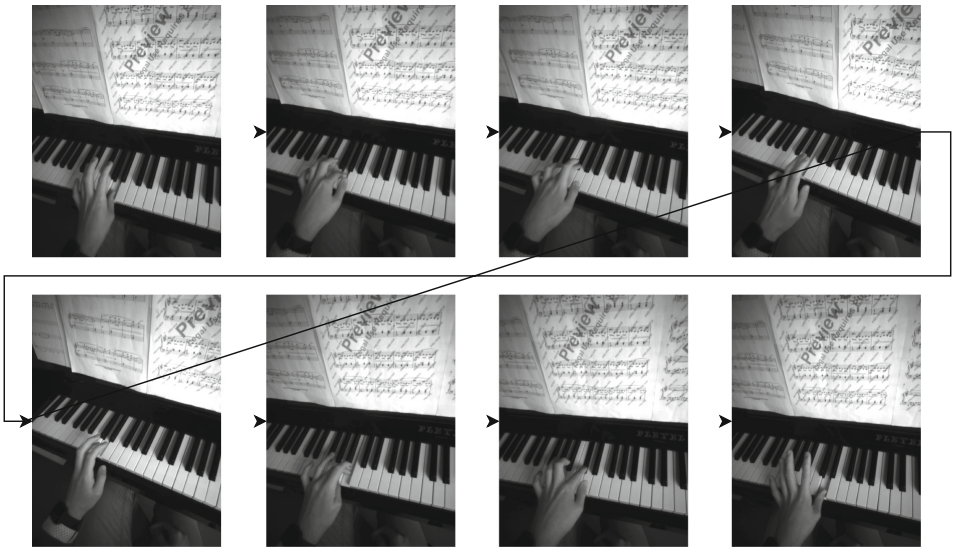

Fig. 2. Gesture imposed to right play a major scale. Example with left hand on D major

7 major scale that can be constructed from any white key. Each of these scales has multiple start emplacement in the piano and is performed with the right and the left hand, both in two other ways, crescendo and crescendo-decrescendo. We decided not to differentiate all those different playing configurations directly; therefore, for rightly-executed scale, there is 7 labels, one for each scale. Since the note compositions of all major scales are unique and different and since there is only a unique and precise gesture for playing a scale, we introduce in the dataset a set of sequences of wrongly-played executed scales. Some of them with wrong notes, extra notes, fewer notes and wrong gestures. Knowing that a sequence can multiply these mistakes, we reached a total of 16 different labels in the dataset.

3.2 Sensors and Acquisition Modalities

For the data acquisition, we decided to directly use the integrated sensors of the Microsoft HoloLens device thanks to the HoloLensForCV project[1] for data extraction. This AR headset directly provides an accessible and constant setup unlike other dataset that use a custom one. The Microsoft HoloLens is equipped with 8 different sensors. Since the frontal RGB camera has a too small viewing angle and since the long throw depth does not provide interesting data, we decided to keep the other 6 sensors to know: the short depth, long throw reflectivity, and the 4 peripheral grayscale sensors (see Fig. 3).

With a total of 66,320 of short throw depth frames, this sensor provides interesting information, especially to have precision to know if a key is pressed on the piano or else to let the possibility to extract hand pose and its skeleton

[1] https://github.com/microsoft/HoloLensForCV.

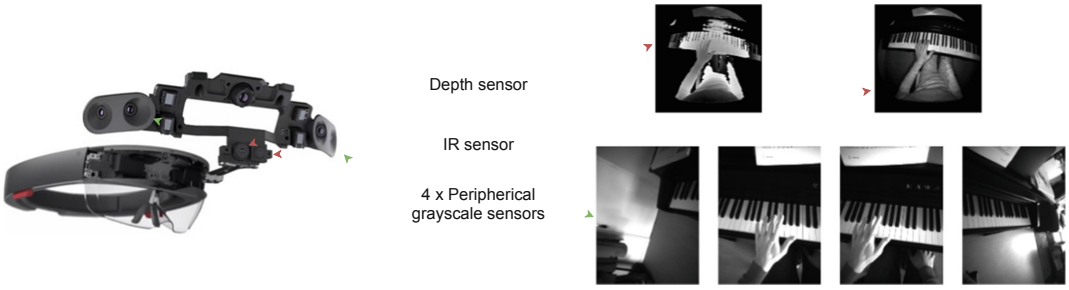

Fig. 3. The 6 sensors of the HoloLens device used and an example of each one inside the FirstPiano dataset

[10,21]. With 66,194 frames, the long throw reflectivity sensor permits to get a good central field of view of the hand of the player. The 4 peripheral grayscale sensors, with a total of 341,378 frames provide a full field of view on the entire piano, divided into 4 more precise points of view.

The acquisition was made in 2 different places and with 3 different subjects to provide a bit of diversity into the speed of execution and ease to play. Finally, each of the 7 major scales has 12 recordings, all with a rightly-played and wrongly-played recording, each in 4 different gestures, with the right hand or the left hand in a crescendo and crescendo-decrescendo way for a total of 672 sequences.

4 Benchmark Evaluation

4.1 Method: 2D Deep Video Capsule Network

We decided to use the 2D Deep Video Capsule Network approach (2D DVCN) [29] for the evaluation of FirstPiano. 2D DVCN architecture consists of the implementing of the temporal shift module [18], which permits adding temporal information into 2D neural network architecture that analyse spatial features, into a deep capsule network [23,26]. Briefly, a capsule is quite similar to a convolutional layer in its function since it captures and analyzes spatial features thanks to convolutional operations. The difference being that a capsule groups together the result of many convolutional operations to encapsulate many complex spatial features. We implemented a temporal shift module applied on the capsule by shifting part or all of the convolution operations composing the capsule of all or part of the capsule in a layer, leading us into 3 different implementations. The first one is shifting the first convolutions of the first capsules so that we share partial spatial information of some capsules to let the network to work with both past and current information. The second one is shifting first convolutions of all the capsules, since all capsules capture independent complex spatial features, it seems logical to consider all of them equally. In the last one is shifting all the convolutions of the first capsules of a layer, since all the convolutions of a capsule are linked together to represent a complex spatial feature, it also seems logical to shift all of them inside a same capsule.

While training 2D DVCN, one of these temporal shift module implementation is initially chosen and does not change during all the training. The shift module is also applied to all capsule layers of the network, whose architecture can be observed on Fig. 4. More precision of all this information can be found in [29].

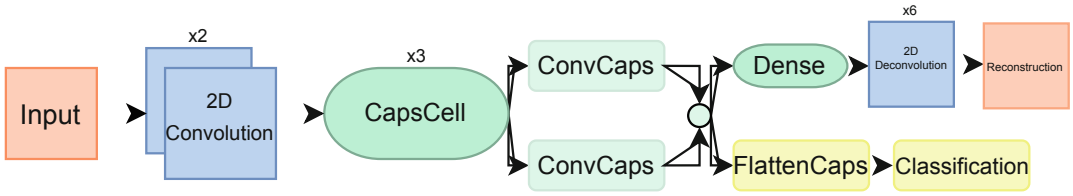

Fig. 4. Illustration of 2D DVCN [29]. ConvCaps is a standard 2d convolutional capsule layer but with temporal shift applied on it. CapsCell is a set of 3 ConvCaps with a residual branch between the first and the last

4.2 Experiments

For our experiments, we use the Keras library with Tensorflow in backend. 2D DVCN was trained on the following hardware configuration: Intel i9 9900k, Nvidia RTX Titan and 32 GB RAM. We used the Adam Optimizer during training with a starting learning rate of 0.001 and trained the network over 500 epochs.

Binary Recognition of Major Scales: For our first experiments, we considered a problem of binary classification. Even if we described that the sequences of FirstPiano are distributed over 16 labels, we firstly decided to consider only two labels, one when the major scale is rightly-played and the other if it is played wrongly-played, whatever the major scale. We justify the approach for two reasons, first one is that it permits us to test our dataset on an easy problem. The second one is that a rightly-played major scale is unique and can be deduced from the very first note. We can then justify having a network for this binary classification and a second method dedicated to identify the major scale with the first frames of a sequence classified as justify.

We start by using only two of the grayscale sensors, the central left and the central right since these two sensors are enough to get a full vision of the piano. Moreover, in this configuration, we let the network learns which hand is playing instead of giving him directly the right sensor associated with the playing hand. The two frames are simply juxtaposed as a panorama before being sent to the network. In this configuration, we reach an accuracy of 76.58% of good classification when using the shifting on the first convolutions of all capsules (see Table 1).

Table 1. Comparison of 2D DVCN results depending on the temporal shift implementation used over our FirstPiano dataset considering 2 labels and using 2 grayscale streams

Temporal shift	Accuracy (%)
1^{rst} convolutions of the 1^{rst} capsules	75.71
All convolutions of the 1^{rst} capsules	76.26
1^{rst} convolutions of all capsules	**78.56**
No shifting	76.13

We then decided to feed the network with only the depth data. This stream has a sufficient angle of view to observe the majority of the piano and since we did not acquire some playing on the extreme sides of the piano, the depth stream is sufficient alone. Moreover, in addition to being visually interpreted the same as a grayscale image, with fewer details, it also contains additional information such as the distance of each pixel to the sensor. We obtain similar results than the configuration of using as input both grayscale streams, with the temporal shift of the first convolutions of all capsules returning the best result. however, it seems that the depth stream alone does not provide as much information as the two grayscale streams since we only reached an accuracy of 73.33% of good classification in this case (see Table 2).

Table 2. Comparison of 2D DVCN results depending on the temporal shift implementation used over our FirstPiano dataset considering 2 labels and using depth stream only

Temporal shift	Accuracy (%)
1^{rst} convolutions of the 1^{rst} capsules	69.14
All convolutions of the 1^{rst} capsules	69.91
1^{rst} convolutions of all capsules	**73.33**
No shifting	70.57

Combining all the 3 previously described video streams by juxtaposing them leads us towards very similar results to those of the case using only depth data (see Table 3). However, given that we are using both grayscale video streams, we can expect to get at least similar results to the case using them only. The most logical conclusion is that the way 2D DVCN was implemented was probably not deep enough to process such high dimensional input and since capsules work with convolutional operations, the juxtaposition of depth and grayscale into a unique image lost them.

Table 3. Comparison of 2D DVCN results depending on the temporal shift implementation used over our FirstPiano dataset considering 2 labels and using 3 video streams

Temporal shift	Accuracy (%)
1^{rst} convolutions of the 1^{rst} capsules	70.14
All convolutions of the 1^{rst} capsules	**73.56**
1^{rst} convolutions of all capsules	72.54
No shifting	69.14

Multi Labelling of Major Scales Recognition: Since we obtained pretty good results in the easy configuration of binary classification, we decided to complicate the problem by multi labelling all the major scales. We then considered 8 labels, one for each different rightly-played major scale, for a total of 7 and the last one, which includes all wrongly-played scales. We can justify this choice because, even if a wrongly-played scale is associated to a note in the hierarchy of our dataset, it can be difficult, even impossible, to tell from which scale there have been wrong notes. Moreover, we think it will be far more interesting to have a specialised method to precisely identify what mistake is made in a wrongly-played scale, especially the exact frames it occurs. Since a single network cannot deal with so many particularities and special cases, we decided it was the best configuration to a multi labelling problem from a unique method.

Table 4. Comparison of 2D DVCN results depending on the temporal shift implementation used over our FirstPiano dataset considering 8 labels and using 2 grayscale streams

Temporal shift	Accuracy (%)
1^{rst} convolutions of the 1^{rst} capsules	59.09
All convolutions of the 1^{rst} capsules	**61.93**
1^{rst} convolutions of all capsules	61.36
No shifting	60.20

Since the problem has become more difficult, we can observe a degradation of the classification accuracy. Using the two grayscale video streams, we reached 61.93% of good classification when shifting all convolutional operations of the first capsules of a layer (see Table 4). Slightly worse results were obtained using all 3 depth and grayscale streams. We reached 59.45% of good classification using the same temporal shift, we also found similar degradation of results using the depth stream juxtapose next to the grayscale ones (see Table 5). These results are coherent with the complication of the classification problem. Indeed, we can explain it since the network now has to also focus exactly on the notes played and not only on the gesture or the number of notes played.

Table 5. Comparison of 2D DVCN results depending on the temporal shift implementation used over our FirstPiano dataset considering 8 labels and using 3 video streams

Temporal shift	Accuracy (%)
1^{rst} convolutions of the 1^{rst} capsules	57.23
All convolutions of the 1^{rst} capsules	**59.45**
1^{rst} convolutions of all capsules	56.46
No shifting	56.89

5 Conclusion

In this paper, we proposed a new dataset for hand action recognition oriented towards AR applications, FirstPiano. The dataset provides 672 video sequences, each one with 6 different video streams, of rightly and wrongly played major scale on a piano in various configurations. Data are extracted directly from the integrated sensors on the Microsoft HoloLens device so that the acquisition setup can be constant for everyone wanted to use the dataset for AR applications or even to complete it. We also provide the first benchmark to evaluate FirstPiano on different configurations, such as binary and multi classification using 1 to 3 video streams as input.

As future works, it could be interesting to complete the dataset even for current actions provided but also with new piano exercises such as dissociation between the left hand and the right hand during playing or even different rhythms. It would also be interesting to deepen the notation of labels, especially by including, for each sequence, the frames when a new note is being played as right as the finger used so that it could be possible to train a precise algorithm that can find the exact location of mistakes.

We believe that this work presenting a new dataset in addition to the tested benchmark can encourage new research in the hand action recognition but also in the human computer interface fields.

References

1. Bambach, S., Lee, S., Crandall, D.J., Yu, C.: Lending a hand: Detecting hands and recognizing activities in complex egocentric interactions. In: IEEE International Conference on Computer Vision (ICCV). pp. 1949–1957 (2015)
2. Bullock, I.M., Feix, T., Dollar, A.M.: The yale human grasping dataset: Grasp, object, and task data in household and machine shop environments. The International Journal of Robotics Research **34**(3), 251–255 (2015)
3. Cai, M., Kitani, K.M., Sato, Y.: A scalable approach for understanding the visual structures of hand grasps. In: IEEE International Conference on Robotics and Automation (ICRA). pp. 1360–1366 (2015)
4. Chen, X., Guo, H., Wang, G., Zhang, L.: Motion feature augmented recurrent neural network for skeleton-based dynamic hand gesture recognition. IEEE International Conference on Image Processing (ICIP), September 2017

5. De Smedt, Q., Wannous, H., Vandeborre, J.P., Guerry, J., Saux, B.L., Filliat, D.: 3D hand gesture recognition using a depth and skeletal dataset: Shrec 2017 track. In: Proceedings of the Workshop on 3D Object Retrieval. 3Dor 2017, pp. 33–38. Eurographics Association, Goslar, DEU (2017)
6. De Smedt, Q., Wannous, H., Vandeborre, J.-P.: 3D hand gesture recognition by analysing set-of-joints trajectories. In: Wannous, H., Pala, P., Daoudi, M., Flórez-Revuelta, F. (eds.) UHA3DS 2016. LNCS, vol. 10188, pp. 86–97. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-91863-1 7
7. Devanne, M., Wannous, H., Daoudi, M., Berretti, S., Bimbo, A.D., Pala, P.: Learning Shape Variations of Motion Trajectories for Gait Analysis. In: International Conference on Pattern Recognition (ICPR). pp. 895–900. Cancun, Mexico (2016)
8. Duarte, K., Rawat, Y., Shah, M.: VideoCapsuleNet : a simplified network for action detection. In: Advances in Neural Information Processing Systems, pp. 7610–7619 (2018)
9. Essig, K., Strenge, B., Schack, T.: ADAMAAS: towards smart glasses for mobile and personalized action assistance.. In: 9th ACM International Conference, pp. 1–4, June 2016
10. Fang, L., Liu, X., Liu, L., Xu, H., Kang, W.: JGR-P2O: joint graph reasoning based pixel-to-offset prediction network for 3D hand pose estimation from a single depth image. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12351, pp. 120–137. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58539-6 8
11. Fathi, A., Ren, X., Rehg, J.M.: Learning to recognize objects in egocentric activities. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3281–3288 (2011)
12. Garcia-Hernando, G., Yuan, S., Baek, S., Kim, T.K.: First-person hand action benchmark with RGB-D videos and 3D hand pose annotations. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 409–419 (2018)
13. Goyal, R., et al.: The something video database for learning and evaluating visual common sense. In: IEEE International Conference on Computer Vision (ICCV) 2017, pp. 5843–5851. Los Alamitos, CA, USA, October 2017
14. Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., Fei-Fei, L.: Large-scale video classification with convolutional neural networks. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1725–1732 (2014)
15. Khan, M.A., Sharif, M., Akram, T., Raza, M., Saba, T., Rehman, A.: Hand-crafted and deep convolutional neural network features fusion and selection strategy: an application to intelligent human action recognition. Appl. Soft Comput. **87**, 105986 (2020)
16. Li, C., Li, S., Gao, Y., Zhang, X., Li, W.: A two-stream neural network for pose-based hand gesture recognition. CoRR abs/2101.08926 (2021)
17. Li, Y., Liu, M., Rehg, J.M.: In the eye of beholder: joint learning of gaze and actions in first person video. In: Proceedings of the European Conference on Computer Vision (ECCV), September 2018
18. Lin, J., Gan, C., Han, S.: Tsm: Temporal shift module for efficient video understanding. In: IEEE International Conference on Computer Vision (ICCV) (2019)
19. Moghimi, M., Azagra, P., Montesano, L., Murillo, A.C., Belongie, S.: Experiments on an RGB-D wearable vision system for egocentric activity recognition. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 611–617 (2014)

20. Molchanov, P., Yang, X., Gupta, S., Kim, K., Tyree, S., Kautz, J.: Online detection and classification of dynamic hand gestures with recurrent 3D convolutional neural network. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 4207–4215 (2016)
21. Oberweger, M., Wohlhart, P., Lepetit, V.: Hands deep in deep learning for hand pose estimation. In: Computer Vision Winter Workshop, pp. 1–10 (2015)
22. Pirsiavash, H., Ramanan, D.: Detecting activities of daily living in first-person camera views. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 2847–2854 (2012)
23. Rajasegaran, J., Jayasundara, V., Jayasekara, S., Jayasekara, H., Seneviratne, S., Rodrigo, R.: DeepCaps: going deeper with capsule networks. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 10717–10725 (2019)
24. Rhif, M., Wannous, H., Farah, I.R.: Action recognition from 3D skeleton sequences using deep networks on lie group features. In: 24th International Conference on Pattern Recognition (ICPR), pp. 3427–3432 (2018)
25. Rogez, G., Supancic, J.S., Ramanan, D.: Understanding everyday hands in action from RGB-D images. In: IEEE International Conference on Computer Vision (ICCV), pp. 3889–3897 (2015)
26. Sabour, S., Frosst, N., Hinton, G.E.: Dynamic routing between capsules. In: Guyon, I., et al. (eds.) Advances in Neural Information Processing Systems. vol. 30. Curran Associates, Inc. Red Hook (2017)
27. Schröder, M., Ritter, H.: Deep learning for action recognition in augmented reality assistance systems. In: ACM SIGGRAPH 2017 Posters, pp. 1–2, June 2017
28. Tang, Y., Tian, Y., Lu, J., Feng, J., Zhou, J.: Action recognition in RGB-D egocentric videos. In: IEEE International Conference on Image Processing (ICIP), pp. 3410–3414 (2017)
29. Voillemin, T., Wannous, H., Vandeborre, J.P.: 2D deep video capsule network with temporal shift for action recognition. In: 25th International Conference on Pattern Recognition (ICPR), pp. 3513–3519 (2021)
30. Wang, L., Qiao, Y., Tang, X.: Action recognition with trajectory-pooled deep-convolutional descriptors. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4305–4314 (2015)
31. Wang, S., Hou, Y., Li, Z., Dong, J., Tang, C.: Combining convnets with hand-crafted features for action recognition based on an HMM-SVM classifier. Multim. Tools Appl. **77**(15), 18983–18998 (2018)