A Deep Learning Approach for Human Action Recognition Using Skeletal Information

Eirini Mathe, Apostolos Maniatis, Evaggelos Spyrou, and Phivos Mylonas

1 Introduction

Human action recognition still remains one of the most challenging research areas in the field of computer vision. Several open challenges in this area include the representation, the analysis, and ultimately the recognition of the human actions (Berretti et al. 2018). Toward this goal, machine learning approaches have been widely used. However, traditional machine learning approaches fail to show robustness when the number of possible actions increases or when the camera angle changes. During the last few years, advances in hardware have facilitated training and application of deep neural network architectures (Krizhevsky et al. 2012), which are able to learn representations from data without the need of hand-crafted

E. Mathe
Institute of Informatics and Telecommunications, National Center for Scientific Research- "Demokritos", Athens, Greece

Department of Informatics, Ionian University, Corfu, Greece
e-mail: emathe@iit.demokritos.gr

A. Maniatis
Department of Computer Engineering T.E, Technological Education Institute of Sterea Ellada, Lamia, Greece
e-mail: amaniatis@teiste.gr

E. Spyrou (✉)
Institute of Informatics and Telecommunications, National Center for Scientific Research- "Demokritos", Athens, Greece

Department of Computer Engineering T.E, Technological Education Institute of Sterea Ellada, Lamia, Greece
e-mail: espyrou@iit.demokritos.gr

P. Mylonas (✉)
Department of Informatics, Ionian University, Corfu, Greece
e-mail: fmylonas@ionio.gr

P. Vlamos (ed.), *GeNeDis 2018*, Advances in Experimental Medicine and Biology 1194, https://doi.org/10.1007/978-3-030-32622-7_9

rules or features, while their accuracy may significantly increase when they are provided with more data. Moreover, recently, several publicly available datasets (Liu et al. 2017a) have emerged in the field of human action recognition, enabling the evaluation of novel architectures and action representations in real-like scenarios. Apart from the aforementioned challenges, the design of novel deep architectures and their application in real-life scenarios are also among the research targets of this field.

In this paper, we propose a novel visual representation of human actions, based on the discrete Fourier transformation (DFT). More specifically, we concatenate raw signal images that result from the 3D motion of human skeletal joints. The input required for the extraction of these joints consists of aligned RGB and depth video sequences. It is performed using the well-known Kinect v2 camera and its accompanying SDK. Moreover, we propose a novel convolutional neural network (CNN) architecture which uses as input the DFT transformation of the aforementioned images. We evaluate the proposed approach using the challenging PKU-MMD dataset (Liu et al. 2017a) consisting of 51 human actions, and we demonstrate that the proposed approach may be used in real-like environments for the recognition of activities of daily living (ADLs) (Lawton and Brody 1969).

The rest of this paper is organized as follows: Section 2 presents related research in the field of human action recognition using deep learning approaches and focusing on those that work on skeletal information. Section 3 presents the concepts of deep learning and convolutional neural networks that have been used in the context of this work. The proposed action representation and deep network architecture are then described in Sect. 4. Experimental results are presented in Sect. 5 and discussed in Sect. 6, which also includes plans for further extensions and applications of this work.

2 Related Work

The problem of human action recognition has attracted many research efforts, which have been continuously growing during the last decade. In this section we aim to present approaches that are based on deep networks. Typically, these types of methods do not include a feature extraction step; they are instead based upon a representation of the action. However, we should herein emphasize that approaches that propose the extraction of features still exist (Mathe et al. 2018).

Skeletal data consist of the 3D positions of human skeleton joints. These may be considered as high-level features for the recognition process. The most popular method for skeletal extraction is based on RGB sequences accompanied by corresponding depth maps, i.e., as the approach adopted by Kinect sensors. Of course, skeletons are prone to errors, due to situations such as occlusion and viewpoint changes. Moreover, certain actions may have significantly different appearance upon abrupt changes of viewpoint. Two major categories of tasks exist: (a) segmented recognition and (b) continuous (online) recognition (Wang et al. 2018a).

The difference between the two categories is that within the first, we assume that the input video sequence only contains the action to be recognized, i.e., frames not depicting the action (before/after the action) have been removed. Note, that for the first category, common deep architectures used are recurrent neural networks (RNNs) (Graves et al. 2013) and convolutional neural networks (CNNs) (LeCun et al. 1998). For the second category, RNNs are typically used. In case a CNN is used, the majority of the approaches include a step of converting skeleton sequences to a single image, in a way that both spatial and temporal information are maintained and reflected to low-level image properties, i.e., color and/or texture may be used for the separation of classes. Note that the proposed approach uses a CNN and a step for converting 1D skeleton sequences to a single image, as it will be described in Sect. 4.

In the work of Du et al. (2015), the authors divide the skeleton joints into five groups (arms, legs, and trunk), i.e., corresponding joints are concatenated as a single vector. All five parts are then concatenated so as to capture the spatial information per frame, with the *x*, *y*, and *z* components of their 3D coordinates corresponding to the R, G, and B components of a color image, respectively. Then, representations of all frames of a sequence are arranged chronologically, to capture its temporal properties. A CNN architecture is used for classification. Wang et al. (2018b) proposed the use of "joint trajectory maps," where hue is used to capture the motion direction information of skeleton joints. Motion trajectories are projected onto three Cartesian planes (i.e., front, top, and side plane), and motion magnitude is encoded by appropriate settings of saturation and brightness, so that changes of texture reflect motion changes. The resulting maps are classified into actions, by CNNs. Similarly, Hou et al. (2018) also encoded skeleton joints' sequences into "skeleton optical spectra," which were also color texture images. The variation of color was used to introduce the temporal information to the representation, as changes of hue.

Li et al. (2017) proposed the use of "joint distance maps," which are also texture images. Contrast to (Hou et al. 2018; Wang et al. 2018b), projections to the three Cartesian planes are unnecessary. Instead, pairwise distances of joints are used. Three maps are used to encode the distances in the three orthogonal 2D planes, and a fourth one is used to encode distances in the 3D space. Hue is used for encoding variations of distances. This way, the description is more robust to changes of viewpoint, which as we have already discussed are common in real-life applications. A CNN is then used for each map, and classification is a result of a late fusion scheme which is applied. In the work of Liu et al. (2017b), transforms are applied to skeleton sequences in an effort to make them invariant to the position and the initial orientation of the skeleton. Skeleton data are considered as points into a 5D space, each consisting of 3D space coordinates, time, and joint label. They are then projected into a 2D image by selecting two of the aforementioned dimensions, while the remaining three are used as R, G, and B values. This way, color images are formed and used as input to a multi-stream CNN scheme. Finally, Ke et al. (2017) presented "SkeletonNet," where contrary to the majority of the approaches, they did not extract 3D coordinates. Instead, they extracted translation, rotation, and scale-invariant features. More specifically, the skeleton was divided into five parts as in

(Du et al. 2015). Then from each part, they extracted vector representations which are generated from pairwise relative positions between joints. Cosine distances between the aforementioned vectors within a specific part and the normalized magnitudes of each vector are extracted. These 10 representations were concatenated and then used as input to a two-stream CNN.

3 Deep Learning and Convolutional Neural Networks

Deep learning is a subfield of machine learning, which has attracted a lot of research interest during the last few years. Its main idea is the use of multiple layers to nonlinearly process the network's input, so as to "learn" to extract features. The output of each layer is fed to the next layer. Ultimately, they become able to learn multiple levels of representations which correspond to multiple levels of abstraction. Deep network architectures play a key role in several application fields such as computer vision, audio analysis, speech recognition, etc., i.e., in tasks where traditional machine learning approaches fail to achieve acceptable levels of accuracy for real-life applications. It is generally accepted that computer vision is the area that has benefited the most. A plethora of deep architectures have been proposed during the last few years and have been successfully applied to traditional computer vision problems as well as to novel applications.

The most common approach when dealing with computer vision problems is the convolutional neural networks (CNNs) (LeCun et al. 1998). The architecture of a CNN resembles to the one of a traditional neural network (NN); however, its goal is to learn a set of convolutional filters. Training takes place as with every other NN; a forward propagation of data and a backward propagation of error take place to update weights. The convolutional layers are those that play the key role in the whole process. Their neurons are grouped in rectangular grids, so that each would perform a convolution in a part of the input image. The learning process aims to learn the parameters of this convolution. Pooling layers are usually placed after a single or a set of serial or parallel convolutional layers. Their input consists of small rectangular image blocks from the convolutional layer. The latter are then subsampled; a single output is produced from each block. Finally, dense layers (which are commonly referred to as "fully connected" layers) are the ones that are responsible for classification, based on the features that were previously extracted by the convolutional layers and subsampled by the pooling layers. Note that each node of a dense layer is connected to all nodes of its previous layer. To avoid overfitting, one approach (which we also adopt in this work) is the use of the dropout regularization technique (Srivastava et al. 2014). When using this technique, at each training stage, several nodes are "dropped out" of the network. This way, complex coadaptations on training data are prevented, leading to the reduction or even total prevention of overfitting.

4 Human Action Recognition

The proposed approach uses as its input, 3D skeletal data that have been captured by the Microsoft Kinect v2 sensor (Zhang 2012) which combines a traditional RGB and a depth camera. Kinect is complemented by its SDK which, among other, is able to provide the 3D positions of a predefined set of human skeletal joints, in real time. A graph representation has been adopted; nodes correspond to body parts (e.g., arms, legs, head, etc.), and edges follow the joints' structure. Note that a parent-child relationship is implied, i.e., HEAD is parent of NECK, while NECK is parent of SPINE_SHOULDER, etc. A total of 25 joints are available. We should emphasize that for each joint, its *x*, *y*, and *z* coordinates are provided. We consider each coordinate of each joint as a single 1D signal, thus resulting in 75 1D signals, for any given video sequence and for each person. In Fig. 1 we illustrate the 25 human skeleton joints that are extracted using the Kinect SDK.

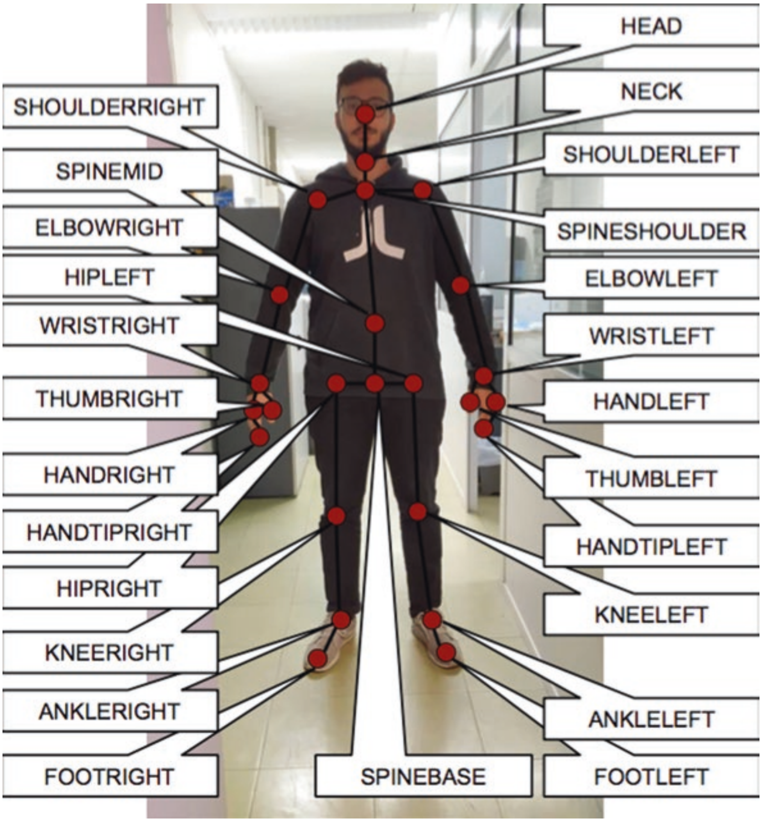

Fig. 1 Extracted human skeleton 3D joints using the Kinect SDK

Inspired by the work of Jiang and Yin (2015), we first create an activity image by concatenating the aforementioned 75 1D signals. We will refer to the result of this concatenation as “signal image.” Then, we apply the 2D discrete Fourier transform (DFT) to the signal image and preserve only the magnitude of the transform (i.e., the phase is discarded). The result is again an image, which we will refer to as “activity” image. In Fig. 2 we illustrate an example of a signal image and the corresponding activity image.

We should herein emphasize that our work focuses only on the classification of a given action into a set of predefined classes. Therefore, we should clarify that it does not perform any temporal segmentation (i.e., to detect the beginning and the ending of a possible action); instead we consider this problem as solved.

For evaluation purposes we work on pre-segmented sequences of videos, aiming to only recognize the performed actions within each segment. We also assume that each segment contains exactly one action. We should highlight that human-performed actions may typically vary in terms of duration, even when performed by the same user; thus an interpolation step is necessary. To tackle this issue and upon experimentation, we decided to set a threshold T_s for the duration of each action. Signals resulting from all actions with a duration $T_a < T_s$ are padded with zeros, while the length of those with $T_a > T_s$ is reduced upon a linear interpolation step. This way, all signal images have a fixed length of T_s 75.

The architecture of our proposed CNN is presented in detail in Fig. 3. The first convolutional layer filters the 159 × 75 input activity image with 32 kernels of size 3 × 3. Then the first pooling layer uses “max pooling” to perform 2 × 2 subsampling. A second convolutional layer filters the 36 × 78 resulting image with 64 kernels of size 3 × 3. Then a second pooling layer uses “max pooling” to perform 2 × 2 subsampling. The third convolutional layer filters the 17 × 38 resulting image with 128 kernels of size 3 × 3. A third pooling layer uses “max pooling” to perform 2 × 2 subsampling. Then, a flatten layer transforms the output image of size 7 × 18 of the last pooling to a vector, which is then used as input to a dense layer using dropout. Finally, a second dense layer produces the output of the network.

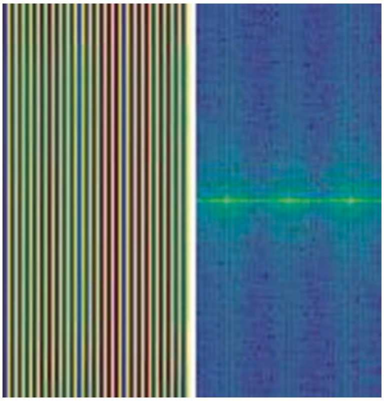

Fig. 2 Left: A signal image. Right: An activity image. For visualization purposes only, the activity image has been processed with a log transformation. Figure best viewed in color

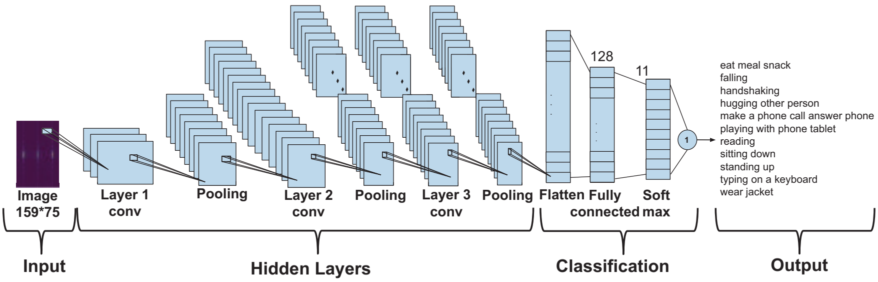

Fig. 3 The proposed CNN architecture

5 Experimental Results

For the experimental evaluation of the proposed approach, we used the PKU-MMD dataset (Liu et al. 2017a). This dataset aims to provide a large-scale benchmark, focusing on 3D human action understanding. It contains approx. 20 K action instances spanning into 5.4 M video frames and belonging to 51 action categories. A total of 66 human subjects have been involved, while video recordings have been captured from 3 camera angles, using the Microsoft Kinect v2 camera. Provided modalities are raw RGB video, depth sequences, infrared radiation captured by the Kinect, and extracted 3D positions of skeletons.

From this dataset we decided to use 11 classes which in our opinion are the most related to ADLs or events that could be recorded at a use case of, e.g., home monitoring. More specifically, these classes were: eat meal snack, falling, handshaking, hugging other person, make a phone call answer phone, playing with phone tablet, reading, sitting down, standing up, typing on a keyboard and wear jacket. As described in Sect. 4, we worked with the provided skeleton positions. For further evaluation, we also included experiments in all 51 classes of PKU-MMD. Sample signal and activity images from the 11 classes are illustrated in Fig. 4. There is a visual difference between these images, which may not be significant, but yet it allows the CNN to learn the differences between the two classes.

We have set $T_s = 158$ frames so as to prevent significant loss of information upon interpolation. The evaluation protocol we followed is as follows: We first performed experiments per camera position, namely, middle (M), left (L), and right (R). In this case, both training and testing sets were derived from the same position. Then, we performed cross-view experiments, where one position was used for training, while the other two were used for testing. The goal therein was to test the robustness of the proposed approach in abrupt changes of camera angle, which are expected to happen in real-life scenarios. In all cases we measured the accuracy of classification. Detailed results are depicted in Table 1.

As it may be observed, the proposed approach in the aforementioned case of 11 classes is able to achieve accuracy ranging from 0.75 to 0.85 when the camera angle remains unchanged (i.e., when samples from the same angle have been used for

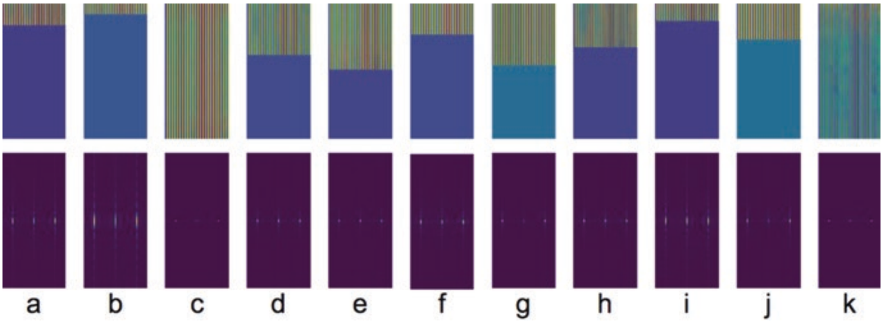

Fig. 4 Sample images from 11 actions of the PKU-MMD dataset that have been used throughout our experiments. Upper row: signal images. Lower row: activity images. (a) Eat meal snack; (b) falling; (c) handshaking; (d) hugging other person; (e) make a phone call/answer phone; (f) playing phone tablet; (g) reading; (h) sitting down; (i) standing up; (j) typing on keyboard; (k) wear jacket. Figure best viewed in color

Table 1 Experimental results of the proposed approach

	Train	M	M	M	L	L	L	R	R	R
Experiment	Test	M	L	R	M	L	R	M	R	L
Dataset	11	0.82	0.56	0.64	0.61	0.85	0.40	0.56	0.75	0.35
	51	0.73	0.29	0.28	0.25	0.55	0.11	0.29	0.73	0.12

M, L, and R denote the middle, left, and right camera angles, respectively. 11 and 51 are the numbers of classes considered for evaluation. Results indicate the achieved accuracy

both training and testing). Also, it achieves adequate performance in case two “neighboring” angles are used, e.g., M for training, L for testing, etc. In this case, accuracy ranges from 0.56 to 0.64. We noticed that when samples from the right camera position are used, a significant drop in performance is observed. Moreover, and as it has been expected, dramatic changes of camera angle, e.g., when L is used for training, R for testing, or vice versa, performance ranges between 0.35 and 0.40. Finally, when the whole set of 51 classes has been used, performance is acceptable only in cases where the same angle has been used for both training and testing; corresponding accuracies range from 0.55 to 0.73. In all other cases, a strong drop in performance is observed.

For the implementation of the CNN, we have used Keras (Chollet 2015) running on top of TensorFlow (Abadi et al. 2016). All data preprocessing and processing steps have been implemented in Python 3.6 using NumPy (http://www.numpy.org/) and SciPy (https://www.scipy.org/).

6 Discussion

In this paper we presented a methodology for the recognition of human actions which was based on a novel image representation of 3D human skeletal information and a novel convolutional neural network architecture. We used an image representation of a human action, which resulted from the concatenation of raw 1D signals corresponding to 3D motion of skeletal joints' coefficients and the application of the discrete Fourier transform to the resulting image.

We evaluated the proposed approach using a state-of-the-art and challenging dataset, which consisted of sequences corresponding to 51 human actions. These sequences had been captured with three Kinect v2 cameras, under different camera angles, and the skeletal joints of the human actors involved had been extracted. We performed experiments involving either one or two cameras (cross-view). We mainly focused on a subset of 11 actions which in our opinion are the most related to real-life ADLs. However, we also experimented with the whole dataset. Our initial results indicate that the proposed approach may be successfully applied to human action recognition in real-like conditions, yet a drop in performance is expected when camera angle would change.

Our future plans include:

(a) Investigation on methods for creating the signal image, possibly with the use of other types of sensor measurements such as wearable accelerometers, gyroscopes, etc.
(b) Investigation on image processing methods for transforming the signal image to the activity image. Toward this goal, transforms such as wavelets, discrete cosine transformation (DCT), etc. may be used.
(c) Exploitation of other types of visual modalities in the process, such as RGB and depth data.
(d) Evaluation of the proposed approach on several other public datasets.
(e) Application into a real-like or even real-live assistive living environment.

Acknowledgments We acknowledge support for this work by the project SYNTELESIS "Innovative Technologies and Applications based on the Internet of Things (IoT) and the Cloud Computing" (MIS 5002521) which is implemented under the "Action for the Strategic Development on the Research and Technological Sector," funded by the "Operational Programme Competitiveness, Entrepreneurship and Innovation" (NSRF 2014–2020) and co-financed by Greece and the European Union (European Regional Development Fund).

References

Abadi M et al (2016) TensorFlow: a system for large-scale machine learning. In: Proceedings of the USENIX symposium on operating systems design and implementation (OSDI)

Berretti S, Daoudi M, Turaga P, Basu A (2018) Representation, analysis, and recognition of 3D humans: a survey. ACM Trans Multim Comput Commun Appl (TOMM) 14(1s):16

Chollet F (2015) Keras. https://github.com/fchollet/keras

Du Y, Fu Y, Wang L (2015) Skeleton based action recognition with convolutional neural network. In: Proceedings of 3rd IAPR Asian conference on pattern recognition (ACPR). IEEE

Graves A, Mohamed AR, Hinton G (2013) Speech recognition with deep recurrent neural networks. In: Proceedings of IEEE international conference of acoustics, speech and signal processing (ICASSP)

Hou Y, Li Z, Wang P, Li W (2018) Skeleton optical spectra-based action recognition using convolutional neural networks. IEEE Trans Circuits Syst Video Technol 28(3):807–811

Jiang W, Yin Z (2015) Human activity recognition using wearable sensors by deep convolutional neural networks. In: Proceedings of ACM international conference on multimedia (MM)

Ke Q, An S, Bennamoun M, Sohel F, Boussaid F (2017) Skeletonnet: mining deep part features for 3-d action recognition. IEEE Signal Process Lett 24(6):731–735

Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems, pp 1097–1105

Lawton MP, Brody EM (1969) Assessment of older people: self-maintaining and instrumental activities of daily living. Gerontologist 9(3 Part 1):179–186

LeCun Y, Bottou L, Bengio Y, Haner P (1998) Gradient-based learning applied to document recognition. Proc IEEE 86(11):2278–2324

Li C, Hou Y, Wang P, Li W (2017) Joint distance maps based action recognition with convolutional neural networks. IEEE Signal Process Lett 24(5):624–628

Liu C, Hu Y, Li Y, Song S, Liu J (2017a) PKU-MMD: a large scale benchmark for continuous multi-modal human action understanding. In: Proceedings of ACM multimedia workshop (MM)

Liu M, Liu H, Chen C (2017b) Enhanced skeleton visualization for view invariant human action recognition. Pattern Recogn 68:346–362

Mathe E, Mitsou A, Spyrou E, Mylonas P (2018) Arm gesture recognition using a convolutional neural network. In: Proceedings of international workshop on semantic and social media adaptation and personalization (SMAP)

Srivastava N, Hinton G, Krizhevsky A, Sutskever I, Salakhutdinov R (2014) Dropout: a simple way to prevent neural networks from overfitting. J Mach Learn Res 15(1):1929–1958

Wang P, Li W, Ogunbona P, Wan J, Escalera S (2018a) RGB-D-based human motion recognition with deep learning: a survey. Comput Vis Image Underst 171:118–139

Wang P, Li W, Li C, Hou Y (2018b) Action recognition based on joint trajectory maps with convolutional neural networks. Knowl-Based Syst 158:43–53

Zhang Z (2012) Microsoft Kinect sensor and its effect. IEEE Multim 19(2):4–10