In: Proceedings of the 24th International Conference on Artificial Neural Networks (ICANN 2014),
LNCS 8681, pp. 89-96, Springer Heidelberg. Hamburg, DE, September 15-19, 2014.

Human Action Recognition with Hierarchical Growing Neural Gas Learning

German I. Parisi, Cornelius Weber, and Stefan Wermter

University of Hamburg - Department of Computer Science
Vogt-Koelln-Strasse 30, D-22527 Hamburg - Germany
{parisi,weber,wermter}@informatik.uni-hamburg.de
http://www.informatik.uni-hamburg.de/WTM/

Abstract. We propose a novel biologically inspired framework for the recognition of human full-body actions. First, we extract body pose and motion features from depth map sequences. We then cluster pose-motion cues with a two-stream hierarchical architecture based on growing neural gas (GNG). Multi-cue trajectories are finally combined to provide prototypical action dynamics in the joint feature space. We extend the unsupervised GNG with two labelling functions for classifying clustered trajectories. Noisy samples are automatically detected and removed from the training and the testing set. Experiments on a set of 10 human actions show that the use of multi-cue learning leads to substantially increased recognition accuracy over the single-cue approach and the learning of joint pose-motion vectors.

Keywords: human action recognition, growing neural gas, motion clustering, assistive system

1 Introduction

Recently, there has been a significant increase of research on ambient intelligence for the recognition of human activity in indoor environments [1]. In this context, the classification of human actions has proven to be a challenging task to accomplish with an artificial system, where the prompt recognition of potentially risky situations can represent a key issue. In the last four years, the prominent use of low-cost depth sensing devices such as the Kinect sensor led to a great number of vision-based applications using depth information instead of, or in combination with, color [2]. These methods generally extract and process motion from depth map sequences in terms of spatiotemporal patterns. Despite the reduced computational cost of processing depth maps instead of RGB pixel matrices, the robust recognition of articulated human actions remains an enticing milestone, also for machine learning and neural network-based approaches [3, 4, 11].

A promising scheme to tackle such a demanding task is the application of biological principles derived from the human visual system and its outperforming ability to process visual information. Studies in neurophysiology suggest a highly

flexible and adaptive biological system for processing visual cues at multiple levels for motion and action perception [5]. In fact, computational implementations of simplified biological models have shown motivating results on the recognition of dynamic pose-motion patterns [6]. In the biological visual system, dynamic scenes are analyzed in parallel by two separate channels [5]. The ventral channel processes shape features while the dorsal channel recognizes location and motion properties in terms of optic-flow patterns. Both channels are composed by hierarchies that extrapolate visual features with increasing complexity of representation. Specific areas of the visual system are composed of topographically arranged structures that organize according to the distribution of the inputs [13]. Input-driven self-organization allows to learn representations with an unsupervised scheme by adaptively obtaining the feature subspace. Under this assumption, the use of self-organizing maps (SOM) [7] has shown to be a plausible and efficient model for clustering visual patterns in terms of multi-dimensional flow vectors. With the use of extended models of hierarchical self-organization it is possible to obtain progressively generalized representations of sensory inputs and learn inherent spatiotemporal dependencies. While depth data-driven techniques currently represent a well-established approach for action recognition, the combination of the above-mentioned bio-inspired approach with this emerging sensory trend has not yet extensively developed.

In this work, we propose a novel learning framework for recognizing human full-body actor-independent actions. We first extract pose and motion features from depth map video sequences and then cluster actions in terms of prototypical pose-motion trajectories. Multi-cue samples from matching frames are processed separately by a two-stream hierarchical architecture based on growing neural gas (GNG) [8]. The GNG is an unsupervised incremental clustering algorithm extended from the SOM and the neural gas (NG) [9], able to dynamically change its topological structure to better represent the input space. Clustered trajectories from the parallel streams are combined to provide joint action dynamics. We process the samples under the assumption that action recognition is selective for temporal order [5]. Therefore, positive recognition of an action occurs only when trajectory samples are activated in the correct temporal order. In order to assign labels to clustered trajectories, we extend the GNG with two offline labelling functions. Noisy samples are automatically detected and removed from the training and the testing set to increase recognition accuracy. We present and discuss experimental results on a data set of 10 articulated actions.

2 Pose-Motion Estimation

The first step in the proposed framework constitutes the extraction of human body features from the visual scene. The use of skeleton model-based techniques for tracking action features in terms of a set of joints and limbs has shown good results, especially for approaches using depth maps [4]. On the other hand, joints are often subject to occlusion during the execution of actions. This may lead to significantly decreased reliability of estimated joints and subsequent tracking

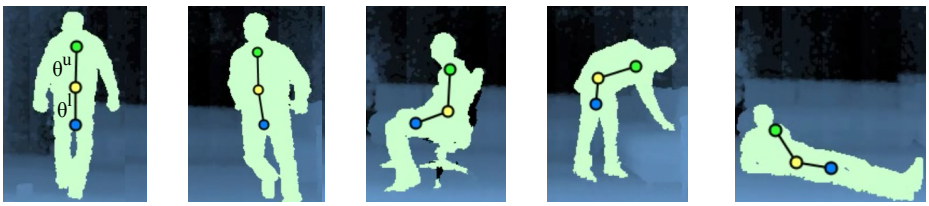

Fig. 1. Full-body representation for pose-motion extraction. We estimate three centroids C_1 (green), C_1 (yellow) and C_1 (blue) for upper, middle and lower body respectively. We compute the segment slopes (θ^u and θ^l) to describe the posture with the overall orientations of the upper and lower body.

inaccuracies. In our approach, we estimate spatiotemporal properties for representing actor-independent actions based on the estimation of body centroids that describe pose-motion features. This technique extrapolates significant action characteristics while maintaining a low-dimensional feature space and increasing tracking robustness for situations of partial occlusions. In [11], we proposed a simpler model to track a spatially extended body with two centroids and a global body orientation. The centroids were estimated as the centers of mass that follow the distribution of the main body masses on each posture.

We now extend our previous model to describe more accurately articulated actions by considering three body centroids (Fig. 1): C_1 for upper body with respect to the shoulders and the torso; C_2 for middle body with respect to the torso and the hips; and C_3 for lower body with respect to the hips and the knees. Each centroid is represented as a point sequence of real-world coordinates $C = (x, y, z)$. We then estimate upper and lower orientations θ^u and θ^l given by the slope angles of the segments $\overline{C_1C_2}$ and $\overline{C_2C_3}$ respectively. As seen in Fig. 1, θ^u and θ^l describe the overall body pose as the orientation of the torso and the legs, which allows to capture significant pose configurations in actions such as walking, sitting, picking up and lying down. We calculate the body velocity S_i as the difference in pixels of the centroid C_1 between two consecutive frames i and $i-1$. The upper centroid was selected based on the consideration that the orientation of the torso is the most characteristic reference during the execution of a full-body action [4]. We then estimate horizontal speed h_i and vertical speed v_i as in [11]. For each action frame i we obtain a pose-motion vector:

$$F_i = (\theta_i^u, \theta_i^l, h_i, v_i). \tag{1}$$

Each action A_j will be composed of a set of sequentially ordered pose-motion vectors $A_j := \{(F_i, l_j) : i \in [1..n], l_j \in L\}$, where l_j is the action label, L is the set of class labels, and n is the number of training vectors for the action j. This representation describes spatiotemporal properties of actions in terms of length-invariant patterns, particularly suitable for feeding into a neural network.

3 Neural Architecture

Our GNG-based architecture consists of three main stages: 1) detection and removal of noisy samples from the data set; 2) hierarchical processing of samples from matching frames by two separate processing streams in terms of prototypical trajectories; and 3) classification of action segments as multi-cue trajectories. An overall overview of the framework is depicted in Fig. 2. Before describing each stage, we will provide a theoretical background on the GNG.

3.1 The Growing Neural Gas

Neural network approaches inspired by biological self-organization such as self-organizing maps (SOM) [7] and neural gas (NG) [9] have been successfully applied to a great number of learning tasks. Their advantage lies in their ability to learn the important topological relations of the input space without supervision. Both methods use the vector quantization technique in which the neurons (nodes) represent codebook vectors that encode a submanifold of the input space. The number of nodes in the SOM and the NG is fixed beforehand and cannot be changed over time. The growing neural gas (GNG) proposed by Fritzke [8] represents an incremental extension of these two networks. The GNG algorithm has the ability to create connections between existing nodes and to add new nodes in order to effectively map the topology of the input data distribution.

A growing network starts with a set N of two nodes at random positions w_a and w_b in the input space. At each iteration, the algorithm is given an input signal ξ according to the input distribution $P(\xi)$. The closest unit s_1 and the second closest unit s_2 of ξ in N are found and if the connection (s_1, s_2) does not exist, it is created. The local error of s_1 is updated by $\Delta E_{s_1} = \|\xi - w_{s_1}\|^2$ and w_{s_1} is moved towards ξ by a fraction ϵ_b. The weight of all the topological neighbors of s_1 are also moved towards ξ by a fraction ϵ_n. If the number of given inputs is a multiple of a parameter λ, a new node is inserted halfway between those two nodes that have maximum accumulated error. A connection-age-based mechanism leads to nodes being removed if rarely used. The algorithm stops when a criterion is met, i.e. some performance measure or network size. For the complete training algorithm see [8].

3.2 GNG-based Noise Detection

Pose-motion vectors F (Eq. 1) are susceptible to tracking errors due to occlusion or systematic sensor errors, which may introduce noise in terms of values highly detached from the dominating point clouds. We consider inconsistent changes in body velocity to be caused by tracking errors rather than actual motion. Therefore, we remove noisy motion samples to create smoother inter-frame transitions. First, the network G^N is trained using only the motion samples from F. Second, the training motion samples are processed again to obtain the set of errors E from the trained network, which contains the distance from the closest unit $d(s_1)$ for each motion sample. We then calculate the empirically defined threshold that

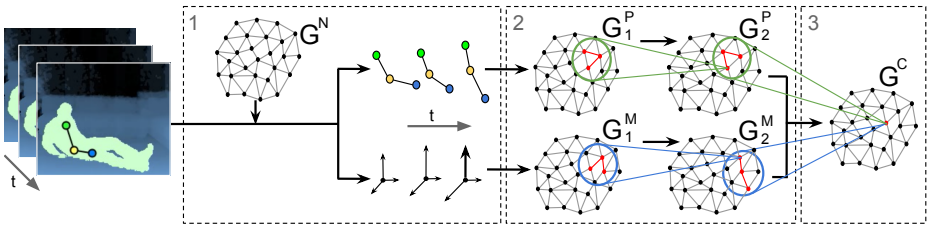

Fig. 2. Three-stage framework for the GNG-based processing of pose-motion samples: 1) detection and removal of sample noise; 2) hierarchical processing of pose-motion trajectories in two parallel streams; 3) classification of multi-cue trajectories.

considers the distribution of the samples as $th = 2\sigma(E)\sqrt{\mu(E)}$, where $\sigma(E)$ is the standard deviation of E and $\mu(E)$ is its mean. For each motion sample, if $d(s_1) > th$, then the sample is considered to be noisy and its associated vector F_i is removed from the training set. We then obtain a new denoised training set from which we create two distinct sets with sequentially ordered pose and motion features, formally defined as $P = \{(\theta^u, \theta^l)\}$ and $M = \{(h, v)\}$ respectively.

3.3 Hierarchical Learning

The second stage is composed of a two-stream hierarchy for processing pose-motion cues separately. Each stream consists of two GNG networks that process prototypical samples under the assumption that recognition is selective for temporal order [5]. Therefore, sequence selectivity results from the use of node trajectories to describe spatiotemporal action segments.

We first train the networks G_1^P and G_1^M with the denoised training sets P and M respectively. After this training phase, chains of codebook nodes for training samples produce time varying trajectories on each network. For a given trained network G and a training set X, we define the set of labelled trajectories as:

$$T(G, X) := \{(s(x_{i-1}), s(x_i), l(x_i)) : l(x_i) = l(x_{i-1}), i \in [2..n(X)]\}, \qquad (2)$$

where the function $s(x)$ returns the closest node s_1 of sample x in G, $l(x) \in L$ returns the label of x, and $n(X)$ is the number of samples in X. We compute the sets $T(G_1^P, P)$ and $T(G_1^M, M)$, for convenience denoted as T^P and T^M, and use them as input for the networks G_2^P and G_2^M respectively. This step produces a mapping with temporally ordered prototypes from consecutive samples. We now couple the outputs from both networks to create a set of multi-cue trajectories:

$$\Omega := \{(T(G_2^P, T_k^P), T(G_2^M, T_k^M), l_j) : k \in [2..g]\}, \qquad (3)$$

where g is the number of elements in T^P and T^M and $l_j \in L$ is the label associated with the multi-cue trajectory. We finally feed the set of pairs into G^C and process Ω again to obtain the set with the mapping of codebook nodes corresponding to multi-cue pairs from consecutive trajectories.

3.4 Action Classification

For assigning labels to clustered trajectories with G^C, we extend the GNG algorithm with two offline labelling functions: one for the training phase and one for predicting the label of unseen samples at recognition time. These labelling techniques are considered to be offline since we assume that the labelled training pairs (ω, l_j) with $\omega \in \Omega$ and $l_j \in L$ are stored in F (Eq. 1). First, we define a labelling function $l : N \rightarrow L$ where N is the set of nodes and L is the set of class labels. According to the minimal-distance strategy [14], the sample $\omega_k \in \Omega$ adopts the label l_j of the closest $\omega \in \Omega$:

$$l(\omega_k) = l_j = l(\arg\min_{\omega \in \Omega} \|\omega_i - \omega\|^2). \tag{4}$$

At recognition time, our goal is to classify unseen samples as pose-motion trajectory prototypes (Eq. 4). Therefore, we define a prediction function $\varphi : \Omega \rightarrow L$ inspired by a single-linkage strategy [14] in which a new sample ω_{new} is labelled with l_j associated to the node n that minimizes the distance to this new sample:

$$\varphi(\omega_{new}) = \arg\min_{l_j}(\arg\min_{n \in N(l_j)} \|n - \omega_{new}\|^2). \tag{5}$$

The adopted labelling techniques have shown to achieve best classification accuracy among other offline labelling strategies [14].

4 Results and Discussion

We collected a data set of 10 actions performed by 13 different actors with a normal physical condition. To avoid biased execution, the actors had not been explained how to perform the actions. The data set contained the following periodic actions (PA) and goal-oriented actions (GA):

- PA: Standing, walking, jogging, sitting, lying down, crawling (10 minutes each);
- GA: Pick up object, jump, fall down, stand up (60 repetitions each).

For the data collection we monitored people in a home-like environment with a Kinect sensor installed 1,30 meters above the ground. Depth maps were sampled with a VGA resolution of 640x480, an operation range from 0.8 to 3.5 meters and a constant frame rate of 30Hz. To reduce sensor noise, we sampled the median value of the last 3 estimated points. Body centroids were estimated from depth map sequences based on the tracking skeleton model provided by OpenNI. Action labels were manually annotated from ground truth of sequence frames. We divided the data equally into training set and testing set: 30 sequences of 10 seconds for each periodic action and 30 repetitions for each goal-oriented action.

We used the following GNG training parameters: learning step sizes $\epsilon_b = 0.05$, $\epsilon_n = 0.005$, node insertion interval $\lambda = 350$, error reduction constant $\alpha = 0.5$, and error reduction factor $d = 0.995$ (see [8] for details). Maximum network size and the number of iterations varied for each GNG and were experimentally adjusted based on the network performance for different input distributions.

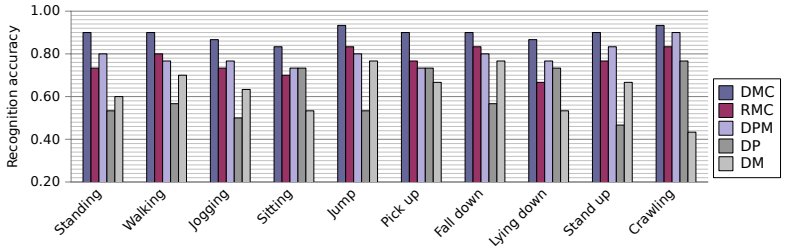

Fig. 3. Evaluation on recognition accuracy under 5 different processing conditions: Denoised multi-cue (DMC), denoised pose-motion vector (DPM), raw multi-cue (RMC), denoised pose-only (DP), and denoised motion-only (DM).

We evaluated the recognition accuracy of the framework under 5 different processing conditions: denoised multi-cue (DMC) and raw multi-cue (RMC) samples, denoised "pose only" (DP) and denoised "motion only" (DM) samples, and joint pose-motion vectors (DPM) as defined in Eq. 1 processed by a single stream. As seen in Fig. 3, the use of denoised multi-cue trajectory prototypes obtains the best average recognition result (89%). The removal of noise from the data sets increases average recognition accuracy by 13%. The DMC approach exhibits average improvements over DP and DM of 28% and 26% respectively.

Our results also show that DMC exhibits increased accuracy over the learning of joint pose-motion vectors (DPM) by 10%. This is partly due to the fact that the DPM approach forces the early convergence of the networks in the joint pose-motion space, while DMC and RMC learn a sparse representation of disjoint pose-motion prototypes that are subsequently combined to provide joint action dynamics. The reported results for actor-independent action recognition were obtained with low latency providing real-time characteristics.

5 Conclusion and Future Work

We presented a novel learning framework for the robust recognition of human full-body actions from pose-motion cues. Multi-cue trajectories from matching frames were processed separately by a hierarchical GNG-based architecture. This approach captures correlations between pose and motion prototypes to provide joint action dynamics. Experiments on a data set of 10 actions have shown that the proposed multi-cue strategy increases recognition accuracy over a single-cue approach and joint pose-motion vectors.

While the use of multi-cue learning has previously shown compelling results for robust action recognition [3, 10, 12], this approach is also supported by neural evidence. Therefore, the obtained results motivate further work in two directions. First, the evaluation of our framework on a wider number of actions and more complex pose-motion characteristics, e.g. including arm movements and hand gestures. Second, an extended neural architecture based on a more biologically plausible model of the visual system.

Acknowledgements. This work was supported by the DAAD German Academic Exchange Service (Kz:A/13/94748) - Cognitive Assistive Systems Project, and by the DFG German Research Foundation (grant #1247) - International Research Training Group CINACS (Cross-modal Interaction in Natural and Artificial Cognitive Systems).

References

1. Roy, P.C., Bouzouane, A., Giroux, S., Bouchard, B.: Possibilistic Activity Recognition in Smart Homes for Cognitively Impaired People, Applied Artificial Intelligence: An International Journal 25:883–926, Taylor and Francis Group (2011)
2. Suarez, J., Murphy, R.: Hand Gesture Recognition with Depth Images: A review. In: IEEE Int. Symposium on Robot and Human Interactive Communication, pp. 411–417, France (2012)
3. Xu, R., Agarwal, P., Kumar, S., Krovi, V.N., Corso, J.J.: Combining Skeletal Pose with Local Motion for Human Activity Recognition. In: Perales, F.J., Fisher, R.B., Moeslund, T.B. (eds.): AMDO 2012. LNCS, vol. 7378, pp. 114–123, Springer-Verlag Berlin Heidelberg (2012)
4. Papadopoulos, G.Th., Axenopoulos, A., Daras, P.: Real-Time Skeleton-Tracking-Based Human Action Recognition Using Kinect Data. In: Gurrin, C., Hopfgartner, F., Hurst, W., Havard, J., Lee, H., O'Connor, N. (eds.): MultiMedia Modeling. LNCS, vol. 8325, pp 473–483, Springer International Publishing (2014)
5. Giese, M.A., Poggio, T.: Neural Mechanisms for the Recognition of Biological Movements. Nature Reviews Neuroscience, 4:179–192 (2003)
6. Escobar, M., Kornprobst, P.: Action Recognition with a Bioinspired Feedforward Motion Processing Model: The Richness of Center-Surround Interactions. In: Forsyth, D., Torr, P., Zisserman, A. (eds.): ECCV 2008. LNCS, vol. 5305, 2008, pp. 186–199, Springer Berlin Heidelberg (2008)
7. Kohonen, T.: Self-organizing Maps. In: Series in Information Sciences, vol. 30, Springer Heidelberg (1995)
8. Fritzke, B.: A Growing Neural Gas Network Learns Topologies. In: Advances in Neural Information Processing Systems 7, pp.625–632, MIT Press (1995)
9. Martinetz, T., Schluten, K.: A "neural-gas" network learns topologies. In: Artificial Neural Networks, pp. 397–402, Elsevier (1991)
10. Jiang, Z., Lin, Z., Davis, L.S.: Recognizing Human Actions by Learning and Matching Shape-Motion Prototype Trees. In: IEEE Transactions on Pattern Analysis and Machine Intelligence 31(3):533–547 (2012)
11. Parisi, G.I., Wermter, S.: Hierarchical SOM-based Detection of Novel Behavior for 3D Human Tracking. In: IEEE Int. Joint Conf. on Neural Networks (IJCNN) pp. 1380–1387, USA (2013)
12. Parisi, G.I., Barros, P., Wermter, S.: FINGeR: Framework for Interactive Neural-based Gesture Recognition. In: European Symposium of Artificial Neural Networks (ESANN), 443–447, Belgium (2014)
13. Miikkulainen, R., Bednar, J. A., Choe, Y., Sirosh J.: Computational Maps in the Visual Cortex. Springer New York (2005)
14. Beyer, O., Cimiano, P.: Online Labelling Strategies for Growing Neural Gas. In: Yin, H., Wang, W., Rayward-Smith, V. (eds.): IDEAL 2011. LNCS, vol. 6936, pp. 76–83, Springer-Verlag Berlin Heidelberg (2012)