ESANN 2015 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 22-24 April 2015, i6doc.com publ., ISBN 978-287587014-8. Available from http://www.i6doc.com/en/.

Dynamic Gesture Recognition Using Echo State Networks

Doreen Jirak, Pablo Barros and Stefan Wermter

University of Hamburg - Department of Informatics, Vogt-Koelln-Str. 30
22527 Hamburg - Germany

Abstract. In the last decade, training recurrent neural networks (RNN) using techniques from the area of reservoir computing (RC) became more attractive for learning sequential data due to the ease of network training. Although successfully applied in the language and speech domains, only little is known about using RC techniques for dynamic gesture recognition. We therefore conducted experiments on command gestures using Echo State Networks (ESN) to investigate both the effect of different gesture sequence representations and different parameter configurations. For recognition we employed the ensemble technique, i.e. using ESN as weak classifiers. Our results show that using ESN is a promising approach for dynamic gesture recognition and we give indications for future experiments.

1 Introduction

In our everyday life, we rely heavily on different gestures to underpin the meaning of what we are saying. Pointing gestures used in accompanying commands like ”Give me that object” or even replacing that sentence are referred to as deictic gestures, while gestures produced along with story-telling e.g. describing the shape of an object, are termed iconic gestures. Both types have dynamic character to convey the gesture meaning, as opposed to postures with specific finger configurations. Often, modeling gestures is a trade-off between capturing the temporal dynamics using probabilistic approaches or representing them in a more bio-inspired fashion, which make use of the hierarchical structure found in the visual cortex. A bridge to both approaches is provided by Recurrent Neural Networks (RNN). Standard RNN as the simple Elman network or more complex networks like Multiple Timescale RNN use gradient-based training applying the Backpropagation Through Time algorithm (BPTT). That way, the network establishes a memory, hence capturing long-term-dependencies as is necessary for sequential data. However, this method suffers from several drawbacks, namely slow convergence, poor local minima and stability problems in terms of bifurcations [1]. An optimized procedure for training is provided by Backpropagation-Decorrelation [2]. In this work we focus on dynamic gestures and aim to investigate Reservoir Computing (RC) methods for gesture recognition. To achieve this, we concentrate on recent models in the area of RC, in particular the Echo State Networks [3] (ESN). From the other direction, there is research about generating movements using ESNs. These approaches incorporate the feedback matrix for learning e.g. arm movements [4]. The focus is not on classification and hence even the role of features for input is not significant.

Combining the fact that gestures do play an important role for communication with the benefits ESNs provide in terms of their implementation and learning (classification and generation), we set up experiments comprising a simple and a complex feature set to investigate the gesture representation on the ESN learning. Further, we explore different parameters of the ESN to report under which configurations the task of gesture recognition can be applied.

2 ESN computations

A dynamical system is described in terms of its state evolution or state trajectory, which is coded in the reservoir neurons. Let x be the state space and u the input space. Evolution of x in a leaky-integrator ESN is given as:

$$\dot{x} = c^{-1}\Bigg(- \alpha x + f\big(W_{in}u + W_{res}x + W_{back}y\big)\Bigg), \tag{1}$$

where f is a network activation function, c is a time constant and $\alpha \in (0, 1]$ is the leakage rate, quantifying the amount of memory in the network. The matrices connect the different layers of the network, hence W_{in} is of size $input \times reservoir$, W_{res} is the square matrix assigning connectivity between reservoir neurons including self-reference, and W_{back} is accordingly determined by the size of the output layer and reservoir size, feeding back the information from the output. The input to the reservoir is represented by u, while y denotes the output. Discretization of equation 1 yields the following:

$$\mathbf{x}(n+1) = \mathbf{r}\mathbf{x}(n) + f(W_{in}\mathbf{u}(n+1) + W_{res}\mathbf{x}(n) + W_{back}\mathbf{y}(n) + \nu(n+1), \tag{2}$$

where $\mathbf{r} = 1-\alpha$ is referred to as the retainment factor and ν is a noise term added for system stability [5]. The ease of training is due to the computation of a linear regression on the weight matrix W_{out}, which connects the reservoir with the output. Computation of W_{out} is often performed with Thikonov regularization to bound growth of output weights: [1]

$$W_{out} = YX^T(XX^T + \kappa\mathbf{I})^{-1} \tag{3}$$

with κ being the regularization coefficient and $\mathbf{I}$ being the identity matrix. Omitting κ is known as the Wiener-Hopf solution. Usage of the pseudo-inverse is mathematically equivalent, thus the equation can be rewritten and easily implemented as:

$$W_{out} = (YX^{\dagger})^T \tag{4}$$

3 Classification Experiments

As our purpose is to establish a visual communicative Human-Robot Interaction (HRI) scenario using gestures, we chose deictic gestures. For initial experiments

[1] We use bold- and capital letters to underline the matrix notation for linear regression

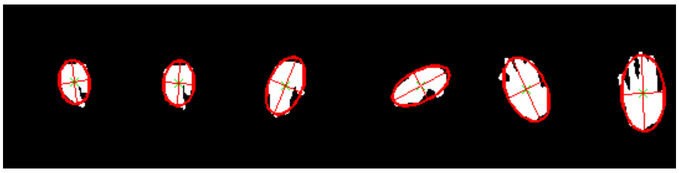


Fig. 1: Visualization of hand orientation computation for subsequent frames of a stop gesture. After hand extraction, an ellipse is fitted to the global shape to compute the orientation.

we defined five gestures: 'Stop', 'Point left', 'Point right', 'Turn around', and 'Drawing a circle'. Gesture sequences were recorded either with the NAO camera or a webcam with the subject standing frontal to the camera. Every gesture is performed with only one hand. The distance to the sensor is constrained so as to resemble a natural conversation.

For hand detection we converted the images into YCbCr colour space to account for luminance changes. Skin-tone regions were filtered with a threshold method following [6]. Morphological operations like hole-filling were applied to achieve consistent large pixel blocks in the image followed by a connected-component analysis. From the resultant remaining blobs, a threshold was applied to get the hands. As we consider dynamic gestures, we rather focus on simple features, i.e. centroid computation and the according hand orientation, as we do not want to recognize specific hand postures. In addition, the performed gestures hardly contain special finger movements.

We also set up experiments using features extracted from a Multichannel Convolutional Neural Networks (MCCNN) [7]. Computations in standard convolutional neural networks (CNN) resemble the hierarchical processing stages in the visual cortex, where neurons in lower brain areas code for rather simple features like edges, but get tuned to more specific properties like shape and motion in the upper brain areas. The underlying major operations in a CNN comprise convolution of images and subsequent max-pooling. Here, we use an extended version using a cubic kernel for spatial feature detection applied to three input channels that are the grayscale images and Sobel-filtered images in x- and y-direction (see Figure 2). As a result we get a 70-dimensional feature vector for network input.

The data was normalized into the range $[-1;1]$ and fed into the network with *tanh* activation and leaky neurons. The input and reservoir matrices are randomly initialized in the range $[-1;1]$. As we do not use the ESN for signal generation, W_{back} and ν were set to 0. To ensure nonlinearity in the reservoir, the input scale factor was set to 1.5 as in comparable work with reservoirs. The spectral radius was set below unity. We also set up different sizes of ensemble ESNs up to 500, which resulted in 10 ensemble sets. In detail, 500 single ESNs with the given parameters were initialized and then merged into ensembles up to one reservoir or, respectively, one big ensemble. The 103 gesture sequences were split into 2/3 train- and 1/3 test sets, randomly subsampled without replacement

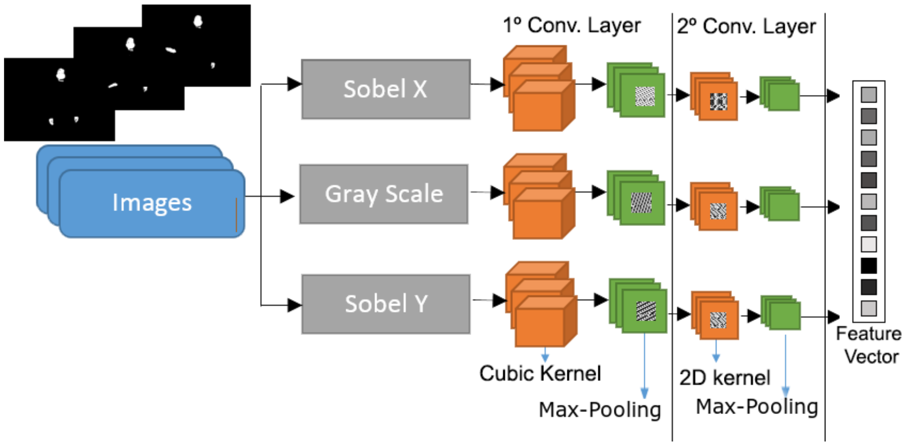
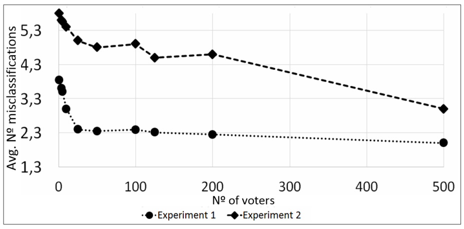


Fig. 2: The MCCNN architecture used for feature extraction. Each image from the gesture sequences is passed through three channels, each one containing two convolution layers. A max-pooling operator is applied to compress the data. Finally, a 1D vector representing the image is obtained.

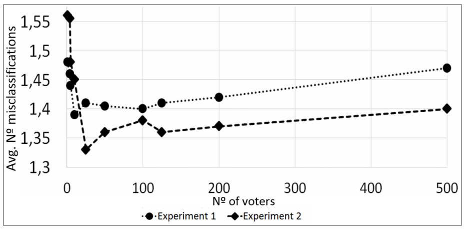
from the whole data to ensure data diversity in both sets. We used and extended the classification idea presented first for a speaker identification task [8]. We decided using the following scheme: Let $\mathbf{s}_n^i$ denote the i-th state sequence from a training sample set with length l_{train}. So, $\mathbf{s}_n^i$ is a vector of the design matrix hosting both the excitation $\mathbf{u}_i(n)$ and neuron state activations $\mathbf{x}_i(n)$. Further, let h_c^i be the hypothesis that the i-th sample belongs to the c-th class, where $1 \leq c \leq 5$. For the training we therefore determined $l_{train} \times c$ binary matrices coding with value 1 the target class and set to value 0 else. Following [8], we chose a small value ℓ, which $\mathbf{s}_i(n)$ will then be equidistantly subdivided with. From the resulting subsequences only the states lying in the intervall $\delta * l_{train}^i/\ell$, with $1 \leq \delta \leq \ell$, ℓ is an integer determining the partition of the states, and l_{train}^i is the length of the i-th sample, are further used as they carry state information picked up at different points while network execution. This results in a vector set for training the according regression weights. In addition, Jaeger and collegues [3] also proposed a voting scheme, known for instance from neural ensembles performing classification using weak classifiers. The idea is to use small leaky ESNs for performance but to retain their output merged into a vote for the best-fitting hypothesis given the test input. We investigate the effect of the ESN-ensembles for both feature sets.

4 Results

We conducted experiments on both feature sets with different values of $\alpha = \{0.1, 0.2, 0.3\}$, $\ell = \{3, ..., 6\}$ and number of reservoir neurons per ensemble $\{3, .., 10\}$. For each parameter configuration we ran 30 trials. When only increasing the number of reservoir neurons ($\alpha = 0.2$, $\ell = 3$), we could detect overfitting in both feature sets. Globally, the average number of misclassifications is rather low for the complex feature set, i.e. across the trials the training error is (near to) 0, also across different ensemble sets. The test results showed in the worst case for overfitting on average 1.5 misclassifications. In comparison, the simple feature set showed highly varying misclassifications across the experiments, specifically the 'circle' and 'turn around' gestures and also 'turn around' and the pointing gestures are incorrectly recognized. The effect changes slightly when also increasing ℓ, but having both high values of reservoir neurons and ℓ will result in divergence of training- and test error. Using the voting scheme for the simple feature set we could detect no benefit. The best results here in terms of misclassification could be achieved combining the ESN into one reservoir. In contrast, using ensemble ESNs for the complex feature set could show that on both extremes (only single voters vs. one reservoir) the mean classification is worse than using reservoir ensembles. Variation of α shows no significant effect on our data. We assume that with a bigger database comprising more motion gestures α will have an influence, i.e. higher values will then lead to a lower retainment in the network, thus the network can respond to new stimuli faster.



(a) Evaluation of the experiments using different ensemble sizes (diamonds, dots) for the simple feature set. Upper graph: $\alpha = 0.2$, $\ell = 3$ and number of reservoir neurons=4. Right:$\alpha = 0.2$, $\ell = 3$ and number of reservoir neurons=9. The dashed line shows the trend for misclassification. Here, the simple feature set works best when using one ensemble or, respectively, a single reservoir.



(b) Evaluation of the experiments using different ensemble sizes (dots) for the complex feature set with the same parameter configuration as described above. The trend highlights the decrease of misclassifications when using ESNs in ensembles. The misclassification rate increases when considering individual ESNs or a single reservoir.

5 Discussion and Conclusion

We have shown that usage of ESN is a viable method for gesture recognition. All gestures are of different lengths, which is useful as to cope with intra- and inter-subject variances in gesture performance when scaling up the ESN to our extended recorded gesture vocabulary database comprising 14 commanding gestures, performed with one and with both hands. With our approach, communication with gestures can be performed quite intuitive, as we do not rely on additional devices like markers or gloves. We also showed the effect of gesture recognition using two feature sets of varying complexity applied to ESN-ensembles. For the simple feature set, our results highlight that using single reservoirs is superior to the ensemble concept. In contrast, the ensembles could show better recognition results than achieved with individual voters or a big ensemble. Based on the work, we conclude to extend the established classification with the generation capabilities mentioned above, as we want to establish also an *active* role for robots in HRI.

References

[1] K. Doya. Bifurcations in the learning of recurrent neural networks. In *Circuits and Systems, 1992. ISCAS '92. Proceedings., 1992 IEEE International Symposium on*, volume 6, pages 2777–2780 vol.6, May 1992.

[2] Jochen J. Steil. Online reservoir adaptation by intrinsic plasticity for backpropagation-decorrelation and echo state learning. *Neural Networks*, 20(3):353 – 364, 2007.

[3] Herbert Jaeger. *Tutorial on training recurrent neural networks, covering BPPT, RTRL, EKF and the" echo state network" approach.* GMD-Forschungszentrum Informationstechnik, 2002.

[4] Kim Joris Bostroem, Heiko Wagner, Markus Prieske, and Marc de Lussanet. Model for a flexible motor memory based on a self-active recurrent neural network. *Hum Mov Sci*, 32(5):880–898, Oct 2013.

[5] Mantas Lukosevicius and Herbert Jaeger. Survey: Reservoir computing approaches to recurrent neural network training. *Computer Science Review*, 3(3):127–149, 2009.

[6] P. Kakumanu, S. Makrogiannis, and N. Bourbakis. A survey of skin-color modeling and detection methods. *Pattern Recogn.*, 40(3):1106–1122, 3 2007.

[7] Pablo Barros, Sven Magg, Cornelius Weber, and Stefan Wermter. A multichannel convolutional neural network for hand posture recognition. In S. Wermter, C. Weber, W. Duch, T. Honkela, P. Koprinkova-Hristova, S. Magg, G. Palm, and A. Villa, editors, *Artificial Neural Networks and Machine Learning ICANN 2014*, volume 8681 of *Lecture Notes in Computer Science*, pages 403–410. Springer International Publishing, 2014.

[8] Herbert Jaeger, Mantas Lukoševičius, Dan Popovici, and Udo Siewert. Optimization and applications of echo state networks with leaky- integrator neurons. *Neural Networks*, 20(3):335 – 352, 2007. Echo State Networks and Liquid State Machines.