Extended Hopfield Network for Sequence Learning: Application to Gesture Recognition

Andre Maurer, Micha Hersch and Aude G. Billard

Ecole Polytechnique Fédéra de Lausanne (EPFL)
Swiss Federal Institute of Technology Lausanne
Autonomous Systems Laboratory
CH-1015 Lausanne, Switzerland Email: aude.billard@epfl.ch

Abstract. In this paper, we extend the Hopfield Associative Memory for storing multiple sequences of varying duration. We apply the model for learning, recognizing and encoding a set of human gestures. We measure systematically the performance of the model against noise. Finally, we compare the performance of the model with that of an encoding using Hidden Markov Models.

1 Introduction

The work we present here is part of a research agenda, that aims at modeling the neural correlates of human ability to learn new motions through the observation and replication of other's motions [1, 2]. In this paper, we investigate the use of a biologically plausible mechanism for recognizing, classifying and reproducing gestures.

Associative memories based on Hebbian learning, such as the *Hopfield network*, are interesting candidates to model the propensy of biological systems to encode and learn complex sequences of motion [3]. The Hopfield network is known predominantly for its ability to code static patterns. However, recent work extended the Hopfield model to encode a time series of patterns [4]. In the present work, we extend this model to encode *several* sequences of patterns in the same model. While the capacity of such RNN models have been studied at length in simulation [5, 6], there has been yet little work demonstrating their application to the storage of real data sequences. Here, we validate the model for encoding human gestures and measure the performance of the model in the face of a large amount of noise.

Fundamental features of human ability to imitate new motions are a) the ability to robustly recognize gestures from partially occluded demonstrations (this is tightly linked to our ability to predict the dynamics of the motion from observing only the onset of the motion); and b) to store and reproduce a generalized version of the motion, that encapsulates only the key features of the motion. We show that the model can successfully reproduce these two key features.

2 Experimental Set-up

Figure 1 shows a schematic of the data flow across the complete architecture. During the learning phase, the system is trained on a set of gestures. During

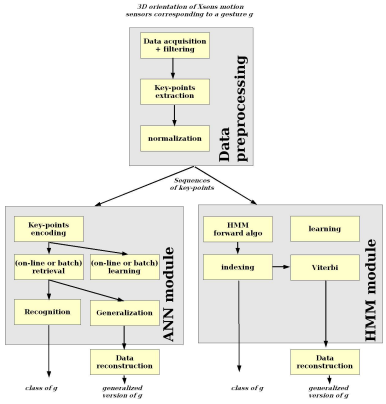

Fig. 1. Schematic of the data flow across the complete architecture.

the testing phase, the system is evaluated on its ability to both recognize and regenerate the data. Input to the system consists of the kinematic data of human motion. The data is first preprocessed to smooth and normalize the trajectories, as well as to reduce the dimensionality of the dataset to a subset of keypoints. The time series of keypoints is, subsequently, encoded and classified in either a set of Artificial Neural Networks (ANNs) or a set of Hidden Markov Models (HMMs). Learning in ANNs and HMMs result in the storage of a generalized form of the demonstrated gestures. The system outputs either the class of the gesture g or the generalized for of the gesture corresponding to the class g.

Data acquisition and preprocessing: Data consist of 45 gestures, composed of the 4 angular trajectories of the arm (shoulder abduction-adduction, flexion-extension and humeral rotation, and elbow flexion-extension) of 8 demonstrators during 5 repetitions of drawing the stylized letters A, B, C, D, E, (see figure 2).

Each trajectory is smoothed using a 1D local Gaussian filter of size 7 (*filtering* module). From those trajectories, we extract a set of p key-points $\{\theta_i^a, t_i^a\}$ (a=1..4,i=..3). A key-point is either the first or last element of the trajectory or an inflexion point (zero velocity). Such a segmentation aims at extracting the correlations between the different joint trajectorie. The duration of the whole trajectory is normalized so that two gestures belonging to the same class but performed at different speed are encoded into similar sequences.

Pattern encoding: Each element of the input sequence $\{t_i^a, \theta_i^a\}$ ($a = 1, .., 4$, $i = 1, .., p$, p being the length of the sequence) is encoded in a pair of real values, as follows:

$$(t_{\tilde{i}}^{\tilde{a}}, \theta_{\tilde{i}}^{\tilde{a}}) \rightarrow \tilde{x}$$

where $\tilde{x}$ is an $M \times N$ matrix. The rows of matrix $\tilde{x}$ encode the time $t_{\tilde{i}}^{\tilde{a}}$ and the columns of matrix $\tilde{x}$ encode the angle $\theta_{\tilde{i}}^{\tilde{a}}$. In order to preserve the notion

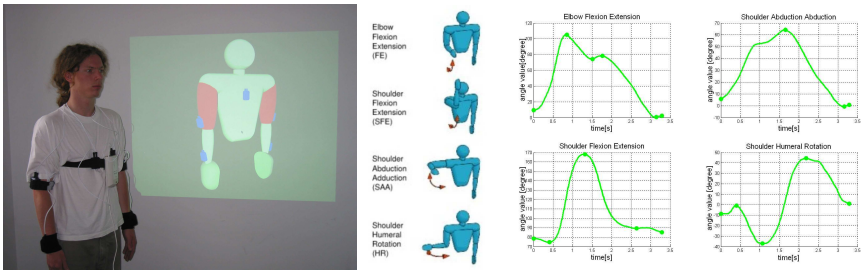

Fig. 2. The demonstrator's motions are recorded by a set of Xsens motion sensors, attached to the torso, upper and lower arms (left). The information is then used to reconstruct the trajectories of 4 joint angles of the arm (middle). (Right:) Raw data trajectories. Each subplot correspond to the trajectory of one of the 4 joint angle. Circles represent the extracted key-points

of neigbourhood across inputs we encode a pair $(t_{\tilde{i}}^{\tilde{a}}, \theta_{\tilde{i}}^{\tilde{a}})$ using a 2D gaussian distribution function centered on $\boldsymbol{\mu} = (\mu_t, \mu_\theta)^T$ with standard deviation $\boldsymbol{\sigma} = (\sigma_t, \sigma_\theta)^T$.

2.1 ANN module

The general topology of the network is presented in Figure 3. Inputs to the network are sequences of key-points. The sequences are stored in a series of Hopfield networks linked to one another through the matrix of weights W. Each sequence is then classified according to a set of classes $c = 1, .., C$ and C, represented by a set of neurons y_c.

The activity $x_i^{\mu,a}$ of each neuron in each layer is comprised between $[0, 1]$ (i=1..$M \cdot N$, $M\dot{N}$ is the total number of neurons, $\mu = 1, .., P$ the sequence, $a = 1..4$ the angular trajectory). The weights $w_{ij}^{\mu,a}$ across two elements of each layer, i.e. across neuron $x_i^{\mu,a}$ of layer μ and neuron $x_j^{\mu+1,a}$ of layer $\mu + 1$ are normalized and bounded between [0..1].

Learning process The learning rule for updating elements of W is a modification of the one presented in [4], so as to be able to store several sequences rather than just one, as well as to using a non-overlapping encoding with $x_i = [0; 1]$, as opposed to $x_i = \pm 1$.

$$w_{ij}^{\mu,a} = \sum_s x_i^{\mu,s,a} x_j^{\mu+1,s,a} \tag{1}$$

s is the indice of the sequence.

When learning a gesture s belonging to a class $\tilde{c}$, we set the output neurons $y_c = 0 \ \ \forall c \neq \tilde{c}$ and $y_{\tilde{c}} = 1$. Updating the elements of the recognition matrix J is done according to:

$$\Delta J_{i,c}^{\mu,a} = x_i^{\mu,s,a} y_c^{s,a} \tag{2}$$

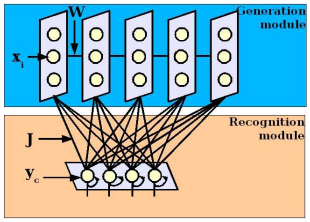

Fig. 3. Network topology. The weight matrix W (generation module) connects all neurons from one layer to all neurons of the next layer. Each output neuron y_c corresponds to a class c of gestures. The weights of the matrix J (recognition module) are set so that the output neuron $y_{\tilde{c}}$ is maximal when a sequence of class $\tilde{c}$ is generated. Each angular trajectory is encoded in a separate network (i.e 4 networks are used).

where i=1..$M \cdot N$.

Retrieval process: In order to retrieve the generalized form of the sequence associated with a given class, we activate one of y_c neuron and then reactivate the neurons in each layer of the extended Hopfield in sequence. That is, we update each neuron $x_i^{\mu+1,a}$ according to:

$$x_i^{\mu+1,a} = \sum_j w_{ij}^{\mu,a} \cdot x_j^{\mu,a} \tag{3}$$

Figure 4 shows the history of the network state after the retrieval of a sequence of four elements.

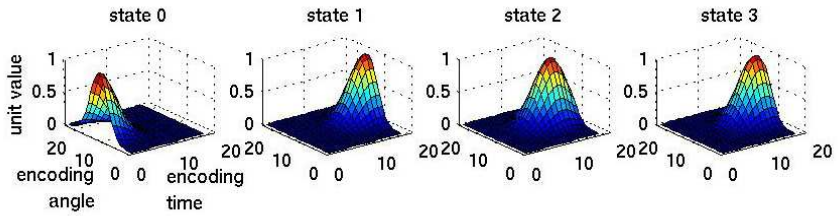

Fig. 4. State of a 4-layer extended Hopfield network while retrieving a sequence of four elements.

During recognition of a gesture, we proceed conversely by activating a subset of the first layers of the extended Hopfield network. Recognition of the class to which the gesture belongs is done by reactivating the output neurons y_c according

to:

$$y_c^{\mu+1} = y_c^{\mu} + \sum_j J_{i,c}^{\mu,a} \tag{4}$$

3 Results

We evaluated the performance of the network to classify and regenerate our set of 45 gestures (stylised drawings of the letters A to E). Further, in order to evaluate the network capacity against a large amount of noise, we generated a *synthetic dataset* of 250 gestures, by adding gaussian noise on one of the gestures belonging to the *real dataset*. Each dataset was divided equally into a *training set* and a *testing set*. Synthetic data were generated by displacing each key-point according to a gaussian distribution function centered on the original key-point and with a given standard deviation $\boldsymbol{\sigma}^D = (\sigma_t^D, \sigma_\theta^D)^T$, see Figure 5. For each value of σ^D, we generated 10 different gestures. We measured a standard deviation (noise) on the *real dataset* of $\boldsymbol{\sigma}^D = (0.88, 22.23)^T$.

The recognition performance of the model was contrasted to that of a Hidden Markov Model (HMM) trained and tested on the *real dataset*, see [7]. The 4 joint angle trajectories are represented by 4 left-right continuous HMMs. The values of each key-point $\{\theta_j^a, t_j^a\}$ is encoded in a bi-dimensional gaussian distribution of the observables. We enable transitions across at most two states at a time. We define a set of 4 HMMs for each class of gesture.

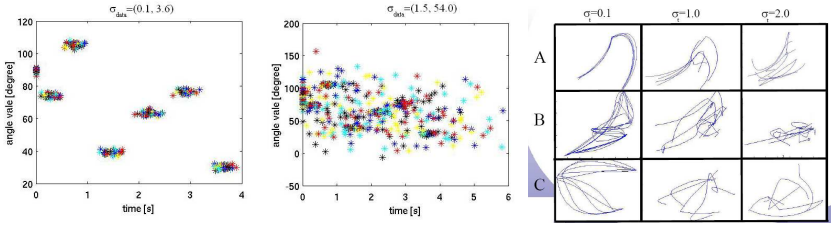

Fig. 5. Left: Sequence of key-points (t, θ) when the noise is generated with $\sigma_t^D = 0.1$ and $\sigma_\theta^D = 3.6$. Clusters do not overlap. Middle: key-points (t, θ) with $\sigma_t^D = 1.5$ and $\sigma_\theta^D = 54.0$. The overlap between clusters is large. (Right:) Distortion of the original gestures with a noise level of $\sigma_t^D = 0.1, 1.0$ and 2 respectively. Sole the gestures on the left are easily recognizable by the human eye.

Recognition Performance: Figure 6 shows the recognition rate of the ANN and HMM on the synthetic testing set as an effect of the temporal noise (average over 10 different gestures for each value of σ) with $\sigma_t^D = 3.6\sigma_D^\theta$. The recognition rate τ is given by *the proportion of correctly recognized patterns relative to the total number of patterns.*

We observe that both ANN and HMM recognize perfectly all gestures when the noise is inferior to ($\sigma_t^D \leq 0.25$). However, for high noise ($\sigma_t^D >= 1.0$),

the recognition rate of the ANN decreases importantly, while that of the HMM remains good even for gestures that the human eye might not even recognize (see Figure 5). This is due to the fact that the HMM could produce false recognition as no anti-model had been trained. Thus, the HMM would always categorize a gesture according to one of the 5 classes (even with very low confidence). In contrast, a fixed threshold on the global activity of the network forced the net to output a class only when the activity (i.e. confidence) was sufficiently high. Further work will compute the network's performance as an effect of varying this threshold.

Data Regeneration: Figure 6 shows 3 examples of regenerated gestures using the ANN model, superimposed to a set of 4 training gestures generated with a noise value $\boldsymbol{\sigma}^D = (0.1, 3.6)$. The network generates a generalised form of the gestures that encapsulate the major qualitative features (point of curvature) of the demonstrations.

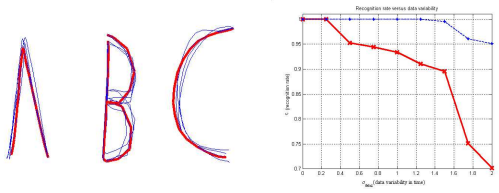

Fig. 6. (Left:) Gestures regenerated by the ANN (Bold line) against a set of 4 examples of demonstrated gestures (thin line) with a noise of $\boldsymbol{\sigma}^D = (0.1, 3.6)^T$. (Right:) Recognition rate of the ANN (Solid Line) and the HMM (dashed line) as an effect of the noise.

Acknowledgments This work was supported in part by the European Commission Division IST Future and Emerging Technologies, Integrated Project ROBOT-CUB and by the Swiss National Science Foundation, through grant 620-066127 of the SNF Professorships program.

References

1. Arbib, M., Billard, A., Iacoboni, M., Oztop, E.: Mirror neurons, imitation and (synthetic) brain imaging. In: Neural Networks. Volume 13 (8/9). (2000) 975–997
2. Billard, A.: Imitation. In: Handbook of Brain Theory and Neural Networks. Volume 2. MIT Press (2002) 566–569
3. Wang, D.: Temporal pattern processing. The Handbook of Brain Theory and Neural Network **2** (2003) 1163–1167
4. Miyoshi, S., Yanai, H., Okada, M.: Associative memory by recurrent neural networks with delay elements. Neural Networks **17** (2004) 55–63
5. Miyoshi, S., Nakayama, K.: A recurrent neural network with serial delay elements for memorizing limit cycles. In: Proc. of ICANN'95. (1995) 1955–1960

6. Mueller, K.R., Ibens, O.: Sequence storage of asymmetric hopfield networks with delay. In: ICANN'91. (1991) 163–168
7. Calinon, S., Billard, A.: Stochasitc gesture production and recognition model for a humanoid robot. Proceedings of 2004 IEEE/RSJ International Confernece on Intelligent Robots and Systems (2004)