Incremental On-line Clustering with a Topology-Learning Hierarchical ART Neural Network Using Hyperspherical Categories

Marko Tscherepanow

Applied Informatics, Bielefeld University
Universitätsstraße 25, 33615 Bielefeld, Germany
marko@techfak.uni-bielefeld.de

Abstract. Incremental on-line learning is an important branch of machine learning. One class of approaches particularly well-suited to such tasks are Adaptive Resonance Theory (ART) neural networks. This paper presents a novel ART network combining the ability of noise-insensitive, incremental clustering and topology learning at different levels of detail from TopoART with Euclidean similarity measures and hyperspherical categories from Hypersphere ART. As a result of the modified internal representations, several limitations of the original TopoART network are lifted. In particular, the presented network can process arbitrarily scaled values even if their range is not entirely known in advance.

Keywords: Incremental learning, On-line learning, Hierarchical clustering, TopoART, Adaptive Resonance Theory.

1 Introduction

In order to solve tasks involving incomplete knowledge or non-stationary data distributions, clustering methods capable of incremental on-line learning are necessary. Some examples for such tasks are the representation of visual concepts in robotic scenarios [1,2], dynamic topic mining [3], and protein localisation [4].

In comparison to traditional clustering approaches, such as the k-means algorithm [5], that require distinct training, validation, and test phases, incremental methods have to deal with additional problems:

1. How are new data incorporated into the model without impairing current knowledge? (*stability-plasticity dilemma* [6])
2. How is noise separated from relevant information?
3. How can data be correctly preprocessed, e.g., normalised to a given interval, if the input distribution is only partially known?

Adaptive Resonance Theory (ART) neural networks are particularly well-suited for incremental on-line learning, as they constitute a solution to the *stability-plasticity dilemma*. The first network of this family (ART1), which is limited to binary data, was published in 1987 [7]. Some well-known extensions,

Petra Perner (Ed.): Advances in Data Mining, Poster and Industry Proceedings of the 12th Industrial Conference on Data Mining (ICDM2012), pp. 22–34, ibai-publishing, Fockendorf (2012).

are Fuzzy ART [8], Gaussian ART [9], and Hypersphere ART [10]. ART neural networks incrementally learn a set of templates called categories. The properties of these categories may differ considerably between different networks. While the categories of Fuzzy ART have the shape of hyperrectangles, Hypersphere ART applies hyperspheres. Furthermore, the categories of Gaussian ART are Gaussians, which diminishes its sensitivity to noise in comparison to Fuzzy ART and Hypersphere ART, but impairs the stability of learnt representations.

From a representational point of view, Gaussian ART is strongly related to on-line Kernel Density Estimation (oKDE) [11]: oKDE incrementally estimates a Gaussian mixture model representing a given data distribution. Although, the focus of oKDE is on on-line adaptability, it achieves a certain degree of stability.

An alternative approach to on-line clustering is provided by topology-learning neural networks, such as Growing Neural Gas (GNG) [12]. Although the original GNG algorithm does not explicitly deal with the problems resulting from the *stability-plasticity dilemma*, there are several extensions tailored to incremental on-line learning, e.g., Incremental Growing Neural Gas (IGNG) [13] and the Self-Organising Incremental Neural Network (SOINN) [14]. But they do not reach the same degree of stability as ART networks. This is partially caused by the chosen type of internal representation: any adaptation of the neurons' weights, which correspond to prototype vectors in the input space, inevitably causes some loss of information. However, on the other hand, these neural networks are less prone to noise.

Recently, a topology-learning ART network called TopoART (see Section 2) [15,16] has been published as a combination of Fuzzy ART and SOINN. Since it is based on Fuzzy ART, TopoART shares some of its properties and limitations: hyperrectangular categories, adaptation mechanisms based on the city-block norm, and the mandatory normalisation of input to the interval $[0, 1]$. These properties may not be optimal as shown in [10,1]. In addition, the input normalisation can barely be performed if too less knowledge about the data is available. Therefore, this paper presents a novel TopoART network using hyperspherical categories and learning mechanisms adopted from Hypersphere ART (see Section 3). As a result, arbitrary real-valued input can be processed directly, the input domain does not need to be fixed, and the city-block norm is substituted by a Euclidean norm. Due to the shape of its categories, this network is called Hypersphere TopoART. It was evaluated using stationary and non-stationary synthetic data as well as real-world data (see Section 4).

2 TopoART

TopoART [15,16] is a modular neural network consisting of two major components clustering the input data at two different levels of detail (see Fig. 1). These components are referred to as TopoART a (TA a) and TopoART b (TA b). They have a similar structure that consists of two layers ($F1$ and $F2$). Input to both modules originates from the common input layer ($F0$). Furthermore, input to TA b is filtered by TA a.

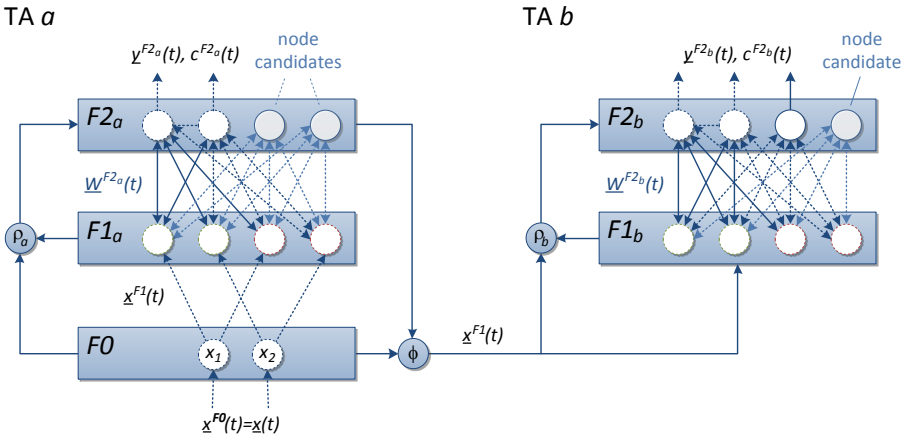

Fig. 1. Structure of a TopoART network for two-dimensional input vectors. TopoART comprises two modules called TA a and TA b that share a common input layer ($F0$).

For training, input vectors have to be provided at the $F0$ layer in discrete time steps t. If a new d-dimensional input vector

$$\underline{x}(t) = \big[x_1(t), \ldots, x_d(t)\big]^T \tag{1}$$

is presented, it is first complement-coded. Then, the encoded vector

$$\underline{x}^{F1}(t) = \big[x_1(t), \ldots, x_d(t), 1 - x_1(t), \ldots, 1 - x_d(t)\big]^T \tag{2}$$

is propagated to the $F1$ layer of TA a. Due to the complement-coding, each element $x_i(t)$ of the input vector $\underline{x}(t)$ must lie in the interval $[0, 1]$.

The complement-coded vector $\underline{x}^{F1}(t)$ is used to activate the nodes of the $F2$ layer. Here, each node j possesses a weight vector

$$\underline{w}_j^{F2}(t) = \big[w_{j,1}(t), \ldots, w_{j,d}(t), w_{j,d+1}(t), \ldots, w_{j,2d}(t)\big]^T. \tag{3}$$

$\underline{w}_i^{F2}(t)$ defines a hyperrectangular category: the elements from 1 to d specify the lower left corner and the elements from d+1 to $2d$ represent the complement of the upper right corner. The activation

$$z_j^{F2}(t) = \frac{\big\|\underline{x}^{F1}(t) \wedge \underline{w}_j^{F2}(t)\big\|_1}{\alpha + \big\|\underline{w}_j^{F2}(t)\big\|_1} \quad \text{with} \quad \alpha = 0.001, \tag{4}$$

which is also called *choice function*, measures the similarity of an input vector with the category of node j. Here, $\wedge$ denotes an element-wise minimum operation. The division by $\alpha + \big\|\underline{w}_j^{F2}(t)\big\|_1$ leads to a preference of small categories over large categories.

Besides the activation, a check (*match function*) is made to determine whether the corresponding category is able to grow and enclose the current input without exceeding a maximum category size

$$S^{\max} = d(1-\rho) \tag{5}$$

depending on the vigilance parameter ρ:

$$\frac{\left\|\underline{x}^{F1}(t) \wedge \underline{w}_j^{F2}(t)\right\|_1}{\left\|\underline{x}^{F1}(t)\right\|_1} \geq \rho. \tag{6}$$

Those nodes having the highest and the second highest activation while fulfilling (6) are referred to as the best-matching (bm) neuron and the second-best-matching (sbm) neuron, respectively. If adequate neurons bm and sbm have been be found, they are adapted:

$$\underline{w}_j^{F2}(t+1) = \beta_j\left(\underline{x}^{F1}(t) \wedge \underline{w}_j^{F2}(t)\right) + (1-\beta_j)\underline{w}_j^{F2}(t) \\ \text{with } j \in \{bm, sbm\} \text{ and } \beta_{bm} = 1. \tag{7}$$

This corresponds to an extension of the categories in the direction of the input vector $\underline{x}(t)$ (cf. [8]). The adapted category of bm even encloses $\underline{x}(t)$. The degree of the adaptation of the weights of sbm depends on the choice of its learning rate β_{sbm}. As the categories cannot shrink, the learning process is entirely stable.

In addition to the growth of the categories, bm and sbm are connected by an edge. Existing edges are not modified. Assuming that no node is available or (6) cannot be fulfilled for any existing node, a new neuron is incorporated. Its weights $\underline{w}_{new}^{F2}(t+1)$ are set to $\underline{x}^{F1}(t)$. This corresponds to a category enclosing only $\underline{x}(t)$.

In order to reduce its sensitivity to noise in comparison to Fuzzy ART, TopoART equips all $F2$ neurons with a counter n_j, counting the number of input samples they have learnt. Every τ learning cycles, all nodes with $n_j < \phi$ are removed.[1] Therefore, such neurons are called node candidates. Once n_j equals or surpasses ϕ, the corresponding neuron becomes a permanent node. While node candidates are subject to node removal, permanent nodes are guaranteed to be stable. Therefore, $\underline{x}^{F1}(t)$ is only propagated to TA b, if a permanent node fulfilling (6) was found in TA a.

The training process of TA b is identical to TA a, but the value of the vigilance parameter is increased:

$$\rho_b = \frac{1}{2}(\rho_a + 1). \tag{8}$$

This relation diminishes the maximum category size by 50%, which results in a refined clustering. In particular, links between nodes of TA a can be split in TA b. In this way, a hierarchical representation of the input data is computed.

If cluster labels are to be predicted for unknown input vectors, input is simultaneously propagated to the $F2$ layers of both modules without any filtering. Here, an alternative activation function is applied:

[1] Based on previous experiments, τ is always set to 200 in this paper.

$$z_j^{F2}(t) = 1 - \frac{\left\| \left(\underline{x}^{F1}(t) \wedge \underline{w}_j^{F2}(t) \right) - \underline{w}_j^{F2}(t) \right\|_1}{d} \tag{9}$$

In contrast to (4), (9) ensures that predictions are independent of the category size. This modification is important, as test samples often lie outside all categories. Using (4), smaller but more dissimilar categories would be preferred of larger but more similar categories.

The predictions are separately computed for each module and consist of an output vector $\underline{y}^{F2}(t)$ with

$$y_j^{F2}(t) = \begin{cases} 0 & \text{if } j \neq bm \\ 1 & \text{if } j = bm \end{cases} \tag{10}$$

as well as a vector $\underline{c}^{F2}(t)$ providing the cluster labels of all $F2$ nodes (cf. [15,16]). The labelling algorithm assigns unique integer labels to connected components of $F2$ nodes. For reasons of stability, node candidates are ignored during the computation of $\underline{y}^{F2}(t)$ and $\underline{c}^{F2}(t)$.

3 Hypersphere TopoART

The structure of Hypersphere TopoART (see Fig. 2) closely resembles the structure of TopoART (cf. Fig. 1). It consists of two modules called HTA a and HTA b sharing the input layer $F0$. Here, each input vector $\underline{x}(t)$ is extended by a single

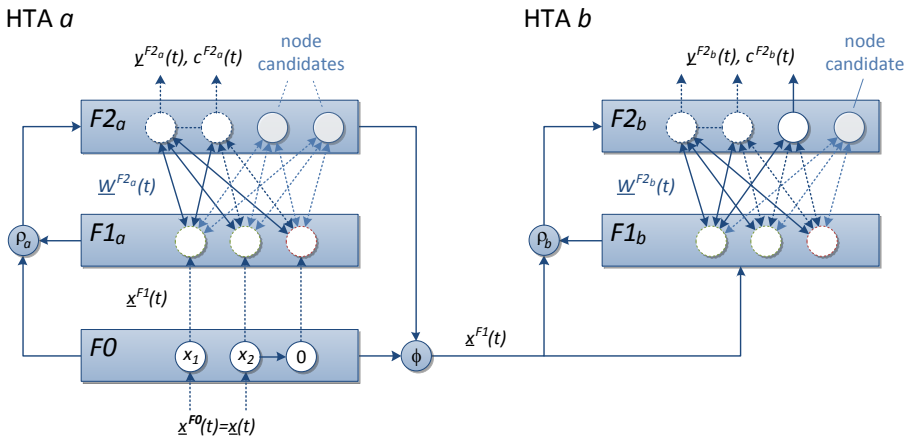

Fig. 2. Structure of a Hypersphere TopoART network for two-dimensional input vectors. Like TopoART, Hypersphere TopoART consists of two modules (HTA a and HTA b), has a three-layered structure, and the propagation of input to HTA b depends on the activation of HTA a.

element of value zero. The resulting vector

$$\underline{x}^{F1}(t) = \left[x_1(t), \ldots, x_d(t), 0\right]^T \tag{11}$$

is propagated to the respective $F1$ layer. This type of encoding has a similar effect to the complement-coding performed by TopoART: it corresponds to an initial category which encloses only the current input vector. This is reflected by the weights of the F2 nodes, which in contrast to TopoART, encode the mean $\underline{\mu}_j(t)$ and the radius $R_j(t)$ of a hyperspherical category:

$$\underline{w}_j^{F2}(t) = \begin{bmatrix} \underline{\mu}_j(t) \\ R_j(t) \end{bmatrix} = \left[\mu_{j,1}(t), \ldots, \mu_{j,d}(t), R_j(t)\right]^T. \tag{12}$$

This type of representation reduces the length of the weight vectors from $2d$ to $d{+}1$, which may be relevant for memory-intensive applications.

Due to the hyperspherical shape of the categories, the activation function of the $F2$ nodes (*choice function*) was adopted from Hypersphere ART [10]:

$$z_j^{F2}(t) = \frac{\bar{R} - \max\left(R_j, \left\|\underline{x}(t) - \underline{\mu}_j(t)\right\|_2\right)}{\bar{R} - R_j + \alpha} \quad \text{with} \quad \alpha = 0.001. \tag{13}$$

Similar to the denominator in the *choice function* of TopoART (4), the division by $\bar{R}{-}R_j{+}\alpha$ results in a preference for small categories. The radial extend $\bar{R}$ originates from Hypersphere ART. It denotes the maximum category radius if the respective vigilance parameter ρ equals zero. In order to be compatible with TopoART, $\bar{R}$ is chosen in such a way that in this case each category can span the entire input domain:

$$\bar{R} = \frac{1}{2}\sqrt{\sum_{i=1}^{d}\left(x_i^{\max} - x_i^{\min}\right)^2}. \tag{14}$$

Here, $x_i^{\max}$ and $x_i^{\min}$ denote the expected maximum value and the expected minimum value of input along dimension i, respectively. Since these values are not used for an input normalisation as required by complement-coding, rough estimates are sufficient and even modifications during learning are possible.

Similar to TopoART, a check is made, whether a category is allowed to grow or not. The *match function*

$$1 - \frac{\max\left(R_j, \left\|\underline{x}(t) - \underline{\mu}_j(t)\right\|_2\right)}{\bar{R}} \geq \rho \tag{15}$$

that tests whether the resulting category would exceed the maximum category size

$$S^{\max} = \bar{R}(1 - \rho) \tag{16}$$

is applied in the same way as with TopoART. However, as $S^{\max}$ does not directly depend on the dimension of the input space d (cf. (5)), the flexibility of the network is increased.

The weights of the best-matching neuron bm and the second-best-matching neuron sbm are adapted as follows:

$$\underline{\mu}_j(t+1) = \underline{\mu}_j(t) + \frac{\beta_j}{2}\left(1 - \frac{\min\left(R_j(t), \left\|\underline{d}_j(t)\right\|_2\right)}{\left\|\underline{d}_j(t)\right\|_2}\right)\underline{d}_j(t) \quad \text{and} \tag{17}$$

$$R_j(t+1) = R_j(t) + \frac{\beta_i}{2}\Big(\max\left(R_j(t), \left\|\underline{d}_j(t)\right\|_2\right) - R_j(t)\Big) \tag{18}$$

$$\text{with } \underline{d}_j(t) = \underline{x}(t) - \underline{\mu}_j(t),\ j \in \{bm, sbm\}, \text{ and } \beta_{bm} = 1. \tag{19}$$

These equations guarantee that all samples enclosed by a category will still be enclosed after adaptation [10]. Hence, Hypersphere TopoART learns in a stable way like TopoART.

The activation function used for prediction had to be adapted to the new type of internal representation as well:

$$z_j^{F2}(t) = 1 - \frac{\max\left(\left\|\underline{x}(t) - \underline{\mu}_j(t)\right\|_2 - R_j, 0\right)}{2\bar{R}}. \tag{20}$$

Instead of the city-block distance, (20) measures the Euclidean distance between a category and an input vector $\underline{x}(t)$. Like (9), it is independent from the category size.

The further functioning of Hypersphere TopoART is identical to TopoART. As a consequence, Hypersphere TopoART adopts its beneficial properties (e.g., the insensitivity to noise and the ability to learn arbitrarily shaped clusters) and shares its input/output behaviour.

4 Results

For the evaluation two datasets that had been used for the analysis of TopoART (TA) [16] were applied. First, a synthetic two-dimensional dataset was used. It consists of five components (A–E), each of which encompasses 18,000 samples. In addition, it contains 10% of uniformly distributed random noise. All samples had been mixed randomly to create a stationary input distribution. In order to illustrate the benefits of Hypersphere TopoART, this dataset was scaled to the interval $[-5, 5]$ (see Fig. 3a). However, the data had to be rescaled to the interval $[0, 1]$ for Fuzzy ART and TopoART so as to allow for complement-coding.

In contrast to Fuzzy ART (see Fig. 3b) and Hypersphere ART (see Fig. 3c), which are very sensitive to noise, TopoART[2] (see Fig. 3f) and Hypersphere TopoART (see Fig. 3h) were able to find the five components of the input distribution, although the corresponding parameters[3] were set to equal values. oKDE (see Fig. 3d) demonstrated a significantly higher tolerance to noise than Fuzzy

[2] LibTopoART (version 0.35), available at `www.LibTopoART.eu`

[3] ρ_a=ρ=0.92, β_{bm}=β=γ=1, and $\bar{R}$ was set according to (14).

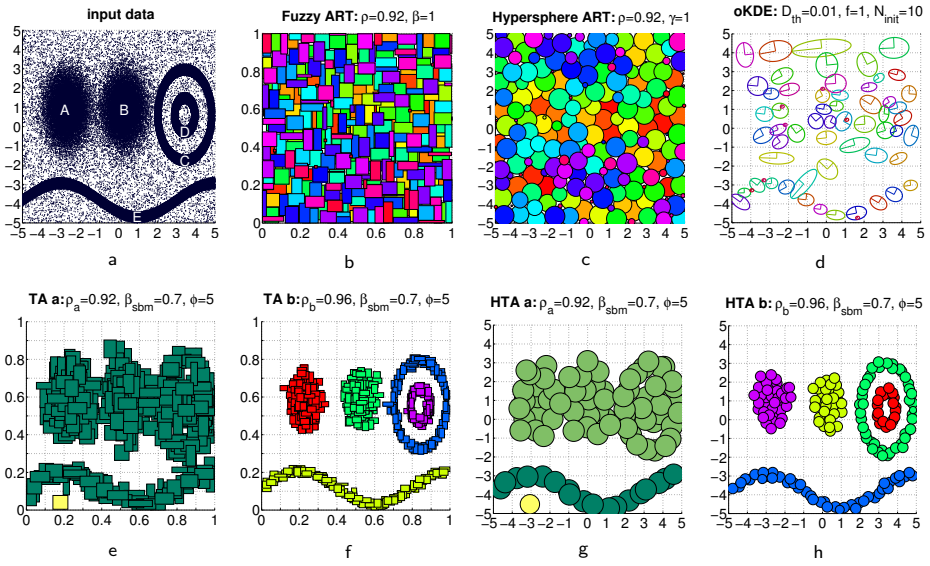

Fig. 3. Two-dimensional data distribution and clustering results. Different clusters are painted with different colours. The Gaussians determined by oKDE are draw as ellipses marking the standard deviations.

ART and Hypersphere ART but does not reflect the topology of the data. Both TopoART and Hypersphere TopoART refined the representation from module a (see Figs. 3e and 3g) to module b (see Figs. 3f and 3h). In addition, the single noise category learnt by module a disappeared in the clustering of module b.

During incremental learning, the data distribution may change over time. Therefore, an incremental on-line clusterer like Hypersphere TopoART has to cope with non-stationary data. In order to analyse this capability, the samples from the previous experiment were reordered and presented in four subsequent phases (see Fig. 4). In addition to the distribution itself, the input domain was modified. It encompasses all samples of the current phase and the regions known from previous training phases. Hence, the considered region of the input space is growing so as to simulate a gradual extension of knowledge. Thus, the parameter $\bar{R}$ was individually set for each phase according to (14). Furthermore, new noise samples fitting into the considered regions of the input space had to be determined: while the percentage of noise was adopted from the first experiment, the modification of the input domain results in a higher density of noise samples during the first three training phases, as the considered region of the input space is diminished there.

Hypersphere TopoART was able to correctly learn the underlying components (cf. Fig. 3a) despite the non-stationary nature of the input distribution and the variable range of the input vectors (see fourth column of Fig. 4). The resulting clustering is qualitatively equal to the one obtained from the stationary

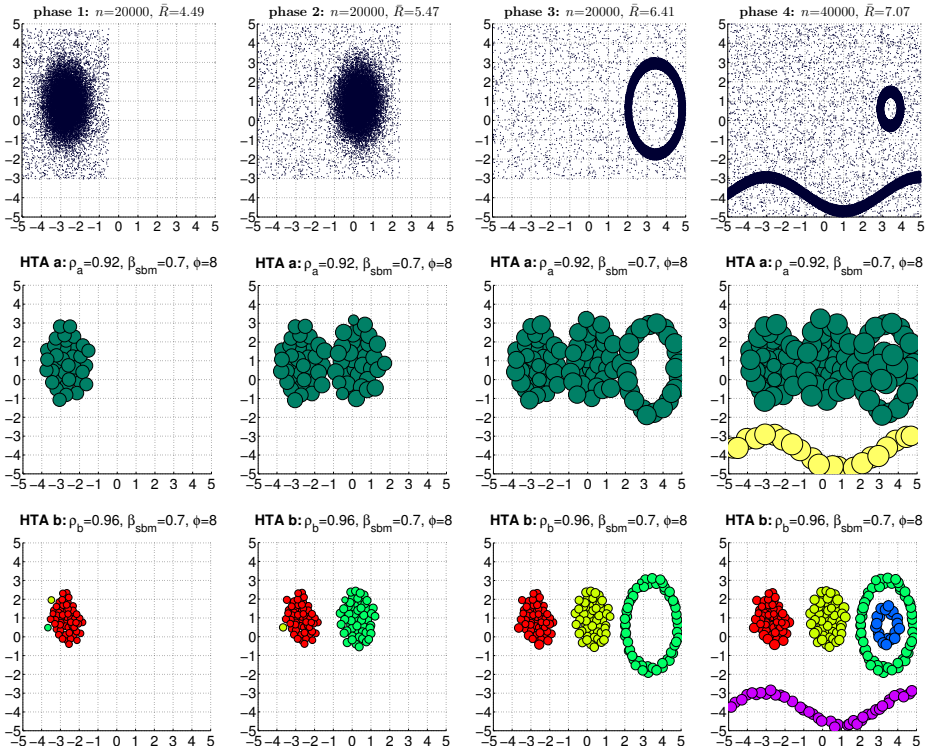

Fig. 4. Clustering results for non-stationary data from a growing input domain. The training of a Hypersphere TopoART network was performed in four subsequent phases with different numbers of training samples n. The respective input data are shown in the top row. The considered regions of the input space are indicated by the distribution of the noise samples. As the input domain is growing, the parameter $\bar{R}$ was individually set for each phase. The clustering results are shown below the respective training data. Here, different clusters are painted with different colours.

distribution (cf. Figs. 3g and 3h). This is remarkable as just one parameter had to be altered in comparison to the first experiment: ϕ was increased so as to account for the higher density of noise samples during the early training phases. The ability to adapt to changes of the input domain is a major improvement in comparison to TopoART, which is limited by complement-coding.

In addition to the synthetic data, the clustering capabilities of Hypersphere TopoART were analysed using a real-world dataset derived from facial images of 32 people (12 female, 20 male) showing 28 predefined facial expressions under two lighting conditions (see Fig. 5).

This dataset was originally compiled for training the user-interface robot iCat to imitate human facial expressions [17]. As during the recording of the data a few facial expressions were skipped by some persons, the total number of available images amounts to 1783. The main advantage of this dataset is the availability

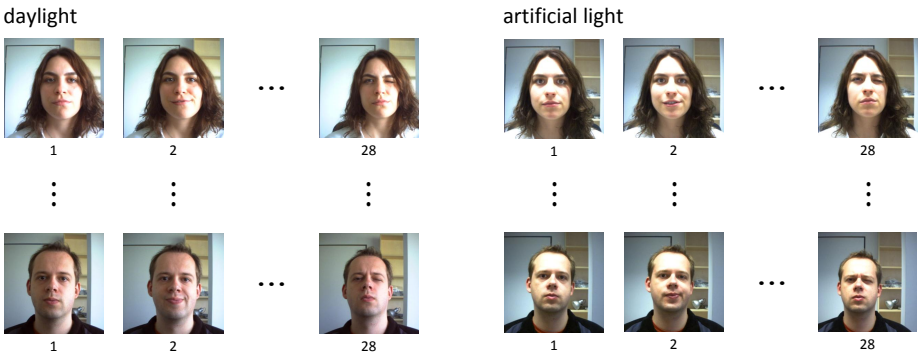

Fig. 5. The iCat dataset. This dataset consists of 1783 images taken by the user-interface robot iCat. They show 32 persons performing 28 different facial expressions under two lighting conditions.

of several partitionings according to different criteria that were collected when the data were recorded. These criteria are the gender, the usage of glasses, the facial expressions, the lighting conditions, and the persons themselves. Due to the different characteristics of the underlying partitionings, this dataset is an excellent basis for comparing different clustering approaches.

In order to reduce the dimensionality of the input space, the images were converted to grayscale. Then, those image regions containing the face were cut out and scaled to a size of 64×64 pixels. Finally, the resulting images were subjected to principal component analysis keeping 90% of the total variance, which had proven advantageous for the direct imitation of human facial expressions (cf. [17]). Due to this extensive dimensionality reduction, the training samples comprise only 45 features.

Using the iCat dataset, we compared those methods that performed well on the synthetic data; i.e., TopoART and Hypersphere TopoART. During the evaluation, two standard measures, namely the Rand index R and the Jaccard coefficient J [18], were used. These similarity measures provide values between 0 and 1, with higher values indicating a higher degree of similarity.

In order to find appropriate values for the relevant parameters, a grid search was performed: β_{sbm}, ϕ, and ρ_a were iterated in their respective intervals in order to maximise R and J for the different partitionings. The lower number of training samples in comparison to the previous experiment was compensated for by presenting the complete dataset 25 times. The best results for both neural networks are given in Table 1.

Table 1 shows that the type of internal representation has an impact on the clustering results. While TopoART performed best for the partitionings according to the gender and the usage of glasses, Hypersphere TopoART achieved the highest similarity with the partitionings according to the lighting conditions and the persons. Hence, both neural networks can be considered as complementary

Table 1. Clustering results for the iCat dataset. The bold numbers indicate the best result for each partitioning. If R or J did not differ between the modules and networks for a certain partitioning, no results were highlighted.

partitioning	**TopoART**		**Hypersphere TopoART**	
	R	***J***	***R***	***J***
	TA a/TA b	TA a/TA b	HTA a/HTA b	HTA a/HTA b
gender	0.583/**0.587**	0.532/0.532	0.532/0.532	0.532/0.532
usage of glasses	0.843/**0.853**	0.838/**0.850**	0.833/0.833	0.833/0.833
facial expressions	0.965/0.965	0.035/0.035	0.965/0.965	0.035/0.035
lighting conditions	0.939/0.939	0.885/0.885	**1.000**/0.986	**1.000**/0.972
persons	0.976/0.976	0.225/0.249	**0.981**/**0.981**	**0.410**/0.401

to each other, since they perform differently for different problems similar to the kernel functions of support vector machines [19].

In addition to the analysis of the general clustering properties, the capability to compute a hierarchical clustering reflecting different partitionings was examined. The partitionings according to the lighting conditions and the persons were chosen as examples, since these partitionings had been used for the analysis of TopoART before [15,16]. Here, the parameters β_{sbm}, ϕ, and ρ_a were iterated in their respective intervals in order to maximise the Jaccard coefficient for the partitioning according to the different people. The results depending on the most sensitive parameter ρ_a are shown in Fig. 6.

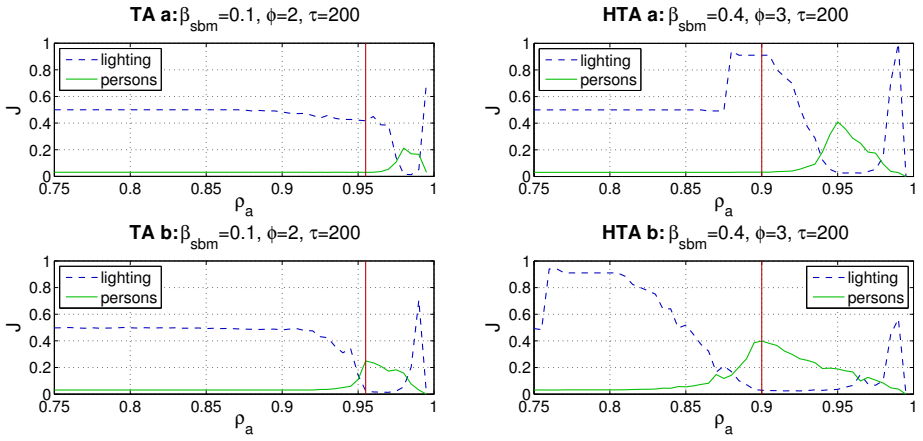

Fig. 6. Results for the hierarchical clustering task. The vertical red line marks exemplary values of ρ_a for which module a is sensitive to the lighting conditions and module b is sensitive to the different people.

For both partitionings, Hypersphere TopoART achieved higher values of the Jaccard coefficient than TopoART. Furthermore, it inherits the property of TopoART to separately represent both partitionings: while HTA a is more sensitive to the lighting conditions, HTA b better reflects the partitioning according to the person. Thus, if the vigilance parameter ρ_a is chosen appropriately, two different clusterings can be learnt simultaneously.

5 Conclusion

In this paper, a novel neural network based on the TopoART architecture was presented. It adopts the basic properties of TopoART, in particular, stable online clustering and topology learning at two levels of detail. But in contrast, it allows for the direct learning of input without rescaling and can adapt to moderate changes of the input domain. In addition, it constitutes an alternative for problems that do not fit the city-block norm or the hyperrectangular categories used by TopoART. In the future, Hypersphere TopoART could be applied in a similar way to TopoART in order to construct further networks, e.g., an associative memory equivalent to TopoART-AM [16] or a regression method like TopoART-R [20].

Acknowledgements. This work was partially funded by the German Research Foundation (DFG), Excellence Cluster 277 "Cognitive Interaction Technology".

References

1. Kammer, M., Tscherepanow, M., Schack, T., Nagai, Y.: A perceptual memory system for affordance learning in humanoid robots. In: Proceedings of the International Conference on Artificial Neural Networks (ICANN). LNCS, vol. 6792, pp. 349–356. Springer (2011)
2. Wyatt, J.L., Aydemir, A., Brenner, M., Hanheide, M., Hawes, N., Jensfelt, P., Kristan, M., Kruijff, G.J.M., Lison, P., Pronobis, A., Sjöö, K., Vrečko, A., Zender, H., Zillich, M., Skočaj, D.: Self-understanding and self-extension: A systems and representational approach. IEEE Transactions on Autonomous Mental Development **2**(4), 282–303 (2010)
3. Lamirel, J.C., Safi, G., Priyankar, N., Cuxac, P.: Mining research topics evolving over time using a diachronic multi-source approach. In: International Conference on Data Mining Workshops (ICDMW). pp. 17–24. IEEE (2010)
4. Tscherepanow, M., Jensen, N., Kummert, F.: An incremental approach to automated protein localisation. BMC Bioinformatics **9**(445) (2008)
5. MacQueen, J.: Some methods for classification and analysis of multivariate observations. In: Proceedings of the Berkeley Symposium on Mathematical Statistics and Probability. vol. 1, pp. 281–297 (1967)
6. Grossberg, S.: Competitive learning: From interactive activation to adaptive resonance. Cognitive Science **11**, 23–63 (1987)
7. Carpenter, G.A., Grossberg, S.: A massively parallel architecture for a self-organizing neural pattern recognition machine. Computer Vision, Graphics, and Image Processing **37**(1), 54–115 (1987)

8. Carpenter, G.A., Grossberg, S., Rosen, D.B.: Fuzzy ART: Fast stable learning and categorization of analog patterns by an adaptive resonance system. Neural Networks **4**, 759–771 (1991)
9. Williamson, J.R.: Gaussian ARTMAP: a neural network for fast incremental learning of noisy multidimensional maps. Neural Networks **9**(5), 881–897 (1996)
10. Anagnostopoulos, G.C., Georgiopoulos, M.: Hypersphere ART and ARTMAP for unsupervised and supervised incremental learning. In: Proceedings of the International Joint Conference on Neural Networks (IJCNN). vol. 6, pp. 59–64 (2000)
11. Kristan, M., Leonardis, A., Skočaj, D.: Multivariate online kernel density estimation with Gaussian kernels. Pattern Recognition **44**(10–11), 2630–2642 (2011)
12. Fritzke, B.: A growing neural gas network learns topologies. In: Neural Information Processing Systems (NIPS). pp. 625–632 (1994)
13. Prudent, Y., Ennaji, A.: An incremental growing neural gas learns topologies. In: Proceedings of the International Joint Conference on Neural Networks (IJCNN). vol. 2., pp. 1211–1216. IEEE (2005)
14. Furao, S., Hasegawa, O.: An incremental network for on-line unsupervised classification and topology learning. Neural Networks **19**, 90–106 (2006)
15. Tscherepanow, M.: TopoART: A topology learning hierarchical ART network. In: Proceedings of the International Conference on Artificial Neural Networks (ICANN). LNCS, vol. 6354, pp. 157–167. Springer (2010)
16. Tscherepanow, M., Kortkamp, M., Kammer, M.: A hierarchical ART network for the stable incremental learning of topological structures and associations from noisy data. Neural Networks **24**(8), 906–916 (2011)
17. Tscherepanow, M., Hillebrand, M., Hegel, F., Wrede, B., Kummert, F.: Direct imitation of human facial expressions by a user-interface robot. In: Proceedings of the IEEE-RAS International Conference on Humanoid Robots (Humanoids). pp. 154–160 (2009)
18. Xu, R., Wunsch II, D.C.: Clustering. Wiley–IEEE Press (2009)
19. Schölkopf, B., Smola, A.J.: Learning with Kernels: Support Vector Machines, Regularization, Optimisation, and Beyond. Adaptive Computation and Machine Learning. MIT Press (2002)
20. Tscherepanow, M.: An extended TopoART network for the stable on-line learning of regression functions. In: Proceedings of the International Conference on Neural Information Processing (ICONIP). LNCS, vol. 7063, pp. 562–571. Springer (2011)