Proc. German Pattern Recognition Conference DAGM 2005, Vienna., pp. 301-308

Rapid Online Learning of Objects in a Biologically Motivated Recognition Architecture

Stephan Kirstein, Heiko Wersing, and Edgar Körner

Honda Research Institute Europe GmbH
Carl Legien Str. 30, 63073 Offenbach am Main, Germany
{stephan.kirstein,heiko.wersing,edgar.koerner}@honda-ri.de

Abstract. We present an approach for the supervised online learning of object representations based on a biologically motivated architecture of visual processing. We use the output of a recently developed topographical feature hierarchy to provide a view-based representation of three-dimensional objects using a dynamical vector quantization approach. For a simple short-term object memory model we demonstrate real-time online learning of 50 complex-shaped objects within three hours. Additionally we propose some modifications of learning vector quantization algorithms that are especially adapted to the task of online learning and capable of effectively reducing the representational effort in a transfer from short-term to long-term memory.

1 Introduction

Most research on trainable object recognition algorithms has so far focused on learning based on collecting large data sets and then performing offline training of the corresponding classifiers. Since in these approaches learning speed is not a primary optimization goal, typical offline training times last many hours. Another problem is that most classifier architectures like e.g. multi layer perceptrons or support vector machines do not allow online training with the same performance as for offline batch training. Due to these drawbacks, research in man-machine interaction for robotics dealing with online learning of objects has used histogram-based feature representations [8] or hashing techniques [1] that offer fast processing, but only limited representational and discriminatory capacity. An interesting approach to supervised online learning for object recognition was proposed by Bekel et al. [2]. Their VPL classifier consists of feature extraction based on vector quantization and PCA and supervised classification using a local linear map architecture.

We suggest to use a biologically motivated strategy similar to the hierarchical processing in the ventral pathway of the human visual system to speed up object learning considerably. The main idea is to use a sufficiently general feature representation that remains unchanged, while object-specific learning is accomplished only in the highest levels of the hierarchy. We perform supervised online learning of objects using a short-term memory with a similarity-based adaptive collection of view templates using the intermediate level feature representation of

the proposed visual hierarchy from [9]. Additionally we propose an incremental learning vector quantization model to achieve a reduction of the representational effort that is related to the transfer from short-term to long-term memory.

After a short introduction to the hierarchical feature processing model we introduce our short-term and refined long-term memory model, based on an incremental learning vector quantization approach in Sect. 2. We demonstrate its effectiveness for an implementation of real-time online object learning of 50 objects in Sect. 3, and give our conclusions in Sect.4.

2 Hierarchical Online Learning Model

Our online learning model consists of three major processing stages: First the input image is processed using a topographically organized feature hierarchy. Object views are then stored using the feature map representation in a template-based short-term memory, that allows immediate online learning and recognition. Finally the short-term memory representatives are accumulated into a condensed long-term memory. We now describe these three stages in more detail:

Initial Processing Architecture. Our hierarchy is based on a feed-forward architecture with weight-sharing [4] and a succession of feature-sensitive and pooling stages (see Fig.1 and [9] for details). The output of the feature representation of the complex feature layer (C2) can be used for robust object recognition that is competitive with other state-of-the-art models [9]. We augment the shape representation from [9] with downsampled color maps in the three RGB channels of the input image with the same resolution as the C2 shape features. We denote the output of the hierarchy for a given input image $\mathbf{I}_i$ as $\mathbf{x}^i(\mathbf{I}_i)$.

Online Vector Quantization as Short-Term Memory. Object views are stored in a set of M representatives $\mathbf{r}^l$, $l = 1, \ldots, M$, that are incrementally collected, and labelled with class Q^l. We define R_q as the set of representatives $\mathbf{r}^l$ that belong to object q. The acquisition of templates is based on a similarity threshold S_T. New object views are only collected into the short-term memory (STM) if their similarity to the previously stored views is less than S_T. The parameter S_T is critical, characterizing the compromise between representation resolution and computation time. We denote the similarity of view $\mathbf{x}^i$ and representative $\mathbf{r}^l$ by A_{il} and compute it based on C2 feature space distance by $A_{il} = \exp(-||\mathbf{x}^i - \mathbf{r}^l||^2/\sigma)$. Here, σ is chosen for convenience such that the average similarity in a generic recognition setup is approximately equal to 0.5.

For one learning step the similarity A_{il} between the current training vector $\mathbf{x}^i$, labelled as object q and all representatives $\mathbf{r}^l \in R_q$ of the same object q is calculated and the maximum value is computed as $A_i^{\max} = \max_{l \in R_q} A_{il}$. The training vector $\mathbf{x}^i$ with its class label is added to the object representation, if $A_i^{max} < S_T$. If M representatives were present before, then choose $\mathbf{r}^{M+1} = \mathbf{x}^i$ and $Q^{M+1} = q$. Otherwise we assume that the vector $\mathbf{x}^i$ is already sufficiently well represented by one $\mathbf{r}^l$, and do not add it to the representation. We call this template-based representation online vector quantization (oVQ). The non-destructive incremental learning process allows online learning and recognition

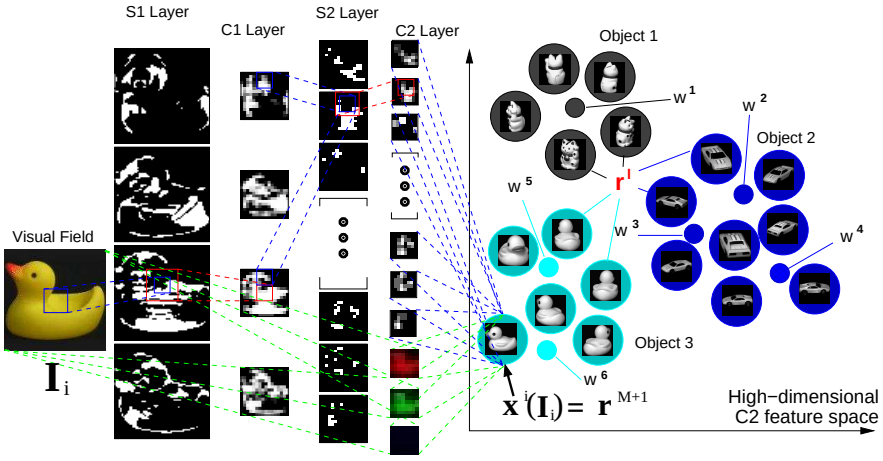

Fig. 1. The visual hierarchical network structure. Based on a color image input $\mathbf{I}_i$ (64×64), shape and color processing is separated in the feature hierarchy and fused in the view-based object representation. In the shape pathway the S1 feature-matching layer computes an initial linear sign-insensitive receptive field summation, a Winner-Take-Most mechanism between features at the same position and a final threshold function. We use Gabor filter receptive fields, to perform a local orientation estimation in this layer. The C1 layer subsamples the S1 features by pooling down to 16×16 resolution using a Gaussian receptive field and a sigmoidal nonlinearity. The 50 features in the intermediate layer S2 are trained by sparse coding and are sensitive to local combinations of the features in the planes of the C1 layer. The layer C2 again performs spatial integration and reduces the resolution to 8×8. When the color pathway is used, three downsampled 8×8 maps of the individual RGB channels are added to the C2 feature maps. The short-term memory consists of template vectors $\mathbf{r}^l$ that are computed as the output $\mathbf{x}^i(\mathbf{I}_i)$ of the hierarchy and added based on sufficient Euclidean distance in the C2 feature space to previously stored representatives of the same object. The refined long-term memory representatives $\mathbf{w}^k$ are learned from the labelled short-term memory nodes $\mathbf{r}^l$ using an incremental vector quantization approach.

at the same time, without a separation into training and testing phases. To model a limited STM capacity we set in some simulations an upper limit of 10 objects that can be represented and if the 11th object is presented, representatives of the oldest learned object are removed from the STM.

Recognition of a unclassified test view $\mathbf{I}_j$ can be done with a nearest neighbour search of the hierarchy output $\mathbf{x}^j(\mathbf{I}_j)$ to the set of STM representatives. The winning node $l_{\max}$ satisfies $l_{\max} = \arg\max_l(A_{jl})$ and then the class label $Q^{l_{\max}}$ of the winning representative $\mathbf{r}^{l_{\max}}$ is assigned to the current test view $\mathbf{x}^j$.

Incremental LVQ as Long-Term Memory. The labelled STM representatives $\mathbf{r}^l$ in the C2 feature space provide the input ensemble for our proposed long-term memory (LTM) representation, which is optimized and built up incrementally based on the set of STM nodes $\mathbf{r}^l$, where we assume a limited STM capacity with only the most recently shown objects being represented. In con-

trast to the typical usage of learning vector quantization networks [7], where every class is trained with a fixed number of LVQ nodes, we use an incremental approach, related to other models like e.g. the growing neural gas [3]. Another related work not dealing with incremental learning, but with clustering of non-stationary or changing datasets was proposed by [5].

For training our incremental LVQ (iLVQ) model, a stream of randomly selected input STM training vectors $\mathbf{r}^l$ is presented, and classified using labelled iLVQ representatives in a Euclidean metrics. The training classification errors are collected, and each time a given sufficient number of classification errors has occurred a set of new iLVQ nodes is inserted. The addition rule is designed to promote insertion of nodes at the class boundaries. During training, iLVQ nodes are adapted with standard LVQ weight learning that moves nodes into the direction of the correct class and away from wrong classes. An important change to the standard LVQ is an adaptive modification of the individual node learning rates to deal with the stability-plasticity dilemma of online learning. The learning rate of winning nodes is more and more reduced to avoid too strong interference of newly learned representatives with older parts of the object LTM.

We denote the set of iLVQ representative vectors at time step t by $\mathbf{w}^k(t)$, $k = 1, \ldots, K$, where K is the current number of nodes. C^k denotes the corresponding class label of the iLVQ center $\mathbf{w}^k$. The training of the iLVQ nodes is based on the current set of STM nodes $\mathbf{r}^l$ with class Q^l that serve as input vectors for the LTM. Each iLVQ node $\mathbf{w}^k$ obtains an individual learning rate $\Theta_k(t) = \Theta(0)\exp(-a_k(t)/d)$ at step t, where $\Theta(0)$ is an initial value, d is a scaling factor, and $a_k(t)$ is an iteration-dependent age factor. The age factor a_k is incremented when the corresponding $\mathbf{w}_k$ becomes the winning node.

New iLVQ nodes are always inserted, if a given number $G_{\max}$ of training vectors was misclassified during iterative presentation of the $\mathbf{r}^l$. The value of $G_{\max} = 30$ is a compromise between convergence speed and representation resolution. Within this error history, misclassifications are memorized with corresponding input $\mathbf{r}^l$ and winning iLVQ node $\mathbf{w}^{k_{\max}}(\mathbf{r}^l)$. We denote S_p as the set of previously misclassified $\mathbf{r}^l$ within this error history that were of original class $p = Q^l$. For each nonempty S_p a new node $\mathbf{w}^m$ is added to the representation. It is initialized to the element of $\mathbf{r}^l \in S_p$ that has the minimal distance to its corresponding winning iLVQ node $\mathbf{w}^{k_{\max}}(\mathbf{r}^l)$ and the class of the iLVQ node is given as $C^m = Q^l$. This insertion rule adds new nodes primarily near to class borders. The formal definition of the iLVQ learning algorithm is then:

1. Choose randomly $\mathbf{r}^l$ from the set of STM nodes. Find winning iLVQ node $k_{\max} = \arg\max_k(-||\mathbf{r}^l - \mathbf{w}^k||)$ and update $\mathbf{w}^{k_{\max}}(t+1) = \mathbf{w}^{k_{\max}}(t) + \kappa\Theta_{k_{\max}}(t)(\mathbf{r}^l - \mathbf{w}^{k_{\max}}(t))$, where $\kappa = 1$ if $C^{k_{\max}} = Q^l$ and $\kappa = -1$ otherwise. The learning rate is given as $\Theta_{k_{\max}}(t) = \Theta(0)\exp(-a_{k_{\max}}(t)/d)$.
2. Increment $a_{k_{\max}}(t+1) = a_{k_{\max}}(t) + 1$.
3. If $C^{k_{\max}} \neq Q^l$ increase $G(t+1) = G(t) + 1$. Add $\mathbf{r}^l$ to the current set of misclassified views S_{Q^l} of object Q^l.
4. If $G = G_{\max}$, then do for each $S_p \neq \emptyset$: Find the object index C^m of the iLVQ representative $\mathbf{w}^m$ with minimal distance to the wrongly classified elements

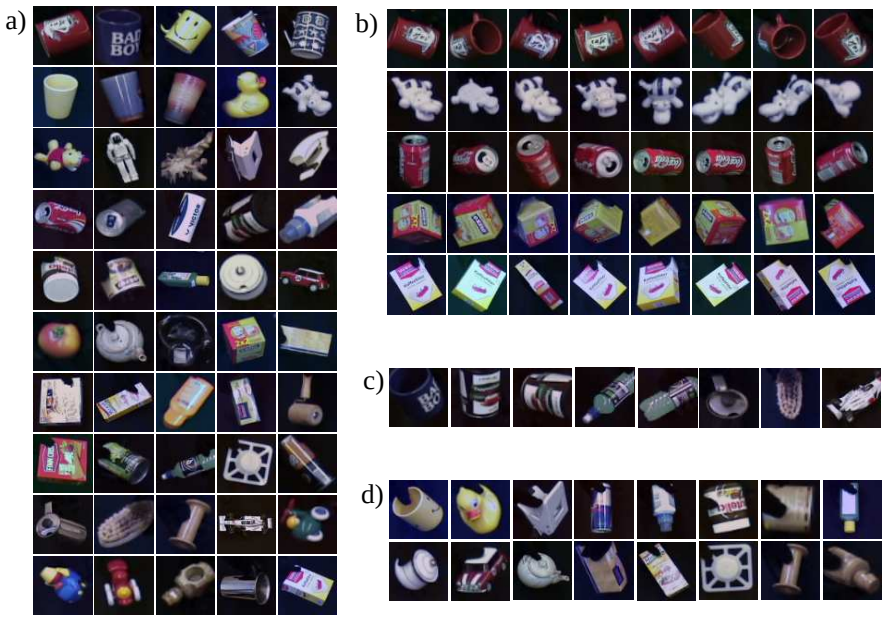

Fig. 2. Test object images. (a) 50 freely rotated objects, taken in front of a dark background and using a black glove for holding. (b) Some rotation examples. (c) A few examples for incomplete segmentation. (d) Examples for minor occlusion effects. The main difficulties of this training ensemble are the high appearance variation of objects during rotation around three axes, and shape similarity among cans, cups and boxes, combined with segmentation errors (c), and slight occlusions (d).

in S_p according to $||\mathbf{r}^l - \mathbf{w}^m|| = \min_{l|\mathbf{r}^l \in S_p} ||\mathbf{r}^l - \mathbf{w}(\mathbf{r}^l)||$, where $\mathbf{w}(\mathbf{r}^l)$ is the winning iLVQ node for view $\mathbf{r}^l$. Insert a new iLVQ node with $\mathbf{w} = \mathbf{r}^l$. Reset $G = 0$ and $S_p = \emptyset$ for all p. Goto step 1 until sufficient convergence.

Classification of a test view $\mathbf{x}^j(\mathbf{I}_j)$ is done by determining the winning iLVQ node $\mathbf{w}^{k_{\max}}$ with smallest distance to $\mathbf{x}^j$ and assigning the class label $C^{k_{\max}}$.

The C2 feature vectors are sparsely activated with only about one third of nonzero entries. During our investigation of the iLVQ approach we noted that convergence can be improved by applying the weight update of $\mathbf{w}(\mathbf{r}^l)$ nodes only on the nonzero entries of the input vectors $\mathbf{r}^l$. The weight update is then defined componentwise as $w_i^{k_{\max}}(t+1) = w_i^{k_{\max}}(t) + H(r_i^l)\kappa\Theta_{k_{\max}}(t)(r_i^l - w_i^{k_{\max}}(t))$, where H is the Heaviside function. We call this modification *sparse* iLVQ.

3 Experimental Results

Setup. For our experiments we use a setup, where we show objects, held in hand with a black glove before a black background. Color images are taken with a camera, segmented using local entropy-thresholding [6], and normalized

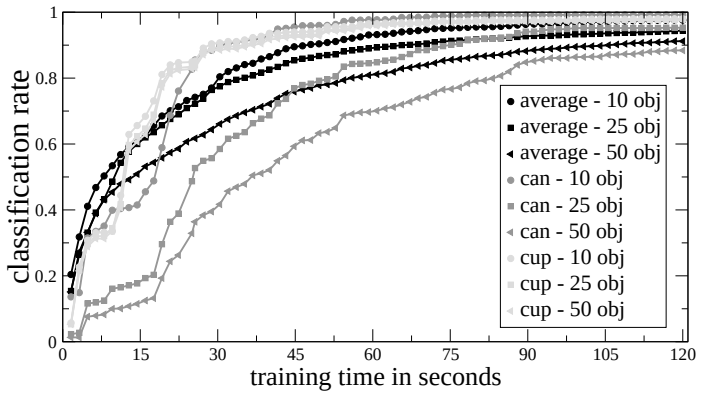

Fig. 3. Classification rate of two selected objects dependent on the training time for learning the 10th, 25th, and 50th object, and same learning curves averaged over 20 object selections. While training proceeds, at each point classification rate is measured on all 750 available test views of the current object. Good recognition performance can be achieved within two minutes, also for the 50th object.

in size (64x64 pixels). We show each object by rotating it freely by hand for some minutes, such that 750 input images $\mathbf{I}_i$ for each object are collected (see Fig.2). Another set of 750 images for each object is recorded for testing.

Online Learning using Short-Term Memory. In the first experiment we investigate the time necessary for training the template-based oVQ short-term memory with up to 50 objects. The training speed is limited by the frame rate of the used camera (12.5 Hz), the computation time needed for the entropy segmentation, the extraction of the corresponding sparse C2 feature vector $\mathbf{x}^i$ with 3200 shape dimensions and 192 color dimensions and the calculation of similarities A_{il} (see Sect.2). The similarity threshold was set to $S_T = 0.85$ for this experiment, and there was no limit imposed on the number of STM representatives. Altogether we achieve an average frame rate of 7 Hz on a 3GHz Xeon processor. For the shown curves of a cup and a can from our database we trained 9, 24 or 49 objects and incrementally added the cup or can as an additional object. Figure 3 shows how long it takes until the newly added object can be robustly separated from all other objects. At the given points the correct classification rate of the current object is computed using the 750 views from the disjoint test ensemble. Additionally we show the learning curves, averaged over 20 randomly chosen object selections. On average, training of one object can be done in less than 2 minutes, with rapid convergence. For all other experiments we used all available views (750 views per object) for training and testing our models.

To evaluate the quality of the feature representation obtained from the visual hierarchy, we compared the use of 8x8x50 C2 shape feature maps, 8x8x(50+3) C2 features with coarse RGB color maps, and plain 64x64x3 pixel RGB images as input $\mathbf{x}_i$ for the STM. The last setting captures the baseline similarity of the plain images in the ensemble, and serves as a reference point, since there are

S_T	input data	#R	error rate
0.55		140	86.9%
0.65	plain	474	76.8%
0.75	color	1956	55.4%
0.85		8832	27.7%
0.90		17906	15.2%
0.55		701	57.8%
0.65		2335	35.1%
0.75	shape	7138	17.2%
0.85		19283	9.4%
0.90		28921	8.3%
0.55		2740	24.5%
0.65	shape	6451	12.5%
0.75	+coarse	14223	6.9%
0.85	color	27104	5.8%
0.90		33831	5.7%

Table 1. Classification rates and number of representatives #R of the online learning oVQ model for different similarity thresholds S_T and different inputs.

method	input data	#R	error rate
oVQ	color+shape	27104	5.8%
iLVQ	color+shape	7304	11.4%
sp. iLVQ	color+shape	3665	9.0%
iLVQ*	color+shape	4574	12.2%
sp. iLVQ*	color+shape	3167	9.9%
oVQ	shape	19283	9.4%
iLVQ*	shape	5500	20.9%
sp. iLVQ*	shape	3780	14.6%

Table 2. Comparison of classification rates of oVQ, incremental LVQ, sparse LVQ, incremental LVQ* with limited short term memory and sparse LVQ* with memory. For all tests we used a similarity threshold of $S_T = 0.85$. The number of selected representatives #R is shown.

currently no other established standard methods for online learning available. Additionally we varied the similarity threshold S_T to investigate the tradeoff between representation accuracy and classification errors. The results are shown in Tab.1. For a fair comparison, error rates for roughly equal numbers of chosen representatives should be compared. The hierarchical shape features reducing the error rates considerably, compared to the plain color images, especially for a small number of representatives $\#R$. The addition of the three RGB feature maps reduces error rates by about one third. For training of all 50 objects that can be done within about three hours, the remaining classification error is about 6% using color and shape and 8% using only shape.

Long-Term Memory and iLVQ. In Tab.2 we show the performance of the iLVQ long-term memory model. We compare the effect of using only a limited memory history for the STM (denoted iLVQ*), in relation to using all data, and the results for the sparse learning rule adaptation described in Sect.2 (denoted sp.iLVQ). The necessary number of representatives #R can be strongly reduced by a factor of 6 and more with the iLVQ network, however, at the price of a slightly reduced classification performance. The differences between the incremental LVQ and the sparse LVQ are that the sparse LVQ reaches slightly better results with fewer number of representatives #R. More important, the sparse iLVQ converges about ten times faster than iLVQ with the standard learning rule, resulting in a training time of only about 3-4 hours. For the experiments using only a limited STM of 10 objects, it can be seen that iLVQ can handle this with almost no performance loss and uses even less resources for representation. We also performed a test using stochastic gradient-based training of linear dis-

criminators (based on the $\mathbf{r}^l$) for each object, where, however the same limited memory history of views from the past 10 objects was applied. Although performance on the current training window of 10 objects normally is below 5% error, the network quickly fails to distinguish objects from the earlier training phases and achieves only a complete final error rate of 73% on all 50 objects.

4 Conclusion

We have shown that the hierarchical feature representation is well suited for online learning using an incremental vector quantization model approach. Of particular relevance is the technical realization of the appearance-based online learning of complex shapes for the context of man-machine interaction and humanoid robotics. This capability introduces many new possibilities for interaction and learning scenarios for incrementally increasing the visual knowledge of a robot. Also for the realistic setting of a limited short-term memory length of 10 objects, we can achieve real-time learning of 50 objects with less than 10% classification error. Although we assume segmentation of the objects in this study, it has been shown previously that the visual hierarchy can also be applied with good results both to learning and recognition of unsegmented objects in clutter [9]. The application to the unsegmented case will therefore be the next step in extending the online object learning approach presented here.

Acknowledgments: We thank C. Goerick, M. Dunn, J. Eggert and A. Ceravola for providing the image acquisition and processing system infrastructure.

References

1. Arsenio, A.: Developmental learning on a humanoid robot. Proc. Int. Joint Conf. Neur. Netw. (2004), Budapest 3167–3172
2. Bekel, H., Bax I., Heidemann G., Ritter H.: Adaptive Computer Vision: Online Learning for Object Recognition. Proc. DAGM, (2004) 447–454
3. Fritzke, B.: A growing neural gas network learns topologies. In G. Tesauro et. al., eds., Adv. Neur. Inf. Proc. Systems. 7. MIT Press, Cambridge MA (1995) 625–632
4. Fukushima, K.: Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. Biological Cybernetics **36 (4)** (1980) 193–202
5. Guedalia, I. D., London, M., Werman, M. An on-line agglomerative clustering method for non-stationary data. Neural Computation, **11(2)** (1999) 521–540
6. Kalinke T., von Seelen, W.: Entropie als Mass des lokalen Informationsgehalts in Bildern zur Realisierung einer Aufmerksamkeitssteuerung. Mustererkennung (1996), Jähne et al., 627–634
7. Kohonen, T.: Self-Organizing and Associative Memory. Springer Series in Information Sciences, Springer-Verlag, third edition (1989)
8. Steels, L., Kaplan, F.: AIBO's first words: The social learning of language and meaning. Evolution of Communication, **vol. 4, no. 1** (2001) 3–32
9. Wersing, H., Körner, E.: Learning Optimized Features for Hierarchical Models of Invariant Object Recognition. Neural Computation **15 (7)** (2003) 1559–1588