Received 15 January 2019; revised 25 March 2019; accepted 8 April 2019. Date of publication 17 April 2019; date of current version 31 May 2019.

Digital Object Identifier 10.1109/JXCDC.2019.2911135

Unsupervised Learning to Overcome Catastrophic Forgetting in Neural Networks

IRENE MUÑOZ-MARTÍN[1] (Student Member, IEEE),
STEFANO BIANCHI[1] (Student Member, IEEE),
GIACOMO PEDRETTI[1] (Student Member, IEEE),
OCTAVIAN MELNIC[1], STEFANO AMBROGIO[2] (Member, IEEE),
AND DANIELE IELMINI[1] (Fellow, IEEE)

[1]Dipartimento di Elettronica, Informazione e Bioingegneria and Italian Universities Nanoelectronics Team, Politecnico di Milano, 20133 Milan, Italy
[2]IBM Research-Almaden, San Jose, CA 95120 USA
CORRESPONDING AUTHOR: D. IELMINI (daniele.ielmini@polimi.it)

This work was supported in part by the European Research Council (ERC) through the European Union's Horizon 2020 Research and Innovation Program under Grant 648635 and in part by the Italian Minister for University and Research under Grant Agreement R164TYLBZP. *Irene Muñoz-Martín and Stefano Bianchi contributed equally to this work.*

This paper has supplementary downloadable material available at http://ieeexplore.ieee.org, provided by the authors.

ABSTRACT Continual learning is the ability to acquire a new task or knowledge without losing any previously collected information. Achieving continual learning in artificial intelligence (AI) is currently prevented by catastrophic forgetting, where training of a new task deletes all previously learned tasks. Here, we present a new concept of a neural network capable of combining supervised convolutional learning with bio-inspired unsupervised learning. Brain-inspired concepts such as spike-timing-dependent plasticity (STDP) and neural redundancy are shown to enable continual learning and prevent catastrophic forgetting without compromising standard accuracy achievable with state-of-the-art neural networks. Unsupervised learning by STDP is demonstrated by hardware experiments with a one-layer perceptron adopting phase-change memory (PCM) synapses. Finally, we demonstrate full testing classification of Modified National Institute of Standards and Technology (MNIST) database with an accuracy of 98% and continual learning of up to 30% non-trained classes with 83% average accuracy.

INDEX TERMS Catastrophic forgetting, continual learning, convolutional neural network (CNN), neuromorphic engineering, phase-change memory (PCM), spike-timing-dependent plasticity (STDP), supervised learning, unsupervised learning.

I. INTRODUCTION

Neural computation currently enables an increasing number of artificial intelligence (AI) tasks such as image recognition [1], face recognition [2], speech recognition, and natural language processing [3]. Artificial neural networks (ANNs) have recently led to significant breakthroughs in object recognition tasks, demonstrating high accuracy in classification with large data sets [4]. This is mostly with regard to the success of supervised training via the backpropagation technique [1], where the synaptic weights are iteratively adjusted in response to the presentation of labeled information. A key limitation of ANNs, however, is the catastrophic forgetting, where training a network on a new task causes the catastrophic loss of the ability of any previously learned task [5]. This issue prevents continual learning, where new features are continuously learned during the whole lifetime of a system [6]. Continual learning is instead possible in the human brain, thanks to the higher flexibility of the biological neuronal network as opposed to the rigid structure of an ANN. Learning in the human brain relies on synaptic plasticity, where synapses are potentiated and depressed according to the mutual spike timing between firing neurons. There are pieces of evidence that spike-timing-dependent plasticity (STDP) [7], [8] is one of the most recurrent learning rules occurring in the brain. As a result, STDP is widely adopted to support unsupervised training in neuromorphic systems [8]–[13]. STDP can also be easily implemented with emerging memory devices, where the change of conductance can be controlled by the timing of the spikes at either the pre-synaptic or postsynaptic ends of the synapse [14], [15]. Although capable of unsupervised learning, STDP cannot generally match the same recognition accuracy and stability of ANNs trained by the backpropagation method [16]. Unsupervised neuromorphic systems and high-accuracy ANNs, thus. feature complementary strengths that fit the so-called “stability/plasticity dilemma” [17], [18].

To achieve high accuracy, high stability of learned tasks, and high flexibility in ANNs, it is essential to combine supervised and unsupervised approaches. Several architectures have been proposed over the years especially from a computer science standpoint aiming to get flexibility in ANNs, including system-level consolidation [19] and synaptic weight consolidation [20], where plasticity in trained synapses are inhibited to prevent forgetting. However, the demonstration of a truly flexible supervised neural network capable of continual learning and verifiable in hardware is not available yet, thus highlighting the necessity of a mixed contribution from supervised and unsupervised approaches.

Here, we present a novel hybrid concept for a supervised-unsupervised neural network able to overcome the “stability/plasticity dilemma.” This architecture is capable of continual learning [6], where the system is tested with new incoming information without catastrophically forgetting previously learned data. The network consists of two parts: the first one is a supervised convolutional neural network (CNN), whereas the second one is an unsupervised STDP classifier. STDP enables fast learning of non-trained patterns by using output neurons that were not already involved in the preliminary training [21]. Our network displays a high accuracy on trained classes, namely, 98% in MNIST data set, and it is capable of accurately classifying up to 30% of untrained classes. In our network, both convolutional filters and STDP synapses are implemented with memory devices [22], such as PCM devices [23] that can be operated in multiple-level analog mode or in binary mode. In the first case, PCMs are programmed in several different equally spaced conductance levels, whereas in the latter one, they work in low or high resistive states. An overall accuracy of 85% is demonstrated after training over the 70% of the data set considering filter discretization due to PCM implementation, thus supporting the mixed supervised/unsupervised design of neuromorphic systems for continual learning.

II. CATASTROPHIC FORGETTING

Fig. 1 illustrates the catastrophic forgetting problem in neural networks: first, the network is trained by supervised training with task A (a), e.g., a subset of a large data set such as the MNIST data set, until task A can be profitably tested (b). Afterward, supervised training of task B is executed on the same network (c), until task B is successfully tested (d). However, the second training causes the total loss of task A due to catastrophic forgetting.

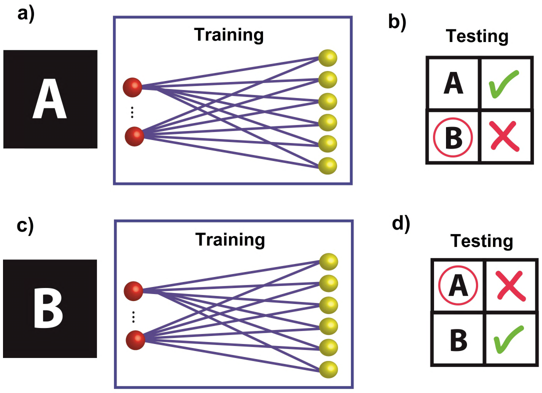

FIGURE 1. **(a) Illustration of the catastrophic forgetting in multilayer perceptron network. (b) After training a certain pattern A, (b) A is recognized, while B is not. (c) If the same network is then trained to learn pattern B, (d) A is forgotten while B is recognized.**

To quantitatively assess the impact of catastrophic forgetting, we trained the MLP shown in Fig. 2(a) with the subset A of the seven MNIST classes 1, 2, . . ., 7. We used the sigmoid as activation function, a learning rate $\eta = 1/35$, a Gaussian weight initialization with $\mu = 0$ and $\sigma = 1/\sqrt{784}$, and the usual regularization algorithms to reduce overfitting. The supervised training leads to a testing efficiency of 97.8% for subset A and no recognition of subset B of the remaining classes 8, 9, and 0, as shown by the confusion matrix of Fig. 2(b). The same network was then trained with subset B, resulting in a testing efficiency of 98.1% for B and no recognition of subset A [Fig. 2(c)], thus evidencing catastrophic forgetting of the previous learned task.

Catastrophic forgetting is a critical problem in machine learning, deeply differentiating the ANNs from the biological networks in the human brain. In biology, the theory of complementary learning systems explains the mutual effort of hippocampus and neocortex to both consolidate previous information and accepting new incoming data [19], [24]. In particular, the hippocampal system accounts for rapid adaptation to new incoming information whereas the neocortex is specialized in consolidating previous knowledge. In the following, the supervised part of our network stands for long-term storage of pretrained information, whereas unsupervised part accepts incoming data for both classifying objects of previously learned classes and storing new data by the STDP protocol.

III. MIXED SUPERVISED-UNSUPERVISED NETWORK

Fig. 3 illustrates the structure of the proposed network, including: 1) a supervised convolutional network; 2) an equalization block to convert the feature maps into normalized patterns with 4 active neurons out of 16; and 3) an unsupervised STDP network as final classifier stage. Each block is described in the following, referring to the learning of

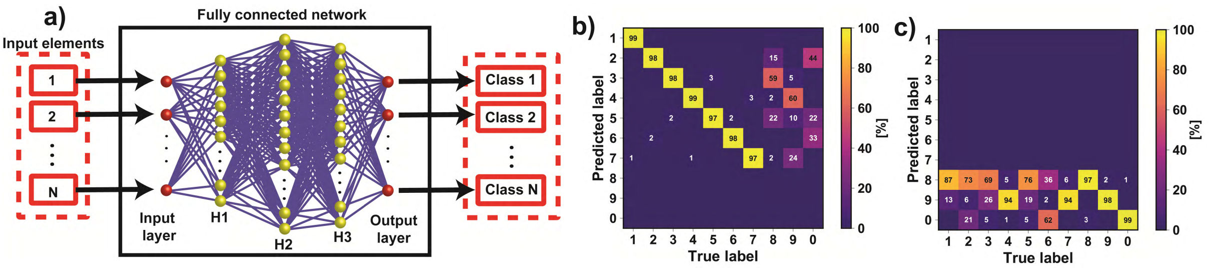

FIGURE 2. (a) Illustration of the multilayer perceptron (MLP) network scheme including three hidden layers, i.e., H1 (600 neurons), H2 (300 neurons), and H3 (150 neurons). We implemented a sigmoidal activation function, a learning rate $\eta = 1/35$, a Gaussian weight initialization with $\mu = 0$ and $\sigma = 1/\sqrt{784}$, and regularization algorithms to control overfitting. (b) Confusion matrix for the testing results after supervised training of the network with the data set A of the Modified National Institute of Standards and Technology (MNIST) classes 1, 2, . . ., and 7. (c) Confusion matrix for the testing results after training the same network first with A, then with B of the MNIST classes 8, 9, and 0. Data set A is forgotten and confused with other classes after training the network with data set B. All the accuracy values shown in the confusion matrix are rounded to zero decimal places.

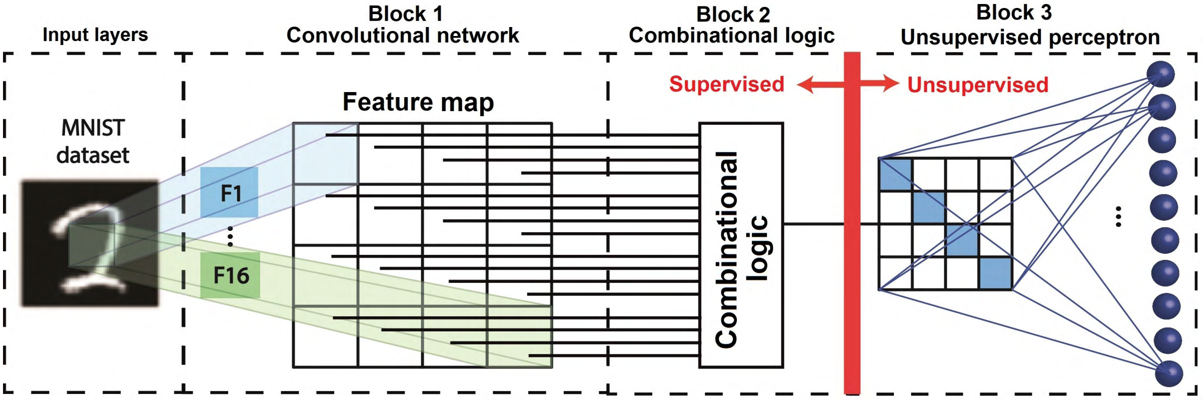

FIGURE 3. General scheme of the network. The input layer includes the input neurons. Block 1 contains the convolutional filters obtained by means of the backpropagation algorithm. The response of the convolutional filters is mapped into the "feature map," resulting in a 4 x 4 matrix. The feature map is then equalized by Block 2, consisting of a combinational logic to transform the 4 x 4 feature map in a 4 x 4 pattern with constant P = 25%. Finally, Block 3 consists of an unsupervised perceptron for the learning of the equalized patterns.

28 × 28 input patterns from the MNIST data set. In the supplementary material, we extend the study to another data set, the "Fashion-MNIST," that groups images of clothes.

A. CONVOLUTIONAL NETWORK

CNNs commonly use filters trained by backpropagation to extract features from input patterns [25]. The responses of the filters are collected into feature maps after convolution of the filters with the input image or with other maps generated by previous convolutional layers.

Zeiler and Fergus [25] showed that, in a CNN, the feature maps usually present original patterns in which some intrinsic features of the original image are highlighted by convolution of the filters. In our network, the input pattern is an MNIST image with dimension 28 × 28, while the filters of the first layer have a dimension of 20 × 20. We chose high filter dimensionality for mainly two reasons: 1) to reduce the number of convolutional steps, thus minimizing the power consumption and 2) to have a binary response from the filters during convolution, i.e., "feature found" and "feature not found." In fact, larger filters than usual gave us the possibility to "memorize" inside a trained filter a simple feature, like a curved line or an angle between two lines, just assuring a fixed bias (or threshold) for each filter. With respect to previous works in [4] and [25], the features are directly visible looking at the filter rather than on the consequent activation maps, i.e., the results coming from convolution. We decided to implement a CNN from scratch with the aforementioned constraints respect to usual programming. It would be possible to use different kinds of training procedures to extract the filters, but we essentially used two approaches within the same CNN, as described in Fig. 4 for the full data set.

In the first approach [Fig. 4(a)], the CNN is trained to directly recognize a specific class of patterns, e.g., the class "1" in the MNIST data set. Convolution of the filter with the input pattern yields a 9 × 9 output pattern, which is reduced to a 1 × 1 pattern by a max-pool operation. A total of N_T class filters are used, aiming at recognizing the first N_T classes of the MNIST data set, thus resulting in an output layer of dimension $N_T \times 1 \times 1$ (it is possible to choose the number of class filters to train). The filters are trained by backpropagation so that each of the output layer neurons responds to one and only one class of the input pattern, e.g., the first output neuron should respond only to the presentation of a "1," and so on.

In the second approach [Fig. 4(b)], nine filters are used to extract a 9 × 9 × 9 pattern, which is then passed to another convolutional layer with a total of N_F filters, each

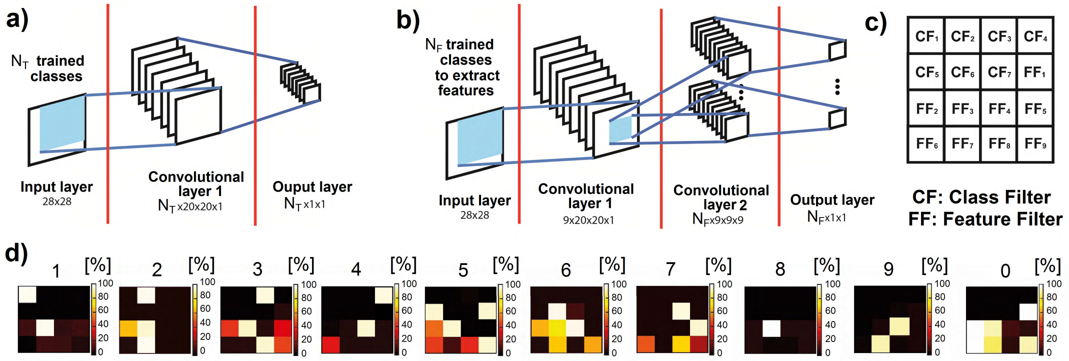

FIGURE 4. Schematic of the (a) CNN for the supervised training of class filters and (b) supervised training of feature filters. (c) Output feature map, summarizing all the responses of the class filters ($CF_1, \ldots, CF_7$) and feature filters ($FF_1, \ldots, FF_9$). During testing, the class/feature filters provide a combination of binary responses (“feature found” or “feature not found”), resulting in the 16 output signals in the figure. (d) Feature maps collecting the output of the CNN in terms of the probability of finding a certain feature in response to the presentation of patterns from each MNIST class, from 0 to 9.

with dimension $9 \times 9 \times 9$, thus resulting in an output layer of dimension $N_F \times 1 \times 1$. The filters are trained by backpropagation as in the first approach for N_F trained classes. Note that the N_F trained classes are not necessarily the same N_T trained classes.

During testing, the overall response of Block 1 is obtained combining the N_T “class filters” shown in Fig. 4(a) with the nine “feature filters” from the first convolutional layer shown in Fig. 4(b). We used pooling operation to select an individual response for each filter, class, and feature. We fixed $N_T = 7$, so a total of 16 output binary signals are used to recognize any incoming pattern, as shown in Fig. 4(c). Note that, if the data set has to be fully trained, the procedure shown in Fig. 4(b) must account, at least, for all the classes of the data set not included in the training shown in Fig. 4(a).

Fig. 4(d) shows the average resulting feature maps, namely, the probability of fire of each neuron collecting the signal from one class/feature filter shown in Fig. 4(a) and (b). All 10 classes of MNIST patterns were used during the supervised training of 60 000 images. Note that the first seven output neurons fire in response to the presentation of one and only one class, e.g., neuron 1 fires in response to the presentation of class “1” and so on. On the other hand, the remaining nine output neurons show a unique combination of firing in response to each MNIST input class. The generalized “feature” filters are key to enable the recognition of non-trained classes by transfer learning [26], which do not belong to the group of the trained classes and thus cannot be directly recognized by a specialized class filter. In the following, in fact, we will demonstrate continual learning of up to 30% non-trained classes, and we will keep constant the number of class filters at 7 and of feature filters at 9.

B. COMBINATIONAL LOGIC

Although the feature maps shown in Fig. 4(d) are unique for each class of the data set, they cannot be directly fed to the STDP network, as they have different pattern densities P, defined as the number of firing neurons divided by the total number of neurons [16]. In the STDP network, in fact, each postsynaptic neuron (POST) compares the incoming current from the pre-synaptic neurons (PREs) with an internal threshold [27]. A different P for each class would cause unfair competition between various patterns in the winner-take-all (WTA) network [28]. To prevent instabilities in the STDP network, a normalization block was thus added to equalize P among the various feature maps shown in Fig. 4(d).

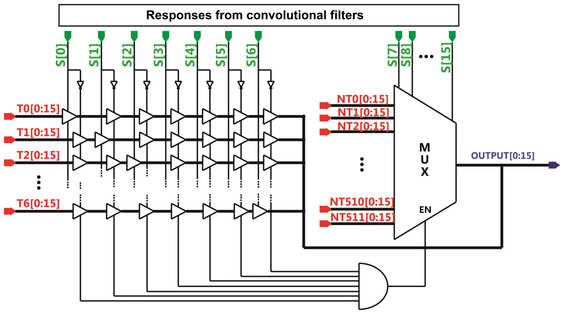

FIGURE 5. Combinational logic for pattern equalization to generate 4 x 4 patterns with uniform pattern density P = 25%. Signals S0–S15 represent the output of Block 1, i.e., the feature maps. These signals select the proper output bus from T0 to T15, or from NT0 to NT511.

Fig. 5 shows a combinational logic which can be used for equalization. There is no need for training the combinational logic, which can be extended to any data set with the same number of classes as the MNIST. As for the supervised part of the network, note that equalization can be achieved by various design implementations, of which Fig. 5 is just one possible solution. Our combinational logic transfers each of the feature maps shown in Fig. 4(d) to a unique equalized pattern of dimension 4×4, shown in Fig. 6(a). In the combinational logic, there are two main blocks which process the input to yield a specific equalized pattern, contained in a 16-bit serial stream. The first block is aimed at transferring the first seven signals, which are specialized on a specific trained class (from S0 to S6), to seven equalized output patterns (from T0 to T6). For instance, if a pattern of the first class is presented, signal S0 is high, which activates the transfer of the externally fed equalized pattern T0 to the output of the combinational logic. The second block, instead, deals with the case when an input class is not trained to respond to a class filter. In this case, the multiplexer assigns a specific non-trained equalized pattern (NT0, NT1, etc.) to each combination of feature signals (from S7 to S15). Note that the system is capable of assigning more equalized patterns than the number of input classes, which is just 10 for the MNIST data set. This is crucial to implement neuron redundancy in the STDP network, which enables efficient unsupervised learning of patterns for non-trained classes.

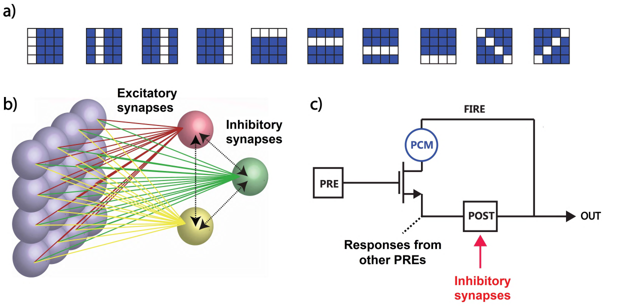

FIGURE 6. (a) Examples of ten equalized patterns with P = 25% from the output of the combinational logic. (b) Schematic of the inhibitory and excitatory synapses in STDP network. (c) Excitatory synapses have 1T1R structure with PCM elements to store the synaptic weight. Excitatory synapses connect the PREs to the POST, while inhibitory synapses are responsible for the competition between POSTs for the WTA behavior of the network.

C. UNSUPERVISED PERCEPTRON

The third block shown in Fig. 3 consists of a classification layer which is trained by STDP and WTA processes. Fig. 6(b) schematically shows an STDP network for the simplified case of three output neurons. The submission of one of the equalized patterns of Fig. 6(a) causes fire in one of the POSTs, and thus the potentiation of the corresponding excitatory synapses by STDP and zeroing of all other output neurons via the inhibitory synapses [29]. STDP can be achieved by the 1T1R synapse shown in Fig. 6(c), including a transistor and a resistive memory, such as a PCM [29] or a resistive switching memory (RRAM) device [30]. Noise patterns can also be submitted randomly to induce depression of previous information stored in the synapses, thus enabling the reconfiguration of the network to a new data set [27]. Each of the POST specializes in one of the equalized patterns, while the WTA process prevents the specialization of more than one POST to the same input pattern. The STDP and WTA processes allow the unsupervised learning of all equalized patterns, hence the recognition of the input data set. By the fact the artificial pattern is trained, the STDP and WTA processes are easier because we know from training procedure the initial combination of synaptic weights.

To experimentally demonstrate the unsupervised network shown in Fig. 6(b), we implemented a spiking neural network (SNN) with 4 × 4 input channels fully connected to 3 POSTs by 48 PCM synapses with 1T1R structure [Fig. 6(c)]. After initializing all synapses in low or high resistive states, we randomly submitted three equalized patterns, corresponding to the first, the second, and the third column, respectively. Fig. 7(a) shows the color map of the final synaptic weights of synapses connected to the three POSTs after the three patterns were submitted for 5 s with a presentation time of 10 ms. Fig. 7(b) shows the synaptic weights as a function of time, whereas Fig. 7(c) shows the spiking activity of the POST. The average conductivity of synapses within the pattern increases with time, while the synaptic conductance in the background decreases due to the uncorrelated noise causing depression. These results support the unsupervised learning by STDP in the classification layer.

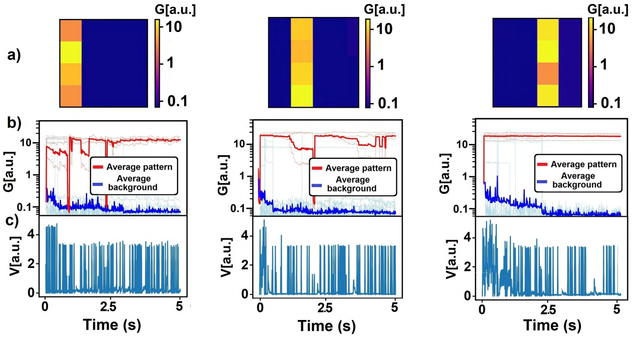

FIGURE 7. Experimental demonstration of unsupervised learning with STDP of three equalized patterns, namely, the first, the second, and the third column. (a) Synaptic weights at the end of the unsupervised learning session. (b) Time evolution of synaptic weights for both pattern and background synapses. (c) POST spiking activity.

IV. CONTINUAL LEARNING

To test the ability of the system in continual learning, we trained the network with a fraction of the MNIST data set, namely, only seven classes out of ten, and subsequently tested the recognition over the entire testing data set with ten output neurons for classification via STDP. The network was organized with seven class filters and nine feature filters as discussed in Section III-A. Although class filters provide high accuracy in detecting the trained classes, the feature filters enable unsupervised learning of non-trained classes. Fig. 8 shows the output of the CNN (Block 1 shown in Fig. 3), namely, the 4 × 4 feature maps for both trained classes (1, 2, 3, 4, 5, 6, and 7) and the remaining non-trained classes (0, 8, and 9). After the supervised training with seven classes, the presentation of the three non-trained classes results in consistent feature maps, which can then be used for the following equalization and unsupervised learning. Note that non-trained classes generally show a negligible response to class filters, which have been specialized to the trained classes. Instead, non-trained classes respond only to feature filters, which are general enough to identify specific features in any pattern. Each of the binary feature maps that can be obtained can thus be equalized (Block 2 shown in Fig. 3) and

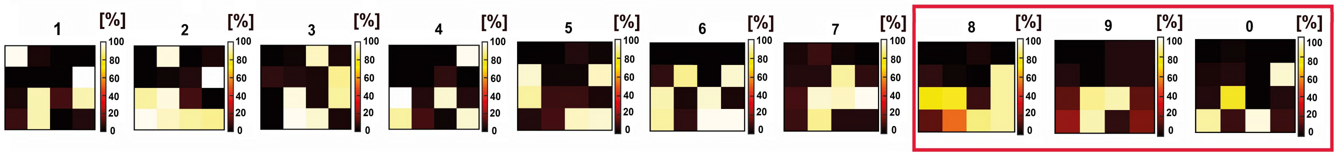

FIGURE 8. **Response to the feature maps for seven trained classes (from 1 to 7) and three non-trained classes (8, 9, and 0). Sixteen convolutional filters are used, seven for recognizing a particular trained class and the others to extract particular features from the trained classes. Feature filters differentiate the new incoming classes by an original combination of responses. With respect to Fig. 4(d), there are three non-trained classes (8, 9, and 0) that are recognized for transfer learning.**

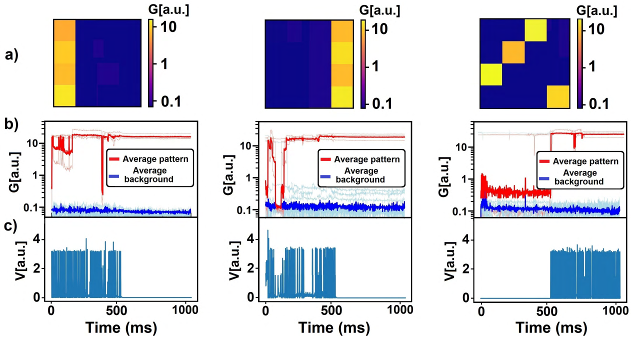

FIGURE 9. **Experimental demonstration of continual learning with STDP, supporting the capability for learning sequential patterns. In the first 500 ms, only two patterns (first and fourth columns) are presented at the input. Then, the third pattern is presented. The first two POSTs remain specialized in the vertical patterns while the third one specializes in the third presented pattern.**

submitted to the STDP layer (Block 3 shown in Fig. 3) for unsupervised learning. The STDP layer is, in fact, the key block for the unsupervised learning of both trained and non-trained patterns, thus enabling to overcome catastrophic forgetting and to achieve continual learning.

To support continual learning by the STDP layer, Fig. 9 shows the experimental results for the same 3-POST perceptron shown in Fig. 6(b), where two patterns (first and fourth column, representing trained patterns) are initially presented for 500 ms, followed by the presentation of a third non-trained pattern for 500 ms, to mimic the sequential training shown in Fig. 1. Fig. 9(a) shows the final conductance for synapses connected to the first, the second, and the third POST, while Fig. 9(b) shows the synaptic weights as a function of time, and Fig. 9(c) shows the firing activity of the three POSTs. The first two patterns are readily learned in the first phase, thanks to the POST specialization in the WTA network, while the later submission of the additional pattern leads to unsupervised learning in the synapses of the third POST without affecting the previously trained synapses. These results support the flexibility of the STDP network, which is capable of overcoming catastrophic forgetting and achieving continual learning.

A. NEURONAL REDUNDANCY

Although trained classes can be uniquely identified by their response to class filters, non-trained classes show stochastic variations in their response to feature maps, which cannot be uniquely equalized by Block 2. This is because patterns belonging to the same class do not show exactly the same features in the same position, thus resulting in some variations in the feature map response. This ambiguity can be overcome by assigning an equalized pattern to each possible feature map of the non-trained classes, which thus need more than just one neuron in the STDP layer for correct learning. A similar neuronal redundancy is indeed found in the motor cortex of the human brain [31].

Fig. 10(a) shows the simulation results for the whole network considering various cases of non-trained classes, reporting the average accuracy and their standard deviation 2σ, as a function of the average number of output neurons for each class. Simulations were carried out for an increasing number of non-trained classes, from zero to three. The adopted procedure to train the neural network is the following: first, we trained the convolutional filters according to the technique discussed in Section III-A, with various numbers of trained and non-trained classes of the MNIST data set. The number of "class filters" and "feature filters" were kept constant, i.e., 7 and 9, respectively. Then, we tested the accuracy of the whole network in recognizing the trained and non-trained classes assuming 1, 2, or 3 average output neurons per input class. The simulation results in the figure indicate that the recognition accuracy increases for an increasing number of output neurons, reaching an accuracy of 83% for the correct classification of three non-trained classes, i.e., 30% of the patterns were not presented during the supervised training.

Fig. 10(b) shows the calculated accuracy for trained classes, non-trained classes, and their average as a function of the number of non-trained classes. The results are compared with an MLP network affected by catastrophic forgetting. In the simulations, we assumed 3 output neurons for each class. As the number of non-trained classes increases, the recognition accuracy of the trained classes slightly increases because the supervised training becomes easier with less competition between trained classes. On the other hand, the overall recognition accuracy decreases reaching, on average, 93% considering three non-trained classes, which is a dramatic improvement with respect to a standard MLP or CNN network shown in Fig. 2. In fact, it is impossible to continually learn if only MLP or CNN are implemented, as the addition of new patterns requires the training of the overall network from scratch. Even a split–apply–combine approach to mimic STDP by repeating the training several

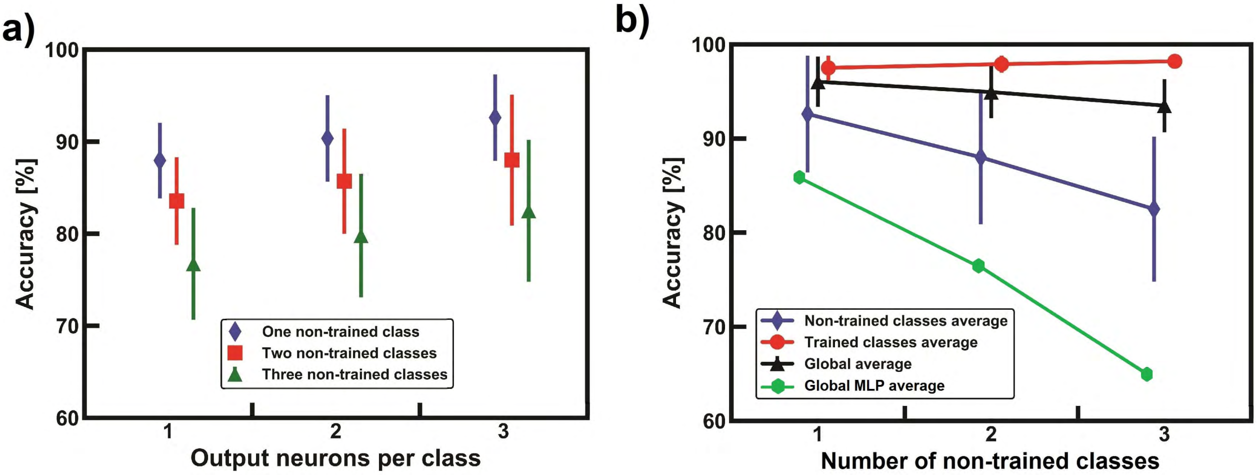

FIGURE 10. (a) Testing accuracy for 1, 2, or 3 non-trained classes as a function of the average number of output neuron per class. (b) Testing accuracy of trained and non-trained classes for a neuronal redundancy of three output neurons per class in the STDP layer, considering 1, 2, or 3 non-trained classes. The results are compared with the global MLP accuracy affected by the catastrophic forgetting.

times on different subdatasets to eventually merge the results, would imply new weights, a decrease of the global MLP accuracy of trained classes, and additional power consumption. On the other hand, the brain-inspired approach of unsupervised learning allows reusing previous knowledge from the supervised training to learn new patterns, where the new knowledge affects only the second layer of the unsupervised STDP network. Thus, the results of Fig. 10(a) and (b) support brain-inspired techniques such as STDP and neuronal redundancy as a very promising method to overcome the limitations of supervised networks, such as catastrophic forgetting, and to achieve continual learning.

B. PERFORMANCE OF THE NETWORK

Fig. 11 shows the confusion matrices, namely, the probability of firing of the output neuron (predicted label) as a function of the submitted class (true label). The figure considers both the case of full training (a) and the case where three classes, namely, 8, 9, and 0, were not presented during the supervised training (b). A neuronal redundancy of three neurons per class was assumed. Although the average accuracy drops from 98% for full training (a) to 93% for continual learning (b), the network overcomes catastrophic forgetting, which cannot be avoided in a fully connected network, as reported in Fig. 2. The results support the mixed supervised-unsupervised approach to achieve continual learning as in the human brain.

V. FULL IMPLEMENTATION WITH PCM DEVICES

Although PCM provides an ideal implementation of plastic synapses for unsupervised learning via STDP (Figs. 7 and 9) [29], neural networks with supervised training also can take advantage of PCM, thanks to its excellent scaling and multilevel state operation [32]–[34]. To demonstrate PCM synapses in the CNN block shown in Fig. 3, we programmed multilevel states in PCM devices, as shown in Fig. 12(a). The figure shows the conductance distributions for ten levels, which were approximately equally spaced between the fully crystalline (set) state and the fully amorphized (reset) state. These levels were then assumed as quantized states for the class and feature filters in Fig. 4.

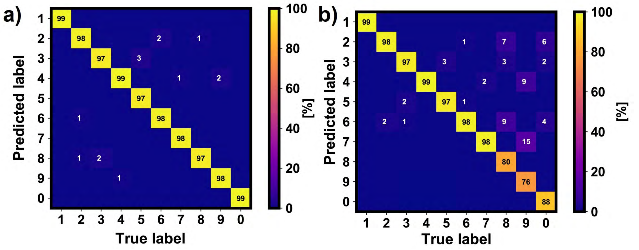

FIGURE 11. (a) Testing results for the case of full training, namely, all ten classes of the MNIST data set were presented during the supervised training. The average accuracy is 98%. (b) Testing results considering seven trained classes (1, . . ., 7) and three not trained classes, namely, 8, 9, and 0. The average accuracy decreases to 93%, which is still much larger than the extremely low accuracy of the fully connected network in Fig. 2. The values in the confusion matrices are rounded.

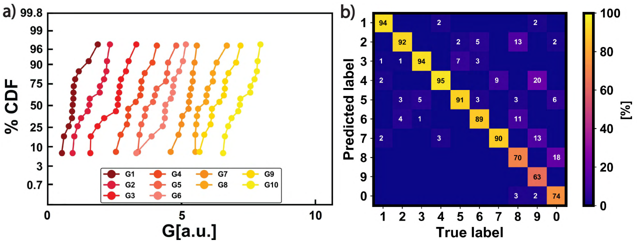

FIGURE 12. (a) Distributions of measured conductance of PCM devices used as analog synaptic elements in the convolutional filters. (b) Confusion matrix for the testing accuracy in the Monte Carlo simulations considering seven trained classes (1, . . ., 7) and three non-trained classes (8, 9, and 0). The overall accuracy is 85.2%. The values in the confusion matrix are rounded.

To represent both positive and negative filter values, we first normalized them into 21 levels between the maximum negative to the maximum positive value. Each quantized filter value G_{syn} was then obtained from the ten PCM levels shown in Fig. 12(a) the difference between two synaptic conductances G_+ and G_-, according to $G_{\mathrm{syn}} = G_+ - G_-$ [35]. Then, we carried out Monte Carlo simulations, where we assumed the average values and standard deviations of G_{syn} according to Fig. 12(a). Fig. 12(b) shows the confusion matrix of the simulated accuracy assuming seven trained classes, namely, 1, 2, 3, 4, 5, 6, and 7, and three non-trained classes, namely, 8, 9, and 0. The average accuracy is around 85.2%, where the 8% decrease with respect to Fig. 11(b) can be attributed to quantization and stochastic variation of the PCM conductance in the filter. Overall, the results support the feasibility of a supervised/unsupervised network combining scalability, flexibility against catastrophic learning, thanks to STDP and accuracy, and supervised training of CNN filters.

VI. CONCLUSION

We present a novel neural network, capable of overcoming catastrophic forgetting by combining supervised and unsupervised learning. The hybrid network can combine supervised training and brain-inspired algorithms, such as STDP and neuronal redundancy, to enable continual learning. Reconfigurable connection between the supervised and unsupervised blocks is achieved thanks to an equalization layer consisting of a logic network. We study the network performance in terms of classification accuracy and robustness against catastrophic forgetting. Results indicate, on average, an accuracy of 93% during full testing with three non-trained classes and seven trained classes. We demonstrate that this network is compatible with a full implementation of PCM synapses, in both the supervised CNN and the unsupervised STDP. This work highlights the relevance of brain-inspired techniques for enabling continual learning in AI systems by a combination of supervised and unsupervised approaches.

ACKNOWLEDGMENT

This work was supported in part by the European Research Council (ERC) through the European Union's Horizon 2020 Research and Innovation Programme under Grant Agreement 648635 and in part by the Italian Minister for University and Research under Grant Agreement R164TYLBZP. S. Ambrogio was with the Dipartimento di Elettronica, Informazione e Bioingegneria and Italian Universities Nanoelectronics Team, Politecnico di Milano, 20133 Milan, Italy.

REFERENCES

[1] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, May 2015.

[2] S. Lawrence, C. L. Giles, A. C. Tsoi, and A. D. Back, "Face recognition: A convolutional neural-network approach," *IEEE Trans. Neural Netw.*, vol. 8, no. 1, pp. 98–113, Jan. 1997.

[3] J. Hirschberg and C. D. Manning, "Advances in natural language processing," *Science*, vol. 349, no. 6245, pp. 261–266, Jul. 2015. doi: 10.1126/science.aaa8685.

[4] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. NIPS*, vol. 1. 2012, pp. 1097–1105.

[5] M. McCloskey and N. J. Cohen, "Catastrophic interference in connectionist networks: The sequential learning problem," *Psychol. Learn. Motivat.*, vol. 24, pp. 109–165, Dec. 1989.

[6] G. I. Parisi, R. Kemker, J. L. Part, C. Kanan, and S. Wermter. (2018). "Continual lifelong learning with neural networks: A review." [Online]. Available: https://arxiv.org/abs/1802.07569

[7] H. Markram, W. Gerstner, and P. J. Sjöström, "Spike-timing-dependent plasticity: A comprehensive overview," *Frontiers Synaptic Neurosci.*, vol. 4, no. 2, p. 3, Jul. 2012.

[8] G. Q. Bi and M. M. Poo, "Synaptic modifications in cultured hippocampal neurons: Dependence on spike timing, synaptic strength, and postsynaptic cell type," *J. Neurosci.*, vol. 18, no. 24, pp. 10464–10472, 1998.

[9] M. Gilson, A. N. Burkitt, and L. van Hemmen, "STDP in recurrent neuronal networks," *Frontiers Comput. Neurosci.*, vol. 4, p. 16, Sep. 2010.

[10] R. Guyonneau, R. VanRullen, and S. Thorpe, "Neurons tune to the earliest spikes through STDP," *Neural Comput.*, vol. 17, no. 4, pp. 859–879, Apr. 2005.

[11] A. Saudargiene, B. Porr, and F. Wörgötter, "Synaptic modifications depend on synapse location and activity: A biophysical model of STDP," *Biosystems*, vol. 79, nos. 1–3, pp. 3–10, Jan. 2005.

[12] U. Weidenbacher and H. Neumann, "Unsupervised learning of head pose through spike-timing dependent plasticity," in *Proc. Int. Tutorial Res. Workshop Perception Interact. Technol. Speech-Based Syst.*, vol. 79, 2008, pp. 123–131.

[13] T. Masquelier and S. J. Thorpe, "Unsupervised learning of visual features through spike timing dependent plasticity," *PLoS Comput. Biol.*, vol. 3, no. 2, pp. 247–257, 2007.

[14] D. Kuzum, S. Yu, and H.-S. P. Wong, "Synaptic electronics: Materials, devices and applications," *Nanotechnol.*, vol. 24, no. 38, p. 22, Sep. 2013.

[15] I. Boybat *et al.*, "Neuromorphic computing with multi-memristive synapses," *Nature Commun.*, vol. 9, p. 2514, Jun. 2018.

[16] G. Pedretti, S. Bianchi, V. Milo, A. Calderoni, N. Ramaswamy, and D. Ielmini, "Modeling-based design of brain-inspired spiking neural networks with rram learning synapses," in *IEDM Tech. Dig.*, Dec. 2017, pp. 28–31.

[17] S. Grossberg, "Competitive learning: From interactive activation to adaptive resonance," *Cognit. Sci.*, vol. 11, no. 1, pp. 23–63, 1987.

[18] A. Robins, "Catastrophic forgetting, rehearsal and pseudorehearsal," *Connection Sci.*, vol. 7, no. 2, pp. 123–146, 1995.

[19] D. Kumaran, D. Hassabis, and J. L. McClelland, "What learning systems do intelligent agents need? complementary learning systems theory updated," *Trends Cognit. Sci.*, vol. 20, no. 7, pp. 512–534, 2016.

[20] K. James *et al.*, "Overcoming catastrophic forgetting in neural networks," *Proc. Nat. Acad. Sci. USA*, vol. 114, no. 13, pp. 3521–3526, Mar. 2017.

[21] C. Zamarreño-Ramos, L. A. Camuñas-Mesa, J. A. Pèrez-Carrasco, T. Masquelier, T. Serrano-Gotarredona, and B. Linares-Barranco, "On spike-timing-dependent-plasticity, memristive devices, and building a self-learning visual cortex," *Frontiers Neurosci.*, vol. 5, p. 26, Mar. 2011.

[22] T. Serrano-Gotarredona, T. Masquelier, T. Prodromakis, G. Indiveri, and B. Linares-Barranco, "STDP and STDP variations with memristors for spiking neuromorphic learning systems," *Frontiers Neurosci.*, vol. 7, no. 2, pp. 1–15, Feb. 2013.

[23] S. Raoux, W. Wełnic, and D. Ielmini, "Phase change materials and their application to nonvolatile memories," *Chem. Rev.*, vol. 110, no. 1, pp. 240–267, 2010.

[24] J. L. McClelland and B. L. McNaughton, "Why there are complementary learning systems in the hippocampus and neocortex: Insights from the successes and failures of connectionist models of learning and memory," *Phychological Rev.*, vol. 102, no. 3, pp. 409–419, Jul. 1995.

[25] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, vol. 8689. 2014, pp. 818–833.

[26] L. Torrey and J. Shavlik, "Transfer learning," *Handbook of Research on Machine Learning Applications and Trends: Algorithms, Methods, and Techniques*, vol. 7. Pennsylvania, PA, USA: IGI Clobal, 2009, pp. 242–264.

[27] G. Pedretti *et al.*, "Memristive neural network for on-line learning and tracking with brain-inspired spike timing dependent plasticity," *Sci. Rep.*, vol. 7, no. 1, p. 5288, 2017.

[28] G. Pedretti *et al.*, "Stochastic learning in neuromorphic hardware via spike timing dependent plasticity with RRAM synapses," *IEEE J. Emerg. Sel. Topics Circuits Syst.*, vol. 8, no. 1, pp. 77–85, Feb. 2017.

[29] S. Ambrogio *et al.*, "Unsupervised learning by spike timing dependent plasticity in phase change memory (PCM) synapses," *Frontiers Neurosci.*, vol. 10, p. 56, Mar. 2016.

[30] S. Ambrogio *et al.*, "Neuromorphic learning and recognition with one-transistor-one-resistor synapses and bistable metal oxide RRAM," *IEEE Trans. Electron Devices*, vol. 63, no. 4, pp. 1508–1515, Apr. 2016.

[31] K. Takiyama and M. Okada, "Maximization of learning speed in the motor cortex due to neuronal redundancy," *PLOS Comput. Biol.*, vol. 8, no. 12, 2012, Art. no. e1002348.

[32] S. Yu, "Neuro-inspired computing with emerging nonvolatile memory," *Proc. IEEE*, vol. 116, no. 2, pp. 260–285, Feb. 2018.

[33] G. W. Burr *et al.*, "Neuromorphic computing using non-volatile memory," *Adv. Phys. X*, vol. 2, no. 1, pp. 89–124, 2017.

[34] S. Ambrogio *et al.*, "Equivalent-accuracy accelerated neural-network training using analogue memory," *Nature*, vol. 558, no. 7708, p. 60, 2018.

[35] M. Suri *et al.*, "Phase change memory as synapse for ultra-dense neuromorphic systems: Application to complex visual pattern extraction," in *Proc. Int. Electron Devices Meeting*, Aug. 2011, pp. 4.4.1–4.4.

IRENE MUÑOZ-MARTÍN (S'19) received the B.S. and M.S. degrees in industrial engineering from the Universidad Politécnica de Madrid (UPM), Madrid, Spain, in 2014 and 2017, respectively, and the M.S. degree in electronics engineering from Politecnico di Milano, Milan, Italy, in 2017, where she is currently pursuing the Ph.D. degree in information technology.

Her current research interests include the integrated design of neuromorphic networks with resistive switching devices.

STEFANO BIANCHI (S'19) received the B.S. and M.S. degrees *(cum laude)* in electrical engineering from Politecnico di Milano, Milan, Italy, in 2015 and 2017, respectively, where he is currently pursuing the Ph.D. degree in information technology.

His current research interests include modeling and design of neuromorphic networks with resistive switching memories.

GIACOMO PEDRETTI (S'17) received the B.S. and M.S. degrees in electrical engineering from Politecnico di Milano, Milan, Italy, in 2013 and 2016, respectively, where he is currently pursuing the Ph.D. degree in electrical engineering.

His current research interests include design and characterization of neuromorphic networks for beyond-CMOS computing systems.

OCTAVIAN MELNIC received the B.S. and M.S. degrees in electrical engineering from Politecnico di Milano, Milan, Italy, in 2015 and 2018, respectively, where he is currently pursuing the Ph.D. degree in electrical engineering.

His current research interests include characterization and modeling of phase change memories.

STEFANO AMBROGIO (M'16) received the B.S., M.S. *(cum laude)*, and the Ph.D. degrees in electrical engineering from Politecnico di Milano, Milan, Italy, in 2010, 2012, and 2016, respectively.

He is currently a Post-Doctoral Researcher with IBM Research-Almaden, San Jose, CA, USA. His current research interests include nonvolatile memory and cognitive computing.

Dr. Ambrogio was a recipient of the IEEE EDS Rappaport Award in 2015.

DANIELE IELMINI (SM'09–F'19) received the Ph.D. degree in nuclear engineering from Politecnico di Milan, Milan, Italy, in 2000.

In 2002, he joined the Dipartimento di Elettronica, Informazione, e Bioingegneria of Politecnico di Milano, Politecnico di Milano as an Assistant Professor, where he became an Associate Professor in 2010. Since 2016, he has been a Professor with the Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano. He conducts research on emerging nanoelectronic devices, such as phase-change memory (PCM) and resistive switching memory (RRAM).

Dr. Ielmini was a recipient of the Intel Outstanding Researcher Award in 2013, the ERC Consolidator Grant in 2014, and the IEEE EDS Rappaport Award in 2015.