


Article

# Towards New Generation, Biologically Plausible Deep Neural Network Learning

Anirudh Apparaju and Ognjen Arandjelović \* 

School of Computer Science, University of St Andrews, St Andrews KY16 9SX, UK

\* Correspondence: ognjen.arandjelovic@gmail.com

**Abstract:** Artificial neural networks in their various different forms convincingly dominate machine learning of the present day. Nevertheless, the manner in which these networks are trained, in particular by using end-to-end backpropagation, presents a major limitation in practice and hampers research, and raises questions with regard to the very fundamentals of the learning algorithm design. Motivated by these challenges and the contrast between the phenomenology of biological (natural) neural networks that artificial ones are inspired by and the learning processes underlying the former, there has been an increasing amount of research on the design of biologically plausible means of training artificial neural networks. In this paper we (i) describe a biologically plausible learning method that takes advantage of various biological processes, such as Hebbian synaptic plasticity, and includes both supervised and unsupervised elements, (ii) conduct a series of experiments aimed at elucidating the advantages and disadvantages of the described biologically plausible learning as compared with end-to-end backpropagation, and (iii) discuss the findings which should serve as a means of illuminating the algorithmic fundamentals of interest and directing future research. Among our findings is the greater resilience of biologically plausible learning to data scarcity, which conforms to our expectations, but also its lesser robustness to additive, zero mean Gaussian noise.

**Keywords:** plasticity; inhibition; Hebbian; local; backpropagation



**Citation:** Apparaju, A.; Arandjelović, O. Towards New Generation, Biologically Plausible Deep Neural Network Learning. *Sci* **2022**, *4*, 46. <https://doi.org/10.3390/sci4040046>

Academic Editors: Andrea Prati and Claus Jacob

Received: 16 October 2022  
Accepted: 29 November 2022  
Published: 1 December 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Artificial neural networks and in recent years especially deep neural networks of various kinds have proven to be highly successful across a wide range of different machine learning tasks, application domains, and learning modalities [1–6]. Notwithstanding these successes, the design and the training of such networks is characterized by a number of weaknesses, both of a practical and a fundamental nature. One such weakness of particular importance concerns the backpropagation algorithm used for training [7]. First, end-to-end backpropagation is notoriously data demanding; that is, the network must be exposed to a large number of training exemplars in order to learn effectively [8,9]. It is also highly susceptible to adversarial attacks [10–12]. These aspects are in sharp contrast to the learning performed by “natural”, biological neural networks that comprise animal brains, including human brains. The contrast between the two is brought to an even sharper focus by the understanding of the latter stemming from decades of neurological research. Succinctly, the adaptational processes that underlie the learning of biological neural networks are nothing like the backpropagation algorithm. Although in backpropagation the information about a network’s current output and the desired output is, as the work itself suggests, backpropagated across the entire network, in biological neural networks the adaptations are largely local. This observation strongly motivates a different approach to artificial neural networks, with the aim of achieving more robust as well as more efficient learning. In the present paper, we describe, evaluate, and analyse an artificial neural network trained with a biologically plausible, semi-supervised learning algorithm incorporating Hebbian synaptic plasticity inspired adaptation.

## 2. Background and Motivation

### 2.1. End-to-End Backpropagation

End-to-end backpropagation is a technique for training multilayer neural networks [7]. Succinctly, after each forward pass through the network, backpropagation uses the chain rule to perform a backward pass to propagate errors to train the network by adjusting the model's synaptic weights. The technique was developed by researchers in the 1960s, but the first to propose using it on neural networks was Werbos in his 1974 Ph.D. thesis [13,14]. Despite solving the question of how multilayer neural networks could be trained, and testing the process while working on his PhD thesis, Werbos did not publish it until 1982 due to the chilling effects of the so-called AI Winter at the time [15]. He even proposed the paper to Marvin Minsky while visiting MIT to no response [14]. It was a decade later that the method was popularized by Rumelhart et al. [16]. In spite of a solution being available years ago, it was this paper in 1986 that showed how multilayer neural networks could be trained, and resolved issues pertaining to limitations of the perceptron that were raised decades earlier by Minsky and Papert [17]. The potential power of artificial neural networks was only properly utilized after this popularisation of end-to-end backpropagation [16], and it has been very effective in facilitating their application in a diverse range of complex tasks including the processing of natural language and images [18], game playing [1,2,19], and more recently, autonomous navigation.

#### 2.1.1. Biological Implausibility of End-to-End Backpropagation

In contrast to artificial neural networks themselves, which were developed as simplified models of biological neural networks, the end-to-end backpropagation algorithm used to train them draws little inspiration from nature and, moreover, is biologically implausible. This implausibility challenges the functional correctness of biological brain models that utilize end-to-end backpropagation in their learning processes.

In short, the following are the main reasons for questioning the biological realism of the end-to-end backpropagation algorithm.

- First is its lack of local error representation and nonlocal learning rules. The classic end-to-end backpropagation algorithm uses a nonlocal rule for updating the weights between nodes, whereas Hebb's work demonstrates that a change in synaptic strength in biological neural network learning should depend only on the neurons local to that synapse [20].
- Second is the symmetry of forward and backward weights. In artificial neural networks, the weights during forward (information) and backward (error) propagation are identical for the same synapse. This symmetry suggests that identically weighted forward and backward connections should exist between biological neurons; however, this is not the case [21].
- Third are the data demands. The end-to-end backpropagation algorithm requires a very large labelled dataset, whereas biological neural systems use unsupervised learning with observations from their extensive sensory experience to train feature detectors [22].
- Fourth are the unrealistic models of neurons. By and large, artificial neural networks use neurons that fire a continuous output whereas real neurons output discrete spikes [23]; in other words, while natural neurons either fire or not (are "on" or "off"), the output of artificial ones exhibit a smoothly behaving degree of action potentiation.

Hence, researchers have striven to develop new models of artificial neural networks by using biologically plausible learning algorithms so as to try to provide more effective insights into processes in the brain [21]. There is a diverse range of such algorithms, which we discuss shortly. Studies in this domain have helped researchers, especially neuroscientists, to build better neurocomputational models that help to better understand and investigate learning processes in the brain [21,22,24–29]. However, herein the interest is in the impact of the biologically plausible algorithms in the field of machine learning

rather than in neuroscience. As a result, spiking neurons are not considered, and discussion is limited to exploring algorithms that attend to the first three issues above.

### 2.1.2. Limitations of End-to-End Backpropagation

From a purely machine learning standpoint, despite the great success of deep learning models trained by using end-to-end backpropagation, models developed in this way suffer several inherent weaknesses. These limitations arise because artificial neural networks trained by using end-to-end backpropagation are:

- highly data demanding, often requiring millions of labelled training examples to learn effectively [8,9],
- easily fooled by adversarial examples [10–12],
- computationally intensive to train and deploy (GPUs required, and sometimes even TPUs) [30],
- susceptible to algorithmic bias [31],
- poor at representing and conveying uncertainty (how can one know what the model knows?) [32],
- difficult in complementing with structure and prior knowledge during learning [33],
- uninterpretable black boxes (which do not easily establish trust) [34], and
- require expert knowledge for their design and fine-tuning [35].

All of these have provided strong motivation into research on biological plausible learning algorithms and raised the question as to what extent the aforementioned limitations can be overcome with an alternative learning strategy.

### 2.2. Biologically Plausible Learning

Aided by the availability of image datasets of varying levels of complexity (MNIST, CIFAR-10, ImageNet), there is an increasing body of work dedicated to developing algorithms that perform similar training as end-to-end backpropagation but without breaking some of the fundamental rules of neurobiology. These biologically plausible algorithms adopt a wide range of algorithmic ideas including supervised learning algorithms like feedback alignment [24,27] or target propagation [28], and various semisupervised learning techniques [25,26,29]. Although many of these algorithms were developed with the primary intention of understanding more about the learning processes of the brain, they have also demonstrated interesting properties and promising behaviour in the context of standard machine learning evaluations tasks on widely used benchmark data sets such as MNIST, CIFAR-10 and ImageNet [25,26,29].

Of particular interest herein are semisupervised algorithms [25,26,29], as the receptive fields generated by these witness progress toward local, bottom-up unsupervised training that is capable of learning useful and task-independent representations [36]. These learned feature descriptors are then used to form a basis of a model that is thereafter exposed to additional *labelled* data in the supervised part of training. As Whittington and Bogacz observe [21], there is a clear benefit to propagating information via fewer synapses, as this facilitates a faster adaptive response and a reduction in the possible origins of noise which, as we noted earlier, are indeed some of the reasons which motivate this line of research. The aforementioned features are shown to be necessary to achieve good generalization performance comparable with networks trained with classic end-to-end backpropagation.

### 2.3. A Balancing Act

The receptive fields generated by all of the aforementioned algorithms support the claim that local bottom-up unsupervised training is capable of learning useful and task-independent representations of images in networks [36]. The feature descriptors are then used as a base of a model that is supplied labelled data in the supervised part of training. These useful features are necessary to achieve a good generalization performance in line with networks trained with end-to-end backpropagation. Furthermore, the results from experiments conducted on these networks suggest that localised receptive fields enable

better generalization than networks with full connectivity [25,36] to the extent that networks with localised receptive fields can reach classic backpropagation performance on the MNIST data set.

That being said, there is a balance to strike between generalization performance and network efficiency and simplicity. In the present work, we follow this neuroscientific theme of favouring efficient and simple architectures as a means of elucidating the advantages and disadvantages of biologically plausible learning as best as the present state of the art allows.

Furthermore, Grinberg et al. [36] best summarize an important finding that describes an important difference between the supervised and unsupervised approaches we highlighted previously. They conclude that supervision is not crucial for learning useful early layer representations from the data, which is arguably at odds with the common belief that the first layer feature detectors should be crafted specifically to solve the narrow task specified by the top layer classifier [36]. Hence, this unsupervised approach not only has biological plausibility, overcoming the limitation of the computationally demanding nature of training and deployment of end-to-end backpropagation-based approaches, but also has strong generalisation performance that makes a strong case for its further study.

#### 2.4. Scalability

Scalability is an important concept to consider when evaluating the performance of a learning algorithms. There are several factors that contribute to the scalability aspect of a model's performance—CPU cycle count, memory usage, and generalization performance amongst others—but the one of greatest importance in the context of the present work is the generalization performance of the learning algorithm to more advanced and complex tasks.

The importance of scalability of generalization performance of learning algorithms is best outlined by Bartunov et al. who argued that the quest should be for learning algorithms that are both more plausible physiologically and that scale up to the sorts of complex tasks that humans are capable of learning. As the crux of their argument is the observation that augmenting a model with adaptive capabilities is unlikely to unveil any truths about the learner if its performance is excessively limited by the learning algorithm. In other words, the premise is that the key to exploring and improving the impact of a learner is not in perpetual augmentation of its model with additional capabilities but rather in the learning algorithm itself. Although the former would work in actually improving empirical performance, they would not necessarily provide deep insight into the learning process. Hence, following the type of methodology and analysis performed by Bartunov et al. [26] and Illing et al. [29], the impact of a biologically plausible learning algorithm in the present article is measured by observing the generalization performance on different datasets.

### 3. Related Work

#### 3.1. Supervised Learning Approaches

##### 3.1.1. Target Propagation

The main idea underlying target propagation is to associate with each feedforward unit's activation value a target value rather than a loss gradient. These computed targets, like gradients, are propagated backward. In a way that is related but different from previously proposed proxies for backpropagation, which rely on a backward network with symmetric weights, target propagation relies on auto-encoders at each layer.

Lee et al. describe how this general idea of target propagation by using autoencoders to assign targets to each layer can be employed for supervised training of deep neural networks [28]. Their experiments show that target propagation performs on a comparable level to backpropagation on ordinary deep networks and de-noising autoencoders. Moreover, target propagation can be directly used on networks with discretized transmission between units and reaches state-of-the-art performance for stochastic neural networks on the MNIST dataset.

### 3.1.2. Feedback Alignment

The feedback alignment (FA) algorithm was proposed by Lillicrap et al. [21,27] as an effective and simple solution to the weight symmetry problem presented in the previous section.

The main idea underlying feedback alignment is that when propagating errors backward, a random matrix is adopted rather than the transpose of the synaptic weight matrix used on the forward pass. Put another way, this algorithm assigns random feedback weights to synapses. Feedback alignment addresses the limitation of weight symmetry present in classic backpropagation. This idea is supported by studies which have shown that good generalisation performance on classification tasks can be achieved by randomly backpropagating errors [26]. Moreover, this mechanism transmits such error signals across multiple layers of neurons and is shown to perform training as effectively as the classic end-to-end backpropagation on several tasks [27].

The feedback alignment algorithm has inspired a number of related approaches, two of the best ones being direct feedback alignment (DFA) and indirect feedback alignment (IFA) [18]. Although the original FA algorithm showed that the weights used for propagating the error backward need not be symmetric with the weights used for forward propagation of the neuron activation, the DFA and IFA algorithms can run on networks that disconnect the feedforward path from the feedback path so that each layer is not reciprocally connected to the layer above (and below).

What sets feedback alignment apart from target propagation is that the former relies on implicit dynamics inherent in the algorithm to evolve the weight matrices, whereas the latter relies on autoencoders, a tangible construct of the network. This implies more simplicity in network architectures that incorporate FA: a theme that is a major factor of interest to the present work.

## 3.2. Semi-Supervised Learning Approaches

A notable amount of prior work, focuses on the examination of the kind of data representations that unsupervised and semi-supervised learning algorithms are extracting (i.e., the features detected) in a biologically plausible manner [25]. We summarize these.

### 3.2.1. Autoencoders (AEs) and Restricted Boltzmann Machines (RBMs)

A popular unsupervised learning approach is to train a hidden network layer to reproduce the input data as done in various AEs and RBMs [37]. AE and RBM networks trained with a single hidden layer are important because adjusting weights of the input-to-hidden-layer connections relies on local gradients, and the representations can be stacked on top of each other to extract hierarchical features.

### 3.2.2. Unsupervised Learning with Hidden Competing Units

The recently proposed Krotov–Hopfield model [22] addresses the problem of learning with local gradients by learning hidden representations solely by using an unsupervised method. In the network, the input-to-hidden connections are trained and additional (nonplastic) lateral inhibition provides competition within the hidden layer. For evaluating the representation, the weights are kept fixed, and a linear classifier trained with labels is used for the final classification. Moreover, this unsupervised algorithm was also used to establish weights in a locally connected convolutional neural network (CNN) [36].

### 3.2.3. Bayesian Confidence Propagation Neural Network (BCPNN)

The feedforward BCPNN model is a probabilistic graphical model with a single hidden layer. It frames the update and learning steps of the neural network as probabilistic computations, the mechanics of which are structurally outlined by Ravichandran et al. [25].

### 3.3. Competitive Learning

Competitive learning refers to a unsupervised learning paradigm which provides a way to discover the salient, general features which can be used to classify a set of patterns [38]. The mechanics of this learning paradigm involves a competitive mechanism that is capable of discovering a set of feature detectors that capture important aspects of input stimulus patterns. Rumelhart and Zipser analysed these learning processes to further the understanding of how simple adaptive networks can discover features important in the description of the stimulus environment the system finds itself in [38]. This learning paradigm is appealing because the salient features learned are useful for classification tasks.

The basic components of such a competitive learning scheme are:

- the starting state comprising a set of units with a randomly distributed parameter which enables each unit to respond slightly differently to inputs;
- limited unit “strength” [25]; and
- unit competition for the “winner-takes-all” right to respond to input.

Applying these ideas to a learning paradigm enables units of the model to learn to specialize on groups of similar patterns (thus becoming feature detectors) [38]. These feature detectors can then be used as starting weights in a semi-supervised model which is supplied labelled data in its supervised component. In this regard, competitive learning can be considered a form of representation learning whereby a good representation is, by definition, one that makes downstream learning easier; for example good representations may be learned from unlabelled data, and then be used in subsequent supervised learning tasks [37].

### 3.4. Learning Robustness

The importance of model robustness as a topic of study was well summarized by Hendrycks et al. [39] by noting the human vision system is robust in ways that existing computer vision systems are not [40,41]. Unlike current deep learning classifiers [42–44], the human vision system is not fooled by small changes in query images. Humans are also not confused by many forms of corruption such as snow, blur, pixelation, or indeed combinations of these. Humans can even deal with abstract changes in structure and style. Achieving these kinds of robustness is an important goal for computer vision and machine learning. It is also essential for creating deep learning systems that can be deployed in safety-critical applications. Most of the work on evaluating robustness of deep learning models for vision has targeted the following challenges [39]: robustness to adversarial examples [10–12]; unknown unknowns [45–47]; and data/model poisoning [48,49]. In addition, Hendrycks et al. [39] conducted a comprehensive series of experiments on the ImageNet dataset to establish rigorous standards on image classifier robustness. Specifically, unlike other recent robustness research, their work evaluates performance on common corruptions and perturbations as opposed to worst-case adversarial perturbations [39]. This comprehensive analysis establishes a precedent and guide as to how to properly explore and evaluate a model’s robustness.

## 4. Our Biologically Plausible Model

### 4.1. Overall Structure

As regards its coarse, high-level structure, our model is a biologically plausible artificial neural network whose architecture consists of one fully connected network with a single hidden layer: there is a layer of visible neurons  $v_i$ , a layer of hidden neurons  $h_\mu$ , and a layer of output neurons  $c_\alpha$ . Thus, we can identify two separately trained components of the network. The first of these consists of hidden layers trained in an unsupervised manner by using a new kind of Hebbian synaptic plasticity based adaptation rule. The second component is a fully connected perceptron trained by using the conventional stochastic gradient descent. This structure requires no backpropagation of signals past the final layer, which utilizes stochastic gradient descent, and which is the only place where any form of gradient descent is applied. Furthermore, all of the information in the network is

transmitted only from forward-propagating signals. This behaviour gives credence to the biologically plausible description, in that all the synapse updates happen locally, with no information propagated from any subsequent nodes in the network, an issue discussed previously in Section 2.1.1. The supervised and unsupervised components are discussed in further detail shortly.

### Forward Pass

The forward pass on the network is simple and can be formally described as

$$h_\mu = r(W_{\mu i}v_i) \tag{1}$$

$$c_\alpha = \tanh(\beta S_{\alpha\mu}h_\mu), \tag{2}$$

where

$$r(x) = \begin{cases} x^n, & x \geq 0 \\ 0, & x < 0 \end{cases} \tag{3}$$

and  $\mu$  is the index of the hidden unit being updated,  $W_{\mu i}$  the  $i$ -th synapse weight that corresponds to the unit,  $\beta$  and  $n > 0$  constants, and  $S_{\alpha\mu}$  are the weights associated with the network's top layer, and  $r(\cdot)$  the modified rectified linear unit (ReLU).

## 4.2. The Unsupervised Component

### 4.2.1. Activations of Hidden Units

At the foundation of the proposed method are the dynamic processes which can be described by using differential equations. The key one defines the steady state activity of the hidden units. Others describe a form of Hebbian learning with competition between hidden units [22]. Often succinctly described by the maxim "cells that fire together wire together", Hebbian or associative learning describes a form of synaptic plasticity whereby the synaptic strength between cells which fire together in response to a stimulus is increased. By its very nature, this form of adaptation is local, though distal effects emerge as a consequence of the propagation and accumulation of local change. It is also worth noting that a strict observance of Hebb's rule also requires an observation of causality, in that the timing of neural activation is important: the synapse strength between two neurons is increased only if the excitatory neuron also fires before the excited one; mere joint firing is not enough [50].

Thus, to reiterate, this stage is unsupervised and thus requires no labels corresponding to the input data presented to the network.

After exposing the network to training data, the activity of the hidden neurons are calculated according to the following differential equation which defines the dynamics that lead to the steady-state activations of the hidden units:

$$\tau \frac{\partial h_\mu}{\partial t} = I_\mu - \omega_{inh} \sum_{v \neq \mu} r(h_v) - h_\mu, \tag{4}$$

where  $\tau$  represents a positive constant that defines the overall timescale of these dynamical processes,  $\mu$  is the index of the hidden unit being updated,  $h_\mu$  the activity of the hidden layer,  $\omega_{inh}$  the parameter that defines the strength of the global inhibition,  $r(\cdot)$  is the rectified linear activation function activation function, and  $h_v$  is the activity of other hidden layer nodes. The term  $\omega_{inh} \sum_{v \neq \mu} r(h_v)$  can be seen as introducing competition between hidden units. When all the data is presented to the network, initially all the hidden units start to get activated. However, if some become more strongly activated than others, those which are more strongly activated end up suppressing the activations of the others, the net effect being that of introducing inhibitory connections between hidden nodes. Lastly, the term  $I_\mu$  represents the input current, computed as the dot product of the vector of weights and the incoming data:

$$I_\mu = \langle \mathbf{W}, \mathbf{v} \rangle = \sum_{i=1}^N W_{\mu i} v_i, \tag{5}$$

where  $N$  is the number of input nodes of the network.

#### 4.2.2. Temporal Competition

Rather than basing our model purely on the principles of Hebbian learning, the activity of the post-synaptic cell is further modulated by a nonlinear function. The effect of this modulation is that the synapses of strongly driven hidden units are even more strongly pushed toward the patterns that drive them, while the synapses of those driven less are pushed away from them. Given a random temporal sequence of the input stimuli, the result is a creation of a dynamic competition between the hidden units and results in the synaptic weights that are different for each hidden unit and, importantly, specific to features of the data.

The modulating function  $g(\cdot)$  is a simple piecewise linear function:

$$g(h) = \begin{cases} 1, & h^* \leq h \\ -\lambda, & 0 \leq h < h^* \\ 0, & h < 0. \end{cases} \tag{6}$$

Intuitively, the value of  $h^*$  flips the nature of the learning taking place: for  $h \geq h^*$ , Hebbian learning is effected, whereas for positive  $h < h^*$  it becomes *anti-Hebbian* learning. Activities that are below zero, corresponding to  $h < 0$ , do not contribute to training and are effectively ignored.

#### 4.2.3. Synaptic Plasticity Rule

Our plasticity rule performs the weight updates corresponding to the network synapses and can be seen as an extension of the Oja rule [51]. The rule is captured by a dynamic process represented as a differential equation:

$$\tau_L \frac{\partial W_{\mu i}}{\partial t} = g(h_\mu) [v_i - (I_\mu W_{\mu i})], \tag{7}$$

where  $\tau_L$  is a positive constant that defines the overall timescale of the learning dynamics (and  $\tau_L \gg \tau$  to capture the longer time scale of the process),  $W_{\mu i}$  represents the synapse weights that correspond to the index of the hidden unit that is being updated, and  $g(\cdot)$  is the Hebbian learning term described previously. The purpose of the term  $[v_i - (I_\mu W_{\mu i})]$  is to ensure that the synaptic weights do not grow to excess. This constraint is inspired by neuroscientific models of biological learning where homeostatic constraints ensure that the biological neural synapses do not become too strong. Moreover, its specific form is chosen so as to ensure that the fixed point of the dynamics in the long term (i.e., as time tends to infinity) satisfies the following constraint:

$$\sum_{i=1}^N W_{\mu k}^2 = R^p. \tag{8}$$

In other words, the weights connecting one hidden unit with all visible units eventually converge to the surface of a sphere of radius  $R$  defined by using the Lebesgue norm  $p$ .

#### 4.2.4. Approximate but Fast(er) Learning

A major limitation of the described learning process as formulated thus far lies in its time demands, that is to say, the learning process is slow. This slowness emerges from two key sources. The first one is the requirement to present training exemplars to the network in a sequential fashion, rather than in batches as usual with conventionally designed and trained artificial neural networks [52]. The other emerges from the very fundamentals



of the network design which inherently demands a large number of iterations for the hidden units to reach their steady state. Herein, we adopt an approximate learning algorithm proposed by Krotov and Hopfield [22] which is explained next for completeness.

The first deviation from the exact learning procedure concerns the dynamical equations, which are not solved exactly. Rather, the current is used as a proxy for ranking of the final activities, thereby pushing the unit that responds the most to a particular training example toward it with activation  $g = 1$  and lower-ranked units away with activation  $g = -\lambda$  as per (6). Clearly, this alteration significantly reduces the computational burden of training and thus reduces the associated time requirement. The second speed-up is conferred by the organization of training examples into minibatches, the aforementioned ranking now being performed for an entire minibatch and the weight updates being averaged over the minibatch.

#### 4.2.5. Summary

In summary, during the first, unsupervised part of training, the network is presented with data (be it in the form of minibatches or in an online fashion), whereafter as governed by the process captured by (4), adaptations take place until the system reaches the equilibrium. After a steady state of hidden neurons is achieved, the synaptic plasticity learning rule described by (7) is iteratively applied to adjust the weights until convergence.

The weight update dependencies can be succinctly summarised as

$$\Delta W_{\mu i} \sim g(h_{\mu})v_i, \quad (9)$$

where the symbols used have the same meaning as heretofore. The idea of using lateral inhibition in conjunction with Hebbian synaptic plasticity is in part inspired and motivated by work on competitive, or winner-takes-all, learning [16,38,53]. Indeed, global inhibition has already demonstrated promising potential [54–56].

This unsupervised algorithm in our model accepts raw data as input and seeks to find a useful representation of the data. This phase is given no explicit task specific knowledge, that is no information about the specific goal the representations ought to be useful for is taken advantage of (e.g., classification, regression etc.). Hence, these learned representations can be likened to the high-level features learned in the feature learning stage of convolutional neural networks (CNNs), and the unsupervised learning stage as a whole can be seen as performing a kind of representation learning [57]. As we show in the next section, the feature detectors learnt by the unsupervised algorithm are similar to those learned by the early convolutional layers of CNNs applied to the same data, and they resemble the responses of neurons in early visual processing areas of the brains of animals.

#### 4.3. The Supervised Component

Following the first, unsupervised training stage, the supervised part of our model's learning is applied only to the very last synapses and the corresponding nodes of the network, which altogether can be treated as a fully connected perceptron. This part of the model is trained by using any of a number of variations which fall under the broad umbrella of classical stochastic gradient descent techniques [58]. It is the only part of the entire network wherein labelled data and stochastic gradient descent are used. The specific configurations used in our experiments are described in Section 5.

## 5. Experimental Analysis

### 5.1. Implementation and Hardware

We used the PyTorch framework to implement and run experiments with the proposed model which is contrasted with a fully connected neural network trained with end-to-end backpropagation, (henceforth referred to as a "classic NN"). All the experiments and training were performed on a machine with a 6 GB NVIDIA GeForce GTX 1060 graphics card by using CUDA version 10.2, and with the Linux distribution CentOS Linux release 7.8.2003.

### 5.2. Datasets

For the sake of replicability and ease of comparison with other published work, two widely used, publicly available datasets were adopted, namely MNIST and CIFAR-10, with a random 80:20 split in the validation stage (training–validation split), and the final testing stage (training–test split).

### 5.3. Architectures

Two fully connected neural networks with one hidden layer of 2000 hidden units were used. To be specific, the network shape for MNIST classification is (784 → 2000 → 10), whereas the shape for CIFAR-10 classification is (3072 → 2000 → 10). For each dataset, our network was trained with the described semi-supervised learning algorithm (as detailed in Section 4.2.4, approximate learning was used for the unsupervised part of the learning), and was compared with a classic neural network based model, trained using the usual end-to-end backpropagation.

### 5.4. Cost Function

The supervised part of the training was done by using the loss function (labels  $t_\alpha$  are one-hot encoded vectors of  $N_c = 10$  units of  $\pm 1$ ):

$$C = \sum_{\text{examples}} \sum_{\alpha=1}^{N_c} |c_\alpha - t_\alpha|^m, \tag{10}$$

where  $\alpha$  represents an index that corresponds to the index of the unit in the final output layer,  $N_c$  represents the number of output neurons,  $c_\alpha$  represents the prediction made by the network in the form of a vector of  $N_c$  units,  $t_\alpha$  represents the actual labels in the form of one-hot-encoded vectors of  $N_c$  units, and  $m$  represents a constant and serves as a hyperparameter.

### 5.5. Hyperparameters

The hyperparameters of the models were learned by using the standard approach whereby data was repeatedly randomly split in proportion 80:20 into training and validation subsets, the training of models performed on the former subset, and the fitness of a specific hyperparameter set assessed by measuring performance on the latter subset. The results are summarized in Tables 1–3.

**Table 1.** Learned hyperparameter values for the unsupervised part of the biologically plausible model.

Data Set	$p$	$k$	$\lambda$	Batch Size	Epoch #	Learning Rate
MNIST	2	2	0.4	100	1000	0.02 → 0.00
CIFAR-10	2	2	0.3	100	1000	0.02 → 0.00

**Table 2.** Learned hyperparameter values for the supervised part of the biologically plausible model.

Data Set	Optimizer	Batch Size	Epoch #	Learning Rate	$m$	$n$	$\beta$
MNIST	Adam	100	600	0.0001	6	4.5	0.01
CIFAR-10	Adam	100	600	0.004 → 0.00001	6	4.5	0.01

**Table 3.** Learned hyperparameter values for the classic neural network model.

ine Data Set	Optimizer	Batch Size	Epoch #	Learning Rate	$m$	$n$	$\beta$
MNIST	Adam	100	600	0.001 → 0.00001	6	1	0.01
CIFAR-10	Adam	100	600	0.004 → 0.001	4	1	0.01

### 5.6. Baseline Compare

We compare the proposed network with a fully connected neural network as the contrasting baseline. As mentioned previously, we are focused on using as simple a model as possible so as to deconfound the problem and to direct our attention on the learning algorithm itself, and not on any specific attributes or characteristics of the network architecture. Hence, we employ a fully connected network with a single hidden layer with all weights learned by using conventional end-to-end backpropagation; the network uses the activation function described in Section 4.2.1.

### 5.7. Experiments

We start our evaluation by establishing the baseline classification performance of the proposed model and the classic neural network on MNIST and CIFAR-10 data sets. Then we perform a series of comparative scarcity and robustness experiments, allowing us to gather further insight into the salient differences between the two types of learning.

#### 5.7.1. Feature Detectors

The early-layer weights learned by the different networks form their feature detectors. These feature detectors represent what the models view as the underlying structure of the image data. Moreover, to reiterate for emphasis, the biologically plausible model learns these feature detectors in a purely unsupervised manner as opposed to the completely supervised processes underlying the classic neural network.

The unsupervised process of the biologically plausible model generates weights that appear to be readily comprehensible and interpretable by humans, and certainly much more so than those obtained by using the classic neural network model, as readily evidenced by the visualization of the same in Figures 1 and 2. Moreover, the unsupervised phase of the biologically plausible model training generates feature detectors that have areas that are both positively and negatively correlated. These negative elements suggest that these feature detectors are not just copies of the training images, which would contain only positive elements.

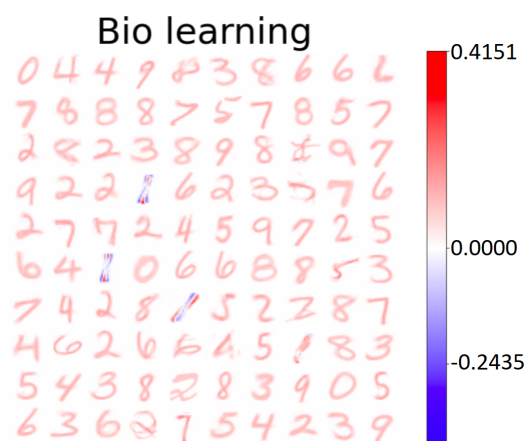
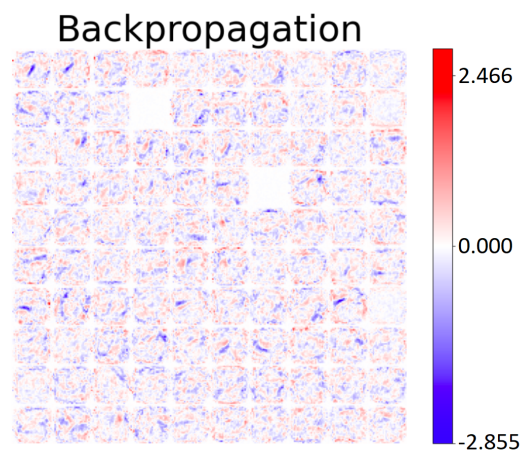
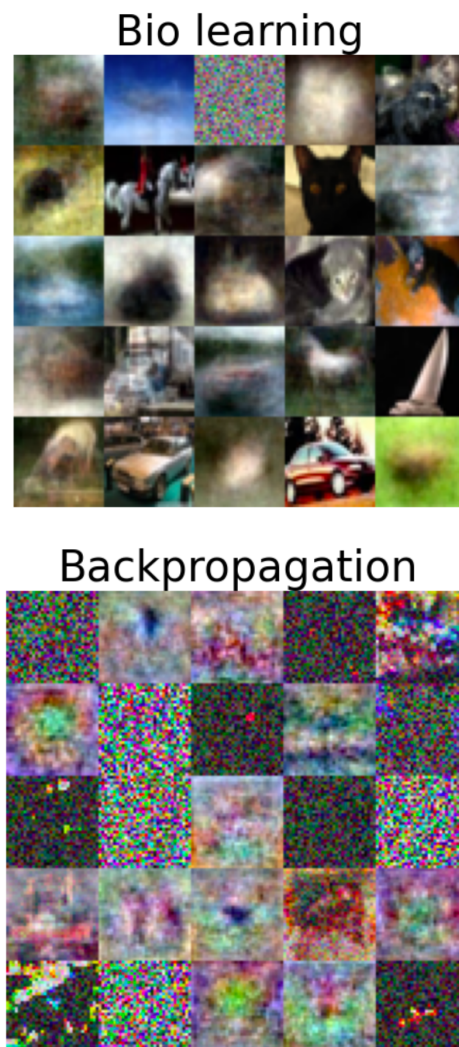


Figure 1. Cont.



**Figure 1.** A hundred randomly selected feature detectors for MNIST classification out of 2000 are shown. On the top are the shown weights learned by the biologically plausible model’s unsupervised learning component; on the bottom are the weights learned by the classic neural network model by using end-to-end backpropagation.



**Figure 2.** Shown are 25 randomly selected feature detectors for CIFAR-10 classification out of 2000. On the top are the weights learned by the biologically plausible model’s unsupervised learning component; on the bottom are the weights learnt by the classic neural network model using end-to-end backpropagation.

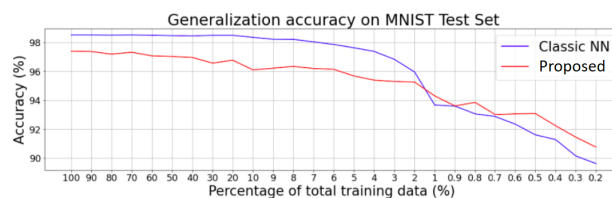
### 5.7.2. Data Scarcity

These experiments evaluate how well the biologically plausible model can overcome the well-known data-hungry limitation of ANNs trained by using classic end-to-end backpropagation. Generalization performance on the hold-out test set is measured for both models after the amount of labelled training data was constricted.

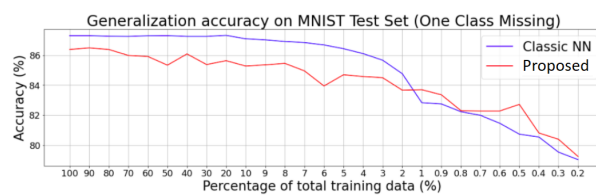
The scarcity of data is introduced into training in multiple ways. First, note that there are three versions of MNIST and CIFAR-10 datasets, namely: (i) with all the data labels present and intact; (ii) with one class of labelled images completely missing; and (iii) with half of the classes of labelled images completely missing. Moreover, for each of these three datasets, the total amount of labelled data is limited further by reducing the corpus thereof available for training the models. Specifically, each version of the labelled training dataset just listed is undersampled according to the following: (i) 100–10% of the training data in decremental steps of 10%; (ii) 10–1% of the training data in decremental steps of 1%; (iii) 1–0.2% of the training data in decremental steps of 0.1%. In this manner, we ensured that for each dataset, regardless of the level of undersampling, there is a balanced number of exemplars of each class presented to the models. Lastly, we conducted experiments by removing entire label sets from the training corpus to evaluate how well the models can classify all images when presented with many that have not been seen at all in the labelled training stage.

### Results

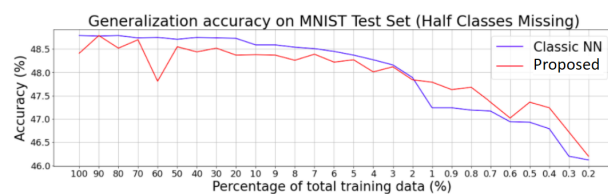
The generalization performance results (based on test set accuracy) show that when a sufficiently large amount of labelled training data is available, the classic neural network performs better than the biologically plausible model. However, the biologically plausible model achieves better generalization performance than the classic neural network when labelled data is highly limited ( $\leq 1\%$  of total available labelled training data for MNIST and  $\leq 4\%$  for CIFAR-10). The generalization performance degradation profiles for both models are shown in Figures 3 and 4.



(a)

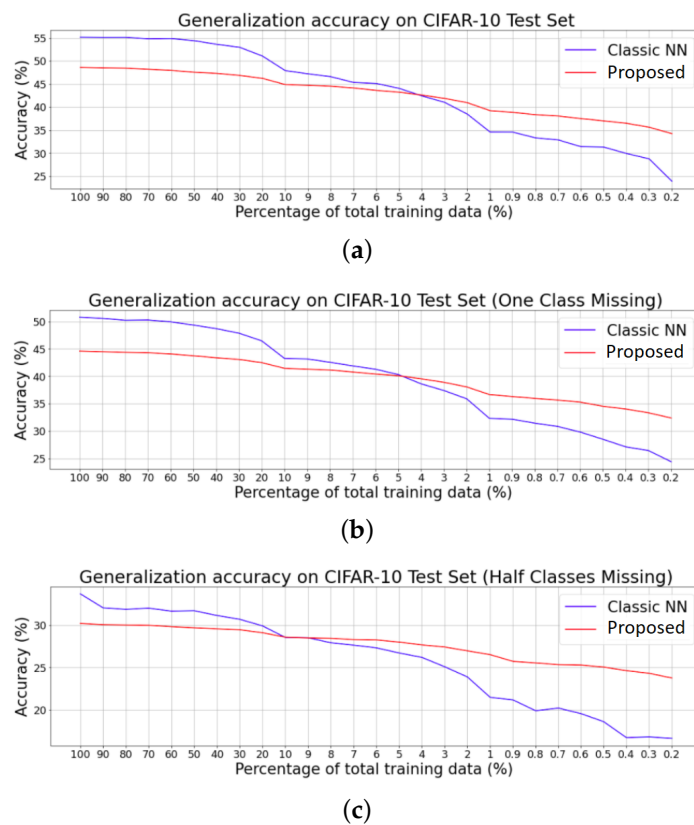


(b)



(c)

**Figure 3.** Results of our data scarcity experiments on the MNIST dataset: (a) all classes present in training, (b) class ‘1’ missing from training, and (c) half of classes missing in training.



**Figure 4.** Results of our data scarcity experiments on the CIFAR dataset: (a) all classes present in training, (b) class ‘car’ missing from training, and (c) half of classes missing in training.

The experiments further indicate that the biologically plausible model performs better than the classic neural network model when labelled data is scarce. The experimental results and feature detectors suggest that the biologically plausible model’s unsupervised component was able to generate an effective latent space representation of the input data which is also readily interpretable. The form of this representation forms a key difference between the biologically plausible and classic neural network models, and plays an important role in the observed differences in generalization performances under conditions of labelled data scarcity.

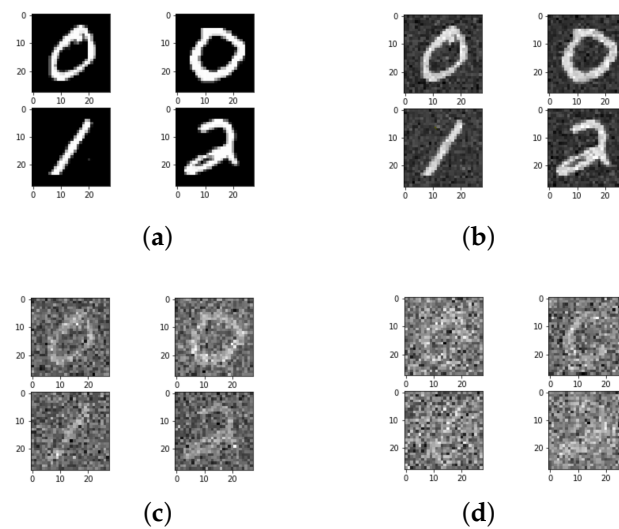
Moreover, it is interesting to note that the difference in generalization performance is greater for the CIFAR-10 data set than for MNIST. This behaviour provides evidence in favour of there being a connection between the complexity inherent in the data in a particular corpus and the discrepancy in generalization performance between the two learning approaches when labelled training data is scarce, with the benefits of biologically plausible learning coming to the fore in such cases.

### 5.7.3. Robustness to Data Corruption

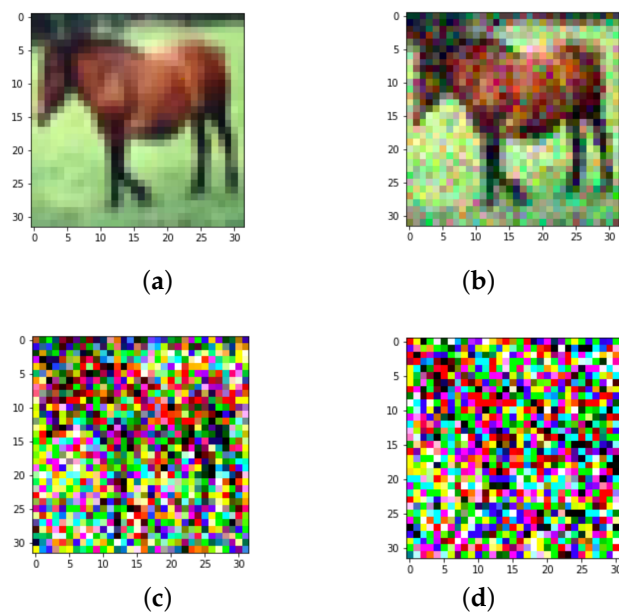
In the next set of experiments, we assessed the robustness of the biologically plausible model by injecting noise from random Gaussian distributions into the MNIST and CIFAR-10 images. Generalization performance of the biologically plausible model on the holdout modified test set was measured and compared with that of the classic NN. Unlike in the previous set of experiments, all available training images were used.

#### Additive Gaussian Noise

MNIST and CIFAR-10 images were modified by adding noise from three different Gaussian distributions, namely with: (i) zero mean and unity standard deviation; (ii) zero mean and the standard deviation of 0.5; (iii) zero mean and the standard deviation of 0.1. Examples of thus corrupted images are shown in Figures 5 and 6.



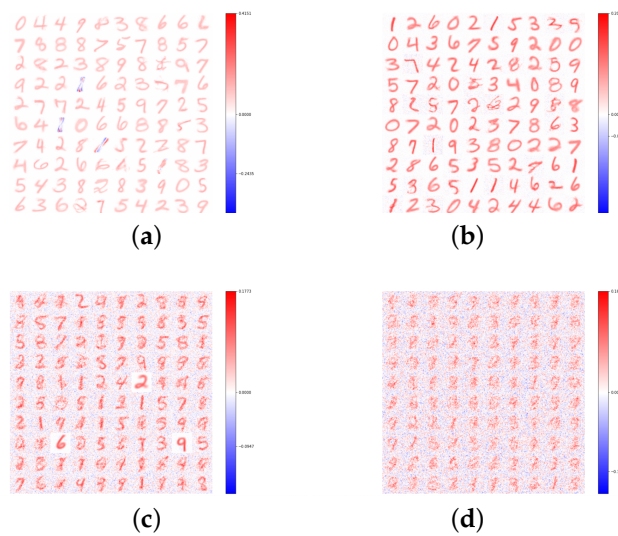
**Figure 5.** Examples of MNIST images corrupted by various amounts of additive Gaussian noise as used in our experiments. (a) Original, no noise. (b) Gaussian noise, zero mean, 0.1 STD. (c) Gaussian noise, zero mean, 0.5 STD. (d) Gaussian noise, zero mean, 1.0 STD.



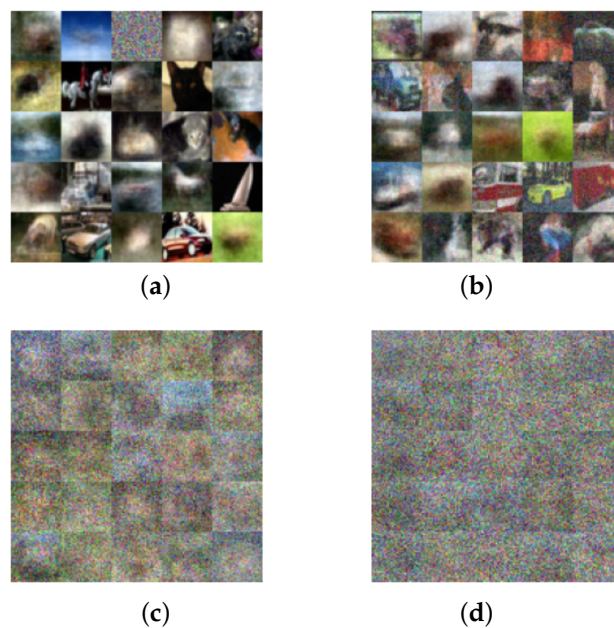
**Figure 6.** Examples of CIFAR images corrupted by various amounts of additive Gaussian noise as used in our experiments. (a) Original, no noise. (b) Gaussian noise, zero mean, 0.1 STD. (c) Gaussian noise, zero mean, 0.5 STD. (d) Gaussian noise, zero mean, 1.0 STD.

### Feature Detectors

The feature detectors learned by the unsupervised component of the biologically plausible model reflect the noise in the input data. The learned feature detectors for this experiment become increasingly speckled and uninterpretable (to the human eye) as the noise in the data increases. These noisy feature detectors learned for both MNIST and CIFAR-10 classification are displayed in Figures 7 and 8.



**Figure 7.** Shown are a hundred randomly selected feature detectors out of 2000 learned by the biologically plausible model for MNIST classification with different levels of added noise. (a) Original, no noise. (b) Gaussian noise, zero mean, 0.1 STD. (c) Gaussian noise, zero mean, 0.5 STD. (d) Gaussian noise, zero mean, 1.0 STD.



**Figure 8.** Shown are 25 randomly selected feature detectors out of 2000 learned by the biologically plausible model for CIFAR-10 classification with different levels of added noise. (a) Original, no noise. (b) Gaussian noise, zero mean, 0.1 STD. (c) Gaussian noise, zero mean, 0.5 STD. (d) Gaussian noise, zero mean, 1.0 STD.

**Robustness to Additive Gaussian Noise**

The generalization accuracy for the biologically plausible model degrades faster than that of the classic neural network model when any level of noise is introduced. These results suggest that the biologically plausible model is more susceptible to random permutations and corruption of incoming signals than the classic neural network model, and that the biologically plausible model may be fooled by small changes or perturbations in the input images. These results are summarized in Tables 4 and 5. The underlying cause may lie in the increasingly hazy and speckled feature detectors learned by the biologically plausible



model under increasing amounts of noise as seen in Figures 7 and 8. These learnt feature detectors may prove to be ineffective when used by the higher layers of the network.

**Table 4.** Generalization accuracy (%) on holdout test sets for MNIST.

ine	Classic NN	Proposed Model
ine No noise	98.52	97.39
Noise (zero mean = 0, 0.1 STD)	98.52	96.87
Noise (zero mean, 0.5 STD)	96.60	92.86
Noise (zero mean, 1 STD)	87.74	73.13
ine		

**Table 5.** Generalization accuracy (%) on holdout test sets for CIFAR-10.

ine	Classic NN	Proposed Model
ine No noise	55.15	48.62
Noise (zero mean = 0, 0.1 STD)	54.84	48.18
Noise (zero mean, 0.5 STD)	51.18	35.70
Noise (zero mean, 1 STD)	43.50	21.07
ine		

## 6. Conclusions and Future Work

Motivated by the well-known shortcomings of the conventional end-to-end backpropagation-based training of artificial neural networks—namely, the high computational demand thereof, both in terms of compute power and memory requirements, and the need for vast amounts of training data—in this paper we described an alternative, biologically plausible (to use the common descriptor, though in the authors’ opinion, “biologically inspired” would be more appropriate) training methodology. Not merely to compare them in the end goals per se, such as classification accuracy (or indeed performance according to any other applicable metric), but also to gather further insight into the advantages and disadvantages of the two approaches and to elucidate the most promising directions for future work, we performed a systemic empirical comparison of the paradigms on two widely used public datasets, CIFAR and MNIST. Amongst the key findings of interest are, expectedly, the better robustness of biologically plausible learning in the presence of data scarcity, and its worse resilience to noise (or, to be precise, additive, zero mean Gaussian noise).

**Author Contributions:** Conceptualization, A.A. and O.A.; methodology, A.A. and O.A.; software, A.A.; validation, A.A.; formal analysis, A.A.; investigation, A.A.; resources, O.A.; data curation, A.A. and O.A.; writing—original draft preparation, A.A.; writing—review and editing, A.A. and O.A.; visualization, A.A.; supervision, O.A.; project administration, O.A. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** All data used is already publicly available.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

- Vinyals, O.; Babuschkin, I.; Czarnecki, W.M.; Mathieu, M.; Dudzik, A.; Chung, J.; Choi, D.H.; Powell, R.; Ewalds, T.; Georgiev, P.; et al. Grandmaster level in StarCraft II using multi-agent reinforcement learning. *Nature* **2019**, *575*, 350–354. [[CrossRef](#)] [[PubMed](#)]
- Silver, D.; Schrittwieser, J.; Simonyan, K.; Antonoglou, I.; Huang, A.; Guez, A.; Hubert, T.; Baker, L.; Lai, M.; Bolton, A.; et al. Mastering the game of go without human knowledge. *Nature* **2017**, *550*, 354–359. [[CrossRef](#)] [[PubMed](#)]

3. Khalil, R.A.; Jones, E.; Babar, M.I.; Jan, T.; Zafar, M.H.; Alhussain, T. Speech emotion recognition using deep learning techniques: A review. *IEEE Access* **2019**, *7*, 117327–117345. [[CrossRef](#)]
4. Chen, H.; Chen, A.; Xu, L.; Xie, H.; Qiao, H.; Lin, Q.; Cai, K. A deep learning CNN architecture applied in smart near-infrared analysis of water pollution for agricultural irrigation resources. *Agric. Water Manag.* **2020**, *240*, 106303. [[CrossRef](#)]
5. Li, Y.; Ma, L.; Zhong, Z.; Liu, F.; Chapman, M.A.; Cao, D.; Li, J. Deep learning for lidar point clouds in autonomous driving: A review. *IEEE Trans. Neural Netw. Learn. Syst.* **2020**, *32*, 3412–3432. [[CrossRef](#)]
6. Caie, P.D.; Dimitriou, N.; Arandjelović, O. Precision medicine in digital pathology via image analysis and machine learning. In *Artificial Intelligence and Deep Learning in Pathology*; Elsevier: Amsterdam, The Netherlands, 2021; pp. 149–173.
7. Rojas, R. The backpropagation algorithm. In *Neural Networks*; Springer: Berlin/Heidelberg, Germany, 1996; pp. 149–182.
8. Li, J.; Wu, Y.; Gaur, Y.; Wang, C.; Zhao, R.; Liu, S. On the comparison of popular end-to-end models for large scale speech recognition. *arXiv* **2020**, arXiv:2005.14327.
9. Li, L.; Gong, B. End-to-end video captioning with multitask reinforcement learning. In Proceedings of the 2019 IEEE Winter Conference on Applications of Computer Vision (WACV), Waikoloa Village, HI, USA, 7–11 January 2019; pp. 339–348.
10. Szegedy, C.; Zaremba, W.; Sutskever, I.; Bruna, J.; Erhan, D.; Goodfellow, I.; Fergus, R. Intriguing properties of neural networks. *arXiv* **2013**, arXiv:1312.6199.
11. Carlini, N.; Wagner, D. Adversarial examples are not easily detected: Bypassing ten detection methods. In Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security, Dallas, TX, USA, 3 November 2017; pp. 3–14.
12. Carlini, N.; Wagner, D. Defensive distillation is not robust to adversarial examples. *arXiv* **2016**, arXiv:1607.04311.
13. Werbos, P.J. *The Roots of Backpropagation: From Ordered Derivatives to Neural Networks and Political Forecasting*; John Wiley & Sons: Hoboken, NJ, USA, 1994; Volume 1.
14. Werbos, P. New Tools for Prediction and Analysis in the Behavioral Sciences. Ph.D. Dissertation, Harvard University, Cambridge, MA, USA, 1974.
15. Hendler, J. Avoiding another AI winter. *IEEE Intell. Syst.* **2008**, *23*, 2–4. [[CrossRef](#)]
16. Rumelhart, D.E.; Hinton, G.E.; Williams, R.J. Learning representations by back-propagating errors. *Nature* **1986**, *323*, 533–536. [[CrossRef](#)]
17. Minsky, M.; Papert, S. *Perceptrons: An Introduction to Computational Geometry*; The MIT Press: Cambridge, MA, USA, 1969.
18. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [[CrossRef](#)] [[PubMed](#)]
19. Silver, D.; Huang, A.; Maddison, C.J.; Guez, A.; Sifre, L.; Van Den Driessche, G.; Schrittwieser, J.; Antonoglou, I.; Panneershelvam, V.; Lanctot, M.; et al. Mastering the game of Go with deep neural networks and tree search. *Nature* **2016**, *529*, 484–489. [[CrossRef](#)] [[PubMed](#)]
20. Hebb, D.O. *The Organization of Behavior: A Neuropsychological Theory*; Psychology Press: New York, NY, USA, 2005.
21. Whittington, J.C.; Bogacz, R. Theories of error back-propagation in the brain. *Trends Cogn. Sci.* **2019**, *23*, 235–250. [[CrossRef](#)] [[PubMed](#)]
22. Krotov, D.; Hopfield, J.J. Unsupervised learning by competing hidden units. *Proc. Natl. Acad. Sci. USA* **2019**, *116*, 7723–7731. [[CrossRef](#)]
23. Tavanaei, A.; Ghodrati, M.; Kheradpisheh, S.R.; Masquelier, T.; Maida, A. Deep learning in spiking neural networks. *Neural Netw.* **2019**, *111*, 47–63. [[CrossRef](#)]
24. Nøklund, A. Direct feedback alignment provides learning in deep neural networks. In Proceedings of the Advances in Neural Information Processing Systems, Barcelona, Spain, 5–10 December 2016; Volume 29.
25. Ravichandran, N.B.; Lansner, A.; Herman, P. Brain-like approaches to unsupervised learning of hidden representations—a comparative study. In Proceedings of the International Conference on Artificial Neural Networks, Bratislava, Slovakia, 14–17 September 2021; Springer: Berlin/Heidelberg, Germany, 2021; pp. 162–173.
26. Bartunov, S.; Santoro, A.; Richards, B.; Marris, L.; Hinton, G.E.; Lillicrap, T. Assessing the scalability of biologically-motivated deep learning algorithms and architectures. In Proceedings of the Advances in Neural Information Processing Systems, Montréal, QC, Canada, 3–8 December 2018; Volume 31.
27. Lillicrap, T.P.; Cownden, D.; Tweed, D.B.; Akerman, C.J. Random synaptic feedback weights support error backpropagation for deep learning. *Nat. Commun.* **2016**, *7*, 13276. [[CrossRef](#)]
28. Lee, D.H.; Zhang, S.; Fischer, A.; Bengio, Y. Difference target propagation. In Proceedings of the Joint European Conference on Machine Learning and Knowledge Discovery in Databases, Porto, Portugal, 7–11 September 2015; pp. 498–515.
29. Illing, B.; Gerstner, W.; Brea, J. Biologically plausible deep learning—But how far can we go with shallow networks? *Neural Netw.* **2019**, *118*, 90–101. [[CrossRef](#)]
30. Chen, X.; Fan, H.; Girshick, R.; He, K. Improved baselines with momentum contrastive learning. *arXiv* **2020**, arXiv:2003.04297.
31. Zhao, Q.; Adeli, E.; Pfefferbaum, A.; Sullivan, E.V.; Pohl, K.M. Confounder-aware visualization of convnets. In Proceedings of the International Workshop on Machine Learning In Medical Imaging, Shenzhen, China, 13 October 2019; pp. 328–336.
32. Xia, Y.; Zhang, J.; Jiang, T.; Gong, Z.; Yao, W.; Feng, L. HatchEnsemble: An efficient and practical uncertainty quantification method for deep neural networks. *Complex Intell. Syst.* **2021**, *7*, 2855–2869. [[CrossRef](#)]
33. Lampinen, J.; Vehtari, A. Bayesian approach for neural networks—Review and case studies. *Neural Netw.* **2001**, *14*, 257–274. [[CrossRef](#)]

34. Cooper, J.; Arandjelović, O.; Harrison, D.J. Believe the HiPe: Hierarchical perturbation for fast, robust, and model-agnostic saliency mapping. *Pattern Recognit.* **2022**, *129*, 108743. [[CrossRef](#)]
35. Dimitriou, N.; Arandjelovic, O. Magnifying Networks for Images with Billions of Pixels. *arXiv* **2021**, arXiv:2112.06121.
36. Grinberg, L.; Hopfield, J.; Krotov, D. Local unsupervised learning for image analysis. *arXiv* **2019**, arXiv:1908.08993.
37. Bengio, Y.; Goodfellow, I.; Courville, A. *Deep Learning*; MIT Press: Cambridge, MA, USA, 2017; Volume 1.
38. Rumelhart, D.E.; Zipser, D. Feature discovery by competitive learning. *Cogn. Sci.* **1985**, *9*, 75–112. [[CrossRef](#)]
39. Hendrycks, D.; Dietterich, T. Benchmarking neural network robustness to common corruptions and perturbations. *arXiv* **2019**, arXiv:1903.12261.
40. Recht, B.; Roelofs, R.; Schmidt, L.; Shankar, V. Do CIFAR-10 classifiers generalize to CIFAR-10? *arXiv* **2018**, arXiv:1806.00451.
41. Azulay, A.; Weiss, Y. Why do deep convolutional networks generalize so poorly to small image transformations? *arXiv* **2018**, arXiv:1805.12177.
42. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Commun. Acm* **2017**, *60*, 84–90. [[CrossRef](#)]
43. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
44. Xie, S.; Girshick, R.; Dollár, P.; Tu, Z.; He, K. Aggregated residual transformations for deep neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1492–1500.
45. Hendrycks, D.; Mazeika, M.; Dietterich, T. Deep anomaly detection with outlier exposure. *arXiv* **2018**, arXiv:1812.04606.
46. Hendrycks, D.; Gimpel, K. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *arXiv* **2016**, arXiv:1610.02136.
47. Liu, S.; Garrepalli, R.; Dietterich, T.; Fern, A.; Hendrycks, D. Open category detection with PAC guarantees. In Proceedings of the International Conference on Machine Learning (PMLR), Stockholm, Sweden, 10–15 July 2018; pp. 3169–3178.
48. Steinhardt, J.; Koh, P.W.W.; Liang, P.S. Certified defenses for data poisoning attacks. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; Volume 30.
49. Hendrycks, D.; Mazeika, M.; Wilson, D.; Gimpel, K. Using trusted data to train deep networks on labels corrupted by severe noise. In Proceedings of the Advances in Neural Information Processing Systems, Montréal, QC, Canada, 3–8 December 2018; Volume 31.
50. Löwel, S.; Singer, W. Selection of intrinsic horizontal connections in the visual cortex by correlated neuronal activity. *Science* **1992**, *255*, 209–212. [[CrossRef](#)] [[PubMed](#)]
51. Oja, E. Simplified neuron model as a principal component analyzer. *J. Math. Biol.* **1982**, *15*, 267–273. [[CrossRef](#)] [[PubMed](#)]
52. Dimitriou, N.; Arandjelovic, O. A new look at ghost normalization. *arXiv* **2020**, arXiv:2007.08554.
53. Linsker, R. From basic network principles to neural architecture: Emergence of spatial-opponent cells. *Proc. Natl. Acad. Sci. USA* **1986**, *83*, 7508–7512. [[CrossRef](#)] [[PubMed](#)]
54. Pehlevan, C.; Hu, T.; Chklovskii, D.B. A Hebbian/anti-Hebbian neural network for linear subspace learning: A derivation from multidimensional scaling of streaming data. *Neural Comput.* **2015**, *27*, 1461–1495. [[CrossRef](#)] [[PubMed](#)]
55. Von der Malsburg, C. Self-organization of orientation sensitive cells in the striate cortex. *Kybernetik* **1973**, *14*, 85–100. [[CrossRef](#)]
56. Seung, H.S.; Zung, J. A correlation game for unsupervised learning yields computational interpretations of Hebbian excitation, anti-Hebbian inhibition, and synapse elimination. *arXiv* **2017**, arXiv:1704.00646.
57. Chakravarthy, A. Visualizing Intermediate Activations of a CNN Trained on the MNIST Dataset. 2019. Available online: <https://towardsdatascience.com/visualizing-intermediate-activations-of-a-cnn-trained-on-the-mnist-dataset-2c34426416c8> (accessed on 29 October 2022).
58. Bottou, L. Stochastic gradient descent tricks. In *Neural Networks: Tricks of the Trade*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 421–436.