INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
Online ISSN 1841-9844, ISSN-L 1841-9836, Volume: 17, Issue: 1, Month: February, Year: 2022
Article Number: 4650, https://doi.org/10.15837/ijccc.2022.1.4650

CCC Publications

AGORA UNIVERSITY PRESS

Information Bottleneck in Deep Learning - A Semiotic Approach

B.Muşat, R. Andonie

Bogdan Muşat
Department of Electrical Engineering and Computer Science
Transilvania University of Braşov
str. Politehnicii nr. 1, Braşov, Romania
bogdan_musat_adrian@yahoo.com

Răzvan Andonie
Department of Computer Science
Central Washington University, USA
400 East University Way, Ellensburg, WA 98926, USA
and
Transilvania University of Braşov
str. Politehnicii nr. 1, Braşov, Romania
andonie@cwu.edu

Abstract

The information bottleneck principle was recently proposed as a theory meant to explain some of the training dynamics of deep neural architectures. Via information plane analysis, patterns start to emerge in this framework, where two phases can be distinguished: fitting and compression. We take a step further and study the behaviour of the spatial entropy characterizing the layers of convolutional neural networks (CNNs), in relation to the information bottleneck theory. We observe pattern formations which resemble the information bottleneck fitting and compression phases. From the perspective of semiotics, also known as the study of signs and sign-using behavior, the saliency maps of CNN's layers exhibit aggregations: signs are aggregated into supersigns and this process is called semiotic superization. Superization can be characterized by a decrease of entropy and interpreted as information concentration. We discuss the information bottleneck principle from the perspective of semiotic superization and discover very interesting analogies related to the informational adaptation of the model. In a practical application, we introduce a modification of the CNN training process: we progressively freeze the layers with small entropy variation of their saliency map representation. Such layers can be stopped earlier from training without a significant impact on the performance (the accuracy) of the network, connecting the entropy evolution through time with the training dynamics of a network.

Keywords: deep learning, information bottleneck, semiotics

1 Introduction

Modern deep neural networks (DNNs) are computational machines capable of representing very complex functions which can solve a suite of extremely difficult tasks ranging from computer vision [37], [36] to natural language processing [30], [35] and robotics [38], [39]. Although possessing a high expressive power, model interpretability has always been a limiting factor for use cases requiring explanations of the features involved in modelling. The field of interpretability/explainability in deep learning has witnessed an explosion of published papers in recent years. Even if there is no fundamental theory that can elucidate all underlying mechanisms present in those networks, multiple works tried to deal with this issue by coming up with partial solutions, either by visual explanations [22], [18] or theoretical insights [14], [9], [19]. Therefore, we can say that the black box interpretation of deep learning is not true anymore, and what we need are better techniques to interpret these models. In the following, we will refer to three methods used for the interpretation of deep learning.

Saliency Maps. Saliency maps are arguably the oldest and most frequently used explanation method for interpreting the predictions of DNNs. These maps are a class of computer vision techniques used to investigate hidden layers of neural networks by generating heat maps to depict the most important or salient areas. The popularity of saliency maps comes from the fact that heatmaps can be easily visualized and visualization plays an essential role in humans' cognitive process [43]. Practically, a saliency map can be built using gradients of the output over the input, and this highlights the areas of the images which were relevant for the specific task (e.g., classification). The concept goes back to the work of Kadir and Brady [21]. Some of the early studies on saliency maps applied to DNNs was the work of Zeiler and Fergus [22], who used an auxiliary network (called deconvnet) to revert the activations of an intermediate layer from a DNN back to the input pixels and visualize the patches that excite most the top 9 neurons with largest activation values. Since then, other methods were proposed to better understand the inner workings of neural networks [18, 23, 24, 25]. In our study, we use the popular Grad-CAM method [18] to generate saliency maps of neural layers in CNN architectures.

Semiotic Superization. We proposed recently [19] an interdisciplinary method for deep learning interpretations. It is an information-theoretical approach combining concepts from semiotics. From the perspective of semiotics, also known as the study of signs and sign-using behavior, the saliency maps of CNN's layers exhibit aggregations: signs are aggregated into supersigns and this process is called *semiotic superization*. Superization can be characterized by a decrease of entropy and interpreted as information concentration. To measure the information concentration in CNNs, we used the spatial aura matrix entropy [20] of saliency maps of neural layers. A saliency map aggregates information from the previous layer and this information is measured by the spatial entropy of the map. Generally, the entropy decreases when progressing through the depth of the network, but this is not always happening.

Information Bottleneck. Another recent information-theoretical tool used for deep learning interpretations is the information bottleneck (IB) principle, introduced by Tishby, Pereira, and Bialek [13]. In the context of DNNs, the core assumption of this principle is that a good internal representation produced by a neural model should maximally compress the input data, while preserving sufficient information about the output. Based on the IB theory, the authors of [9] popularized the analysis of the information plane (IP), in which estimates of the two mutual information quantities $I(X;T)$ and $I(Y;T)$ are the coordinate axes. Two distinct phases, *fitting* and *compression*, characterize the mutual information of: *a)* the input X and the internal representation T; and *b)* the internal representation T and the output Y [9]. Several recent works attempted to understand DNNs using the IB principle [33, 34].

The main motivation for our work is to test the IB hypothesis on a variety of new situations, considering that there are contradictory opinions and results about the IB theory (see, for instance, [8]). Our thesis it that there is a significant similarity between the IB principle and semiotic superization.

To the extend of our knowledge, this synergetic aspect was never discussed before.

We investigate the IB theory in the context of semiotic superization processes in CNN learning. In practical terms, we study the evolution of spatial entropy of CNN saliency maps to validate/invalidate the IB principle. In our experiments, we noticed a pattern similar to the fitting and compression phases appearing in the evolution of spatial entropy. We use these experimental results to train a network by freezing the redundant layers with low spatial entropy variability. This enables us to discover interesting analogies between the IB theory and semiotic superization.

Our contributions are twofold: we establish a link between the IB hypothesis of fitting and compression, and semiotic superization via the evolution of spatial entropy applied to saliency maps. As a practical application, we design a heuristic training strategy for layer-wise early stopping based on spatial entropy variability through time, which may be used to prevent overfitting during learning.

The rest of the paper is structured as follows. Section 2 reviews previous results related to applications of the IB principle in deep learning. Section 3 introduces the mathematical framework used to compute the spatial entropy and mutual information in CNNs. We also review here the fundamental concepts of semiotic superization. Section 4 presents our thesis about the relationships between semiotic information adaptation and the IB principle. Section 5 analyses experimentally the patterns present in the evolution of spatial entropy for saliency maps through time. It shows that there is a surprising connection between the IB theory of fitting-compression and the evolution of spatial entropy applied to the saliency maps of neural layers. As an application, we introduce a layer-wise early stopping criterion based on the spatial entropy and also discuss the relevance of our experiments for a semiotic interpretation of the IB principle. Finally, Section 6 presents conclusions and future research directions.

2 Related Work: The Information Bottleneck Principle for Neural Networks

Our paper focuses on the application of the IB principle to neural networks. This section reviews published results which may be connected to our work. An overview of IB with applications in machine learning can be found, for instance, in [4].

The original formulation of the IB concept was elaborated in [13] as an information theoretical technique whose purpose is to find the best tradeoff between prediction accuracy of a variable Y and compression of the input random variable X in the code T. This is realized by the minimization of the following Lagrangian [13]:

$$\min_{P_{T|X}} I(X;T) - \beta I(Y;T) \tag{1}$$

where $I(\cdot,\cdot)$ is the mutual information of two random variables and β is a trade-off parameter.

Recently, the IB principle was applied to deep learning and presented by Tishby and Zaslavsky [14], as a theoretical concept meant to offer a possible explanation for the underlying mechanisms that govern modern deep learning architectures. The mechanism behind is similar with the one from the original formulation, optimizing for a latent representation T that represents a minimum sufficient statistic for an input X, by compressing any redundant information about it, while preserving the needed information to predict label Y. This is done in the same way as in Equation 1. Their work also proposes theoretical bounds on the generalization capability of a neural network. It is argued that good generalization is caused by good compression of the input X in the latent variable T. The IB principle suggests that deeper layers correspond to smaller mutual information values, providing increasingly compressed statistics [4]. It is important to notice that the authors do not provide any training experiments in which they use the IB formulation.

One of the first practical successes of the IB applied to a real problem was by Alemi *et al.* [1] who used a variational approximation technique, termed as Variational Information Bottleneck (VIB), to estimate equation 1. It is known that this equation is intractable, unless X, Y and T are all discrete or jointly Gaussian. However, these assumptions are unrealistic in practice. By using this variational approximation, Alemi *et al.* trained an MLP with two hidden layers on the MNIST dataset [6] using

VIB as a form of regularization. They demonstrated that VIB acts as a better regularizer than other popular methods like Dropout [10] or Label Smoothing [7]. They also tackled upon the notion of adversarial robustness and how the VIB framework fits into this. Adversarial robustness is considered as the resilience of a DNN in front of adversarial attacks [5, 12] - small and imperceptible to human eye modifications on input pixels that can easily fool a neural network. They showed that a network trained with the VIB regularizer and large enough β can resist much better than ordinary trained networks to such adversarial attacks.

Shwartz-Ziv and Tishby [9] viewed the layers of a DNN as a Markov chain of successive internal representations of the input X. Any latent representation T is defined through the use of an encoder $P(T|X)$ and a decoder $P(\hat{Y}|T)$, where $\hat{Y}$ is the neural prediction. They defined the notion of information plane (IP) as the coordinate plane of the mutual information quantities $I_X = I(X;T)$ and $I_Y = I(T;Y)$ during many training epochs. For a multi-layer perceptron (MLP) with a few layers, trained on a synthetic data problem, they noticed two important phases during training: a fitting phase, where $I(X;T)$ and $I(T;Y)$ both increase, and a compression phase, where the mutual information $I(X;T)$ starts decreasing, while $I(T;Y)$ stays mostly constant. They associated the decrease of $I(X;T)$ with compression of input X in the latent T, which avoids overfitting, thus explaining the good generalization achieved by overparametrized DNNs.

Saxe *et al.* [8] criticized the IP hypothesis of Shwartz-Ziv and Tishby, arguing that it is not applicable to general DNNs. They argued that the two phases observed in [9] are caused by the double-sided saturating nature of the *tanh* activation function used in their MLP, and binning of continuous activations to discrete values. The two-phase behaviour, as they empirically proved, is not present in networks which use non-saturating activation functions (like ReLU), employed by most modern DNNs. They also tested the supposed link between compression and generalization using the network from [9], but trained it on a smaller percentage of the data, showing that even if the compression phase is noticeable in the IP, the train vs test accuracy suffers from severe overfitting.

Rana Ali Amjad *et al.* [2] also noticed issues with the mutual information quantity $I(X;T)$ in the Lagrangian optimization equation: for any continuous variable $X \in \mathbb{R}^N$, the term will always be infinite. This case will be true for most deterministic DNNs. Shwartz-Ziv and Tishby [9] solved this issue by quantizing the output activations, but this procedure failed to capture the true mutual information value as it encoded nontrivial information about the feature maps. Another proposed solution was to use a stochastic system by injecting Gaussian noise ϵ into the intermediate representations T, and minimizing the following surrogate optimization criterion [2]:

$$\min_{P_{T|X}} \tilde{I}(X;T) - \beta \tilde{I}(Y;T) \tag{2}$$

where $\tilde{I}(X;T) = I(X;T+\epsilon)$ and $\tilde{I}(Y;T) = I(Y;T+\epsilon)$.

Indeed, [34] uses this idea to inject intrinsic noise to the network during training. When ϵ is relatively small ($\approx 10^{-2}$), they showed that the network performs similarly to deterministic ones and that the representations learned are closely related. Furthermore, the binning strategy can now accurately estimate the term $\tilde{I}(X;T)$, when the standard deviation σ of the injected noise is of the order of the quantization cell length. They measured the mutual information of $\tilde{I}(X;T)$ using a rate-optimal estimator based on Monte Carlo integration, and proved for the synthetic experiment from [14] that a small noisy DNN undergoes a long-term compression phase. The experiments also uncovered that compression in noisy DNNs can be attributed to clustering of internal representations, with clusters comprising mostly samples from the same class.

Wickstrøm *et al.* [15] conducted the first large scale experiment using the IB principle, studying the VGG16 architecture [28] trained on CIFAR-10 [16]. They proposed a matrix-based Rényi's entropy coupled with tensor kernels over convolutional layers to estimate the intractable mutual information, in order to analyze the IP. Using this method, there is no need anymore for binning operations. One of their observation was that compression appears mostly on the training data, and is less visible in the test dataset. Going forward, they used an early stopping criterion based on a patience parameter. The training stops if the validation accuracy does not change after a predefined number of epochs. They noticed that training can be sometimes stopped even before the compression phase starts. The assumption here is that compression is linked to overfitting.

Another practical application of the IB principle was studied by Elad *et al.* [3]. They used the IB objective as a training criterion by maximizing the value of the Lagrangian, layer by layer. They trained one layer at a time, while freezing the previously learned layers, and added a linear classification layer at the end. While this kind of training strategy is prohibitive in terms of computational time for a modern DNN, they managed to reach on par results with the end-to-end counterpart trained with the cross entropy loss on the MNIST and CIFAR-10 datasets, using a three layered MLP. This is the first experimental illustration of the IB principle used as a training criterion instead of the well-established cross-entropy.

Other comprehensive reviews on the applications of the IB theory can be found in [4, 17]. Some of the conclusions drawn from these reviews are that IB needs further exploration. The compression seen in IPs does not necessarily represent learning a minimum sufficient statistic, nor that it produces good generalization. Yet, it can provide a good geometrical explanation for some of the inherent behaviour underlying DNNs, and might even open the doors for deeper theoretical understandings.

From a practical point of view, the most similar work to ours is [15]. The authors used an early stopping criterion based on a patient parameter applied to the validation accuracy and noticed that training is stopped before the compression phase begins, concluding that compression is not necessarily useful for preventing overfitting, as suggested in [9]. Different from [15], we embed in our training phase a layer-wise heuristic technique to progressively stop layers from training by observing the variability of the spatial entropy of the saliency maps. We employ the value of the entropy to stop the training on a local level, while in [15] the training is stopped at a global level based on a more traditional criterion.

3 Background: Spatial Entropy and Superization in CNNs

Our approach is based on the spatial aura matrix entropy of saliency maps calculated for different neural layers. We measure the information concentration in CNNs using the spatial entropy of saliency maps, and relate the decrease of entropy to semiotic superization. To make the paper self-contained, this section summarizes the basic techniques used to compute the spatial entropy and reviews the superization effect in CNN standard operations (pooling and convolution).

3.1 Spatial entropy in saliency maps

Our work analyzes the entropy variations of 2D saliency maps. The entropy is generated by the gradient method in Grad-CAM. We describe in the following how we calculate the entropy. The formulas used are from [19].

Let us define the joint probability of pixels at spatial locations (i, j) and $(i+k,\ j+l)$ to take the value g, respectively g' as:

$$p_{gg'}(k,l) = P(X_{i,\,j} = g, X_{i+k,\,j+l} = g') \tag{3}$$

where g and g' are pixel intensity values $(0-255)$. If we assume that $p_{gg'}$ is independent of $(i,\ j)$ (the homogeneity assumption [40]), we define for each pair $(k,\ l)$ the entropy

$$H(k,\ l) = -\sum_{g}\sum_{g'} p_{gg'}(k,\ l)\log p_{gg'}(k,\ l) \tag{4}$$

where the summations are over the number of outcome values (256 in our case). A standardized relative measure of bivariate entropy is [40]:

$$H_R(k,\ l) = \frac{H(k,\ l) - H(0)}{H(0)} \in [0,\ 1] \tag{5}$$

The maximum entropy $H_R(k,\ l) = 1$ corresponds to the case of two independent variables. $H(0)$ is the univariate entropy, which assumes all pixels as being independent, and we have $H(k,\ l) \geq H(0)$.

Based on the relative entropy for $(k,\ l)$, the Spatial Disorder Entropy (SDE) for an $m \times n$ image $\mathbf{X}$ was defined in [40] as:

$$H_{SDE}(\mathbf{X}) \approx \frac{1}{mn} \sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{k=1}^{m} \sum_{l=1}^{n} H_R(i-k,\ j-l) \tag{6}$$

Since the complexity of SDE computation is high, we decided to use a simplified version - the Aura Matrix Entropy (AME, see [20]), which only considers the second order neighbors from the SDE computation:

$$H_{AME}(\mathbf{X}) \approx \frac{1}{4}\Big(H_R(-1,\ 0) + H_R(0,\ -1) + H_R(1,\ 0) + H_R(0,\ 1)\Big) \tag{7}$$

Putting it all together, starting from a map obtained by the Grad-CAM method, we compute the probabilities $p_{gg'}$ in equation (3), and finally the AME in equation (7), which results in the spatial entropy quantity of a saliency map.

The mutual information between two saliency maps $M1$ and $M2$ can be described by the following formula:

$$I(M1, M2) = H(M1) + H(M2) - H(M1, M2) \tag{8}$$

where $H(\cdot)$ is the (spatial) entropy of a variable, and $H(\cdot,\cdot)$ is the joint (spatial) entropy between two variables. We modify the simplified aura matrix entropy to be applicable for joint entropy calculation by changing Equation 3 to:

$$p_{gg'g''g'''}(k, l) = P(M1_{i,\,j} = g, M1_{i+k,\,j+l} = g', M2_{i,\,j} = g^{''}, M2_{i+k,\,j+l} = g^{'''}) \tag{9}$$

where g, g', $g^{''}$, $g^{'''}$ are pixel intensity values. The upcoming equations from the spatial entropy computation will use $p_{gg'g''g'''}$ instead of $p_{gg'}$. The final modification will be to equation (7), where we take into consideration four spatial positions instead of two, the first two from $M1$ and the last two from $M2$:

$$H_{AME}(\mathbf{X}) \approx \frac{1}{4}\Big(H_R(-1,\ 0, -1,\ 0) + H_R(0,\ -1, 0,\ -1) + H_R(1,\ 0, 1,\ 0) + H_R(0,\ 1, 0,\ 1)\Big) \tag{10}$$

We apply equation (10) to compute the joint spatial entropy between two saliency maps. We have now all the members from equation (8) computed and can calculate the mutual information.

3.2 Semiotic superization in CNNs

Semiotics is the study of signs, symbols, and signification. The fundamental semiotic building block is the triad Sign-Object-Interpretant [44]. Together, the *object* and *interpretant* make up the *sign*, which is the smallest unit of meaning.

In a communication process, signs are agglomerated into more abstract signs called *supersigns*. Iterating the process, we obtain from *k-th level supersigns* $(k+1)$-th level supersigns, and this process is known as *superization* [11, 45, 46, 47, 48]. Frank [47] identified two types of superization: **Type I** - Building equivalence classes and thus reducing the number of signs; and **Type II** - Building compound supersigns from simpler component supersigns.

If we consider the entropy values H_k and H_{k+1} computed at two successive levels of supersigns, the extracted information by the interpretant can be measured by the difference $H_k - H_{k+1}$. We have the following property, proved in [47]: Superization tends to concentrate information by decreasing entropy between successive levels: $H_k - H_{k+1}$ is non-negative. This property holds for both types of superization.

However, this is a mathematical result which omits other interesting aspects. In communication processes, the entropy is not necessarily monotonically decreasing. From an informational psychology perspective, the entropy increases until it reaches its peak value. This phase may be associated to the informational adaptation of the perceiver [11, 45]. The subsequent entropy decrease is related to the processing of structural information and the rate of decrease depends largely upon the amount of structural information. The entropy falls quickly when little structural information is available, whereas when major structural information is present, the entropy will remain high over most of its range [11, 45]. We will discuss informational adaptation later, in Section 4.

It is known [22] that in a CNN complex objects are composed of simpler object parts as the receptive field of the network grows and combines multiple neurons from previous layers. Therefore it is interesting to observe if any form of superization is present in the training process of a CNN. In [19] we applied the above theorem to the neural layers of CNNs. We computed superization with respect to the spatial entropy variations of the saliency maps. Type I superization appears when we reduce the spatial resolution of a layer $k+1$ by subsampling layer k. This is similar to class formation because we reduce the variation of the input values (i.e., we reduce the number of signs). In CNNs, this is typically performed by a pooling (down-sampling) operator. Type II superization is produced when applying a convolutional operator to a neural layer k; as an effect, layer $k+1$ will focus on more complex objects, composed of objects already detected by layer k.

A multi-layered classification can be interpreted as a semiotic process [19]. At the end of a successful recognition process, the entropy of the output layer becomes 0 and no further information needs to be extracted. The last layer (the fully connected layer in a CNN network) is connected to the outer world, the world of objects.

For CNN layers, it may happen that both types of superization operate concurrently. In this case, it becomes difficult to separate their effects. According to our results [19], the first type of superization is more effective for decreasing the entropy, whereas the second superization type is more responsible for building supersigns with semantic roles.

4 Superization and the IB principle

This section presents our thesis on the analogy between information adaptation via superization and the IB principle.

Beside superization, it is also interesting to study another semiotic aspect in a CNN model - informational adaptation. This aspect was never discussed before. In our preliminary experiments, we observed that during training the entropy of each neural layer increases until it reaches its peak value. This phase may be associated with informational adaptation of the model. The subsequent decrease of the entropy is related to the processing of the structural information. The rate of decrease is largely dependent upon the amount of structural information in the input layer. When there is little structural information, the entropy falls quickly, whereas when there are major structural elements, the entropy of the neural layers stays high over most of its range. The manner in which the entropy changes indicate the type of information in the input layer [11]. In our opinion, this information adaptation can be related to the two distinct phases of the IB principle - fitting and compression.

In order to explore the presence of the IB hypothesis in saliency maps, we investigate:

- The evolution of the mutual information during training, between input and intermediate saliency maps, and between intermediate and output saliency maps.
- The evolution of spatial entropy for saliency maps during training.

The IB plane analysis as described in [9] tracked the two mutual information quantities $I(X;T)$ and $I(Y;T)$ and noticed the fitting and compression patterns emerging. As such, we analyze the information planes between $I(X;T)$ and $I(Y;T)$ by computing the mutual information from equation 8 between the first and an intermediate saliency map, and between the last and the same intermediate saliency map. The proposed experiment is meant to uncover any resemblance to the original results from [9], but applied to a different concept like saliency maps.

Going a bit further, we test a possible link between the fitting and compression patterns present in IB theory with the spatial entropy of saliency maps. In [19] the spatial entropy was studied at a single point in time (after training), going along the depth of the network. We now intend to capture the dynamics of the entropy during the whole training to see if it is governed by the same patterns.

As we will see, while we can not draw any conclusions from the first scenario, in the second case there is a visible trend, similar to the fitting-compression phases studied in IPs.

Regarding saliency maps, while the superization process acts in the depth of the network, the IB principle acts on a single layer. In order to connect those two concepts, we verify the dynamics of the spatial entropy of saliency maps through the whole training process in conjuction with its layer-wise behaviour. We uncover some form of continuity pattern, presented in the next section.

5 Experiments

We experimentally discover in this section a connection between the IB theory of fitting-compression and the evolution of spatial entropy applied to saliency maps based on similar forming patterns. We also verify the practical applicability of the spatial entropy patterns and a possible connection with the superization process. For training, we use the deep learning programming framework PyTorch [26] (version 1.6.0) and the public implementation of Grad-CAM, modified to our needs.

5.1 Evolution of mutual information

We start analyzing the information planes between $I(X;T)$ and $I(Y;T)$ by computing the mutual information from equation 8 between the first and an intermediate saliency map, and between the last and the same intermediate saliency map. We plot the resulted values after each training epoch and we look for any visible patterns, similar to the ones observed in [9].

Since the saliency maps obtained by Grad-CAM are discretized 8-bit arrays consisting of integer values in the range $[0, 255]$, it is straightforward to compute the mutual information between two such maps using the equations from the previous section, unlike the quantities in [9] which are almost always continuous and require a binning operation.

We train a standard VGG16 architecture on the CIFAR-10 dataset and then apply the Grad-CAM method to obtain layer-wise saliency maps. We study the behaviour of the mutual information obtained from those saliency maps after each epoch. To obtain a more accurate estimate of the mutual information, we average the mutual information quantities obtained from 50 samples chosen randomly from the training set. In Table 1, there are representative plots for the computed quantities at selected layers. We select only four layers for visualization (VGG16 is comprised of ≈ 30 layers, including ReLUs and MaxPooling). The other layers have a similar behaviour.

It can be observed that for the first few epochs, the plots resemble the ones in [9], with an increase in $I(X;T)$ and $I(Y;T)$. Afterwards, the mutual information does not obey any logical pattern which would indicate any form of compression. Thus, we empirically conclude that the IB concept is not present in the mutual information of saliency maps.

5.2 Evolution of entropy

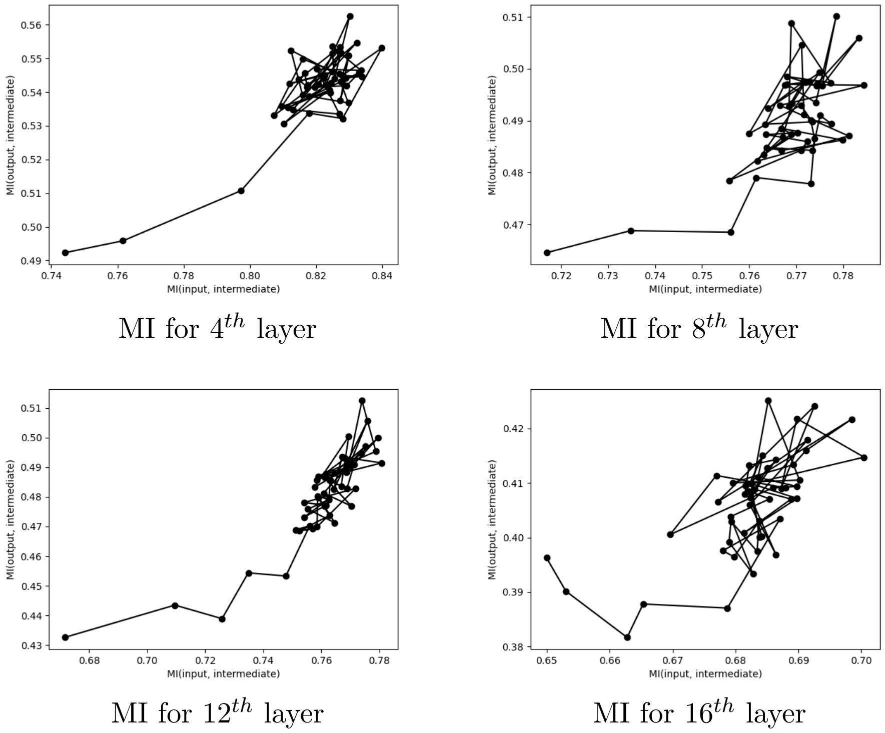
We derive a bit from the study of mutual information and analyze the evolution of spatial entropy for saliency maps through time for the same VGG16 architecture. Whereas in [19] the spatial entropy was studied at a single point in time (after training), going along the depth of the network, we intend to capture now the dynamics of the entropy during the whole training, looking for any patterns.

The spatial entropy was computed using the formulas from Section 3 and averaged over 50 random samples from the training dataset. In Table 2, there are plots for the computed spatial entropy through time for selected layers. We visualize again, only four layers.

A pattern is now visible, where the spatial entropy increases during the initial phase of the training and at some point flattens out. Very interesting, we noticed the same patterns on other well-known network architectures: ResNet [31], DenseNet [42] and GoogleNet [29].

We notice that early layers exhibit a more abrupt increase of the entropy values during the first few epochs. This phenomenon could be attributed to the fact that the first layers of a CNN learn to detect easy concepts like edges, and the network learns to perform this task faster than latter layers which are responsible for detecting more complex concepts, like whole object parts [22].

Table 1: Information planes for the mutual information of saliency maps



MI for 4^{th} layer

MI for 8^{th} layer

MI for 12^{th} layer

MI for 16^{th} layer

In [27] it is stated that early layers are faster to learn by employing a self-supervised pretraining scheme on a single heavily augmented image. The authors prove that a single image is sufficient to learn good representations for the first few layers. In conjunction with [27], we also hypothesize that the abrupt entropy increase, observed for the first layers, is due to easier concepts being learned faster.

There are however some exceptions to the patterns in Table 2, present only for a few layers, like in Figure 1. While we were not able yet to find a good explanation for those different patterns, we make the following supposition. As in [19], where an entire layer was pruned if a drop in spatial entropy would not happen, we assume that we could prune a layer if the spatial entropy does not follow the patterns presented in Table 2 because that layer contains redundant information.

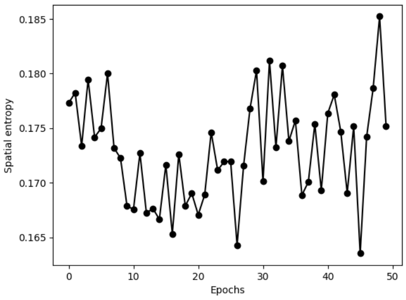
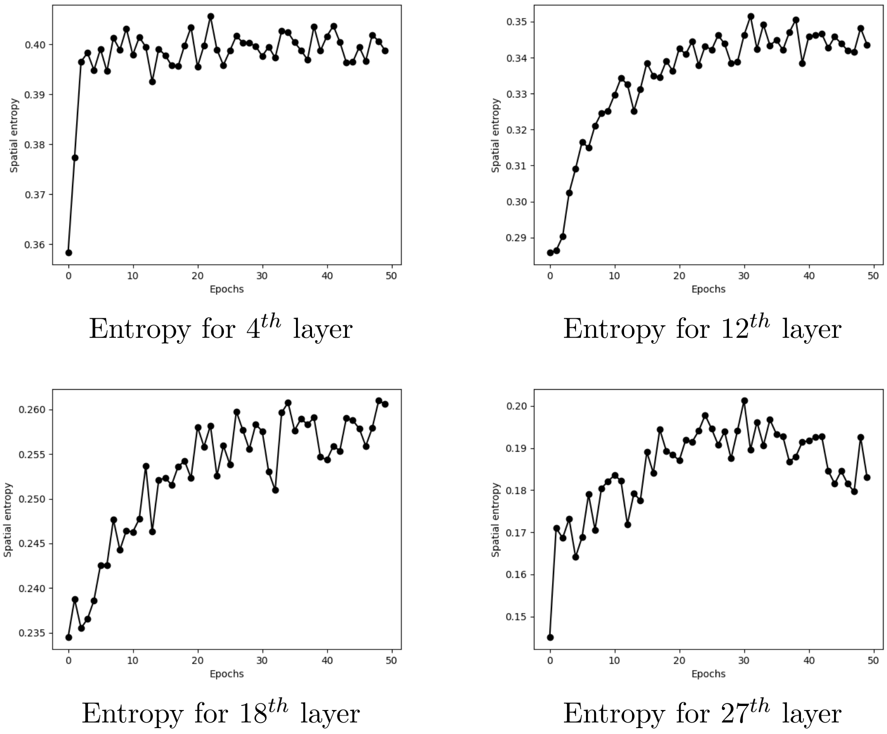


Figure 1: Entropy for the 28^{th} layer

5.3 Freezing layers during training

From the patterns observed in Table 2, we test the hypothesis that there are links between the dynamics of the spatial entropy and the evolution of the training process. As such, we train the same VGG16 on the CIFAR-10 dataset and freeze the layers in which the spatial entropy of the saliency

Table 2: Spatial entropy through time for saliency maps



Entropy for 4^{th} layer

Entropy for 12^{th} layer

Entropy for 18^{th} layer

Entropy for 27^{th} layer

map averaged over the last five epochs enters a compression phase with little variation and is below some threshold ϵ, and observe if it achieves the same accuracy as a fully trained network in the same or less number of epochs.

For a large ϵ, the layers are frozen early in the training and learning becomes prohibitive. For a small ϵ, layers are generally not frozen and the network is trained as usual. In practice, we found an ϵ value of $5e - 05$ to work best. In Table 3, there are empirical results for a fully trained VGG16 vs a VGG16 with some layers frozen during training. The max accuracy column indicates the maximum accuracy achieved on the CIFAR-10 test set by the two versions until the specified epoch in the first column.

Table 3: VGG16 performance - normal vs frozen layers. Experiments performed on a Tesla K80 GPU on Google Colaboratory [41]

Epoch	**Max accuracy**		**Layers frozen**	**Running time (minutes)**	
	Frozen	Normal		Frozen	Normal
30	85.67%	85.42%	0, 17, 28	66	36
40	86.59%	86.13%	0, 7, 17, 26, 28	88	48
50	86.89%	86.81%	0, 7, 14, 17, 26, 28	110	60
60	87.45%	87.45%	0, 7, 14, 17, 26, 28	133	72

As can be seen, the network is trained with some layers frozen, but still performs as good or better than the version with all the layers trained continuously. This training scheme can be considered as a form of early stopping applied at layer level, usually used to prevent a network from overfitting. Hence, we make a connection between the patterns observed in the spatial entropy of saliency maps and the training dynamics of a DNN. We observe that layer 0, which is the first convolutional layer, is among the first ones to be frozen, which empirically proves our assumption from Subsection 5.2

that early layers are the fastest to be learned. The downside of this method is an overhead to the computational running time, but it was not the target of our experiment.

The most similar experiment with ours is in [15], where the authors used the validation accuracy as a proxy to apply the early stopping procedure and notice that the training can be stopped before the compression phase starts. Unlike their work, we use quantities observed directly in the training dynamics of the network, and apply early stopping to prove that it has effect on the validation accuracy as well.

5.4 Discussion: A semiotic interpretation of the IB principle

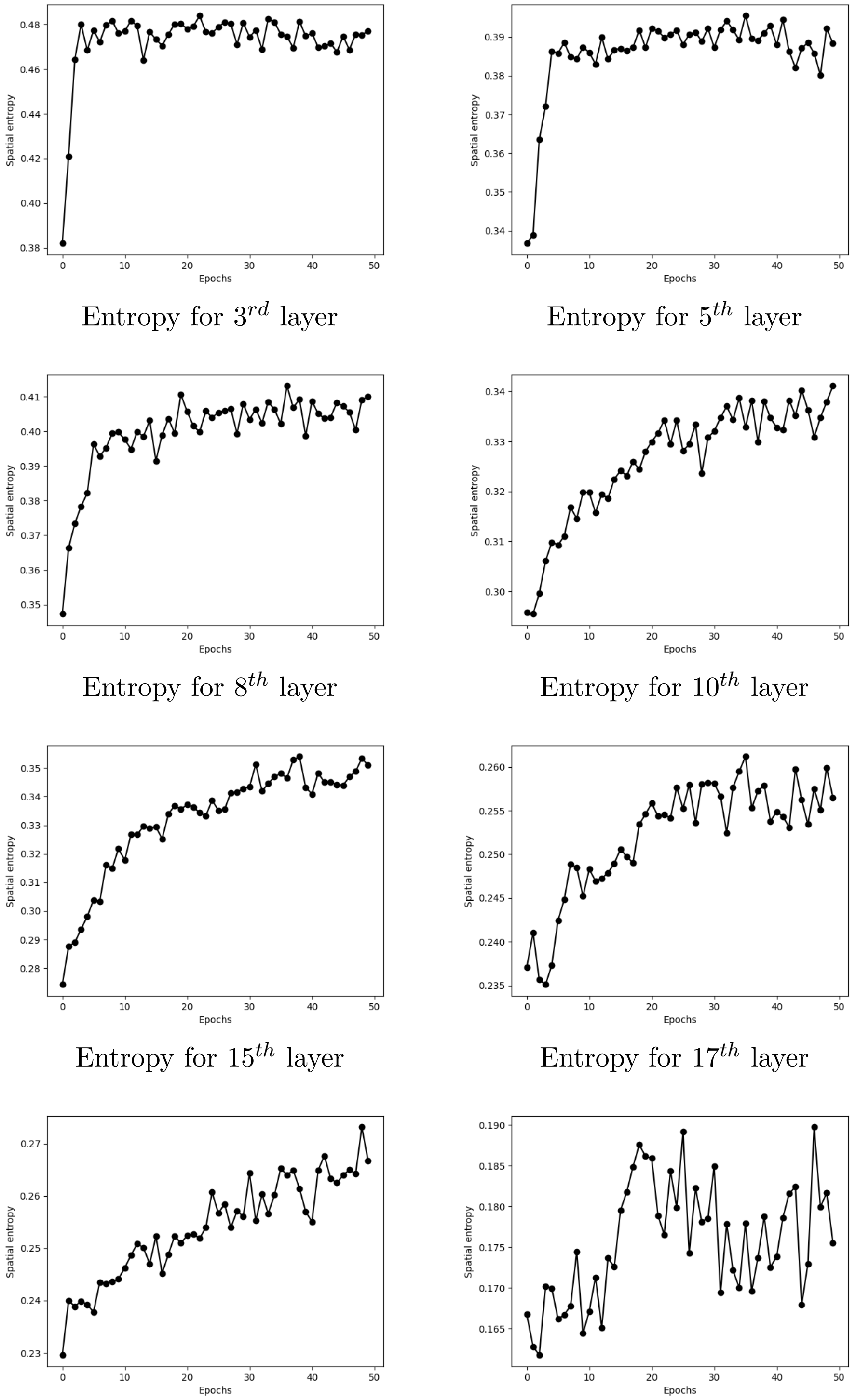
In Table 4, we noticed an interesting property of the spatial entropy for saliency maps. After the superization process takes place (i.e., after a drop of the entropy value), the magnitude resulted after the compression phase is approximately the same with the magnitude of where the entropy starts before superization. We observed a tendency of continuation among layers, directed by evolution of entropy and the superization process. This might represent another inherent property of a DNN's training dynamics: the need to increase the spatial entropy up to an upper bound determined by previous layers through superization.

Driven by these empirical observations, we noticed an interesting connection between IB and superization. From [19] we know that a superization process takes place inside a DNN, which concentrates information, resulting in a decrease of spatial entropy. In order to reach the starting entropy from previous layers, an increase in entropy value is required for latter layers. This increase can be of any form: linear, polynomial, exponential, but as it turns out it follows very closely the same trend of fitting and compression observed in the information bottleneck theory, described in Subsection 5.2. The phenomenon visible in those plots is possible only if there is a mutual dependency between the IB theory (fitting-compression) and superization. This empirical observation might explain some of the training dynamics governing modern DNNs, from an information-theoretical perspective.

There are cases, when superization is not present in the form of spatial entropy decrease (see [19]), but we noticed that the fitting and compression phases are still present. While there are situations when this mutual dependency does not exist, most modern CNNs include some form of subsampling through the max-pooling or strided convolution [28, 29, 31, 32], and the superization process is present nevertheless. Generally, superization and fitting-compression coexist during the training process.

The above semiotic superization process is in principle similar to the IB theory: fitting (entropy increase) followed by compression (entropy decrease). In our model, the spatial entropy of saliency maps measures the information content of neural layers. In our visual experiments, however, the two phases of entropy increase and decrease are separated. The spatial entropy of the neural layers increases during the fitting phase, but we did not notice any entropy decrease after the compression phase starts. The decrease in spatial entropy can be observed only after the superization process takes place, in the form of subsampling, for the subsequent layers.

Table 4: Continuation of spatial entropy for saliency maps after superization



Entropy for 3^{rd} layer

Entropy for 5^{th} layer

Entropy for 8^{th} layer

Entropy for 10^{th} layer

Entropy for 15^{th} layer

Entropy for 17^{th} layer

Entropy for 22^{th} layer

Entropy for 24^{th} layer

6 Conclusions

According to our experiments, there is a connection between the evolution of spatial entropy of saliency maps through time and the IB theory of fitting-compression. We noticed a mutual dependency relation between the IB theory and superization, present in DNNs where there is a drop in spatial entropy magnitude and latter layers reach the same spatial entropy from which former layers start.

We analyzed if the patterns present in the spatial entropy affect the training dynamics of DNNs. We noticed that some layers can be stopped earlier from training, based on the variability of the spatial entropy during the compression phase, and still achieve on par accuracies with fully trained counterparts. This can be regarded as a form of early stopping, applied layer-wise.

To the extent of our knowledge, this is the first application of the IB concept to saliency maps and semiotic superization. Additional experiments are due in order to draw stronger conclusions from the observations described in this work: different DNN architectures, more practical applications of the spatial entropy variability through time, a more robust theoretical understanding of the phenomena described in this work.

References

[1] Alex Alemi, Ian Fischer, Josh Dillon, and Kevin Murphy. Deep variational information bottleneck. In *ICLR*, 2017.

[2] Rana Ali Amjad and Bernhard Geiger. Learning representations for neural network-based classification using the information bottleneck principle. *(submitted to) IEEE Transactions on Pattern Analysis and Machine Intelligence*, PP, 02 2018.

[3] Adar Elad, Doron Haviv, Yochai Blau, and Tomer Michaeli. Direct validation of the information bottleneck principle for deep nets. In *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, pages 758–762, 2019.

[4] Ziv Goldfeld and Yury Polyanskiy. The information bottleneck problem and its applications in machine learning. *CoRR*, abs/2004.14941, 2020.

[5] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples, 2015.

[6] Yann LeCun and Corinna Cortes. MNIST handwritten digit database. 2010.

[7] Gabriel Pereyra, George Tucker, Jan Chorowski, Lukasz Kaiser, and Geoffrey E. Hinton. Regularizing neural networks by penalizing confident output distributions. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Workshop Track Proceedings*. OpenReview.net, 2017.

[8] Andrew Michael Saxe, Yamini Bansal, Joel Dapello, Madhu Advani, Artemy Kolchinsky, Brendan Daniel Tracey, and David Daniel Cox. On the information bottleneck theory of deep learning. In *International Conference on Learning Representations*, 2018.

[9] Ravid Shwartz-Ziv and Naftali Tishby. Opening the black box of deep neural networks via information. *CoRR*, abs/1703.00810, 2017.

[10] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(56):1929–1958, 2014.

[11] Răzvan Andonie, “Semiotic aggregation in computer vision,” *Revue roumaine de linguistique, Cahiers de linguistique théorique et appliquée*, vol. 24, pp. 103–107, 1987.

[12] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian J. Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In Yoshua Bengio and Yann LeCun, editors, *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014.

[13] Naftali Tishby, Fernando C. Pereira, and William Bialek. The information bottleneck method. In *Proc. of the 37-th Annual Allerton Conference on Communication, Control and Computing*, pages 368–377, 1999.

[14] Naftali Tishby and Noga Zaslavsky. Deep learning and the information bottleneck principle. In *2015 IEEE Information Theory Workshop (ITW)*, pages 1–5, 2015.

[15] Kristoffer Wickstrøm, Sigurd Løkse, Michael Kampffmeyer, Shujian Yu, Jose Principe, and Robert Jenssen. Information plane analysis of deep neural networks via matrix-based renyi's entropy and tensor kernels, 2019.

[16] Alex Krizhevsky. Learning multiple layers of features from tiny images. *University of Toronto*, 05 2012.

[17] Bernhard C. Geiger. On information plane analyses of neural network classifiers–a review. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–13, 2021.

[18] R. R. Selvaraju, A. Das, R. Vedantam, M. Cogswell, D. Parikh, and D. Batra, "Grad-CAM: Why did you say that? Visual explanations from deep networks via gradient-based localization," *CoRR*, vol. abs/1610.02391, 2016. [Online]. Available: http://arxiv.org/abs/1610.02391

[19] Bogdan Muşat and Răzvan Andonie. Semiotic aggregation in deep learning. *Entropy*, 22(12), 2020.

[20] E. Volden, G. Giraudon, and M. Berthod, "Modelling image redundancy," in *1995 International Geoscience and Remote Sensing Symposium, IGARSS '95. Quantitative Remote Sensing for Science and Applications*, vol. 3, 1995, pp. 2148–2150.

[21] Timor Kadir and Michael Brady. Saliency, scale and image description. *International Journal of Computer Vision*, 45:83–105, 11 2001.

[22] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," *CoRR*, vol. abs/1311.2901, 2013. [Online]. Available: http://arxiv.org/abs/1311.2901

[23] K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep inside convolutional networks: Visualising image classification models and saliency maps." *CoRR*, vol. abs/1312.6034, 2013. [Online]. Available: http://dblp.uni-trier.de/db/journals/corr/corr1312.html#SimonyanVZ13

[24] D. Smilkov, N. Thorat, B. Kim, F. B. Viégas, and M. Wattenberg, "SmoothGrad: removing noise by adding noise," *CoRR*, vol. abs/1706.03825, 2017. [Online]. Available: http://arxiv.org/abs/1706.03825

[25] A. Mahdi, J. Qin, and G. Crosby, "DeepFeat: A bottom-up and top-down saliency model based on deep features of convolutional neural networks," *IEEE Transactions on Cognitive and Developmental Systems*, vol. 12, no. 1, pp. 54–63, 2020.

[26] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "Pytorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems 32*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, Eds. Curran Associates, Inc., 2019, pp. 8026–8037. [Online]. Available: http://papers.nips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf

[27] Asano YM., Rupprecht C., and Vedaldi A. A critical analysis of self-supervision, or what we can learn from a single image. In *International Conference on Learning Representations*, 2020.

[28] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*, 2015.

[29] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. E. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," *CoRR*, vol. abs/1409.4842, 2014. [Online]. Available: http://arxiv.org/abs/1409.4842

[30] Tom Young, Devamanyu Hazarika, Soujanya Poria, and Erik Cambria. Recent trends in deep learning based natural language processing, 2018.

[31] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *CoRR*, vol. abs/1512.03385, 2015. [Online]. Available: http://arxiv.org/abs/1512.03385

[32] M. Tan and Q. V. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," *CoRR*, vol. abs/1905.11946, 2019. [Online]. Available: http://arxiv.org/abs/1905.11946

[33] Charlie Nash, Nate Kushman, and Christopher K.I. Williams. Inverting supervised representations with autoregressive neural density models. In Kamalika Chaudhuri and Masashi Sugiyama, editors, *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, volume 89 of *Proceedings of Machine Learning Research*, pages 1620–1629. PMLR, 16–18 Apr 2019.

[34] Ziv Goldfeld, Ewout Van Den Berg, Kristjan Greenewald, Igor Melnyk, Nam Nguyen, Brian Kingsbury, and Yury Polyanskiy. Estimating information flow in deep neural networks. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 2299–2308. PMLR, 09–15 Jun 2019.

[35] Yoav Goldberg and Graeme Hirst. *Neural Network Methods in Natural Language Processing.* Morgan & Claypool Publishers, 2017.

[36] Athanasios Voulodimos, Nikolaos Doulamis, Anastasios Doulamis, and Eftychios Protopapadakis. Deep learning for computer vision: A brief review. *Computational Intelligence and Neuroscience*, 2018:1–13, 02 2018.

[37] Niall O' Mahony, Sean Campbell, Anderson Carvalho, Suman Harapanahalli, Gustavo Adolfo Velasco-Hernández, Lenka Krpalkova, Daniel Riordan, and Joseph Walsh. Deep learning vs. traditional computer vision. *CoRR*, abs/1910.13796, 2019

[38] Hai Nguyen and Hung La. Review of deep reinforcement learning for robot manipulation. In *2019 Third IEEE International Conference on Robotic Computing (IRC)*, pages 590–595, 2019.

[39] Harry A. Pierson and Michael S. Gashler. Deep learning in robotics: A review of recent research, 2017.

[40] A. G. Journel and C. V. Deutsch, "Entropy and spatial disorder," *Mathematical Geology*, vol. 25, no. 3, pp. 329–355, 1993.

[41] Ekaba Bisong. *Google Colaboratory*, pages 59–64. Apress, Berkeley, CA, 2019.

[42] Gao Huang, Zhuang Liu, and Kilian Q. Weinberger. Densely connected convolutional networks. *CoRR*, abs/1608.06993, 2016.

[43] Paul Parsons and Kamran Sedig. Common visualizations: Their cognitive utility. In Handbook of Human Centric Visualization, pages 671–691. Springer, 2014.

[44] Charles S. Peirce, *Collected papers of charles sanders peirce.* Harvard University Press, 1960, vol. 2.

[45] Răzvan Andonie, "A semiotic approach to hierarchical computer vision," in *Cybernetics and Systems (Proceedings of the Seventh International Congress of Cybernetics and Systems, London, Sept. 7-11, 1987)*, J. Ross, Ed. Lytham St. Annes, U.K.: Thales Publication, 1987, pp. 930–933.

[46] Max Bense, *Semiotische Prozesse und Systeme in Wissenschaftstheorie und Design, Ästhetik und Mathematik.* Baden-Baden: Agis-Verlag, 1975.

[47] Helmar Frank, *Kybernetische Grundlagen der Pädagogik: eine Einführung in die Informationspsychologie und ihre philosophischen, mathematischen und physiologischen Grundlagen*, second edition ed. Baden-Baden: Agis-Verlag, 1969.

[48] Ioan Stan and Răzvan Andonie, "Cybernetical model of the artist-consumer relationship (in Romanian)," *Studia Universitatis Babes-Bolyai*, vol. 2, pp. 9–15, 1977.

Cite this paper as:
Muşat B.; Andonie R. (2022). Information Bottleneck in Deep Learning - A Semiotic Approach, *International Journal of Computers Communications & Control*, 17(1), 4650, 2022.
https://doi.org/10.15837/ijccc.2022.1.4650