Convolutional Neural Networks: Challenges and Trends

Seyyed Hossein Hasanpour*
Department of Computer Science
IAU. Science and research branch of Ayatollah Amoli
Amol, Iran
St.h.hasanpour@iauamol.ac.ir

Reza Saadati
Department of Mathematics
Iran University of Science and Technology
Noor, Iran
Rsaadati@iust.ac.ir

***Abstract*— Convolutional neural network has gained enormous success in recent years, and is one of the most popular deep learning algorithms that has been extensively used in many machine learning related fields. The success and different applications of CNN have been studied and addressed in many studies in the literature, however, some aspects which interestingly are very important are either less worked on or ignored completely. In this paper we study and address some of the aspects and respective trends that affect the application of CNN in various fields.**

Keywords-Convolutional neural network, deep learning, computer vison, machine learning

I. Introduction

Deep learning is a branch of machine learning in which it is tried to learn high level features and abstractions using hierarchical architecture. Although the idea of deep learning has been known for decades, it's only recently that it has become popular and practical solely due to the elimination of computational and data availability barriers. The abundance and easy access to needed data and powerful GPUs, along with considerable advances in machine learning algorithms made deep learning possible. In recent years many deep learning methods have been studied [1-5], and Convolutional Neural Network (CNN) has been one of the most successful methods amongst them.

CNN, is a type of feed-forward artificial neural network by Yann Lecun[6] that has gained enormous popularity and success in many machine learning related fields in recent years[7-10]. CNN design is inspired by the organization and biological processes in primate visual cortex and is the successor to Neocognitron [11] which was also inspired by Hubel and Wiesel's researches on cat's visual cortex[12]. CNN is a variation of multi-layer perceptron and is designed to use the minimum amount of preprocessing. Its shared weight architecture and the concept of receptive field made it less prune to over-fitting and take input locality into account respectively. Although CNN has been used since its introduction in 1989, its deeper versions are only used as one of the most popular Deep Learning algorithms in recent years.

The success of CNN in various fields has been studied and talked about a lot, but its other aspects is either less discussed or entirely neglected. In this paper we pay our attention to aspects that are less discussed or are entirely neglected, we address some of the challenges and trends that currently exist in the literature, and hope to shed some light on the other side of CNNs.

II. Theoretical understanding

Although astonishingly good results are achieved using deep learning methods, specifically using CNN, the underlying and theoretical basis is yet to be discovered, there still does not exists a firm understanding based on which one can know which architecture works better than the other one or which design, or how many layers, or neurons is the optimal choice for a specific task. Choosing the proper values for important hyper parameters such as learning rate and regularization is also a daunting task that is yet to be unraveled. Designing architectures has been an ad-hoc practice so far. Chu et al [13] however, tried to propose a theoretical method for specifying the optimal number of feature maps, but their method is only applicable to the networks with extremely small receptive fields. In order to better understand how a CNN architecture works, several visualization methods have been proposed in recent years. Zeiler et al [14] devised a novel visualization technique that provided a new insight into underlyings of a CNN architecture, and what happens inside it. Using identifiable patterns, their method could provide facilities to better design a CNN architecture. In their method they map activations in the intermediate

layers back to input pixel space, showing which input patterns or features cause the corresponding activations. They also used a sliding window occlusion approach to create a heat map visualizing the parts of the images that the most affect the classification.

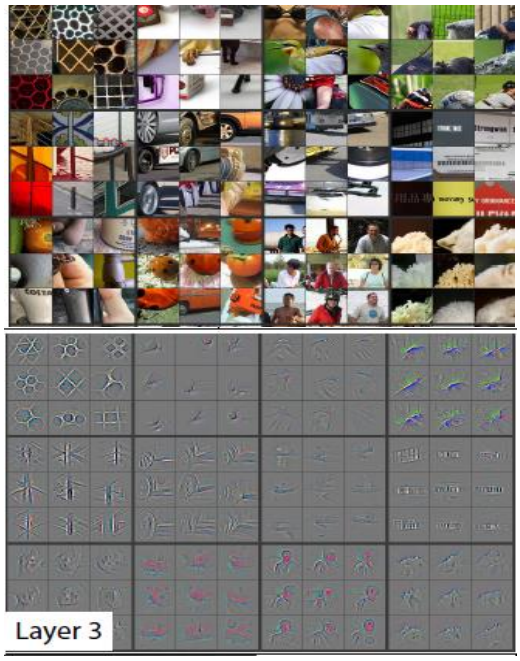

Fig. 1. *Showing visualization technique proposed by Zeiler et al*

Fig. 2 *showing two samples generated using Inceptionism technique (tabby cat, flamingos)*

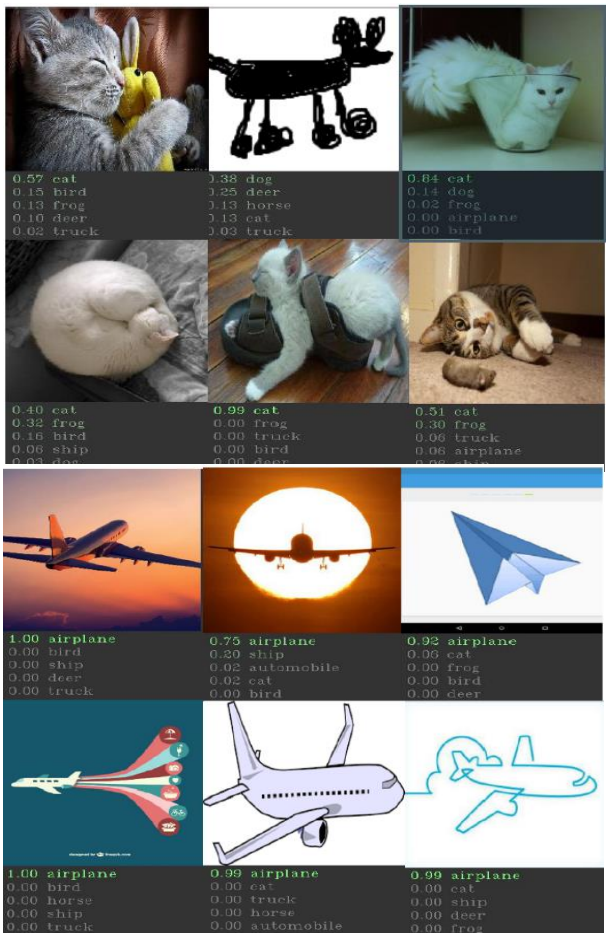

Fig. 3 *showing SimpleNet generalization power achieved through proper 1x1 conv filters*

Yu et al [15] tried to shed light on the internal work mechanism of CNNs by probing the internal representations in two comprehensive aspects, visualizing patches in the representation spaces from different layers, and visualizing visual information in each layer. They then further compared CNNs with different depths and showed the advantages brought by deeper architecture. Mordvintsev et al proposed a new technique called Inceptionism [16], which is a gradient-based reconstruction approach with which one can see what is encoded in a CNN by inverting their deep representation [17]. They achieved this by maximizing activations in a layer, by altering the input image so that it will make the most strongly activated features even stronger [18].

Zhou et al. proposed a different visualization method in which irrelevant regions in images are masked out to accentuate the significant region based on the actual receptive field and feature map [19].

Despite feature visualization, Girshick et al [18] tried to discover the CNN learning pattern, they inspected each layer during training phase, and found out that

convolutional layers, learn general features more often, and withhold the largest learning capacity in the network. Fully connected layers however, are domain specific. In addition to convolutional neural networks feature analysis, Argawal et al [20] investigated the effect of using some commonly used strategies such as fine-tuning and pre-training on the performance of convolutional neural network and provided evidence to be applied in computer vision related tasks.

Rusu et al [21] recently published their paper in which they showed how fine-tuning causes the network to lose what it has already learned and perform worse!, they proposed a new architecture, named progressive network, in which they are immune to forgetting and can leverage prior knowledge via lateral connections to previously learned features. Using a novel sensitivity measure, they demonstrate that transfer occurs at both low-level sensory and high-level control layers of the learned policy. Hasanpour et al [22] also proposed a new simple fully convolutional architecture that despite of having only 13 layers, outperforms all previously deeper architectures such as ResNet[23] and GoogleNet[24]. They explained in their paper, that 1x1 convolutional neural network can act as a feature combiner and proper appliance of them can increase the networks generalization power and achieve higher level abstraction much faster compared to the using of consecutive convolutional layer with conventional filter sizes. This clearly shows despite the recent achievements in deep learning theory, there is still significant room for further understanding and optimization of neural networks and a lot needs to be discovered in this regard.

III. Reaching human level vision

Human level vision has a significant and profound use in the applications and activities of computer vision field. Both in simple visual representations, and under geometric, background, or occlusion alterations. Since CNN could achieve ground breaking success in Computer vision, it is not surprising to try to harness the power and flexibility it provides. CNN is exceptionally great when it comes to images and doing great under diverse deformations. It is so good to the point where it exceeds human level precision on ImageNet challenge in 2015[23], making one believe it has finally bridged the semantic gap, which was a far reaching dream for researchers for years, to achieve a human level vision in computer vision. But is it true? Does CNN really perform just as good as human brain or are there any complications?

Compared to traditional low level features, CNN, imitates the organization and structure of human brain and creates different levels of features. [7] Conducted a study and sought to assess how much improvement can be achieved using deep learning techniques, and whether deep features are the lost key for filling the semantic gap in the long term. The image classification error in ImageNet competition, has decreased from 10% in 2012[7] to 4.82% in 2015[23]. This improvement, verifies the performance of CNN, Specifically the results achieved by ResNet[23] surpasses human level accuracy. But it is still too soon to say the performance of a CNN is on same level as of human brain. A practical clue to back up our statement, can be seen by devising a simple experiment. Creating images that can be unrecognizable by humans are relatively easy, but this is not the case for a convolutional neural network. A CNN can be as sure as 99% that the very same picture that is unrecognizable to humans, contains an inferable object [7]

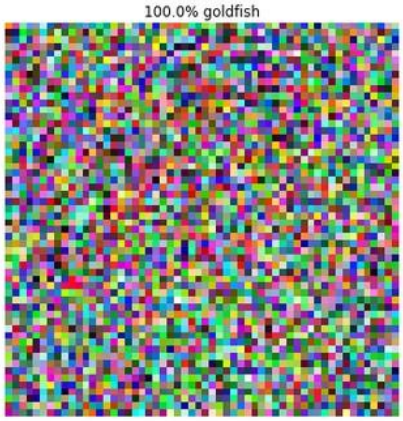

Fig. 4 *Noise image recognized as 100% goldfish!*

This clearly shows the different between human and inhuman vision achieved by current CNN models, and questions the CNNs generality in computer vision. Apart from that, a conducted study by [25] showed that, like human brain, CNN creates similar feature space for each category and different feature spaces for other categories, and this shows that CNN may provide an insight for primate visual processes. So it is still not apparent whether CNN models which are based on calculation mechanisms, look like primate visual system. Though there is potential for further improvement by Imitating and using primate visual system ..Larger models show larger capacity and the trends have been towards this direction with some exceptions [22, 26], but the lack of training data may restrict the size and power of these models. Especially when getting fully labeled data is very expensive. There is not a firm answer to questions such as how can one overcome the crucial need for training data? Or how to train gigantic networks. However currently there are several methods which are used to solve the first problem. One being data-augmentations such as scaling, rotation, cropping, from existing training images. Or as wu et al [27] proposed, one can use color casting, vignetting and lense distortions. Weak learning is another method that one can use to collect more training samples. Some has also used search engines to collect their needed training samples [28, 29]. Zhou et al [30] proposed a method called Concept learner, in order to scale up computer vision systems, which can automatically learn thousands of visual concept detectors from a weakly labeled dataset. A very interesting approach that was initially proposed by Goodfellow et al [31] in 2014, and achieved very promising results lately is the Generative Adversarial

Network (GAN) [32]. The main idea behind a GAN is to have two competing neural network models. One takes noise as input and generates samples (and so is called the generator). The other model (called the discriminator) receives samples from both the generator and the training data, and has to be able to distinguish between the two sources. These two networks play a continuous game, where the generator is learning to produce more and more realistic samples, and the discriminator is learning to get better and better at distinguishing generated data from real data. These two networks are trained simultaneously, and the hope is that the competition will drive the generated samples to be indistinguishable from real data.

So far GANs have been primarily applied to modelling natural images. They are now producing excellent results in image generation tasks, generating images that are significantly sharper than those trained using other leading generative methods based on maximum likelihood training objectives.

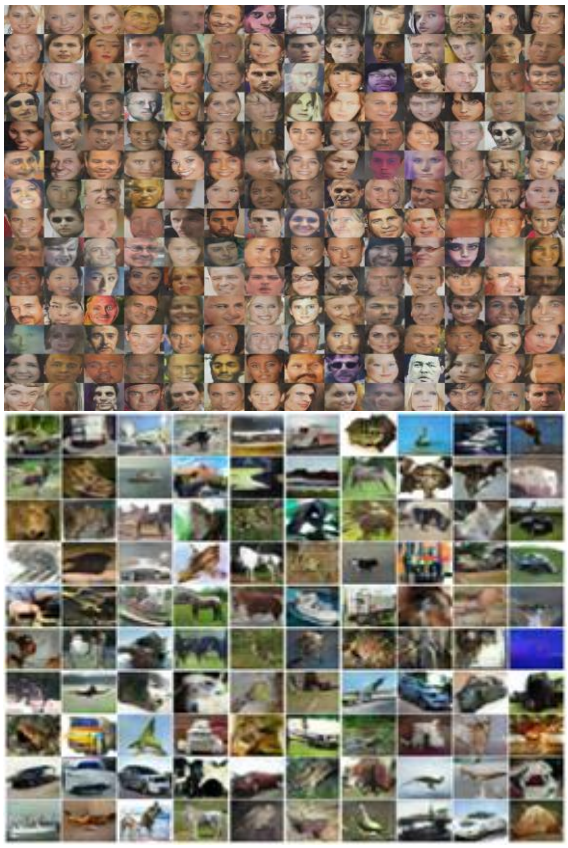

Fig. 5 *Showing Deep Convolutional Generative Adversarial Network (DCGAN) generated images*

Fig. 6 *Showing Deep Convolutional Generative Adversarial Network (DCGAN) generated images*

Looking at these promising techniques specially GANS, and further enhancing them can result in more training data and therefor will allow networks to better learn more robust features and utilize the capacity they withhold.

IV. Computational Overhead

Deep Convolutional neural networks need a lot of computational resources and therefor they can't be easily used in real-time applications on devices with limited computation resources. One of today's trends in this field, is developing architectures that allow running CNN in real-time with less resources. A study was conducted by [33], in which a series of tests were conducted in a limited time, and models were suggested that were applicable for real-time applications and at the same time perform competitive to normal CNNs. [33]conducted another study in which they eliminated all extra computations in forward and back ward passes and achieved a speed up of over 1500 times more. This model has a good flexibility with CNN with diverse designs and structures and because of its optimal GPU implementation, it achieves a high performance. Ren et al vectorized key operators in CNN to achieve high parallelism using matrix operators.

Bucila et al.[34], proposed a method in which they tried to create a network that performs like a complex and large ensemble. They used the ensemble to label unlabeled data with which they train the new neural network, thus learning the mappings learned by the ensemble and achieving similar accuracy. This idea was further worked on by Ba & Caruana [35] by compressing deep and wide networks into shallower but even wider ones.

Recently Han et al[36] released their work on model compression. They introduced "deep compression", a three staged pipeline: pruning, trained quantization and Huffman coding, that work together to reduce the storage requirement of neural networks by 35 to 49 times without affecting their accuracy. In this method, the network is first pruned by learning only the important connections. Next, the weights are quantized to enforce weight sharing, finally, the Huffman coding is applied. After the first two steps they retrain the network to fine tune the remaining connections and the quantized centroids. Pruning, reduces the number of connections by 9 to 13 times; Quantization then reduces the number of bits that represent each connection from 32 to 5. On the ImageNet dataset, their method reduced the storage required by AlexNet by 35 times, from 240MB to 6.9MB, without loss of accuracy.

Iandola et al[26] proposed a novel architecture called, squeezenet, a small CNN architecture that achieves AlexNet-level[7] accuracy on ImageNet With 50 times fewer parameters.

Hasanpour et al[22] proposed a simple architecture which achieves state of the art result on CIFAR10. Their network has much fewer parameters (2 to 25 times less) compared to all previous deep architectures, and performs either superior to them or on par with them despite the huge difference in parameters.

V. Directions toward creating more powerful models

Since Deep learning algorithms could achieve surprisingly very high results, advancing and enhancing their results has been a daunting task. There are several research directions that one can pursue for creating more powerful models. The first direction that has been being used extensively in the past 3 years, is increasing network generalization capability by increasing the depth of the network. Larger networks are usually able to provide better performance.

Using this guideline, one needs to pay careful attention to issues such as over-fitting and increased computational and memory overhead. The second direction is combining information from multiple sources. Feature fusion has been popular for some time now. DensNet[37] is a new architecture that uses similar concept to achieve high performance. The third direction that is very new and is being practiced independently from the beginning of 2016, is to created smaller architectures that perform on part, a prime example for such paradigm are [22, 26]

VI. Security and privacy challenges

Are Convolutional Neural networks reliable? Answering to this question may seem trivial, but it may not be as easy as it seems. We already talked about how a CNN can make up something out of a completely unrecognizable image to humans. But how serious can the implications of this simple difference be?

VII. Fooling CNNs

It is possible to take an image that a CNN already classifies as one class and perturb it in a way imperceptibly to the human eye in such a way that the very same network suddenly classifies the image as any other class of choice.

This phenomenon has been addressed by several researchers [38-40] , and it is not a random artifact pertaining to the learning strategy, the same perturbation can cause a different network, which was trained on a different subset of the dataset, to misclassify the same input. This have very important implications but should also be noted that these results are not CNN exclusive, and similar observations have been reported from older features, e.g. HOG features as well[41].

Early attempts at explaining this phenomenon focused on nonlinearity and overfitting. But the primary cause of vulnerability to these perturbation is their linear nature[38] . In order to counter this vulnerability, training with more diverse data and adversarial samples is proposed.

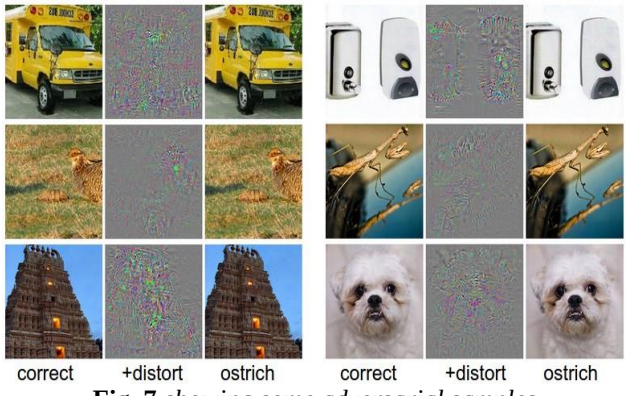

Fig. 7 *showing some adversarial samples*

VIII. Extracting private data out of a trained models

With the huge success of CNNs, they have become a crucial component of almost any image understanding system. In the previous section we observed that how a CNN can be fooled and the implications of such vulnerability is serious enough that makes one to take necessary precautions in its application in certain areas. What we talked about in previous section covered the active part of a CNN backed system. Does the CNN model itself pose a thread of any kind? Let's rephrase the question as How much information can be extracted form a pre-trained CNN model?

Mahendran and Vedaldi[17] tried to answer this question by conducting a research on, SIFT, HOG and also CNN models and see how much information can be

extracted from a model. The results they achieved is surprising. Among their findings, they show that several layers in CNNs retain photographically accurate information about the image, with different degrees of geometric and photometric invariance. In their paper they conducted a direct analysis of the visual information contained in representations and tried to answer, given an encoding of an image, to which extent is it possible to reconstruct the image itself? They contributed a general framework to invert representations. They then used this technique to study the inverse of recent state-of-the-art CNN image representations for the first time.

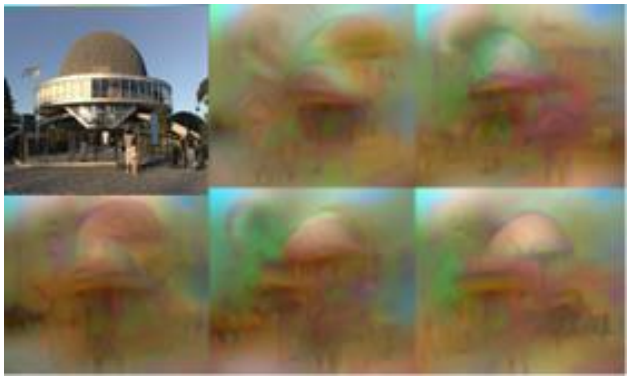

Fig. 8 *showing five possible reconstructions of the reference image obtained from the 1,000-dimensional code extracted at the penultimate layer of a reference CNN trained on the ImageNet data*

Fig. 10 *showing how the depth of the network affects the retrievable information*

Early convolutional layers, contain a lot of information about the image, and one can extract almost an identical representation from early layers, whereas in later layers, these photographically accurate information decreases dramatically as one gets closer to the end of the network.

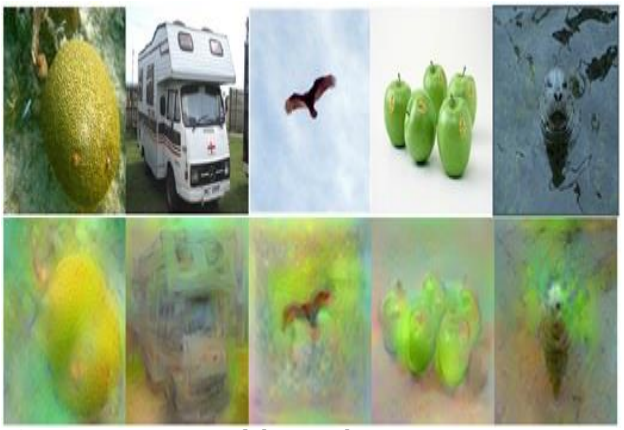

Figure 9. *A CNN model. Mpool 5 reconstructions, showing how much information the network retains at such deep levels.*

This poses another vulnerability, this time in a passive sense. It clearly shows how critical it is to pay more attention to the safety and security of a simple CNN model, and how one's intellectual assets can be at risk, especially if it was trained on sensitive data.

IX. Conclusion

In this paper we talked about several key aspects and respective trends about CNNs and how they affect their applications in different fields. We explained the underlying theory that makes CNN to perform this well is not known, but there are methods to give a clue what happens in a CNN, we reviewed some of these techniques and got familiar with challenges and trends in that regard. We explained about computational overhead and how it negatively impacts the growth of deep learning applications utilizing CNN, we then talked about current trends and future research directions in CNN design space. We also talked about the Security vulnerabilities that exists in any CNN model and how it can affect the organization which uses it and if possible how it can be.

REFERENCES

[1] Schmidhuber, J., *Deep learning in neural networks: An overview.* Neural Networks, 2015. **61**: p. 85-117.

[2] Bengio, Y., *Deep Learning of Representations: Looking Forward*, in *Statistical Language and Speech Processing: First International Conference, SLSP 2013, Tarragona, Spain, July 29-31, 2013. Proceedings*, A.-H. Dediu, et al.,

Editors. 2013, Springer Berlin Heidelberg: Berlin, Heidelberg. p. 1-37.

[3] Bengio, Y., *Learning Deep Architectures for AI*. Found. Trends Mach. Learn., 2009. **2**(1): p. 1-127.

[4] Bengio, Y.a.C., A. and Vincent, P., *Representation Learning: A Review and New Perspectives.* Pattern Analysis and Machine Intelligence, IEEE Transactions on, 2013. **35**: p. 1798-1828.

[5] Deng, L., *A tutorial survey of architectures, algorithms, and applications for deep learning.* APSIPA Transactions on Signal and Information Processing, 2014. **3**: p. null-null.

[6] LeCun, Y., et al., *Gradient-based learning applied to document recognition.* Proceedings of the IEEE, 1998. **86**(11): p. 2278-2324.

[7] Krizhevsky, A., I. Sutskever, and G.E. Hinton. *Imagenet classification with deep convolutional neural networks*. in *Advances in neural information processing systems*. 2012.

[8] Girshick, R., *Fast R-CNN*. The IEEE International Conference on Computer Vision (ICCV), 2015.

[9] LeCun, Y., Y. Bengio, and G. Hinton, *Deep learning.* Nature, 2015. **521**(7553): p. 436-444.

[10] Long, J., E. Shelhamer, and T. Darrell. *Fully convolutional networks for semantic segmentation*. in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015.

[11] Fukushima, K., *Neural Network Model for a Mechanism of Pattern Recognition Unaffected by Shift in Position- Neocognitron.* ELECTRON. & COMMUN. JAPAN, 1979. **62**(10): p. 11-18.

[12] Hubel, D.H. and T.N. Wiesel, *Receptive fields, binocular interaction and functional architecture in the cat's visual cortex.* The Journal of physiology, 1962. **160**(1): p. 106-154.

[13] Chu, J.L. and A. Krzyżak, *Analysis of feature maps selection in supervised learning using convolutional neural networks*, in *Advances in Artificial Intelligence*. 2014, Springer. p. 59-70.

[14] Zeiler, M.D. and R. Fergus, *Visualizing and understanding convolutional networks*, in *Computer vision–ECCV 2014*. 2014, Springer. p. 818-833.

[15] Yu, W., et al., *Visualizing and Comparing Convolutional Neural Networks.* arXiv preprint arXiv:1412.6631, 2014.

[16] Mordvintsev, A., C. Olah, and M. Tyka, *Inceptionism: Going deeper into neural networks.* Google Research Blog. Retrieved June, 2015. **20**.

[17] Mahendran, A. and A. Vedaldi. *Understanding deep image representations by inverting them*. in *2015 IEEE conference on computer vision and pattern recognition (CVPR)*. 2015. IEEE.

[18] Mahendran, A. and A. Vedaldi, *Visualizing deep convolutional neural networks using natural pre-images.* International Journal of Computer Vision, 2016: p. 1-23.

[19] Zhou, B., et al., *Object detectors emerge in deep scene cnns.* arXiv preprint arXiv:1412.6856, 2014.

[20] Agrawal, P., R. Girshick, and J. Malik, *Analyzing the performance of multilayer neural networks for object recognition*, in *Computer Vision–ECCV 2014*. 2014, Springer. p. 329-344.

[21] Rusu, A.A., et al., *Progressive neural networks.* arXiv preprint arXiv:1606.04671, 2016.

[22] HasanPour, S.H., et al., *Lets keep it simple: using simple architectures to outperform deeper architectures.* arXiv preprint arXiv:1608.06037, 2016.

[23] He , K., et al., *Deep Residual Learning for Image Recognition.* CoRR, 2015. **abs/1512.03385**.

[24] Szegedy, C., et al. *Going deeper with convolutions*. in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015.

[25] Cadieu, C.F., et al., *Deep neural networks rival the representation of primate it cortex for core visual object recognition.* PLoS Comput Biol, 2014. **10**(12): p. e1003963.

[26] Iandola, F.N., et al., *SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and< 1MB model size.* arXiv preprint arXiv:1602.07360, 2016.

[27] Wu, R., et al., *Deep image: Scaling up image recognition.* arXiv preprint arXiv:1501.02876, 2015. **22**: p. 388.

[28] Divvala, S., A. Farhadi, and C. Guestrin. *Learning everything about anything: Webly-supervised visual concept learning*. in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2014.

[29] Chen, X., A. Shrivastava, and A. Gupta. *Neil: Extracting visual knowledge from web data*. in *Proceedings of the IEEE International Conference on Computer Vision*. 2013.

[30] Zhou, B., V. Jagadeesh, and R. Piramuthu. *Conceptlearner: Discovering visual concepts from weakly labeled image collections*. in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015.

[31] Goodfellow, I., et al. *Generative adversarial nets*. in *Advances in Neural Information Processing Systems*. 2014.

[32] Radford, A., L. Metz, and S. Chintala, *Unsupervised representation learning with deep convolutional generative adversarial networks.* arXiv preprint arXiv:1511.06434, 2015.

[33] He, K. and J. Sun. *Convolutional neural networks at constrained time cost*. in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015.

[34] Buciluǎ, C., R. Caruana, and A. Niculescu-Mizil. *Model compression*. in *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*. 2006. ACM.

[35] Ba, J. and R. Caruana. *Do deep nets really need to be deep?* in *Advances in neural information processing systems*. 2014.

[36] Han, S., H. Mao, and W.J. Dally, *Deep compression: Compressing deep neural network with pruning, trained quantization and huffman coding.* CoRR, abs/1510.00149, 2015. **2**.

[37] Huang, G., Z. Liu, and K.Q. Weinberger, *Densely Connected Convolutional Networks.* arXiv preprint arXiv:1608.06993, 2016.

[38] Goodfellow, I.J., J. Shlens, and C. Szegedy, *Explaining and harnessing adversarial examples.* arXiv preprint arXiv:1412.6572, 2014.

[39] Nguyen, A., J. Yosinski, and J. Clune. *Deep neural networks are easily fooled: High confidence predictions for unrecognizable images*. in *Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on*. 2015. IEEE.

[40] Szegedy, C., et al., *Intriguing properties of neural networks.* arXiv preprint arXiv:1312.6199, 2013.

[41] Tatu, A., et al. *Exploring the representation capabilities of the HOG descriptor*. in *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*. 2011. IEEE.