STATE OF THE ART RESEARCH SURVEY IN DEEEP LEARNING

Luis Ángel Seda Marcos - A01795301, Eduardo Rodrigo Silva Orozco - A01795937, Gerardo Solís Hernández - A00952702, José Adán Vega Pérez - A01796093, Owen Jáuregui Borbón - A01638122
Instituto Tecnológico y de Estudios Superiores de Monterrey

***Abstract*— This paper explores recent advancements and future trends in Deep Learning, a rapidly evolving field that has revolutionized artificial intelligence. We analyze breakthroughs in Computer Vision, summarizing key contributions from the last three years. A critical review of literature highlights the impact, challenges, and limitations of current methods. Furthermore, we identify research gaps and propose future directions to drive innovation in Deep Learning. This survey aims to provide researchers and practitioners with a comprehensive understanding of recent developments and emerging trends.**

***Index Terms*— Deep Learning, Computer Vision, Convolutional Neural Networks (CNNs), Object Detection, Image Segmentation, Artificial Intelligence**

I. Introduction

Deep Learning has become a cornerstone of modern artificial intelligence, achieving state-of-the-art performance in various domains such as computer vision, natural language processing (NLP), and reinforcement learning. In this paper, we survey the recent advancements in Computer Vision and discuss their implications for research and industry. The motivation for this study is to analyze current trends, highlight innovative techniques, and provide insights into the challenges and future directions of Deep Learning in Computer Vision.

II. Literature Review

A. Convolutional Neural Networks (CNNs)

CNNs are characterized by being complex Deep Learning structures that use *filters* that are used to apply the same parameters to patches of the previous layer. In order for a newral network to be called a CNN, at east one later of the network has to use the convolution operation (be a convolutional layer, explained further on this paper) [1]. This type of structure has a really unique translational invariance, allowing the network to apply the learned patterns to objects anywhere in the input data. [2] There are 3 essential types of layers used in CNNs: convolutional layers (section II-A.1), pooling layers (section II-A.2), and fully connected layers (section II-A.3). [2] Each of these bring value to the complete CNN in a different way.

1) Convolutional Layer: Convolutional layers utilize multiple kernels to convolve the input or the feature map from the previous layer. Each of this kernels is applied to the whole input by sliding the convolution kernel along the data, executing a dot product between the input's current window and the kernel [3]. This convolutions generate feature maps, extracting key elements from the image [2].

Due to applying the same kernels to the full image, less trainable parameters are required to get good results compared to fully connected layers. This also leads to a better training performance, increasing the model's accuracy while taking less time to train [4].

2) Pooling Layer: After a convolutional layer, it is common to use pooling layers to down-sample the output [3]. After a convolution layer the output increases its dimensions according to the amount of filters used. The output dimensions to a convolutional layer will be determined by the input width and height, as well as the amount of filters used. After multiple convolutional layers and filters applied, this dimension can escalate greatly, increasing the computational complexity at the same time.

This layer is a solution to this problem. To avoid the increasing dimensionality, the pooling layers sample the output from the previous convolutional layer. This sampling can vary according to implementation, but some of the most commonly used methods are average pooling and max pooling [2]. Both of these do as the name suggests, taking the maximum or the average for the current window to assign the value to the current neuron. The stride for this type of layer is the same as the dimensions from the kernel. Therefore, by using this kernel and stride, the operation reduces the dimensions by the size of the kernel used [3].

3) Fully Connected Layer: Opposed to a convolutional layer or a pooling layer, the fully connected layer doesn't use any kernels. Instead, like a multilayer perceptron neural network, each neuron is connected one by one with all the neurons from the preceding layer [3].

This layer is better explained if we see the convolutional layers as feature extraction layers. Each of the convolutional layers refines the previous features and creates more complex features. Then, we can infer that the last convolutional layer has the best features to match our data. Using a fully connected layer at the end allows the CNN to determine what features better represent the output. As another way of explaining this layer, this could be considered as a mapping layer that pull all the feature mappings from the previous layer and maps them to the output neurons.

B. CNNs in Medical Image Understanding

Medical imaging has a large spectrum, some of these examples are: X-ray, CT, MRI, PET and ultrasound. Imaging is necessary to detect abnormalities in anatomy and function of different organs. When doctors look to this images they search for patterns to determine if the organ is healthy or not.

If an abnormality is detected, then more information about it is required, like shape, size and other important information. Some of the currently existing CNNs trained for these tasks can be grouped in 5 different types of models based on the networks goal: classification, segmentation, localization, and detection [5]. [6]

Medical image CNNS have a large range of applications currently. Some examples of these applications are for detecting lung diseases, breast tumors, heart diseases, eye diseases, brain disorders, among others. Each of these tasks has a different architecture that works best for it. Specially due to the difference between each type of imaging, it makes sense to have a different structure to extract the most features out of the input image [5].

Let's use brain disorders classification as an example. One specific case is to classify Alzheimer's disease. For this task, the fusion for two-dimensional and three-dimensional CNNs proved to get a higher accuracy. Since this disease is characterized by the destruction of brain-cells, it is logical to think that using both 3D and 2D images help to classify the disease with higher accuracy. The 3 dimensional CNN is used to determine key information that would be missed if omitting the Z axis. At the same time, a 2D CNN analyzes a CT image that was normalized. The output from the last convolutional layer for the 2D CNN is then fuzed with the 3D convoluted data to get 3 different classes: Alzheimer's, lesions, and healthy data [5].

C. Deep Video Codec Control for Vision Models

In this paper authors explain how traditional video codecs, that are mainly designed for human perception are not optimized to fit needs of deep vision models. Codecs such as H.364 are optimized for reducing distortion perceived by human eye, however they do not take into consideration how they will impact performance of deep vision models.

Conventional video codecs for compressing video present challenges when applied to deep vision models, these models are trained using big sets of visual data and quality is crucial for performance. Codecs such as H.264 compress videos suppressing redundant data and applying techniques that minimize distortion for human eye visualization, however this may imply a significant loss of information for deep vision models impacting their performance on downstream tasks such as segmentation or optical flow [7].

In order to improve on this problem authors propose a new technique called Deep Video Codec Control, this technique focuses on a refined control of video compression, optimizing performance for deep vision models. This approach considers a self learning model that can be trained alongside deep vision models. Instead of using predefined conventional codecs the proposal adjusts compression in such way that it preserves most important visual features for downstream. This adjustment forks on an end-to-end fashion in order to make such adjustments according to downstream.

Authors applied this technique and compared results to traditional codecs, downstream tested application were segmentation and optical flow, using common performance metrics, they proved that Deep Video Codec Control performed better than default codecs such as H.264. This paper presents an interesting proposal and very relevant to current applications of vision systems. Adapting codecs to be more effective for deep vision applications is innovative, however they may be some challenges, for example the end-to-end approach might be computationally heavy, which could be a limiting feature for real time applications. Besides, even though evidence resulted on better performance, implementation of this approach might be difficult on infrastructure cases where design is based on traditional existing codecs.

D. Challenges and Practices of Deep Learning Model Reengineering: A Case Study

Published in Empirical Software Engineering, 2024, this papers reviews the process of reengineering models of deep learning models in the field of computer vision. This process involves reuse, reproduce, adapt and improve existing models in order to satisfy new needs and improve performance. This study focuses on identifying challenges and practices related to reengineering fo these models, looking for a detailed vision of difficulties engineers face and strategies used to overcome these difficulties. [8]

Authors combine quantitative and qualitative analysis, such as reviewing defects in open source projects related to reengineering of computer vision models and interviewing engineers who participated in these kind of projects. This method gives an analysis of the challenges in this kind of applications, generally most of research is focused on technical challenges, however this perspective turns out to be interesting since there are also human factor challenges.

Main findings are related to documentation, many models lack detailed documentation and it makes learning difficult. Requirements are also anther field of improvement, many situations are related to this topic such as rapid evolving needs, poor description, or very optimistic timing. Implementation and testing resources, the more resources needed the more expensive project becomes. Software development experience, engineers that are specialized in deep learning discipline may lack knowledge in software development practices such as documentation and code writing guidelines, which may add inconsistencies and difficulties to development process.

Based on findings, authors propose a workflow that consists phases like problem comprehension, data preparation, model development and deployment, highlighting importance of continuous improvement and requirements adaptability.

III. Future Improvements

Several key areas require further exploration to advance Deep Learning in Computer Vision:

1) Efficiency Improvements: Novel model architectures, such as Vision Transformers (ViTs), challenge traditional CNNs by demonstrating superior performance in various vision tasks. Future research should explore hybrid models combining CNNs and ViTs for optimized accuracy and efficiency.

2) Self-Supervised Learning: Reducing dependence on large labeled datasets remains a major challenge. Future work should focus on self-supervised learning techniques that leverage unlabeled data for more scalable model training.
3) Adaptation of Video Codecs: The integration of deep learning-based compression techniques in real-time applications could enhance performance while maintaining computational feasibility.
4) Standardization in Deep Learning Software Engineering: The implementation of structured methodologies, similar to those in the automotive industry, could improve software development processes for deep learning models.

IV. Conclusion

Computer vision, and CNNs in particular, has a large range of applications. The use of convolutional layers help the neural network converge faster and get better accuracy for computer vision problems. The use of this networks in medicin has the opportunity to better detect abnormalities in the body and save lives. By having accurate trained models, we can detect diseases in earlier stages. The multiple different architectures allow us to better fit the structure of the CNN.

Regarding Deep Video Codec Control for Vision Models, this paper presents a promising improvement to a topic that with increasing importance due to the multiple fields of application such as is the computer vision and high volume of video data, however more investigation may be necessary in order to verify efficiency in other downstream applications and also to verify its feasibility in systems that are already designed to work with traditional codecs.

Most of papers related to these topics go for technical analysis, theoretical improvements or developments, however Challenges and Practices of Deep Learning Model Reengineering: A Case Study, presents a perspective of challenges related to these disciplines from human perspective, let us take into consideration that at the end of the day it is us engineers that will type code, collect data, prepare and maintain data infrastructure, verify results, implement adjustments, implement improvements, deploy new systems, it would be also worth it to start looking for standardized software development processes such as in the automotive industry which has multiple options for project management and different development paradigms, these methodologies might probe to be effective in reducing human errors in the process related to deep learning model development and deployment.

References

[1] X. Zhang, X. Zhang, and W. Wang, "Convolutional neural network," in *Intelligent information processing with Matlab*, Springer, 2023, pp. 39–71.

[2] A. Voulodimos, N. Doulamis, A. Doulamis, and E. Protopapadakis, "Deep learning for computer vision: A brief review," *Computational intelligence and neuroscience*, vol. 2018, no. 1, p. 7 068 349, 2018.

[3] X. Zhao, L. Wang, Y. Zhang, X. Han, M. Deveci, and M. Parmar, "A review of convolutional neural networks in computer vision," *Artificial Intelligence Review*, vol. 57, no. 4, p. 99, 2024.

[4] M. M. Taye, "Theoretical understanding of convolutional neural network: Concepts, architectures, applications, future directions," *Computation*, vol. 11, no. 3, p. 52, 2023.

[5] D. Sarvamangala and R. V. Kulkarni, "Convolutional neural networks in medical image understanding: A survey," *Evolutionary intelligence*, vol. 15, no. 1, pp. 1–22, 2022.

[6] A. Parvaiz, M. A. Khalid, R. Zafar, H. Ameer, M. Ali, and M. M. Fraz, "Vision transformers in medical computer vision—a contemplative retrospection," *Engineering Applications of Artificial Intelligence*, vol. 122, p. 106 126, 2023.

[7] C. Reich, B. Debnath, D. Patel, T. Prangemeier, D. Cremers, and S. Chakradhar, "Deep video codec control for vision models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 5732–5741.

[8] W. Jiang, V. Banna, N. Vivek, *et al.*, "Challenges and practices of deep learning model reengineering: A case study on computer vision," *Empirical Software Engineering*, vol. 29, no. 6, p. 142, 2024.