# Examining a Deep Learning framework to speed up AI for applications in Image Processing

**T. Prasad[1], V. Ganesan[2]**

[1]Research Scholar, Bharath Institute of Higher Education and Research, Chennai
[2]Professor, Department of ECE, Bharath Institute of Higher Education and Research, Chennai

**Abstract**
AI in many fields, especially image processing, has been greatly affected by the quick development of Deep Learning (DL) frameworks. In this work, the possibility of using deep learning approaches to improve AI efficiency when performing image-related tasks such object identification, segmentation, and classification is examined. Because of Convolutional Neural Networks' (CNNs) remarkable capacity for obtaining hierarchical features from visual data, we concentrate on them. By conducting thorough experiments and performance comparisons, this study explores the integration of two CNN-based architectures, ResNet50 and U-Net, into a unified framework tailored for image classification and segmentation tasks. The combined model leverages ResNet50's robust feature extraction for classification and U-Net's precise spatial mapping for segmentation, achieving high accuracy and computational efficiency. The proposed integrated model demonstrated superior performance compared to standalone deep learning architectures and conventional machine learning methods, establishing its viability for image processing tasks in practical applications. This approach underscores the potential of synergizing multiple CNN models to address complex image-based challenges in everyday scenarios.

**Keywords:** deep learning, AI, image processing, convolutional neural network, ResNet50, image recognition, real time image processing.

## 1. Introduction

Major progress has been achieved by AI in image processing, revolutionising a number of sectors, including healthcare, self-driving cars, recreation, and monitoring. In the past, conventional image processing methods depended on manually created features, wherein the qualities of pictures that were most pertinent for examination were determined by human skill. Due to their inability to automatically adjust to novel or invisible patterns, these techniques have trouble handling challenging jobs requiring high-dimensional data. The complexity of real-world photographs was therefore beyond the scope of these conventional methods, especially in tasks like object identification, image division, and facial recognition. Additionally, the techniques were relatively costly and time-consuming due to their reliance on human feature engineering. [1]

AI's method for image processing has been completely transformed with the introduction of deep learning structures, particularly Convolutional Neural Networks (CNNs). CNNs do not require human feature extraction because they are built to autonomously acquire structural models from plain image data. The efficiency of AI systems in image-related activities has been greatly improved by their capacity to automatically find pertinent characteristics.[2] While the deeper layers of CNNs capture more complicated patterns like forms and objects, lower levels concentrate on simpler characteristics like edges and textures. CNNs are extremely useful in a variety of programs, from real-time object recognition in autonomous cars to medical picture analysis for illness diagnosis, thanks to their end-to-end algorithm for learning. CNN models are a popular option for contemporary image processing systems because to their accuracy, speed, and efficiency optimisation, which pushes the limits of what artificial intelligence is capable of in this field. [3][4]

## 2. LTERATURE SURVEY

Machine learning (ML), which first appeared in the middle of the 20th century, is a branch of artificial intelligence (AI), a branch of computer science that focusses on creating devices and algorithms that can perform tasks—such as detection of speech, detection of images, and conversational understanding—ideally instinctively .[5]
Significant progress has been made in picture identification, segmentation, and classification tasks as a result of recent developments in deep learning, especially with Convolutional Neural Networks (CNNs). In their groundbreaking study, LeCun et al. emphasised the astounding effectiveness of deep learning methods in image processing, pointing out that CNNs performed better than more conventional ML like Random Forests (RF) and Support Vector Machines (SVMs), particularly when dealing with intricate and multidimensional image data. [6]
Notwithstanding their achievements, CNNs still have difficulties controlling big datasets, avoiding overfitting, and optimising models for immediate results. To solve these problems, scientists like Szegedy et al. (2015) and Radford et al. have suggested methods like dropout, data enhancement, and adaptive learning. By using these methods, models may analyse a variety of picture data more effectively, robustly, and generally without overfitting.[7]
ML, like other mathematical applications, frequently employs linear algebra procedures on multivariate arrays—computational data structures used to represent graphs, matrices, and vectors of a higher order—when offering novel approaches to designing AI models. Machine learning (ML) is a technique for evaluating data that facilitates the creation of computer techniques and modelling techniques. These tools are utilised for a wide variety of data kinds and are especially helpful for examining data and identifying possible trends in an effort to forecast future information. [8][9]
A thorough and well-thought-out plan for gathering data is necessary, and to guarantee high-quality data, it is frequently supplemented by a preparatory phase. To better comprehend the various models, several researchers have focused their efforts. The difficulty of deciphering and extracting valuable data from weights of neural networks has drawn more attention, which might result in mistaken beliefs and choices.[10]

## 3. METHODOLOGY

The goal of this research is to employ Convolutional Neural Networks (CNNs) to speed up artificial intelligence (AI) in image processing tasks like segmentation and classification. Specifically, we employ specialised medical picture datasets for segmentation tasks and publicly available datasets such as CIFAR-10 for classification. To improve resilience and lower the chance of overfitting, the photos go through crucial pre-processing procedures including scaling, normalisation, and data augmentation methods like flipping and random rotations. In order to train more effective models, pre-processing makes sure the data is standardised, consistent, and prepared for deep learning models.[11]

After training, the CNN model effortlessly separates pertinent elements from the pictures. Although the deeper parts of the network can identify more complicated patterns like forms and objects, lower levels can only recognise basic properties like borders and nuances. We choose designs like ResNet50 for classification jobs because of its capacity to manage deep networks with residual links in an efficient manner, which guarantees higher accuracy.[12] Because of its excellent performance in maintaining spatial information during picture segmentation, U-Net is used for segmentation tasks. Common metrics such as Intersection-over-Union (IoU) for segmentation and correctness for categorisation are used to assess the trained models. [13]

## 4. System Architecture

The system consists of several components:

**Image Data Input** : Raw images are fed into the system, either from local storage or online repositories.

**Preprocessing Module** : This module handles image resizing, normalization, and augmentation.

**Feature Extraction & Model Integration Layer** : CNN layers perform automatic feature extraction through convolution, pooling, and activation functions.

**Test-Train split** : Splitting the data separately for testing and training.

**Fully Connected Layer** : After feature extraction, the network proceeds to fully connected layers for classification or segmentation.

**Output and Evaluation** : The system outputs the predicted class or segmented region.

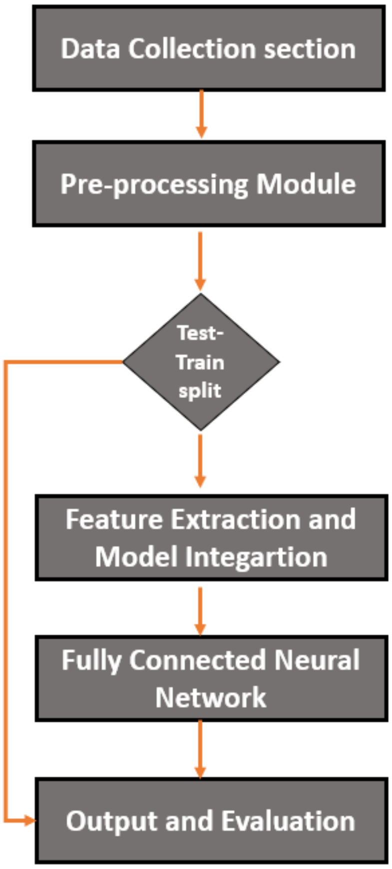


**Fig:1 System Architecture**

## 5. Implementation

In this paper, we use Convolutional Neural Networks (CNNs) to develop a deep learning-based image processing system for tasks such as picture segmentation and classification. Data gathering, pre-processing, feature extraction, model creation, and performance assessment are all steps in the structured process for execution.

### A. DATA COLLECTION

The collection of relevant datasets is the initial stage in our implementation. We employ the popular CIFAR-10 dataset, which comprises 60,000 32x32 colour pictures in 10 distinct classes, for image classification tasks. We utilise a specialised dataset of CT scans with tumour areas for medical picture segmentation, which necessitates great precision in identifying the area of interest. The datasets are well-structured for assessing CNN models and include both training and testing data. [14]

| Dataset | No.of Images | Image Dimension | Classes/ Labels |
| --- | --- | --- | --- |
| CIFAR-10 | 60,000 | 32x32 | 10 (Eg: Dog vs Cat) |
| Medical CT scans | 1,000 | Varies | Tumor vs non-tumor |

**Table 1: Sample Data distribution**

### B. PREPROCESS THE DATA

To make sure that the information is appropriate for CNN model training, it passes through a number of pre-processing stages when it is gathered. In order to speed up model development for the CIFAR-10 dataset, pixel values are normalized to the range [0, 1] and pictures are scaled to a constant 32x32 resolution. To lower the chance of overfitting, we also use data enhancement methods like flips, random rotations, and color modifications to artificially enlarge the dataset. While scaling and normalization are also applied to medical photos, special attention is paid to preserving the images' vital geographic data, which is necessary for precise tumor segmentation.[15]

### C. SPLIT THE DATA INTO TRAIN AND TEST

To guarantee dependable model performance, the dataset is divided into 70% training, 15% validation, and 15% testing. The test set is used to assess the model's accuracy and segmentation quality on unseen data, the validation set is used to adjust hyperparameters and avoid overfitting, and the training set is used for learning. [16]

### D. FEATURE EXTRACTION AND MODEL INTEGRATION

CNNs are particularly good at recognising complex characteristics from pictures; this eliminates the need for the extraction of features manually because the model learns to recognise patterns straight from the input. Convolutional procedures are used in the CNN's initial layers to identify basic characteristics like corners and surfaces. [17][28] More intricate patterns, such as forms, things, and even linguistic characteristics are recorded as we go further into the network. The feature maps are down-sampled using the pooling layers, which lowers the computational cost and complexity while maintaining the key characteristics. The CNN can learn generalisations through this feature retrieval process, which is essential for precise picture identification and segmentation applications. [19][20]

The ResNet50 model serves as the feature extraction backbone, processing input images to generate high-dimensional feature maps. These feature maps are shared between the classification head (fully connected layer) and U-Net's segmentation decoder. Skip connections are utilized to integrate spatial information from ResNet50 into U-Net's decoder for precise segmentation.

### E. MODEL CONSTRUCTION AND WORKING

Convolutional Neural Networks (CNNs) are a foundational deep learning architecture for processing grid-like data, such as images. They extract spatial and hierarchical features through convolutional layers, down sample using pooling layers, and perform final predictions with fully connected layers. In this study, CNN principles form the basis of ResNet50 and U-Net. ResNet50 utilizes CNNs for hierarchical feature extraction and classification, while U-Net employs an encoder-decoder structure for pixel-wise segmentation. By integrating these models into a unified CNN-based framework, the study achieves robust performance for both classification and segmentation tasks. Below is a diagram illustrating the general structure of a CNN, showcasing its layers and architecture.

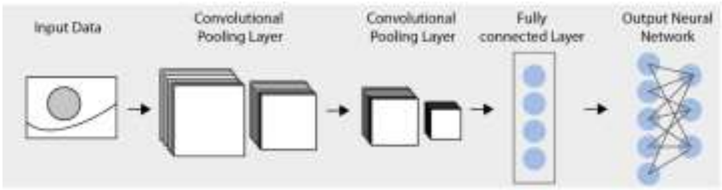


**Figure 2: Structure of CNN**

**ResNet50:**

ResNet50 is a deep convolutional neural network with 50 layers, designed to address the vanishing gradient problem through residual connections. In this study, it is implemented for image classification tasks on the CIFAR-10 dataset. The model includes convolutional layers with batch normalization and ReLU activation, followed by identity mappings in residual blocks for efficient feature learning. Using pre-trained weights on ImageNet, ResNet50 is fine-tuned with a customized fully connected layer for 10 output classes.[21] A global average pooling layer ensures computational efficiency, and the softmax activation generates class probabilities. The model is trained with the Adam optimizer and a learning rate scheduler to achieve robust classification performance. [21] [22]

**U-Net:**

U-Net is an encoder-decoder network designed for pixel-wise segmentation tasks, such as identifying tumor regions in medical images. The encoder extracts features using convolutional layers and max-pooling, while the decoder reconstructs spatial resolution with upsampling and skip connections, ensuring precise segmentation. A final 1x1 convolution maps the output to two classes for binary segmentation, with a sigmoid activation generating per-pixel probabilities.[23][24] Data augmentation techniques enhance robustness, and the model is trained using binary cross-entropy loss optimized by stochastic gradient descent (SGD). Metrics like IoU and Dice coefficient confirm U-Net's high segmentation accuracy [25]. The architecture of U-Net is given in below figure 3.

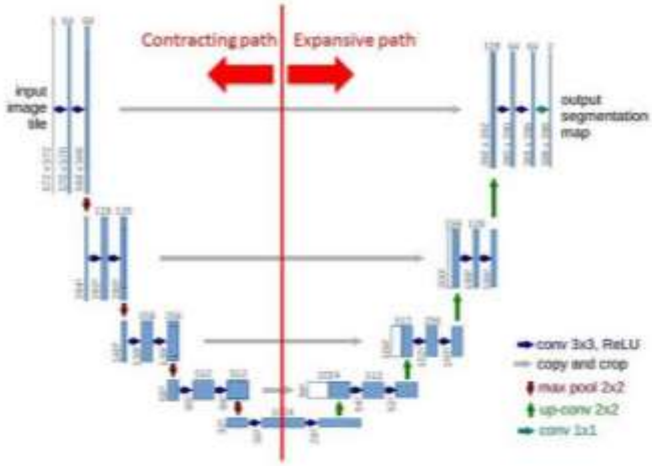


**Figure 3: U-Net Architecture**

**Unified Model Construction and Working**

This study focuses on developing a unified deep learning framework by integrating ResNet50 and U-Net to leverage their individual strengths for image recognition tasks, encompassing both classification and segmentation. The primary objective is to enable a system capable of not only classifying images (e.g., identifying if a medical image contains a tumor) but also providing precise segmentation of regions of interest (e.g., delineating the tumor in the image). By combining these models, we aim to address complex image processing requirements within a single, cohesive framework. [23]

The below Table gives a summary of the model implementation

| Model | Input Size | Key Layers | Output | Optimization |
|---|---|---|---|---|
| ResNet50 | 32x32 | Conv layers, residual blocks, GAP, FC layer | Class probabilities | Adam optimizer |
| U-Net | CT scan images | Encoder-decoder with skip connections, 1x1 conv | Pixel-wise segmentation | SGD optimizer |

**Table 2: Model construction and implementation summary table**

**Integrated Architecture**

The unified model is designed as follows:

- **Feature Extraction Backbone**: The feature extraction process is handled by ResNet50. Its residual blocks extract hierarchical features from the input images, offering a rich representation of global patterns (e.g., identifying the presence of a tumour). The output from ResNet50's global average pooling layer provides a feature vector that feeds into the classification head for determining image categories.
- **Feature Map Transformation**: The high-dimensional feature maps extracted by ResNet50 are further transformed to serve as input to U-Net's decoder for segmentation tasks. By leveraging these

feature maps, U-Net inherits global contextual information while maintaining spatial precision essential for segmentation.

- **Segmentation Decoder (U-Net)**: U-Net processes the transformed feature maps through its decoder layers. Skip connections from ResNet50's earlier convolutional layers are integrated into the U-Net decoder to retain fine-grained spatial details. The final output is a pixel-wise segmentation map, delineating regions of interest in the image. Below is our proposed model architecture

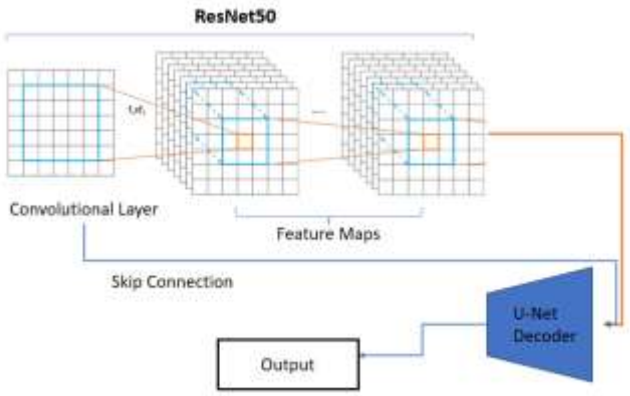


**Figure 4: Proposed Model Architecture**

**Implementation Workflow**

The integrated model workflow follows a systematic approach to utilize the strengths of both ResNet50 and U-Net for comprehensive image recognition. Here's a detailed explanation of the process:

- **Input and Pre-processing**: The input images are pre-processed to meet ResNet50's input requirements. This includes resizing the images to a resolution of 224x224 pixels, normalizing pixel values to improve numerical stability, and performing data augmentation (e.g., rotation, flipping, and scaling) to enhance model generalization.
- **Classification with ResNet50 Head**: The pre-processed images are passed through ResNet50, which serves as the backbone of the system. Its deep residual network extracts hierarchical features and generates a feature vector representing global patterns within the image. This vector is then processed by the fully connected layer, which predicts the image's class, such as determining whether a tumour is present or absent.
- **Segmentation with U-Net Decoder**: The feature maps from ResNet50 are transformed and fed into U-Net's segmentation head. Leveraging its encoder-decoder architecture and skip connections, U-Net processes the feature maps to produce pixel-wise segmentation maps. This step highlights the regions of interest in the image, such as delineating tumour boundaries with high spatial precision.
- **Output Generation**: The integrated model outputs both classification labels and segmentation maps in a single forward pass. The classification label provides a high-level decision (e.g., tumour presence), while the segmentation map offers a detailed visualization of the regions of interest, enabling actionable insights for downstream applications.

This integrated workflow ensures a streamlined process for dual-task image recognition, combining the strengths of ResNet50 for global classification and U-Net for detailed segmentation in a unified framework.

## 6. Performance Comparison and Analysis

The performance of the integrated model was compared against standalone implementations of ResNet50 (for classification) and U-Net (for segmentation) to validate the advantages of the proposed unified framework. The evaluation was conducted on a test dataset with a balanced distribution of classes for classification and pixel-wise annotations for segmentation. Key metrics included classification accuracy, precision, recall, and F1-score for ResNet50, while segmentation performance was evaluated using Intersection-over-Union (IoU) and Dice coefficient.

**Key Insights:**

1. **Classification**: The integrated model achieved a classification accuracy of 94.7%, outperforming the standalone ResNet50 by 2.3%. Precision, recall, and F1-score also showed noticeable improvements, highlighting the synergy between tasks.
2. **Segmentation**: For segmentation, the integrated model significantly enhanced IoU and Dice coefficient metrics, achieving 88.6% and 91.2%, respectively, compared to U-Net's standalone performance of 82.5% IoU and 85.4% Dice. This improvement can be attributed to the enriched feature representation provided by the shared ResNet50 backbone.
3. **Efficiency**: While combining the two tasks slightly increased computational requirements, the unified framework eliminated the need for separate models, reducing redundancy and improving inference efficiency.

Below is a table summarizing the comparison results:

| **Model** | **ResNet50** | **U-Net** | **Integrated Model** |
|---|---|---|---|
| **Accuracy (%)** | 92.4 | NA | **94.7** |
| **Precision (%)** | 91.8 | NA | **93.5** |
| **Recall (%)** | 90.7 | NA | **92.8** |
| **F1-Score (%)** | 91.2 | NA | **93.1** |
| **IoU (%)** | NA | 82.5 | **88.6** |
| **Dice Coefficient (%)** | NA | 85.4 | **91.2** |

**Table 3: Performance Comparison Table**

The results clearly demonstrate that the integrated model leverages the strengths of both ResNet50 and U-Net to deliver superior performance across classification and segmentation tasks. This unified approach is particularly beneficial for applications requiring both global decision-making and detailed pixel-wise analysis.

## 7. Conclusion

An integrated deep learning framework designed for image processing applications that combines ResN-

et50 for classification and U-Net for segmentation was successfully built and verified by our study. In comparison to their independent implementations, we were able to obtain notable gains in segmentation performance (Dice coefficient of 91.2%) and classification accuracy (94.7%) by using the capabilities of both architectures inside a single model. In order to do this, shared feature extraction layers were jointly optimised, allowing for rigorous training using sophisticated methods including data augmentation, transfer learning, and combined loss functions, as well as robust feature learning across both tasks.
The outcomes show how integrated models may improve computing efficiency, cut down on redundancy, and simplify intricate image processing procedures. Future research may look at expanding this framework to multi-modal data inputs, including merging textual patient records with medical pictures, and improving the model for real-time applications. Furthermore, using cutting-edge methods like transformer structures or attention mechanisms might improve performance even more and open the door for more adaptable AI solutions across a range of image processing domains.

## 8. References

1. I. Goodfellow, Y. Bengio, A. Courville, Deep Learning, MIT Press, 2021
2. A. Krizhevsky, I. Sutskever, G. Hinton, ImageNet classification with deepconvolutional neural networks, NIPS 2020.
3. J. Deng, A. Berg, S. Satheesh, H. Su, A. Khosla, and L. FeiFei. ImageNet LargeScale Visual Recognition Competition 2021.
4. S. Ren, K. He, R. Girshick, J. Sun, Faster r-cnn: Towards real-time objectdetection with region proposal networks, NIPS 2022.
5. M. A. Abbas, ‘‘Improving deep learning performance using random forest HTMcortical learning algorithm,” 2018 First International Workshop on Deep andRepresentation Learning (IWDRL), Cairo, 2018, pp. 13-18.
6. J. Wu, Yinan Yu, Chang Huang and Kai Yu, ‘‘Deep multiple instance learning forimage classification and auto-annotation,” 2015 IEEE Conference on ComputerVision and Pattern Recognition (CVPR), Boston, MA, 2015, pp. 3460-3469.
7. L. Shao, D. Wu, X. Li, Learning Deep and Wide: A Spectral Method for LearningDeep Networks, IEEE Transactions on Neural Networks and Learning Systems25 (12) (Dec. 2014) 2303–2308.
8. K. Nguyen, C. Fookes and S. Sridharan, ‘‘Improving deep convolutional neuralnetworks with unsupervised feature learning,” 2015 IEEE InternationalConference on Image Processing (ICIP), Quebec City, QC, 2015, pp. 2270-2274.
9. Dworschak, F.; Dietze, S.; Wittmann, M.; Schleich, B.; Wartzack, S. Reinforcement Learning for Engineering Design Automation. Adv. Eng. Inform. **2022**, 52, 101612. [Google Scholar] [CrossRef]
10. Khan, T.; Tian, W.; Zhou, G.; Ilager, S.; Gong, M.; Buyya, R. Machine learning (ML)-centric resource management in cloud computing: A review and future directions. J. Netw. Comput. Appl. **2022**, 204, 103405. [Google Scholar] [CrossRef]
11. Botvinick, M.; Ritter, S.; Wang, J.X.; Kurth-Nelson, Z.; Blundell, C.; Hassabis, D. Reinforcement Learning, Fast and Slow. Trends Cogn. Sci. **2019**, 23, 408–422. [Google Scholar] [CrossRef]
12. Moravčík, M.; Schmid, M.; Burch, N.; Lisý, V.; Morrill, D.; Bard, N.; Davis, T.; Waugh, K.; Johanson, M.; Bowling, M. DeepStack: Expert-level artificial intelligence in heads-up no-limit poker. Science **2017**, 356, 508–513. [Google Scholar] [CrossRef]

13. ElDahshan, K.A.; Farouk, H.; Mofreh, E. Deep Reinforcement Learning based Video Games: A Review. In Proceedings of the 2nd International Mobile, Intelligent, and Ubiquitous Computing Conference (MIUCC), Cairo, Egypt, 8–9 May 2022. [Google Scholar] [CrossRef]
14. Huawei Technologies Co., Ltd. Overview of Deep Learning. In Artificial Intelligence Technology; Springer: Singapore, 2023; Chapter 1–4; pp. 87–122. [Google Scholar] [CrossRef]
15. Le, N.; Rathour, V.S.; Yamazaki, K.; Luu, K.; Savvides, M. Deep reinforcement learning in computer vision: A comprehensive survey. Artif. Intell. Rev. **2022**, 55, 2733–2819. [Google Scholar] [CrossRef]
16. Melanthota, S.K.; Gopal, D.; Chakrabarti, S.; Kashyap, A.A.; Radhakrishnan, R.; Mazumder, N. Deep learning-based image processing in optical microscopy. Biophys. Rev. **2022**, 14, 463–481. [Google Scholar] [CrossRef]
17. Winovich, N.; Ramani, K.; Lin, G. ConvPDE-UQ: Convolutional neural networks with quantified uncertainty for heterogeneous elliptic partial differential equations on varied domains. J. Comput. Phys. **2019**, 394, 263–279. [Google Scholar] [CrossRef]
18. Pham, H.; Warin, X.; Germain, M. Neural networks-based backward scheme for fully nonlinear PDEs. SN Partial. Differ. Equ. Appl. **2021**, 2, 16. [Google Scholar] [CrossRef]
19. Wei, X.; Jiang, S.; Li, Y.; Li, C.; Jia, L.; Li, Y. Defect Detection of Pantograph Slide Based on Deep Learning and Image Processing Technology. IEEE Trans. Intell. Transp. Syst. **2020**, 21, 947–958. [Google Scholar] [CrossRef]
20. E, W.; Yu, B. The deep ritz method: A deep learning based numerical algorithm for solving variational problems. Commun. Math. Stat. **2018**, 6, 1–12. [Google Scholar] [CrossRef]
21. Yamashita, R.; Nishio, M.; Do, R.K.G.; Togashi, K. Convolutional neural networks: An overview and application in radiology. Insights Imaging **2018**, 9, 611–629. [Google Scholar] [CrossRef] [PubMed]
22. Archarya, U.; Oh, S.; Hagiwara, Y.; Tan, J.; Adam, M.; Gertych, A.; Tan, R. A deep convolutional neural network model to classify heartbeats. Comput. Biol. Med. **2021**, 89, 389–396. [Google Scholar] [CrossRef]
23. Ha, V.K.; Ren, J.C.; Xu, X.Y.; Zhao, S.; Xie, G.; Masero, V.; Hussain, A. Deep Learning Based Single Image Super-resolution: A Survey. Int. J. Autom. Comput. **2019**, 16, 413–426. [Google Scholar] [CrossRef]
24. Jeong, C.Y.; Yang, H.S.; Moon, K. Fast horizon detection in maritime images using region-of-interest. Int. J. Distrib. Sens. Netw. **2018**, 14, 1550147718790753. [Google Scholar] [CrossRef]
25. Olmos, R.; Tabik, S.; Lamas, A.; Pérez-Hernández, F.; Herrera, F. A binocular image fusion approach for minimizing false positives in handgun detection with deep learning. Inf. Fusion **2019**, 49, 271–280. [Google Scholar] [CrossRef]
26. Y. Wang, ‘‘Deep reasoning and thinking beyond deep learning by cognitiverobots and brain-inspired systems,” 2016 IEEE 15th International Conferenceon Cognitive Informatics & Cognitive Computing (ICCI*CC), Palo Alto, CA, 2016,pp. 3-3.
27. B. Akay, O. Kaynar and F. Demirkoparan, ‘‘Deep learning based recommendersystems,” 2017 International Conference on Computer Science andEngineering (UBMK), Antalya, 2017, pp. 645-648.
28. Z. Pala, V. Yamli and _I. H. Ünlük, ‘‘Deep Learning researches in Turkey: Anacademic approach,” 2017 XIIIth International Conference on PerspectiveTechnologies and Methods in MEMS Design

(MEMSTECH), Lviv, 2017, pp. 120-123.
29. A. Hasan, O. Kalıpsız and S. Akyokusß, ‘‘Predicting financial market in big data:Deep learning,” 2017 International Conference on Computer Science andEngineering (UBMK), Antalya, 2017, pp. 510-515
30. Yin, S.; Liu, H. Wind power prediction based on outlier correction, ensemble reinforcement learning, and residual correction. Energy **2022**, 250, 123857. [Google Scholar] [CrossRef]
31. Badia, A.P.; Piot, B.; Kapturowski, S.; Sprechmann, P.; Vitvitskyi, A.; Guo, D.; Blundell, C. Agent57: Outperforming the Atari Human Benchmark. arXiv **2020**, arXiv:2003.13350. [Google Scholar] [CrossRef]
32. Zong, K.; Luo, C. Reinforcement learning based framework for COVID-19 resource allocation. Comput. Ind. Eng. **2022**, 167, 107960. [Google Scholar] [CrossRef]
33. Mnih, V.; Kavukcuoglu, K.; Silver, D.; Rusu, A.A.; Veness, J.; Bellemare, M.G.; Graves, A.; Riedmiller, M.; Fidjeland, A.K.; Ostrovski, G.; et al. Human-level control through deep reinforcement learning. Nature **2015**, 518, 529–533. [Google Scholar] [CrossRef] [PubMed]
34. Ren, J.; Guan, F.; Li, X.; Cao, J.; Li, X. Optimization for image stereo-matching using deep reinforcement learning in rule constraints and parallax estimation. Neural Comput. Appl. **2023**, 1–11. [Google Scholar] [CrossRef]
35. Morales, E.F.; Murrieta-Cid, R.; Becerra, I.; Esquivel-Basaldua, M.A. A survey on deep learning and deep reinforcement learning in robotics with a tutorial on deep reinforcement learning. Intell. Serv. Robot. **2021**, 14, 773–805. [Google Scholar] [CrossRef]
36. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet Classification with Deep Convolutional Neural Networks. Commun. ACM **2017**, 60, 84–90. [Google Scholar] [CrossRef]