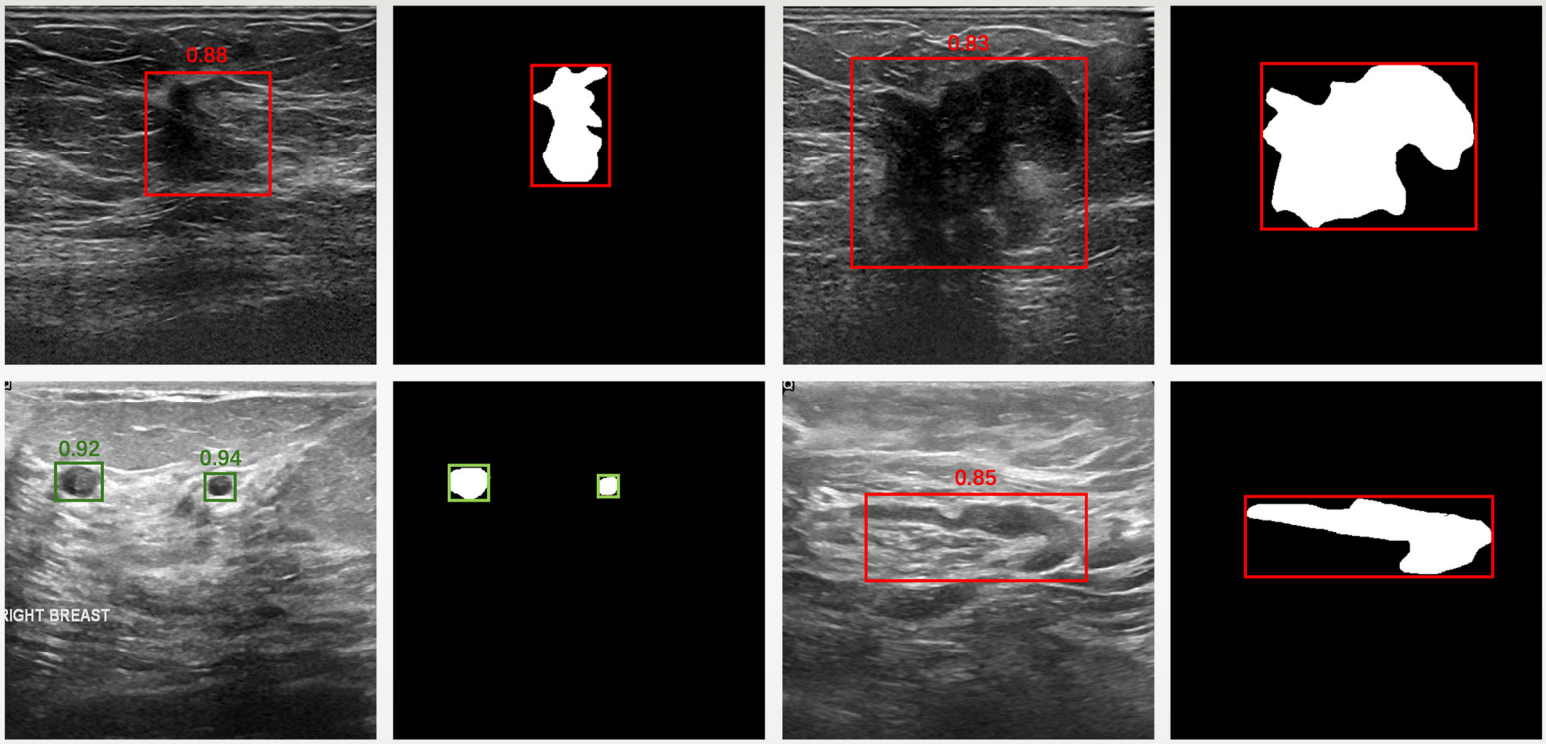
https://www.sciltp.com/journals/aim
Online ISSN: 2982-1711
Volume 1, Issue 1, 2024

AI Medicine



A Comparative Study of Deep Learning in Breast Ultrasound Lesion Detection: From Two-Stage to One-Stage, from Anchor-Based to Anchor-Free

Scilight

Contents

Vol.1 No.1 December 2024

AI Medicine
https://www.sciltp.com/journals/aim

Editorial

AI Medicine: Pioneering the Integration of Artificial Intelligence in Healthcare

Yu-Dong Yao
Department of Electrical and Computer Engineering, Stevens Institute of Technology, Hoboken, NJ 07030, USA; yyao@stevens.edu

How To Cite: Yao, Y.-D. *AI Medicine*: Pioneering the Integration of Artificial Intelligence in Healthcare. *AI Medicine* **2024**, *1*(1), 1. https://doi.org/10.53941/aim.2024.100001.

It is with great excitement and a sense of responsibility that I welcome our readers to the first issue of *AI Medicine*. As the Editor-in-Chief, I am privileged to steer this innovative journal towards excellence in the realm of artificial intelligence applications within medicine and healthcare.

1. Current Trends in AI and Medicine

We stand at a transformative juncture where artificial intelligence is reshaping the very fabric of healthcare. AI Medicine is launched at a time when interdisciplinary approaches combining AI, machine learning, and healthcare are not just experimental but imperative for progress.

The application of AI in medical fields such as radiology, pathology, and patient care management demonstrates the capacity of machine learning algorithms to aid and augment human expertise. The emergence of deep learning has revolutionized medical image analysis, providing unparalleled accuracy in diagnostics. Similarly, telehealth services are becoming increasingly sophisticated with AI-driven solutions for remote patient monitoring and diagnostics.

The role of AI is expanding rapidly, delving into areas of wearable technology, where intelligent analysis of data from devices assists in real-time health monitoring and preemptive healthcare strategies. The confluence of AI with genomics is paving the way for personalized medicine, tailoring treatments to the genetic makeup of individuals.

2. Scope of *AI Medicine*

AI Medicine commits to encapsulating this vast and evolving landscape of AI in healthcare. We aim to publish innovative research that not only advances the technology but also critically examines its implications in practice. Our focus spans a diverse spectrum, including but not limited to:

- **Medical Image Analysis:** Leveraging AI for diagnostic accuracy and efficiency;
- **Telehealth:** Innovations in remote care facilitated by AI technologies;
- **Wearable Medical Devices:** AI integration in wearable health monitoring systems;
- **Clinical Decision Support:** AI systems to support clinical decision-making processes;
- **Robotics in Medicine:** From surgical assistance to rehabilitation robotics;
- **Medical Data Analysis:** Data mining, predictive modeling, and interpretation in medical datasets;
- **Ethical Considerations:** Addressing the ethical, legal, and social implications of AI in healthcare.

This broad scope ensures that *AI Medicine* remains an authoritative source for the latest research and a forum for the rigorous debate of AI's role in health sciences.

3. For Authors, Reviewers, and Editors

AI Medicine's journey is one of collective effort and shared vision. The dedication of authors, diligence of reviewers, and guidance from our editorial board are the foundation of our journal's integrity.

We advocate a nurturing editorial process that aims to bring out the best in every submission. Constructive feedback, transparency in the review process, and a commitment to scholarly excellence are our guiding principles.

We stand for diversity in thought and inclusivity in participation, believing that groundbreaking ideas emerge from the confluence of varied perspectives.

4. Outlook

As we chart the course for AI Medicine, I envision a journal that not only disseminates pioneering research but also fosters a robust community of collaboration. Our commitment extends beyond publication to nurturing a dialogue that propels the field forward.

We invite researchers to present their innovative work, challenge the status quo, and contribute to a future where AI and medicine converge to enhance human health and wellbeing.

In closing, I extend my heartfelt thanks to everyone who has made AI Medicine a reality. Our shared voyage into the nexus of artificial intelligence and healthcare promises to be as inspiring as it is impactful.

Conflicts of Interest: The author declares no conflict of interest.

AI Medicine
https://www.sciltp.com/journals/aim

Review

A Short Survey on Computer-Aided Diagnosis of Alzheimer's Disease: Unsupervised Learning, Transfer Learning, and Other Machine Learning Methods

Si-Yuan Lu

School of Communications and Information Engineering, Nanjing University of Posts and Telecommunications, Nanjing 210003, China; 352888@njupt.edu.cn

How To Cite: Lu, S.-Y. A Short Survey on Computer-Aided Diagnosis of Alzheimer's Disease: Unsupervised Learning, Transfer Learning, and Other Machine Learning Methods. *AI Medicine* **2024**, *1*(1), 2. https://doi.org/10.53941/aim.2024.100002.

Received: 15 April 2024
Revised: 7 May 2024
Accepted: 14 May 2024
Published: 31 May 2024

Abstract: Alzheimer's Disease (AD) is a neurodegenerative disorder, which is irreversible and incurable. Early diagnosis plays a significant role in controlling the progression of AD and improving the patient's quality of life. Computer-aided diagnosis (CAD) methods have shown great potential to assist doctors in analyzing medical data, such as magnetic resonance images, positron emission tomography, and mini-mental state examination. Contributed by the advanced deep learning models, predictions of CAD methods for AD are becoming more and more accurate, which can provide a reference and verification for manual screening. In this paper, a short survey on the application of recent CAD methods in AD detection is presented. The advantages and drawbacks of these methods are discussed in detail, especially the methods based on convolutional neural networks, and the future research directions are summarized subsequently. With this survey, we hope to promote the development of CAD for early detection of AD.

Keywords: Alzheimer's disease; computer-aided diagnosis; magnetic resonance image; positron emission tomography; convolutional neural network

1. Introduction

Alzheimer's Disease (AD) is a progressive disorder of the neural system in humans, which accounts for about 80% of all dementia [1]. The main symptoms of AD are gradual memory decline, regression of cognitive function, language disorders, and changes in emotional personality [2]. The severity of these symptoms gradually intensifies as the disease progresses, and patients in the late stages of AD may even completely lose their self-care ability, fail to recognize family members, and ultimately die.

Currently, the exact cause of AD has not been elucidated, but studies suggest that various factors may be associated with the disease, such as genetic factors, abnormal protein deposition, and neurotransmitter imbalances. According to the progression of AD, it can be divided into three stages: mild, moderate, and severe, with a time-span of up to 10 years or more. Although current treatments cannot completely cure AD, early diagnosis is of great significance for delaying the progression of the disease and improving the quality of life of patients [3,4].

The diagnosis of AD primarily relies on neuropsychological assessments, blood tests, spinal fluid tests, and imaging examinations. Among these, Magnetic Resonance Imaging (MRI) is the most commonly used method for brain imaging in clinical settings. MRI images can be used to observe structural changes in the patient's brain and detect changes in brain volume in AD patients, such as atrophy of the hippocampus and temporal lobe cortex, ventricular enlargement, and white matter microlesions. However, manually analyzing high-dimensional brain MRI images is not only time-consuming but also requires specialized knowledge and extensive experience [5,6]. Moreover, manual analysis is highly subjective, and different doctors may provide different diagnostic results for the same set of images, leading to inconsistencies in the results.

Computer-aided diagnosis (CAD) is a method that uses computer algorithms and technology to assist doctors in disease diagnosis. With the development of artificial intelligence and particularly significant progress in computer vision and deep learning over the past decade, CAD applications in the medical field have become increasingly widespread and play an important role in the diagnosis of AD [7]. CAD leverages large amounts of case data and deep learning models to automatically analyze and judge the brain MRI images of suspected patients, for example, quantitatively analyzing the degree of atrophy in the hippocampus and brain volume on MRI [8]. This helps doctors make more accurate, reliable, and consistent diagnoses, reduces subjectivity, and improves the sensitivity and specificity of diagnosis.

The remainder of this review is organized as follows. Information for famous public AD datasets is discussed in section 2. Section 3 presents a comprehensive review of existing CAD methods for AD detection, including models by transfer learning, models trained from scratch, unsupervised models, and other related models. In section 4, the conclusions are summarized, and future research directions are given.

2. Public Datasets for AD

Public AD datasets are vital to train and validate deep models for early AD detection. In this section, three well-known datasets are discussed, including Alzheimer's Disease Neuroimaging Initiative, Open Access Series of Imaging Studies, and Australian Imaging, Biomarker & Lifestyle Flagship Study of Ageing.

- **Alzheimer's Disease Neuroimaging Initiative (ADNI)**: This initiative offers a comprehensive dataset comprising MRI and PET images, genetic data, and various biomarkers for AD. The dataset is designed to help researchers develop and validate advanced diagnostic tools and methodologies. Access is granted upon application approval through the ADNI website.
- **Open Access Series of Imaging Studies (OASIS)**: Focused on both normal aging and clinical populations, OASIS datasets include longitudinal MRI data across a broad age range. These datasets are freely available to the scientific community and can be accessed online without extensive application procedures.
- **Australian Imaging, Biomarker & Lifestyle Flagship Study of Ageing (AIBL)**: This study provides data on imaging, lifestyle, biomarkers, and the progression of AD, as well as healthy controls. Access to the data requires registration and approval.

2. CAD Methods for AD Classification

Generally, CAD methods for AD classification are based on either supervised learning or unsupervised learning. For supervised learning, the data samples are annotated and labeled, so the training is aimed at minimizing the error between the predictions of the deep model and the ground truth labels. On the other side, for unsupervised learning, ground truth labels are not available, so deep models are trained with proxy tasks, such as reconstruction, colorization, and contrastive learning. In this section, we will discuss these methods in detail.

2.1. *CAD Methods for AD Using Transfer Learning*

Transfer learning is the most popular approach for applying deep models in downstream tasks. With pre-trained weights, deep models can converge faster on medical datasets. Raza, et al. [9] leveraged the AlexNet to detect AD from normal control (NC). They used the ADNI and OASIS for training and testing. The accuracies were 98.74% and 95.93% for ADNI and OASIS, respectively. Puente-Castro, et al. [10] used a pre-trained ResNet as the backbone for representation learning. The representations were combined with the age and sex information of the subjects. Finally, an SVM was trained for multi-class classification. The accuracies were 86.81% and 78.64% for OASIS and ADNI, respectively. Ashraf, et al. [11] employed 13 different CNN models for AD detection using transfer learning, including AlexNet, DenseNet, ResNet, VGG, and SqueezeNet. They found that DenseNet outperformed other models with an accuracy of 99.05% on the MRIs from ADNI. Cilia, et al. [3] leveraged the handwriting data of the subjects to classify AD. They employed four models for feature learning, including ResNet-50, VGG-19, InceptionV3, and InceptionResNetV2. Data augmentation techniques were used to generate synthetic handwriting images for training. The deep features were combined with handcrafted features to train four traditional classifiers, including SVM, random forest, multi-layer perceptron, and k nearest neighbors. The best accuracy was 81.03%. Helaly, et al. [8] used a pre-trained VGG-19 as the backbone for AD classification. The pre-trained VGG-19 was fine-tuned on the 2D brain MRIs and achieved an accuracy of 97%. Loddo, et al. [12] utilized three pre-trained CNNs for AD detection in brain MRIs, including ResNet-101, AlexNet, and InceptionResNetV2. The three pre-trained models were fine-tuned on the MRIs, and their predictions were obtained by averaging across the three models. Their method was experimented on three datasets: OASIS, ADNI, and the Kaggle dataset, yielding an accuracy of over 98% for binary classification and multi-class classification.

A summary of the abovementioned methods is given in Table 1.

Table 1. CAD methods for AD classification using transfer learning.

Author	Model	Dataset	Result
Raza, et al. [9]	AlexNet	MRIs from ADNI and OASIS	The accuracies were 98.74% and 95.93% for ADNI and OASIS, respectively.
Puente-Castro, et al. [10]	ResNet and SVM	MRIs from ADNI and OASIS	The accuracies were 86.81% and 78.64% for OASIS and ADNI, respectively.
Ashraf, et al. [11]	AlexNet, DenseNet, ResNet, VGG, and SqueezeNet	MRIs from ADNI	The best accuracy was 99.05% by transferring DenseNet.
Cilia, et al. [3]	ResNet-50, VGG-19, InceptionV3, InceptionResNetV2, SVM, random forest, multi-layer perceptron, and k nearest neighbors	Private handwriting images	The best accuracy was 81.03%.
Helaly, et al. [8]	VGG-19	MRIs from ADNI	The model achieved an accuracy of 97%.
Loddo, et al. [12]	ResNet-101, AlexNet, and InceptionResNetV2	MRIs from OASIS, ADNI, and the Kaggle dataset	Their method yielded an accuracy of over 98% for binary classification and multi-class classification.

Note: CAD: Computer-Aided Diagnosis; AD: Alzheimer's Disease; MRI: Magnetic Resonance Imaging; ADNI: Alzheimer's Disease Neuroimaging Initiative; OASIS: Open Access Series of Imaging Studies.

2.2. CAD Methods for AD Trained from Scratch

Medical images vary significantly from natural images, so pre-trained weights cannot always work because of this gap between the source domain and the target domain. In addition, if the structure of the backbone model is modified or a new deep model is constructed, there are no pre-trained weights available. Therefore, training from scratch is preferred, which allows high flexibility in architecture design and customization for AD classification. Islam and Zhang [13] developed a CNN based on Inception-V4 for AD classification. The configurations of the original Inception-V4 were modified to fit the resolution of the MRI slices. In experiments, the Open Access Series of Imaging Studies (OASIS) dataset was employed for evaluation. Their model achieved an accuracy of 73.75%, which was not satisfactory. Bi, et al. [14] employed a CNN and a recurrent neural network (RNN) for feature extraction from the brain network generated from MRIs. An extreme learning machine (ELM) was trained to identify AD from mild cognitive impairment (MCI). They leveraged the brain MRIs from the AD neuroimaging initiative (ADNI) for evaluation. Traditional handcrafted features with the SVM classifier were implemented for comparison. The area under the curve (AUC) was chosen as the performance metric, and the best value was 84.7% for the classification of AD, MCI, and normal control (NC). Feng, et al. [15] designed a 3D-CNN to generate latent features from brain MRIs and PETs and developed a bi-directional long short-term memory (LSTM) structure for AD classification. Their model achieved an accuracy of 94.82% in recognizing AD versus NC. Hussain, et al. [16] suggested building a 12-layer CNN to classify AD in brain MRIs. In their experiments, pre-trained CNNs were leveraged using transfer learning for comparison, including MobilenetV2, VGG, InceptionV3, and Xception. Their 12-layer CNN outperformed the four models. Wang, et al. [4] used functional MRI time series data to detect AD. A CNN was trained to generate spatial representations, and an LSTM was implemented to get temporal information. Their model was evaluated on the ADNI dataset, and the accuracy was 71.76% for the classification of AD, MCI, and NC. Kundaram and Pathak [17] designed a deep CNN using 3 convolutional layers, 3 max-pooling layers, and 2 fully-connected layers. The network was trained on the brain MRIs and produced an accuracy of 87.72% for validation. Zhu, et al. [18] designed a Patch-Net to generate local representations from the brain MRIs. Then, an attention-based pooling block was developed for feature fusion. Fully-connected layers served for final predictions. The model was experimented on ADNI and AIBL datasets, and the best accuracy was 92.4% in distinguishing AD and NC. Alorf and Khan [19] developed two different

networks for AD classification using brain MRIs from the ADNI dataset. The first model was a stacked sparse autoencoder with softmax activation for classification. The second one was built upon a graph neural network, which exploits the connectivity of different brain regions. Their models were evaluated using the ADNI dataset, and the graph network outperformed with an accuracy of 84.03%. El-Sappagh, et al. [20] employed brain MRIs and time series data to detect AD and MCI and predict the conversion time. An LSTM and a feedforward neural network were combined and trained for classification and prediction. Results from the ADNI dataset revealed that their model produced an accuracy of 93.87%. Houria, et al. [21] used MRIs and diffusion tensor images (DTIs) to detect AD and MCI. They first developed a 2D-CNN structure to generate features from different images, and fused them. An SVM was trained as the classification model. The performance of the model was evaluated on the ADNI dataset, and satisfactory results were obtained.

A summary of the abovementioned methods is given in Table 2.

Table 2. CAD Methods for AD Classification Trained from Scratch.

Author	Model	Dataset	Result
Islam and Zhang [13]	CNN based on Inception-V4	MRIs from OASIS	The best accuracy was 73.75%.
Bi, et al. [14]	CNN, RNN, and ELM	MRIs from ADNI	The AUC for the 3-type classification was 84.7%.
Feng, et al. [15]	3D-CNN and LSTM	MRIs and PETs from ADNI	For AD and NC classification, the accuracy was 94.82%.
Hussain, et al. [16]	12-layer CNN	MRIs from OASIS	Their model achieved an accuracy of 97.75% for binary classification.
Wang, et al. [4]	CNN and LSTM	MRIs from ADNI	The accuracy was 71.76% for the classification of AD, MCI, and NC.
Kundaram and Pathak [17]	CNN	MRIs from ADNI	The model produced an accuracy of 87.72% for validation.
Zhu, et al. [18]	CNN with an attention mechanism	MRIs from ADNI and AIBL	The best accuracy was 92.4% in distinguishing AD and NC.
Alorf and Khan [19]	Stacked sparse autoencoder and graph neural network	MRIs from ADNI	The graph network achieved an accuracy of 84.03%.
El-Sappagh, et al. [20]	LSTM and feedforward neural network	MRIs and time series data from ADNI	Their model produced an accuracy of 93.87%.
Houria, et al. [21]	2D-CNN and SVM	MRIs from ADNI	The accuracy for CN and MCI classification was 97.00%.

2.3. CAD Methods for AD Using Unsupervised Learning

Unsupervised learning can learn patterns from data without label information, which is often used in medical applications because it is difficult to get labels without expertise. Ju, et al. [22] generated brain networks from the MRIs in the ADNI dataset and constructed an autoencoder for representation learning. The pre-training of the autoencoder was based on unsupervised learning, and the labels were used with a softmax output layer during fine-tuning. The autoencoder yielded an accuracy of 86.47% on the correlation coefficient data. Bi, et al. [23] utilized a PCANet to generate representations from the brain MRIs and used the k-means algorithm for classification. In the PCANet, convolutional layers and PCA operations were constructed. Therefore, the entire model can be trained by unsupervised learning. The average accuracy was 92.5% on the MRIs from the ADNI dataset. Jin, et al. [24] used a variational autoencoder as the encoder of the generative adversarial network for data augmentation. The reconstructed brain MRI and the original one were used to generate the residual image, which was fed into a multi-layer perceptron for AD classification. Cabreza, et al. [25] developed a generative adversarial network for detecting AD in brain MRIs. Their model was trained by unsupervised learning, and an anomaly score was proposed to classify the AD and NC samples. MRIs from OASIS were used for training and testing, and the accuracy of their method was 74.44%. Shi, et al. [26] proposed a generative adversarial network for segmentation of regions of interest for tau decomposition and AD classification in tau PET images. In the training of the model, multiple losses were used to achieve better generalization performance. The final AUC for binary classification was 92.9%. Zhang, et al. [27] developed a generative adversarial network with pyramid attention blocks to obtain more training PETs. The metabolic features in PETs were combined with MRIs for classifier training. For AD, MCI, and NC classification, the accuracy was 89.9%.

A summary of the abovementioned methods is given in Table 3.

Table 3. CAD methods for AD using unsupervised learning.

Author	Model	Dataset	Result
Ju, et al. [22]	Autoencoder	MRIs from ADNI	Based on the correlation coefficient data, the accuracy was 86.47%.
Bi, et al. [23]	PCANet and k-means	MRIs from ADNI	The average accuracy was 92.5%.
Jin, et al. [24]	Variational autoencoder, generative adversarial network, and multi-layer perceptron	MRIs from ADNI	The accuracy was 94%.
Cabreza, et al. [25]	Generative adversarial network	MRIs from OASIS	The overall accuracy was only 74.44%.
Shi, et al. [26]	Generative adversarial network with multiple losses	Tau PETs from ADNI	The final AUC for binary classification was 92.9%.
Zhang, et al. [27]	Generative adversarial network with pyramid attention blocks	MRIs and PETs from ADNI	For AD, MCI, and NC classification, the accuracy was 89.9%.

2.4. Other CAD Methods for AD

There are some AD detection methods based on traditional machine learning algorithms and networks other than CNNs or recurrent neural networks. For instance, Almubark, et al. [28] employed principal component analysis (PCA) with machine learning classifiers to detect AD from neuropsychological and cognitive data, including SVM, random forest, gradient boosting, and AdaBoost models. Uysal and Ozturk [29] attempted to diagnose AD based on hippocampal atrophy conditions. They segmented the brain MRIs to obtain the volume information of the hippocampal, which was fused with age and gender information. The SVM, Logistic regression, Gaussian naïve Bayes classifier, decision tree, random forest, and k-nearest neighbors were trained for identification of AD. The highest accuracy was 98% for AD and NC classification. Alvi, et al. [2] employed a gated-recurrent unit, a variant of the recurrent neural network to detect MCI using electroencephalography data. The electroencephalography data were pre-processed and segmented before feature extraction. Subsequently, a gated-recurrent unit was trained to identify MCI and NC. The experiment results showed that their method achieved an accuracy of 96.91%. Ilias and Askounis [30] proposed that transformer-based language models can be employed to detect AD in transcript data. The results were obtained on the ADReSS challenge dataset, and the model achieved an accuracy of 86.25% for multi-class classification. Meanwhile, they also analyzed the transcript and found out the words related to AD. Khan and Zubair [31] tried to detect AD using cognitive and demographic data from the ADNI dataset. Six different traditional machine learning classifiers were trained and compared in their experiments, and the best accuracy was 93.90%.

A summary of the abovementioned methods is given in Table 4.

Table 4. Other CAD methods for AD.

Author	Model	Dataset	Result
Almubark, et al. [28]	PCA with SVM, random forest, gradient boosting, and AdaBoost	Neuropsychological and cognitive data	The best accuracy was 91.08%.
Uysal and Ozturk [29]	SVM, Logistic regression, Gaussian naïve Bayes classifier, decision tree, random forest, and k nearest neighbors	MRIs from ADNI	The highest accuracy was 98% for AD and NC classification.
Alvi, et al. [2]	Gated-recurrent unit	Private electroencephalography data	The experiment results showed that their method achieved an accuracy of 96.91%.
Ilias and Askounis [30]	Transformer	ADReSS challenge dataset	The model achieved an accuracy of 86.25% for multi-class classification.
Khan and Zubair [31]	SVM, extreme Gradient Boosting, Logistic regression, naïve Bayes classifier, decision tree, and random forest	Cognitive and demographic data from ADNI dataset	The best accuracy was 93.90%.

3. Conclusion

This paper presents a comprehensive survey of CAD methods for AD detection. The review highlights the critical role of early diagnosis in managing AD progression and improving patient quality of life. In recent years, CAD methods utilizing advanced deep learning models have shown promising results in analyzing medical data such as MRI, PET, and cognitive assessments to aid in accurate diagnosis.

The CAD methods can be categorized into supervised learning, unsupervised learning, and other techniques. This study describes the application of pre-trained deep models like AlexNet, ResNet, and VGG in transfer learning, the development of custom CNN and RNN architectures for training from scratch. Unsupervised learning approaches, including autoencoders, generative adversarial networks, and PCA networks, are also explored for AD detection. Additionally, the use of traditional machine learning, transformer models, and other networks beyond CNNs in AD classification is discussed. The comparison of the three main methods is presented in Table 5.

Table 5. Comparison of three main methods.

Method	Advantages	Limitations
Transfer Learning	- Requires less training data - Faster convergence - Leverages pre-trained models to enhance feature extraction	- Potential for overfitting on small datasets - Dependent on the relevance of pre-trained model
Training from Scratch	- Customized to specific tasks - Full control over architecture	- Requires large datasets - Long training times
Unsupervised Learning	- No need for labeled data - Can discover unexpected patterns	- Less accurate than supervised methods - Complex interpretation of results

Despite the advancements in CAD methods, several challenges remain, including the need for larger and more diverse datasets, the incorporation of multimodal data, and improvements in model generalization. Future research directions should emphasize the importance of continued research to develop more accurate and robust CAD systems, leveraging advanced deep learning techniques and integrating multimodal data, to assist doctors in the early detection and diagnosis of AD. The application of CAD methods in clinical practice is yet to be achieved currently. This is because the CAD systems need to be subjected to rigorous regulatory approval processes, which can be lengthy and costly, especially for tools that use machine learning. Issues such as patient data privacy, consent for using patient data in training models, and the potential for bias in algorithmic decisions must be carefully managed. Moreover, clinicians may be skeptical of CAD systems, especially if they do not understand how decisions are made by the algorithms.

Funding

The paper is supported by Natural Science Research Start-up Foundation of Recruiting Talents of Nanjing University of Posts and Telecommunications (Grant No. XK0020923211).

Institutional Review Board Statement

Not Applicable.

Informed Consent Statement

Not Applicable.

Data Availability Statement

Not Applicable.

Conflicts of Interest

The authors declare no conflict of interest.

References

1. Mirzaei, G.; Adeli, H. Machine learning techniques for diagnosis of alzheimer disease, mild cognitive disorder, and other types of dementia. *Biomed. Signal Process Control* **2022**, *72*, 1–13. https://doi.org/10.1016/j.bspc.2021.103293.

2. Alvi, A.M.; Siuly, S.; Wang, H.; Wang, K.; Whittaker, F. A deep learning based framework for diagnosis of mild cognitive impairment. *Knowledge-Based Syst.* **2022**, *248*, 108815. https://doi.org/10.1016/j.knosys.2022.108815.
3. Cilia, N.D.; D'Alessandro, T.; De Stefano, C.; Fontanella, F.; Molinara, M. From Online Handwriting to Synthetic Images for Alzheimer's Disease Detection Using a Deep Transfer Learning Approach. *IEEE J. Biomed. Health Inform.* **2021**, *25*, 4243–4254. https://doi.org/10.1109/jbhi.2021.3101982.
4. Wang, M.; Lian, C.; Yao, D.; Zhang, D.; Liu, M.; Shen, D. Spatial-Temporal Dependency Modeling and Network Hub Detection for Functional MRI Analysis via Convolutional-Recurrent Network. *IEEE Trans. Biomed. Eng.* **2020**, *67*, 2241–2252. https://doi.org/10.1109/tbme.2019.2957921.
5. Yao, Z.; Mao, W.; Yuan, Y.; Shi, Z.; Zhu, G.; Zhang, W.; Wang, Z.; Zhang, G. Fuzzy-VGG: A fast deep learning method for predicting the staging of Alzheimer's disease based on brain MRI. *Inform. Sci.* **2023**, *642*, 1–9. https://doi.org/10.1016/j.ins.2023.119129.
6. Warren, S.L.; Moustafa, A.A. Functional magnetic resonance imaging, deep learning, and Alzheimer's disease: A systematic review. *J. Neuroimaging* **2022**, *33*, 5–18. https://doi.org/10.1111/jon.13063.
7. Hu, J.; Wang, Y.; Guo, D.; Qu, Z.; Sui, C.; He, G.; Wang, S.; Chen, X.; Wang, C.; Liu, X. Diagnostic performance of magnetic resonance imaging–based machine learning in Alzheimer's disease detection: A meta-analysis. *Neuroradiology* **2022**, *65*, 513–527. https://doi.org/10.1007/s00234-022-03098-2.
8. Helaly, H.A.; Badawy, M.; Haikal, A.Y. Deep Learning Approach for Early Detection of Alzheimer's Disease. *Cogn. Comput.* **2021**, *14*, 1711–1727. https://doi.org/10.1007/s12559-021-09946-2.
9. Raza, M.; Awais, M.; Ellahi, W.; Aslam, N.; Nguyen, H.X.; Le-Minh, H. Diagnosis and monitoring of Alzheimer's patients using classical and deep learning techniques. *Expert Syst. Appl.* **2019**, *136*, 353–364. https://doi.org/10.1016/j.eswa.2019.06.038.
10. Puente-Castro, A.; Fernandez-Blanco, E.; Pazos, A.; Munteanu, C.R. Automatic assessment of Alzheimer's disease diagnosis based on deep learning techniques. *Comput. Biol. Med.* **2020**, *120*, 1–7. https://doi.org/10.1016/j.compbiomed.2020.103764.
11. Ashraf, A.; Naz, S.; Shirazi, S.H.; Razzak, I.; Parsad, M. Deep transfer learning for Alzheimer neurological disorder detection. *Multimed. Tools Appl.* **2021**, *80*, 30117–30142. https://doi.org/10.1007/s11042-020-10331-8.
12. Loddo, A.; Buttau, S.; Di Ruberto, C. Deep learning based pipelines for Alzheimer's disease diagnosis: A comparative study and a novel deep-ensemble method. *Comput. Biol. Med.* **2022**, *141*. https://doi.org/10.1016/j.compbiomed.2021.105032.
13. Islam, J.; Zhang, Y. A Novel Deep Learning Based Multi-class Classification Method for Alzheimer's Disease Detection Using Brain MRI Data. In Proceedings of the International Conference on Brain Informatics 2017, Beijing, China, 16–18 November 2017.
14. Bi, X.; Zhao, X.; Huang, H.; Chen, D.; Ma, Y. Functional Brain Network Classification for Alzheimer's Disease Detection with Deep Features and Extreme Learning Machine. *Cogn. Comput.* **2019**, *12*, 513–527. https://doi.org/10.1007/s12559-019-09688-2.
15. Feng, C.; Elazab, A.; Yang, P.; Wang, T.; Zhou, F.; Hu, H.; Xiao, X.; Lei, B. Deep Learning Framework for Alzheimer's Disease Diagnosis via 3D-CNN and FSBi-LSTM. *IEEE Access* **2019**, *7*, 63605–63618. https://doi.org/10.1109/access.2019.2913847.
16. Hussain, E.; Hasan, M.; Hassan, S.Z.; Azmi, T.H.; Rahman, M.A.; Parvez, M.Z. Deep Learning Based Binary Classification for Alzheimer's Disease Detection using Brain MRI Images. In Proceedings of 15th IEEE Conference on Industrial Electronics and Applications, Kristiansand, Norway, 9–11 November 2020; pp. 1115–1120.
17. Kundaram, S.S.; Pathak, K.C. Deep Learning-Based Alzheimer Disease Detection. In Lecture Notes in Electrical Engineering, Proceedings of the Fourth International Conference on Microelectronics, Computing and Communication Systems, Ranchi, India, 11–12 May 2019; Springer Singapore: Singapore, Singapore, 2020; Chapter 50, pp. 587–597.
18. Zhu, W.; Sun, L.; Huang, J.; Han, L.; Zhang, D. Dual Attention Multi-Instance Deep Learning for Alzheimer's Disease Diagnosis With Structural MRI. *IEEE Trans. Med Imaging* **2021**, *40*, 2354–2366. https://doi.org/10.1109/tmi.2021.3077079.
19. Alorf, A.; Khan, M.U.G. Multi-label classification of Alzheimer's disease stages from resting-state fMRI-based correlation connectivity data and deep learning. *Comput. Biol. Med.* **2022**, *151*, 106240. https://doi.org/10.1016/j.compbiomed.2022.106240.
20. El-Sappagh, S.; Saleh, H.; Ali, F.; Amer, E.; Abuhmed, T. Two-stage deep learning model for Alzheimer's disease detection and prediction of the mild cognitive impairment time. *Neural Comput. Appl.* **2022**, *34*, 14487–14509. https://doi.org/10.1007/s00521-022-07263-9.
21. Houria, L.; Belkhamsa, N.; Cherfa, A.; Cherfa, Y. Multi-modality MRI for Alzheimer's disease detection using deep learning. *Phys. Eng. Sci. Med.* **2022**, *45*, 1043–1053. https://doi.org/10.1007/s13246-022-01165-9.

22. Ju, R.; Hu, C.; Zhou, P.; Li, Q. Early Diagnosis of Alzheimer's Disease Based on Resting-State Brain Networks and Deep Learning. *IEEE/ACM Trans. Comput. Biol. Bioinform.* **2019**, *16*, 244–257. https://doi.org/10.1109/tcbb.2017.2776910.
23. Bi, X.; Li, S.; Xiao, B.; Li, Y.; Wang, G.; Ma, X. Computer aided Alzheimer's disease diagnosis by an unsupervised deep learning technology. *Neurocomputing* **2020**, *392*, 296–304. https://doi.org/10.1016/j.neucom.2018.11.111.
24. Jin, S.; Zou, P.; Han, Y.; Jiang, J. Unsupervised detection of individual atrophy in Alzheimer's disease. In Proceedings of the 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), Virtual, 30 1–5 November 2021.
25. Cabreza, J.N.; Solano, G.A.; Ojeda, S.A.; Munar, V. Anomaly Detection for Alzheimer's Disease in Brain MRIs via Unsupervised Generative Adversarial Learning. In Proceedings of the 2022 International Conference on Artificial Intelligence in Information and Communication (ICAIIC), Jeju Island, Korea, 21–24 February 2022.
26. Shi, R.; Wang, L.; Jiang, J. An unsupervised region of interest extraction model for tau PET images and its application in the diagnosis of Alzheimer's disease. In Proceedings of the 44th Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), Glasgow, UK, 11–15 July 2022.
27. Zhang, M.; Sun, L.; Kong, Z.; Zhu, W.; Yi, Y.; Yan, F. Pyramid-attentive GAN for multimodal brain image complementation in Alzheimer's disease classification. *Biomed. Signal Process. Control* **2024**, *89*, 1–8. https://doi.org/10.1016/j.bspc.2023.105652.
28. Almubark, I.; Chang, L.-C.; Nguyen, T.; Turner, R.S.; Jiang, X. Early Detection of Alzheimer's Disease Using Patient Neuropsychological and Cognitive Data and Machine Learning Techniques. In Proceedings of IEEE International Conference on Big Data, Los Angeles, CA, USA, 9–12 December 2019; pp. 5971–5973.
29. Uysal, G.; Ozturk, M. Hippocampal atrophy based Alzheimer's disease diagnosis via machine learning methods. *J. Neurosci. Method.* **2020**, *337*, 108669. https://doi.org/10.1016/j.jneumeth.2020.108669.
30. Ilias, L.; Askounis, D. Explainable Identification of Dementia From Transcripts Using Transformer Networks. *IEEE J. Biomed. Health Inform.* **2022**, *26*, 4153–4164. https://doi.org/10.1109/jbhi.2022.3172479.
31. Khan, A.; Zubair, S. Development of a three tiered cognitive hybrid machine learning algorithm for effective diagnosis of Alzheimer's disease. *J. King Saud Univ.-Comput. Inf. Sci.* **2022**, *34*, 8000–8018. https://doi.org/10.1016/j.jksuci.2022.07.016.

AI Medicine
https://www.sciltp.com/journals/aim

Article

A State-of-the-Art Survey of Deep Learning for Lumbar Spine Image Analysis: X-ray, CT, and MRI

Ruyi Zhang [1,2]

[1] College of Medicine and Biological Information Engineering, Northeastern University, Chuangxin Road, Shenyang 110016, China; 2390160@stu.neu.edu.cn
[2] Research Institute for Medical and Biological Engineering, Ningbo University, Fenghua Road, Ningbo 315211, China

How To Cite: Zhang, R. A State-of-the-Art Survey of Deep Learning for Lumbar Spine Image Analysis: X-ray, CT, and MRI. *AI Medicine* **2024**, *1*(1), 3. https://doi.org/10.53941/aim.2024.100003.

Received: 17 April 2024
Revised: 12 June 2024
Accepted: 22 June 2024
Published: 17 July 2024

Abstract: Lumbar spine diseases not only endanger patients' physical health but also bring about severe psychological impacts and generate substantial medical costs. Reliable lumbar spine image analysis is crucial for diagnosing and treating lumbar spine diseases. In recent years, deep learning has rapidly developed in computer vision and medical imaging, with an increasing number of researchers applying it to the field of lumbar spine imaging. This paper studies the current state of research in deep learning applications across various modalities of lumbar spine image analysis, including X-ray, CT, and MRI. We first review the public datasets available for various tasks involving lumbar spine images. Secondly, we study the different models used in various lumbar spine image modalities (X-ray, CT, and MRI) and their applications in different tasks (classification, detection, segmentation, and reconstruction). Finally, we discuss the challenges of using deep learning in lumbar spine image analysis and provide an outlook on research and development prospects.

Keywords: deep learning; convolutional neural network; X-ray; computed tomography; magnetic resonance imaging

1. Introduction

Lumbar spine disease is one of the leading causes of disability worldwide [1], which includes degenerative diseases, inflammatory conditions, trauma, and tumors [2]. Not only do lumbar spine diseases cause severe physical pain, such as varying degrees of leg pain, weakness, and back pain [3], but also inflict significant psychological and emotional impacts, such as anxiety, depression, and social isolation often experienced by those suffering from chronic pain [4]. Long-term pain may also lead to dependency on pain management strategies, such as the prolonged use of painkillers [5]. Furthermore, lumbar spine diseases are among the major causes of work absenteeism and workers' compensation claims, reducing labor participation and increasing medical and social security costs, thus imposing a significant economic burden on individuals and nations [6]. Therefore, it is important to effectively diagnose and treat lumbar spine disease. With the continuous development of medical imaging technologies, including X-ray, computed tomography (CT), and magnetic resonance imaging (MRI), these imaging methods have become essential tools for diagnosis, treating, and prognosis prediction of lumbar spine diseases. Lumbar spine imaging provides valuable information about bones, joints, and surrounding soft tissues, helping doctors accurately diagnose spinal pathology [7]. Moreover, image-guided surgery, known for its precision and safety, is widely used in spinal surgical surgery [8]. Additionally, lumbar spine imaging also provides effective postoperative spinal assessments and care for healthcare providers [9]. Traditionally, lumbar spine images are visually observed and manually analyzed by radiologists based on their medical knowledge and experience. However, lumbar spine imaging still faces challenges such as limited contrast, insufficient spatial

resolution, and artifacts [10–12]. Hence, accurate evaluation requires extensive knowledge and experience, and training such experts takes a considerable amount of time.

To address these problems, researchers have proposed various methods to guide and assist doctors in lumbar spine image analysis. The commonly used techniques include digital image processing and machine learning methods, which often require manually designed feature extraction methods, making them time-consuming but also require expert knowledge [13]. Deep learning has achieved significant breakthroughs in computer vision, image processing, and analysis in recent years. Deep learning models allow for end-to-end training directly from raw data to learn outputs, automatically extracting features from large datasets without manual design or selection [14]. Furthermore, deep learning models can use pre-training and fine-tuning techniques to perform transfer learning between different but related tasks, effectively addressing the problem of scarce annotated data [15]. Due to these advantages, deep learning has achieved excellent results in lumbar spine image analysis.

Qu et al. [16] published a review on deep learning in spinal image analysis in 2022. They comprehensively introduced the application of deep learning in spinal image segmentation, detection and diagnosis. Lee et al. [17] published a review of deep learning for orthopedic diseases based on medical image analysis in 2022. They comprehensively introduced the application of deep learning in spinal image fractures, osteoarthritis, and joint-specific soft tissue diseases. This paper distinguishes itself from other surveys by providing a comprehensive review of the application of deep learning in lumbar spine image analysis across multiple imaging modalities, including X-ray, CT, and MRI. Unlike previous reviews that often focus on a single modality or specific task, our paper systematically covers deep learning techniques across different modalities and tasks, such as classification, detection, segmentation, and reconstruction. We have reviewed standard deep learning models based on task types and compiled the datasets available from the referenced papers. Additionally, we have summarized deep learning applications across different tasks based on imaging modalities. We have also discussed the optimization techniques and challenges deep learning technology faces in lumbar spine image analysis. Table 1 describes the coverage of this lumber spine image research survey paper, including image modalities and deep learning tasks.

Table 1. Lumber Spine Image Research Using Deep Learning.

Image Modality	Deep Learning Tasks
X-ray	Classification, detection, segmentation
CT	Classification, detection, segmentation, registration, reconstruction
MRI	Classification, detection, segmentation, reconstruction

Note: CT: computed tomography; MRI: magnetic resonance imaging.

The structure of this paper is organized as follows. Section 2 introduces deep learning methods and public datasets. Sections 3–5 discuss the specific applications of deep learning in various task types across different imaging modalities. Section 6 discusses key optimization methods and challenges affecting existing deep learning methods in the field of lumbar spine image analysis. Section 7 summarizes the advantages and future prospects of deep learning in the field of lumbar spine imaging.

2. Deep Learning Methods and Data

2.1. Classification Models

In 1998, LeCun et al. [18] introduced the LeNet-5 model, which was successfully applied to handwritten digit recognition (MNIST dataset), marked a breakthrough in the practical application of Convolutional Neural Network (CNN). CNNs can automatically learn features with spatial hierarchy from images by stacking convolutional layers, pooling layers, and fully connected layers. This feature learning method fr om local to global enables CNNs to excel in image classification tasks. In 2012, AlexNet [19] achieved overwhelming success in the ImageNet large scale visual recognition challenge (ILSVRC). AlexNet utilized ReLU activation functions, dropout regularization, and GPU acceleration, significantly improving classification accuracy. ResNet [20] addressed the difficulty of deep network training by introducing residual learning. It allows network layers to fit a residual mapping directly, rather than the mapping itself, enabling the network to improve performance by increasing depth without gradient vanishing or exploding issues. Following ResNet, the deep learning community has continuously explored classification models, including deeper and more complex network architectures (such as DenseNet [21], EfficientNet [22]), the introduction of attention mechanisms (such as the application of Transformer [23] in image classification applications), as well as model design optimized for specific tasks or efficiency.

2.2. Detection Models

Detection models are mainly divided into two categories two-stage models and one-stage models. Two-stage models are characterized by first generating candidate regions, and then classifying these regions and regressing their bounding boxes. Region-based Convolutional Neural Network (RCNN) [24] initially extracts candidate regions through a selective search algorithm, then uses CNN to extract features and then classifies them through the SVM classifier. Fast RCNN [25] achieved the sharing of feature extraction by introducing the region of interest (RoI) pooling layer. It inputs the entire image into a CNN to generate a feature map, and then extracts features for each candidate region from this shared feature map for classification and regression. Faster RCNN [26] introduced the region proposal network (RPN) for automatically generating high-quality candidate regions, further improving detection speed and accuracy.

One-stage models directly predict the category and location of objects on the image, omitting the generation step of candidate regions, and thus are usually generally faster. You only look once (YOLO) [27] treated the object detection task as a single regression problem, directly mapping from image pixels to bounding box coordinates and class probabilities. YOLOv2 [28] has made a number of improvements over YOLOv1, including the introduction of batch normalization, use of high-resolution classifiers for pre-training, and improved anchor mechanism. YOLOv3 [29] further improved the accuracy and speed of detection. It introduced multi-scale prediction and used a deep darknet as the feature extractor. Single Shot MultiBox Detector (SSD) [30] performed detection on feature maps of different scales, better-handling objects of various sizes.

2.3. Segmentation Models

Fully convolutional network (FCN) [31] was the first model to apply deep learning to semantic segmentation successfully. It transformed the fully connected layers in traditional convolutional neural networks into convolutional layers, enabling the network to accept input images of any size and output segmentation maps of corresponding dimensions. FCN is trained end-to-end, significantly improving the accuracy and efficiency of segmentation tasks. U-Net [32], by introducing skip connections, fuses feature maps from the encoder (downsampling) phase with those from the decoder (upsampling) phase, thereby preserving more contextual information. This design enables U-Net to perform exceptionally well on small sample datasets, especially in medical image segmentation. SegNet [33] uses pooling indices from the encoder phase for upsampling in the decoder phase, reducing the model's parameter count while improving segmentation accuracy. DeepLabv1 [34] introduced atrous convolution, which increases the receptive field size without adding parameters, enhancing segmentation precision. Building on v1, DeepLabv2 [35] introduced atrous spatial pyramid pooling (ASPP), further improving the model's ability to segment objects of different scales.

2.4. Evaluation Metrics

Evaluation metrics in deep learning are standards used to measure the performance of deep learning models, aiding in understanding how models perform on specific tasks. Different metrics are employed across various tasks and application scenarios to comprehensively represent a model's performance, enabling a more thorough comparison between models [36]. Typically, evaluation metrics can be defined using a confusion matrix, where true positive (*TP*) represents the number of samples correctly predicted as positive, false positive (*FP*) represents the number of samples incorrectly predicted as positive, true negative (*TN*) represents the number of samples correctly predicted as negative, and false negative (*FN*) represents the number of samples incorrectly predicted as negative.

Accuracy represents the proportion of samples that are correctly predicted by the model out of the total samples. It is the most fundamental metric for assessing model performance in balanced class situations.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Precision represents the proportion of actual positives among all samples predicted as positive by the model. A high precision means fewer false positives.

$$Precision = \frac{TP}{TP + FP}$$

Recall represents the proportion of samples predicted as positive by the model among all actual positives. A high recall means fewer false negatives. Recall is also known as sensitivity.

$$Recall = \frac{TP}{TP + FN}$$

F1-*Score* is the harmonic mean of precision and recall, used when both precision and recall are considered. The F1 score provides a single metric that balances precision and recall, also known as DSC (Dice Similarity Coefficient).

$$F1 - Score = \frac{2TP}{2TP + FP + FN}$$

Average precision (AP) is mainly used to evaluate the performance of models in classification and object detection tasks. In object detection, it specifically measures the precision performance of a model at different recall levels. The Precision-Recall Curve is drawn by calculating the model's precision and recall at different threshold settings. A high AP value indicates the model can detect positive objects with high precision while maintaining a high recall rate. AP is calculated for each class, and the average of all class AP values, known as mean average precision (mAP), indicates overall model performance.

Mean intersection over union (mIoU) is a common metric for evaluating model performance in image segmentation tasks. It calculates the average ratio of the intersection to the union of the predicted segmentation area and the actual segmentation area. Specifically, for a single class, IoU is calculated as follows.

$$IoU = \frac{TP}{TP + FP + FN}$$

After calculating *IoU* for all classes, mIoU is the average of these *IoU* values. This metric provides a way to quantify model accuracy in segmenting different classes. A higher mIoU value indicates better segmentation performance of the model.

Table 2 summarizes the evaluation metrics for deep learning models.

Table 2. Evaluation Metrics for Deep Learning Models.

Metric	Description	Formula	Application Tasks
Accuracy	Proportion of correctly predicted samples among the total samples	$\frac{TP + TN}{TP + TN + FP + FN}$	Classification, Detection
Precision	Proportion of true positives among all samples predicted as positive	$\frac{TP}{TP + FP}$	Classification, Detection
Recall (Sensitivity)	Proportion of true positives among all actual positives	$\frac{TP}{TP + FN}$	Classification, Detection
F1-Score	Harmonic mean of precision and recall	$\frac{2TP}{2TP + FP + FN}$	Classification, Detection
Average Precision (AP)	Average precision values at different recall levels	Calculated from the Precision-Recall Curve	Detection
Mean Average Precision (mAP)	Mean of AP values across all classes	Average of AP values for all classes	Detection
Intersection over Union (IoU)	Ratio of the overlap between predicted and actual segmentation areas to their union	$\frac{TP}{TP + FP + FN}$	Segmentation
Dice Similarity Coefficient (DSC)	Measure of overlap between the predicted and actual segments	$\frac{2 * TP}{2 * TP + FP + FN}$	Segmentation

Note: TP: true positive; TN: true negative; FP: false positive; FN: false negative.

2.5. Data

Data is crucial for deep learning models. Deep learning models rely on large data for training and validation. They learn features from the datasets through multi-level feature extraction to further enhance their generalizability and robustness, thereby enabling effective classification and prediction [37]. Conversely, when datasets are insufficient, models may only learn specific data features, leading to overfitting [38].

In the medical field, it is difficult to collect high-quality datasets. First, medical data involves patients' health information, which is subject to strict privacy protections and legal regulations, and current healthcare systems are yet to have the capability to provide the necessary protection for patient privacy [39]. Secondly, there are limitations in the cost and resources of collecting and processing medical data. Typically, experienced doctors are

needed to analyze and annotate the data, and annotating large datasets also incurs a significant time cost [40]. Additionally, collecting sufficient data for rare diseases is challenging due to the scarcity of cases [41].

To develop deep learning applications in the field of lumbar spine images, it is imperative to construct public datasets. Public datasets can, to some extent, compensate for the lack of data in private datasets, while improving the generalizability and robustness of models. Moreover, they provide a fair comparison of the performance of models trained on different datasets. Additionally, even images of the same type but different parts can assist model training through transfer learning. Al-kubaisi et al. [42] used MRI images of brain tumors to train a VGG model from scratch and used the transferred weights for training a classification task on lumbar MRI images. The results showed further improvement in model performance.

Table 3 shows the public datasets collected in the papers we reviewed. The Lumbar Spine MRI Dataset [43] is the most used dataset among the papers reviewed, with Masood et al. [44] using it for MRI image vertebral segmentation tasks, Liawrungrueang et al. [45] for MRI image disc detection tasks, and Le Van et al. [46] for simulating X-ray data for image classification tasks. Spineweb [47] is an online collaborative platform that includes 16 spine image datasets for various modalities and tasks. Scoliosis Test Dataset [48] is MICCAI 2019 Challenge dataset containing 98 X-ray images of the spine. VerSe2020 [49], VerSe2019 [50], xVertSeg Challenge [51] are CT spine image datasets from different challenges, containing 300, 160, and 25 images, respectively. BUU Spine Dataset [52] is a Burapha University dataset containing 400 labeled X-ray images of the spine.

Table 3. Lumbar Datasets.

Dataset	Image Type	Size	Labeled	Citation
Lumbar Spine MRI Dataset	MRI	515	No	[43]
Scoliosis Test Dataset	X-ray	98	No	[48]
BUU Spine Dataset	X-ray	400	Yes	[52]
VerSe2020	CT	300	Yes	[49]
VerSe2019	CT	160	Yes	[50]
xVertSeg Challenge	CT	25	Yes	[51]
Spineweb	X-ray/ CT/MRI	-	-	[47]

3. X-ray

3.1. Classification

Classification tasks based on deep learning are widely used in lumbar X-ray images and are used mainly for classifying various diseases, including spondylolisthesis, stenosis, osteoporosis, etc. Khare et al. [53] employed the DenseNet-201 model to predict vertebral slippage in the lumbar spine. In the preprocessing stage, they used contrast stretching to eliminate incorrect boundaries and adaptive histogram equalization to reduce the impact of image noise. In comparative experiments with four other models (LumbarNet, VGG19, AlexNet, and GoogleNet), the DenseNet-201 model achieved the highest classification accuracy. Varçin et al. [54] predicted lumbar spondylolisthesis through a deep learning system. The model first detected the L4, L5 vertebrae, and S1 sacrum using the YOLOv3 model, followed by the classification of lumbar spondylolisthesis through a fine-tuned MobileNet model.

Multiclass prediction tasks are also applicable to lumbar X-ray imaging. Sugiura et al. [55] used AlexNet to measure the tangential incident X-ray angles of the intervertebral disc space (IDS). They constructed a deep learning model using neural network console (NNC) and performed data augmentation and automatic model parameter selection through NNC. The study results demonstrated the effectiveness of deep learning in automatically classifying lumbar spine X-ray deflection angles, reducing patient burden, and improving imaging process efficiency. Nissinen et al. [56] analyzed and predicted pathological features in lumbar spine X-ray images using deep learning techniques, including scoliosis, instability, and fractures. They employed various visualization techniques to qualitatively evaluate the model's performance, including generating image heatmaps with gradient-weighted class activation mapping, indicating shapes and textures extracted by the network using the vanilla gradient method, rendering feature maps of individual input samples, and generating artificial input samples to visualize specific layers and kernels using activation maximization. Zhang et al. [57] proposed a DCNN model for osteopenia and osteoporosis screening. The model includes two channels for processing anteroposterior and lateral films and classifies patients from three sets of views: anteroposterior, lateral, and anteroposterior-lateral. Results indicate that the model can be effectively applied to identify osteopenia and osteoporosis in postmenopausal women. Table 4 summarizes the applications of deep learning models for classifying lumbar X-ray images.

Table 4. Deep Learning (DL) in Classification of Lumbar X-ray Images.

Target Class	Dataset Size	DL Model	Performance (%)		Paper List
			Accuracy	Recall	
Spondylolisthesis, and normal	299	VGG16	98	100	[58]
Anterior slippage, and normal	200	DenseNet-201	95.2	96.5	[53]
Stenosis, and normal	12442	VGG19	82.8	81.0	[59]
Scoliosis, and normal	598	DenseNet	93.5	97	[46]
Scoliosis, unreliability, and fracture	2949	CNN	94.1 (Scoliosis); 82.4 (Unreliability); 58.9 (Fracture)	70.5 (Scoliosis) 78.3 (Unreliability); 60.0 (Fracture)	[56]
Osteoporosis, and normal	162	CNN	100	100	[60]
Spondylolisthesis, and normal	272	GoogleLeNet	93.7	91.6	[61]
Osteoporosis, osteopenia, and normal	1616	DCNN	>72.6 (Osteoporosis) >78.7 (Osteopenia)	>68.4 (Osteoporosis) >81.8 (Osteopenia)	[57]
Five classes of deflection angle	500	AlexNet	83.0	83.0	[55]
Anteroposterior view, and Lateral view	1000	CNN	99.4	-	[62]
Spondylolisthesis, and normal	2707	MobileNet	99	98	[54]

Note: VGG: visual geometry group; CNN: convolutional neural network; DCNN: deep convolutional neural network.

3.2. Detection

Detecting vertebrae in lumbar spine images allows for rapid and effective localization of the vertebrae, enabling further analysis of parameters or diseases. An et al. [63] designed a novel landmark detection network for detecting lumbar vertebrae. The network is divided into two parts: first, the centers of the lumbar vertebrae and sacrum are detected based on Pose-Net, followed by the detection of landmarks on the lumbar vertebrae and sacrum using M-Net. In the first part, they proposed a random spinal incision enhancement technique to improve detection robustness, and in the second part, they enhanced detection accuracy through CoordConv and partial affinity fields. Nguyen et al. [64] used a deep learning system to detect keypoints on vertebral angles to calculate specific angles between vertebrae. First, a VGG model was trained to predict keypoints. Since the model did not perform well in cases of severe slippage in extension and bending between adjacent vertebrae, a second CNN regression model was subsequently used to predict the left and right boundaries of the vertebrae and align them with the center predictions of the first model. Experimental results indicate that this method is effectively applicable for Meyerding classification. Zhou et al. [65] developed a deep learning-based model for detecting the L5 vertebra and S1 sacrum to measure lumbar-sacral anatomical parameters further. Based on the EfficientDet model structure, local keypoints localization was enhanced with skip connection modules, and heatmap regression was used instead of direct coordinate regression.

In addition to vertebrae, automatic detection is also applicable to other lumbar spine structures. Sa et al. [66] automatically detected intervertebral discs based on Faster-RCNN. They conducted shallow and deep tuning of the model, specifically adjusting the last two and four layers, and evaluated the performance changes through smooth L1 Loss. Experimental results indicated that fine-tuning deeper layers of the model results in better detection performance. Table 5 summarizes the applications of deep learning models for detecting lumbar X-ray images.

Table 5. Deep Learning in Detection of Lumbar X-ray Images.

Target Class	Dataset Size	DL Model	Performance (%)		Paper List
			Accuracy	AP	
Vertebrae	1524	Pose-net, M-Net	98.38	-	[63]
Vertebrae	1000	SSD, MobileNet	95.6 (AP) 93.5 (LA)	-	[62]
Vertebrae	100	CNN	99.7	-	[67]
Vertebrae	1000	VGG, CNN	-	-	[64]
Intervertebral discs	1082	Faster-RCNN	-	90.5	[66]
L5 vertebra and S1 sacrum	1791	EfficientDet	>90	-	[65]
L4, L5 vertebra and S1 sacrum	2707	YOLOv3	-	-	[54]

Note: SSD: single shot multiBox detector.

3.3. Segmentation

Automatic segmentation of the lumbar vertebrae can further assist doctors in accurately measuring structural parameters, or further predicting disease states, thus improving their work efficiency. Kim et al. [68] combined deep learning techniques and level set methods to segment the lumbar vertebrae. First, the five lumbar vertebrae were located using Pose-net, followed by segmentation of the located vertebrae through M-net. The level set

method was used for fine-tuning the results segmented by M-net. Trinh et al. [69] designed the LumbarNet model for segmenting the lumbar vertebrae and sacrum. Based on the U-net structure, they added a feature fusion module (FFM) to the encoder module to enhance the encoder's efficiency. After obtaining the segmentation results, they calculated the P-grade of the vertebrae based on pedicle slope detection (PSD) and dynamic shift (DS) to determine the presence of lumbar spondylolisthesis.

For more complex structural analysis requirements, Chen et al. [70] used the scSE U-net model to segment various anatomical features of the lumbar spine, such as the lumbar vertebrae, pelvis, spinous processes, and intervertebral foramina. This model implements spatial and channel squeeze & excitation (scSE) blocks in the U-net structure, which recalibrate the feature maps along spatial and channel dimensions, respectively. The model includes two U-shaped networks, the first for segmenting anatomical features and the second for identifying them. Tran et al. [71] designed MBNet for lumbar spine segmentation and prediction of related parameters. This model includes two branches. The first branch performs semantic segmentation of the vertebrae using BiLuNet, which is based on an improved U-Net, and the second branch calculates relevant parameters based on the segmentation results to assist doctors in diagnosing low back pain. Table 6 summarizes the applications of deep learning models for the segmentation of lumbar X-ray images.

Table 6. Deep Learning in Segmentation of Lumbar X-ray Images.

Target Class	Dataset Size	DL Model	Performance (%)		Paper List
			mIoU	DSC	
Multiple anatomical features of the lumbar spine	2782	U-net	-	91 (AP) 87 (LA) 80 (OP)	[70]
Vertebrae	797	Pose-net, M-Net	-	91.6	[68]
Vertebrae	830	Comprehensive	-	-	[72]
Vertebrae, sacrum, and femoral heads	750	U-net	85.0	-	[71]
Vertebrae, and sacrum	706	U-net	88	-	[69]
Vertebrae, and sacrum	780	U-Net	-	82.1	[73]
Vertebrae, sacrum, and femoral heads	1000	ResNet	88.5	-	[74]
Vertebrae	2073	U-Net	-	>94	[75]

4. CT

Automatic classification for lumbar CT images is primarily used for gender classification and bone mineral density (BMD) prediction. Malatong et al. [76] applied a deep learning model to classify gender based on the upper and lower endplates of the L3 lumbar vertebra. They adjusted the last two layers of GoogLeNet, including modifying the parameters of the fully connected layer and replacing the new classification layer. Random rotations, reflections, and horizontal translations were employed during training to prevent model overfitting. Yasaka et al. [77] predicted lumbar spine BMD using deep learning techniques. They trained the model using the L2-L4 vertebrae of patients and tested it using the L1 vertebra. Finally, the BMD prediction results were used to assess whether patients had osteoporosis.

Thoracic and lumbar spine injuries pose significant risks to human health. Automated vertebra detection can effectively locate the vertebrae and predict the damage and severity simultaneously. Doerr et al. [78] used the Faster R-CNN model to locate the lumbar spine, and simultaneously perform a five-category classification of thoracolumbar injury classification and severity score (TLICS) morphology types and binary classification of posterior ligamentous complex (PLC) integrity scores. They trained two models for the two respective localization and classification tasks. Research findings showed that deep learning methods effectively predict PLC and morphological components of TLICS.

Accurate vertebrae segmentation from CT images is important for many tasks, including vertebral morphological analysis and disease prediction. Lu et al. [79] designed a deep learning-based 3D multi-scale spinal segmentation method. First, the lumbar spine was located and cropped using U-Net, followed by 3D vertebral segmentation using XUNet. XUNet incorporated inception blocks for feature extraction, aggregating features across different semantic scales and improving the network’s expressive ability. Malinda et al. [80] proposed a hybrid deep segmentation generative adversarial network for lumbar image segmentation. To increase data usability, they improved the training scheme on the CycleGAN model, combining paired and unpaired training data.

Image-guided surgery is now widely applied in spinal surgery, and image registration allows surgeons to observe real-time changes during surgery better. Gao et al. [81] registered lumbar vertebrae using a deep learning model. They proposed an end-to-end framework named ACSGRegNet, which is mainly divided into two parts. The first is an affine registration network to calculate affine transformation parameters. The second is a deformable

registration network, which includes self-attention modules, cross-attention modules, and gated fusion modules to output the final dense deformation field.

Image reconstruction for CT images can reduce noise and improve image quality, obtaining high-quality CT images with reduced radiation doses, and enabling conversions between CT images and other image types. Greffier et al. [82] used both deep learning and hybrid iterative reconstruction algorithms for image reconstruction. Through quantitative analysis of image quality and dose, it was verified that the deep learning reconstruction algorithm can optimize the CT dose plan. Morbée et al. [83] reconstructed CT images from MRI images based on deep learning methods and compared them with traditional CT images, demonstrating their equivalence. Yeoh et al. [84] applied a deep learning reconstruction algorithm to low-dose CT images. The experimental results from the quantitative and qualitative analysis showed that this method could achieve both image denoising and edge-sharpening effects. Table 7 summarizes the applications of deep learning models for lumbar CT images.

Table 7. Deep Learning for Lumbar CT Images.

Task	Target Class	Dataset Size	DL Model	Performance (%)	Paper List
Classification	Female, and male	1100	GoogLeNet	Accuracy = 92.5	[76]
	Female, and male	117	LeNet5	Accuracy = 86.4	[85]
	BMD	1665	CNN	PCCs > 84.0 (p < 0.001)	[77]
Detection	Vertebrae	111	Faster R-CNN	DSC = 92 (morphology), 88 (PLC)	[78]
Segmentation	Vertebrae	522	CNN	DSC > 90	[86]
	Bone, disc, and nerve	1681	U-net	DSC = 94 (Bone), 92 (Disc), 92 (Nerve)	[87]
	Vertebrae	656	U-net	DSC > 88.8	[79]
	Vertebrae	8040	CycleGAN	DSC = 94.2	[80]
Registration	Vertebrae	15	FCN	DSC = 95.77	[88]
	Vertebrae	61	CNN	DSC = 96.3	[81]
Reconstruction	Vertebrae	3	Integrated	Noise magnitude < i4	[82]
	Vertebrae	30	Integrated	Quantitative image noise analysis	[89]
	Full image	30	Integrated	Bland Altman analysis	[83]
	Vertebrae	52	Integrated	Quantitative image noise analysis	[84]

Note: FCN: fully convolutional network.

5. MRI

5.1. Classification

Spinal stenosis and disc herniation are among the causes of lower back pain (LBP) and are two of the most common lumbar disorders. This task is typically performed by radiologists or orthopedic doctors through imaging analysis. Al-kubaisi et al. [42] used a deep learning model to classify lumbar disc status as normal or abnormal. They analyzed the impact of transfer learning and model fine-tuning on image classification through comparative experiments, including training with ImageNet images and brain tumor MRI images, and incorporated Grad-CAM visualization technique to explain the model. Experimental results showed that transfer learning using datasets from the same field could improve model performance and mitigate the effects of dataset limitations.

Grading specific diseases is also a typical application of deep learning in lumbar MRI images. Chen et al. [90] designed an auxiliary diagnostic system for lumbar disc herniation (LDH) based on the CDCGAN model, capable of outputting six indicators for quantitative analysis of MRI images. In the model, they combined Tanh and ReLU activation functions to enhance the model's classification efficiency. Cheung et al. [91] assessed lumbar disc degeneration using a deep learning model. They employed the integrated MRI-SegFlow and visual geometry group-medium (VGG-M) to predict Schneiderman scores, disc bulging, and Pfirrmann grading. Experimental results demonstrated that deep learning models could be effectively applied in lumbar disc degeneration (LDD) prediction tasks. Table 8 summarizes the applications of deep learning models for classifying lumbar MRI images.

Table 8. Deep Learning in Classification of Lumbar MRI Images.

Target Class	Dataset Size	DL Model	Performance (%)		Paper List
			Accuracy	Recall	
Normal, and abnormal	1448	VGG	87.91	> 90.91	[42]
Six indexes of lumbar disc herniation	-	CDCGAN	-	-	[90]
Four classes of Schneiderman score; disc bulging, and normal; Five classes of Pfirrmann grade	2686	CNN	90.2 (Schneiderman) 90.4 (Disc bulging) 89.9 (Pfirrmann)	96.0 (Schneiderman) 76.5 (Disc bulging) 60.4 (Pfirrmann)	[91]
Five classes of Pfirrmann grade	2500	CNN	86	-	[92]

Table 8. *Cont.*

Target Class	Dataset Size	DL Model	Performance (%)		Paper List
			Accuracy	Recall	
Five classes of Pfirrmann grade; four classes of spondylolisthesis; four classes of central canal stenosis	882	SpineNet	-	-	[93]
Three classes of foraminal stenosis severity	22796	ResNet	>80	-	[93]

5.2. Detection

Vertebral detection tasks include center localization and candidate bounding box localization. Deep learning-based vertebral image detection can provide doctors with effective localization of vertebral segments or disease areas. Zhou et al. [94] designed a deep learning method to detect and locate the L1-S1 lumbar vertebrae. The proposed method includes two phases of image detection: the first detects the S1 vertebra, and the second detects the L1-L5 vertebrae. The detection model is trained only on public datasets and does not require annotated MRI images as a training set. Compared to other deep learning methods, this model learns the similarities between vertebrae. Mushtaq et al. [95] combined the YOLOv5 and HED U-Net models to detect and diagnose the lumbar spine. First, YOLOv5 is used to detect the vertebrae, then L1, L5, and S1 are extracted from the detection results to calculate the lumbar lordosis angle (LLA) using L1 and S1, and the lumbosacral angle (LSA) using L5 and S1.

To detect more vertebral structures, effective diagnosis of diseases should be pursued, including lumbar disc herniation and intervertebral disc degeneration. Tsai et al. [96] used deep learning to detect lumbar disc herniation. Due to a small training set size, they used data augmentation methods such as rotation, contrast, and brightness adjustments and employed multiple strategies to expand the volume and features of images. The model can detect abnormalities in the lumbar, sacral, and fifth lumbar vertebral regions. Yi et al. [97] used deep learning models to detect degenerative cervical diseases. They trained two modified 3D Resnet18 networks, one for sagittal view MR images and the other for axial view MR images. A multi-modal cross-attention module from Transformer was introduced in the models, and AdamW was used as the optimizer. Table 9 summarizes the applications of deep learning models for detecting lumbar MRI images.

Table 9. Deep Learning in Detection of Lumbar MRI Images.

Target Class	Dataset Size	DL Model	Performance (%)			Paper List
			Accuracy	Precision	mAP	
Vertebrae	903	CNN	>99.3	>99.6	-	[98]
Disc	1000	YOLOv5	95	-	-	[45]
Vertebrae, sacrum, disc	714	YOLOv3	81.1	87.2	-	[96]
Vertebrae, disc	804	Resnet18	-	>73.7	-	[97]
Disc	80	Faster RCNN	96.25	-	-	[99]
Vertebrae	2739	CNN	98.6	98.9	-	[94]
Vertebrae	575	YOLOv5	-	-	95.2	[95]

5.3. Segmentation

Automatic segmentation of MRI lumbar spine images can help doctors more accurately identify the different structures of the lumbar spine, while also helping doctors reduce diagnosis time and improve diagnosis efficiency. Li et al. [100] used deep learning methods to segment the spine in MRI images, including vertebrae, laminae, and the dural sac. They introduced a multi-scale attention mechanism based on the U-Net model, where the upsampling and downsampling convolutional layer structures were replaced with a convolutional layer and a dual-branch multi-scale attention module, enhancing the model's segmentation efficiency. Masood et al. [44] designed a deep learning model to segment vertebrae in images to further assess spinal spondylolisthesis and lumbar lordosis. They customized an algorithm (VBSeg) in the machine learning field for comparison with deep learning methods, and combined various models in the deep learning approach to configure the encoder-decoder setup for optimal results. Zheng et al. [101] used a deep learning model to segment specific structures according to the Pfirrmann grading, covering 5 types across 14 regions. The proposed BianqueNet architecture, built on DeepLabv3+, incorporated a swin Transformer with skip connection modules. Compared to traditional Transformer modules, this module uses a moving window mechanism, which is more efficient in network computation.

For 3D segmentation of MRI images, Chen et al. [102] used a 3D-UNet model to segment the L4-5 spinal structures to reconstruct a 3D lumbar intervertebral foramen (LIVF) model. After obtaining the measurement results, further calculations were made on the morphological parameters of the LIVF, including the foramen area, height, and width. Experimental results showed that the model could be effectively applied to MRI spinal structure

tasks, and based on the segmentation results, it could generate complete and accurate 3D LIVF models. Table 10 summarizes the applications of deep learning models for segmenting lumbar MRI images.

Table 10. Deep Learning in Segmentation of Lumbar MRI Images.

Target Class	Dataset Size	DL Model	Performance (%)		Paper List
			MIoU	DSC	
Vertebrae, and discs	300	U-Net	94.7 (Vertebrae) 92.6 (Discs)	-	[103]
Vertebral body, lamina, and dural sac	1080	CNN	-	92.52	[100]
Vertebrae, and sacrum	22796	U-Net	-	93	[104]
Vertebrae	514	ResNet, UNet	86	97	[44]
Discs	382	VGG 16	93.3	-	[105]
Vertebrae, sacrum, presacral fat area, cerebrospinal fluid area and IVDs	>1000	DeepLabv3+	90.35	94.70	[101]
Vertebrae	1360	U-Net	>74.4	>84.9	[106]
L4-5 spine structures	100	U-Net	-	91.8	[102]
L5/S1 bone structures, and discs	100	U-Net	-	>90.39	[107]

Note: MIoU: mean intersection over union; IVD: intervertebral disc; DSC: dice similarity coefficient.

5.4. Reconstruction

Deep learning techniques for MRI image reconstruction can accelerate imaging speed and enhance image quality. Chazen et al. [108] validated the effectiveness of image reconstruction from image evaluation and statistical analysis. In image evaluation, they graded overall image clarity on a 3-point scale, motion artifacts on a 4-point scale, and used multi-planar reconstruction (MPR) to grade foraminal stenosis. Fujiwara et al. [109] validated the effectiveness of rapid image reconstruction through statistical analysis, including Cohen's kappa statistic, and the interchangeability between the rapid reconstruction protocol and traditional protocols. Han et al. [110] analyzed reconstructed images using deep learning quantitatively. They employed two convolutional neural networks incorporating the 2D V-Net architecture; the first network segmented the intervertebral discs to calculate disc height, while the second network segmented the vertebral bodies to calculate vertebral volume. To validate their effectiveness, Zerunian et al. [111] performed noise analysis on reconstructed images. They measured signal intensity to calculate signal-to-noise ratio (SNR) and contrast-to-noise ratio (CNR), and used a five-point Likert scale to assess image quality for qualitative analysis. Gao et al. [112] trained a ResNet model to denoise MRI images to remove Rician noise. They compared the model's denoising results with the weighted stable matching (WSM) algorithm and denoising CNN (DnCNN) algorithm, verifying the model's reliability on MRI lumbar spine images. Table 11 summarizes the applications of deep learning models for reconstructing lumbar MRI images.

Table 11. Deep Learning in Reconstruction of Lumbar MRI Images.

DL Model/Software	Patient Number	Quantitative Analysis Indicators	Paper List
AIR Recon DL	35	Cohen's kappa statistic	[108]
Advanced Intelligent Clear-IQ Engine	58	Cohen's kappa statistic	[109]
AIR Recon DL	18	Disc heights and vertebral body volumes	[110]
AIR Recon DL	35	Conger's kappa statistic	[113]
AIR Recon DL	80	Quantitative image noise analysis	[111]
ResNet	127	Quantitative image noise analysis	[112]

6. Discussions

From the papers reviewed, we can find that deep learning has been extensively used across various fields of lumbar spine image analysis, including the diversity of image modalities and the variety of processing methods, with some research results already reliably applied in clinical applications. Compared to traditional algorithms, deep learning stands out with its robust feature extraction capabilities, multi-level abstraction, and excellent flexibility and universality in image analysis tasks. A wide range of image preprocessing methods have significantly contributed, including data augmentation [54, 75, 96], and image quality enhancement [53]. Tsai et al. [96] employed image rotation and adjusted brightness and contrast to enhance MRI images, achieving an 86.2% accuracy in LHD detection with just 350 original images. Transfer learning offers another effective way to enhance performance by using pre-trained models on other datasets as a starting point and further fine-tuning them to align closely with specific task requirements, thus addressing tasks with fewer samples. In Al-kubaisi et al.'s research [42], the fine-tuned VGG16 model's classification performance on an MRI dataset increased from 78.2% to 87.91%. Additionally, appropriate modifying network structures [70,100,101], improving activation functions [90,97], and post-processing of model results [68,75] can all effectively enhance the overall performance of tasks.

Different deep learning models exhibit various strengths and weaknesses across different imaging modalities and tasks. CNNs, such as LeNet, AlexNet, VGG, and ResNet, are widely used in classification, detection, and segmentation tasks for X-ray, CT, and MRI images due to their powerful feature extraction capabilities, although they require significant computational resources. R-CNNs and its variants excel in object detection tasks in CT and MRI images with high accuracy but at the cost of higher computational demands. Single-stage detectors like YOLO and SSD are favored for real-time applications in X-ray and CT images, offering faster detection speeds with slightly lower accuracy. FCN and U-Net are highly effective for segmentation tasks, particularly in MRI images, but depend heavily on high-quality annotated data. GANs are useful for data augmentation and image reconstruction, producing high-quality synthetic images, though their training can be unstable and complex to tune.

Deep learning in lumbar spine image processing still faces shortcomings and challenges. First, compared to other body parts like the breast [114] and heart [115], lumbar spine images lack sufficient public datasets. Although data augmentation can somewhat mitigate this issu, there is still a gap between the training performance of models and their potential maximum performance. Therefore, establishing high-quality public datasets is necessary. Secondly, deep learning has poor generalization ability. Models trained solely on data from a single field often fail to generalize when applied to other fields [116], and lumbar spine images often show significant variation across different modalities or even within the same modality under different acquisition devices. With the continuous development of large-scale pre-trained models [117] in recent years, this issue might be addressed. Moreover, the substantial computational resources required for processing and analyzing lumbar images also hinder the widespread application of deep learning in practical settings. Although deep learning models have shown potential in diagnosing and predicting lumbar diseases, the ability to process large volumes of patient data in real-time, and the demand for computational resources by these models, remain practical challenges that need to be overcome. In some studies [54,62], the application of compact networks like MobileNet [118] has been able to mitigate the impact of these issues.

With the advancement of data sharing and privacy protection technologies, public datasets of lumbar spine images are expected to become more abundant. This will help enhance the training effectiveness and generalizability of deep learning models. Additionally, developing techniques such as transfer learning and self-supervised learning will further improve model performance in data-scarce situations. In clinical applications, deep learning is expected to further improve doctors' work efficiency and diagnostic accuracy by integrating with other technologies such as augmented reality (AR) and virtual reality (VR). For example, real-time image analysis based on deep learning can provide more precise guidance for surgical navigation, thereby increasing surgical success rates and reducing postoperative complications. Moreover, with the continuous improvement in the performance of computing devices, the inference speed and processing power of deep learning models will also be significantly enhanced. This will enable deep learning technologies to be more widely applied in real clinical settings, achieving real-time, accurate diagnosis and treatment of lumbar spine diseases.

7. Conclusion

We have summarized the latest applications of deep learning in various modalities of lumbar spine imaging while also compiling a list of available public datasets and discussing common models used in different tasks. Deep learning has now become one of the mainstream directions in the field of lumbar spine image analysis. The rapid and accurate performance demonstrated by deep learning in image classification, detection, segmentation, and reconstruction can be reliably applied to the diagnosis, treatment, and prognosis of lumbar spine diseases, effectively enhancing doctors' work efficiency. Although some problems and challenges exist, with the future emphasis on privacy protection, the improvements in model interpretability and generalization abilities, as well as the continuous development of computing devices, deep learning is expected to become an important tool for managing spinal diseases.

Funding

This research received no external funding.

Institutional Review Board Statement

Not applicable.

Informed Consent Statement

Not applicable.

Data Availability Statement

Not applicable.

Conflicts of Interest

The author declares no conflict of interest.

References

1. PlCervin Serrano, S.; González Villareal, D.; Aguilar-Medina, M.; Romero-Navarro, J.G.; Romero Quintana, J.G.; Arámbula Meraz, E.; Osuna Ramírez, I.; Picos-Cárdenas, V.; Granados, J.; Estrada-García, I.; et al. Genetic polymorphisms of interleukin-1 alpha and the vitamin d receptor in mexican mestizo patients with intervertebral disc degeneration. *Int. J. Genomics* **2014**, *2014*, 302568.
2. Hodler, J.; Kubik-Huch, R.A.; von Schulthess, G.K. *Musculoskeletal Diseases 2021–2024: Diagnostic Imaging*; Springer Cham: Cham, Switzerland, 2021.
3. Ravindra, V.M.; Senglaub, S.S.; Rattani, A.; Dewan, M.C.; Härtl, R.; Bisson, E.; Park, K.B.; Shrime, M.G. Degenerative lumbar spine disease: Estimating global incidence and worldwide volume. *Global Spine J.* **2018**, *8*, 784–794.
4. Heikkinen, J.; Honkanen, R.; Williams, L.; Leung, J.; Rauma, P.; Quirk, S.; Koivumaa-Honkanen, H. Depressive disorders, anxiety disorders and subjective mental health in common musculoskeletal diseases: A review. *Maturitas* **2019**, *127*, 18–25.
5. Hooten, W.M.; Cohen, S.P. Evaluation and treatment of low back pain: A clinically focused review for primary care specialists. *Mayo Clin. Proc* **2015**, *90*, 1699–1718.
6. Russo, F.; De Salvatore, S.; Ambrosio, L.; Vadalà, G.; Fontana, L.; Papalia, R.; Rantanen, J.; Iavicoli, S.; Denaro, V. Does workers' compensation status affect outcomes after lumbar spine surgery? A systematic review and meta-analysis. *Int. J. Environ. Res. Public Health* **2021**, *18*, 6165.
7. Kim, G.U.; Park, W.T.; Chang, M.C.; Lee, G.W. Diagnostic Technology for Spine Pathology. *Asian Spine J.* **2022**, *16*, 764–775.
8. Tjardes, T.; Shafizadeh, S.; Rixen, D.; Paffrath, T.; Bouillon, B.; Steinhausen, E.S.; Baethis, H. Image-guided spine surgery: State of the art and future directions. *Eur. Spine J.* **2010**, *19*, 25–45.
9. Corona-Cedillo, R.; Saavedra-Navarrete, M.-T.; Espinoza-Garcia, J.-J.; Mendoza-Aguilar, A.-N.; Ternovoy, S.K.; Roldan-Valadez, E. Imaging assessment of the postoperative spine: An updated pictorial review of selected complications. *Biomed. Res. Int.* **2021**, *2021*, 9940001.
10. Ou, X.; Chen, X.; Xu, X.; Xie, L.; Chen, X.; Hong, Z.; Bai, H.; Liu, X.; Chen, Q.; Li, L.; et al. Recent development in X-ray imaging technology: Future and challenges. *Research* **2021**, *2021*, 9892152.
11. Van Reeth, E.; Tham, I.W.; Tan, C.H.; Poh, C.L. Super-resolution in magnetic resonance imaging: A review. *Concepts Magn. Reson.* **2012**, *40A*, 306–325.
12. Zaitsev, M.; Maclaren, J.; Herbst, M. Motion artifacts in MRI: A complex problem with many partial solutions. *J. Magn. Reson. Imaging* **2015**, *42*, 887–901.
13. O'Mahony, N.; Campbell, S.; Carvalho, A.; Harapanahalli, S.; Velasco Hernandez, G.; Krpalkova, L.; Riordan, D.; Walsh, J. Deep learning vs. traditional computer vision. In *Advances in Computer Vision: Proceedings of the 2019 Computer Vision Conference (CVC)*; Springer: Cham, Switzerland, 2020; Volume 1, pp. 128–144.
14. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444.
15. Zhuang, F.; Qi, Z.; Duan, K.; Xi, D.; Zhu, Y.; Zhu, H,; Xiong, H.; He, Q. A comprehensive survey on transfer learning. *Proc. IEEE* **2020**, *109*, 43–76.
16. Qu, B.; Cao, J.; Qian, C.; Wu, J.; Lin, J.; Wang, L.; Ou-Yang, L.; Chen, Y.; Yan, L.; Hong, Q.; et al. Current development and prospects of deep learning in spine image analysis: A literature review. *Quant Imaging Med Surg.* **2022**, *12*, 3454.
17. Lee, J.; Chung, S.W. Deep learning for orthopedic disease based on medical image analysis: Present and future. *Appl. Scie.* **2022**, *12*, 681.
18. LeCun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based learning applied to document recognition. *Proc. IEEE* **1998**, *86*, 2278–2324.
19. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet classification with deep convolutional neural networks. *Commun. ACM* **2017**, *60*, 84–90.
20. He, K.; Zhang, X.; Ren, S.; Sun. J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
21. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely connected convolutional networks. In Proceedings of the IEEE Conference on Computer Vision And Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4700–4708.
22. Tan, M.; Le, Q. Efficientnet: Rethinking model scaling for convolutional neural networks. In Proceedings of International Conference on Machine Learning, Long Beach, CA, USA, 9–15 June 2019; pp. 6105–6114.
23. Devlin, J.; Chang, M.-W.; Lee, K.; Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv* **2018**, arXiv:1810.04805.
24. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic

segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 580–587.
25. Girshick, R. Fast R-CNN. In Proceedings of the IEEE International Conference on Computer Vision, Boston, MA, USA, 7–12 June 2015; pp. 1440–1448.
26. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *39*, 1137–1149.
27. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognitio*n*, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
28. Redmon, J.. Farhadi, A. YOLO9000: Better, faster, stronger. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7263–7271.
29. Redmon, J. Farhadi, A. YOLOv3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
30. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; Berg, A.C. Ssd: Single shot multibox detector. In Proceedings of Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, 11–14 October 2016; pp. 21–37.
31. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3431–3440.
32. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Proceedings of Medical Image Computing and Computer-Assisted Intervention–Miccai 2015: 18th International Conference, Munich, Germany, 5–9 October 2015; pp. 234–241.
33. Badrinarayanan, V.; Kendall, A.; Cipolla, R. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 2481–2495.
34. Chen, L.-C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. Semantic image segmentation with deep convolutional nets and fully connected CRFs. *arXiv* **2014**, arXiv:1412.7062.
35. Chen, L.-C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *40*, 834–848.
36. Hossin, M. Sulaiman, M.N. A review on evaluation metrics for data classification evaluations. *Int. J. Data Min. Knowl. Manag. Process* **2015**, *5*, 1.
37. Stuckner, J.; Harder, B.; Smith, T.M. Microstructure segmentation with deep learning encoders pre-trained on a large microscopy dataset. *NPJ Comput. Mater.* **2022**, *8*, 200.
38. Laroca, R.; Cardoso, E.V.; Lucio, D.R.; Estevam, V.; Menotti, D. On the cross-dataset generalization in license plate recognition. *arXiv* **2022**, arXiv:2201.00267.
39. Duong-Trung, N.; Son, H.X.; Le, H.T.; Phan, T.T. Smart care: Integrating blockchain technology into the design of patient-centered healthcare systems. In Proceedings of the 2020 4th International Conference on Cryptography, Security and Privacy, Nanjing, China, 10–12 January 2020; pp. 105–109.
40. El Alaoui, S. Lindefors, N. Combining time-driven activity-based costing with clinical outcome in cost-effectiveness analysis to measure value in treatment of depression. *PLoS ONE* **2016**, *11*, e0165389.
41. Bansal, M.A.; Sharma, D.R.; Kathuria, D.M. A systematic review on data scarcity problem in deep learning: Solution and applications. *ACM Comput. Surv.* **2022**, *54*, 1–29.
42. Al-kubaisi, A. Khamiss, N.N. A transfer learning approach for lumbar spine disc state classification. *Electronics* **2021**, *11*, 85.
43. Lumbar Spine MRI Dataset. Available online: https://data.mendeley.com/datasets/k57fr854j2/2 (accessed on 16 April 2024).
44. Masood, R.F.; Taj, I.A.; Khan, M.B.; Qureshi, M.A.; Hassan, T. Deep learning based vertebral body segmentation with extraction of spinal measurements and disorder disease classification. *Biomed. Signal Process. Control* **2022**, *71*, 103230.
45. Liawrungrueang, W.; Kim, P.; Kotheeranurak, V.; Jitpakdee, K.; Sarasombath, P. Automatic detection, classification, and grading of lumbar intervertebral disc degeneration using an artificial neural network model. *Diagnostics* **2023**, *13*, 663.
46. Le Van, C.; Bao, L.; Puri, V.; Thao, N.T.; Le, D.-N. Detecting lumbar implant and diagnosing scoliosis from vietnamese X-ray imaging using the pre-trained api models and transfer learning. *CMC Comput. Mater. Contin* **2021**, *66*, 17–33.
47. Spineweb. Available online: http://spineweb.digitalimaginggroup.ca/Index.php?n=Main.Datasets (accessed on 16 April 2024).
48. MICCAI 2019 Challenge. Available online: https://aasce19.github.io/ (accessed on 16 April 2024).
49. VerSe2020. Available online: https://osf.io/t98fz/ (accessed on 16 April 2024).
50. VerSe 2019. Available online: https://osf.io/nqjyw/ (accessed on 16 April 2024).
51. xVertSeg. Available online: https://lit.fe.uni-lj.si/xVertSeg/database.php (accessed on 16 April 2024).
52. BUU Spine Dataset. Available online: https://services.informatics.buu.ac.th/spine/ (accessed on 16 April 2024).
53. Khare, M.R.; Havaldar, R.H. Predicting the anterior slippage of vertebral lumbar spine using Densenet-201. *Biomed. Signal Process. Control,* **2023**, *86*, 105115.
54. Varçın, F.; Erbay, H.; Çetin, E.; Çetin, İ.; Kültür, T. End-to-end computerized diagnosis of spondylolisthesis using only lumbar X-rays. *J. Digit. Imaging* **2021**, *34*, 85–95.

55. Sugiura, A.; Yasuda, E. Algorithmic Attempt of Deflection Angle on The Frontal Lumbar Image By Lateral Lumbar Image Analysis. *Int. J. Innov. Sci. Eng. Technol.* **2022**, *9*, 10–19.
56. Nissinen, T.; Suoranta, S.; Saavalainen, T.; Sund, R.; Hurskainen, O.; Rikkonen, T.; Kröger, H.; Lähivaara, T.; Väänänen, S.P. Detecting pathological features and predicting fracture risk from dual-energy X-ray absorptiometry images using deep learning. *Bone Rep.* **2021**, *14*, 101070.
57. Zhang, B.; Yu, K.; Ning, Z.; Wang, K.; Dong, Y.; Liu, X.; Liu, S.; Wang, J.; Zhu, C.; Yu, Q. Deep learning of lumbar spine X-ray for osteopenia and osteoporosis screening: A multicenter retrospective cohort study. *Bone* **2020**, *140*, 115561.
58. Saravagi, D.; Agrawal, S.; Saravagi, M.; Chatterjee, J.M.; Agarwal, M. Diagnosis of lumbar spondylolisthesis using optimized pretrained CNN models. *Comput. Intell. Neurosci.* **2022**, *2022*. https://doi.org/10.1155/2022/7459260.
59. Kim, T.; Kim, Y.G.; Park, S.; Lee, J.K.; Lee, C.H.; Hyun, S.J.; Kim, C.H.; Kim, K.J.; Chung, C.K. Diagnostic triage in patients with central lumbar spinal stenosis using a deep learning system of radiographs. *J. Neurosurg. Spine* **2022**, *37*, 104–111.
60. Patil, K.A.; Prashanth, K.M.; Ramalingaiah A. Law Texture Analysis for the Detection of Osteoporosis of Lumbar Spine (L1-L4) X-ray Images Using Convolutional Neural Networks. *IAENG Int. J. Comput. Sci.* **2023**, *50*, 71–85.
61. Varçin, F.; Erbay, H.; Çetin, E.; Çetin, İ.; Kültür T. Diagnosis of lumbar spondylolisthesis via convolutional neural networks. In Proceedings of 2019 International Artificial Intelligence and Data Processing Symposium (IDAP), Malatya, Turkey, 21–22 September 2019; pp. 1–4.
62. Klinwichit, P.; Chinnasarn, K.; Onuean, A.; Limchareon, S.; Lee, S.-H.; Jang, J.-S. The Radiographic view classification and localization of Lumbar spine using Deep Learning Models. In Proceedings of 2022 13th International Conference on Information and Communication Technology Convergence (ICTC), Jeju Island, Republic of Korea, 19–21 October 2022; pp. 1316–1319.
63. An, C.-H.; Lee, J.-S.; Jang, J.-S.; Choi, H.-C. Part affinity fields and CoordConv for detecting landmarks of lumbar vertebrae and sacrum in X-ray images. *Sensors* **2022**, *22*, 8628.
64. Nguyen, T.P.; Chae, D.-S.; Park, S.-J.; Kang, K.-Y.; Yoon, J. Deep learning system for Meyerding classification and segmental motion measurement in diagnosis of lumbar spondylolisthesis. *Biomed. Signal Process. Control* **2021**, *65*, 102371.
65. Zhou, S.; Yao, H.; Ma, C.; Chen, X.; Wang, W.; Ji, H.; He, L.; Luo, M.; Guo, Y. Artificial intelligence X-ray measurement technology of anatomical parameters related to lumbosacral stability. *Eur. J. Radiol.* **2022**, *146*, 110071.
66. Ruhan, Sa.; Owens, W.; Wiegand, R.; Studin, M.; Capoferri, D.; Barooha, K.; Greaux, A.; Rattray, R.; Hutton, A.; Cintineo, J.; et al. Intervertebral disc detection in X-ray images using faster R-CNN. In Proceedings of 2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Jeju Island, Korea, 11–15 July 2017; pp. 564–567.
67. Lee, J.H.; Woo, H.J.; Lee, J.H.; Kim, J.I.; Jang, J.S.; Na, Y.C.; Kim, K.R.; Park, T.Y. Comparison of concordance between Chuna manual therapy diagnostic methods (palpation, X-ray, artificial intelligence program) in lumbar spine: An exploratory, cross-sectional clinical study. *Diagnostics* **2022**, *12*, 2732.
68. Kim, K.C.; Cho, H.C.; Jang, T.J.; Choi, J.M.; Seo, J.K. Automatic detection and segmentation of lumbar vertebrae from X-ray images for compression fracture evaluation. *Comput. Meth. Programs Biomed.* **2021**, *200*, 105833.
69. Trinh, G.M.; Shao, H.-C.; Hsieh, K.L.-C.; Lee, C.-Y.; Liu, H.-W.; Lai, C.-W.; Chou, S.-Y.; Tsai, P.-I.; Chen, K.-J.; Chang, F.-C.; et al. LumbarNet: A Deep Learning Network for the Automated Detection of Lumbar Spondylolisthesis From X-ray Images. *Preprints* **2022**. https://doi.org/10.20944/preprints202206.0043.v1.
70. Chen, X.; Deng, Q.; Wang, Q.; Liu, X.; Chen, L.; Liu, J.; Li, S. Wang M and Cao G Image quality control in lumbar spine radiography using enhanced U-Net neural networks. *Front. Public Health* **2022**, *10*, 891766.
71. Tran, V.L.; Lin, H.-Y.; Liu, H.-W. MBNet: A multi-task deep neural network for semantic segmentation and lumbar vertebra inspection on X-ray images. In Proceedings of the Asian Conference on Computer Vision, Virtual Kyoto, 30 November–4 December 2020.
72. Kónya, S.; Natarajan, T.S.; Allouch, H.; Nahleh, K.A.; Dogheim, O.Y.; Boehm, H. Convolutional neural network-based automated segmentation and labeling of the lumbar spine X-ray. *J. Craniovertebral Junction Spine* **2021**, *12*, 136–143.
73. Cho, B.H.; Kaji, D.; Cheung, Z.B.; Ye, I.B.; Tang, R.; Ahn, A.; Carrillo, O.; Schwartz, J.T.; Valliani, A.A.; Oermann, E.K.; et al. Automated measurement of lumbar lordosis on radiographs using machine learning and computer vision. *Glob. Spine J.* **2020**, *10*, 611–618.
74. Lin. H.-Y.; Liu, H.-W. Multitask deep learning for segmentation and lumbosacral spine inspection. *IEEE Trans. Instrum. Meas.* **2022**, *71*, 1–10.
75. Ryu, S.M.; Lee, S.; Jang, M.; Koh, J.M.; Bae, S.J.; Jegal, S.G.; Shin, K.; Kim, N. Diagnosis of osteoporotic vertebral compression fractures and fracture level detection using multitask learning with U-Net in lumbar spine lateral radiographs. *Comput. Struct. Biotechnol. J.* **2023**, *21*, 3452–3458.
76. Malatong, Y.; Intasuwan, P.; Palee, P.; Sinthubua, A.; Mahakkanukrauh, P. Deep learning and morphometric approach for Sex determination of the lumbar vertebrae in a Thai population. *Med.; Sci. Law* **2023**, *63*, 14–21, 2023.
77. Yasaka, K.; Akai, H.; Kunimatsu, A.; Kiryu, S.; Abe, O. Prediction of bone mineral density from computed tomography: Application of deep learning with a convolutional neural network. *Eur. Radiol.* **2020**, *30*, 3549–3557.
78. Doerr, S.A.; Weber-Levine, C.; Hersh, A.M.; Awosika, T.; Judy, B.; Jin, Y.; Raj, D.; Liu, A.; Lubelski, D.; Jones, C.K.; Sair, H.I.; Theodore, N. Automated prediction of the Thoracolumbar Injury Classification and Severity Score from CT

using a novel deep learning algorithm. *Neurosurg. Focus* **2022**, *52*, E5.
79. Lu, H.; Li, M.; Yu, K.; Zhang, Y.; Yu, L. Lumbar spine segmentation method based on deep learning. *J. Appl Clin. Med. Phys.* **2023**, *24*, e13996.
80. Malinda, V.; Lee, D. Lumbar vertebrae synthetic segmentation in computed tomography images using hybrid deep generative adversarial networks. In Proceedings of 2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), Virtual, 20–24 July 2020; pp. 1327–1330.
81. Gao, X.; Zheng, G. ACSGRegNet: A Deep Learning-based Framework for Unsupervised Joint Affine and Diffeomorphic Registration of Lumbar Spine CT via Cross-and Self-Attention Fusion. In Proceedings of the 2022 International Conference on Intelligent Medicine and Health, New York, NY, USA, 19–21 August 2022; pp. 57–63.
82. Greffier, J.; Frandon, J.; Durand, Q.; Kammoun, T.; Loisy, M.; Beregi, J.P.; Dabli, D. Contribution of an artificial intelligence deep-learning reconstruction algorithm for dose optimization in lumbar spine CT examination: A phantom study. *Diagn. Interv. Imaging* **2023**, *104*, 76–83.
83. Morbée, L.; Chen, M.; Herregods, N.; Pullens, P.; Jans, L.B. MRI-based synthetic CT of the lumbar spine: Geometric measurements for surgery planning in comparison with CT. *Eur. J. Radiol.* **2021**, *144*, 109999.
84. Yeoh, H.; Hong, S.H.; Ahn, C.; Choi, J.Y.; Chae, H.D.; Yoo, H.J.; Kim, J.H. Deep learning algorithm for simultaneous noise reduction and edge sharpening in low-dose CT images: A pilot study using lumbar spine CT. *Korean J. Radiol.* **2021**, *22*, 1850.
85. Oura, P.; Korpinen, N.; Machnicki, A.L.; Junno, J.-A. Deep learning in sex estimation from a peripheral quantitative computed tomography scan of the fourth lumbar vertebra—A proof-of-concept study. *Forens. Sci. Med. Pathol.* **2023**, *19*, 534–540.
86. Dheivya, I.; Gurunathan, S.K. Deep Learning Based Lumbar Metastases Detection and Classification from Computer Tomography Images. In Proceedings of 2022 IEEE-EMBS Conference on Biomedical Engineering and Sciences (IECBES), Kuala Lumpur, Malaysia, 7–9 December 2022; pp. 398–403.
87. Fan. G.; Liu, H.; Wang, D.; Feng, C.; Li, Y.; Yin, B.; Zhou, Z.; Gu, X.; Zhang, H.; Lu, Y.; He, S. Deep learning-based lumbosacral reconstruction for difficulty prediction of percutaneous endoscopic transforaminal discectomy at L5/S1 level: A retrospective cohort study. *Int. J. Surg.* **2020**, *82*, 162–169.
88. Janssens, R. Zheng, G. Deep learning based segmentation of lumbar vertebrae from CT images. *CAOS* **2018**, *2*, 94–97.
89. Miyo, R.; Yasaka, K.; Hamada, A.; Sakamoto, N.; Hosoi, R.; Mizuki, M.; Abe, O. Deep-learning reconstruction for the evaluation of lumbar spinal stenosis in computed tomography. *Medicine* **2023**, *102*, e33910.
90. Chen, G.; Xu, Z. Usage of intelligent medical aided diagnosis system under the deep convolutional neural network in lumbar disc herniation. *Appl. Soft Comput.* **2021**, *111*, 107674.
91. Cheung, J.P.Y.; Kuang, X.; Lai, M.K.L.; Cheung, K.M.; Karppinen, J.; Samartzis, D.; Wu, H.; Zhao, F.; Zheng, Z.; Zhang, T. Learning-based fully automated prediction of lumbar disc degeneration progression with specified clinical parameters and preliminary validation. *Eur. Spine J.* **2021**, *31*, 1–9.
92. Gao, F.; Liu, S.; Zhang, X.; Wang, X.; Zhang, J. Automated Grading of Lumbar Disc Degeneration Using a Push-Pull Regularization Network Based on MRI. *J. Magn. Reson. Imaging* **2021**, *53*, 799–806.
93. Grob, A.; Loibl, M.; Jamaludin, A.; Winklhofer, S.; Fairbank, J.C.T.; Fekete, T.; Porchet, F.; Mannion, A.F. External validation of the deep learning system "SpineNet" for grading radiological features of degeneration on MRIs of the lumbar spine.*Eur. Spine J.* **2022**, 2137–2148.
94. Zhou, Y.; Liu, Y.; Chen, Q.; Gu, G.; Sui, X. Automatic lumbar MRI detection and identification based on deep learning. *J. Digit. Imaging* **2019**, *32*, 513–520.
95. Mushtaq, M.; Akram, M.U.; Alghamdi, N.S.; Fatima, J.; Masood, R.F. Localization and edge-based segmentation of lumbar spine vertebrae to identify the deformities using deep learning models. *Sensors* **2022**, *22*, 1547.
96. Tsai, J.Y.; Hung, I.Y.; Guo, Y.L.; Jan, Y.K.; Lin, C.Y.; Shih, T.T.; Chen, B.B.; Lung, C.W. Lumbar disc herniation automatic detection in magnetic resonance imaging based on deep learning. *Front. Bioeng. Biotechnol.* **2021**, *9*, 708137.
97. Yi, W.; Zhao, J.; Tang, W.; Yin, H.; Yu, L.; Wang, Y. Tian, W. Deep learning-based high-accuracy detection for lumbar and cervical degenerative disease on T2-weighted MR images. *Eur. Spine J.* **2023**, *32*, 3807–3814.
98. Forsberg, D.; Sjöblom, E.; Sunshine, J.L. Detection and labeling of vertebrae in MR images using deep learning with clinical annotations as training data. *J. Digit. Imaging* **2017**, *30*, 406–412.
99. Zeybel, M.; Akgul, Y.S. Localization and Identification of Lumbar Intervertebral Discs on Spine MR Images with Faster RCNN Based Shortest Path Algorithm. In Proceedings of Annual Conference on Medical Image Understanding and Analysis, Oxford, UK, 15–17 July 2020; pp. 143–154.
100. Li, H.; Luo, H.; Huan, W.; Shi, Z.; Yan, C.; Wang, L.; Mu, Y. Liu, Y. Automatic lumbar spinal MRI image segmentation with a multi-scale attention network. *Neural Comput. Appl.* **2021**, *33*, 11589–11602.
101. Zheng, H.D.; Sun, Y.L.; Kong, D.W.; Yin, M.-C.; Chen, J.; Lin, Y.-P.; Ma, X.-F.; Wang, H.-S.; Yuan, G.-J.; Yao, M,; et al. Deep learning-based high-accuracy quantitation for lumbar intervertebral disc degeneration from MRI. *Nat. Commun.* **2022**, *13*, 841.
102. Chen, T.; Su, Z.-h.; Liu, Z.; Wang, M.; Cui, Z.-F; Zhao, L.; Yang, L.-J.; Zhang, W.-C.; Liu, X.; Liu, J.; et al. Automated Magnetic Resonance Image Segmentation of Spinal Structures at the L4-5 Level with Deep Learning: 3D Reconstruction of Lumbar Intervertebral Foramen. *Orthop. Surg.* **2022**, *14*, 2256–2264.
103. Huang, J.; Shen, H.; Wu, J.; Hu, X.; Zhu, Z.; Lv, X.; Liu, Y.; Wang, Y. Spine Explorer: A deep learning based fully

automated program for efficient and reliable quantifications of the vertebrae and discs on sagittal lumbar spine MR images. *Spine J.* **2020**, *20*, 590–599.
104. Lu, J.-T.; Pedemonte, S.; Bizzo, B.; Doyle, S.; Andriole, K.P.; Michalski, M.H.; Gonzalez, B.G.; Pomerantz, S.R. Deep spine: Automated lumbar vertebral segmentation, disc-level designation, and spinal stenosis grading using deep learning. In Proceedings of Machine Learning for Healthcare Conference, Stanford, CA, USA, 16–18 August 2018; pp. 403–419.
105. Mbarki, W.; Bouchouicha, M.; Frizzi, S.; Tshibasu, F.; Farhat, L.B.; Sayadi, M. Lumbar spine discs classification based on deep convolutional neural networks using axial view MRI. *Interdiscip. Neurosurg.* **2020**, *22*, 100837.
106. Zhou, J.; Damasceno, P.F.; Chachad, R.; Cheung, J.R.; Ballatori, A.; Lotz, J.C.; Lazar, A.A.; Link, T.M.; Fields, A.J.; Krug, R. Automatic vertebral body segmentation based on deep learning of Dixon images for bone marrow fat fraction quantification. *Front. Endocrinol.* **2020**, *11*, 612.
107. Liu, Z.; Su, Z.; Wang, M.; Chen, T.; Cui, Z.; Chen, X.; Li, S.; Feng, Q.; Pang, S.; Lu, H. Computerized characterization of spinal structures on MRI and clinical significance of 3D reconstruction of lumbosacral intervertebral foramen. *Pain Phys.* **2022**, *25*, E27.
108. Chazen, J.L.; Tan, E.T.; Fiore, J.; Nguyen, J.T.; Sun, S.; Sneag, D.B. Rapid lumbar MRI protocol using 3D imaging and deep learning reconstruction. *Skelet. Radiol.* **2023**, *52*, 1331–1338.
109. Fujiwara, M.; Kashiwagi, N.; Matsuo, C.; Watanabe, H.; Kassai, Y.; Nakamoto, A.; Tomiyama, N. Ultrafast lumbar spine MRI protocol using deep learning–based reconstruction: Diagnostic equivalence to a conventional protocol. *Skelet. Radiol.* **2023**, *52*, 233–241.
110. Han, M.; Bahroos, E.; Hess, M.E.; Chin, C.T.; Gao, K.T.; Shin, D.D.; Villanueva-Meyer, J.E.; Link, T.M.; Pedoia, V.; Majumdar, S. Technology and Tool Development for BACPAC: Qualitative and Quantitative Analysis of Accelerated Lumbar Spine MRI with Deep-Learning Based Image Reconstruction at 3T. *Pain Med.* **2023**, *24*, S149–S159.
111. Zerunian, M.; Pucciarelli, F.; Caruso, D.; De Santis, D.; Polici, M.; Masci, B.; Nacci, I.; Del Gaudio, A.; Argento, G.; Redler, A.; et al. Fast high-quality MRI protocol of the lumbar spine with deep learning-based algorithm: An image quality and scanning time comparison with standard protocol. *Skelet. Radiol.* **2024**, *53*, 151–159.
112. Gao, F.; Wu, M. Deep learning-based denoised MRI images for correlation analysis between lumbar facet joint and lumbar disc herniation in spine surgery. *J. Healthc. Eng.* **2021**, *2021*, 9687591.
113. Sun, S.; Tan, E.T.; Mintz, D.N.; Sahr, M.; Endo, Y.; Nguyen, J.; Lebel, R.M.; Carrino, J.A.; Sneag, D.B. Evaluation of deep learning reconstructed high-resolution 3D lumbar spine MRI. *Eur. Radiol.* **2022**, *32*, 6167–6177.
114. Debelee, G.; Schwenker, F.; Ibenthal, A.; Yohannes, D. Survey of deep learning in breast cancer image analysis. *Evol. Syst.* **2020**, *11*, 143–163.
115. Chen, C.; Qin, C.; Qiu, H.; Tarroni, G.; Duan, J.; Bai, W.; Rueckert, D. Deep learning for cardiac image segmentation: A review. *Front. Cardiovasc. Med.* **2020**, *7*, 25.
116. Bayasi, N.; Hamarneh, G.; Garbi, R. BoosterNet: Improving domain generalization of deep neural nets using culpability-ranked features. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 538–548.
117. Han, X.; Zhang, Z.; Ding, N.; Gu, Y.; Liu, X.; Huo, Y.; Qiu, J.; Yao, Y.; Zhang, A.; Zhang, L. Pre-trained models: Past, present and future. *AI Open* **2021**, *2*, 225–250.
118. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv* **2017**, arXiv:1704.04861.

AI Medicine
https://www.sciltp.com/journals/aim

Article

Ultrasonic Image's Annotation Removal: A Self-Supervised Noise2Noise Approach

Yuanheng Zhang [1], Nan Jiang [2], Zhaoheng Xie [3], Junying Cao [2,*] and Yueyang Teng [1,*]

[1] College of Medicine and Biological Information Engineering, Northeastern University, Shenyang 110016, China
[2] The Department of Ultrasound, General Hospital of Northern Theater Command, Shenyang 110169, China
[3] The Institute of Medical Technology, Peking University, Beijing 100191, China
* Correspondence: shenzongchaosheng@163.com (J.C.); tengyy@bmie.neu.edu.cn (Y.T.)

How To Cite: Zhang, Y., Jiang, N., Xie, Z., Cao, J.; Teng, Y. Ultrasonic Image's Annotation Removal: A Self-Supervised Noise2Noise Approach. *AI Medicine* **2024**, *1*(1), 4. https://doi.org/10.53941/aim.2024.100004.

Received: 11 March 2024
Revised: 25 May 2024
Accepted: 28 May 2024
Published: 17 July 2024

Abstract: Accurately annotated ultrasonic images are vital components of a high-quality medical report. Hospitals often have strict guidelines on the types of annotations that should appear on imaging results. However, manually inspecting these images can be a cumbersome task. While a neural network could potentially automate the process, training such a model typically requires a dataset of paired input and target images, which in turn involves significant human labor. This study introduces an automated approach for detecting annotations in images. This is achieved by treating the annotations as noise, creating a self-supervised pretext task and using a model trained under the Noise2Noise scheme to restore the image to a clean state. We tested a variety of model structures on the denoising task against different types of annotation, including body marker annotation, radial line annotation, etc. Our results demonstrate that most models trained under the Noise2Noise scheme outperformed their counterparts trained with noisy-clean data pairs. The costumed U-Net yielded the most optimal outcome on the body marker annotation dataset, with high scores on segmentation precision and reconstruction similarity. Our approach streamlines the laborious task of manually quality-controlling ultrasound scans, with minimal human labor involved, making the quality control process efficient and scalable.

Keywords: image restoration; Noise2Noise; segmentation; U-Net; ultrasonic

1. Introduction

Annotations, typically comprised of various labels and marks, are commonly utilized to record critical information from an ultrasonic exam, including the precise location of potential lesions or suspicious findings, on archived results. Such annotations prove beneficial in aiding physicians in interpreting the exam results, particularly when surrounding structures do not provide any indication of the anatomic location of the image. Additionally, hospitals often mandate the inclusion of annotations, especially in cases involving inter-hospital patient transfers [1]. If the report lacks comprehensive annotations, patients are usually required to undergo an equivalent radiography exam at the facility of transfer.

Commonly employed types of annotations include body marker annotation [2], radial line annotation, and vascular flow annotation. The presence of these annotations serves as evidence for the standardization of the diagnostic process. Annotations not only document the reasoning behind the diagnostic assessment but also facilitate comparison between pre- and post-treatment imaging findings to gain further insights into the patient's condition.

However, the utilization of annotations during ultrasound exams may vary depending on the proficiency of the sonographer performing the procedure. Ultrasound being a live examination makes it hard to implement

additional reviews, thereby relying solely on the expertise of the operator to determine the presence of annotations. Furthermore, the need for repetitive manual verification increases the likelihood of forgetting the task, particularly during busy schedules at hospitals. As such, it is possible for the absence of annotations to occur.

Given the strict regulations and obvious beneficiation surrounding the need for annotations in medical imaging, sonographers need to manually validate that the stored data satisfies these requirements to ensure that diagnoses meet the standard continuously. However, this is a cognitively demanding undertaking as it entails the fulfillment of diverse annotation obligations tailored to specific image outcomes. In addition, dealing with archived files manually is a cumbersome task as most medical data management systems do not consider this necessary and have no relevant feature implemented.

The utilization of neural networks for the automatic assessment of whether the stored data meets particular criteria is a logical approach. To address the current issue, several approaches can be adopted using different types of deep learning models. The first approach would involve treating the task as a semantic segmentation problem, where the goal is to classify each pixel in the image into one of several predefined categories. Alternatively, the task could be framed as an instance segmentation problem, where the aim is to identify and label individual objects within the scene. To accomplish these goals, attention-based models such as the Pyramid Attention Network [3] or the Reverse Attention Network [4] could be employed. Alternatively, generative models like variants of Generative Adversarial Networks (GANs) [5] are also viable. Regardless of the method, once segmentation is completed successfully, the results can be utilized to ascertain the presence or absence of an annotation. (To illustrate, the determination can be achieved by examining the number of white pixels that remain following the application of a filter designed to eliminate noise on the segmentation result.)

This task could also be viewed as an object recognition challenge, and for this purpose, models such as Single Shot MultiBox Detector (SSD) [6] or You Only Look Once (YOLO) [7] could be utilized to obtain the four coordinates of the bonding box of a detected object, which will serve as demonstrative evidence of the necessary annotations.

To train a model using deep learning, it is important to have a suitable training dataset that includes paired input and output data, regardless of the specific task being performed. However, building an appropriate training dataset is a challenging task due to the absence of high-quality data such as segmentation masks, object coordinates and clean targets. Acquiring such data requires a considerable amount of manual effort.

Addressing the challenge of limited labeled data for annotation recognition, this study proposes a self-supervised Noise2Noise approach. The Noise2Noise method stands as a novel training paradigm that departs from the conventional Noise2Clean approach. Unlike Noise2Clean, which necessitates paired noisy-clean image datasets, Noise2Noise leverages innovative mathematical principles to train a denoising model solely on noisy data. This eliminates the requirement for a large and often impractical collection of clean images.

Building upon the Noise2Noise framework, we propose a novel self-supervised strategy, where common annotations are treated as noise and randomly superimposed, in a repetitive manner, onto a limited set of unannotated images. This process effectively generates a synthetic training dataset specifically tailored for the Noise2Noise framework. The trained model, equipped to remove noise (in this case, annotations), can then be employed for annotation recognition without requiring clean image counterparts.

We trained multiple network structures such as FCN, U-Net++, MultiResUNet, etc., under both training paradigms to select an ideal one. We noted that the majority of Noise2Noise-based methods surpassed the corresponding Noise2Clean (supervised learning) methods in which the former even received a Sørensen-Dice coefficient (Dice) increase of up to 300%, an Intersection over Union (IoU) increase of up to 384%, and a Peak Signal to Noise Ratio Human Visual System Modified (PSNR HVS M) increase of up to 38% in some cases. Among them, our costumed U-Net achieved the best results, both quantitative and qualitatively.

The remainder of the paper is organized as follows: Section 2 discusses related works. Section 3 outlines our methodology, data sources, dataset-building pipeline, and model structures used in this work. In Section 4, quantitative metric scores and qualitative image results are provided to support our claim regarding the optimal model structure, loss function, and observations on Noise2Noise's effect. Finally, Section 5 concludes the paper.

2. Related Work

2.1. Self-Supervised Learning

Self-supervised learning is a way of training deep-learning models without human guidance or explicit instructions. Unlike supervised learning which uses labeled examples, self-supervised models learn from unlabeled data by identifying patterns and relationships on their own. It uses the structure of images (e.g., edges, shapes) to

teach the deep-learning model how to identify important parts of an image automatically, rather than having to be explicitly told what to look for. This is particularly helpful considering the abundance of unlabeled data that exists today and the amount of work required to create a properly constructed dataset. To create a robust, large model, self-supervised learning is an essential tool.

The general process of self-supervised learning involves first creating a pretext task for the model to solve. By completing this task, the model can gain an understanding of the structural information embedded within the data. This understanding can then be transferred to downstream tasks using different forms of transfer learning.

Examples of pretext tasks include rotating an image for the model to predict the degree of rotation, reconstructing images from an altered view, or reconstructing images from a corrupted version of the original data.

In this work, we developed a pretext task where we asked the model to generate another noisy image from the noisy input while keeping the same original clean image beneath it. Specifically, we manually extracted several common annotations from stored data and randomly superimposed them on a small set of unannotated images to create a large dataset. The idea behind this approach was to train the model to recognize the crucial features of the original so that it could distinguish between noise and clean images.

2.2. Noise2Noise Training Scheme

Noise2Noise is originally proposed in [8] as a novel statistical reasoning for the task of image denoising. It is shown that, under certain key constraints, it is possible to train a denoising model using only corrupted images. The constraints are: the distribution of the added noise must have a mean of zero and no correlation with the desired clean image, and the correlation between the noise in the input image and the target image should be close to zero [9].

By utilizing deep learning, a denoising task can be transformed into a regression problem, where a neural network is used to learn the mapping between corrupted samples $\hat{x}_i$ and clean samples y_i by minimizing the empirical risk [8].

In [8], inspecting the form of a typical training process shows that training a neural network is a generalization of a point estimating problem. We can see that it is essentially solving the point estimating problem for each separate input. This means that by finding the optimal parameters, the trained neural network will output the expectation or median of all possible mapping for input x. This property often leads to unwanted fuzziness in many deep-learning applications. However, in a denoising scenario, when the noise satisfies the above constraints and exists in both the model input and training target, the task of empirical risk minimization, given infinite data,

$$\underset{\theta}{\operatorname{argmin}} \sum_i L(f_\theta(\hat{x}_i), \hat{y}_i) \tag{1}$$

is equivalent to the original regression problem

$$\underset{\theta}{\operatorname{argmin}} \sum_i L(f_\theta(\hat{x}_i), y_i) \tag{2}$$

where $f_\theta(x)$ is the model parameterized by θ, L is the loss function, $\hat{x}_i, \hat{y}_i$ are samples drawn from a noisy distribution and y_i representing clean samples.

The idea of using self-supervised learning in conjunction with Noise2Noise training scheme aligns well with our goal of obtaining a clean image. With a clean image, we can easily produce a segmentation map for various kinds of annotations, facilitating the models to recognize and categorize them accurately.

3. Methodology

Building on the aforementioned theories, we address the challenge of limited data for segmentation by treating the desired object (annotations, in this paper) as noise. After we create a Noise2Noise dataset, we train a denoising model to remove this object. The resulting denoised image, when subtracted from the original input, provides a segmentation mask. This mask allows us to determine the presence of the target object with a simple score based on the number of white pixels.

To be more specific, initially, our data includes collections of data that may or may not have specific annotations. We manually examined and filtered the data to create a clean dataset for each annotation. Next, we studied the individual components of different annotations and identified a general pattern for each one. Using this pattern, we generated large datasets containing noisy data and trained a denoising model using the Noise2Noise approach, and designed a pretext task with this dataset. Finally, we trained various model structures using both the

Noise2Noise and conventional Noise2Clean techniques to obtain denoising models for performance comparison (based on the denoised result and segmentation mask).

3.1. Dataset

To manually synthesize a self-supervised Noise2Noise dataset, which our training requires, it is essential to know the scheme of the different annotations and to construct a dataset according to it.

Our original data consists mainly of ultrasonic images provided by the General Hospital of Northern Theater Command. These images were captured using external video capture cards and are in 8-bit sRGB format.

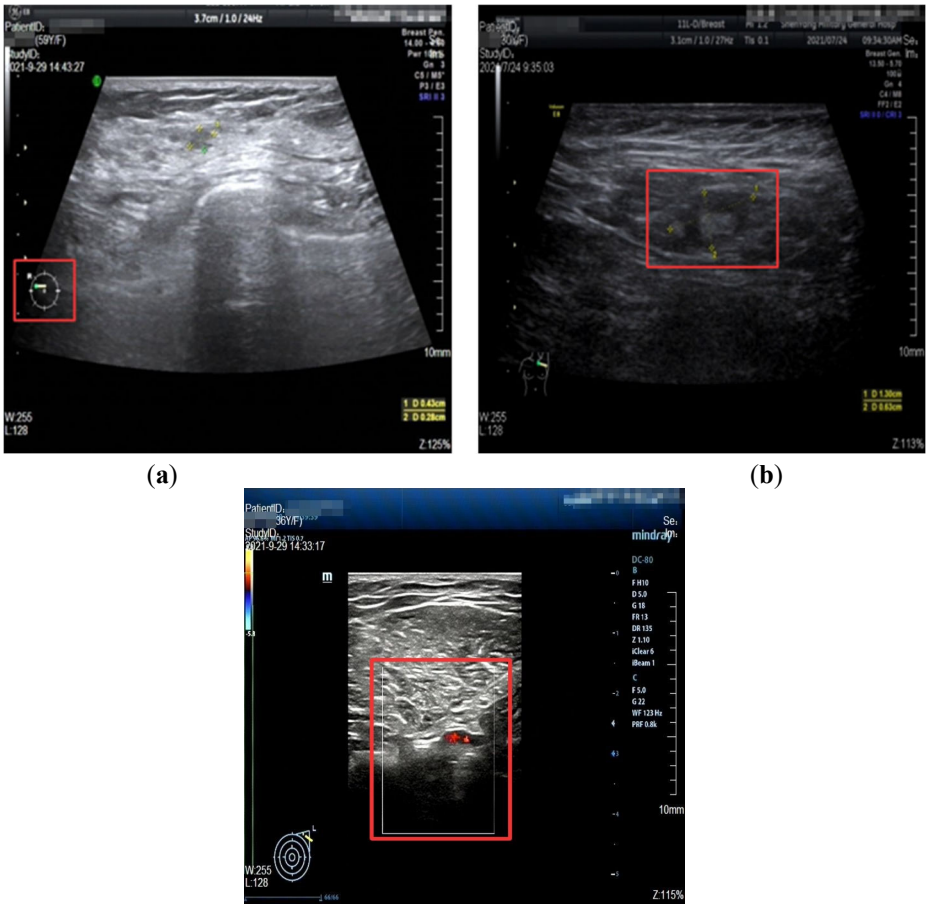
According to the type of noise, we divided these data into six categories:

- Images with body marker annotation
- Images without body marker annotation
- Images with radial line annotation
- Images without radial line annotation
- Images with vascular flow annotation
- Images without vascular flow annotation

Images with certain annotations are considered noisy images in the context of the noise removal task, and corresponding images without these annotations are considered clean. Some typical images with various annotations are provided in Figure 1.

To safeguard the confidentiality of the patient, any personal data displayed in the margin of the image is blurred using pixelization. This same technique is also used to obscure any similar information present in other images.

In essence, a body marker annotation is a marker selected from a fixed set of icons that indicates different regions of the human body and its current orientation. It is typically located at the edge of the ultrasonic image area and is labeled by the sonographer. On some ultrasound machines, the body marker annotation has a fixed position.



(**a**) (**b**) (**c**)

Figure 1. Images with various annotations: (**a**) body marker annotation; (**b**) radical line annotation; (**c**) vascular flow annotation.

However, from a statistical and training perspective, each real instance can be viewed as an image sample from a conditional distribution where the condition is the body marker annotation's location. By randomly placing body marker annotation at any position within the image, we draw samples from a distribution without the aforementioned condition. By learning to denoise samples from the unconditioned distribution, the model can effectively denoise samples from conditional distribution as well.

While we introduced randomness to annotation shapes, we did not completely randomize their placement. After analyzing existing data, we observed that body marker annotations rarely appear in the image center. So, we limited the program to placing annotations only within a 20% border around the image edges.

Other commonly used annotations that we introduced later comply with the same reasoning.

The radial line annotation indicates pairs of connected cross markers. They are usually placed at the edge of the lesion area, with its placement determined by the size of the lesion. One to three pairs of cross markers may be present in an image, corresponding to the three axes of 3D space, but typically there are only two pairs.

The vascular flow annotation is not an additional labeling feature meant to simplify identification. Rather, it serves as a bounding box that identifies the specific area of the image being examined by the ultrasound flowmeter. However, to keep things simple, we will continue to call it a form of annotation. The presence of this annotation indicates that the relevant examination has been conducted.

To synthesize a Noise2Noise training dataset for the above annotations, we first manually extracted the necessary annotation icons from existing annotated data, and then we randomly overlay different annotations on the clean images we have. To improve the model's ability to handle variations (generalization), we also introduced randomness into the shape of annotations. For instance, the lines connecting markers in vascular flow annotations have a random, constantly changing appearance. This approach accounts for the different annotation styles used by various ultrasound machines. The randomness of the noise overlay allows for the creation of a relatively large dataset.

By constructing training datasets in the above-mentioned process, each noisy image has three corresponding images for different tasks.

- A clean image which the noisy image originated from.
- A different noisy image is created from the same clean image, using a different (in terms of position, form, etc.) noise sampled from the same distribution.
- A binary image recorded the position and form of the noise appended to the clean image.

An instance of the training dataset is presented in Figure 2. Using these images, the same dataset can be used for Noise2Noise training, conventional Noise2Clean training, and normal segmentation training.

Our approach to creating this training dataset can minimize the amount of human labor required. Even with a limited amount of clean data, we can generate a large noisy dataset for training. The flow chart of the above process is also shown in Figure 2.

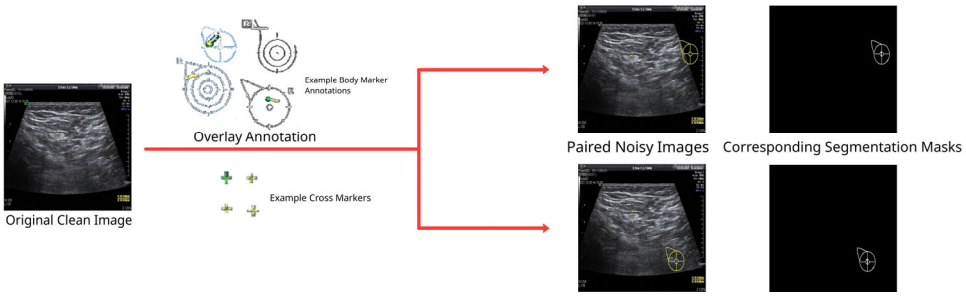


Figure 2. Flow chart of training dataset building.

3.2. Network Structures

In this research, we trained several structures to find the optimal solution and compare the two different training schemes: Noise2Noise and traditional Noise2Clean.

We adopted most of the structures from the traditional image segmentation model. The models we adopted include FCN, DeepLabv3, LinkNet, MANet, U-Net Plus Plus, MultiResUNet and a costumed U-Net.

FCN is one of the models utilizing convolutional networks in semantic segmentation. Long et al. [10,11] use fully-convolutional layers instead of fully-connected layers so that this model is compatible with non-fixed sized input and ouputs.

DeepLabv3 is a subsequent model of the DeepLab model family, developed by Chen et al. [12]. The main feature of this model is the use of dilated convolution, also known as "atrous" convolution. This method is advocated to combat the issue of feature resolution reduction in deep convolutional networks (due to pooling operations and strides in convolution operations) and the difficulties in multi-scale segmentation.

LinkNet is proposed by Chaurasia and Culurciello [13] to address the problem of the long processing time of most segmentation models. By using a skip connection to pass spatial information directly to the corresponding decoder, LinkNet manages to preserve low-level information without additional parameters and re-learning operations.

MANet, or Multi-scale Attention Net, is developed to improve accuracy in semantic segmentation of remote sensing images. By using a novel attention mechanism, treating attention as a kernel function, Li et al. [14,15] reduce the complexity of the dot product attention mechanism to $O(N)$.

U-Net is a well-known encoder-decoder segmentation model. It is originally proposed by Ronneberger et al. [16,17] for segmenting biological microscopy images.

U-Net++ is a variant of U-Net proposed by Zhou et al. [18]. In their work, they proposed a novel skip connection block in which a dense convolution block is used to process the input from the encoder feature map so that the semantic level of the input is closer to the corresponding decoder feature map.

MultiResUNet is another modern variant of U-Net proposed by Ibtehaz and Rahman [19] as a potential successor. They used an Inception-like layer to replace the consecutive convolution layers after each pooling and transpose-convolution layers, to percept objects at different scales. They adopted a chain of convolution layers with residual connections instead of plain skip connection to process the feature map inputs before concatenating them to decoder feature maps.

In our work, since the vanilla U-Net does not match the spatial resolution of our dataset, we used a costumed U-Net similar to [8] in all of our tests. Our architecture utilizes convolutional layers with strategically chosen stride and padding values to maintain consistent spatial dimensions between the network's input and output. Within the costumed U-Net implementation employed in this work, the encoder stage leverages 3x3 convolutional kernels with a stride of 2 and padding of 1. This configuration progressively increases the feature map dimensionality (from 3 to 38, 96, and finally 144) while downsampling the spatial resolution. The corresponding decoder stage mirrors these convolutional layers to achieve dimensionality reduction (from 144 to 96, 38, and finally 3). To achieve upsampling within the decoder, transposed convolutional layers are employed with parameters identical to their corresponding counterparts in the encoder. ReLU activation is utilized as the non-linearity after each convolutional or transposed convolutional layer, except for the final output layer which employs LeakyReLU activation. This specific configuration ensures that the processed data retains its original spatial dimensions throughout the costumed U-Net architecture. The detailed structure is presented in Table 1.

Table 1. Detailed structure of our costumed U-Net.

Layer Name	N_out (Channels)	Function
Input	3	-
Conv2d	48	Convolution (3 × 3 kernel, Stride 1 × 1, Padding 1 × 1)
ReLU	48	ReLU activation
Conv2d	48	Convolution (3 × 3 kernel, Stride 1 × 1, Padding 1 × 1)
ReLU	48	ReLU activation
MaxPool2d	48	Max Pooling (2 × 2 kernel, Stride 2 × 2, Padding 0 × 0)
Conv2d	48	Convolution (3 × 3 kernel, Stride 1 × 1, Padding 1 × 1)
ReLU	48	ReLU activation
MaxPool2d	48	Max Pooling (2 × 2 kernel, Stride 2 × 2, Padding 0 × 0)
Conv2d	48	Convolution (3 × 3 kernel, Stride 1 × 1, Padding 1 × 1)
ReLU	48	ReLU activation
ConvTranspose2d	48	Transposed Convolution (3 × 3 kernel, Stride 2 × 2, Padding 1 × 1, Output Padding 1 × 1)
Concat	96	Concatenate Feature Maps
Conv2d	96	Convolution (3 × 3 kernel, Stride 1 × 1, Padding 1 × 1)
ReLU	96	ReLU activation
Conv2d	96	Convolution (3 × 3 kernel, Stride 1 × 1, Padding 1 × 1)
ReLU	96	ReLU activation
ConvTranspose2d	96	Transposed Convolution (3 × 3 kernel, Stride 2 × 2, Padding 1 × 1, Output Padding 1 × 1)

Concat	144	Concatenate Feature Maps
Conv2d	96	Convolution (3 × 3 kernel, Stride 1 × 1, Padding 1 × 1)
ReLU	96	ReLU activation
Conv2d	96	Convolution (3 × 3 kernel, Stride 1 × 1, Padding 1 × 1)
ReLU	96	ReLU activation
ConvTranspose2d	96	Transposed Convolution (3 × 3 kernel, Stride 2 × 2, Padding 1 × 1, Output Padding 1 × 1)
Concat	99	Concatenate Feature Maps
Conv2d	64	Convolution (3 × 3 kernel, Stride 1 × 1, Padding 1 × 1)
ReLU	64	ReLU activation
Conv2d	32	Convolution (3 × 3 kernel, Stride 1 × 1, Padding 1 × 1)
ReLU	32	ReLU activation
ConvTranspose2d	3	Transposed Convolution (3 × 3 kernel, Stride 2 × 2, Padding 1 × 1, Output Padding 1 × 1)
LeakyReLU	3	LeakyReLU activation

4. Results

In this section, we provide performance evaluations and comparative studies.

4.1. Evaluation

We evaluate the model's performance based on segmentation precision and reconstruction similarity.

4.1.1. Segmentation Precision

In terms of noise reduction precision, for a typical segmentation model, we can use the output to compare it with a binary image known as the truth mask to compute a score based on the number of pixels that get classified into the right categories. For a restoration model like ours, we subtract the model output from the model input to compute the binary segmentation result. We compare the results with the segmentation truth mask to compute the Dice, IoU, and Pixel Accuracy (PA).

4.1.2. Reconstruction Similarity

For assessing reconstruction similarity, we use two metrics: Structural Similarity Index Measure (SSIM) and PSNR_HVS_M. SSIM is a commonly used measure of image similarity. The PSNR metric known as PSNR_HVS_M [20] is considered to be a more accurate representation of image quality, which takes into consideration the Contrast Sensitivity Function (CSF) and the between-coefficient contrast masking of Discrete Cosine Transform (DCT) basis functions.

4.2. Training

The neural networks discussed in the previous section were trained using PyTorch 1.10.1. RMSprop [21], a variant of stochastic gradient descent that divides gradients by an average of their recent magnitude, was used as the optimizer with a learning rate of 0.00001, momentum of 0.9, weight decay of $1e^{-8}$, and default values [22] for other parameters.

Three datasets were created in the aforementioned process to train various denoising models. For body marker annotation, a dataset of 83,900 pairs of noisy images generated from 4,975 clean images was used. For radial line annotation, 80,000 pairs of noisy images were generated from 3,936 clean images. For vascular flow annotation, 80,000 pairs of noisy images were generated from 250 clean images.

4.3. Optimal Model Structure

To find the most effective combination of network structure and training scheme for the given task, we trained different network structures under the Noise2Noise and Noise2Clean schemes using the body mark annotation dataset. Though utilizing only one type of annotation, this experiment's results could demonstrate the likely most suitable structure for other annotations as well. L_1 loss is used to train these models. The results were compared using segmentation precision and reconstruction similarity, and are presented in Tables 2 and 3.

We observed that Noise2Noise training scheme improves segmentation precision and reconstruction similarity in most cases. The results presented in Tables 2 and 3 indicate that the models trained using the

Noise2Noise scheme generally achieved higher Dice scores, IoU scores, PA scores, and PSNR_HVS_M scores. Specifically, for the costumed U-Net, we observed an increase in the Dice and IoU of 0.151 and 0.155, respectively, and an increase of 11.625 for the PSNR_HVS_M when using linearly normalized input.

According to our hypothesis, the Noise2Noise training process improves the model's ability to understand the features of annotations through solving an "impossible" task of relocating the annotation. This task is essentially a self-supervised pretext training task that helps the model gain a better understanding of the annotations and the spatial structure of the ultrasonic images, thus gaining higher performance. To highlight the advantage of our approach, let's consider the limitations of traditional Noise2Clean denoising. In Noise2Clean, the neural network learns convolutional kernels to remove noise. These kernels may develop a complex mask that essentially averages pixels within an applied area. This approach circumvents the need for the model to learn the specific relationship between the target noise and the underlying clean image. The acquisition of structural features by the model is not guaranteed. During training, the model may converge on an optimal solution that captures these features, leading to successful performance. However, the possibility exists that the model converges at a local optimum, neglecting this information even with abundant training data. Conversely, our Noise2Noise approach with overlaid annotations trains the model to essentially move the annotation (treated as noise) within the image. This process necessitates learning the structural information of both the annotation and the underlying clean background. It's well-established that core self-supervised learning tasks, such as rotation prediction, jigsaw puzzle solving, and missing patch prediction, all hinge on the model's ability to grasp structural information. Our proposed task shares this very property. We posit that this fundamental difference in training objectives is a key factor contributing to the performance improvement observed in our method.

We also noted that the costumed U-Net structure performed the best out of all the structures tested. It achieved the highest Dice, IoU, SSIM, and PSNR_HVS_M scores under both training schemes. The costumed U-Net trained using the Noise2Noise scheme achieved the highest segmentation precision and reconstruction similarity of all models, with a Dice of 0.712, an IoU of 0.596, an SSIM of 0.967, and a PSNR_HVS_M of 41.628.

Our findings suggest that a model's capacity to retain graphical details is a critical performance factor. As shown in Tables 2 and 3, a significant performance gap exists between models employing skip connections (facilitating detail preservation) and those lacking such structures. Interestingly, our results indicate that a concise skip connection pathway is preferable for this task. Models with intricate skip connection architectures (such as U-Net++ and MultiResUNet) exhibited lower performance compared to U-Net's straightforward skip connections. This might be attributed to the desired outcome: preserving most of the input information in the output, only removing annotation in an area of interest. Therefore, simpler skip connection pathways are preferred. Complex architectures introduce additional weights and parameters, potentially hindering the model's ability to faithfully transmit the input information.

Given the above results, we chose the costumed U-Net as the optimal model for later experiments.

Table 2. Segmentation Precision on Body Marker Annotation (Average + Var) N2C stands for Noise2Clean, N2N stands for Noise2Noise SMN indicates the model is trained with data normalized according to standard deviation and mean Models without SMN are trained with linearly normalized data.

Method	Training Mode	Dice	IoU	PA
FCN_101	N2C SMN	0.07 ± 0.003	0.039 ± 0.001	$0.97 \pm 7.2 \times e^{-5}$
FCN_101	N2N SMN	0.07 ± 0.003	0.04 ± 0.001	$0.97 \pm 8 \times e^{-5}$
DeepLab V3	N2C	0.073 ± 0.003	0.039 ± 0.001	0.969 ± 0.005
DeepLab V3	N2N	0.074 ± 0.003	0.04 ± 0.001	0.969 ± 0.005
LinkNet	N2C	0.447 ± 0.105	0.346 ± 0.007	0.976 ± 0.008
LinkNet	N2N	0.343 ± 0.139	0.280 ± 0.106	0.938 ± 0.008
MANet	N2C	0.531 ± 0.113	0.430 ± 0.091	0.943 ± 0.015
MANet	N2N	0.543 ± 0.128	0.451 ± 0.105	0.917 ± 0.024
U-Net++	N2C	0.551 ± 0.08	0.437 ± 0.07	0.983 ± 0.007
U-Net++	N2N	0.613 ± 0.114	0.516 ± 0.09	0.943 ± 0.016
MultiResUNet	N2C SMN	0.416 ± 0.05	0.594 ± 0.04	$0.998 \pm 2.75 \times e^{-6}$
MultiResUNet	N2N SMN	0.661 ± 0.06	0.539 ± 0.06	$0.99 \pm 5 \times e^{-4}$
Costumed U-Net	N2C SMN	0.408 ± 0.05	0.286 ± 0.04	$0.998 \pm 2.54 \times e^{-6}$
Costumed U-Net	N2N SMN	0.676 ± 0.05	0.552 ± 0.05	$0.999 \pm 5 \times e^{-7}$
Costumed U-Net	N2C	0.561 ± 0.077	0.441 ± 0.072	0.990 ± 0.005
Costumed U-Net	N2N	0.712 ± 0.053	0.596 ± 0.058	0.993 ± 0.007

Table 3. Reconstruction Similarity on Body Marker Annotation (Average + Var).

Model	Training Mode	SSIM	PSNR_HVS_M
FCN_101	N2C	0.459 ± 0.001	10.264 ± 1.751
FCN_101	N2N	0.453 ± 0.016	10.181 ± 2.430
DeepLab V3	N2C	0.680 ± 0.004	15.919 ± 2.578
DeepLab V3	N2N	0.678 ± 0.005	15.827 ± 3.282
LinkNet	N2C	0.933 ± 0.000	25.691 ± 6.425
LinkNet	N2N	0.945 ± 0.000	26.307 ± 8.466
MANet	N2C	0.923 ± 0.002	21.920 ± 7.015
MANet	N2N	0.923 ± 0.002	23.027 ± 3.903
U-Net++	N2C	0.923 ± 0.000	21.245 ± 1.846
U-Net++	N2N	0.927 ± 0.000	24.366 ± 7.121
MultiResUNet	N2C SMN	0.856 ± 0.002	23.712 ± 3.936
MultiResUNet	N2N SMN	0.792 ± 0.004	21.256 ± 6.160
Costumed U-Net	N2C SMN	0.833 ± 0.003	11.828 ± 20.299
Costumed U-Net	N2N SMN	0.791 ± 0.004	20.746 ± 10.223
Costumed U-Net	N2C	0.961 ± 0.000	29.976 ± 30.140
Costumed U-Net	N2N	0.967 ± 0.000	41.628 ± 41.775

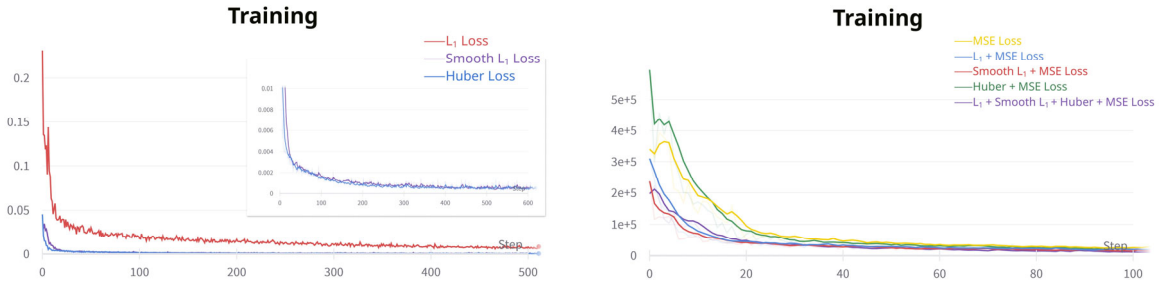
4.4. Optimal Loss Function

To find the optimal loss function, we evaluate the convergence speed of different loss functions. The loss functions we tested include L_1 loss, Huber loss, Smooth L_1 loss, MSE loss and several combinations of aforementioned loss functions. The result is shown in Figure 3.

To better visualize the differences in convergence speed between the losses, we present them in separated subplots. As shown in Figure 3a, the L_1 loss and its variants (Huber loss and Smooth L_1 loss) are displayed on one subplot, while the MSE loss-related losses are presented on another subplot in Figure 3b.

We observed that implementing MSE loss results in faster convergence, allowing the model to reach convergence in under 100 steps, as shown in Figure 3b. Meanwhile, as depicted in Figure 3a, the loss functions based on L_1 loss achieve a much slower convergence after approximately 500 to 600 steps. Although Huber loss and Smooth L_1 loss seem to have a quicker rate of convergence, closer examination in Figure 3a reveals that they both take around 500 steps to converge, which is similar to the standard L_1 loss.

We also noted from Figure 3b that using a combination of MSE loss and different L_1 based losses doesn't significantly affect the rate of convergence, likely because the difference in scale between the MSE loss and L_1 loss and its variants causes MSE loss to remain the primary determinant of convergence speed.



(**a**) (**b**)

Figure 3. Loss functions convergence comparison: (**a**) Loss of L_1 and its variants. (**b**) Loss of MSE loss and other combined losses.

Our study also conducted an evaluation of the costumed U-Net trained using various loss functions. Our findings in Tables 4 and 5 revealed that there was minimal difference between the performances of these models, with the largest discrepancies in Dice, IoU, PA, SSIM and PSNR_HVS_M amounting to 0.023, 0.019, 0.003, 0.011 and 4.031, respectively. These outcomes suggest that the selection of alternative loss functions has little influence on the overall performance of the model. As such, we decided not to employ the MSE loss function in subsequent experiments and instead continued to utilize the L_1 loss.

Table 4. Segmentation Precision on Body Marker Annotation for the Costumed U-Net Trained with Different Loss Functions (Average + Var).

Loss Function	Dice	IoU	PA
L 1	0.712 ± 0.053	0.596 ± 0.058	0.993 ± 0.007
Huber	0.708 ± 0.05	0.592 ± 0.005	0.993 ± 0.005
Smooth L1	0.717 ± 0.05	0.599 ± 0.055	0.993 ± 0.005
L2	0.716 ± 0.053	0.599 ± 0.056	0.993 ± 0.005
L1 + L2	0.712 ± 0.053	0.596 ± 0.057	0.993 ± 0.005
Huber + L2	0.713 ± 0.052	0.596 ± 0.057	0.993 ± 0.005
Smooth L1 + L2	0.692 ± 0.068	0.580 ± 0.066	0.990 ± 0.005
All Loss Sum	0.715 ± 0.052	0.598 ± 0.057	0.993 ± 0.005

Table 5. Reconstruction Similarity on Body Marker Annotation for the Costumed U-Net Trained with Different Loss Functions (Average + Var).

Loss Function	SSIM	PSNR_HVS_M
L 1	0.967 ± 0.000	41.628 ± 41.775
Huber	0.968 ± 0.000	38.110 ± 91.416
Smooth L1	0.967 ± 0.000	41.737 ± 38.719
L2	0.966 ± 0.000	40.982 ± 37.608
L1 + L2	0.966 ± 0.000	38.689 ± 46.355
Huber + L2	0.977 ± 0.000	42.141 ± 53.215
Smooth L1 + L2	0.968 ± 0.000	39.186 ± 47.249
All Loss Sum	0.968 ± 0.000	40.443 ± 57.084

4.5. Noise2Noise with Other Annotations

The improvement observed in the costumed U-Net trained using the Noise2Noise scheme is also apparent in other annotation datasets, as shown in Tables 6–9. In the provided tables, the costumed U-Net has been trained using other two annotation datasets along with two different training schemes. The outcomes show a substantial enhancement in comparison to the Noise2Clean models, as there is approximately a half-unit gain observed in both Dice and IoU metrics, an increase of around 0.01 in SSIM, and a rise of 5 units in PSNR_HVS_M for both types of annotations.

The performance improvement observed in the Noise2Noise model further strengthens our hypothesis. This is because both radial line annotations (cross markers) and vascular flow annotations (boxes) share similarities with other highly prevalent elements in ultrasonic imaging results. Models trained with the traditional Noise2Clean approach struggle to develop kernels that can differentiate these desired annotations from other image information. Conversely, the Noise2Noise model circumvents this limitation.

Table 6. Segmentation Precision on Radial Line Annotation (Average + Var).

Method	Training Mode	Dice	IoU
Costumed U-Net	N2C	0.226 ± 0.013	0.132 ± 0.006
Costumed U-Net	N2N	0.747 ± 0.004	0.639 ± 0.059

Table 7. Reconstruction Similarity on Radial Line Annotation (Average + Var).

Method	Training Mode	SSIM	PSNR_HVS_M
Costumed U-Net	N2C	0.932 ± 0.000	21.660 ± 5.391
Costumed U-Net	N2N	0.942 ± 0.000	26.376 ± 0.681

Table 8. Segmentation Precision on Vascular Flow Annotation (Average + Var).

Method	Training Mode	Dice	IoU	PA
Costumed U-Net	N2C	0.243 ± 0.028	0.149 ± 0.013	0.989 ± 1.115
Costumed U-Net	N2N	0.728 ± 0.031	0.599 ± 0.039	0.998 ± 1.423 × e^{-5}

Table 9. Reconstruction Similarity on Vascular Flow Annotation (Average + Var).

Method	Training Mode	SSIM	PSNR_HVS_M
Costumed U-Net	N2C	0.938 ± 0.000	21.584 ± 5.384
Costumed U-Net	N2N	0.948 ± 4.853 × e−5	26.717 ± 0.511

4.6. Qualitative Results

In this section, we present denoised images from models trained under different schemes to further support our claim.

As can be seen in Figures 4–6, the output from the Noise2Clean model contains obvious artifacts, whereas models trained using the Noise2Noise scheme do not suffer from this problem.

It is also worth noting that in the output images from Noise2Clean models, information in the edge area is compromised. In contrast, the Noise2Noise models preserve this information well. The evidence implies that models trained with the Noise2Noise scheme possess superior capabilities in identifying and distinguishing noise.

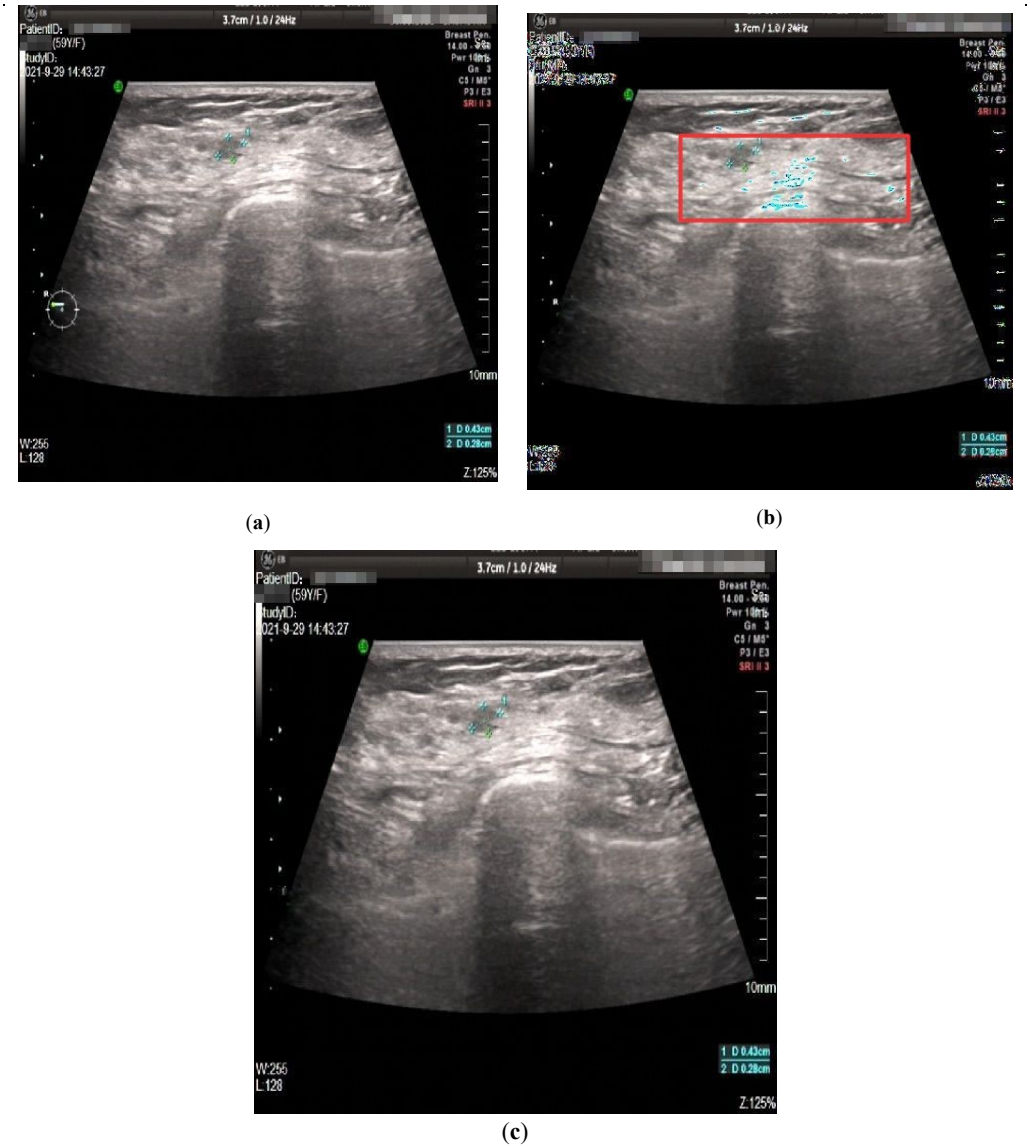
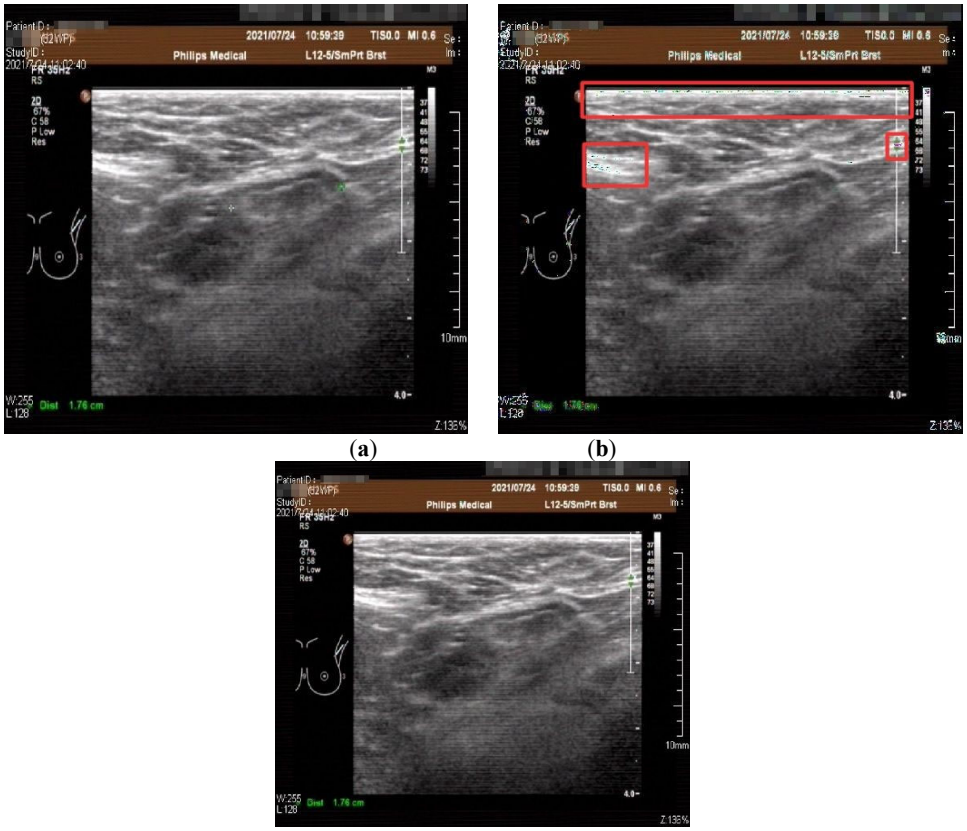


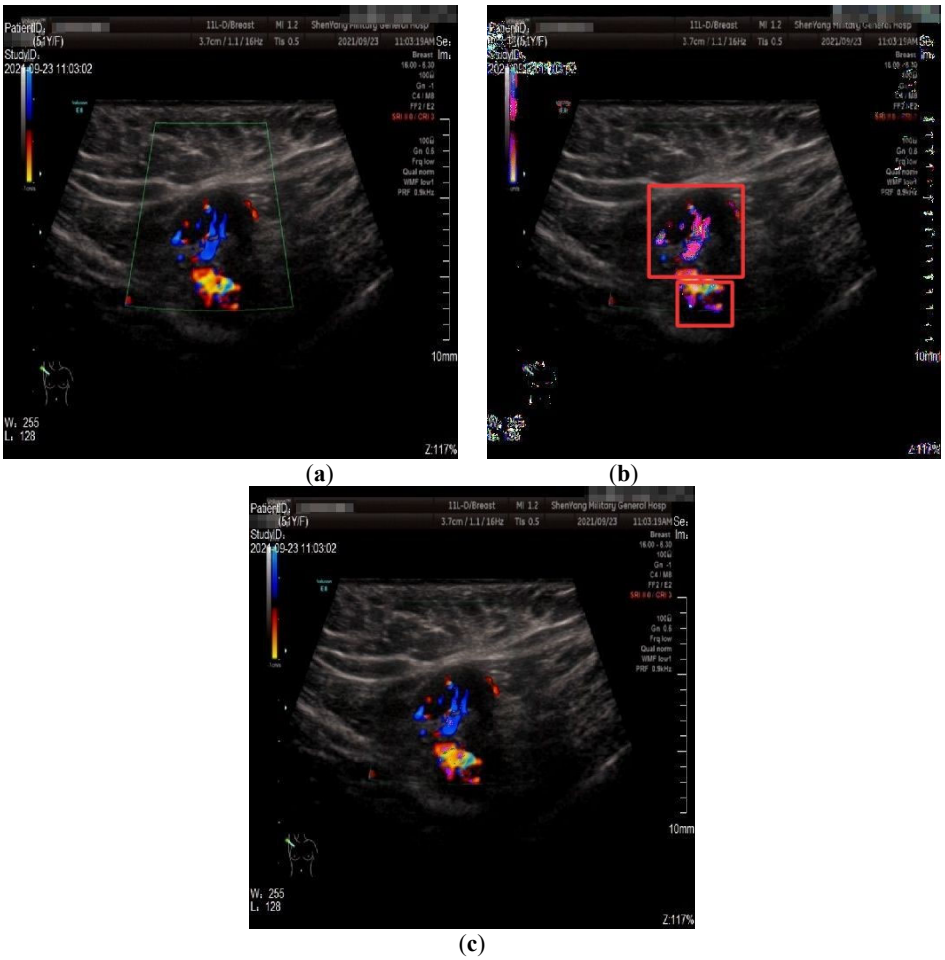
Figure 4. Body marker annotation: (**a**) input image; (**b**) output from N2C model; (**c**) output from N2N model.



(**a**) (**b**)

(**c**)

Figure 5. Radial line annotation: (**a**) input image; (**b**) output from N2C model; (**c**) output from N2N model.



(**a**) (**b**)

(**c**)

Figure 6. Vascular flow annotation: (**a**) input image; (**b**) output from N2C model; (**c**) output from N2N model.

5. Discussion and Conclusions

This study proposed a self-supervised data generation and training approach to build large and diverse datasets starting from a small dataset with only a few clean images. We find that the costumed U-Net trained with the Noise2Noise scheme outperformed other models in terms of segmentation precision and reconstruction similarity in the annotation removal task. The benefits of Noise2Noise training were observed across most model structures tested, and the models trained using this scheme produced fewer artifacts.

Our study has some limitations: Firstly, we used separate parameter sets for the segmentation task of different annotations. However, with the recent advancement of deep learning theories, it is now possible to use a single parameter set for the segmentation of all annotations presented in the image. Additionally, there is potential for further research in the area of language-guided segmentation models, which would provide a more precise and flexible interface for medical professionals. We find building a model that incorporates these innovations intriguing.

We also noted that our model was trained in a self-supervised manner, meaning it has potentially gained a strong understanding of the structural features of ultrasonic images. This understanding is beneficial for downstream models such as the object detection model. Different ways of fine-tuning, like Low-Rank Adaptation (LoRA), adapter layers, etc. should be explored to find the optimal method to effectively transfer this understanding. We plan to address these issues in future studies.

Author Contributions

All authors contributed to the study's conception and design. Y.T. developed initial experiment setting. Z.X. refined the experiment details. Material preparation and data collection were performed by N.J., J.C. The first draft of the manuscript and the codebase was written by Y.Z. All authors have read and agreed to the published version of the manuscript.

Funding

This work was supported by the Natural Science Foundation of Liaoning Province (Grant numbers 2022-MS-114). It was also supported by the Key R&D Plan Projects of Liaoning Province (Grant numbers 2020JH2/10300122).

Institutional Review Board Statement

No ethical approval is needed for this study, as it does not involve any human or animal subjects.

Informed Consent Statement

No consent is needed for this study, as it does not involve any human subjects.

Data Availability Statement

The data that support the findings of this study are available, upon reasonable request, from the corresponding authors. The data are not publicly available due to their containing information that could compromise the privacy of the patients to whom these ultrasound imaging results pertain. We released our code at https://github.com/ZhangYH-Z1RZcigZrw78AD-TR59O/UltrasonicImage-N2N-Approach.

Conflicts of Interest

The authors have no relevant financial or non-financial interests to disclose.

References

1. Kulshrestha, A.; Singh, J. Inter-hospital and intra-hospital patient transfer: Recent concepts. *Indian J. Anaesth.* **2016**, *60*, 451–457.
2. Jackson, P.; Chenal, C. Ultrasonic Imaging System with Body Marker Annotations. Google Patents. US Patent 9,713,458, 25 July 2017.
3. Li, H.; Xiong, P.; An, J.; Wang, L. Pyramid attention network for semantic segmentation. *arXiv* **2018**, arXiv:1805.10180.
4. Huang, Q.; Xia, C.; Wu, C.; Li, S.; Wang, Y.; Song, Y.; Kuo, C.-C.J. Semantic segmentation with reverse attention. *arXiv* **2017**, arXiv:1707.06426.

5. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial networks. *Commun. ACM* **2020**, *63*, 139–144.
6. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; Berg, A.C. *Ssd: Single Shot Multibox Detector*; Springer: Amsterdam, The Netherlands, 2016; pp. 21–37.
7. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
8. Lehtinen, J.; Munkberg, J.; Hasselgren, J.; Laine, S.; Karras, T.; Aittala, M.; Aila, T. Noise2noise: Learning image restoration without clean data. *arXiv* **2018**, arXiv:1803.04189.
9. Kashyap, M.M.; Tambwekar, A.; Manohara, K.; Natarajan, S. Speech denoising without clean training data: a noise2noise approach. *arXiv* **2021**, arXiv:2104.03838.
10. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3431–3440.
11. Minaee, S.; Boykov, Y.Y.; Porikli, F.; Plaza, A.J.; Kehtarnavaz, N.; Terzopoulos, D. Image segmentation using deep learning: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022***, 44,* 3523–3542.
12. Chen, L.-C.; Papandreou, G.; Schroff, F.; Adam, H. Rethinking atrous convolution for semantic image segmentation. *arXiv* **2017**, arXiv:1706.05587.
13. Chaurasia, A.; Culurciello, E. Linknet: Exploiting encoder representations for efficient semantic segmentation. In Proceedings of the 2017 IEEE Visual Communications and Image Processing (VCIP), St. Petersburg, FL, USA, 10–13 December 2017; pp.1–4.
14. Li, R.; Zheng, S.; Zhang, C.; Duan, C.; Su, J.; Wang, L.; Atkinson, P.M. Multiattention network for semantic segmentation of fine-resolution remote sensing images. *IEEE Trans. Geosci. Remote Sens.* **2021**, *60*, 1–13.
15. Iakubovskii, P. Segmentation Models Pytorch. Available online: https://github.com/qubvel/segmentation_models.pytorch (accessed on 17 April 2024).
16. Chlap, P.; Min, H.; Vandenberg, N.; Dowling, J.; Holloway, L.; Haworth, A. A review of medical image data augmentation techniques for deep learning applications. *J. Med. Imaging Radiat Oncol.* **2021**, *65*, 545–563.
17. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation, In *Proceedings of the Medical Image Computing and Computer-Assisted Intervention—MICCAI 2015: 18th International Conference, Munich, Germany, 5–9 October 2015*; Springer: Berlin, Germany; pp. 234–241.
18. Zhou, Z.; Rahman Siddiquee, M.M.; Tajbakhsh, N.; Liang, J. Unet++: A nested u-net architecture for medical image segmentation. In *Proceedings of the Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support: 4th International Workshop, DLMIA 2018, and 8th International Workshop, ML-CDS 2018, Granada, Spain, 20 September 2018*; Springer: Berlin, Germany, 2018; pp. 3–11.
19. Ibtehaz, N.; Rahman, M.S. Multiresunet: Rethinking the u-net architecture for multimodal biomedical image segmentation. *Neur. Netw.* **2020**, *121*, 74–87.
20. Ponomarenko, N.; Silvestri, F.; Egiazarian, K.; Carli, M.; Astola, J.; Lukin, V. On between-coefficient contrast masking of dct basis functions. In Proceedings of the Third International Workshop on Video Processing and Quality Metrics for Consumer Electronics, Scottsdale, AZ, USA, 13–15 January 2007; Volume 4.
21. Tieleman, T.; Hinton, G. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neur. Netw. Mach. Learn.* **2012**, *4*, 26–31.
22. Foundation, T.P. RMSprop. Available online: https://pytorch.org/docs/stable/generated/torch.optim.RMSprop.html#torch.optim.RMSprop (accessed on 12 October 2022).

AI Medicine
https://www.sciltp.com/journals/aim

Article

A Comparative Study of Deep Learning in Breast Ultrasound Lesion Detection: From Two-Stage to One-Stage, from Anchor-Based to Anchor-Free

Yu Wang [1], Qi Zhao [1], Baihua Zhang [2], Dingcheng Tian [1], Ruyi Zhang [1] and Wan Zhong [3,*]

[1] College of Medicine and Biological Information Engineering, Northeastern University, Shenyang 110024 , China
[2] Research Institute for Medical and Biological Engineering, Ningbo University, Ningbo 315000, China
[3] General Hospital of Northern Theater Command, Shenyang 110024, China
* Correspondence: wzhong_88@163.com

How To Cite: Wang, Y.; Zhao, Q.; Zhang, B.; Tian, D.; Zhang, R.; Zhong, W. A Comparative Study of Deep Learning in Breast Ultrasound Lesion Detection: From Two-Stage to One-Stage, from Anchor-Based to Anchor-Free. *AI Medicine* **2024**, *1*(1), 5. https://doi.org/10.53941/aim.2024.100005.

Received: 16 July 2024
Revised: 26 August 2024
Accepted: 27 August 2024
Published: 4 September 2024

Abstract: Breast cancer is one of the most common tumors among women in the world, and its early screening is crucial to improve the survival rate of patients. Breast ultrasound, with the characteristics of non radiation, real-time imaging and easy operation, has become a common method for breast cancer detection. However, this method has some problems, such as low imaging quality and strong subjectivity of diagnosis results, which affect the accurate diagnosis of breast cancer. With the ongoing advancement of deep learning technology, intelligent breast cancer detection systems have effectively overcome these challenges, enhancing diagnostic accuracy and efficiency. This study uses nine popular deep learning object detection networks (including two-stage, one-stage, anchor-based, and anchor-free networks) for the detection of breast lesions and compares the results of these methods. The experiments show that the anchor-based Single Shot MultiBox Detector (SSD) network excels in overall performance, while the anchor-free Fully Convolutional One-stage Object Detector (FCOS) exhibits the best generalization ability. Moreover, the results also indicate that, in the context of breast lesion detection, anchor-based networks generally outperform anchor-free networks.

Keywords: deep learning; breast ultrasound image; breast cancer; breast lesion detection; object detection

1. Introduction

According to the global cancer statistics report of 2018, 11.6% of cancer patients worldwide are diagnosed with breast cancer, making it the second most prevalent cancer globally [1]. Each year, approximately 2.89 million women are diagnosed with breast cancer, accounting for 24.2% of all female cancer cases [1]. Clinical studies show that the survival rate of breast cancer is closely related to the early detection and staging of the disease; the earlier it is detected, the higher the possibility of survival [2]. Therefore, early screening for breast cancer is crucial.

In clinic, the detection of breast cancer typically relies on three types of medical imaging technologies, Mammography, Digital Breast Tomosynthesis (DBT) and medical ultrasound imaging. Each of these technologies has its own advantages and unique limitations in breast cancer detection. While Mammography can reach a detection sensitivity of up to 85% in general female populations, its sensitivity decreases to 47.8–64.4% in women with dense breast tissue [3]. This is due to the lower distinction between breast tissue and tumors in dense breast, leading to potential missed-detection. Moreover, Mammography carries radiation risks and is relatively costly. DBT also faces similar issues of high costs and radiation exposure. In contrast, breast ultrasound imaging is a comparatively lower-cost, non-ionizing radiation method that provides real-time imaging. It performs well in detecting hidden

breast cancers in dense breast tissues [4], thus becoming an important tool in breast cancer detection. However, the diagnostic results of breast ultrasound largely depends on the doctor's skill and experience level. Variations in training backgrounds and clinical experiences can lead to different diagnostic results for the same ultrasound images [5]. Additionally, ultrasound images often suffer from issues like noise interference, strong artifacts, and low contrast between tissue structures [6].

To address the above issues, many researchers have conducted research on the automated diagnosis of breast ultrasound images. Breast cancer automatic diagnosis typically includes two steps: lesion detection and lesion classification. In earlier studies, researchers generally used traditional digital image processing methods for lesion recognition and classification. For instance, Drucker et al. [7] identified breast lesion areas using radial gradient index filtering in a study on breast cancer classification, and then input the identified areas into a Bayesian classifier for benign-malignant lesion classification. In another study, Liu et al. [8] on lesion area identification, they initially conducted a preliminary analysis of breast ultrasound images using texture features, followed by refining the coarse identification results with active contour method, achieving precise segmentation of breast lesions to assist subsequent lesion classification. With the advancement of artificial intelligence algorithms, machine learning algorithms have been increasingly applied in the automated diagnosis of breast cancer. For example, Shan et al. [9] first determined the approximate location of breast lesions using traditional image processing methods in a lesion segmentation study, then extracted frequency and spatial domain features of the lesion area, and fed these features into a shallow artificial neural network for feature analysis, obtaining precise segmentation results of breast lesions. However, shallow artificial neural networks based on traditional machine learning algorithms still have limited feature extraction capabilities and cannot meet the requirements for high-precision breast lesion detection and classification.

With the development of computer hardware and advancements in deep learning algorithms, coupled with the powerful feature extraction and analysis capabilities of deep neural networks, deep learning has achieved remarkable successes in various fields. Consequently, researchers have shifted from using traditional machine learning algorithms to deep learning algorithms for automated breast cancer diagnosis. As mentioned earlier, automated breast cancer diagnosis mainly includes lesion detection and classification, which aligns well with the task of object detection in deep learning. Therefore, many researchers have applied deep learning object detection methods to the automated detection of breast cancer. Yap et al. [10] used Faster R-CNN [11] for the identification of breast lesions in ultrasound images and achieved good breast cancer detection performance through transfer learning and multi-feature image fusion methods. In a study on breast lesion detection, Wang et al. [12] used segmentation-based image enhancement techniques to enhance the contrast of breast ultrasound images, then input them into a Fully Convolutional One-stage Object Detector (FCOS) [13], achieving a mean average precision (mAP) of 90.2%. Cao et al. [14] compared the performance of five anchor-based object detection methods in detecting lesions in breast ultrasound images, with the Single Shot MultiBox Detector (SSD) [15] network achieving the best accuracy and Recall. Mo et al. [16] improved the preset anchor size of You Only Look Once (YOLO) V3 [17] using clustering methods and applied it to breast ultrasound lesion detection, achieving an mAP of 89.34%. Yu et al. [18] presented GFNet, a novel framework for breast mass detection, which integrates patch extraction, feature extraction, and mass detection modules. GFNet demonstrates high robustness and adaptability across different imaging devices.

As previously mentioned, researchers have used various categories of object detection networks for the automated detection of breast cancer, including Two-Stage (Faster R-CNN), One-Stage (YOLO V3), Anchor-based (SSD), and Anchor-free (FCOS) networks. However, in past work, there has been a scarcity of comparative studies on the performance of these different categories of object detection networks in detecting breast lesions. In this paper, we select nine popular object detection algorithms, encompassing Two-Stage, One-Stage, Anchor-based, and Anchor-free categories, and conduct a comprehensive comparison of their performance in breast lesion detection. The nine object detection networks are Faster R-CNN (Two-Stage, Anchor-based), SSD (One-Stage, Anchor-based), YOLO V3 (One-Stage, Anchor-based), RetinaNet [19] (One-Stage, Anchor-based), YOLOF [20] (One-Stage, Anchor-based), CornerNet [21] (One-Stage, Anchor-free), FCOS, TTFNet [22] (One-Stage, Anchor-free), and YOLOX [23] (One-Stage, Anchor-free).

2. Materials and Methods

In this section, we will introduce the datasets and object detection networks used in this study.

2.1. Datasets

This study uses data from two public datasets, BUS dataset [24] and BUSI dataset [25], with the images from these datasets as shown in Figure 1. As shown Figure 1, we can observe that compared to BUS dataset, the

ultrasound images in BUSI dataset have lower grayscale values and also contain more noise.

The BUS dataset from the UDIAT Diagnostic Centre of the Parc Tauli Corporation, Sabadell (Spain), where images were collected using the Siemens ACUSON Sequoia C512 17L5 HD linear array sensor (8.5 MHz). BUS dataset contains 163 breast ultrasound images with varying original size, averaging 760 × 570, and each image includes one or more lesion areas. Of these 163 lesion images, 53 are malignant and 110 are benign. The malignant breast images include 40 cases of invasive ductal carcinoma, 4 cases of ductal carcinoma in situ, 2 cases of invasive lobular carcinoma, and 7 cases of other unspecified malignancies. In terms of benign breast images, there are 65 cases of unspecified cysts, 39 fibroadenomas, and 6 other types of benign lesions. All images were manually segmented and classified by radiologists, marking the lesion areas. Both the original breast images and the annotated images are saved in png format, and an xlsx file provides lesion type information for each image.

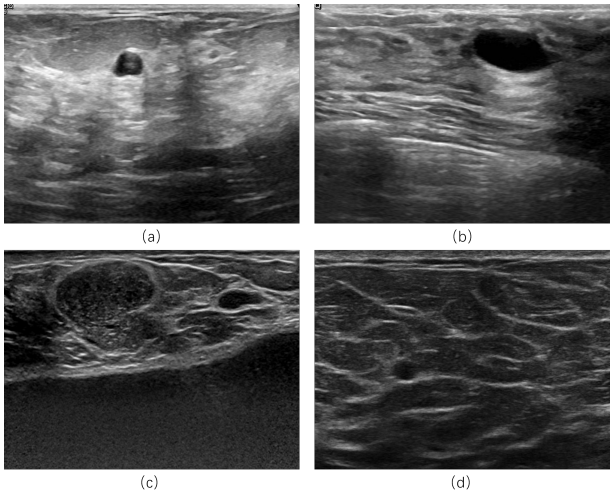


Figure 1. BUS and BUSI dataset images. (**a**,**b**) from BUS dataset, (**c**,**d**) from BUSI dataset.

BUSI dataset is from Baheya Hospital for Early Detection & Treatment of Women's Cancer, Cairo, Egypt, collected using the LOGIQ E9 and LOGIQ E9 Agile ultrasound systems. The breast ultrasound images were gathered from 600 female subjects aged between 25 and 75 years. Initially, this dataset contained a total of 1100 images. Each image's lesion area was manually segmented using Matlab software and classified as normal, benign, or malignant. However, after radiologists at Baheya Hospital removed duplicate and incorrectly annotated images, a total of 780 images remained, comprising 437 benign images, 210 malignant images, and 133 normal breast images (without lesions). Notably, the original size of BUSI images was 1280 × 1024, but due to the presence of large amounts of irrelevant areas in the original images, they were cropped to a size of 500 × 500 and saved in png format.

Both BUS dataset and BUSI dataset contain accurate labels for breast lesion edge segmentation and benign-malignant classification. However, these labels are not suitable for the labeling requirements of object detection task. Therefore, we reprocess the labels of both BUS dataset and BUSI dataset to make them appropriate for breast lesion detection task, as shown in Figure 2. We traverse the points of the breast lesion contours in Figure 2b to locate the top, bottom, left, and right endpoints and then determine the top-left and bottom-right points of the lesion area and to obtain height and width of the lesion, as depicted in Figure 2c.

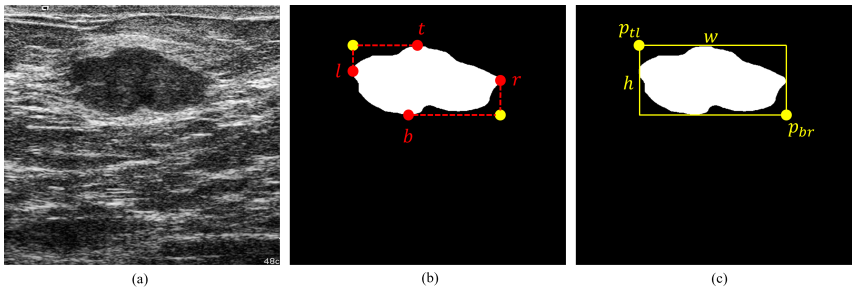


Figure 2. The process of creating labels for breast lesion detection. (**a**) Original ultrasound images; (**b**) ground truth in binary mask, yellow points represent the top-left and bottom-right corners of the ground truth; (**c**) represents a bounding box made according to the yellow points.

2.2. Deep Learning Neural Networks

Since the development of R-CNN [26], various highly accurate object detection networks based on deep learning have emerged. Generally, object detection networks can be categorized by the number of stages into two-stage and one-stage methods, or by the use of preset anchors into anchor-based and anchor-free methods. In this study, we select nine currently popular object detection networks and compare their performance in breast lesion detection tasks. The chosen networks include two-stage, one-stage, anchor-based, and anchor-free object detection methods, with specific descriptions of these networks provided in Table 1.

Table 1. Description of object detection networks.

Model	Number of Stage	Anchor Setting	Network Description
Faster R-CNN	Two-Stage	Anchor-based	Faster R-CNN introduced a Region Proposal Network to achieve real-time detection. Efficiency is improved through the sharing of convolutional features, and preset anchors are used to regress the position of the object, significantly enhancing detection speed and accuracy.
RetinaNet	One-Stage	Anchor-based	RetinaNet addresses the issue of class imbalance in object detection by introducing Focal Loss, which focuses on samples that are difficult to classify.
SSD	One-Stage	Anchor-based	SSD detects objects of various sizes effectively by predicting categories and bounding boxes on feature maps at multiple scales.
YOLO V3	One-Stage	Anchor-based	YOLO V3 can classify and locate in a single forward pass. It introduces multi-scale detection, using feature maps at three different scales to improve the detection of small objects.
YOLOF	One-Stage	Anchor-based	YOLOF simplifies the network structure by reducing the number of feature pyramid layers, maintaining high detection performance. This design lowers computational costs while increasing speed.
CornerNet	One-Stage	Anchor-free	CornerNet uses a corner detection method, locating objects by detecting their top-left and bottom-right corners.
FCOS	One-Stage	Anchor-free	FCOS predicts the size and center point of the object's bounding box directly on the feature map, offering a straightforward method to handle objects of various shapes and sizes.
TTFNet	One-Stage	Anchor-free	TTFNet uses a dense detection head and an efficient feature fusion strategy. It maintains high detection accuracy while significantly enhancing detection speed.
YOLOX	One-Stage	Anchor-free	YOLOX introduces an anchor-free design and decoupled head, and optimizes the label assignment strategy.

During the experimentation, we substitute the backbone of some networks to further compare the performance of different networks in breast lesion detection. We select ResNet [27], VGG [28], and DarkNet [29] as the backbones for most of the networks.

3. Results

In this section, we will introduce the performance metrics used in our experiments, the details of the experiments, and the performance results of each network. We chose the output results of the SSD network for demonstration, as shown in Figure 3. The network draws bounding boxes in different colors based on the predicted nature of the lesion, red for lesions predicted to be malignant and green for those predicted to be benign. The confidence level of the prediction is displayed above the bounding box.

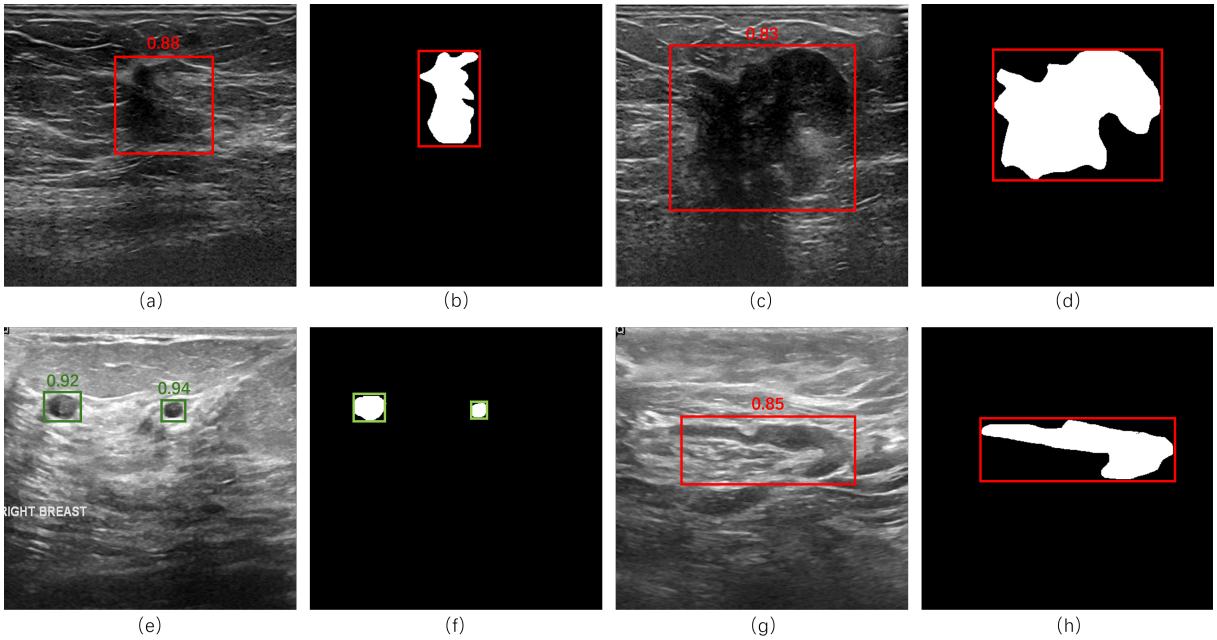


Figure 3. Breast lesion detection results of SSD network. (**a**,**c**,**e**,**g**) are prediction results. (**b**,**d**,**f**,**h**) are ground truth.

3.1. Performance Metrics

In this study, we use commonly used metrics in object detection, Average Precision (AP), Average Recall (AR), and Frames Per Second (FPS) as the performance metrics for our study.

AP represents the area under the Precision-Recall (PR) curve in object detection and is calculated based on the following values. First, it is necessary to compute the Intersection over Union (IoU) threshold (T) between the predicted and actual bounding boxes, as well as the confidence scores for the classification prediction of the bounding boxes. We have,

$$\text{IoU} = \frac{\text{Area of Overlap}}{\text{Area of Union}} \tag{1}$$

Then we have,

True Positives (TP): The prediction BBox with IoU $>$ T and meeting the category Confidence threshold.

False Positives (FP): The prediction BBox with IoU $<$ T and meeting the category Confidence threshold.

False Negatives (FP): The prediction BBox with IoU = 0.

Based on the TP, FP, and FN, we have,

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{2}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{3}$$

Based on different confidence thresholds for each category, we can plot the Precision-Recall (PR) curve, thereby determining the AP value. By adjusting various IoU thresholds, we can calculate AP50 (T $>$ 0.5) and AP75 (T $>$ 0.75). AR10 refers to the average recall rate when the IoU threshold is set to T $>$ 0.1.

3.2. Experiment Implementation

In this study, we implement all comparative networks using PaddleDetection [30]. Each network trains for 300 epochs, evaluating performance on the validation set after each epoch. The model parameters that with the best performance on the validation set during these 300 epochs are retained as the final parameters. During the training process, the first five epochs use model warm-up, and for the remainder of training, a cosine learning rate decay strategy [31] reduces the learning rate to one percent of the initial rate. We apply random rotation as a preprocessing method. The image size is 320 $\times$ 320.

We conduct model training on both the combined BUS+BUSI mixed-dataset and the single BUSI dataset. Both data groups are divided into training, validation, and test sets in an 8:1:1 ratio. All breast ultrasound images are resized to 320 $\times$ 320 with the learning rate set to 0.01 and the batch size set to 8, and during the training process, we use image augmentation methods such as random rotation, random flipping, and Mosaic [32].

3.3. Results of BUS+BUSI Mixed-Dataset

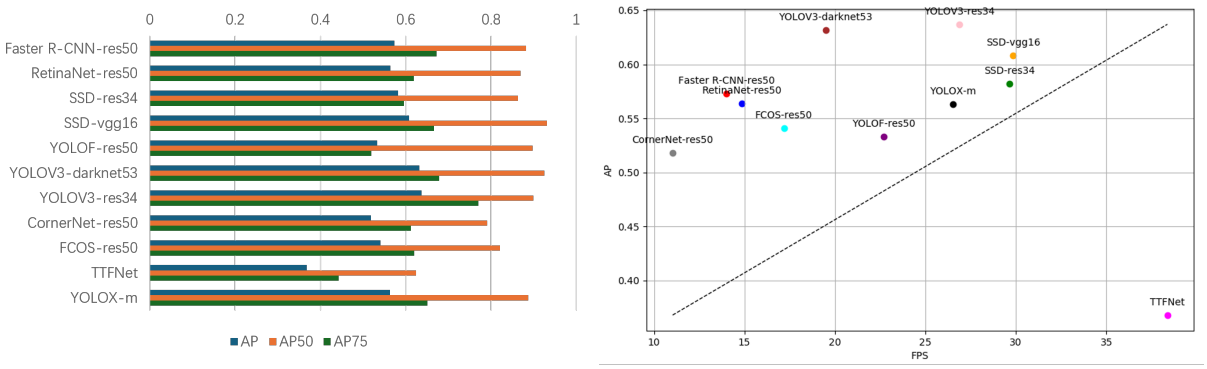
First, we evaluate the performance of object detection networks using BUS+BUSI mixed-dataset, with results shown in Table 2. From Table 2, we observe that within the anchor-based networks, YOLOV3-res34 performs best in terms of AP, reaching 0.637, and also leads in AP75 and AR10, indicating its advantages in accuracy. In terms of processing speed, SSD-vgg16 and SSD-res34, with nearly 30 FPS, outperform other networks. Additionally, SSD achieves the best result in AP50, indicating its excellent overall capabilities. Among the anchor-free networks, YOLOX-m leads with an AP of 0.563 and shows good performance in AP50, AP75, and AR10, exhibiting a balanced performance advantage. FCOS achieves slightly lower performance than YOLOX-m. On the other hand, although TTFNet reaches the highest FPS (38.37), it significantly behind in terms of accuracy.

Table 2. Performance comparison of different object detection networks on mixed-dataset.

	Model	AP	AP50	AP75	AR10	FPS
Anchor-based networks	Faster R-CNN-res50	0.573	0.882	0.672	0.677	14.01
	RetinaNet-res50	0.564	0.869	0.619	0.655	14.87
	SSD-res34	0.582	0.863	0.596	0.631	29.65
	SSD-vgg16	0.608	**0.931**	0.666	**0.691**	29.85
	YOLOF-res50	0.533	0.897	0.519	0.617	22.7
	YOLOV3-darknet53	0.632	0.925	0.678	0.683	19.5
	YOLOV3-res34	**0.637**	0.899	**0.77**	0.686	26.88
Anchor-free networks	CornerNet-res50	0.518	0.791	0.612	0.627	11.03
	FCOS-res50	0.541	0.821	0.62	0.629	17.18
	TTFNet	0.368	0.624	0.443	0.476	**38.37**
	YOLOX-m	0.563	0.887	0.651	0.69	26.53

Note: Bold font indicates the best performance results.

Figure 4 presents the performance results and AP-FPS plot of each network. In Figure 4b, the closer a network's performance is to the top-right corner, the stronger its overall performance. Overall, the two anchor-based object detection networks, YOLOV3 and SSD, show excellent performance, while the anchor-free networks are slightly behind the anchor-based networks in terms of performance.



(a) (b)

Figure 4. Networks performance results and AP-FPS plot on BUS+BUSI mixed-dataset. (**a**) is the performance results of different networks, (**b**) is the FPS and AP scatter plot of the networks.

3.4. Results of BUSI Dataset

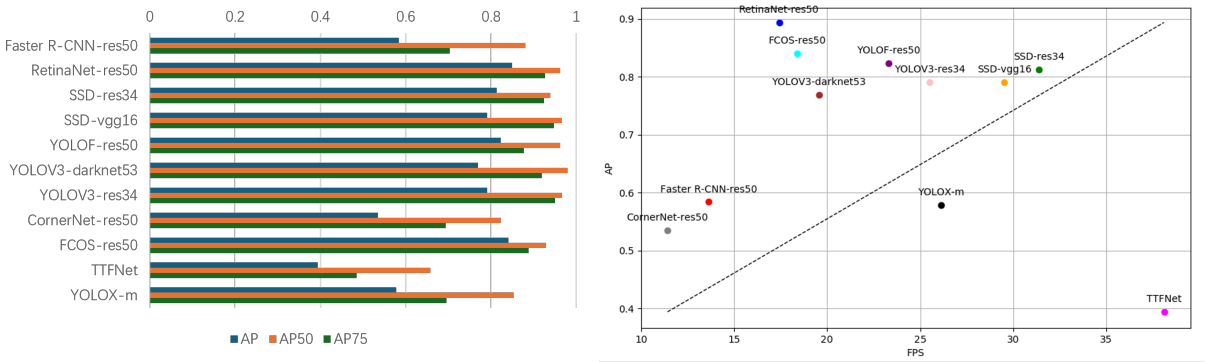
Next, we compare the performance of the nine networks on BUSI dataset, with results shown in Table 3. Among the anchor-based networks, RetinaNet performs the best on BUSI dataset, achieving the highest AP and AR, as well as the second-highest AP75, but it shows some disadvantages in network speed. YOLOV3 and SSD, which perform well on BUS+BUSI mixed-dataset, still show excellent performance on BUSI dataset, achieving balanced results in both accuracy and network speed. For anchor-free networks, FCOS achieves an AP of 0.841, close to the best-performing anchor-based model RetinaNet-res50, and it achieves the best results among anchor-free networks in AP50, AP75, and AR. However, YOLOX-m, which performs relatively well on the BUS+BUSI mixed-dataset, has a significant decrease in performance on BUSI dataset.

Table 3. Performance comparison of different object detection networks on BUSI.

	Model	AP	AP50	AP75	AR10	FPS
Anchor-based networks	Faster R-CNN-res50	0.584	0.88	0.703	0.725	13.62
	RetinaNet-res50	**0.849**	0.962	0.927	**0.885**	17.42
	SSD-res34	0.813	0.939	0.924	0.845	31.4
	SSD-vgg16	0.791	0.965	0.947	0.845	29.52
	YOLOF-res50	0.823	0.962	0.877	0.856	23.3
	YOLOV3-darknet53	0.769	**0.979**	0.919	0.81	19.55
	YOLOV3-res34	0.791	0.966	**0.95**	0.825	25.52
Anchor-free networks	CornerNet-res50	0.535	0.823	0.694	0.702	11.42
	FCOS-res50	0.841	0.928	0.888	0.881	18.4
	TTFNet	0.394	0.658	0.485	0.526	**38.11**
	YOLOX-m	0.578	0.853	0.695	0.714	26.14

Note: Bold font indicates the best performance results.

Figure 5 presents the performance results and AP-FPS plot of each network on the BUSI dataset. Overall, YOLOV3 and SSD still demonstrate the most ovweall performance, similar to the results with BUS+BUSI mixed-dataset. Although RetinaNet and FCOS show impressive performance in AP, their lower FPS affects their overall performance.



(a) (b)

Figure 5. Networks performance results and AP-FPS plot on BUSI dataset. (**a**) is the performance results of different networks, (**b**) is the FPS and AP scatter plot of the networks.

3.5. Results of Generalization Performance

In medical image analysis, the generalization ability of a model is particularly important, as it directly relates to the model's practicality and reliability. A model with good generalization ability can adapt to a diverse range of cases, reducing the risk of misdiagnosis and missed diagnosis, thereby enhancing the accuracy and reliability of diagnoses. It ensures that the model accurately identifies and classifies data that differ in lesion shape, size or appearance from the training data. Strong generalization also means that the model can adapt to images from different devices and protocols, enhancing its application value in real clinical environments. Therefore, in this study, we compare the generalization performance of the nine networks. We train the models using BUSI dataset and validate them on BUS dataset, with validation results shown in Table 4.

From Table 4, we observe that FCOS achieves excellent performance in generalization, achieving the best results in AP, AP50, and AR, and the second-best in AP50, demonstrating its strong generalization ability. RetinaNet, which performs well on the BUSI dataset, also achieves good results, with the second-best AP. As shown in Figure 6b, SSD approaches the top right corner, indicating excellent overall performance, achieving a balance between speed and accuracy.

Table 4. Performance comparison of different object detection networks training on BUSI and testing on BUS.

	Model	AP	AP50	AP75	AR10	FPS
Anchor-based networks	Faster R-CNN-res50	0.603	0.925	0.676	0.625	7.14
	RetinaNet-res50	0.835	**1**	0.911	0.849	10.45
	SSD-res34	0.83	0.96	0.889	0.863	14.72
	SSD-vgg16	0.771	**1**	0.94	0.817	13.63
	YOLOF-res50	0.804	0.943	0.804	0.844	11.79
	YOLOV3-darknet53	0.762	**1**	**0.952**	0.787	10.45
	YOLOV3-res34	0.773	0.995	0.924	0.8	12.82
Anchor-free networks	CornerNet-res50	0.514	0.883	0.574	0.587	6.35
	FCOS-res50	**0.871**	**1**	0.946	**0.894**	11.15
	TTFNet	0.349	0.615	0.418	0.427	**20.24**
	YOLOX-m	0.573	0.888	0.611	0.619	12.25

Note: Bold font indicates the best performance results.

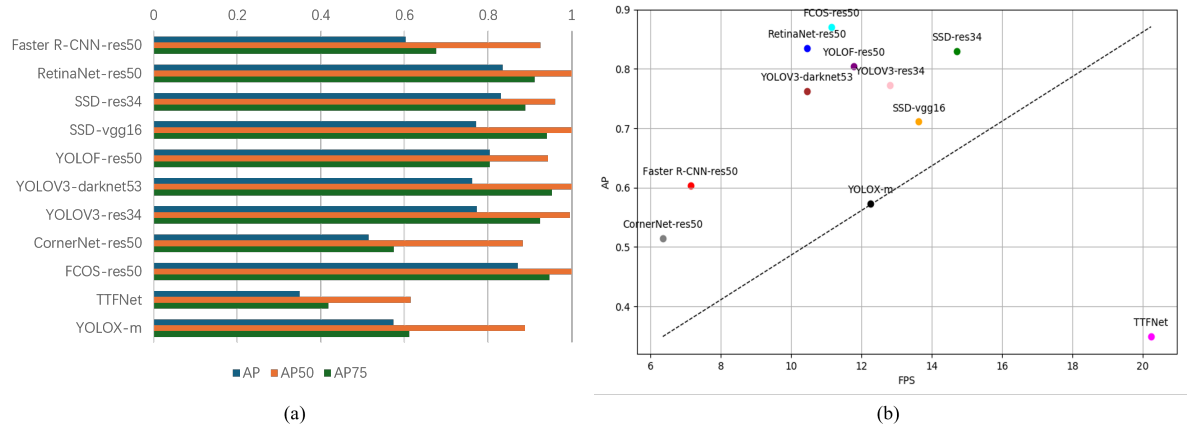


Figure 6. Networks performance results and AP-FPS plot of generalization experiments. (**a**) is the performance results of different networks, (**b**) is the FPS and AP scatter plot of the networks.

4. Conclusions

This study comprehensively compares the performance of nine object detection networks in breast lesion detection, encompassing four types: two-stage, one-stage, anchor-based, and anchor-free. This range covers all current types of object detection networks, ensuring a comprehensive and representative evaluation. We validate model performance on two datasets and compare their generalization abilities. The results demonstrate the strengths and limitations of different types of networks in breast lesion detection tasks. In terms of performance on a single dataset, anchor-based networks generally outperform anchor-free networks. Notably, the SSD model, while maintaining a high AP, also exhibits rapid detection speed, proving its practicality and effectiveness in breast cancer detection. This also indicates that anchor-based methods have strong detection capabilities for common lesion types in breast ultrasound images. The superior performance of anchor-based networks can be attributed to their predefined anchor boxes, which provide better assistance in detecting objects of varying sizes and aspect ratios. These anchor boxes serve as priors, helping the network focus on regions of interest, thereby enabling more accurate localization and classification of lesions.

However, in the comparison of generalization performance, the anchor-free network FCOS shows superior performance. This finding highlights the advantage of anchor-free networks in handling lesions with varying shapes and sizes. Since the FCOS network does not rely on preset anchors, it can adapt more flexibly to targets of different sizes, thereby performing better on new or unknown datasets. This is particularly important for breast cancer detection, as lesion shapes and sizes can vary among patients.

Early detection of breast cancer is crucial for improving patient survival rates, and developing accurate and rapid breast cancer auxiliary diagnostic systems is essential. In summary, this research provides valuable insights for the early detection and diagnosis of breast cancer, offering important guidance for the development of efficient and accurate breast cancer auxiliary diagnostic systems in the future.

Author Contributions

Y.W.: methodology, software and writing; Q.Z.: data preprocess; B.Z.: investigation; D.T. and R.Z.: data post-process; W.Z.: writing—reviewing and editing. All authors have read and agreed to the published version of the manuscript.

Funding

This research received no external funding.

Institutional Review Board Statement:

Not applicable.

Informed Consent Statement

Not applicable.

Data Availability Statement

Research Data Policies at https://doi.org/10.1109/JBHI.2017.2731873 and https://doi.org/10.1016/j.dib.2019.104863.

Conflicts of Interest

The authors declare no conflict of interest.

References

1. Bray, F.; Ferlay, J.; Soerjomataram, I.; Siegel, R.L.; Torre, L.A.; Jemal, A. Global cancer statistics 2018: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries. *CA: A Cancer J. Clin.* **2018**, *68*, 394–424.
2. Sun, Y.S.; Zhao, Z.; Yang, Z.N.; Xu, F.; Lu, H.J.; Zhu, Z.Y.; Shi, W.; Jiang, J.; Yao, P.P.; Zhu, H.P. Risk factors and preventions of breast cancer. *Int. J. Biol. Sci.* **2017**, *13*, 1387.
3. Elmore, J.G.; Armstrong, K.; Lehman, C.D.; Fletcher, S.W. Screening for breast cancer. *JAMA* **2005**, *293*, 1245–1256.
4. Geisel, J.; Raghu, M.; Hooley, R. *The Role of Ultrasound in Breast Cancer Screening: The Case for and against Ultrasound*; Seminars in Ultrasound, CT and MRI. Elsevier: Amsterdam, The Netherlands, 2018; Volume 39, pp. 25–34.
5. Qian, X.; Pei, J.; Zheng, H.; Xie, X.; Yan, L.; Zhang, H.; Han, C.; Gao, X.; Zhang, H.; Zheng, W.; et al. Prospective assessment of breast cancer risk from multimodal multiview ultrasound images via clinically applicable deep learning. *Nat. Biomed. Eng.* **2021**, *5*, 522–532.
6. Zhu, Z.; Wang, S.H.; Zhang, Y.D. A Survey of Convolutional Neural Network in Breast Cancer. *Comput. Model. Eng. Sci.* **2023**, *136*, 2127–2172.
7. Drukker, K.; Giger, M.L.; Horsch, K.; Kupinski, M.A.; Vyborny, C.J.; Mendelson, E.B. Computerized lesion detection on breast ultrasound. *Med. Phys.* **2002**, *29*, 1438–1446.
8. Liu, B.; Cheng, H.; Huang, J.; Tian, J.; Liu, J.; Tang, X. Automated segmentation of ultrasonic breast lesions using statistical texture classification and active contour based on probability distance. *Ultrasound Med. Biol.* **2009**, *35*, 1309–1324.
9. Shan, J.; Cheng, H.; Wang, Y. Completely automated segmentation approach for breast ultrasound images using multiple-domain features. *Ultrasound Med. Biol.* **2012**, *38*, 262–275.
10. Yap, M.H.; Goyal, M.; Osman, F.; Martí, R.; Denton, E.; Juette, A.; Zwiggelaar, R. Breast ultrasound region of interest detection and lesion localisation. *Artif. Intell. Med.* **2020**, *107*, 101880.
11. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *39*, 1137–1149.
12. Wang, Y.; Yao, Y. Breast lesion detection using an anchor-free network from ultrasound images with segmentation-based enhancement. *Sci. Rep.* **2022**, *12*, 14720.
13. Tian, Z.; Shen, C.; Chen, H.; He, T. Fcos: Fully convolutional one-stage object detection. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019; pp. 9627–9636.
14. Cao, Z.; Duan, L.; Yang, G.; Yue, T.; Chen, Q. An experimental study on breast lesion detection and classification from ultrasound images using deep learning architectures. *BMC Med. Imaging* **2019**, *19*, 1–9.
15. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. Ssd: Single shot multibox detector. In Proceedings of the Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, 11–14 October 2016; Part I 14, pp. 21–37.
16. Mo, W.; Zhu, Y.; Wang, C. A method for localization and classification of breast ultrasound tumors. In Proceedings of the Advances in Swarm Intelligence: 11th International Conference, ICSI 2020, Belgrade, Serbia, 14–20 July 2020; pp. 564–574.
17. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
18. Yu, X.; Zhu, Z.; Alon, Y.; Guttery, D.S.; Zhang, Y. GFNet: A Deep Learning Framework for Breast Mass Detection. *Electronics* **2023**, *12*, 1583.

19. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal loss for dense object detection. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2980–2988.
20. Chen, Q.; Wang, Y.; Yang, T.; Zhang, X.; Cheng, J.; Sun, J. You only look one-level feature. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 13039–13048.
21. Law, H.; Deng, J. Cornernet: Detecting objects as paired keypoints. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 734–750.
22. Liu, Z.; Zheng, T.; Xu, G.; Yang, Z.; Liu, H.; Cai, D. Training-time-friendly network for real-time object detection. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, New York, USA, 7–12 February 2020; Volume 34, pp. 11685–11692.
23. Ge, Z.; Liu, S.; Wang, F.; Li, Z.; Sun, J. Yolox: Exceeding yolo series in 2021. *arXiv* **2021**, arXiv:2107.08430.
24. Yap, M.H.; Pons, G.; Marti, J.; Ganau, S.; Sentis, M.; Zwiggelaar, R.; Davison, A.K.; Marti, R. Automated breast ultrasound lesions detection using convolutional neural networks. *IEEE J. Biomed. Health Inform.* **2017**, *22*, 1218–1226.
25. Al-Dhabyani, W.; Gomaa, M.; Khaled, H.; Fahmy, A. Dataset of breast ultrasound images. *Data Brief* **2020**, *28*, 104863.
26. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 580–587.
27. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
28. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
29. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
30. PaddlePaddle. *PaddleDetection, Object Detection and Instance Segmentation Toolkit Based on PaddlePaddle*; PaddlePaddle: Haidian, Beijing, 2019.
31. Loshchilov, I.; Hutter, F. Sgdr: Stochastic gradient descent with warm restarts. *arXiv* **2016**, arXiv:1608.03983.
32. Bochkovskiy, A.; Wang, C.Y.; Liao, H.Y.M. Yolov4: Optimal speed and accuracy of object detection. *arXiv* **2020**, arXiv:2004.10934.

AI Medicine
https://www.sciltp.com/journals/aim

Article

ML-Based RNA Secondary Structure Prediction Methods: A Survey

Qi Zhao [1], Jingjing Chen [1], Zheng Zhao [2], Qian Mao [3], Haoxuan Shi [1] and Xiaoya Fan [4,*]

[1] School of Medicine and Biological Information Engineering, Northeastern University, Shenyang 110000, China
[2] School of Artificial Intelligence, Dalian Maritime University, Dalian 116000, China
[3] Department of Food Science and Engineering, College of Light Industry, Liaoning University, Shenyang 110000, China
[4] School of Software, Dalian University of Technology, Key Laboratory for Ubiquitous Network and Service Software, Dalian 116000, China
* Correspondence: xiaoyafan@dlut.edu.cn

How To Cite: Zhao, Q.; Chen, J.; Zhao, Z.; Mao, Q.; Shi, H.; Fan, X. ML-Based RNA Secondary Structure Prediction Methods: A Survey. *AI Medicine* **2024**, *1*(1), 6. https://doi.org/10.53941/aim.2024.100006.

Received: 6 May 2024
Revised: 17 October 2024
Accepted: 22 October 2024
Published: 29 October 2024

Abstract: The secondary structure of noncoding RNAs (ncRNA) is significantly related to their functions, emphasizing the importance and value of identifying ncRNA secondary structure. Computational prediction methods have been widely used in this field. However, the performance of existing computational methods has plateaued in recent years despite various advancements. Fortunately, the emergence of machine learning, particularly deep learning, has brought new hope to this field. In this review, we present a comprehensive overview of machine learning-based methods for predicting RNA secondary structures, with a particular emphasis on deep learning approaches. Additionally, we discuss the current challenges and prospects in RNA secondary structure prediction..

Keywords: RNA secondary structure prediction; machine learning; deep learning

1. Introduction

Ribonucleic acid (RNA) is an essential substance in most living organisms and plays a nonnegligible role in regulating proteins and biological processes [1]. The RNA molecule comprises a specific sequence of nucleotides arranged in a 5' to 3' direction. Nucleotides include four kinds of bases which are adenine (A), cytosine (C), guanine (G), or uracil (U), and pair up through hydrogen bonds to form the secondary structure [2]. Typically, each base pairs up with only one other base, the most common examples are, the Watson-Crick base pairs (A-U and G-C) and the wobble base pair (G-U). These base pairings often result in a nested structure, where multiple stacked base pairs form a helix, and unpaired base pairs form loops (Figure 1a). It's worth noting that there are three types of special base pairings [2] commonly found in native RNA secondary structures: noncanonical base pairs (Figure 1b), base triples (Figure 1c), and pseudoknots (Figure 1d). Noncanonical base pairs refer to base pairing other than the Watson-Crick and the wobble base pairs, constituting around 40% of all base pairs in structured RNAs [3]. Base triples also widely exist in RNA structures, which involve three bases interacting jointly to stabilize various RNA tertiary interactions [4, 5]. Pseudoknots [6] occur when bases from different loops pair up and then create a non-nested structure between two separated bases. Though pseudoknots only represent a few base pairs in known secondary structures, they play an important role in RNA function [7].

RNA has long been believed to serve only as a messenger between DNA and proteins until the discovery of non-coding RNAs (ncRNAs). It is found that less than 2% of the human genome belongs to protein-coding regions and the rest is transcribed into ncRNAs [8]. NcRNAs are RNAs that do not encode proteins and play functions depending on their structures [9, 10]. Their function includes catalysis, translation, RNA modification, RNA stability, protein synthesis, expression regulation, and protein degradation [11–16]. Moreover, they are important in various human diseases such as cancer, diabetes, and atherosclerosis [13, 17]. Therefore, recognizing ncRNA structure and its involvement in both normal biological processes and pathological conditions has opened new avenues

for researches and potential therapeutic interventions.

Generally, ncRNA molecules form higher-order structures. Unlike proteins, which fold globally driven by hydrophobic forces, RNA folding follows a hierarchical process [18]. RNA in the linear primary structure is folded to form a secondary structure rapidly, resulting in a significant energy loss [19, 20]. Then the secondary structure further forms the tertiary structure at a much slower speed. Though increasing amounts of abundant ncRNA sequences are public [21], most ncRNA structures remain unknown. This structural information deficiency makes it challenging to infer their functions, thus, gaining valid information on ncRNA structures holds great research value. High stability and variety of secondary structures within cells contribute to the crucial role it plays in ncRNA function [22, 23]. Therefore, even without knowing the higher-order structures, the secondary structure alone is often sufficient for inferring function and practical applications [23].

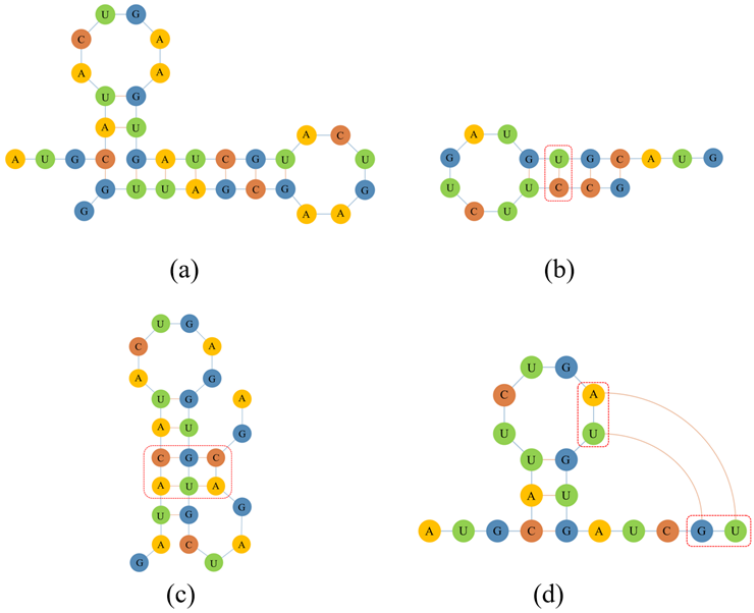


Figure 1. The yellow, green, blue, and orange circles represent A, U, G, and C, respectively. Yellow lines represent hydrogen bonds, and blue lines represent covalent bonds. (**a**) Common base pair. (**b**) Noncanonical base pair. (**c**) Base triple. (**d**) Typical pseudoknot.

Since the 1970s, a spring of prediction methods has been developed, and computational methods have become the dominant approach for RNA secondary structure prediction. However, the development of both the accuracy and processing speed have stagnated in recent years. Machine learning (ML)-based approaches emerge to address these limitations. Initially, ML-based prediction methods were overlooked due to their simple models and limited accuracy resulting from data scarcity. However, with the rise of RNA datasets and advancements in deep learning (DL), ML methods now surpass traditional approaches in accuracy and applicability, which paves the way for the development of next-generation RNA structure prediction tools [24].

This paper focuses on reviewing methods for predicting RNA secondary structures based on ML and offers a detailed discussion of their pros and cons. Though other previous reviews have covered RNA secondary structure prediction [24–27], there is a lack of reviews focusing especially on DL techniques. This review aims to assist researchers in understanding the current status of RNA secondary structure prediction based on ML and DL while gaining insight into the existing challenges and prospects in this area.

2. Traditional Prediction Methods

RNA secondary structure prediction has been advancing over the past two decades with a variety of methods. Traditional prediction methods include wet-lab experimental experiments and computational predicting methods by algorithm (Figure 2). For wet-lab experiments, Nuclear Magnetic Resonance (NMR) [28] and X-ray crystallography [29] are the two most accurate methods, but they are time-consuming, costly, and limited in applicability. Chemical probing [30, 31] or enzymatic [32, 33] with next-generation sequencing [34, 35] are mainly suitable for in vitro studies but not for in vivo conformation. Only a small fraction of known ncRNAs have been experimentally determined up to now [36], thus, computational methods are accessible alternatives for predicting RNA secondary structures. Comparative sequence analysis [37, 38] is considered the most accurate computational method, which relies on the conservation of RNA secondary structures across evolution compared to primary sequences and

identifies base pairs that covary to maintain Watson-Crick and wobble base pairs [39]. Several algorithms have been designed to improve the performance [40–45] and process pseudoknots [46–48]. However, comparative sequence analysis requires a large set of homologous sequences as a basis [49]. When homologous sequences are lacking, another method, namely the score-based method, has been widely used in this field. These methods assume that the native RNA structure has a minimum or maximum total score, transforming the prediction problem into an optimization problem. The scoring schemes for these methods [50–53] typically involve multiple parameters and are based on free energy calculations using the nearest neighbor model. Several methods have been developed to predict structures with pseudoknots [54–59]. However, accurately predicting special base pairs in RNA structures remains a challenging task. In addition, the folding mechanism hypotheses they rely on may not always hold, and the computational cost is extremely high for longer RNA sequences.

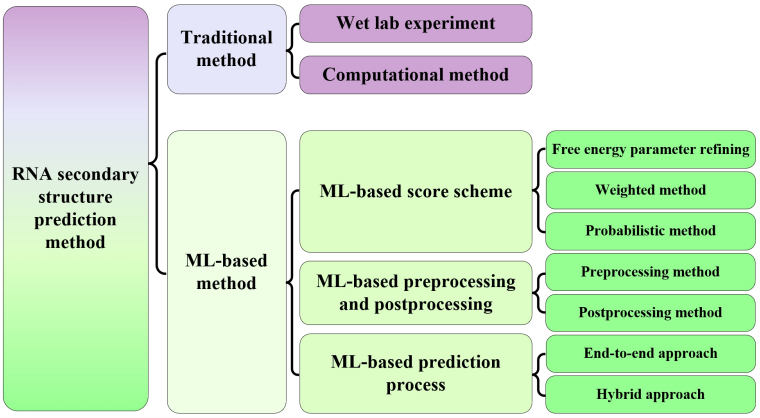


Figure 2. Categories of RNA secondary prediction methods.

3. ML-Based Methods

We classify ML-based RNA secondary structure prediction methods into three categories (Figure 2), namely ML-based score scheme, ML-based preprocessing and postprocessing, and ML-based prediction process, partly referencing the method from our previous paper [24]. These categories are defined based on the subprocess ML applied, and their advantages and limitations are summarized in Table 1. All these methods are trained using supervised learning [60]. These models are trained to learn functions that map input features (such as free energy parameters, encoded RNA sequences, sequence patterns, and evolutionary information) to outputs (including continuous values such as free energy or classification labels such as paired or unpaired bases) by adjusting their parameters using known data. The supervised training process enables ML-based methods to learn patterns from the available data, empowering them to make predictions on unseen data. When a new input is provided, the model can assign a corresponding label or predict a corresponding value based on the learned mapping function [60].

Table 1. Advantages and Limitations of Three Categories of ML-based RNA Secondary Structure Prediction Methods.

Categories	Advantages	Limitations
ML-based score scheme	Provide available parameters for the traditional prediction algorithms to improve the prediction accuracy.	Limited prediction accuracy, particularly for noncanonical base pairs, base triples, and pseudoknots.
ML-based preprocessing and postprocessing	Simplify the prediction process and compatible with traditional prediction algorithms.	The prediction accuracy relies heavily on the intermediate RNA secondary structure prediction model.
ML-based prediction process	Greatly improve the speed and accuracy of prediction, and can predict noncanonical base pairs, base triples, and pseudoknots.	Poor interpretation, high computing costs in model training, and not guaranteed generalization ability on new types of RNA.

3.1. ML-Based Score Scheme

ML-based score scheme aims to train models capable of generating new score schemes, replacing the traditional score scheme (Figure 3). While ML-based methods enhance accuracy through parameter estimation in score schemes, structural prediction remains an optimization problem, where the estimated scores scheme is used for evaluating the potential conformations. ML-based score schemes can be categorized into three types (Figure 2) based on the meaning of scores: the free energy parameter-refining approach, the weighted approach, and the probabilistic approach. The models based on these approaches are detailed in the Supplementary Table S1.

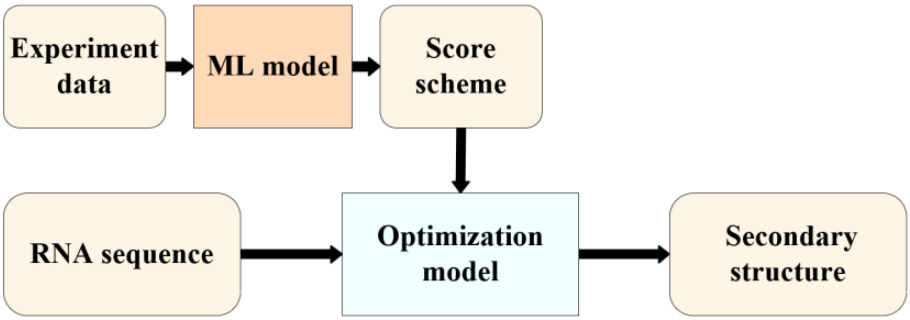


Figure 3. Framework of ML-based score scheme methods. RNA sequences are input into the ML model, and the scoring scheme evaluates the scores of potential conformations and picks out the optical results to output the RNA secondary structures.

3.1.1. ML-Based Free Energy Parameter Refining

Since the publication of the free energy theory, the free energy-focused approach has been widely adopted in score schemes, particularly in assigning free energy values to elements of RNA structures. Among these, Turner's NN model [53] is widely accepted due to its accuracy in approximating free energy. However, determining the multiple thermodynamic parameters in the NN model requires many optimal melting experiments, which are labor-intensive and time-consuming [61, 62]. Then ML techniques have been employed to refine parameters in the energy model, which utilize models to score and provide more abundant features based on known RNA secondary structure data or thermodynamic data. Xia et al. [50] used known thermodynamic data to train a linear regression model for inferring thermodynamic parameters. However, certain structural element parameters are predetermined prior to the computation of other parameters, thereby constraining the potential options for the entire parameter set. To address this limitation, Andronescu et al. [63] put forward a constraint generation approach that employs various constraints to ensure that the energy of reference structures is the lowest among other alternative structures. The team trained this model on a mass of thermodynamic and structural data to infer free energy parameters, achieving a 7% higher F-measure than standard Turner parameters. Further, the research team proposed a Boltzmann-likelihood model and loss-augmented max-margin constraint generation model using a larger set of data to impose constraints on parameters [64]. In addition, it is worth noting that the parameters derived from the above approaches are thermodynamic, so they can be directly applied in algorithms embedded within the same energy model, examples are, RNA folding kinetics simulation [65] and miRNA target prediction [66].

3.1.2. ML-Based Weighted Methods

ML-based free energy parameter refining approaches successfully improve the accuracy of prediction, however, those methods can only be alternatives for wet lab experiments aimed at obtaining energy parameters. Thus, weighting methods were proposed, of which the scoring scheme is independent of the free energy assumption, treating the parameters of the RNA structure as weights rather than free energy changes. Zakov et al. [67] utilized a discriminative structured-prediction learning framework along with an online learning algorithm and significantly expanded the number of weights to around 70,000. The authors achieved this by investigating a wider range of structural elements with more extensive sequential contexts and employing thousands of training datasets. Then ContextFold as a substantial accuracy enhancement model is introduced based on these resulting weights [67]. Akiyama et al. [68] improved a structured support vector machine (SSVM) by the thermodynamic approach to obtain a large set of weights for detailed structural elements. To mitigate overfitting, they applied L1 regularization. Subsequently, they developed MXfold by merging the ML-based weights with experimentally determined thermodynamic parameters, yielding better performance than models solely based on thermodynamic parameters or ML-based weights. Sato et al. improved the model as MXfold2 [69], which uses CNN to calculate the

folding scores of RNA sequences, and applies dynamic programming (DP) and the max-margin framework to predict the structure. The max-margin framework includes structural hinge loss function, thermodynamic regularization, and L1 regularization, ensuring that the folding scores are closely aligned with the free energy calculated using the thermodynamic parameters. MXfold2 displayed robust predictions in both sequence-wise and family-wise cross-validation. These studies indicate that weighted approaches based on ML can break through the limitations of the thermodynamic parameter approach. They separate structure prediction from energy estimation, making it advantageous for both tasks. In this case, weighted approaches can achieve more satisfying results. However, a drawback is that the learned weights lack explainability due to the inherent black-box nature of ML algorithms. Therefore, the obtained scores cannot be directly used for computations such as the partition function, centroid structures, or base pair binding probabilities, among others.

3.1.3. ML-Based Probabilistic Methods

As ML technologies improve, stochastic context-free grammars (SCFGs) appear as a significant method for probabilistic approaches for RNA structure prediction [70–74]. SCFGs extend traditional context-free grammars (CFGs) by assigning probabilities to production rules, allowing them to generate structures with different probabilities. It provides a framework for generating diverse possible structures and estimating their probabilities. In an SCFG model, each production rule in the grammar is associated with a probability parameter that assigns a probability to each derived sequence. It typically estimates probability parameters by learning RNA sequence datasets with known secondary structures, eliminating the requirement for external laboratory experiments [73]. The application of SCFGs for tRNA secondary structure prediction was first introduced by Sakakibara et al. [70]. They used an expectation-maximization (EM) method to learn the probability parameters. Knudsen and Hein [72] further enhanced the SCFG model by incorporating evolutionary information, leading to the development of the robust and practical tool Pfold [72].

Sato et al. [75] proposed a nonparametric Bayesian extension of SCFGs using the hierarchical Dirichlet process (HDP). Traditional SCFGs are required to define a fixed number of generation rules and parameters in advance, whereas non-parametric Bayesian extension allows SCFGs to adaptively learn the complexity and structure of the model. Thus, it is flexible to different data and identifies an optimal RNA grammar from the training dataset expressively and adaptably. To leverage the abundance of RNA sequences with unknown structures, Yonemoto et al. [76] proposed a semi-supervised learning algorithm to improve prediction accuracy. This algorithm determines probability parameters in a probabilistic model that combines SCFG and a conditional random field, enabling the incorporation of both labeled and unlabeled data. Even though, the probabilistic approach, such as SCFG, cannot fully take the place of Minimum Free Energy (MFE) methods, since the accuracy of the best SCFG models still falls short of the top-performing free energy-based models. Additionally, SCFGs have limitations in describing certain RNA structures, such as those containing special base pairs that deviate from the conventional Watson-Crick base pairing. These constraints highlight the importance of considering both probabilistic and free energy-based approaches to achieve more accurate and comprehensive RNA structure predictions.

Do et al. [77] proposed a new method CONTRAfold without physics-based models. Novelly, CONTRAfold applies the conditional log-linear model (CLLM) to determine probability parameters that effectively differentiate correct RNA structures from incorrect ones. CLLM is a flexible probabilistic ML model that allows easy parameter estimation and incorporation of any chosen features into the model, providing a framework for capturing complex relationships between input features and target variables. Compared to previously available probabilistic models, CONTRAfold reaches the highest accuracy in single-sequence RNA structure prediction. However, CLLM is computationally slower than SCFGs, limiting its application to large-scale training datasets. Since CLLM imposes fewer structural constraints on the output sequence, when encountering sequences with specific base pairs, it potentially leads to the possibility of generating false RNA structures. In addition, the estimated parameters lack explicit biological interpretation due to its black-box feature.

3.2. ML-Based Preprocessing and Postprocessing

ML can be applied in preprocessing to simplify the prediction process (Figure 4). Hor et al. [78] introduced a tool based on support vector machine (SVM) that aims to choose the most effective prediction method. They believe that different RNA sequences possess distinct features so that specific prediction methods perform better for one RNA sequence. By utilizing SVM, the tool can identify the prediction method that is likely to yield the best results for a given RNA sequence. Similarly, based on the assumption that folding rules differ from RNA sequences, Zhu et al. [79] put forth an SCFG model to identify the most probable folding rules for an RNA sequence ahead of the prediction process. By doing so, the accuracy of the prediction can be improved.

Additionally, processing long RNA sequences can be costly, time-consuming, and complicated. To solve this problem, Zhao and colleagues [80] designed a DL-based model RNA-par using transfer learning. RNA-par splits RNA sequence into several independent fragments (i-fragments) to improve prediction performance. RNA-par consists of a 4-layer 1D-CNN block for extracting sequence features, a Bi-LSTM block capturing information from both sequences, and a 2-layer ResNet block acting as prediction head to generate the outputs i-fragments. Since i-fragments are shorter sequences, RNA-par makes it convenient for the following prediction process.

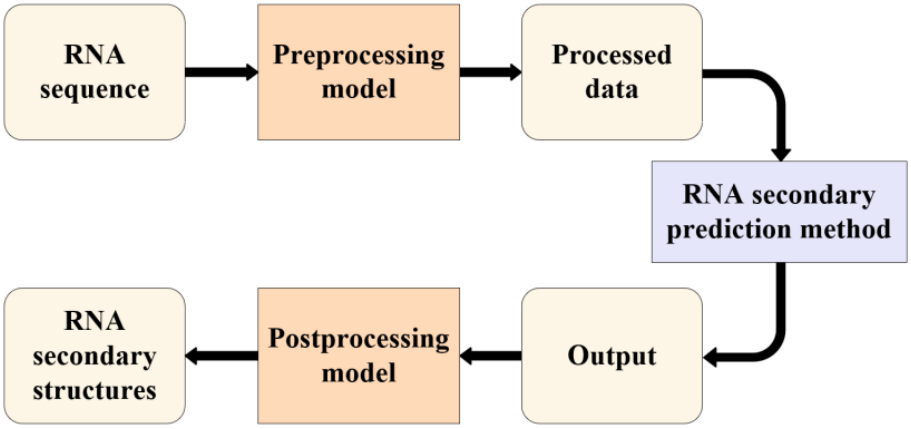


Figure 4. Framework of preprocessing and postprocessing. To simplify the prediction process in the ML-based prediction model, the preprocessing model processes RNA sequences into other forms of data, and the postprocessing model transcript outputs into RNA secondary structures.

ML is also used in postprocessing to reach a better result. Since various methods yield multiple structures for an RNA sequence, postprocessing models can be utilized to determine the most probable structures among the outcomes (Figure 4). Haynes et al. [81] combined ML with graph theory to represent RNA graphical structures using trees, where edges represent helices and vertices represent bulges or loops. Using graphical invariants as input features, a multilayer perceptron (MLP) model is trained to identify whether the result is an RNA structure. This approach enables the ML model to distinguish between structures that are likely to represent RNA structures and those that are not. Additionally, an assumption from Koessler et al. [82] indicates that a larger one is formed when two smaller RNA secondary structures bond together. They extract a feature vector from the merged trees and apply it to an MLP model to predict the probability of an RNA-like structure. By leveraging this MLP model, they were able to estimate the likelihood of a given structure resembling an RNA-like structure. Details of above models are summarized in Supplementary Table S2.

3.3. ML-Based Predicting Process

ML techniques can be utilized as end-to-end prediction approaches or integrated with other algorithms as filters or optimizers. The models based on both approaches are detailed in the Supplementary Table S3.

3.3.1. End-To-End Approaches

End-to-end approaches usually directly predict the secondary structure from the RNA sequence without intermediate steps or external information (Figure 5). They aim to capture the inherent structure-sequence relationship in training sample, and learn the mapping between the sequence and its secondary structure in a single integrated model.

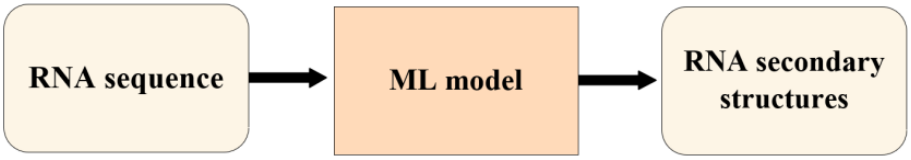


Figure 5. Framework of end-to-end approaches. End-to-end approaches directly predict secondary structures from RNA sequences without intermediate steps or external information.

Built upon Nussinov and Jacobson's hypothesis [46], ML techniques were first introduced to RNA secondary structure prediction by Takefuji et al. [83]. They used a system of interactional neurons to obtain a near-maximum

independent set (MIS) from an adjacent graph representing base pairs. To improve this work, Qasim et al. [84] built a new MLP model with h neurons in the hidden layer to obtain MIS, and its activation function is based on Kolgomorov's theorem (h representing the number of possible base pairs in an RNA sequence). In other aspects, Liu et al. [85] considered the energy contribution of base pairs and employed a Hopfield neural network (HNN) to get the MIS. Apolloni et al. [86] enhanced computational speed and applied mean-field approximation in both the instant resolution and learning phases, slightly enlarging the input RNA length for this approach. Unfortunately, HNN was limited by its susceptibility to local minima, so Steeg and Evan [87] utilized mean field theory (MFT) networks coupled with an objective function and biological constraints to identify the optimal structure. In this method, MFT receives four types of RNA bases that are encoded in a one-hot fashion and outputs a CT-like table.

However, since ML-based models are limited to processing tRNAs only due to the lack of data, DL techniques are thriving to break through the challenges. Singh et al. [80] proposed SPOT-RNA, the first end-to-end DL model for RNA secondary structure prediction. SPOT-RNA turns sequences into CT tables and employs ultra-deep hybrid networks consisting of ResNets and Bi-BLSTMs. ResNets obtain the contextual information while Bi-BLSTMs capture dependencies between distant nucleotides in the RNA sequence. SPOT-RNA has an outstanding performance on benchmark datasets compared to score-based methods and SCFG-based methods. The same team later introduced the SPOT-RNA2 model [88], which incorporated evolution-derived sequence profiles and mutational coupling, outperforming the model SPOT-RNA. Furtherly, Fu and colleagues [89] introduced a special model UFold which converts the sequence into an image of all possible base pairs, and processes through U-net and a 1D-convolution to generate contact scores between bases. Unlike other models, it innovatively abandons raw sequences but adopts a 3D vector analogous to an image as input, making the model fully convolutional to achieve higher efficiency and ability to process pseudoknots. Another DL model, E2Efold, put forward by Chen et al. [90], consists of a transformer-based deep model and a multilayer network based on an unrolled algorithm. E2Efold takes the RNA sequence as input, employs the deep model to encode the sequence information, and then the multilayer network to filter the output. One of the advantages of E2Efold is its ability to process longer RNA sequences, including those large molecules with complex structures. It is also able to capture non-local interactions in the sequence and take these relationships into account when generating secondary structures. However, E2Efold suffers from a severe overfitting, and generalized on unseen RNAs.

Besides primary sequences, DL models can be combined with other information. Calonaci et al. [91] trained an ensemble model that combines co-evolutionary data (DCA), SHAPE data, and RNA sequence data. It has an MLP subnetwork based on DCA data and a CNN subnetwork based on SHAPE data for predicting penalties, then its folding module generates structures using penalties and RNA sequences.

3.3.2. Hybrid Approaches

Hybrid approaches that combine ML models with other methods have been explored for predicting RNA secondary structure prediction. One of these approaches combined ML models with filters to predict a possible structure and another is to hybrid ML models with optimization methods. The framework is shown in Figure 6.

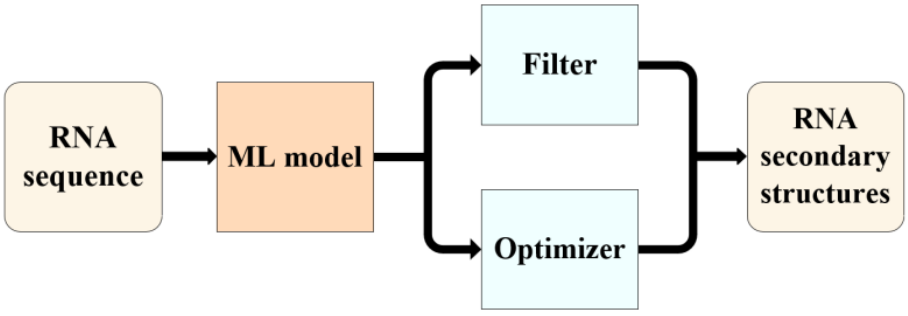


Figure 6. Framework of hybrid approaches. One of the hybrid approaches combines ML models with filters to predict a possible structure, the other combines ML models with optimization methods.

ML Filter Combined Methods

For these methods, they typically include an ML model and a filter in the process to achieve the output. Bindewald and Shapiro [42] integrated the ML model with a filter to reach the consensus structure of a set of aligned RNAs. The model gets the possibility score for each pair of alignment columns by employing a hierarchical network of k-nearest neighbor models. Filters with rules of native RNA structures constrain the result of the ML model to get outputs. Wu et al. [92] and Lu et al. [93] regarded predicting structure as a sequence-labeling problem, and

they predicted the state of bases by Bi-LSTM and applying a rule-based filter to cope with controversial pairings. Another innovative model DpacoRNA [94] used Bi-LSTM as a structured filter and employed a parallel ant colony optimization method to hunt for the maximum probable structures. A recent study [95] constructed an composite network that integrates Bi-LSTM [96], Transformer [97], and U-Net [98] to calculate pairwise scores between bases. Utilizing four established rules, the network constructs a filter to discern potential RNA structures.

ML Optimization Combined Methods

In these approaches, the ML model finds the relationship between each base or each pair of bases, and the optimization method picks out the optimal structure. CNNFold model, published by Booy et.al [99], consists of multiple residual blocks and a readout layer post-processing to predict a score matrix for all possible pairings. They also developed an algorithm called Argmax post-processing converting the score matrix into the best secondary structure. It is worth noting that CNNFold and its variants can predict structures with pseudoknots well. Similar to CNNFold, Liu's group [100] proposed a model combined with DL and DP, that predicts the status distribution of each base by the CNN model and finds the most probable structure using the DP algorithm. To improve the result, they replaced the CNN with the Bi-LSTM model integrated with another optimization algorithm [101]. Willmott et al. [102] adopted the SHAPE-directed method (SDM) to predict optimal structure rather than developing a new optimization model, meanwhile, they trained a Bi-LSTM model that generates SHAPE-like data of an RNA sequence as the inputs for SDM. Recently, Chen and Chan [103] proposed a DL-based model, REDFold, which utilizes the UFold [89] architecture. Its encoder accepts a 2D contact matrix as input, while the decoder yields a score map. They employ constrained optimization instead of DP to identify the optimal structure, thereby enabling their model to predict non-nested folding patterns.

4. Databases

In ML-based RNA structure prediction research, access to comprehensive and reliable structural data is essential for model training and performance evaluation. Generally, the quantity and quality of training data directly influence the learning ability and prediction accuracy of an ML-based model. A rich set of training samples helps models capture a more comprehensive range of RNA structural features and enhances its robustness when faced with unseen data. In addition, the representativeness of the database is vital. A sample containing various types of RNA secondary structures enables the model to better understand the complexity of RNA structures.

Several databases have been developed to provide researchers with extensive resources for studying RNA sequences and structures. Among these, some databases offer a broad spectrum of RNA data (Such as Comprehensive Databases), including diverse species and structural conformations, while others focus on specific aspects or types of RNA (Such as Specialized Databases).

4.1. Comprehensive Databases

Comprehensive databases are large, general-purpose collections of RNA structures that include a wide variety of RNA species and structural conformations. These databases often contain a large number of experimentally obtained RNA structures or computational predictions, making them useful for ML-based RNA structure prediction. RNA STRAND [104] is a database that provides a diverse collection of RNA sequences, containing 4,666 RNA samples. It is designed to offer structural and sequence information for RNA research. RCSB Protein Data Bank (PDB) [105] is an authoritative database for biomolecular structures, providing 4,962 RNA structures. It primarily includes tertiary structures obtained through experimental methods, such as X-ray crystallography and nuclear magnetic resonance, which offer a solid foundation for analyzing the conformation of RNA. bpRNA-1m [106] is a large-scale database, offering 102,348 RNA structures, which are mainly constructed using a novel annotation tool called bpRNA. While the accuracy of secondary structures provided by bpRNA-1m is relatively lower, its vast data volume makes it valuable for ML-based RNA secondary structure prediction models. RNAcentral [107] is the largest RNA secondary structure database, containing secondary structures obtained by computational methods R2DT [108].

4.2. Specialized Databases

The tRNAdb 2009 database [109] is one of the earliest specialized databases, which focuses on the structures and functions of tRNA. It provides detailed tRNA sequences and their corresponding structural information. The rRNA database [110] is dedicated to structural data of ribosomal RNA (rRNA). The tmRDB database [111] focuses on post-transcriptionally modified RNA (tmRNA), which plays an important role in bacteria, participating in protein synthesis and quality control. In addition, there are also some specialized structure databases. These databases

typically focus on one specific type of RNA structure, such as loop [112], pseudoknot [113], or non-canonical base pair [114]. In addition, based on these specialized databases, benchmark datasets such as ArchiveII [115] and RNAStralign [116] have been established. They contain various types of RNA sequences with high sequence diversity, making them especially suitable for training and evaluating the performance of RNA structure prediction models.

4.3. Other Important Databases

In addition, there are other important databases such as Rfam [117] and NNDB [53]. Rfam is a widely used RNA family database, which provides classification information including consensus secondary structures and a covariance model for each RNA family. NNDB provides crucial thermodynamic parameters for modeling the stability of RNA secondary structures, especially when calculating RNA folding energies. NNDB provides foundational data for machine learning models, helping improve the accuracy of RNA secondary structure prediction.

5. Discussions

As it is known to all, the abundance of transcripts is widely recognized as a valuable indicator for identifying transcripts of interest in different conditions, while understanding RNA structure is crucial for unraveling their functional mechanisms. A highly accurate RNA structure prediction method also has implications for various downstream investigations, including but not limited to, simulations of folding dynamics [118], the detection of ncRNAs [64, 119, 120], applications in oligonucleotide [121, 122] or drug design [123–127], and assessment of hybridization stability [128]. Even more, RNA secondary structure prediction also serves as a valuable tool in studying viruses, an example is, the SARS-CoV-2 virus [129, 130].

5.1. Pros of ML-Based Methods

ML-based methods offer several advantages over comparative sequence analysis and traditional score-based methods. Firstly, rather than rely on intricate biological mechanisms, ML-based methods tend to leverage information from diverse data types, bypassing performance limitations imposed by specific mechanism hypotheses. ML-based methods are also easy to integrate with known biological mechanisms, providing a flexible framework for analysis. Having approaches to the mass of available datasets, models knowing less knowledge of biological mechanisms usually outperform those models dependent on biological mechanisms. This infers the guess that current biological mechanisms of RNA folding might be faulty. Secondly, ML-based methods, particularly end-to-end DL models, eliminate the need to consider base matching rules. In traditional score-based methods, they utilize complex algorithms to meet base matching rules, which causes high time complexity, leaving difficulties for them to improve. In contrast, end-to-end DL models [80] can be trained to predict all base pairs in RNA structures without these rules, despite whether these base pairs are associated with secondary or tertiary interactions. Thirdly, ML-based methods offer considerable flexibility compared to traditional methods. The input data for ML models are various, no matter whether they are one-dimensional or multidimensional, homogeneous or heterogeneous, features extracted from the data or encoded bases, matrixes, or diagrams. Similarly, the outputs of ML models can also vary, including CT tables, nucleotide states, labeled sequences, or free energy values. ML models can be constructed using a diverse array of techniques, ranging from simple Hopfield networks to complex ensemble DNNs. Additionally, similarly to tasks in Natural Language Processing, RNA secondary structure prediction can also benefit as a downstream task by utilizing representations obtained from pre-trained foundation models [131–133] as inputs, thereby enhancing the accuracy of the predictive model's structure predictions. Lastly, end-to-end prediction methods exhibit fast runtime once ML models are trained. Outperformed the DP algorithm, the time complexity of ML models remains independent of the input scale, providing potential capacities for processing long RNA sequences. In summary, ML-based methods offer advantages such as flexibility, independence from base matching rules, and fast runtime, making them a promising approach for RNA structure prediction.

5.2. Remaining Challenges and Prospects

Though RNA secondary structure prediction methods using ML techniques are considered state-of-the-art in terms of prediction performance across various measures, there are still some issues needed to be addressed. To begin with, there is a need to further enhance the accuracy of predictions. Surveys [69, 88] show that there is a long way to go in improving the accuracy of RNA secondary structure prediction methods. On one hand, since RNA structures unpredictably vary in different cellular environments [134], multiple structure options instead of the most possible one should be considered when analyzing input sequences to gain predictions is worthy of consideration. On the other hand, combining an ML-based method with an optimization approach shows promise in enhancing

prediction performance. ML-based methods can leverage their ability to learn complex patterns from data and make accurate predictions while optimization methods can refine and optimize the predicted structures to align with known structural constraints and principles. This combination offers a synergistic approach that combines the strengths of both paradigms, showing its potential for future development. ML-based prediction of RNA secondary structures relies on capturing the interactions between distant nucleotides, however, when dealing with long RNA sequences, getting these long-range interactions within RNA sequences can be a challenge. Meanwhile, training a large-scale inputs ML model demands impractical computational resources. To face this sequence length limitation, striking a balance between computational efficiency and capturing long-range interactions is considerable. Innovative approaches such as hierarchical modeling, integration of experimental data, and leveraging parallel and distributed computing resources are expected solutions to develop. Furthermore, numerous traditional approaches disregard special base pairs to minimize the occurrence of false positives and reduce computational complexity [57, 135]. Though certain methods can process structures with non-canonical base pairs [136] or pseudoknots [48] none of them can accurately predict both simultaneously, even ML-based methods suffer limited accuracy. Therefore, finding solutions for predicting special base pairs is an inevitable future trend. Overfitting is another critical concern for ML-based RNA secondary structure prediction models, particularly for DL models [74]. Overfitted models tend to perform well on RNAs that are structurally similar to training data but poor on structure-dissimilar ones. Instead of truly learning the folding mechanism, these models often end up memorizing the secondary structure patterns present in the training data. Although DL-based methods employ various techniques to mitigate overfitting, such as regularization [91], constraint addition [90], dataset enlargement [80], or integration of Turner's nearest neighbor free energy parameters, concerns regarding overfitting persist.

6. Conclusion

Understanding RNA structure is crucial for comprehending biological processes, and the prediction of RNA secondary structure remains a prominent topic in the fields of computation and biology. Though ML techniques have significantly enhanced the accuracy, applicability, and computational speed of the prediction process, more sophisticated ML models and DL technologies are needed to facilitate the development of a new generation of RNA secondary structure prediction tools with improved accuracy and computational efficiency.

Supplementary Materials

The following supporting information can be downloaded at: https://www.sciltp.com/journals/aim/2024/1/363/s1. Table S1: ML-based scorescheme, Table S2: ML-based preprocessing and postprocessing, Table S3: ML-based predicting process.

Author Contributions

Q.Z.: Project administration, writing–review, editing; J.C.: Writing–original draft, visualization; Z.Z.: Writing–review, editing; Q.M.: Writing–review, editing; H.S.: Writing–review, editing, visualization; X.F.: Project administration, supervision, Writing–review, editing.

Funding

This research was funded by grants from General Project of Science and Technology Foundation of Liaoning Province of China, grant number 2023-MS-091.

Data Availability Statement

Not applicable.

Conflicts of Interest The authors declare no conflict of interest.

References

1. Wang, D.; Farhana, A. Biochemistry, RNA Structure. In *StatPearls*; StatPearls Publishing: Treasure Island, FL, USA, 2024.
2. Zhao, Y.; Wang, J.; Zeng, C.; Xiao, Y. Evaluation of rna secondary structure prediction for both base-pairing and topology. *Biophys. Rep.* **2018**, *4*, 123–132.
3. Leontis, N.B.; Westhof, E. Geometric nomenclature and classification of RNA base pairs. *RNA* **2001**, *7*, 499–512.
4. Almakarem, A.S.A.; Petrov, A.I.; Stombaugh, J.; Zirbel, C.L.; Leontis, N.B. Comprehensive survey and geometric classification of base triples in RNA structures. *Nucleic Acids Res.* **2012**, *40*, 1407–1423.
5. Doherty, E.A.; Batey, R.T.; Masquida, B.; Doudna, J.A. A universal mode of helix packing in RNA. *Nat.*

Struct. Biol. **2001**, *8*, 339–343.
6. Van Batenburg, F.H.D.; Gultyaev, A.P.; Pleij, C.W.A. PseudoBase: structural information on RNA pseudoknots. *Nucleic Acids Res.* **2001**, *29*, 194–195.
7. Staple, D.W.; Butcher, S.E. Pseudoknots: RNA Structures with Diverse Functions. *PLoS Biol.* **2005**, *3*, e213.
8. ENCODE Project Consortium. An Integrated Encyclopedia of DNA Elements in the Human Genome. *Nature* **2012**, *489*, 57–74.
9. Kovalchuk, I. Chapter 24-Non-coding RNAs in genome integrity. In *Genome Stability*, 2nd ed.; Kovalchuk, I., Kovalchuk, O., Eds.; Volume 26 of Translational Epigenetics; Academic Press: Boston, MA, USA 2021; pp. 453–475.
10. Kasprzyk, M.E.; Kazimierska, M.; Sura, W.; Dzikiewicz-Krawczyk, A.; Podralska, M. Chapter 3-Non-coding RNAs: Mechanisms of action. In *Navigating Non-Coding RNA*; Sztuba-Solinska, J., Ed.; Academic Press: Cambridge, MA, USA, 2023; pp. 89–138.
11. Doudna, J.A.; Cech, T.R. The chemical repertoire of natural ribozymes. *Nature* **2002**, *418*, 222–228.
12. Higgs, P.G.; Lehman, N. The RNA World: Molecular cooperation at the origins of life. *Nat. Rev. Genet.* **2015**, *16*, 7–17.
13. Mortimer, S.A.; Kidwell, M.A.; Doudna, J.A. Insights into RNA structure and function from genome-wide studies. *Nat. Rev. Genet.* **2014**, *15*, 469–479.
14. Meister, G.; Tuschl, T. Mechanisms of gene silencing by double-stranded RNA. *Nature* **2004**, *431*, 343–349.
15. Serganov, A.; Nudler, E. A Decade of Riboswitches. *Cell* **2013**, *152*, 17–24.
16. Wu, L.; Belasco, J.G. Let me count the ways: Mechanisms of gene regulation by miRNAs and siRNAs. *Mol. Cell* **2008**, *29*, 1–7.
17. Zou, Q.; Li, J.; Hong, Q.; Lin, Z.; Wu, Y.; Shi, H.; Ju, Y. Prediction of MicroRNA-Disease Associations Based on Social Network Analysis Methods. *BioMed Res. Int.* **2015**, *2015*, 810514, .
18. Tinoco, I.; Bustamante, C. How RNA folds. *J. Mol. Biol.* **1999**, *293*, 271–281.
19. Georgakopoulos-Soares, I.; Parada, G.E.; Hemberg, M. Secondary structures in RNA synthesis, splicing and translation. *Comput. Struct. Biotechnol. J.* **2022**, *20*, 2871–2884.
20. Celander, D.W.; Cech, T.R. Visualizing the higher order folding of a catalytic RNA molecule. *Science* **1991**, *251*, 401–407.
21. Stephens, Z.D.; Lee, S.Y.; Faghri, F.; Campbell, R.H.; Zhai, C.; Efron, M.J.; Iyer, R.; Schatz, M.C.; Sinha, S.; Robinson, G.E. Big Data: Astronomical or Genomical? *PLoS Biol.* **2015**, *13*, e1002195.
22. Zarrinkar, P.P.; Williamson, J.R. Kinetic intermediates in RNA folding. *Science* **1994**, *265*, 918–924.
23. The statistical mechanics of RNA folding. *Physics* **2006**, *35*, 218–229.
24. Zhao, Q.; Zhao, Z.; Fan, X.; Yuan, Z.; Mao, Q.; Yao, Y. Review of machine learning methods for RNA secondary structure prediction. *PLoS Comput. Biol.* **2021**, *17*, e1009291.
25. Condon, A. Problems on RNA Secondary Structure Prediction and Design. In *Automata, Languages and Programming*; (Baeten, J.C.M., Lenstra, J.K., Parrow, J., Woeginge, G., Eds.; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2003; pp. 22–32.
26. Fallmann, J.; Will, S.; Engelhardt, J.; Grüning, B.; Backofen, R.; Stadler, P.F. Recent advances in RNA folding. *J. Biotechnol.* **2017**, *261*, 97–104.
27. Seetin, M.G.; Mathews, D.H. RNA structure prediction: An overview of methods. *Methods Mol. Biol.* **2012**, *905*, 99–122.
28. Fürtig, B.; Richter, C.; Wöhnert, J.; Schwalbe, H. NMR spectroscopy of RNA. *ChemBioChem* **2003**, *4*, 936–962.
29. Westhof, E. Twenty years of RNA crystallography. *RNA* **2015**, *21*, 486–487.
30. Tijerina, P.; Mohr, S.; Russell, R. DMS Footprinting of Structured RNAs and RNA-Protein Complexes. *Nat. Protoc.* **2007**, *2*, 2608–2623.
31. Wilkinson, K.A.; Merino, E.J.; Weeks, K.M. Selective 2'-hydroxyl acylation analyzed by primer extension (SHAPE): Quantitative RNA structure analysis at single nucleotide resolution. *Nat. Protoc.* **2006**, *1*, 1610–1616.
32. Kertesz, M.; Wan, Y.; Mazor, E.; Rinn, J.L.; Nutter, R.C.; Chang, H.Y.; Segal, E. Genome-wide Measurement of RNA Secondary Structure in Yeast. *Nature* **2010**, *467*, 9322.
33. Underwood, J.G.; Uzilov, A.V.; Katzman, S.; Onodera, C.S.; Mainzer, J.E.; Mathews, D.H.; Lowe, T.M.; Salama, S.R.; Haussler, D. FragSeq: transcriptome-wide RNA structure probing using high-throughput

sequencing. *Nat. Methods* **2010**, *7*, 995–1001.
34. Bevilacqua, P.C.; Ritchey, L.E.; Su, Z.; Assmann, S.M. Genome-Wide Analysis of RNA Secondary Structure. *Annu. Rev. Genet.* **2016**, *50*, 235–266.
35. Tian, S.; Das, R. RNA structure through multidimensional chemical mapping. *Q. Rev. Biophys.* **2016**, *49*, e7.
36. RNAcentral: A comprehensive database of non-coding RNA sequences. *Nucleic Acids Res.* **2017**, *45*, D128–D134.
37. Gutell, R.R.; Lee, J.C.; Cannone, J.J. The accuracy of ribosomal RNA comparative structure models. *Curr. Opin. Struct. Biol.* **2002**, *12*, 301–310.
38. Madison, J.T.; Everett, G.A.; Kung, H. Nucleotide sequence of a yeast tyrosine transfer RNA. *Science* **1966**, *153*, 531–534.
39. Gutell, R.R.; Weiser, B.; Woese, C.R.; Noller, H.F. Comparative anatomy of 16-S-like ribosomal RNA. *Prog. Nucleic Acid Res. Mol. Biol.* **1985**, *32*, 155–216.
40. Ruan, J.; Stormo, G.D.; Zhang, W. An iterated loop matching approach to the prediction of RNA secondary structures with pseudoknots. *Bioinformatics* **2004**, *20*, 58–66.
41. Hofacker, I.L.; Fekete, M.; Flamm, C.; Huynen, M.A.; Rauscher, S.; Stolorz, P.E.; Stadler, P.F. Automatic detection of conserved RNA structure elements in complete RNA virus genomes. *Nucleic Acids Res.* **1998**, *26*, 3825–3836.
42. Bindewald, E.; Shapiro, B.A. Rna secondary structure prediction from sequence alignments using a network of k-nearest neighbor classifiers. *RNA* **2006**, *12*, 342–352.
43. Legendre, A.; Angel, E.; Tahi, F. Bi-objective integer programming for RNA secondary structure prediction with pseudoknots. *BMC Bioinformatics* **2018**, *19*, 1–15.
44. Han, K.; Kim, H.J. Prediction of common folding structures of homologous RNAs. *Nucleic Acids Res.* **1993**, *21*, 1251–1257.
45. Tahi, F.; Gouy, M.; Régnier, M. Automatic RNA secondary structure prediction with a comparative approach. *Comput. Chem.* **2002**, *26*, 521–530.
46. Nussinov, R.; Jacobson, A.B. Fast algorithm for predicting the secondary structure of single-stranded RNA. *Proc. Natl. Acad. Sci. USA* **1980**, *77*, 6309–6313.
47. Engelen, S.; Tahi, F. Tfold: Efficient in silico prediction of non-coding RNA secondary structures. *Nucleic Acids Res.* **2010**, *38*, 2453–2466.
48. Bellaousov, S.; Mathews, D.H. ProbKnot: Fast prediction of RNA secondary structure including pseudoknots. *RNA* **2010**, *16*, 1870–1880.
49. Burge, S.W.; Daub, J.; Eberhardt, R.; Tate, J.; Barquist, L.; Nawrocki, E.P.; Eddy, S.R.; Gardner, P.P.; Bateman, A. Rfam 11.0: 10 years of RNA families. *Nucleic Acids Res.* **2013**, *41*, D226–D232.
50. Xia, T.; SantaLucia, J.; Burkard, M.E.; Kierzek, R.; Schroeder, S.J.; Jiao, X.; Cox, C.; Turner, D.H. Thermodynamic parameters for an expanded nearest-neighbor model for formation of RNA duplexes with Watson-Crick base pairs. *Biochemistry* **1998**, *37*, 14719–14735.
51. Mathews, D.H.; Sabina, J.; Zuker, M.; Turner, D.H. Expanded sequence dependence of thermodynamic parameters improves prediction of RNA secondary structure. *J. Mol. Biol.* **1999**, *288*, 911–940.
52. Andronescu, M.; Condon, A.; Turner, D.H.; Mathews, D.H. The determination of RNA folding nearest neighbor parameters. *Methods Mol. Biol.* **2014**, *1097*, 45–70.
53. Turner, D.H.; Mathews, D.H. NNDB: The nearest neighbor parameter database for predicting stability of nucleic acid secondary structure. *Nucleic Acids Res.* **2010**, *38*, D280–D282.
54. Bon, M.; Micheletti, C.; Orland, H. McGenus: A Monte Carlo algorithm to predict RNA secondary structures with pseudoknots. *Nucleic Acids Res.* **2013**, *41*, 1895–1900.
55. Reeder, J.; Giegerich, R. Design, implementation and evaluation of a practical pseudoknot folding algorithm based on thermodynamics. *BMC Bioinform.* **2004**, *5*, 104.
56. Dirks, R.M.; Pierce, N.A. A partition function algorithm for nucleic acid secondary structure including pseudoknots. *J. Comput. Chem.* **2003**, *24*, 1664–1677.
57. Rivas, E.; Eddy, S.R. A dynamic programming algorithm for RNA structure prediction including pseudoknots. *J. Mol. Biol.* **1999**, *285*, 2053–2068.
58. Sato, K.; Kato, Y. Prediction of RNA secondary structure including pseudoknots for long sequences. *Brief. Bioinform.* **2021**, *23*, bbab395.
59. Poolsap, U.; Kato, Y.; Akutsu, T. Prediction of RNA secondary structure with pseudoknots using integer

programming. *BMC Bioinformatics* **2009**, *10*, 1–11.
60. Jordan, M.I.; Mitchell, T.M. Machine learning: Trends, perspectives, and prospects. *Science* **2015**, *349*, 255–260.
61. Lorenz, R.; Bernhart, S.H.; Siederdissen, C.H.Z.; Tafer, H.; Flamm, C.; Stadler, P.F.; Hofacker, I.L. ViennaRNA Package 2.0. *Algorithms Mol. Biol.* **2011**, *6*, 26.
62. Bellaousov, S.; Reuter, J.S.; Seetin, M.G.; Mathews, D.H. RNAstructure: web servers for RNA secondary structure prediction and analysis. *Nucleic Acids Res.* **2013**, *41*, W471–W474.
63. Andronescu, M.; Condon, A.; Hoos, H.H.; Mathews, D.H.; Murphy, K.P. Efficient parameter estimation for RNA secondary structure prediction. *Bioinformatics* **2007**, *23*, i19–i28.
64. Washietl, S.; Will, S.; Hendrix, D.A.; Goff, L.A.; Rinn, J.L.; Berger, B.; Kellis, M. Computational analysis of noncoding RNAs. Wiley Interdiscip. *Rev. RNA* **2012**, *3*, 759–778.
65. Tang, X.; Thomas, S.; Tapia, L.; Giedroc, D.P.; Amato, N.M. Simulating RNA folding kinetics on approximated energy landscapes. *J. Mol. Biol.* **2008**, *381*, 1055–1067.
66. Rehmsmeier, M.; Steffen, P.; Höchsmann, M.; Giegerich, R. Fast and effective prediction of microRNA/target duplexes. *RNA* **2004**, *10*, 1507–1517.
67. Zakov, S.; Goldberg, Y.; Elhadad, M.; Ziv-Ukelson, M. Rich parameterization improves RNA structure prediction. *J. Comput.Biol. A J. Comput. Mol. Cell Biol.* **2011**, *18*, 1525–1542.
68. Akiyama, M.; Sato, K.; Sakakibara, Y. A max-margin training of RNA secondary structure prediction integrated with the thermodynamic model. *J. Bioinform. Comput. Biol.* **2018**, *16*, 1840025.
69. Sato, K.; Akiyama, M.; Sakakibara, Y. Rna secondary structure prediction using deep learning with thermodynamic integration. *Nat. Commun.* **2021**, *12*, 941.
70. akakibara, Y.; Brown, M.; Hughey, R.; Mian, I.S.; Sjölander, K.; Underwood, R.C.; Haussler, D. Stochastic context-free grammars for tRNA modeling. *Nucleic Acids Res.* **1994**, *22*, 5112–5120.
71. Woodson, S.A. Recent insights on RNA folding mechanisms from catalytic RNA. *Cell. Mol. Life Sci.* **2000**, *57*, 796–808.
72. Knudsen, B.; Hein, J. RNA secondary structure prediction using stochastic context-free grammars and evolutionary history. *Bioinformatics* **1999**, *15*, 446–454.
73. Dowell, R.D.; Eddy, S.R. Evaluation of several lightweight stochastic context-free grammars for RNA secondary structure prediction. *BMC Bioinform.* **2004**, *5*, 71.
74. Rivas, E.; Lang, R.; Eddy, S.R. A range of complex probabilistic models for RNA secondary structure prediction that includes the nearest-neighbor model and more. *RNA* **2012**, *18*, 193–212.
75. Sato, K.; Hamada, M.; Mituyama, T.; Asai, K.; Sakakibara, Y. A non-parametric bayesian approach for predicting RNA secondary structures. *J. Bioinform. Comput. Biol.* **2010**, *8*, 727–742.
76. Yonemoto, H.; Asai, K.; Hamada, M. A semi-supervised learning approach for RNA secondary structure prediction. *Comput. Biol. Chem.* **2015**, *57*, 72–79.
77. Do, C.B.; Woods, D.A.; Batzoglou, S. CONTRAfold: RNA secondary structure prediction without physics-based models. *Bioinformatics* **2006**, *22*, e90–e98.
78. Hor, C.-Y.; Yang, C.-B.; Chang, C.-H.; Tseng, C.-T.; Chen, H.-H. A tool preference choice method for RNA secondary structure prediction by SVM with statistical tests. *Evol. Bioinform.* **2013**, *9*, EBO–S10580.
79. Zhu, Y.; Xie, Z.; Li, Y.; Zhu, M.; Chen, Y.-P.P. Research on folding diversity in statistical learning methods for RNA secondary structure prediction. *Int. J. Biol. Sci.* **2018**, *14*, 872–882.
80. Singh, J.; Hanson, J.; Paliwal, K.; Zhou, Y. RNA secondary structure prediction using an ensemble of two-dimensional deep neural networks and transfer learning. *Nat. Commun.* **2019**, *10*, 5407.
81. Haynes, T.; Knisley, D.; Knisley, J. Using a neural network to identify secondary RNA structures quantified by graphical invariants. *Commun. Math. Comput. Chem.* **2008**, *60*, 277–290.
82. Koessler, D.R.; Knisley, D.J.; Knisley, J.; Haynes, T. A predictive model for secondary RNA structure using graph theory and a neural network. *BMC Bioinform.* **2010**, *11*, S21.
83. Takefuji, Y.; Chen, L.L.; Lee, K.C.; Huffman, J. Parallel algorithms for finding a near-maximum independent set of a circle graph. *IEEE Trans. Neural Netw.* **1990**, *1*, 263–267.
84. Qasim, R.; Kauser, N.; Jilani, T. Secondary Structure Prediction of RNA using Machine Learning Method. *Int. J. Comput. Appl.* **2010**, *10*, 15–22.
85. Liu, Q.; Ye, X.; Zhang, Y. A Hopfield Neural Network Based Algorithm for RNA Secondary Structure Prediction. In Proceedings of the First International Multi-Symposiums on Computer and Computational

Sciences (IMSCCS'06), Hangzhou, China, 20–24 June 2006; Volume 1, pp. 10–16.
86. Apolloni, B.; Lotorto, L.; Morpurgo, A.; Zanaboni, A.M. RNA Secondary Structure Prediction by MFT Neural Networks. *Psychol. Forsch.* **2003**, *2003*, 143–148.
87. Steeg, E.W. *Neural Networks, Adaptive Optimization, and RNA Secondary Structure Prediction*; American Association for Artificial Intelligence: Palo Alto, CA, USA, 1993; pp. 121–160.
88. Singh, J.; Paliwal, K.; Zhang, T.; Singh, J.; Litfin, T.; Zhou, Y. Improved RNA secondary structure and tertiary base-pairing prediction using evolutionary profile, mutational coupling and two-dimensional transfer learning. *Bioinformatics* **2021**, *37*, 2589–2600.
89. Fu, L.; Cao, Y.; Wu, J.; Peng, Q.; Nie, Q.; Xie, X. UFold: Fast and accurate RNA secondary structure prediction with deep learning. *Nucleic Acids Res.* **2022**, *50*, e14.
90. Chen, X.; Li, Y.; Umarov, R.; Gao, X.; Song, L. RNA Secondary Structure Prediction By Learning Unrolled Algorithms. *arXiv* **2020**, arXiv:2002.05810.
91. Calonaci, N.; Jones, A.; Cuturello, F.; Sattler, M.; Bussi, G. Machine learning a model for RNA structure prediction. *NAR Genom. Bioinform.* **2020**, *2*, lqaa090.
92. Wu, H.; Tang, Y.; Lu, W.; Chen, C.; Huang, H.; Fu, Q. RNA Secondary Structure Prediction Based on Long Short-Term Memory Model. In *Intelligent Computing Theories and Application*; Huang, D.-S., Bevilacqua, V., Premaratne, P., Gupta, P., Eds.; Lecture Notes in Computer Science; Springer International Publishing: Cham, Switzerland, 2018; pp. 595–599.
93. Lu, W.; Tang, Y.; Wu, H.; Huang, H.; Fu, Q.; Qiu, J.; Li, H. Predicting RNA secondary structure via adaptive deep recurrent neural networks with energy-based filter. *BMC Bioinform.* **2019**, *20*, 684.
94. Quan, L.; Cai, L.; Chen, Y.; Mei, J.; Sun, X.; Lyu, Q. Developing parallel ant colonies filtered by deep learned constrains for predicting RNA secondary structure with pseudo-knots. *Neurocomputing* **2020**, *384*, 104–114.
95. Fei, Y.; Zhang, H.; Wang, Y.; Liu, Z.; Liu, Y. LTPConstraint: a transfer learning based end-to-end method for RNA secondary structure prediction. *BMC Bioinformatics* **2022**, *23*, 354.
96. Hochreiter, S. *Long Short-Term Memory*; Neural Computation MIT-Press: Cambridge, MA, USA, 1997.
97. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; Polosukhin, I. Attention is all you need. In *Advances in Neural Information Processing Systems*; Curran Associates, Inc.: Red Hook, NY, USA, 2017.
98. Ronneberger, O.; Fischer, P.; Brox, T. U-Net: Convolutional Networks for Biomedical Image Segmentation. *arXiv* **2015**, arXiv:1505.04597.
99. Booy, M.; Ilin, A.; Orponen, P. RNA secondary structure prediction with convolutional neural networks. *BMC Bioinformatics* **2022**, *23*, 58.
100. Zhang, H.; Zhang, C.; Li, Z.; Li, C.; Wei, X.; Zhang, B.; Liu, Y. A New Method of RNA Secondary Structure Prediction Based on Convolutional Neural Network and Dynamic Programming. *Front. Genet.* **2019**, *10*, 467.
101. Wang, L.; Liu, Y.; Zhong, X.; Liu, H.; Lu, C.; Li, C.; Zhang, H. Dmfold: A novel method to predict rna secondary structure with pseudoknots based on deep learning and improved base pair maximization principle. *Front. Genet.* **2019**, *10*, 143.
102. Willmott, D.; Murrugarra, D.; Ye, Q. Improving RNA secondary structure prediction via state inference with deep recurrent neural networks. *Comput. Math. Biophys.* **2020**, *8*, 36–50.
103. Chen, C.C.; Chan, Y.M. REDfold: Accurate RNA secondary structure prediction using residual encoder-decoder network. *BMC Bioinform.* **2023**, *24*, 122.
104. Andronescu, M.; Bereg, V.; Hoos, H.H.; Condon, A. RNA STRAND: The RNA Secondary Structure and Statistical Analysis Database. *BMC Bioinform.* **2008**, *9*, 340.
105. Burley, S.K.; Bhikadiya, C.; Bi, C.; Bittrich, S.; Chen, L.; Crichlow, G.V.; Christie, C.H.; Dalenberg, K.; Costanzo, L.D.; Duarte, J.M.; et al. RCSB Protein Data Bank: Powerful new tools for exploring 3D structures of biological macromolecules for basic and applied research and education in fundamental biology, biomedicine, biotechnology, bioengineering and energy sciences. *Nucleic Acids Res.* **2020**, *49*, D437–D451.
106. Danaee, P.; Rouches, M.; Wiley, M.; Deng, D.; Huang, L.; Hendrix, D. bprna: large-scale automated annotation and analysis of rna secondary structure. *Nucleic Acids Res.* **2018**, *46*, 5381–5394.
107. Rnacentral 2021: Secondary structure integration, improved sequence search and new member databases. *Nucleic Acids Res.* **2021**, *49*, D212–D220.
108. Sweeney, B.A.; Hoksza, D.; Nawrocki, E.P.; Ribas, C.E.; Madeira, F.; Cannone, J.J.; Gutell, R.; Maddala, A.; Meade, C.D.; Williams, L.D.; et al. R2DT is a framework for predicting and visualising RNA secondary structure using templates. *Nat. Commun.* **2021**, *12*, 3494.

109. Jühling, F.; Mörl, M.; Hartmann, R.K.; Sprinzl, M.; Stadler, P.F.; Pütz, J. tRNAdb 2009: Compilation of tRNA sequences and tRNA genes. *Nucleic Acids Res.* **2009**, *37*, D159–D162.
110. Gutell, R.R. Collection of small subunit (16S-and 16S-like) ribosomal RNA structures. *Nucleic Acids Res.* **1994**, *22*(17), 3502–3507.
111. Zwieb, C.; Gorodkin, J.; Knudsen, B.; Burks, J.; Wower, J. tmRDB (tmRNA database). *Nucleic Acids Res.* **2003**, *31*, 446–447.
112. Richardson, K.E.; Kirkpatrick, C.C.; Znosko, B.M. RNA CoSSMos 2.0: An improved searchable database of secondary structure motifs in RNA three-dimensional structures. *Database J. Biol. Databases Curation* **2020**, *2020*, baz153.
113. Korunes, K.L.; Myers, R.B.; Hardy, R.; Noor, M.A.F. PseudoBase: a genomic visualization and exploration resource for the Drosophila pseudoobscura subgroup. *Fly* **2021**, *15*, 38–44.
114. Nagaswamy, U.; Larios-Sanz, M.; Hury, J.; Collins, S.; Zhang, Z.; Zhao, Q.; Fox, G.E. NCIR: A database of non-canonical interactions in known RNA structures. *Nucleic Acids Res.* **2002**, *30*, 395–397.
115. Sloma, M.F.; Mathews, D.H. Exact calculation of loop formation probability identifies folding motifs in RNA secondary structures. *RNA* **2016**, *22*, 1808–1818.
116. Tan, Z.; Fu, Y.; Sharma, G.; Mathews, D.H. TurboFold II: RNA structural alignment and secondary structure prediction informed by multiple homologs. *Nucleic Acids Res.* **2017**, *45*, 11570–11581.
117. Kalvari, I.; Nawrocki, E.P.; Ontiveros-Palacios, N.; Argasinska, J.; Lamkiewicz, K.; Marz, M.; Griffiths-Jones, S.; Toffano-Nioche, C.; Gautheret, D.; Weinberg, Z.; et al. Rfam 14: Expanded coverage of metagenomic, viral and microrna families. *Nucleic Acids Res.* **2021**, *49*, D192–D200.
118. Wolfinger, M.T.; Svrcek-Seiler, W.A.; Flamm, C.; Hofacker, I.L.; Stadler, P.F. Efficient computation of RNA folding dynamics. *J. Phys. A: Math. Gen.* **2004**, *37*, 4731.
119. Gruber, A.R.; Findeiß, S.; Washietl, S.; Hofacker, I.L.; Stadler, P.F. RNAz 2.0: Improved noncoding RNA detection. *Pac. Symp. Biocomputing.* **2010**, *2010*, 69–79.
120. Moulton, V. Tracking down noncoding RNAs. *Proc. Natl. Acad. Sci. USA* **2005**, *102*, 2269–2270.
121. Lu, Z.J.; Mathews, D.H. Efficient siRNA selection using hybridization thermodynamics. *Nucleic Acids Res.* **2008**, *36*, 640–647.
122. Tafer, H.; Ameres, S.L.; Obernosterer, G.; Gebeshuber, C.A.; Schroeder, R.; Martinez, J.; Hofacker, I.L. The impact of target site accessibility on the design of effective siRNAs. *Nat. Biotechnol.* **2008**, *26*, 578–583.
123. Sazani, P.; Gemignani, F.; Kang, S.-H.; Maier, M.A.; Manoharan, M.; Persmark, M.; Bortner, D.; Kole, R. Systemically delivered antisense oligomers upregulate gene expression in mouse tissues. *Nat. Biotechnol.* **2002**, *20*, 1228–1233.
124. Childs-Disney, J.L.; Wu, M.; Pushechnikov, A.; Aminova, O.; Disney, M.D. A small molecule microarray platform to select RNA internal loop-ligand interactions. *ACS Chem. Biol.* **2007**, *2*, 745–754.
125. Palde, P.B.; Ofori, L.O.; Gareiss, P.C.; Lerea, J.; Miller, B.L. Strategies for Recognition of Stem-loop RNA Structures by Synthetic Ligands: Application to the HIV-1 Frameshift Stimulatory Sequence. *J. Med. Chem.* **2010**, *53*, 6018–6027.
126. Castanotto, D.; Rossi, J.J. The promises and pitfalls of RNA-interference-based therapeutics. *Nature* **2009**, *457*, 426–433.
127. Gareiss, P.C.; Sobczak, K.; McNaughton, B.R.; Palde, P.B.; Thornton, C.A.; Miller, B.L. Dynamic Combinatorial Selection of Molecules Capable of Inhibiting the (CUG) Repeat RNA – MBNL1 Interaction in vitro: Discovery of Lead Compounds Targeting Myotonic Dystrophy (DM1). *J. Am. Chem. Soc.* **2008**, *130*, 16254–16261.
128. Rouillard, J.M.; Zuker, M.; Gulari, E. OligoArray 2.0: design of oligonucleotide probes for DNA microarrays using a thermodynamic approach. *Nucleic Acids Res.* **2003**, *31*, 3057–3062.
129. Tavares, R.D.C.A.; Mahadeshwar, G.; Wan, H.; Huston, N.C.; Pyle, A.M. The Global and Local Distribution of RNA Structure throughout the SARS-CoV-2 Genome. *J. Virol.* **2021**, *95*, e02190-20.
130. Vandelli, A.; Monti, M.; Milanetti, E.; Armaos, A.; Rupert, J.; Zacco, E.; Bechara, E.; Ponti, R.D.; Tartaglia, G.G. Structural analysis of SARS-CoV-2 genome and predictions of the human interactome. *Nucleic Acids Res.* **2020**, *48*, 11270–11283.
131. Wang, X.; Gu, R.; Chen, Z.; Li, Y.; Ji, X.; Ke, G.; Wen, H. Uni-Rna: Universal Pre-Trained Models Revolutionize Rna Research. *bioRxiv* **2023**, *2023*, 548588.
132. Chen, J.; Hu, Z.; Sun, S.; Tan, Q.; Wang, Y.; Yu, Q.; Zong, L.; Hong, L.; Xiao, J.; Shen, T.; et al. Interpretable RNA Foundation Model from Unannotated Data for Highly Accurate RNA Structure and Function Predictions.

arXiv **2022**, arXiv:2204.00300.

133. Akiyama, M.; Sakakibara, Y. Informative rna base embedding for rna structural alignment and clustering by deep representation learning. *NAR Genom. Bioinform.* **2022**, *4*, lqac012.
134. Zhang, J.; Fei, Y.; Sun, L.; ; Zhang, Q.C. Advances and opportunities in RNA structure experimental determination and computational modeling. *Nat. Methods* **2022**, *19*, 1193–1207.
135. Lyngsø, R.B.; Pedersen, C.N. RNA pseudoknot prediction in energy-based models. *J. Comput. Biol.* **2000**, *7*, 409–427.
136. Parisien, M.; Major, F. The MC-Fold and MC-Sym pipeline infers RNA structure from sequence data. *Nature* **2008**, *452*, 51–55.

AI Medicine
https://www.sciltp.com/journals/aim

Article

Low Dose CT Image Denoising: A Comparative Study of Deep Learning Models and Training Strategies

Heng Zhao [1], Like Qian [1], Yaqi Zhu [1] and Dingcheng Tian [1,2,*]

[1] Research Institute for Medical and Biological Engineering, Ningbo University, Ningbo 315211, China
[2] College of Medicine and Biological Information Engineering, Northeastern University, Shenyang 110016, China
* Correspondence: 2310520@stu.neu.edu.cn

How To Cite: Zhao, H.; Qian, L.; Zhu, Y. and Tian D. Low Dose CT Image Denoising: A Comparative Study of Deep Learning Models and Training Strategies. *AI Medicine* **2024**, *1*(1), 7. https://doi.org/10.53941/aim.2024.100007.

Received: 8 August 2024
Revised: 10 October 2024
Accepted: 14 October 2024
Published: 5 November 2024

Abstract: Low-dose computed tomography (LDCT) denoising is an important topic in CT image research. Compared with normal-dose CT images, LDCT can reduce the radiation dose of X-rays, decreasing the radiation burden on the human body, which is beneficial to human health. However, quantum noise caused by low-dose rays will reduce the quality of CT images, thereby decreasing the accuracy of clinical diagnosis. In recent years, deep learning-based denoising methods have shown promising advantages in this field. Researchers have proposed some optimized models for low-dose CT image denoising. These methods have enhanced the application of low-dose CT image denoising from different aspects. From the perspective of experimental research, this paper investigates and evaluates some top deep learning models proposed in the field of low-dose image denoising in recent years, with the aim of determining the best models and training strategies for this task. We conducted experiments on seven deep learning models (REDCNN, EDCNN, QAE, OCTNet, UNet, WGAN, CTformer) on the AAPM dataset and the Piglet dataset. Our research shows that UNet has the best denoising effect among the models, obtaining PSNR = 33.06 (AAPM dataset) and PSNR = 31.21 (Piglet dataset), and good generalization capacity is also observed. However, UNet has a large number of parameters, and the time it takes to process an image is about 8 ms, while EDCNN takes about 4.8 ms to process an image, and its average PSNR is ranked second after UNet. EDCNN strikes a balance between denoising performance and processing efficiency, making it ideal for low-dose CT image denoising tasks.

Keywords: deep learning; low dose CT; image denoising; convolutional neural network

1. Introduction

Computed tomography imaging system, as a non-invasive imaging device, has been widely applied in medical diagnosis and treatment [1]. However, excessive CT scans may cause some cancers and diseases, towing to the effect of the radiation dose [2,3]. Therefore, in clinical diagnosis, it is advocated to adhere to the ALARA (As Low As Reasonably Achievable) principle [4], that is, to minimize the damage of X-rays to the human body on the premise of ensuring that the quality of CT images meets the diagnostic needs. However, during low-dose imaging, the radiation dose will affect the density distribution of X-ray photons, which will increase quantum noise, causing noise and stripe artifacts in the reconstructed image, and further, will lead to disconnected edges, smooth the target subtle structures and lack of X-ray photons resulting in low-contrast visual effects, impairing the quality of CT images and affecting the accuracy of clinical diagnosis. Since Naidich et al. [5] first proposed the concept of low-dose CT (LDCT) denoising in 1990, the issue of effectively suppressing noise and artifacts in

LDCT images has attracted much attention. Researchers have continuously optimized the design scheme from different perspectives and have achieved some outstanding results [6–12].

Early in the field of LDCT image denoising research, some studies focused on applying denoising techniques directly to the raw sinogram data [13,14]. The sinogram denoising algorithm [15,16] relies on the projection data and uses the characteristic that noise obeys the Poisson distribution in the projection data to eliminate the noise in the projection data [17]. The iterative reconstruction algorithms operate on raw data and reconstructed CT image [18,19]. These methods transfer the raw data between the image domain and the projection domain multiple times and each time update and modify the results to obtain clear CT images. In practice, raw data from commercial scanners are difficult to obtain. Therefore, many studies directly denoise the reconstructed CT images [20,21]. These methods do not require raw data and can be easily integrated into the workflow. These methods are usually based on techniques such as filtering [22,23], wavelet transform [24], dictionary learning [25], etc, to improve image quality and reduce the impact of noise. Sparse representation and non-local means have been applied to remove noise from LDCT images [8,26]. The state-of-the-art Block Matching 3D (BM3D) [27] is also employed in multiple studies to perform this task with successful results [28].

In recent years, deep learning methods have achieved advanced performance in LDCT image denoising [29–32]. Deep residual networks and convolutional Neural networks (CNN) [33] are early applications of LDCT denoising. Chen et al. [34] first proposed an LDCT denoising method based on deep neural network, this method can convert LDCT images into normal-dose CT images. Compared with traditional denoising methods, the model improved the denoising effect and computation time. Zhang et al. [35] combined dense blocks and deconvolution structures to build a lightweight network that can reuse image features to improve image quality. In addition to supervised learning methods, unsupervised learning [36,37] and semi-supervised learning models [38,39] have also achieved significant accomplishments. Generative adversarial network (GAN) are also used to improve the quality of LDCT images [40,41]. Xin et al. [42] added a sharpness detection network to the GAN network to guide the training process. The processed images have minimal resolution loss and achieve advanced performance. Yang et al. [43] proposed a CT image denoising method based on the GAN with Wasserstein distance and perceptual loss, the network reduces noise while maintaining the key information of the image. Autoencoders [44–47] achieve image denoising by learning to encode input data into a low-dimensional representation and decoding it back to the original data space. Self-supervised learning [48] and unsupervised learning use the characteristics of the data itself for training and do not require a large amount of labeled data. The semi-supervised learning method combines labeled data and unlabeled data [49], it can better utilize the information of the data and improve the generalization ability of the model and the effect of image denoising. Recent years, the emergence of Transformer [50] has also achieved remarkable results in the field of medical image processing. Luthra et al. [51] proposed a Transformer model based on edge enhancement to build the encoder and decoder. The network uses the self-attention mechanism to learn the relationship between pixels and other pixels in image blocks containing non-overlapping windows. By integrating the features of all positions to generate image details, and introducing a trainable Sobel operator to enhance the edges of the image, it provides higher performance on the AAPM dataset. In addition, a large number of combinations of deep learning and traditional denoising algorithms have also been proposed and achieved excellent results [52,53].

Although there are so many deep learning methods for LDCT image denoising [54–56], previous studies have major differences in dataset, training strategy, and performance indicators, making it impossible to evaluate the results of the models in a relatively fair manner. This paper addresses the issue through investigation and fair comparation of seven deep learning models. We conduct experiments on all models based on two datasets (AAPM and Piglet, Figure 1 shows some examples from the two datasets.) and implement seven deep learning models (REDCNN [34], EDCNN [57], QAE [58], OCTNet [59], UNet [59], WGAN [43], CTformer [60]). We systematically evaluate them from several aspects to find the best model and training strategy for the LDCT image denoising task. In summary, the work and contributions of this paper are as follows: (1) We evaluate the performance of seven deep learning models under the same metrics (PSNR, SSIM, and RMSE). (2) We conduct cross-experiments on the models based on different datasets to examine the generalization ability of the models. (3) The experiments evaluate the denoising performance of the models on CT images at various dose levels. (4) To measure the efficiency of the models, we calculate and compare the training cost and processing speed of the models.

The rest of this paper is organized as follows. Section II details the dataset and performance metrics. Section III presents the experimental results and data analysis, including the experimental setup and evaluation methods. In Section IV, we analyze and discuss the experimental results. Section V concludes the paper and presents future work.

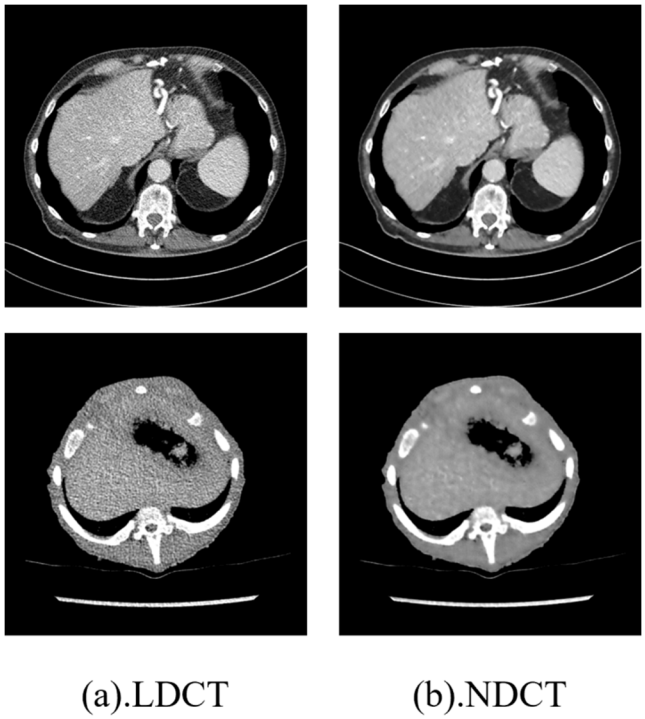


Figure 1. The dataset we used. The first row is the LDCT image and NDCT image of the AAPM dataset, and the second row is the LDCT image and NDCT image of the Piglet dataset.

2. Method

Standard datasets are crucial for model training. The datasets in this article use the Piglet dataset and the AAPM dataset. The LDCT image of the Piglet dataset is obtained by reducing the tube current, and the LDCT image of the AAPM dataset is obtained by adding Poisson noise to the original image.

2.1. Dataset

2.1.1. Piglet Dataset

The real dataset uses the Piglet dataset [42]. The LDCT image of this dataset is obtained by using a GE scanner (Discovery CT750 HD), setting the source potential and slice thickness to 100 kVp and 0.625 mm, and adjusting the tube current (or voltage), obtained by X-ray scanning with different intensities. Among them, the radiation dose when the tube current is 300 mA is the normal dose, and the tube current is reduced to 50%, 25%, 10% and 5%, respectively to obtain 4 different dose LDCT images. With different X-ray radiation doses, reconstructed CT images are subject to varying degrees of noise and artifacts. In the experiment,720 CT images are selected from the dataset as the training dataset, and 180 images are used as the test dataset. The images of the Piglet dataset during training are one-dimensional. We extracted 48,000 pairs of image patches from the 720 CT images as input and Label, size is 64 × 64. Notably, 11,520 pairs of image patches were extracted from another 180 CT images for testing. The Table 1 below provides details of the dataset. The Piglet dataset is available from the original author's GitHub repository: https://github. com/xinario/SAGAN (accessed on 2 August 2024) [42].

Table 1. Dose used for the Piglet dataset. In all 5 series, tube potential was 100 KV with 0.625 mm slice thickness. Tube current decreased to 50, 25, 10 and 5% of full-dose tube current (300 mAs) to obtain images with different doses. CTDI is the CT dose index and DLP is the dose-length product.

Dose Level	FULL	50%	25%	10%	5%
Tube current (mAs)	300	150	75	30	15
$CTDI_{vol}$ (mGy)	30.83	15.41	7.71	3.08	1.54
DLP (mGy-cm)	943.24	471.62	235.81	94.32	47.1
Effective dose (mSv)	14.14	7.07	3.54	1.41	0.71

2.1.2. AAPM Dataset

The simulation dataset is from "the 2016 NIH-AAPM-Mayo Clinic Low Dose CT Grand Challenge" [61]. The dataset contains 2378 slices from 10 anonymous patients, with a slice thickness of 1.0 mm, including LDCT images and NDCT images. The dataset is contrast-enhanced abdominal CT patient scans, each acquired during the portal venous phase using a Siemens SOMATOM Flash scanner. Among them, LDCT images are obtained by simulating noise pollution under 1/4 standard dose. In the experiment, 8 patients were selected as training data for the model, and the other 2 patients were selected as test data. Our approach is similar to other studies [1,62]. Table 2 lists the imaging conditions for each patient's original scans and the respective tube current intensities. It is worth noting that the noise in LDCT may no longer strictly follow the Poisson distribution, but the Poisson distribution is a good approximation when describing the statistical properties of X-rays, especially when the noise is relatively low. The benefit of using the Poisson noise model is that it simplifies the image reconstruction algorithm and interpretability, and it has been proven in many practical applications [63].

Table 2. Imaging conditions for the AAPM dataset. This table lists the imaging parameters for the AAPM dataset, including patient ID, number of slices, field of view (FOV) size, kilovolt peak (KVP), exposure time, and X-ray tube current (mA).

Patient ID	Numbers of Slices	Size of FOV	KVP	Exposure Time (ms)	X-ray Tube Current (mA)
L067	224	370	100	500	234.1
L096	330	430	120	500	327.6
L109	128	400	100	500	322.3
L143	234	440	120	500	416.9
L192	240	380	100	500	431.6
L286	210	380	120	500	328.9
L291	343	380	120	500	322.7
L310	214	380	120	500	300.0
L333	244	400	100	500	348.7
L506	211	380	100	500	277.7

2.2. Data Preprocessing

The data preprocessing part of this study aims to optimize the training process to better adapt to the input data requirements of the neural network model. For the original CT image size of 512 × 512, we took the following steps to process the data.

First, we introduce a key parameter patch-size, which defines the size of dividing small image patches. By dividing the image into smaller chunks, we are able to increase computational efficiency and allow the network to better learn local features. The actual size of the input image is 512 × 512, the size of the image block input to the network in the experiment is 64 × 64. Next, we preprocessed the image. First, we scaled the images to facilitate batch operations. We then convert the image data type to floating point to meet the input requirements of the neural network model. Regarding data shape conversion, we determine whether the patch-size parameter is defined based on conditions. If the patch-size parameter is set to a non-zero value, we perform

a shape transformation operation on the image. By reshaping the image into patch-size patches, we can input each patch into the neural network as an independent sample. Through these data preprocessing steps, we effectively change the form of the original CT images to better suit the needs of the neural network model. This preprocessing can improve the effectiveness of network training and enable the network to better learn the local features of the image. Our data preprocessing process is key to improving model performance and training effectiveness.

2.3. Performance Metrics

Medical images contain more subtle structures and fewer channels than natural images, and appropriate evaluation metrics are crucial to evaluating LDCT images. We choose peak signal-to-noise ratio (*PSNR*), structural similarity (*SSIM*) and root mean square error (*RMSE*) as image quality evaluation metrics. In addition to these objective metrics, radiologist evaluations are also critical to the success of the denoising task. We will include actual radiologist evaluations in subsequent studies to support the validity of diagnosis based on denoised images.

2.3.1. PSNR

PSNR is an objective measure of the error between image pixels and is typically used for error-sensitive images. It is defined according to the mean square error (*MSE*), which is defined as

$$MSE = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[Y(i,j)-X(i,j)]^2 \tag{1}$$

where *MSE* represents the mean square error between the real image *Y* and the input noise image *X*, $X(i,j)$ and $Y(i,j)$ respectively correspond to the pixel values at the coordinates. m and n represent the height and width of the image, respectively. The smaller the *MSE*, the closer the two images are and the smaller the distortion. Correspondingly, *PSNR* is expressed as

$$PSNR = 10\log_{10}(\frac{(2^n-1)^2}{MSE}) \tag{2}$$

where n is the number of bits per pixel, which is generally 8. The larger the *PSNR* value, the smaller the distortion and the better the image effect [64].

2.3.2. SSIM

SSIM (structural similarity index) stands for structural similarity. It is an index used to measure the similarity of two images. It is better in line with human visual perception. The *SSIM* formula is based on three parameters between image X and Y : Luminance, Contrast, and Structure. The formula is as follows:

$$L(x,y) = \frac{2u_x u_y + c_1}{u_x^2 + u_y^2 + c_1} \tag{3}$$

$$C(x,y) = \frac{2\sigma_x \sigma_y + c_2}{\sigma_x^2 + \sigma_y^2 + c_2} \tag{4}$$

$$S(x,y) = \frac{\sigma_x \sigma_y + c_3}{\sigma_x \sigma_y + c_3} \tag{5}$$

where U_x and U_y are the means of x and y respectively, σ_x and σ_y are the variances of x and y respectively, and σ_{xy} is the covariance of x and y , c_1, c_2 and c_3 are constants that make the result non-zero. The definition of *SSIM* can be obtained from the above three parameters:

$$SSIM(x,y) = [L(x,y)^{\alpha} * C(x,y)^{\beta} * S(x,y)^{\gamma}] \tag{6}$$

The structural similarity index defines structural information from the perspective of image composition, which reflects the structural information, brightness and contrast of the object. In the calculation of structural similarity, the mean is used as the estimate of brightness, the standard deviation is used as the estimate of contrast, and the covariance is used to measure the degree of structural similarity.

2.3.3. RMSE

Root mean square error (*RMSE*) is a common image quality evaluation metric, used to measure the degree of difference between the output image and the original image. It is defined by calculating the square root of the mean square error (*MSE*),

$$RMSE = \sqrt{MSE} \tag{7}$$

Mean squared error (*MSE*) calculates the squared difference between the pixels in two images and then averages all differences. *RMSE* is the square root of *MSE*. The smaller the value, the smaller the difference between the output image and the original image, indicating the better the processing effect.

2.4. Deep Learning Models

With the development of deep learning, medical image processing has attempted to use neural networks as a problem-solving tool. Deep learning has important roles including lesion detection and segmentation, disease prevention and diagnosis, etc. In these applications, clear medical images are crucial to solving problems. Deep learning methods have also shown good results on low-level tasks such as medical image denoising. In this paper, we evaluate seven deep learning models (REDCNN [34], EDCNN [57], QAE [58], OCTNet [59], UNet [59], WGAN [43], CTformer [60]), REDCNN, EDCNN, OCTNet, and UNet are CNN-based denoising methods, QAE s an autoencoder-based denoising method, WGAN is a GAN-based denoising method, and CTformer is a Transformer-based denoising method. Table 3 summarizes some features and parameters of the deep learning model we used.

Table 3. An overview of deep learning models. The table summarizes the deep learning models used in our study, detailing the reference numbers, the number of trainable parameters, and key features or remarks for each model.

Model	Ref	Trainable Parameters	Remarks
REDCNN	[34]	1848865	(a) Combine the autoencoder, deconvolution and shortcut connections into the ResNet.
EDCNN	[57]	80961	(a) Design an edge enhancement module based on trainable Sobel convolution. (b) Construct a dense connection to fuse edge features. (c) Introduce the compound loss which integrates the MSE loss and multi-scale perceptual loss.
QAE	[58]	49818	(a) Propose quadratic neurons by replacing the inner product. (b) Encoder-decoder structure.
OCTNet	[59]	371073	(a) Adopt multi-scale method to represent the CT denoising problem. (b) Octave convolution proposed in CT image denoising.
UNet	[59]	7819201	(a) Multiple residual connections are used for CT denoising.
WGAN	[43]	34071842	(a) Introduce a new CT image denoising method based on GAN with Wasserstein distance. (b) Use perceptual loss suppresses image noise.
CTFORMER	[60]	1448265	(a) Propose a convolution-free Token2Token dilated vision Transformer. (b) An overlapped inference mechanism effectively eliminate the boundary artifacts.

3. Experiment and Evaluation

This section shows the configuration of the experiments, presents the experimental results and brief analysis, and evaluates the LDCT image denoising performance of the deep learning models.

3.1. Experiment Design

We conducted four experiments aimed at performing denoising analysis on different deep learning models and evaluating their generalization capabilities as well as model complexity and inference speed.

In the first experiment, we performed denoising analysis using the AAPM dataset. We calculated the performance indicators of the region of interest and the enlarged image and evaluated their performance on the LDCT image denoising task. In the second experiment, to verify the generalization ability of the models on different datasets, we conducted a cross-experiment on the two datasets. The experiment allows comprehensive evaluation of the performance of the models on different datasets and verify their abilities to adapt to unseen data. In the third experiments, we conducted experiments based on the Piglet dataset. The experiment tested the generalization ability of the deep learning model to CT images with different noise levels. In the fourth experiment, we evaluated the complexity and inference speed of different models. The experiment analyzed indicators such as the number of parameters, computing resource requirements, and inference time of the models to find models with lower computational costs and efficient speed in practical applications. Through these four experiments, we can comprehensively evaluate the performance, generalization ability and computational efficiency of different deep learning models in LDCT image denoising, and explore the LDCT denoising method most suitable for this task.

3.2. Experiment Setup

We use the PyTorch 1.10 deep learning framework to implement all deep learning models, and the compilation environment for experimental is Python 3.8. We use NVIDIA RTX3090 24G GPU and Intel i9-10900X CPU to complete all model training and testing. During the optimization process, we use the Adam optimizer with default configuration and use 512 × 512 pixel size LDCT images as input. We set the learning rate to 0.00001, the batch size to 8, and conducted 200 rounds of iterative training to make the model converge. In our study, all models were trained from scratch, rather than being fine-tuned based on pre-trained models from other datasets. After training, we save the model with the best performance and evaluate it on the validation dataset. These settings were kept the same for all models.

3.3. Experiment Result and Analysis

3.3.1. Performance Result of Deep Learning Models

To quantitatively analyze the denoising performance of the deep learning model, we use peak signal-to-noise ratio (*PSNR*), structural similarity (*SSIM*) and root-mean square error (*RMSE*) as objective metrics. Table 4 shows the denoising results of the deep learning model on the AAPM dataset, the test data comes from an abdominal image of patient L506. In this experiment, the size of the input image and output image of the AAPM dataset are both 512 × 512. During training, the image is divided into small image blocks. We extracted 123072 pairs of image blocks from 1923 CT images as training Input and label, size is 64 × 64. 29,120 pairs of image blocks were extracted from another 455 CT images for testing.

Table 4. Performance of different models on the AAPM dataset. The table compares the denoising performance of various deep learning models on the AAPM dataset using three metrics: PSNR (Peak Signal-to-Noise Ratio), SSIM (Structural Similarity Index), and RMSE (Root Mean Square Error). Higher PSNR and SSIM values indicate better denoising performance, while lower RMSE values are preferred. UNet achieved the highest PSNR score, suggesting its superior ability to retain overall image quality, while EDCNN obtained the best SSIM, highlighting its strength in preserving structural details. Visual examples of the denoised images are also provided to qualitatively compare the models' outputs. Bold indicates the best results.

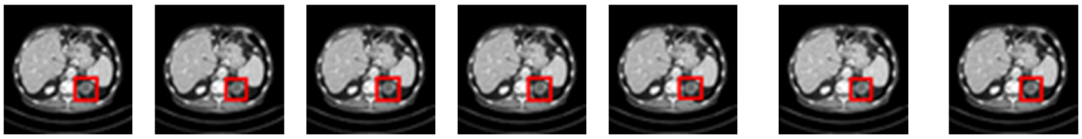
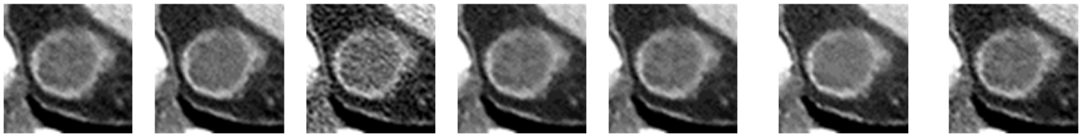
Model	REDCNN	EDCNN	QAE	OCTNet	UNet	WGAN	CTformer
PSNR	31.6918	31.8518	28.1326	31.9020	**32.2510**	30.2021	31.4673
SSIM	0.8841	**0.8972**	0.8581	0.8853	0.8884	0.8359	0.8821
RMSE	10.6134	9.9714	15.9625	9.9126	**9.7624**	12.7829	10.6773
Image							

Quadratic Autoencoder is a special autoencoder network. The network introduces a quadratic term loss function, which improves image feature capabilities and makes network training more time-consuming. It's denoised image has more noise points, and its objective indicators such as PSNR and SSIM are the worst. WGAN's images can retain the texture information of the original image, but cannot completely remove stripe artifacts. Its PSNR and SSIM results are lower than those using MSE-loss as the loss function (REDCNN, EDCNN, UNet, OCTNet). At the same time, since REDCNN only uses the MSE loss function for training, the texture of the image is blurred. EDCNN adds a trainable Sobel operator for edge enhancement before training, so the edge details of its images are more prominent. It obtains better PSNR and the best SSIM on the AAPM dataset, with SSIM = 0.9009. The denoising effects of OCTNet and UNet are close, but the denoised images of OCTNet suffer from loss of details. CTformer uses the powerful feature extraction capability of the attention mechanism to remove noise in images and achieves excellent results. Most deep learning models use gradient loss as the loss function, which will pay more attention to the subtle structure of the image, but will smooth some areas of the LDCT image. UNet can effectively remove noise and stripe artifacts and maintain high structural similarity with NDCT images. It has the highest peak signal-to-noise ratio and the smallest root mean square error, with PSNR = 32.2510 and RMSE = 9.7624.

Furthermore, we evaluate the performance metrics of regions of interest (ROI). As shown in Table 5, we zoomed in on the aorta in the chest image, the red box represents the ROI, and we calculated the test results of the local ROI and drew the ROI image. The enlarged ROI image that all models show varying degrees of denoising

effects. REDCNN and EDCNN based on the MSE loss function perform well on values and images, but have edge blur in details. The ROI image of QAE is still not good in experiments. OCTNet and UNet achieved good results with their large number of dense cascades and residual connections. The area of interest (aorta) of the two still maintains good structural similarity after amplification, and the edge information is not blurred. Although the visual effects are not as good as CTformer, it achieves the best numerical results. The WGAN network based on Wasserstein distance has texture blur at both the macro level and the micro level, which may be related to the instability of its training. The CTformer enlarged image has significant noise and blur at the edges, indicating that the ability of this type of model to process single-channel CT images has some limitations.

Table 5. Test results of the AAPM dataset on abdominal images, the red box is the region of interest ROI. This table shows the denoising performance of various models on abdominal CT images from the AAPM dataset using PSNR, SSIM, and RMSE metrics. UNet achieves the highest PSNR, indicating better overall image quality, while EDCNN achieves the highest SSIM, highlighting better structural preservation. Visualized results include both the full image and the zoomed-in ROI for a detailed comparison. Bold indicates the best results.

Model	REDCNN	EDCNN	QAE	OCTNet	UNet	WGAN	CTformer
PSNR	26.1672	26.3421	22.0271	26.5634	**26.6253**	23.7721	25.4271
SSIM	0.5669	**0.5676**	0.4831	0.5661	0.5653	0.5659	0.5422
RMSE	21.3844	21.1312	28.9304	21.4931	**21.0823**	26.7823	22.2216
Pred-img							
ROI-img							

We calculated the average PSNR and SSIM of ROI images on the AAPM dataset and Piglet dataset. As shown in Figures 2 and 3, the results show that UNet has the best PSNR and EDCNN has the best SSIM. In general, all models show certain denoising capabilities. Different networks have different problems, the output image exhibits differences in the visual effects. REDCNN and EDCNN have texture blur but rich colors and contrast. UNet performs well in terms of structure preservation and texture details, and the test results at the macro and micro levels are excellent. UNet's excellent denoising performance deserves further study for designing better models.

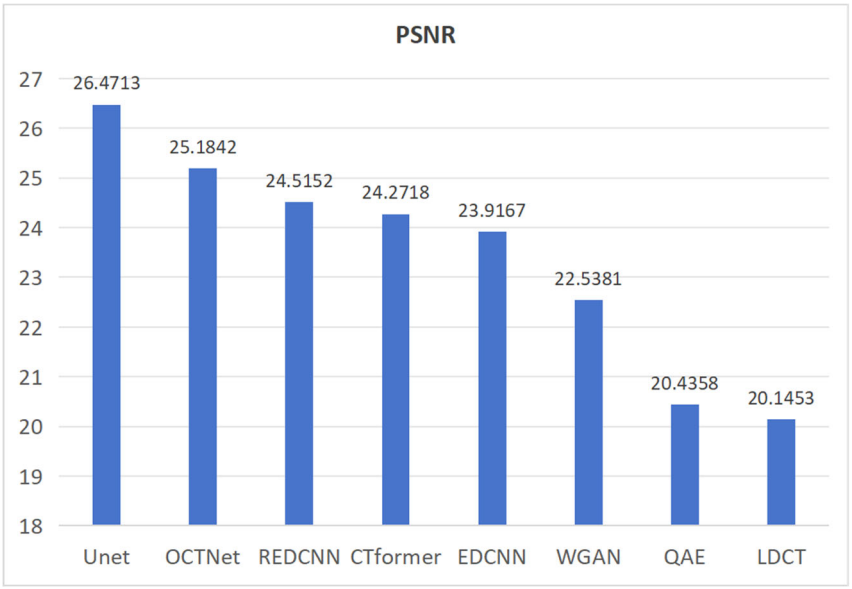


Figure 2. Average PSNR of ROI on two dataset. This figure compares the denoising performance of various models based on the average PSNR values for the ROI. Higher PSNR indicates better preservation of image quality after denoising, with UNet achieving the highest score.

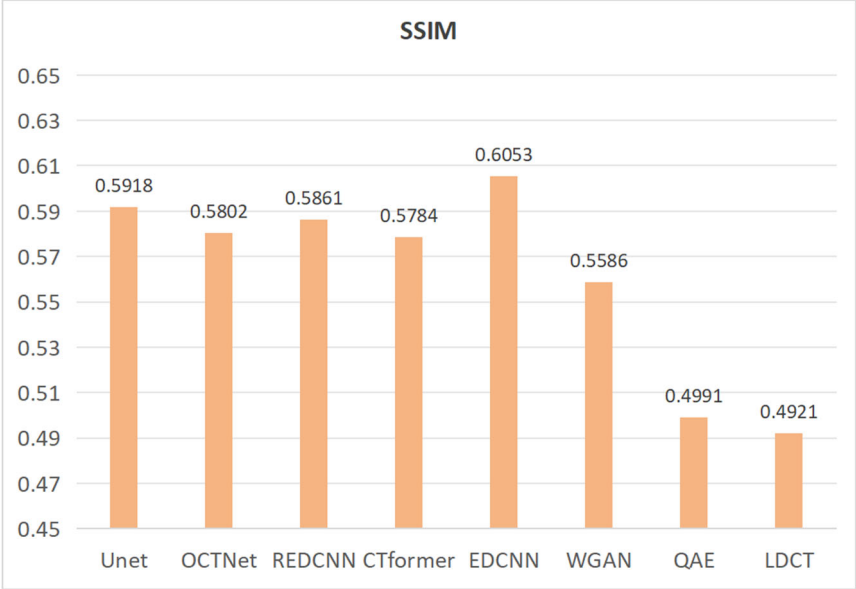


Figure 3. Average SSIM of ROI on two dataset. This figure shows the average SSIM values for the ROI, comparing the structural preservation capabilities of various models. Higher SSIM values indicate better structural similarity between the denoised and reference images, with EDCNN achieving the highest score.

3.3.2. Assessment of Generalization Performance

When using deep learning models to process LDCT denoising in actual clinical practice, their performance may be affected by acquisition parameters, equipment and other factors. To evaluate the generalization performance of deep learning models when processing new LDCT images, we tested their performance based on different datasets from two CT scanners. Specifically, we first train the model using CT images from one CT scanner, and then, we test the model using data from another CT scanner.

As shown in Table 6, when the model is trained using the AAPM dataset, the best results on the AAPM dataset are PSNR = 33.0712, SSIM = 0.9221, and the best results on the Piglet dataset are PSNR = 27.9170, SSIM = 0.8616. When the model is trained using the Piglet dataset, the best results on the Piglet dataset are PSNR = 31.2192, SSIM = 0.8969, and the best results on the AAPM dataset are PSNR = 28.9191, SSIM = 0.8615. Among all models, OCTNet achieved better results in the model generalization performance test. The results show that the deep learning model performs better when the test and training data come from CT images from the same CT scanner. In summary, it is difficult to obtain the same performance when using a trained deep learning model to test new LDCT images.

Table 6. Image quality evaluation results of the model on two datasets. This table displays PSNR and SSIM results for each model when trained and tested on AAPM and Piglet datasets, highlighting their performance on both same-dataset and cross-dataset scenarios. UNet excels in same-dataset tests, while OCTNet shows better cross-dataset generalization. Bold indicates the best results.

Model	**Train: AAPM Test: AAPM**		**Train: AAPM Test: PIGLET**		**Train: PIGLET Test: PIGLET**		**Train: PIGLET Test: AAPM**	
INDEX	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
LDCT	29.2454	0.8732	25.0054	0.7234	25.0054	0.7234	29.2454	0.8732
REDCNN	32.3221	0.9103	27.3561	0.8371	30.6551	0.8965	28.5451	0.8472
EDCNN	32.9791	0.9037	27.0181	0.8231	31.1081	0.8899	28.0169	0.8349
QAE	29.2291	0.8759	25.0063	0.7762	25.0172	0.7881	26.0071	0.7876
OCTNet	32.7813	0.9082	**27.9170**	**0.8616**	31.1102	0.8971	**28.9191**	**0.8615**
UNet	**33.0712**	**0.9221**	27.6974	0.8421	**31.2192**	**0.8969**	28.6885	0.8571
WGAN	30.5192	0.8882	26.9159	0.8334	27.8292	0.8251	27.6601	0.8102
CTformer	32.2071	0.9092	27.0331	0.8264	30.3482	0.8020	28.0330	0.8351

3.3.3. Assessment of Training Strategies

To further explore the denoising ability of the deep learning model on CT images with different noise levels, we conducted the following experiments. Observe the performance of various deep learning models by varying the radiation dose. We use the Piglet dataset to train the model and test it on LDCT images with different noise

levels. Less radiation dose means more noise. Figure 4 shows CT images of the Piglet dataset denoised using different methods. Table 7 shows the objective indicators of various methods.

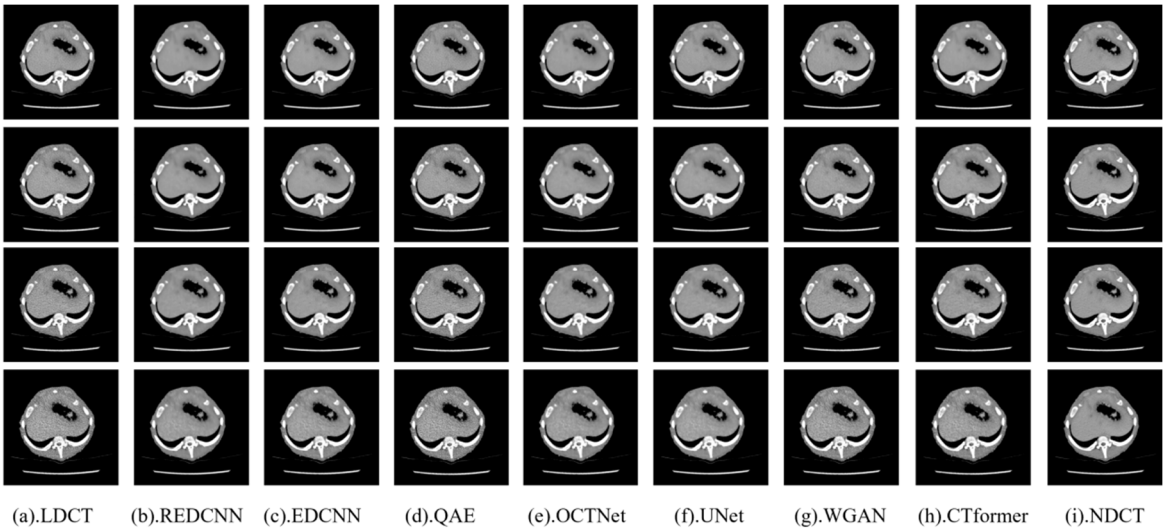


Figure 4. The above figure is a visualization of four low doses CT images in the Piglet dataset using different methods to denoise. The first row is LDCT (50% of full dose reconstructed by FBP). The second row is LDCT (25% of full dose reconstructed by FBP). The third row is LDCT (10% of full dose reconstructed by FBP). The last row is LDCT (5% of full dose reconstructed by FBP). The last column NDCT is reconstructed by the FBP algorithm with a 100% dose.

Table 7. Test results when trained on the Piglet dataset with 100% dose images. This table shows the PSNR, SSIM, and RMSE results for models tested on CT images with different dose levels (50%, 25%, 10%, and 5%). It highlights how each model adapts to lower dose levels, with EDCNN and UNet showing superior results at various dose reductions. Bold indicates the best results.

Model	50% Dose			25% Dose			10% Dose			5% Dose		
	PSNR	SSIM	RMSE	PSNR	SSIM	RMSE	PSNR	SSIM	RMSE	PSNR	SSIM	RMSE
LDCT	31.0736	0.8771	11.1785	28.0292	0.8414	15.8708	26.7750	0.8114	18.3361	24.1211	0.7779	24.8889
REDCNN	28.5556	0.8940	14.9375	28.2353	0.8935	15.1827	31.4215	0.8963	10.7395	28.2437	0.8949	13.9151
EDCNN	**32.8634**	0.9004	**9.9734**	**30.7741**	0.8877	**12.1470**	31.5269	0.9077	10.5100	29.7512	**0.8962**	12.8965
QAE	31.0968	0.8769	11.1486	28.0512	0.8411	15.8307	26.7689	0.8113	18.3491	25.0084	0.7893	22.9941
OCTNet	31.5507	0.9041	12.1175	30.7046	0.8904	12.4709	31.5017	0.9017	10.6408	28.8929	0.8846	14.3683
UNet	32.2654	**0.9198**	10.6591	30.2083	**0.8999**	12.1786	**31.5763**	**0.9066**	**10.5206**	**29.9512**	0.8874	**12.6204**
WGAN	29.2374	0.8226	13.1043	29.0836	0.8812	14.0566	28.8378	0.8620	14.4600	26.0153	0.8141	20.0122
CTformer	29.8627	0.8720	12.1072	29.8460	0.8944	13.2094	29.3495	0.8605	13.6329	28.7136	0.8469	14.9342

As shown in Figure 4, in the first column, with the gradual reduction of radiation dose, the noise in CT images increases significantly. When the dose is 5% of the full dose, noise seriously affects the visual effect of CT images. Notably, the RED-CNN can remove noise to some extent, but its images inevitably exhibit a smoothing effect and lose some details. WGAN performs well when noise is low, but when it processes LDCT images with the highest noise levels, its denoised images will have many noise points. In contrast, EDCNN, OCTNet, UNet and CTformer denoise CT images of different doses, and the image quality obtained is significantly better than other algorithms. The the model based on MSE-loss and compound loss function performs well in obtaining LDCT denoised images at different radiation doses. The denoised image can retain the rich details and texture structure of the CT image well. Among all models, UNet has better visual effects on denoised images for all four dose levels.

We quantitatively analyze the denoising performance of different algorithms for different LDCT images. We calculated three objective indicators of the experimental results. The summary data are in Table 7. Of note, the LDCT image with a radiation dose of 50%, PSNR and SSIM are higher, which indicates that the 50% reduction of radiation dose has little effect on image quality. However, for images with a radiation dose of 5%, the values of the three metrics decreased significantly. That is, the image quality of Figure 4 becomes worse as the dose

decreases. The data show that UNet and EDCNN have better results in most cases, and both of them are ranked first and second in four rounds of tests (bold numbers represent the best, italics numbers represent the second best.). UNet has been ranked first in three rounds of tests many times, indicating that UNet has some robustness. Therefore, UNet performs well in denoising experiments on CT images with different noise levels.

3.3.4. Model Complexity Evaluation

Model efficiency is an important issue in deep learning. An excellent deep learning model should have both excellent denoising capabilities and fast inference speed. Based on the above criteria, we compared the number of trainable parameters (params), memory usage (MACs), and inference speed based on different devices (CPU and GPU) of the seven models. The experiment completes all experiments using Intel i9-10900X CPU and NVIDIA RTX3090 24G GPU.

QAE uses 15 3 × 3 kernels in each convolutional layer, while REDCNN has 32 5 × 5 kernels, which means that REDCNN has 4 times more parameters than QAE. As shown in Table 8, WGAN occupies the largest real-time memory. WGAN includes a generator and a discriminator and uses perceptual loss as the loss function. Its trainable parameter amount and memory usage are the highest, which makes it difficult to deploy the whole model. On the contrary, EDCNN has a relatively small number of parameters but a high memory usage, indicating that the network can effectively fuse image information, which can also explain its better PSNR. In addition to this, models process images with a CPU takes much longer than with a GPU.

Table 8. CPU computation speed and GPU computation speed for the two datasets on seven models. This table presents the parameter count, MACs, and computation times on both CPU and GPU for seven models, along with their average PSNR scores. It highlights the efficiency and speed differences between models when processing the AAPM and Piglet datasets, with CTformer achieving the fastest CPU computation time on both datasets. Bold indicates the best results.

Model	Params	MACs (G)	AAPM-Dataset		Piglet-Dataset		Avg-PSNR
			CPU Times (ms)	GPU Times (ms)	CPU Times (ms)	GPU Times (ms)	
REDCNN	1848865	4.3	3182.1	12.1	109.2	5.2	31.4277
EDCNN	80961	5.2	571.6	5.9	159.1	3.7	32.0321
QAE	49818	2.5	1024.4	**4.8**	315.6	**2.8**	27.1272
OCTNet	371073	3.1	684.2	13.4	229.3	6.5	31.8922
UNet	7819201	4.9	726.3	11.4	238.2	4.6	**32.4521**
WGAN	34071842	6.4	3682.1	28.2	1317.2	15.6	29.1281
CTformer	1448265	6.2	**531.8**	9.4	**134.6**	3.5	31.3180

In our experiments, the number of images in the AAPM and piglet test datasets were 1923 and 720, respectively. We calculated the processing speed of a single 512 × 512 LDCT image. QAE has the fastest inference speed, with a single image taking 3.9 ms. WGAN is the slowest, which also means that training WGAN takes more time. Therefore, we plotted a scatter plot of the model's inference time versus PSNR, as shown in Figure 5, with the best results in the upper left. The results show that only UNet and EDCNN have PSNRs exceeding 32. Among them, UNet has the highest PSNR, but UNet's inference time is slower. EDCNN achieves a balance between inference time and de-noising effect. Therefore, in terms of calculation speed and denoising performance, EDCNN is the strongest competitor compared to other models.

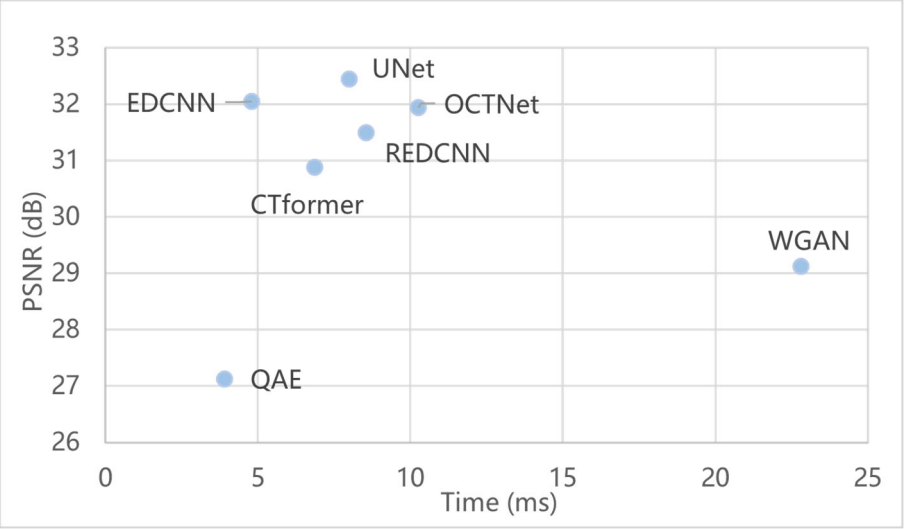


Figure 5. Performance results and inference speed of different deep learning models. This figure illustrates the trade-off between denoising performance (measured by PSNR) and inference speed (time in ms) for various models, highlighting the balance between image quality and computational efficiency.

4. Discussion

Deep learning already occupies a significant position in medical image processing. In LDCT image denoising research, many studies have different experimental conditions and training strategies. To accurately judge and compare the denoising performance of networks, in our study, we trained and evaluated seven deep learning models under the same conditions and studied their training strategies.

In the denoising performance analysis, UNet has the best effect. Through multi-layer residual connections, UNet can extract more image information and obtain the best results, with a PSNR of 32.25. The successful performance of the UNet architecture is due in part to its features specifically designed for biomedical image segmentation, including efficient utilization of small amounts of training data. However, it is worth noting that regardless of the size of the dataset, UNet is also likely to improve performance due to its efficient architecture. In our first experiment (3.1.1) and third experiment (3.1.3), the two experiments are based on different datasets and the number of images in the datasets is different. In both experiments, UNet achieved excellent results. Therefore, UNet's own efficient architecture is the main reason for its success in small datasets. Similarly, REDCNN also achieves good performance using symmetric encoders and decoders. EDCNN introduces a trainable sobel operator before the residual connection to enhance the edge information of the output image, thus achieving better results. The LDCT denoising network based on CNN showed better performance, while the denoising network based on GAN and transformer was overall inferior to the denoising network based on CNN. Additionally, we zoom in on the region of interest to focus on detail recovery and edge information. Regarding detail recovery, EDCNN's denoised image shows no obvious noise after enlargement, and the edge details of blood vessels are not blurred, indicating that its denoising effect is better than other models. Overall, the top-performing models did not exhibit significant differences in structural similarity.

In deep learning, generalization performance is one of the important metrics for evaluating the stability of the model. In our study, we conducted cross-experiments to explore the model's generalization ability on different datasets. UNet shows excellent denoising performance on the same dataset, while OCTNet has better denoising performance on different datasets. Therefore, OCTNet has stronger generalization ability than other models.

In LDCT image processing, model running speed is also an important metric. Experimental results show that QAE has the fastest computing speed, which is consistent with its size, but its denoising performance is poor. WGAN consists of a generator and a discriminator and uses the VGG network as a feature extractor and complex loss function, and its calculation speed is the slowest. Overall, EDCNN balances computational speed and denoising performance.

Our work has some shortcomings that we hope to address in the future. First, in this study, we employed the AAPM dataset and the Piglet dataset. The AAPM dataset primarily contains contrast-enhanced abdominal CT images, while the Piglet dataset consists of low-dose CT images of experimental pigs obtained by reducing the tube current. These two datasets represent different anatomical regions—human abdomen and experimental pig—providing an opportunity to evaluate the applicability of the models across varying anatomical areas. Further validation of the models' performance on CT data from different anatomical regions allows comprehensive

assessment of the denoising methods' generalizability and applicability. In future research, we plan to test these denoising methods on CT datasets from other anatomical regions, such as the brain and chest, to more thoroughly evaluate their effectiveness and explore their potential in various clinical scenarios. Secondly, generalizability is a key issue. When evaluating the generalization ability of denoising methods, we did not consider their versatility. For example, a method trained on brain CT scans should also be applicable to chest CT scans. In future research, we plan to test these denoising methods on CT datasets from more anatomical regions (such as the brain, chest, etc.) to further explore their transferability and applicability across different anatomical regions. Moreover, transfer learning can utilize models pre-trained on large-scale datasets to provide better initial feature representations for other related tasks, thereby improving model training efficiency and generalization ability. This is especially recognized in medical image analysis [65–67]. In our study, although all models were trained from scratch, we acknowledge the potential advantages of transfer learning in enhancing model performance. In future research, we plan to evaluate the effectiveness of transfer learning in LDCT denoising, such as using models pre-trained on other CT image tasks as initial models and observing their impact on LDCT denoising. We believe that this exploration could further improve the denoising performance of the models and validate their broader applicability across different datasets and tasks. Finally, although we have conducted many sets of experiments, considering many top deep learning models, We need to continue updating the latest neural network models to take advantage of the new deep learning advancement.

5. Conclusion

In our study, we implemented and evaluated the performance and efficiency of seven LDCT denoising models. The results show that UNet has the best performance in terms of PSNR, due to its multi-layer residual connected encoder. The output image of EDCNN is most similar to the original image and has the highest structural similarity. UNet has better denoising effect, but the calculation time is longer, which will increase the time consumption in actual clinical processing. In contrast, EDCNN can balance performance and efficiency, which has potential for practical applications. In addition, to assess the model's performance on new data, we evaluated its generalization performance, providing a benchmark for future research.

Author Contributions

All authors contributed to the study's conception and design. Material preparation and data collection were performed by H.Z., Y.Z., L.Q., and D.T. The first draft of the manuscript was written by H.Z. and D.T. All authors have read and agreed to the published version of the manuscript.

Funding

This research received no external funding.

Institutional Review Board Statement

Not applicable.

Informed Consent Statement

Not applicable.

Data Availability Statement

We are using a public dataset, and all the data used is publicly available.

Conflicts of Interest

The authors have no relevant financial or non-financial interests to disclose.

References

1. Zhang, Z.; Yu, L.; Liang, X.; Zhao, W.; Xing, L. TransCT: Dual-path transformer for low dose computed tomography. In Proceedings of the Medical Image Computing and Computer Assisted Intervention–MICCAI 2021: 24th International Conference, Strasbourg, France, 27 September–1 October 2021; Part VI 24, pp. 55–64.
2. Jiang, H. *Computed Tomography: Principles, Design, Artifacts, and Recent Advances*; SPIE: Bellingham, WA, USA, 2009.

3. Brenner, D.J.; Hall, E.J. Computed Tomography — An Increasing Source of Radiation Exposure. *N. Engl. J. Med.* **2007**, *357*, 2277–2284.
4. de Gonzalez, A.B.; Darby, S. Risk of cancer from diagnostic X-rays: Estimates for the UK and 14 other countries. *Lancet* **2004**, *363*, 345–351.
5. Naidich, D.P.; Marshall, C.H.; Gribbin, C.; Arams, R.S.; McCauley, D.I. Low-dose CT of the lungs: preliminary observations. *Radiology* **1990**, *175*, 729–731.
6. Yin, X.; Coatrieux, J.-L.; Zhao, Q.; Liu, J.; Yang, W.; Yang, J.; Quan, G.; Chen, Y.; Shu, H.; Luo, L. Domain Progressive 3D Residual Convolution Network to Improve Low-Dose CT Imaging. *IEEE Trans. Med Imaging* **2019**, *38*, 2903–2913.
7. Han, Y.S.; Yoo, J.; Ye, J.C. Deep residual learning for compressed sensing CT reconstruction via persistent homology analysis. *arXiv* 2016, arXiv:1611.06391.
8. Chen, Y.; Yin, X.; Shi, L.; Shu, H.; Luo, L.; Coatrieux, J.-L.; Toumoulin, C. Improving abdomen tumor low-dose CT images using a fast dictionary learning based processing. *Phys. Med. Biol.* **2013**, *58*, 5803–5820.
9. Thanh, D.; Surya, P.; Hieu, L.M. A Review on CT and X-Ray Images Denoising Methods. *Informatica* **2019**, *43*, 151–159.
10. Diwakar, M.; Kumar, M. A review on CT image noise and its denoising. *Biomed. Signal Process. Control.* **2018**, *42*, 73–88.
11. Wang, H.; Chi, J.; Wu, C.; Yu, X.; Wu, H. Degradation adaption localto-global transformer for low-dose CT image denoising. *J. Digit. Imaging* **2023**, *36*, 1894–1909.
12. Chen, Z.; Gao, Q.; Zhang, Y.; Shan, H. Ascon: Anatomy-aware supervised contrastive learning framework for low-dose CT denoising. In Proceedings of the International Conference on Medical Image Computing and Computer Assisted Intervention, Vancouver, BC, Canada, 8–12 October 2023; pp. 355–365.
13. Manduca, A.; Yu, L.; Trzasko, J.D.; Khaylova, N.; Kofler, J.M.; McCollough, C.M.; Fletcher, J.G. Projection space denoising with bilateral filtering and CT noise modeling for dose reduction in CT. *Med. Phys.* **2009**, *36*, 4911–4919.
14. Kachelriess, M.; Watzke, O.; Kalender, W.A. Generalized multidimensional adaptive filtering for conventional and spiral single-slice, multi-slice, and cone-beam CT. *Med. Phys.* **2001**, *28*, 475–490.
15. Hsieh, J. Adaptive streak artifact reduction in computed tomography resulting from excessive X-ray photon noise. *Med. Phys.* **1998**, *25*, 2139–2147.
16. Wang, J.; Li, T.; Lu, H.; Liang, Z. Penalized weighted least-squares approach to sinogram noise reduction and image reconstruction for low-dose X-ray computed tomography. *IEEE Trans. Med. Imaging* **2006**, *25*, 1272–1283.
17. Zeng, D.; Huang, J.; Bian, Z.; Niu, S.; Zhang, H.; Feng, Q.; Liang, Z.; Ma, J. A Simple Low-Dose X-Ray CT Simulation from High-Dose Scan. *IEEE Trans. Nucl. Sci.* **2015**, *62*, 2226–2233.
18. Fletcher, J.G.; Grant, K.L.; Fidler, J.L.; Shiung, M.; Yu, L.; Wang, J.; Schmidt, B.; Allmendinger, T.; McCollough, C.H. Validation of dual source single-tube reconstruction as a method to obtain half-dose images to evaluate radiation dose and noise reduction: Phantom and human assessment using CT colonography and sinogram-affirmed iterative reconstruction (safire). *J. Comput. Assist. Tomogr.* **2012**, *36*, 560–569.
19. Pickhardt, P.J.; Lubner, M.G.; Kim, D.H.; Tang, J.; Ruma, J.A.; del Rio, A.M.; Chen, G.-H. Abdominal CT with Model-Based Iterative Reconstruction (MBIR): Initial Results of a Prospective Trial Comparing Ultralow-Dose with Standard Dose Imaging. *Am. J. Roentgenol.* **2012**, *199*, 1266–1274.
20. Litjens, G.; Kooi, T.; Bejnordi, B.E.; Setio, A.A.A.; Ciompi, F.; Ghafoorian, M.; van der Laak, J.A.W.M.; van Ginneken, B.; Sánchez, C.I. A survey on deep learning in medical image analysis. *Med. Image Anal.* **2017**, *42*, 60–88.
21. Kaur, P.; Singh, G.; Kaur, P. A review of denoising medical images using machine learning approaches. *Curr. Med. Imaging* **2018**, *14*, 675–685.
22. Buades, A.; Coll, B.; Morel, J.M. A non-local algorithm for image denoising. In Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA, 20–25 June 2005; Volume 2, pp. 60–65.
23. Balda, M.; Hornegger, J.; Heismann, B. Ray Contribution Masks for Structure Adaptive Sinogram Filtering. *IEEE Trans. Med Imaging* **2012**, *31*, 1228–1239.
24. Mallat, S.G. A theory for multiresolution signal decomposition: The wavelet representation. *IEEE Trans. Pattern Anal. Mach. Intell.* **1989**, *11*, 674–693.
25. Yu, F.; Chen, Y.; Luo, L. CT image denoising based on sparse representation using global dictionary. In Proceedings of the 2013 ICME International Conference on Complex Medical Engineering, Beijing, China, 25–28 May 2013; pp. 408–411.
26. Chen, Y.; Yang, Z.; Hu, Y.; Yang, G.; Zhu, Y.; Li, Y.; Luo, L.; Chen, W.; Toumoulin, C. Thoracic low-dose CT image processing using an artifact suppressed large-scale nonlocal means. *Phys. Med. Biol.* **2012**, *57*, 2667–2688.
27. Dabov, K.; Foi, A.; Katkovnik, V.; Egiazarian, K. Image Denoising by Sparse 3-D Transform-Domain Collaborative Filtering. *IEEE Trans. Image Process.* **2007**, *16*, 2080–2095.

28. Hashemi, S.; Paul, N.S.; Beheshti, S.; Cobbold, R.S.C. Adaptively Tuned Iterative Low Dose CT Image Denoising. *Comput. Math. Methods Med.* **2015**, *2015*, 638568.
29. Ha, S.; Mueller, K. Low dose CT image restoration using a database of image patches. *Phys. Med. Biol.* **2015**, *60*, 869–882.
30. Zhang, Z.; Han, X.; Pearson, E.; Pelizzari, C.; Sidky, E.Y.; Pan, X. Artifact reduction in short-scan CBCT by use of optimization-based reconstruction. *Phys. Med. Biol.* **2016**, *61*, 3387–3406.
31. Chen, H.; Zhang, Y.; Kalra, M.K.; Lin, F.; Chen, Y.; Liao, P.; Zhou, J.; Wang, G. Low-Dose CT With a Residual Encoder-Decoder Convolutional Neural Network. *IEEE Trans. Med. Imaging* **2017**, *36*, 2524–2535.
32. Shan, H.; Padole, A.; Homayounieh, F.; Kruger, U.; Khera, R.D.; Nitiwarangkul, C.; Kalra, M.K.; Wang, G. Competitive performance of a modularized deep neural network compared to commercial algorithms for low-dose CT image reconstruction. *Nat. Mach. Intell.* **2019**, *1*, 269–276.
33. Kang, E.; Chang, W.; Yoo, J.; Ye, J.C. Deep Convolutional Framelet Denosing for Low-Dose CT via Wavelet Residual Network. *IEEE Trans. Med Imaging* **2018**, *37*, 1358–1369.
34. Chen, H.; Zhang, Y.; Zhang, W.; Liao, P.; Li, K.; Zhou, J.; Wang, G. Low dose CT denoising with convolutional neural network. In Proceedings of the 2017 IEEE 14th International Symposium on Biomedical Imaging (ISBI 2017), Melbourne, VIC, Australia, 18–21 April 2017; pp. 143–146.
35. Zhang, Y.; Tian, Y.; Kong, Y.; Zhong, B.; Fu, Y. Residual dense network for image super-resolution. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 2472–2481.
36. Rai, S.; Bhatt, J.S.; Patra, S.K. Augmented Noise Learning Framework for Enhancing Medical Image Denoising. *IEEE Access* **2021**, *9*, 117153–117168.
37. Rai, S.; Bhatt, J.S.; Patra, S.K. Accessible, affordable and low-risk lungs health monitoring in COVID-19: Deep cascade reconstruction from degraded lr-uldct. In Proceedings of the 2022 IEEE 19th International Symposium on Biomedical Imaging (ISBI), Kolkata, India, 28–31 March 2022; pp. 1–5.
38. Choi, K.; Vania, M.; Kim, S. Semi-supervised learning for lowdose CT image restoration with hierarchical deep generative adversarial network (hd-gan). In Proceedings of the 2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Berlin, Germany, 23–27 July 2019; pp. 2683–2686.
39. Wang, L.; Gao, Q.; Meng, M.; Li, S.; Zhu, M.; Li, D.; Chen, G.; Zeng, D.; Xie, Q.; Zhao, Q.; et al. Semi-supervised noise distribution learning for low-dose CT restoration. *Med. Imaging 2020 Phys. Med. Imaging* **2020**, *11312*, 1026–1030.
40. Bizopoulos, P.; Vretos, N.; Daras, P. Comprehensive comparison of deep learning models for lung and COVID-19 lesion segmentation in CT scans. *arXiv* 2020, arXiv:2009.06412, 2020.
41. Shahidi, F.; Daud, S.M.; Abas, H.; Ahmad, N.A.; Maarop, N. Breast Cancer Classification Using Deep Learning Approaches and Histopathology Image: A Comparison Study. *IEEE Access* **2020**, *8*, 187531–187552.
42. Yi, X.; Babyn, P. Sharpness-Aware Low-Dose CT Denoising Using Conditional Generative Adversarial Network. *J. Digit. Imaging* **2018**, *31*, 655–669.
43. Yang, Q.; Yan, P.; Zhang, Y.; Yu, H.; Shi, Y.; Mou, X.; Kalra, M.K.; Zhang, Y.; Sun, L.; Wang, G. Low-Dose CT Image Denoising Using a Generative Adversarial Network with Wasserstein Distance and Perceptual Loss. *IEEE Trans. Med. Imaging* **2018**, *37*, 1348–1357.
44. Nishio, M.; Nagashima, C.; Hirabayashi, S.; Ohnishi, A.; Sasaki, K.; Sagawa, T.; Hamada, M.; Yamashita, T. Convolutional auto-encoder for image denoising of ultra-low-dose CT. *Heliyon* **2017**, *3*, e00393.
45. Liu, Y.; Zhang, Y. Low-dose CT restoration via stacked sparse denoising autoencoders. *Neurocomputing* **2018**, *284*, 80–89.
46. Liu, H.; Liao, P.; Chen, H.; Zhang, Y. ERA-WGAT: Edge-enhanced residual autoencoder with a window-based graph attention convolutional network for low-dose CT denoising. *Biomed. Opt. Express* **2022**, *13*, 5775–5793.
47. Wang, D.; Xu, Y.; Han, S.; Yu, H. Masked autoencoders for low-dose CT denoising. In Proceedings of the 2023 IEEE 20th International Symposium on Biomedical Imaging (ISBI), Cartagena, Colombia, 18–21 April 2023; pp. 1–4.
48. Li, M.; Hsu, W.; Xie, X.; Cong, J.; Gao, W. SACNN: Self-Attention Convolutional Neural Network for Low-Dose CT Denoising With Self-Supervised Perceptual Loss Network. *IEEE Trans. Med. Imaging* **2020**, *39*, 2289–2301.
49. Karimi, D.; Dou, H.; Warfield, S.K.; Gholipour, A. Deep learning with noisy labels: Exploring techniques and remedies in medical image analysis. *Med. Image Anal.* **2020**, *65*, 101759.
50. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 1–11.
51. Luthra, A.; Sulakhe, H.; Mittal, T.; Iyer, A.; Yadav, S. Eformer: Edge enhancement based transformer for medical image denoising. *arXiv* 2021, arXiv:2109.08044.
52. Yuan, J.; Zhou, F.; Guo, Z.; Li, X.; Yu, H. HCformer: Hybrid CNN-Transformer for LDCT Image Denoising. *J. Digit. Imaging* **2023**, *36*, 2290–2305.

53. Chyophel Lepcha, D.; Goyal, B.; Dogra, A. Low-dose CT image denoising using sparse 3dD transformation with probabilistic non-local means for clinical applications. *Imaging Sci. J.* **2023**, *71*, 97–109.
54. Othman, A.E.; Brockmann, C.; Yang, Z.; Kim, C.; Afat, S.; Pjontek, R.; Nikoubashman, O.; Brockmann, M.A.; Nikolaou, K.; Wiesmann, M.; et al. Impact of image denoising on image quality, quantitative parameters and sensitivity of ultra-low-dose volume perfusion CT imaging. *Eur. Radiol.* **2015**, *26*, 167–174.
55. Kulathilake, K.A.S.H.; Abdullah, N.A.; Sabri, A.Q.M.; Lai, K.W. A review on Deep Learning approaches for low-dose Computed Tomography restoration. *Complex Intell. Syst.* **2021**, *9*, 2713–2745.
56. Mück, J.; Reiter, E.; Klingert, W.; Bertolani, E.; Schenk, M.; Nikolaou, K.; Afat, S.; Brendlin, A.S. Towards safer imaging: A comparative study of deep learning-based denoising and iterative reconstruction in intraindividual low-dose CT scans using an in-vivo large animal model. *Eur. J. Radiol.* **2023**, *171*, 111267.
57. Liang, T.; Jin, Y.; Li, Y.; Wang, T. Edcnn: Edge enhancement-based densely connected network with compound loss for low-dose CT denoising. In Proceedings of the 2020 15th IEEE International Conference on Signal Processing (ICSP), Beijing, China, 6–9 December 2020; Volume 1, pp. 193–198.
58. Fan, F.; Shan, H.; Kalra, M.K.; Singh, R.; Qian, G.; Getzin, M.; Teng, Y.; Hahn, J.; Wang, G. Quadratic Autoencoder (Q-AE) for Low-Dose CT Denoising. *IEEE Trans. Med. Imaging* **2019**, *39*, 2035–2050.
59. Won, D.K.; An, S.; Park, S.H.; Ye, D.H. Low-dose CT denoising using octave convolution with high and low frequency bands. In *Predictive Intelligence in Medicine: Third International Workshop, PRIME 2020, Held in Conjunction with MICCAI 2020, Lima, Peru, 8 October 2020*; Springer: Cham, Switzerland, 2020; pp. 68–78.
60. Wang, D.; Fan, F.; Wu, Z.; Liu, R.; Wang, F.; Yu, H. CTformer: convolution-free Token2Token dilated vision transformer for low-dose CT denoising. *Phys. Med. Biol.* **2023**, *68*, 065012.
61. AAPM. Low Dose CT Grand Challenge. 2017. Available online: http://www.aapm.org/grandchallenge/lowdosect/ (accessed on 2 August 2024).
62. Yang, L.; Shangguan, H.; Zhang, X.; Wang, A.; Han, Z. High-Frequency Sensitive Generative Adversarial Network for Low-Dose CT Image Denoising. *IEEE Access* **2019**, *8*, 930–943.
63. Lee, S.; Lee, M.S.; Kang, M.G. Poisson–Gaussian Noise Analysis and Estimation for Low-Dose X-ray Images in the NSCT Domain. *Sensors* **2018**, *18*, 1019.
64. Liu, H.; Jin, X.; Liu, L. Low-Dose CT Image Denoising Based on Improved DD-Net and Local Filtered Mechanism. *Comput. Intell. Neurosci.* **2022**, *2022*, 2692301.
65. Yu, X.; Wang, J.; Hong, Q.Q.; Teku, R.; Wang, S.H.; Zhang, Y.D. Transfer learning for medical images analyses: A survey. *Neurocomputing* **2022**, *489*, 230–254.
66. Huang, C.; Wang, J.; Wang, S.H.; Zhang, Y.D. Applicable artificial intelligence for brain disease: A survey. *Neurocomputing* **2022**, *504*, 223–239.
67. Tian, D.; Zhu, B.; Wang, J.; Kong, L.; Gao, B.; Wang, Y.; Xu, D.; Zhang, R.; Yao, Y. Brachial plexus nerve trunk recognition from ultrasound images: A comparative study of deep learning models. *IEEE Access* **2022**, *10*, 82003–82014.