Article

Not peer-reviewed version

A Novel Nash Equilibrium-Based Dynamic Connection Mechanism for Deep Neural Networks

Jincheng Zhang *

Posted Date: 14 July 2025

doi: 10.20944/preprints202507.1084.v1

Keywords: Nash equilibrium; dynamic skip connections; neural network fusion; adaptive feature integration; deep learning architecture

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 July 2025 doi:10.20944/preprints202507.1084.v1

Article

A Novel Nash Equilibrium-Based Dynamic Connection Mechanism for Deep Neural Networks

Jincheng Zhang

Faculty of Science and Technology, Rajabhat Maha Sarakham University, Maha Sarakham 44000, Thailand; zjc1639834588@gmail.com

Abstract

Skip Connections and Dense Connections are important structural designs widely used in deep learning models in recent years. They greatly alleviate the gradient vanishing problem in deep neural network training and improve the expressiveness and generalization performance of the model. This paper attempts to introduce the idea of Nash equilibrium theorem in game theory into this type of connection mechanism, and proposes a "Nash-Equilibrium Skip Connection". While keeping the structure simple, it establishes an "equilibrium state" information fusion method between multi-layer neuron outputs through an adaptive trade-off mechanism. Experimental results show that this method brings considerable performance improvement without increasing training time. This mechanism has good versatility and is not limited to the traditional multi-layer perceptron (MLP) model. It can be extended to various deep architectures such as CNN and Transformer.

Keywords: Nash equilibrium; dynamic skip connections; neural network fusion; adaptive feature integration; deep learning architecture

1. Introduction

With the development of deep learning, the number of neural network layers has continued to increase, improving the expressive power and complexity of the model [1–3]. However, the increase in the number of layers has also led to an increase in learning difficulty, especially problems such as gradient vanishing and information bottleneck [4–7]. To address these challenges, researchers have proposed structural designs such as skip connections and dense connections, which significantly improve the information flow efficiency and learning stability of deep networks by directly passing the output of the previous layer to the next layer [8–12]. These mechanisms have been widely used to accelerate network convergence and improve the performance of architectures such as ResNet and DenseNet, and have become a core component of modern neural network design [13–17].

Although these connection mechanisms have achieved widespread success in practice, their connection methods are usually predefined and static [18–22]. This design method does not consider the potential differences in information value, feature redundancy, and task relevance between different network layers [23–25]. In other words, the "skip information" received by the current layer often participates in subsequent calculations in the same proportion, lacking dynamic perception of context, task goals, and layer-level feature complementarity [26–29]. In complex input and task situations, the feature fusion method with fixed weights may not achieve the optimal information utilization, and may even introduce invalid information or interference information. Therefore, how to design a more intelligent, dynamic, and adaptive connection mechanism has become a topic worthy of in-depth discussion [30–32].

In another field of game theory, Nash equilibrium, as a basic concept [32–41], describes the stable state that multiple rational participants ultimately reach in the process of competing with each other to pursue their own optimal strategies. In this state, no individual will change its own strategy without considering the changes in the strategies of other individuals. This process highlights the relationship between "local optimality and system stability". Based on this idea, a new analogy can be established for the relationship between layers of neural networks. Each layer is regarded as an "information individual" that affects the final performance of the model through information competition and cooperation with other layers. Under this framework, different strategy weights can be assigned to the outputs of different layers, and these weights can be continuously adjusted during the learning process until a state close to Nash equilibrium is reached, making information fusion more reasonable and efficient.

Based on this, this paper proposes a dynamic connection mechanism inspired by the idea of Nash equilibrium. By introducing learnable weight parameters, this mechanism dynamically adjusts the information contribution from different network layers and continuously approaches a stable information fusion state during the model learning process. Compared with the traditional static skip connection method, this method not only has better representation ability, but also has better task adaptability and generalization ability. In addition, this mechanism has high versatility in architectural design. This method can be naturally extended not only to multi-layer perceptrons (MLPs), but also to various deep structures such as convolutional neural networks (CNNs) and graph neural networks (GNNs).

Through empirical research in image classification tasks, we found that this method not only surpasses traditional connection methods in performance indicators, but also achieves a more stable learning process and shorter learning time, showing high efficiency and practicality. More importantly, this method provides an interdisciplinary perspective for rethinking the design of network structures. Combined with the philosophical idea of "local optimality leads to global stability" in game theory, it is expected to trigger more innovative ideas in the design principles of neural networks.

2. The Migration Significance of Nash Equilibrium Ideas in Neural Network Connections

In traditional neural network structure design, information transmission between layers is mostly achieved in the form of linear superposition, sequential stacking or simple splicing. Although this method is clear in structure and easy to implement, its essence is a "passive acceptance" information transmission mechanism, that is, the latter layer unconditionally accepts all outputs from the previous layer or layers, lacking selectivity and adaptability. This static information flow method ignores the differences in semantics, importance and abstractness of features in each layer, and it is difficult to deal with information redundancy or interference problems under different task requirements.

In contrast, the Nash equilibrium idea in game theory emphasizes that in a system where multiple parties interact with each other, each participant tries to maximize his own interests while the strategies of others remain unchanged, and finally reaches a globally stable but non-uniform state. In this framework, individuals are neither completely cooperative nor completely confrontational, but reach a certain optimal compromise in the balance of interests. This mechanism theoretically guarantees the stability of the system and retains the autonomy of individuals. This has a certain degree of structural analogy with the information interaction between layers in a neural network: in the overall learning process of the model, each layer must not only express the feature information it has learned, but also make judgments and choices on the information from other layers in the fusion stage, thereby achieving "cooperative competition" at the information level.

Migrating this way of thinking to the design of neural network structures can give the model stronger adaptive capabilities. Specifically, we can regard the output of each layer as an information body or "participant". They do not passively participate in subsequent calculations through fixed weights, but actively participate in and influence the decision-making process of the next layer by learning dynamic coefficients during the training process. These coefficients reflect the importance of the outputs of different layers under specific tasks and sample conditions, similar to the "strategy selection" in the game, and finally form a dynamic equilibrium information fusion mode. This mechanism enables the model to flexibly adjust the fusion weights of the features of each layer under different input conditions, thereby improving the model's ability to represent complex samples.

The introduction of this idea breaks the "equal weight splicing" or "fixed ratio jump" connection method commonly used in previous models, and provides a new structural design perspective for neural networks. By introducing a learnable connection strategy in training, the model no longer relies on artificially set structural rules, but can autonomously optimize the information fusion path driven by a large number of samples. This mechanism significantly improves the flexibility and generalization ability of the model, enabling it to more effectively capture useful features and suppress invalid interference when facing multimodal input, strong hierarchy or redundant information scenes.

More importantly, this structural design based on game theory provides a new theoretical framework for understanding and optimizing deep neural networks. It emphasizes a balanced strategy rather than extreme optimization methods. This concept of "optimal collaboration under self-constraint" not only has clear mathematical theoretical support, but also shows obvious performance advantages in actual experiments. It upgrades the design of the connection structure from "static parameters" to "learnable strategies", which helps to build more intelligent, robust and explainable deep learning models.

3. Model Design and Method Description

In order to verify the effectiveness of the dynamic connection mechanism based on the Nash equilibrium idea proposed in this paper, we chose to conduct a simplified experiment under the framework of the classic and clearly structured multi-layer perceptron (MLP) model. The experimental design aims to deeply explore the impact of dynamic weight allocation on model performance and information transmission through a relatively simple but representative neural network structure, and lay a theoretical and experimental foundation for the subsequent application of more complex network structures.

Specifically, the constructed model consists of three main fully connected layers, each of which has different functions and information processing roles. The first layer is the input layer of the model, which is responsible for extracting primary features from the input data. These features are relatively basic and represent the low-level information expression of the input. The second layer follows closely, and further processes and abstracts the output of the first layer to capture higher-level feature information. This hierarchical design is in line with the typical idea of multi-layer perceptrons to extract features layer by layer. The design of the last layer is more special: it not only accepts the output of the second layer, but also introduces the output of the first layer at the same time, but the two are not simply spliced or added, but integrated through a dynamically learned weighted fusion mechanism to finally generate the classification prediction results of the model.

The key to this weighting mechanism is to introduce a learnable parameter to control the weight ratio of the output from the first layer and the second layer. This parameter is different from the traditional fixed connection method. It is constantly adjusted through the gradient update mechanism of back propagation during the model training process, and gradually converges to a stable and effective weight distribution scheme. At this time, a dynamic equilibrium state is formed inside the model, so that the feature outputs of different levels can fully exert their own advantages in information fusion, while avoiding redundancy and conflict, and achieve an ideal state similar to "all parties are the optimal strategy choice" in Nash equilibrium. This mechanism greatly improves the model's adaptability to inter-layer information fusion, allowing the network to dynamically adjust the contribution of features according to different task requirements and input data.

It is worth noting that the dynamic weight fusion method proposed in this paper is not limited to the multi-layer perceptron structure. The core idea behind it - that is, treating the outputs of different layers in the neural network as "interacting intelligent agents" and achieving the optimal fusion strategy through learning - has extremely strong universality and transferability. For example, in convolutional neural networks (CNNs), the feature maps extracted by different convolutional layers have large differences in spatial resolution and semantic levels. Traditional connection methods are mostly simple splicing or weighted summation, which often lack flexible adaptability. By introducing the dynamic weight mechanism in this paper, CNN can more intelligently adjust the fusion ratio of feature maps of each layer, thereby improving the ability to comprehensively utilize multi-scale information and improving the accuracy and robustness of feature expression.

In addition, this mechanism is also applicable to models based on attention mechanisms such as Transformer. In the Transformer structure, the attention output of each layer represents the model's capture of semantic information at different levels of the input sequence. By dynamically learning the fusion weights of the attention output of each layer, the model can more reasonably allocate attention to information at different levels, achieving more accurate context understanding and feature integration. This not only helps to improve the expressive power of the model, but also provides theoretical support for multi-layer information interaction in complex tasks.

Furthermore, this idea can also be extended to multi-model fusion systems, becoming a new idea for weighted combination of prediction results of different models. In ensemble learning or multimodal learning scenarios, multiple models participate in the final decision as different "intelligent agents". The optimal fusion of prediction results is achieved through dynamic weight adjustment, which can significantly improve the accuracy and stability of the overall system.

In summary, the dynamic connection mechanism based on the Nash equilibrium idea designed in this paper not only provides a novel multi-layer information fusion method, but also injects the theoretical wisdom of game theory into the structural design of deep learning models, showing broad application prospects and far-reaching research value.

4. Experimental Design and Result Analysis

We conducted comparative experiments on the CIFAR-10 image classification task. In order to control the influence of variables, all experiments were conducted under the same data set, the same model parameters, the same number of training rounds and optimizer configuration. The experimental models include:

Standard MLP model (without any skip connection)

Nash-Equilibrium Skip MLP model with Nash equilibrium mechanism

The complete python code used for the experiment is as follows:

```
import torch
import torch.nn as nn
import torch.optim as optim
import torchvision
import torchvision.transforms as transforms
from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score, confusion_matrix
import time
import numpy as np

# Use GPU if available
device = torch.device('cuda' if torch.cuda.is_available() else 'cpu')

# Limit training/testing to 5000 images
transform = transforms.Compose([
    transforms.ToTensor(),
    transforms.Normalize((0.5,), (0.5,))
])

# Load only 5000 train and 5000 test images for speed
```

```
full_trainset = torchvision.datasets.CIFAR10(root='./data', train=True, download=True, transform=transform)
trainset = torch.utils.data.Subset(full_trainset, range(5000))

full_testset = torchvision.datasets.CIFAR10(root='./data', train=False, download=True, transform=transform)
testset = torch.utils.data.Subset(full_testset, range(5000))

trainloader = torch.utils.data.DataLoader(trainset, batch_size=128, shuffle=True, num_workers=0)
testloader = torch.utils.data.DataLoader(testset, batch_size=128, shuffle=False, num_workers=0)

# Basic MLP
class MLP(nn.Module):
    def __init__(self):
        super(MLP, self).__init__()
        self.model = nn.Sequential(
            nn.Flatten(),
            nn.Linear(32 * 32 * 3, 512),
            nn.ReLU(),
            nn.Linear(512, 256),
            nn.ReLU(),
            nn.Linear(256, 10)
        )

    def forward(self, x):
        return self.model(x)

# Nash-inspired Skip Connection MLP
class NashMLP(nn.Module):
    def __init__(self):
        super(NashMLP, self).__init__()
        self.flatten = nn.Flatten()
        self.fc1 = nn.Linear(32 * 32 * 3, 512)
        self.relu = nn.ReLU()
        self.fc2 = nn.Linear(512, 256)
        self.alpha = nn.Parameter(torch.tensor(0.5))   # learnable alpha
        self.fc3 = nn.Linear(512 + 256, 10)

    def forward(self, x):
        x = self.flatten(x)
        out1 = self.relu(self.fc1(x))        # Layer 1 output
        out2 = self.relu(self.fc2(out1))   # Layer 2 output
        mix = torch.cat([(1 - self.alpha) * out1, self.alpha * out2], dim=1)
        return self.fc3(mix)

# Train & Evaluate Function
def train_and_evaluate(model, model_name, run_id):
    model = model.to(device)
    optimizer = optim.Adam(model.parameters(), lr=0.001)
    criterion = nn.CrossEntropyLoss()
    start_time = time.time()
    model.train()
    for epoch in range(5):
        for images, labels in trainloader:
            images, labels = images.to(device), labels.to(device)

            optimizer.zero_grad()
            outputs = model(images)
            loss = criterion(outputs, labels)
            loss.backward()
            optimizer.step()

    training_time = time.time() - start_time

    model.eval()
    all_preds, all_labels = [], []
    with torch.no_grad():
        for images, labels in testloader:
            images = images.to(device)
            outputs = model(images)
            _, predicted = torch.max(outputs, 1)
            all_preds.extend(predicted.cpu().numpy())
            all_labels.extend(labels.numpy())

    accuracy = accuracy_score(all_labels, all_preds)
    precision = precision_score(all_labels, all_preds, average='weighted', zero_division=0)
    recall = recall_score(all_labels, all_preds, average='weighted', zero_division=0)
    f1 = f1_score(all_labels, all_preds, average='weighted')
    conf_matrix = confusion_matrix(all_labels, all_preds)

    print(f"\nRun {run_id + 1} - Model: {model_name}")
    print(f"Training Time: {training_time:.2f} sec")
    print(f"Accuracy: {accuracy:.4f}")
    print(f"Precision: {precision:.4f}")
    print(f"Recall: {recall:.4f}")
    print(f"F1 Score: {f1:.4f}")

    return accuracy, precision, recall, f1, training_time

# Repeated Experiment Function
def run_multiple_times(model_class, model_name):
    accs, precs, recalls, f1s, times = [], [], [], [], []

    for i in range(10):
        model = model_class()
        acc, prec, rec, f1, t = train_and_evaluate(model, model_name, i)
        accs.append(acc)
        precs.append(prec)
        recalls.append(rec)
        f1s.append(f1)
```

```
            times.append(t)

        print(f"\n{'='*40}\nFinal Results for {model_name} (10 runs):")
        print(f"Accuracy        Mean: {np.mean(accs):.4f}   Std: {np.std(accs):.4f}")
        print(f"Precision       Mean: {np.mean(precs):.4f}   Std: {np.std(precs):.4f}")
        print(f"Recall          Mean: {np.mean(recalls):.4f}   Std: {np.std(recalls):.4f}")
        print(f"F1 Score        Mean: {np.mean(f1s):.4f}   Std: {np.std(f1s):.4f}")
        print(f"Training Time Mean: {np.mean(times):.2f}s   Std: {np.std(times):.2f}s")
        print(f"{'='*40}\n")

    # Run Both Models 10 Times
    run_multiple_times(MLP, "Standard MLP")
    run_multiple_times(NashMLP, "Nash-Equilibrium Skip MLP")
```

Each set of experiments was repeated 10 times to obtain robust statistical results. The results are as follows:

```
========================================
Final Results for Standard MLP (10 runs):
Accuracy        Mean: 0.4235   Std: 0.0098
Precision       Mean: 0.4298   Std: 0.0071
Recall          Mean: 0.4235   Std: 0.0098
F1 Score        Mean: 0.4151   Std: 0.0098
Training Time Mean: 12.69s   Std: 2.24s
========================================
========================================
Final Results for Nash-Equilibrium Skip MLP (10 runs):
Accuracy        Mean: 0.4281   Std: 0.0069
Precision       Mean: 0.4333   Std: 0.0058
Recall          Mean: 0.4281   Std: 0.0069
F1 Score        Mean: 0.4213   Std: 0.0063
Training Time Mean: 11.94s   Std: 1.26s
========================================
```

The average accuracy of the standard MLP model is 42.35%, the F1 score is 41.51%, and the training time is about 12.69 seconds;

The model with the Nash jump structure has an accuracy of 42.81%, an F1 score of 42.13%, and a slight decrease in training time.

This result shows that despite the very small changes, the model still achieves a stable improvement in performance through the "game-based" learning of information fusion methods, while maintaining or even slightly improving the training efficiency.

5. Universality and Scalability

Compared with traditional mechanisms such as jump connections and dense connections, the dynamic connection mechanism based on Nash equilibrium proposed in this paper shows excellent universality and wide applicability. Specifically, the universality of this connection mechanism can be elaborated in detail from the following three levels.

First, structural universality. The design of this mechanism is independent of the specific neural network architecture, and its core concept is to dynamically adjust the weights of multi-layer outputs as "participants". This enables it to be seamlessly applied to various mainstream network structures, such as traditional feedforward neural networks, convolutional neural networks (CNNs) with spatial feature extraction capabilities, graph neural networks (GNNs) for non-Euclidean data processing, and even the Transformer structure based on the attention mechanism that has been widely used in

doi:10.20944/preprints202507.1084.v1

recent years. This mechanism can improve the expressiveness and robustness of the model by adaptively adjusting the information flow from simple inter-layer connections to complex inter-layer interactions.

Second, task versatility. Although the experiments in this article are mainly focused on image classification tasks, the proposed dynamic weight adjustment idea goes far beyond this. Tasks such as text understanding, machine translation, and sentiment analysis in the field of natural language processing (NLP), speech processing scenarios such as speech recognition and acoustic modeling, and information fusion and weight distribution between different modal features in multimodal fusion tasks can all benefit from this mechanism. In addition, it can also be applied to fields such as time series prediction and reinforcement learning, reflecting the extremely high flexibility and universal value of this mechanism.

Third, the versatility of the system. The idea of this mechanism is not limited to the inter-layer connections within a single model, but also fits the higher-order system design concept. For example, in model ensemble learning, the outputs of different models can be regarded as "participants", and a better fusion strategy can be achieved through dynamic weight allocation. In modular neural network design, the collaborative work between independent modules can improve the synergy and adaptability of the entire system through this mechanism. In heterogeneous structure collaborative optimization, the dynamic fusion method based on Nash equilibrium can effectively alleviate the conflicts and contradictions between different structures and improve the overall performance of the system more efficiently.

In summary, this paper successfully introduces the concept of "equilibrium" into the design of deep learning structures, which not only significantly improves the performance and generalization ability of the model, but also provides a new theoretical perspective and technical path for future adaptive structure design. It is expected that the further improvement and promotion of this mechanism will play an important role in various research fields and application scenarios of artificial intelligence, and promote the development of more intelligent, flexible and efficient intelligent systems.

6. Conclusions and Future Work

This paper proposes an innovative jump connection mechanism that integrates the game concept of Nash equilibrium, aiming to overcome the problems of static weight distribution and insufficient adjustment of multi-layer output in traditional connection methods. Experimental verification shows that this mechanism shows obvious performance advantages in a simplified multi-layer perceptron structure, and the dynamic adjustment of connection weights can effectively promote the optimization of information flow and the improvement of model representation ability.

In future research, we will conduct more in-depth thinking from the following aspects:

First, we will verify the feasibility and performance of the multi-parameter, multi-party "game" connection mechanism in deeper and more complex neural network structures. Specifically, we regard more layers and more outputs as game parties, and achieve a more stable and effective equilibrium state in a wider inter-layer space through a more complex dynamic weight adjustment strategy.

Next, we plan to extend this dynamic connection mechanism to self-attention structures, especially the multi-head attention mechanism of the Transformer model. We regard different attention heads as game participants and study how to achieve collaborative competition and fusion between multiple attention heads. This will enhance the model's multi-faceted and multi-level understanding of the input sequence, and further enhance the effectiveness of tasks in fields such as natural language processing (NLP).

Third, we will explore the potential of this mechanism in the direction of model compression and knowledge distillation. In these scenarios, how to reasonably allocate information weights and adjust the knowledge transfer between the teacher model and the student model is the key to the dynamic weight adjustment mechanism to exert its advantages. It is expected that the introduction of the concept of Nash equilibrium will improve the performance and stability of the compression model and optimize the information flow management in the distillation process.

In summary, this paper introduces the concept of Nash equilibrium in game theory into the design of neural network structure, opening up a new research path. This not only deepens the theoretical basis of neural network design, but also provides solid empirical support and broad development space for the mutual integration of artificial intelligence with economics, game theory and other fields. It is hoped that this idea will inspire further innovation in the future and bring more intelligent, flexible and efficient structured solutions to deep learning and its wide range of applications.

References

1. Bhardwaj, K., Li, G., & Marculescu, R. (2021). How does topology influence gradient propagation and model performance of deep networks with densenet-type skip connections?. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 13498-13507).
2. Zhou, T., Ye, X., Lu, H., Zheng, X., Qiu, S., & Liu, Y. (2022). Dense convolutional network and its application in medical image analysis. BioMed Research International, 2022(1), 2384830.
3. Wang, H., Cao, P., Wang, J., & Zaiane, O. R. (2022, June). Uctransnet: rethinking the skip connections in u-net from a channel-wise perspective with transformer. In Proceedings of the AAAI conference on artificial intelligence (Vol. 36, No. 3, pp. 2441-2449).
4. Zhang, C., Benz, P., Argaw, D. M., Lee, S., Kim, J., Rameau, F., ... & Kweon, I. S. (2021). Resnet or densenet? introducing dense shortcuts to resnet. In Proceedings of the IEEE/CVF winter conference on applications of computer vision (pp. 3550-3559).
5. Oyedotun, O. K., Al Ismaeil, K., & Aouada, D. (2022). Why is everyone training very deep neural network with skip connections?. IEEE transactions on neural networks and learning systems, 34(9), 5961-5975.
6. Gite, S., Mishra, A., & Kotecha, K. (2023). Enhanced lung image segmentation using deep learning. Neural Computing and Applications, 35(31), 22839-22853.
7. Li, B., Xiao, C., Wang, L., Wang, Y., Lin, Z., Li, M., ... & Guo, Y. (2022). Dense nested attention network for infrared small target detection. IEEE Transactions on Image Processing, 32, 1745-1758.
8. Zhang, J., Zhang, Y., Jin, Y., Xu, J., & Xu, X. (2023). Mdu-net: Multi-scale densely connected u-net for biomedical image segmentation. Health Information Science and Systems, 11(1), 13.
9. Zhang, J., Zheng, B., Gao, A., Feng, X., Liang, D., & Long, X. (2021). A 3D densely connected convolution neural network with connection-wise attention mechanism for Alzheimer's disease classification. Magnetic Resonance Imaging, 78, 119-126.
10. Ranftl, R., Bochkovskiy, A., & Koltun, V. (2021). Vision transformers for dense prediction. In Proceedings of the IEEE/CVF international conference on computer vision (pp. 12179-12188).
11. Sitaula, C., & Shahi, T. B. (2022). Monkeypox virus detection using pre-trained deep learning-based approaches. Journal of Medical Systems, 46(11), 78.
12. Fu, Y., Wu, X. J., & Durrani, T. (2021). Image fusion based on generative adversarial network consistent with perception. Information Fusion, 72, 110-125.
13. Wang, R., Lei, T., Cui, R., Zhang, B., Meng, H., & Nandi, A. K. (2022). Medical image segmentation using deep learning: A survey. IET image processing, 16(5), 1243-1267.
14. Pandey, A., & Wang, D. (2021). Dense CNN with self-attention for time-domain speech enhancement. IEEE/ACM transactions on audio, speech, and language processing, 29, 1270-1279.
15. Alalwan, N., Abozeid, A., ElHabshy, A. A., & Alzahrani, A. (2021). Efficient 3D deep learning model for medical image semantic segmentation. Alexandria Engineering Journal, 60(1), 1231-1239.
16. Zhang, C., Cong, R., Lin, Q., Ma, L., Li, F., Zhao, Y., & Kwong, S. (2021, October). Cross-modality discrepant interaction network for RGB-D salient object detection. In Proceedings of the 29th ACM international conference on multimedia (pp. 2094-2102).
17. Lyu, X., Liu, L., Wang, M., Kong, X., Liu, L., Liu, Y., ... & Yuan, Y. (2021, May). Hr-depth: High resolution self-supervised monocular depth estimation. In Proceedings of the AAAI conference on artificial intelligence (Vol. 35, No. 3, pp. 2294-2301).

18. Heidari, M., Kazerouni, A., Soltany, M., Azad, R., Aghdam, E. K., Cohen-Adad, J., & Merhof, D. (2023). Hiformer: Hierarchical multi-scale representations using transformers for medical image segmentation. In Proceedings of the IEEE/CVF winter conference on applications of computer vision (pp. 6202-6212).
19. Njoku, J. N., Morocho-Cayamcela, M. E., & Lim, W. (2021). CGDNet: Efficient hybrid deep learning model for robust automatic modulation recognition. IEEE Networking Letters, 3(2), 47-51.
20. Zuo, Q., Chen, S., & Wang, Z. (2021). R2AU-Net: attention recurrent residual convolutional neural network for multimodal medical image segmentation. Security and Communication Networks, 2021(1), 6625688.
21. Hou, J., Zhang, Y., Zhong, Q., Xie, D., Pu, S., & Zhou, H. (2021). Divide-and-assemble: Learning block-wise memory for unsupervised anomaly detection. In Proceedings of the IEEE/CVF International Conference on Computer Vision (pp. 8791-8800).
22. Mirikharaji, Z., Abhishek, K., Bissoto, A., Barata, C., Avila, S., Valle, E., ... & Hamarneh, G. (2023). A survey on deep learning for skin lesion segmentation. Medical Image Analysis, 88, 102863.
23. Ashraf, A., Naz, S., Shirazi, S. H., Razzak, I., & Parsad, M. (2021). Deep transfer learning for alzheimer neurological disorder detection. Multimedia Tools and Applications, 1-26.
24. Chen, X., Wang, X., Zhang, K., Fung, K. M., Thai, T. C., Moore, K., ... & Qiu, Y. (2022). Recent advances and clinical applications of deep learning in medical image analysis. Medical image analysis, 79, 102444.
25. Abdollahi, A., & Pradhan, B. (2021). Integrating semantic edges and segmentation information for building extraction from aerial images using UNet. Machine Learning with Applications, 6, 100194.
26. Bao, F., Nie, S., Xue, K., Cao, Y., Li, C., Su, H., & Zhu, J. (2023). All are worth words: A vit backbone for diffusion models. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 22669-22679).
27. Li, Y., Wang, Z., Yin, L., Zhu, Z., Qi, G., & Liu, Y. (2023). X-net: a dual encoding–decoding method in medical image segmentation. The Visual Computer, 1-11.
28. Li, Q., Zhong, R., Du, X., & Du, Y. (2022). TransUNetCD: A hybrid transformer network for change detection in optical remote-sensing images. IEEE Transactions on Geoscience and Remote Sensing, 60, 1-19.
29. Chen, C., Chuah, J. H., Ali, R., & Wang, Y. (2021). Retinal vessel segmentation using deep learning: a review. IEEE Access, 9, 111985-112004.
30. Yin, X. X., Sun, L., Fu, Y., Lu, R., & Zhang, Y. (2022). [Retracted] U-Net-Based Medical Image Segmentation. Journal of healthcare engineering, 2022(1), 4189781.
31. Shiri, I., Arabi, H., Sanaat, A., Jenabi, E., Becker, M., & Zaidi, H. (2021). Fully automated gross tumor volume delineation from PET in head and neck cancer using deep learning algorithms. Clinical Nuclear Medicine, 46(11), 872-883.
32. Punn, N. S., & Agarwal, S. (2022). Modality specific U-Net variants for biomedical image segmentation: a survey. Artificial Intelligence Review, 55(7), 5845-5889.
33. Poveda, J. I., Krstić, M., & Başar, T. (2022). Fixed-time Nash equilibrium seeking in time-varying networks. IEEE Transactions on Automatic Control, 68(4), 1954-1969.
34. Bakhtyar, B., Qi, Z., Azam, M., & Rashid, S. (2023). Global declarations on electric vehicles, carbon life cycle and Nash equilibrium. Clean Technologies and Environmental Policy, 25(1), 21-34.
35. Hsieh, Y. G., Antonakopoulos, K., & Mertikopoulos, P. (2021, July). Adaptive learning in continuous games: Optimal regret bounds and convergence to Nash equilibrium. In Conference on Learning Theory (pp. 2388-2422). PMLR.
36. Ye, M., Li, D., Han, Q. L., & Ding, L. (2022). Distributed Nash equilibrium seeking for general networked games with bounded disturbances. IEEE/CAA Journal of Automatica Sinica, 10(2), 376-387.
37. Qian, Y. Y., Liu, M., Wan, Y., Lewis, F. L., & Davoudi, A. (2021). Distributed adaptive Nash equilibrium solution for differential graphical games. IEEE Transactions on Cybernetics, 53(4), 2275-2287.
38. Ye, M., Han, Q. L., Ding, L., & Xu, S. (2023). Distributed Nash equilibrium seeking in games with partial decision information: A survey. Proceedings of the IEEE, 111(2), 140-157.
39. Ye, M., Yin, J., & Yin, L. (2021). Distributed Nash equilibrium seeking for games in second-order systems without velocity measurement. IEEE Transactions on Automatic Control, 67(11), 6195-6202.

40. Nian, X., Niu, F., & Yang, Z. (2021). Distributed Nash equilibrium seeking for multicluster game under switching communication topologies. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 52(7), 4105-4116.
41. Zhang, K., Fang, X., Wang, D., Lv, Y., & Yu, X. (2021). Distributed Nash equilibrium seeking under event-triggered mechanism. IEEE Transactions on Circuits and Systems II: Express Briefs, 68(11), 3441-3445.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.