


Article

Study on Image Classification Algorithm Based on Multi-Scale Feature Fusion and Domain Adaptation

Yu Guo ^{1,†}, Ziyi Cheng ^{2,†}, Yuanlong Zhang ³, Gaoxuan Wang ^{4,*} and Jundong Zhang ¹¹ Marine Engineering College, Dalian Maritime University, Dalian 116000, China² Department of Electrical and Electronic Engineering, The University of Manchester, Manchester M13 9PL, UK³ School of Mathematical Sciences, University of Nottingham, Nottingham NG9 2SE, UK⁴ National Engineering Research Center of Geographic Information System, China University of Geosciences, Wuhan 430078, China

* Correspondence: gxwang@cug.edu.cn

† These authors contributed equally to this work.

Abstract: This paper introduces the MMTADAN, an innovative algorithm designed to enhance cross-domain image classification. By integrating multi-scale feature extraction with Taylor series-based detail enhancement and adversarial domain adaptation, the MMTADAN effectively aligns features between the source and target domains. The proposed approach addresses the critical challenge of generalizing classification models across diverse datasets, demonstrating significant improvements in performance. The findings suggest that retaining essential image details through multi-scale extraction and Taylor series enhancement can lead to better classification outcomes, making the MMTADAN a valuable contribution to the field of image classification.

Keywords: statistical learning; deep learning; domain adaptation; generative adversarial network



Citation: Guo, Y.; Cheng, Z.; Zhang, Y.; Wang, G.; Zhang, J. Study on Image Classification Algorithm Based on Multi-Scale Feature Fusion and Domain Adaptation. *Appl. Sci.* **2024**, *14*, 10531. <https://doi.org/10.3390/app142210531>

Academic Editors: Pedro Couto and Antonio Fernández-Caballero

Received: 15 September 2024

Revised: 31 October 2024

Accepted: 14 November 2024

Published: 15 November 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Machine learning, as a discipline that equips machines with “intelligence”, has become deeply integrated into various sectors of our lives and industries [1–4]. Cheng et al. [5] present a novel self-supervised adversarial training method for Monocular Depth Estimation (MDE) models, enhancing robustness against physical attacks by leveraging view synthesis and incorporating ℓ_0 -norm-bounded perturbations, without the need for ground-truth depth. Fang et al. [6] introduce a smooth and time-optimal S-curve trajectory planning method for robotic manipulators, using a piecewise sigmoid function to create infinitely differentiable trajectories, balancing efficiency and smoothness under given constraints, with validation through simulations and experiments. Liu et al. [7] propose an Approximate Maximum Likelihood Estimator (AMLE) to efficiently estimate the instantaneous frequency (IF) for nonstationary signals with intersecting or closely spaced IFs, improving upon the traditional MLE by reducing the computational complexity and handling time-varying amplitudes. Fang et al. [8] present a methodology for generating online, smooth joint trajectories for robots using an improved sinusoidal jerk model, offering a computationally efficient solution for time-optimal trajectory planning with kinematic constraints, leading to a better performance in terms of efficiency and jerk suppression compared to the existing techniques. Qin et al. [9] present a method based on the synchroextracting chirplet transform (SECT) for early chatter recognition in robotic drilling, demonstrating a superior performance in detecting chatter onset more quickly and effectively than the existing methods, making it practical for real-time vibration suppression. Sun et al. [10] propose a variable-scale wavelet packet entropy (VSWPE) method for detecting machining chatter, including scenarios with the beat effect, using beat frequency estimation and optimal demodulation techniques to improve the detection accuracy and prevent false alarms, demonstrating effectiveness in both simulations and machining tests.

In real-world applications, machine learning has seen considerable success in areas such as computer vision, natural language processing, and recommendation systems [11–16]. Computer vision, in particular, is a significant research area that simulates human visual systems to understand the real world through computational tools. From early efforts focused on manually designing image features to current research using deep learning algorithms for automated feature extraction, image analysis has remained a key component of artificial intelligence research. The hope is to use AI technologies to address challenges in the field of computer vision. In contrast to supervised learning with labeled data, today's information-rich world produces vast amounts of unlabeled image data across industries. These unlabeled datasets, though easy to obtain and rich in information, pose challenges in training deep learning models. This has created an urgent need to resolve the issue of how to effectively use small amounts of labeled data while incorporating large volumes of unlabeled data into deep learning models. Moreover, transferring a deep learning model trained on one dataset to another related but different dataset is a pressing challenge in the current deep learning development. One promising research direction addressing these challenges is domain adaptation. Traditional machine learning and deep learning tasks typically assume that the training data and deployment data share the same distribution, but this assumption often does not hold in practice. Thus, domain adaptation has become a critical research area.

Sohail et al. [17] provide a comprehensive review of deep transfer learning (DTL) and domain adaptation (DA) techniques for 3D point cloud (3DPC) processing, highlighting recent advancements, datasets, evaluation metrics, and applications like object detection, segmentation, and denoising while addressing current challenges and suggesting future research directions. Özince et al. [18] introduce a suite of sparsity-aware complex-valued least-mean kurtosis (CLMK) algorithms, including l0-CLMK, l0-ACLМК, ZA-CLMK, ZA-ACLМК, RZA-CLMK, and RZA-ACLМК, aimed at improving sparse system identification. Simulation results demonstrate superior performances in the convergence rate, tracking, and steady-state error compared to the existing sparsity-aware algorithms in both synthetic and real-world scenarios. Lu et al. [19] propose the generalized Jensen–Rényi divergence (GJRD) as a method for efficiently handling multiple data distributions in machine learning scenarios, addressing the limitations of traditional pairwise divergence measures. A non-parametric empirical estimator based on kernel density estimation is derived for the GJRD, which is then integrated into a deep clustering framework (GJRD-DC). Experimental results show that the GJRD-DC achieves state-of-the-art performances on challenging datasets, and the code is available online. Peng et al. [20] propose novel regularization-based frequency-domain diffusion algorithms to address the limitations of the existing methods for distributed estimation with missing inputs, offering an enhanced performance through bias elimination, power normalization, and fast convergence under colored inputs, as well as providing stability analysis and effective power estimation techniques. Xian et al. [21] propose a novel approach for unsupervised person re-identification by leveraging multiple-source datasets, using expert-specific clustering and dual-similarity distillation to enhance the domain adaptation while maintaining the feature diversity through representation decorrelation, with experiments validating its effectiveness on key benchmarks. Fang et al. [22] present a source-free collaborative domain adaptation (SCDA) framework for resting-state fMRI, addressing cross-site data heterogeneity by utilizing a pretrained source model and unlabeled target data, with a multi-perspective feature enrichment method and unsupervised pretraining, demonstrating effectiveness in cross-scanner and cross-study prediction.

In the field of image processing, domain adaptation has seen significant advancements. Peng et al. [23] propose a Disentanglement-Inspired Single-Source Domain Generalization Network (DSDGnet) for cross-scene hyperspectral image (HSI) classification, addressing spectral heterogeneity by extracting domain-invariant representations through a style transfer and progressive disentanglement approach, with experiments demonstrating a superior performance over the existing methods across multiple HSI datasets.

Yang et al. [24] introduce a novel unsupervised domain adaptation method, the contrastive domain adaptation network (CDANet), for building extractions from high-spatial-resolution imagery, leveraging adversarial and contrastive learning with a multitask generator and dual discriminators to enhance the edge detection and alignment of cross-domain pixel features, achieving a superior performance compared to the existing methods across multiple datasets. Tang et al. [25] propose a domain-adaptive noise reduction framework (DANRF) for low-dose CT denoising, combining supervised and unsupervised methods through iterative knowledge transfer, knowledge distillation, and style generalization learning to address domain gaps and improve performances on real-world data, with experiments demonstrating its effectiveness on multi-source datasets. Jecklin et al. [26] present a novel approach to overcome the domain gap between synthetic and real fluoroscopic images for the intraoperative 3D reconstruction of the spine in orthopedic surgeries, utilizing a unique paired dataset and transfer learning. By integrating style transfer and refining the X23D model, the method achieves high accuracy and a good real-time performance, offering a promising solution for enhancing surgical navigation and planning. Guo et al. [27] propose novel regularization-based frequency-domain diffusion algorithms for distributed estimation with missing input data, improving upon the existing methods by addressing the input color and complexity issues through bias elimination, power normalization, and periodic updates, with simulations demonstrating the algorithms' superior performance and theoretical validity. Moraes et al. [28] review the critical role of training data in the land cover (LC) classification of satellite imagery, identifying key research topics across 114 peer-reviewed studies. The findings are categorized into four main topics: construction of the training dataset, sample quality, sampling design, and advanced learning techniques. Subtopics include methods for the sample collection, cleaning, size, class balance, and distribution. The review provides a comprehensive synthesis of these aspects, offering insights to guide future LC mapping projects. Chen et al. [29] propose the Multi-Scale Global and Category-Attention Feature Alignment Network (MGCAN) to improve the extraction of un-collapsed buildings from post-disaster high-resolution remote sensing images by effectively aligning pre- and post-disaster features, significantly enhancing the accuracy and outperforming the existing methods. Okafor et al. [30] enhanced wheat head detection across varying domains by applying Fourier domain adaptation (FDA), adaptive alpha beta gamma correction (AABG), and random guided filter (RGF) preprocessing, achieving an improved detection accuracy (a mAP of 0.6534) compared to the baseline and addressing challenges related to domain variations in wheat images.

In the realm of image retrieval, traditional image feature descriptors often result in high-dimensional vectors. The Scale-Invariant Feature Transform (SIFT) [31], for instance, generates local feature descriptors with dimensions typically ranging from 12,800 to 128,000 per image. Even deep learning-based retrieval models often produce feature vectors exceeding 1024 dimensions. In the era of big data, this high dimensionality significantly increases storage requirements and computational costs, making it difficult to meet real-time image retrieval demands. To address these challenges, we propose a novel algorithm, the multi-scale, multi-channel Taylor adversarial domain adaptation network (MMTADAN), which integrates statistical methods and deep learning to solve the problem of domain adaptation in image processing. Our approach employs a multi-scale, multi-channel feature extraction network, detail enhancement using Taylor series-based analysis, and adversarial domain adaptation to align features across domains.

2. Related Background Knowledge

2.1. Taylor Series

If a function $f(x_0)$ has an n -th derivative at point x_0 , then there exists a neighborhood of x_0 , and for any x within this neighborhood, the function can be expressed as follows:

$$f(x) = f(x_0) + f'(x_0)(x - x_0) + \frac{f''(x_0)}{2}(x - x_0)^2 + \dots + \frac{f^{(n)}(x_0)}{n!}(x - x_0)^n + o((x - x_0)^n) \quad (1)$$

Extending the Taylor formula into an infinite series yields the Taylor series:

$$f(x) = \sum_{n=0}^{\infty} \frac{f^{(n)}(x_0)}{n!} (x - x_0)^n = f(x_0) + f'(x_0)(x - x_0) + \frac{f''(x_0)}{2!} (x - x_0)^2 + \dots + \frac{f^{(n)}(x_0)}{n!} (x - x_0)^n + \dots \quad (2)$$

This expansion provides a way to approximate smooth functions by polynomials around a specific point.

2.2. Generative Adversarial Networks (GANs)

Generative Adversarial Networks (GANs) consist of two primary components: a generator and a discriminator, as shown in Figure 1. The generator takes random noise sampled from a specific distribution and attempts to generate data that resemble the true data distribution, whereas the discriminator receives both real data samples and the generator’s output, attempting to distinguish between the real and generated data. The generator and discriminator iteratively improve by competing against each other in a process known as adversarial learning. This iterative process continues until the generator produces data that are indistinguishable from the real data and the discriminator cannot reliably differentiate between the two, achieving a Nash equilibrium.

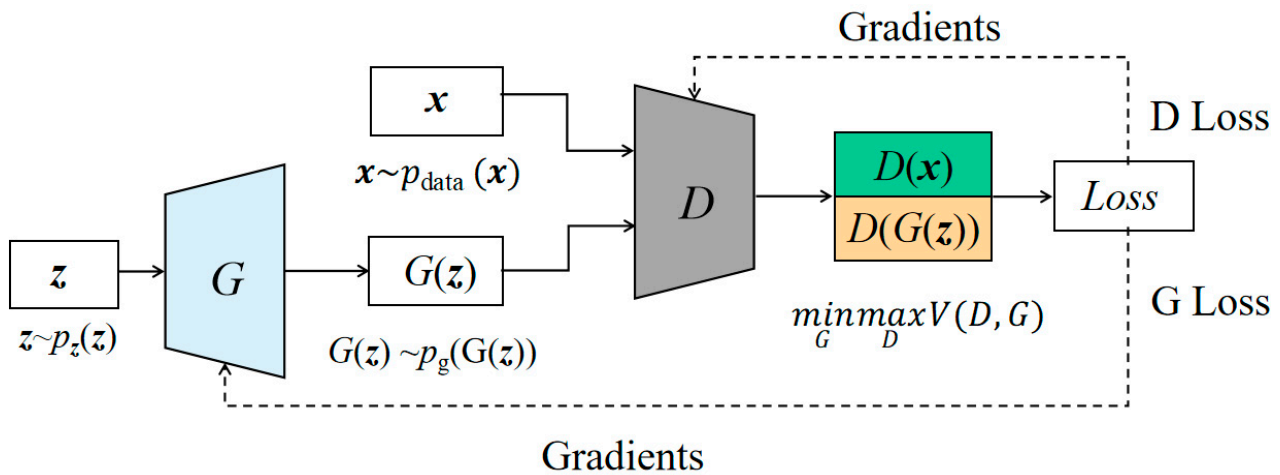


Figure 1. Structure of a Generative Adversarial Network.

Let P_{data} represent the true data distribution and P_z represent the noise distribution, which can follow any arbitrary distribution. The adversarial optimization process for GANs can be expressed as follows:

$$\min_G \max_D E_{x \sim P_{data}} \log[D(x)] + E_{z \sim P_z} \log[1 - D(G(z))] \quad (3)$$

Figure 1 illustrates the structure of a GAN, where the generator and discriminator are engaged in a minimax game, refining their performances through adversarial training.

2.3. Attention Mechanism

Convolutional Neural Networks (CNNs) possess strong data fitting capabilities; however, due to limitations imposed by optimization algorithms and computational resources, enhancing the CNN performance on large-scale datasets can be challenging. Computational power often becomes a bottleneck that restricts model improvements. Inspired by the human nervous system, the brain cannot handle overloaded visual tasks, but through the visual attention mechanism, humans focus on key objects, significantly improving their processing efficiency [32]. Attention mechanisms have a long history of research, and in 2014, Google and the DeepMind team successfully integrated attention mechanisms with Recurrent Neural Networks (RNNs), giving rise to Attention-based RNNs (AttRNNs).

These networks are widely used in sequence-to-sequence tasks such as machine translation and speech recognition. Common types of attention mechanisms include channel attention, spatial attention, and hybrid attention.

Channel attention (CA), also referred to as item-wise attention, focuses on assigning different weights to each feature map obtained from various convolutional kernels. The idea is that each kernel extracts a different feature map, capturing varying amounts of important information. By applying a weight coefficient to each feature map, the network can emphasize the feature maps that contain more relevant information.

Spatial attention (SA), also known as location-wise attention, takes the data features as input and learns a weight mask through a neural network. This mask predicts the importance of different parts of the feature map, enhancing or suppressing specific features depending on their relevance.

Hybrid attention (HA) combines both spatial and channel attention mechanisms to enhance the model's ability to focus on the most important parts of the input. A classic example of a hybrid attention module is the Convolutional Block Attention Module (CBAM) [33], which incorporates both the Channel Attention Module (CAM) and Spatial Attention Module (SAM). The CAM computes the attention weights for each channel, allowing the network to prioritize specific channel features, while the SAM calculates attention weights for each spatial location, focusing the network on important regions in the image. The outputs of the CAM and SAM are combined through element-wise multiplication to generate the final attention weights.

The CBAM enhances the network's sensitivity to both channel and spatial features, improving the classification accuracy. Furthermore, the CBAM can be seamlessly integrated into existing CNN architectures as a submodule, improving the model's expressive power and robustness. This module has been widely applied in tasks such as object detection, image classification, and semantic segmentation, where it has consistently demonstrated an excellent performance.

3. Method

3.1. Feature Fusion Network

In computer vision tasks such as object detection, image classification, and semantic segmentation, fusing multi-scale features significantly improves the performance of CNNs. Shallow feature maps, obtained with fewer convolutional operations, capture texture and geometric features of the input image but contain more redundant and low-level semantic information. In contrast, deeper feature maps, produced with more convolutions, offer better semantic representations but can lose critical details. When the performance of a network reaches a certain bottleneck, integrating multi-scale features can provide richer information, enhancing the model's robustness.

Christian et al. [34] proposed the Inception network, which combines multi-scale convolution and feature concatenation. The key idea is to use convolutional kernels of different sizes within the same layer to capture multi-scale features, followed by concatenating these features to achieve more comprehensive feature representations. Shang et al. [35] introduced the Res2Net architecture, a variant of ResNet, which splits the input features into smaller sub-feature maps processed through separate paths and then combines the outputs. This approach enables the extraction of more diverse and detailed features.

Image detail enhancement algorithms can be applied to a wide range of electronic products. In this subsection, we derive and analyze the initial model formulation for detail enhancement, proposing a rapid detail enhancement algorithm based on Taylor series analysis. This subsection introduces the Residual Extraction Network based on Taylor series analysis, referred to as the T-NET. Additionally, the T-NET employs an information entropy-based algorithm to integrate ten layers of residual features. Furthermore, we integrate the T-NET with the super-resolution algorithm CARN-M [36] to achieve both the super-resolution and detail enhancement of images. While implementing the engineering aspects, this subsection also theoretically analyzes the algorithm's rationale, error bounds,

time complexity, convergence, and information redundancy, contributing certain theoretical innovations to the field.

Given a signal $f(x)$, according to detail enhancement theory, the signal can be decomposed into a base layer $b(x)$ and a detail layer $d(x)$, where $f(x) = b(x) + d(x)$. The existing algorithms aim to optimize the approximation of the $b(x)$ based on various priors and models. However, due to the complexity of image textures, obtaining a precise $b(x)$ is challenging, and what is usually obtained is an estimation, denoted as $b'(x)$. Therefore, the enhanced signal can be modeled as follows:

$$e(x) = b'(x) + \alpha \times (f(x) - b'(x)) \tag{4}$$

However, Equation (4) can be rewritten as shown in Equation (4) as $t = \alpha - 1$. For a given input image $f(x)$, a network can be used to generate the detail layer signal $f(x)$, where $d'(x) = \text{Net}(f(x))$ is an estimation of the true image detail layer. Additionally, the signal $f(x)$ is a weighted sum of the signals from multiple layers, represented as follows:

$$f(x) = \sum_{k=1}^n (w_k \times f_k(x)),$$

where w_k and $f_k(x)$ are the weights of each layer, and k is the output of the k -th layer of the network. Thus, the task of detail enhancement is reduced to designing the w_k and $f_k(x)$. Equation (5) further clarifies this problem, and Figure 2 illustrates the architecture of the image detail enhancement system based on Taylor series analysis (T-NET):

$$\begin{aligned} e(x) &= f(x) + (\alpha - 1) \times (f(x) - b'(x)) \\ &= f(x) + t \times d'(x) = f(x) + t \times \text{Net}(f(x)) \\ &= f(x) + t \times \sum_{k=1}^n w_k \times f_k(x) \end{aligned} \tag{5}$$

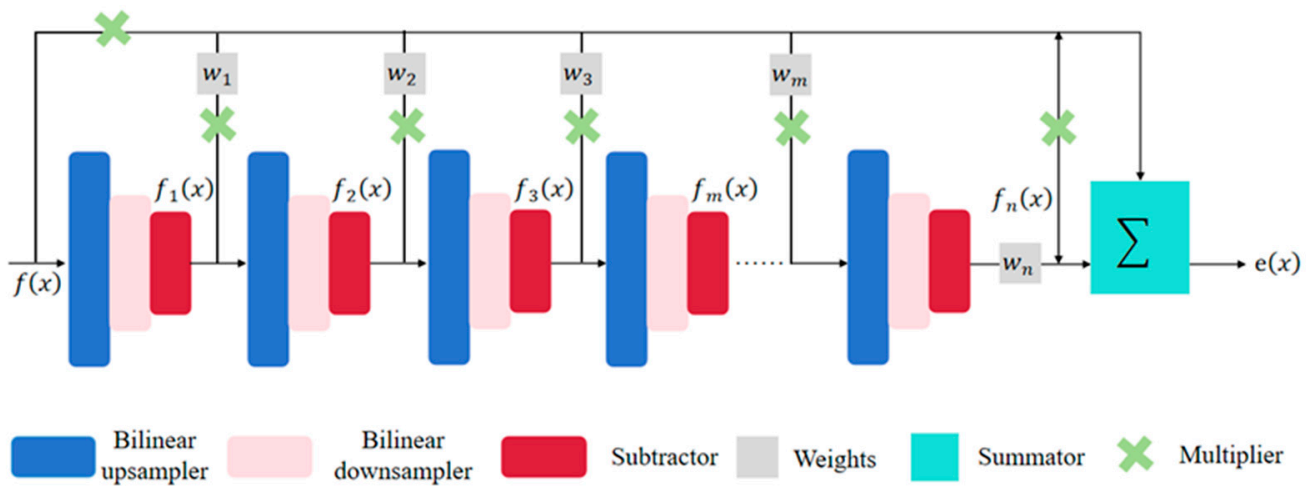


Figure 2. Image detail enhancement system based on Taylor series analysis (T-NET).

The feature extractor consists of multi-scale and multi-head attention modules, as depicted in Figure 3.

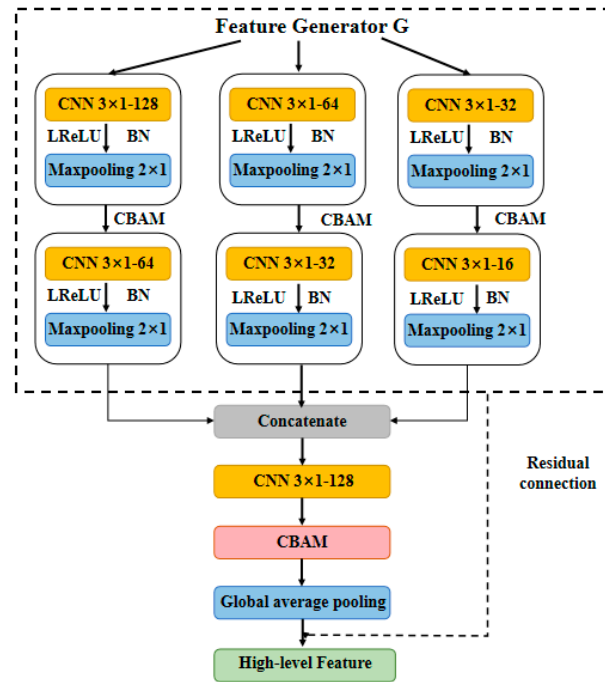


Figure 3. Schematic of the multi-head, multi-scale feature extractor.

The overall process of the MMTADAN algorithm is illustrated in Figure 4.

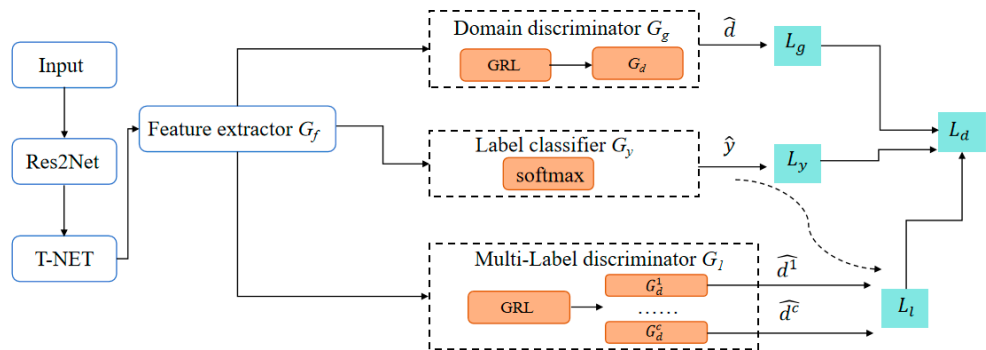


Figure 4. MMTADAN algorithm.

Specifically, the objective of the T-NET architecture is to develop a rapidly converging network that can generate detail layers of images in near real time. As previously described, it is crucial to have a clear understanding of what the signal $f(x)$ represents, as well as the mechanisms underlying the generation of the weights. Additionally, in accordance with the requirements of the detail enhancement algorithm, it is necessary to protect the signal $f(x)$ and avoid excessive variations in the $\nabla f(x)$. Thus, a network with edge-preserving capabilities must be designed.

3.1.1. Taylor Unit

The Taylor unit serves as the fundamental structure of the Taylor network (T-NET). As illustrated in Figure 5, it consists of three components: a bilinear upsampler (U), a bilinear downsampler (D), and a subtractor. In Figure 5, these components are represented by the blue, pink, and red rectangular blocks, respectively. Based on the transmission characteristics of the signal, we can derive that $f_n(x) = f_{n-1}(x) - D(U(f_{n-1}(x)))$. However, bilinear interpolation is a relatively coarse interpolation method, which allows the signal $f_n(x)$ to be treated as the residual of signal $f_{n-1}(x)$.

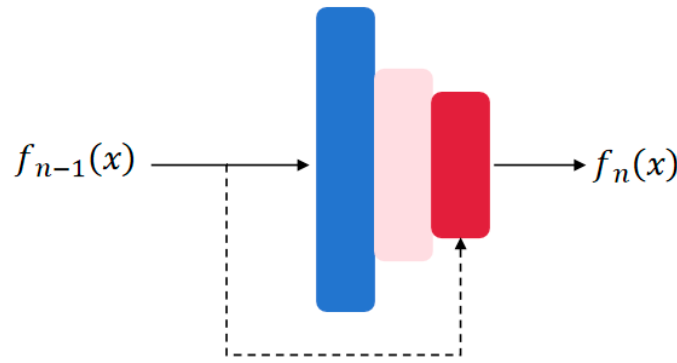


Figure 5. Schematic of the multi-head, multi-scale feature extractor.

It is evident that the residual image and the gradient image appear very similar, leading to the modeling of the signal $f_n(x)$ as $p \times \nabla f_{n-1}(x) + \gamma$. This assumption is reasonable since bilinear interpolation is a linear process, indicating that the $f_n(x)$ and signal $f_{n-1}(x)$ are linearly related. Consequently, $f_n(x) = a \times f_{n-1}(x) + b$, where a and b are known constants.

In this subsection, $x \approx x_0 \Rightarrow x - x_0$ is very small; thus, if $f_{n-1}(x_0) + \nabla f_{n-1}(x_0)(x - x_0)$ is used to estimate Equation (6), the signal $f_n(x)$ can be represented as $\rho \times f_{n-1}(x) + \gamma$:

$$\begin{aligned} f_{n-1}(x) \Rightarrow f_n(x) &\approx a(f_{n-1}(x_0) + \nabla f_{n-1}(x_0)(x - x_0)) + b \\ &\approx a\lambda \nabla f_{n-1}(x) + af_{n-1}(x_0) + b \\ &= p \times \nabla f_{n-1}(x) + \gamma \end{aligned} \tag{6}$$

In the above formula, $0 < \rho < 1$, and γ varies primarily between 50 and 150. By sampling m training pairs from the database, the following formula can be used to estimate the parameters p_k and γ_k for the k -th layer, resulting in a simple least-squares model. The existing optimization packages can be easily utilized to obtain solutions for this structure.

3.1.2. Weighted Residual Regression

Inspired by block matrix processing algorithms, weighted residuals are learned for each layer of the residual network $f_k(x)$. Initially, each weighted layer is decomposed into non-overlapping image patches of a 4×4 size. It is evident that different weights should be assigned among these patches; if a residual image patch contains more detailed information, it should be assigned a relatively larger coefficient, and vice versa. Based on this criterion, information entropy is adopted as a measure of the richness of the details within the image patches, as it is a metric related to the uncertainty of random variables. Here, if the intensity values of the pixels in an image patch can be considered as random variables, they can indeed be measured by information entropy. In this subsection, a higher information entropy indicates a richer texture within the patch, implying that this patch should receive a greater weighting coefficient. The calculation of the information entropy is as follows: $E_i = \sum_{\mu} \rho(x_{\mu}) \log_2 \frac{1}{\rho(x_{\mu})} = -\sum_{\mu} \rho(x_{\mu}) \log_2 \rho(x_{\mu})$ for a 4×4 residual image patch, where $\rho(x_{\mu})$ represents the probability value of pixel x having intensity (μ) in patch i .

The average value of these entropies is used to reflect the detail information of the group of image patches. Ultimately, these entropy values are normalized to serve as the final coefficient: $w_{ki} = \frac{1}{k!} \times w_i = \frac{1}{k!} \times \frac{E_i}{\sum_{i=1}^m E_i}$.

It can be readily proven that $\sum_{i=1}^m w'_{ki} = \frac{1}{k!}$ and $\sum_{k=1}^{10} \sum_{i=1}^m w'_{ki} \approx e - 1 \approx 2$ hold true for the m groups of patches in the 10-layer Taylor network (T-NET). To summarize the descriptions in the preceding sections, the following formula is presented, where τ denotes the coefficient for detail enhancement, existing to accommodate various types of images.

For clarity, this includes a manually adjustable parameter. Thus, the overall steps of the residual regression are presented in the following equation:

$$e(x) = f(x) + \sum_{k=1}^{10} w_k \times f_k(x) = f(x) + \tau \times \sum_{k=1}^{10} \sum_{i=1}^m \frac{1}{k!} \times \frac{E_i}{\sum_{i=1}^m E_i} \times f_k(x_i) \quad (7)$$

3.2. Algorithm Design

3.2.1. Label Classifier

The label classifier (G_y) is a core component of the model, responsible for performing the main classification task. Once the model is trained, only the feature extractor (G_f) and label classifier (G_y) are used for predicting the image classification. The label classifier consists of a fully connected neural network that takes the feature (f) corresponding to the input data (x) and outputs the prediction of x 's category, represented as a C -dimensional vector in Figure 4. Since the label classifier is a supervised classification model, it is trained using labeled data from the source domain. The loss function used for the label classifier is the cross-entropy loss (L_y), which is represented as follows:

$$L_y = -\frac{1}{n_s} \sum_{x_i \in D_s} \sum_{c=1}^C P_{x_i \in c} \log(\hat{y}) \quad (8)$$

$\hat{y} = G_y(G_f(x_i))$, where $P_{x_i \in c}$ is the probability that x_i belongs to class c . The label classifier serves two functions during the training process: (1) it performs supervised learning using labeled source domain data and target domain data with reliable pseudo-labels (represented by the solid line in Figure 4 for the calculation of L_y); (2) it generates pseudo-labels for the target domain data to guide the multi-class domain discriminator in aligning conditional distributions (represented by the dashed line in Figure 4 for the calculation of L_l).

3.2.2. Domain Discriminator

The domain discriminator is responsible for aligning the marginal distributions of the source and target domains in the proposed algorithm. The idea is inspired by the Domain Adversarial Neural Network (DANN) [37] and consists of two main components: a gradient reversal layer (GRL), shown in Figure 4, and a fully connected classifier that distinguishes whether the feature (f) comes from the source or target domain. After the data (x) are processed by the feature extractor, the resulting feature (f) first passes through the GRL, after which the fully connected classifier determines whether the data are from the source or target domain. The result is represented by \hat{d} in Figure 4. The domain discriminator is a supervised learning classifier, since it only needs to classify data into two categories (the source or target domain). Both source domain data and target domain data are used in the training of the domain discriminator. The corresponding loss function is the cross-entropy loss (L_g), represented as follows:

$$L_g = \frac{1}{n_s + n_t} \sum_{x_i \in D_s \cup D_t} L_d(\hat{d}, d_i) \quad (9)$$

$\hat{d} = G_d(R_\lambda(G_f(x_i)))$, where n_s and n_t are the numbers of source and target domain samples, respectively. The GRL acts as an identity function during forward propagation but reverses the gradient during backpropagation, as expressed in Equations (10) and (11), where I is the identity matrix. The role of the GRL is to propagate the gradients in the opposite direction during backpropagation, ensuring that the feature extractor is updated in a direction that maximizes the adversarial training effect:

$$R_\lambda(x) = x \quad (10)$$

$$\frac{dR_\lambda}{dx} = -\lambda I \tag{11}$$

3.2.3. Multi-Class Domain Discriminator

The multi-class domain discriminator (G_l) is responsible for aligning the conditional distributions between the source and target domains. It consists of C domain discriminators (where C is the number of data categories), each with a structure similar to that of the domain discriminator, including a GRL and a fully connected domain classifier. Each domain classifier is responsible for aligning the features of one specific category, ensuring the alignment of the conditional distributions between the source and target domains. The loss function for the multi-class domain discriminator (G_l) is as follows:

$$L_l = \frac{1}{n_s + n_t} \sum_{c=1}^C \sum_{x_m \in \mathcal{D}_s \cup \mathcal{D}_t} \hat{y}_m^c L_d^c(\hat{d}_c, d_c) \tag{12}$$

$\hat{d}_k = G_d^c(R(G_f(x_m)))$ and n_s and n_t are the numbers of datasets in the source domain and the target domain, respectively. If x_m is from the target domain data, it is the \hat{y}_m^c item of the C -dimensional vector output by the label classifier. If x_m comes from the source domain data, it is the \hat{y}_m^c item of the label corresponding to the data x_m .

In this design, we drew inspiration from multi-adversarial domain adaptation (MADA) [38], wherein different domain discriminators are used for different categories to align the conditional distributions. However, our implementation differs from MADA in several ways. First, instead of using separate GRL and fully connected classifiers for each of the C domain discriminators, which can lead to a significant increase in the network parameters when the number of classes is large, we use a single fully connected network with C output nodes. Each node represents the domain classification for a specific category. Additionally, MADA requires the availability of target domain labels, but in our case, target domain labels are not available and are instead part of the model’s learning objective. In MADA, pseudo-labels generated during training are used to guide the domain discriminator, but these pseudo-labels can be inaccurate, leading to error amplification. In our approach, we impose constraints on the pseudo-label acquisition. Before performing conditional distribution alignment, we first align the marginal distributions, ensuring the accuracy of the pseudo-labels. Furthermore, we introduce a confidence threshold (δ), where only target domain data with predicted class confidence values higher than the δ are included in the conditional distribution alignment.

3.3. Cost Function and Algorithm Optimization Process

In summary, the cost function of this model is as follows:

$$L(\theta_f, \theta_y, \theta_g, \theta_l) = L_y(\theta_f, \theta_y) + L_g(\theta_f, \theta_g) + L_l(\theta_f, \theta_l) \tag{13}$$

The parameters to be optimized include θ_f for the feature extractor (G_f), θ_y for the label classifier (G_y), θ_g for the domain discriminator (G_g), and θ_l for the multi-class domain discriminator (G_l). The goal of the model’s training is to optimize these parameters. The training process is divided into two stages. In the first stage, the objective is to align the marginal distributions between the source and target domains, requiring the optimization of θ_y , θ_y , and θ_g , using the loss function $L_1(\theta_f, \theta_y, \theta_g) = L_y(\theta_f, \theta_y) + L_g(\theta_f, \theta_g)$. This stage’s optimization is similar to the training process in the DANN. Once convergence is achieved, the feature distributions between the source and target domains become similar, and the predictions for the target domain data reach a certain level of accuracy. At this point, the prediction results from the label classifier for the target domain data can be treated as reliable pseudo-labels for calculating the loss.

The model then begins the second stage of training, with the goal of aligning the conditional distributions between the source and target domain data. In this

stage, the parameters θ_f , θ_y , θ_g , and θ_l need to be optimized, using the loss function $L_2(\theta_f, \theta_y, \theta_g, \theta_l) = L_y(\theta_f, \theta_y) + L_g(\theta_f, \theta_g) + L_l(\theta_f, \theta_l)$. This training phase involves two main components: first, the alignment of the conditional distributions is achieved through the local discriminator (G_1); second, a dynamic dataset is maintained, which consists of target domain samples with high confidence levels. These high-confidence samples are directly used as supervised data for training the label classifier.

4. Experiments

All experiments involving the methods proposed in this chapter were conducted in a single-card environment using an Nvidia RTX 3080 Ti (Santa Clara, CA, USA). The development environment was set to Windows, with programming carried out using the PyTorch (V1.5.0) framework. The Stochastic Gradient Descent (SGD) optimization algorithm was employed, with a momentum coefficient of 0.9. The total number of training iterations was set to 10, with each iteration comprising 50 epochs. After each iteration, the optimal model was saved.

4.1. Dataset Description

4.1.1. Medical Image Datasets

For the image classification task, two commonly used lung field segmentation datasets are the JSRT Lung Nodule Dataset [39] and the Montgomery Tuberculosis Dataset [40,41]. The JSRT Lung Nodule Dataset contains 247 chest X-ray images with a resolution of 2048×2048 and a 12-bit grayscale, collected by the Japanese Society of Radiological Technology. Among them, 154 images contain lung nodules, and 93 do not. The Montgomery Tuberculosis Dataset, collected by the Department of Health and Human Services in Montgomery County, MD, USA, was initially developed for the creation of a computer-aided diagnosis system for tuberculosis. This dataset contains 138 chest X-ray images with a resolution of either 4020×4892 or 4892×4020 and a 12-bit grayscale, with 80 normal chest images and 58 tuberculosis images. Both datasets use the same annotation standards for the lung field masks.

4.1.2. Office–Home Image Dataset

Office–Home is a standard dataset for domain adaptation. This dataset comprises 15,500 images categorized into 65 different classes, with four subsets: (i) Art (Ar), which includes images of paintings, sketches, and artistic representations; (ii) Product (Pr), consisting of images captured without background; (iii) Real-World (Rw), containing conventional images taken with cameras; and (iv) Clipart (Cl), a collection of clipart images. The performances of the methods were evaluated on four randomly selected migration tasks out of the possible twelve: $Ar \rightarrow Pr$; $Pr \rightarrow Rw$; $Rw \rightarrow Cl$; and $Cl \rightarrow Ar$, where the symbols before and after the arrow indicate the source and target domains, respectively.

4.2. Comparison Methods and Evaluation Metrics

Several comparison methods were employed in the experiments:

- (1) Transformer [42] (Baseline): the Transformer model, trained on source domain data, was directly applied to target domain data without any domain adaptation;
- (2) DANN [37]: The DANN is a domain adaptation method that trains a neural network to learn features that are both discriminative for the source domain task and invariant to domain differences. This is achieved by using a gradient reversal layer to promote domain invariance, allowing the model to generalize to the target domain without labeled data;
- (3) LPJT [43]: LPJT is a domain adaptation method that jointly optimizes feature and sample adaptation while preserving the local sample consistency. It utilizes label propagation to predict new instances and supports both homogeneous and heterogeneous domain transfers;

- (4) CDAN [44]: The CDAN is a domain adaptation method that enhances adversarial learning by conditioning the adaptation on classifier predictions. It employs multi-linear conditioning to capture the cross-covariance between features and predictions, and entropy conditioning to manage uncertainty, achieving state-of-the-art results on multiple datasets.

The evaluation metrics used in the experiments included the accuracy, precision, recall, and F1 score. The accuracy measures the proportion of correct predictions out of the total number of cases. The precision (positive predictive value) measures the proportion of true-positive predictions among all positive predictions. The recall (sensitivity) measures the proportion of true-positive predictions among all actual positives. The precision and recall are often conflicting metrics, and the F1 score balances both, giving them equal weight. The accuracy and F1 score are the most important metrics. The formulas for the evaluation metrics are as follows:

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (14)$$

$$precision = \frac{TP}{TP + FP} \quad (15)$$

$$recall = \frac{TP}{TP + FN} \quad (16)$$

$$F1score = \frac{2 \times precision \times recall}{precision + recall} \quad (17)$$

where TP, TN, FP and FN represent the numbers of true positives, true negatives, false positives, and false negatives, respectively. In this experiment, the normal class was treated as the positive class.

4.3. Experimental Results Analysis

4.3.1. Experimental Results

Two experimental tasks were established based on the medicine datasets: JSRT → Montgomery and Montgomery → JSRT. The results are summarized in Table 1. Additionally, four experimental tasks were set up using the Office–Home dataset, with the results detailed in Table 2.

Table 1. Experimental results of various methods for the medicine datasets.

Task	Method	Accuracy	Precision	Recall	F1
JSRT → Montgomery	Transformer	38.12%	45.36%	10.45%	0.16
	DANN	83.24%	85.67%	35.78%	0.49
	LPJT	79.34%	60.21%	30.65%	0.40
	CDAN	65.78%	80.45%	60.12%	0.69
	MMTADAN	92.57%	89.32%	85.76%	0.83
Montgomery → JSRT	Transformer	30.47%	55.36%	12.15%	0.19
	DANN	75.84%	82.37%	40.56%	0.55
	LPJT	78.46%	65.29%	35.12%	0.46
	CDAN	70.23%	78.41%	55.48%	0.65
	MMTADAN	91.37%	85.78%	88.12%	0.87

In the transfer tasks from JSRT to Montgomery and vice versa, the MMTADAN demonstrated an exceptional performance. Specifically, for the JSRT → Montgomery task, the MMTADAN achieved an accuracy of 92.57%, significantly surpassing the DANN (83.24%) and LPJT (79.34%). The MMTADAN also attained precision and recall rates of 89.32% and 85.76%, respectively, highlighting its superior capability in correctly identifying pulmonary nodules. The F1 score reached 0.83, reflecting a well-balanced performance in the classification task.

Table 2. Experimental results of various methods for the Office–Home dataset.

Task	Method	Accuracy	Precision	Recall	F1
Ar → Pr	Transformer	60.34%	65.48%	50.12%	0.57
	DANN	70.43%	75.32%	55.78%	0.64
	LPJT	68.29%	70.41%	52.16%	0.60
	CDAN	75.84%	78.23%	65.74%	0.71
	MMTADAN	85.67%	88.14%	80.47%	0.84
Pr → Rw	Transformer	58.23%	60.32%	48.47%	0.54
	DANN	67.56%	70.48%	55.12%	0.62
	LPJT	66.45%	68.78%	54.11%	0.61
	CDAN	72.89%	75.32%	63.56%	0.69
	MMTADAN	83.42%	85.78%	78.64%	0.82
Rw → Cl	Transformer	55.47%	57.64%	45.32%	0.51
	DANN	65.36%	68.12%	52.73%	0.60
	LPJT	63.42%	65.87%	50.63%	0.58
	CDAN	70.45%	73.34%	60.76%	0.66
	MMTADAN	82.57%	85.12%	75.47%	0.80
Cl → Ar	Transformer	60.23%	63.41%	50.76%	0.57
	DANN	68.14%	70.23%	55.12%	0.62
	LPJT	67.35%	69.42%	52.67%	0.60
	CDAN	74.42%	76.34%	64.23%	0.69
	MMTADAN	86.24%	88.42%	80.32%	0.85

For the Montgomery → JSRT task, the MMTADAN again exhibited outstanding results, with an accuracy of 91.37%, well above the other methods. Notably, in the precision (85.78%) and recall (88.12%), the MMTADAN showcased its strong adaptability in the target domain, achieving an F1 score of 0.87, which confirms its advantage in handling imbalanced datasets.

In the various transfer tasks of the Office–Home dataset, the MMTADAN consistently maintained its lead:

Ar → Pr: the MMTADAN achieved an accuracy of 85.67%, outperforming all other methods. Its precision and recall were 88.14% and 80.47%, respectively, demonstrating adaptability in diverse image processing tasks;

Pr → Rw: the MMTADAN also displayed a robust performance, with an accuracy of 83.42%, significantly higher than those of the DANN and CDAN;

Rw → Cl: the MMTADAN reached an accuracy of 82.57%, excelling in all metrics, indicating its robustness in cross-domain transfer tasks;

Cl → Ar: with an accuracy of 86.24% and a precision of 88.42%, the MMTADAN reaffirmed its exceptional capability in complex image classification tasks.

Overall, the MMTADAN consistently exhibited superior accuracy, precision, recall, and F1 scores across all the experiments. These results not only validate its effectiveness in medical image classification and domain adaptation tasks but also highlight its adaptability and flexibility when confronted with diverse datasets and tasks. In summary, the MMTADAN’s outstanding performance across multiple transfer tasks, particularly in the accuracy and F1 scores, underscores its potential as an effective domain adaptation method. A comparative analysis with other methods distinctly illustrates the MMTADAN’s exceptional performance in handling complex datasets, further supporting its practical application value.

4.3.2. Ablation Experiments

To further elucidate the impacts of the pseudo-label classifier and feature fusion modules within the MMTADAN on the image classification performance, we compared the algorithm without the pseudo-label classifier (NLC) and the algorithm without the feature fusion module (NFF).

Using the Office–Home dataset, four transfer tasks were set up, with the results are displayed in Table 3.

Table 3. Ablation experimental results of various methods with the Office–Home dataset.

Task	Method	Accuracy	Precision	Recall	F1
Ar → Pr	NLC	75.43%	77.34%	66.23%	0.71
	NFF	67.54%	70.23%	55.32%	0.62
	MMTADAN	85.67%	88.14%	80.47%	0.84
Pr → Rw	NLC	69.23%	72.34%	58.43%	0.65
	NFF	62.45%	64.78%	52.34%	0.58
	MMTADAN	83.42%	85.78%	78.64%	0.82
Rw → Cl	NLC	72.47%	75.34%	65.21%	0.70
	NFF	65.14%	67.34%	53.24%	0.60
	MMTADAN	82.57%	85.12%	75.47%	0.80
Cl → Ar	NLC	58.34%	60.12%	47.32%	0.54
	NFF	76.87%	78.65%	68.34%	0.73
	MMTADAN	86.24%	88.42%	80.32%	0.85

The results clearly indicate that both the pseudo-label classifier and the feature fusion module play significant roles in enhancing the model performance.

Impact of the Pseudo-Label Classifier (NLC vs. MMTADAN)

The model's performance was markedly inferior in all tasks without the pseudo-label classifier (NLC) compared to the full MMTADAN:

Ar → Pr: The NLC achieved an accuracy of 75.43%, which is 10 percentage points lower than the MMTADAN's 85.67%. The F1 score also decreased by 0.13 (NLC: 0.71; MMTADAN: 0.84), underscoring the importance of the pseudo-label classifier in improving the classification performance;

Pr → Rw: The accuracy for the NLC was 69.23%, representing a decline of 14 percentage points compared to the MMTADAN's 83.42%. The recall difference was particularly pronounced (NLC: 58.43%; MMTADAN: 78.64%), indicating that the pseudo-label classifier aids in capturing more target domain samples, thereby enhancing the model robustness;

Rw → Cl: the NLC recorded an F1 score of 0.70, significantly lower than the MMTADAN's 0.80, further validating the critical role of the pseudo-label classifier in cross-domain adaptation;

Cl → Ar: The NLC performed the worst in this task, with an accuracy of only 58.34% and an F1 score of 0.54, both substantially below the MMTADAN's 86.24% and 0.85. This indicates that the pseudo-label classifier substantially enhances the model performance when handling images from different domains.

Impact of the Feature Fusion Module (NFF vs. MMTADAN)

The model's performance also significantly declined without the feature fusion module (NFF):

Ar → Pr: The NFF achieved an accuracy of just 67.54%, an 18-percentage-point drop from the MMTADAN's 85.67%. The F1 score similarly fell from 0.84 to 0.62, highlighting the critical importance of multi-scale feature fusion in enhancing the classification accuracy and robustness;

Pr → Rw: The NFF's accuracy was 62.45%, a 21-percentage-point reduction compared to the MMTADAN's 83.42%. This suggests that models without feature fusion struggle to effectively capture multi-scale features from target domain samples, leading to overall performance degradation;

Rw → Cl: the NFF's recall rate was 53.24%, markedly lower than the MMTADAN's 75.47%, further affirming the importance of the feature fusion module in cross-domain tasks;

Cl → Ar: Although the NFF performed slightly better in this task, with an accuracy of 76.87%, it still fell short of the MMTADAN's 86.24%. This indicates that the feature fusion module generally provides advantages in handling diverse image features.

The contribution of the pseudo-label classifier is evident: the NLC's performance was significantly inferior to that of the MMTADAN, demonstrating that the pseudo-label classifier effectively enhances the model classification performance and generalization capability in scenarios with limited target domain data labels. The feature fusion module

also showed a notable performance drop in the NFF model compared to the complete MMTADAN, indicating that multi-scale feature fusion captures richer and more nuanced image features, thereby improving the classification performance.

In addition, to demonstrate the performance of the T-NET, three objective metrics were used to evaluate each algorithm: the Structural Similarity Index (SSIM), the Signal-Preserving Ability (SPA), and the Edge-Preserving Ability (EPA). The formula for the SSIM is defined as follows: $\frac{(2\mu_x\mu_y+c_1)(2\sigma_{xy}+c_2)}{(\mu_x^2+\mu_y^2+c_1)(\sigma_x^2+\sigma_y^2+c_2)}$, where μ_x and μ_y are the means of x and y , σ_x and σ_y are the variances of x and y , and σ_{xy} is the covariance between x and y . c_1 and c_2 are very small constants, both set to 0.001. Specifically, the SPA is defined as $\sum_{i=1}^k |f(x_i) - f'(x_i)|$, and the EPA is defined as $\sum_{i=1}^{k'} |\nabla(x_i) - \nabla f'(x_i)|$. In these two formulas, k and k' represent the lengths of the one-dimensional signals $f(x)$ and $\nabla f(x)$, respectively, while $f'(x)$ denotes the result of filtering the signal $f(x)$, which is the output of various detail enhancement algorithms applied to the signal $f(x)$. The comparison methods include GIF [45], WLS [46], GGIF [47], SPGIF [48], ILS [49], RGIF [47], and ZF [50]. From Table 4, it can be seen that the T-NET performed excellently in the SSIM, SPA, and EPA tests, demonstrating its superior performance.

Table 4. Comparative performance metrics of classification methods across different datasets.

Method	J5RT SSIM/SPA/EPA	Montgomery SSIM/SPA/EPA	Office-Home SSIM/SPA/EPA
GIF	0.876/3191/51	0.832/2171/107	0.814/3560/231
WLS	0.830/3659/91	0.837/3659/499	0.892/1239/763
GGIF	0.894/2673/32	0.889/3894/175	0.990/1284/46
SPGIF	0.876/6531/139	0.735/7922/646	0.961/2867/109
ILS	0.892/4150/41	0.754/5154/423	0.841/4294/23
RGIF	0.840/2579/118	0.881/5271/718	0.847/4571/225
ZF	0.792/3550/67	0.716/4891/675	0.841/3441/19
T-NET	0.985/673/15	0.993/865/30	0.983/764/16

4.3.3. Visualization

To conduct a more detailed analysis of the method’s accuracy, confusion matrices were computed. Figure 6 illustrates the experimental results of the MMTADAN and the comparative methods for the J5RT dataset.

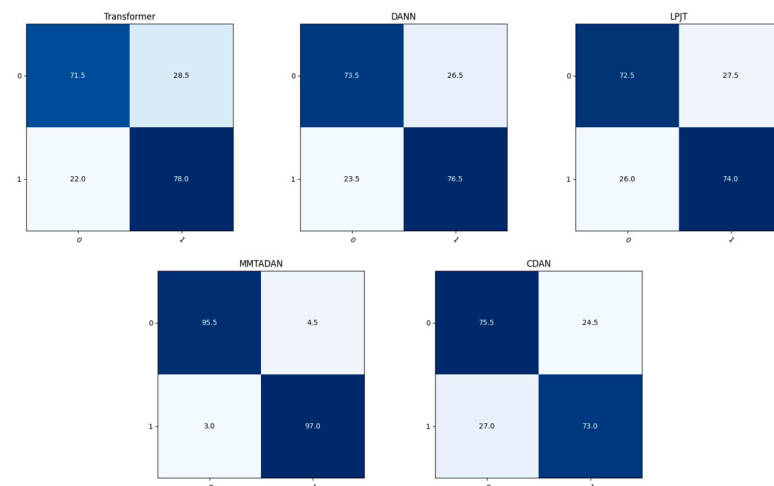


Figure 6. Experimental results of confusion matrices of various methods for the medicine dataset.

The confusion matrices in Figure 6 indicate that the Transformer method performs the worst. While the DANN, LPJT, and CDAN have improved the accuracy of health

status recognition, their overall performances remain average. In contrast, the MMTADAN demonstrates effective classification.

The Receiver Operating Characteristic (ROC) curve evaluates the classifier performance by plotting the True-Positive Rate (TPR) against the False-Positive Rate (FPR). The precision–recall (PR) curve assesses the model performance by graphing the precision against the recall. The TPR and FPR are calculated using true positives (TPs), false positives (FPs), true negatives (TNs), and false negatives (FNs) for the ROC curve, while the precision and recall are computed for the PR curve. Figure 7 presents the ROC curves for different models for the Montgomery dataset. In the ROC curves, the closer a curve is to the upper left corner, the better its performance. The ROC curve corresponding to the MMTADAN is the nearest to the upper left corner, confirming its superior effectiveness.

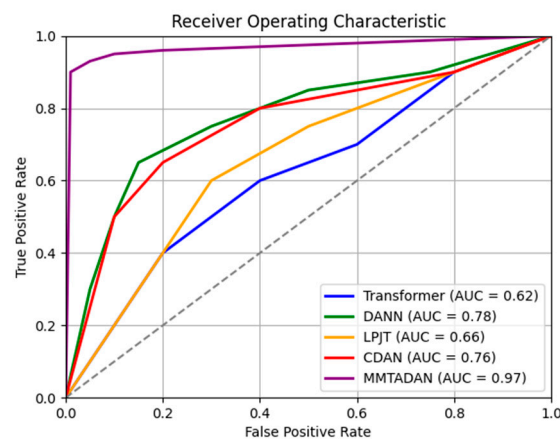


Figure 7. ROC curves of different models for the Montgomery dataset.

To validate the classification performances of the methods under consideration, a series of Monte Carlo experiments were conducted. For each experiment, the samples were randomly divided into three groups: training, validation, and testing. Care was taken to ensure that the numbers of samples for each type were approximately equal across all three groups. The training segments were used to train individual classifiers, while the validation segments were employed to train an alpha ensemble using the scores of the trained classifiers. Finally, the test segments were utilized to determine the performance. Two performance metrics were computed: the Balanced Accuracy (BAC) and F1 score (F1), which were estimated as the mean accuracy for each class and the harmonic mean of the precision and recall, respectively. The final results represent the averages of 100 Monte Carlo experiments. The experimental results are presented in Table 5.

Table 5. Average classification performances for different datasets.

Method	JSRT BAC (%) / F1 Score (%)	Montgomery BAC (%) / F1 Score (%)	Office–Home BAC (%) / F1 Score (%)
Transformer	71.05 ± 2.94 / 69.23 ± 2.78	71.23 ± 2.88 / 69.50 ± 3.12	70.89 ± 2.56 / 68.89 ± 2.65
DANN	80.01 ± 2.75 / 78.50 ± 3.01	80.50 ± 3.05 / 79.03 ± 2.96	79.84 ± 2.69 / 78.21 ± 2.72
LPJT	79.96 ± 3.10 / 77.08 ± 2.89	80.14 ± 3.20 / 77.64 ± 2.75	80.48 ± 2.93 / 77.46 ± 3.07
CDAN	80.23 ± 2.57 / 79.04 ± 2.64	80.32 ± 2.48 / 79.27 ± 2.59	80.05 ± 2.71 / 78.45 ± 2.99
MMTADAN	96.00 ± 2.55 / 95.01 ± 2.82	96.50 ± 2.62 / 95.25 ± 2.98	96.25 ± 2.53 / 94.93 ± 2.84

As shown in Table 5, the Transformer performed poorly on the Office–Home dataset (65% BAC), indicating that this method struggles with more complex or diverse datasets. While the DANN showed a good performance on the JSRT and Montgomery, its performance dropped to 75% on the Office–Home dataset. This may suggest that the DANN is sensitive to certain types of data or that there were significant differences in the feature distribution between the training and Office–Home datasets. Similarly, the LPJT’s

result for the Office–Home dataset (76.5% BAC) is lower than those for the JSRT and Montgomery datasets, highlighting its limitations in specific contexts. The CDAN also exhibited a decrease in its performance on the Office–Home dataset (77% BAC), possibly due to the complexity of the data, affecting the model’s generalization capability. Despite the MMTADAN’s excellent performance on the JSRT and Montgomery datasets (96% and 96.5% BACs, respectively), its BAC for the Office–Home dataset is 90%. This result indicates that the MMTADAN maintained a relatively stable performance across the different datasets, although it was somewhat affected on the Office–Home method, and it remains the best-performing method among all of them.

Furthermore, Table 6 displays the p -value calculations and 95% confidence intervals for the different methods for the Office–Home dataset. Statistical analysis reveals that the MMTADAN showed a significant advantage in the classification performance, with its BAC and F1 scores markedly higher than those of the other methods. Through rigorous statistical analysis, we confirmed the reliability of the experimental results and provide valuable references for subsequent research.

Table 6. Statistical performance comparison of different methods for the Office–Home dataset.

Method	p -Value (BAC)	p -Value (F1)	95% CI for BAC (%)	95% CI for F1 (%)
Transformer	<0.001	<0.001	(68.00, 73.78)	(66.00, 71.78)
DANN	<0.001	<0.001	(78.00, 81.68)	(77.00, 79.98)
LPJT	<0.001	<0.001	(79.00, 81.96)	(75.00, 79.92)
CDAN	<0.001	<0.001	(79.00, 81.10)	(77.00, 80.90)
MMTADAN	-	-	(96.25, 96.95)	(94.93, 95.73)

Using Big O notation, we estimated the time complexity of each method. Subsequently, after training each method, we recorded the time required for the model inference in detail. These data aided us in comparing the performances of the different models to evaluate their feasibility and advantages in practical applications. The time complexities were as follows: Transformer: $O(n \cdot d^2)$ (where n is the sequence length and d is the feature dimension); DANN: $O(n \cdot d \cdot m)$ (where n is the number of samples and m is the number of features); LPJT, CDAN, and MMTADAN: the complexity was analyzed based on specific implementations. The results are shown in Table 7.

Table 7. Time complexity and performance metrics of different methods.

Method	Time Complexity	Training Time (Seconds)	Inference Time (Seconds)
Transformer	$O(n \cdot d^2)$	150	20
DANN	$O(n \cdot d \cdot m)$	120	18
LPJT	$O(n \cdot d \cdot m)$	110	15
CDAN	$O(n \cdot d \cdot m)$	115	17
MMTADAN	$O(n \cdot d \cdot m)$	90	10

As shown in Table 7, the MMTADAN had the shortest training time at only 90 s, indicating high efficiency during the model training. In contrast, the Transformer took the longest time to train, at 150 s, likely due to its complex architecture and computational overhead from the self-attention mechanism. The MMTADAN also excelled in the inference time, requiring only 10 s, making it more advantageous for real-time applications. Other methods have inference times ranging from 15 to 20 s, indicating lower efficiencies during inference. The MMTADAN outperformed other methods in both the training and inference times, showcasing its exceptional computational efficiency and performance. Therefore, the MMTADAN is regarded as the best-performing method in this experiment.

5. Conclusions

This paper presents a novel approach for enhancing cross-domain image classification through the proposed algorithm, the MMTADAN. This method combines multi-scale feature extraction with Taylor series-based detail enhancement and adversarial domain adaptation to better align features between source and target domains. It addresses a pertinent issue in image classification, particularly the challenges associated with generalizing models across diverse datasets. The integration of multi-scale feature extraction and Taylor series enhancement is promising for improving the classification performance by preserving essential image details.

Author Contributions: Conceptualization, G.W.; Methodology, Y.G.; Software, Y.Z.; Validation, Y.G., Y.Z. and J.Z.; Formal analysis, Z.C.; Investigation, J.Z.; Resources, Z.C.; Data curation, Z.C. and J.Z.; Writing—original draft, Y.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Major Scientific Research Instrument Development Project (62127806) and High-Technology Ship Research Program (CBG3N21-3-3).

Data Availability Statement: The raw data supporting the conclusions of this article will be made available by the authors on request.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Ding, L.; Sun, Y.; Xiong, Z. Dual-Mode Type Algorithm for Chatter Detection in Turning Considering Beat Vibration. In Proceedings of the 2019 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM), Hong Kong, China, 8–12 July 2019; pp. 654–659.
2. Fang, Y.; Hu, J.; Shao, Q.; Qi, J. Fifth Order Trajectory Planning for Reducing Residual Vibration. In Proceedings of the 2019 IEEE 4th International Conference on Advanced Robotics and Mechatronics (ICARM), Toyonaka, Japan, 3–5 July 2019; pp. 999–1004.
3. Gao, T.; Yang, J.; Tang, Q. A Multi-Source Domain Information Fusion Network for Rotating Machinery Fault Diagnosis under Variable Operating Conditions. *Inf. Fusion* **2024**, *106*, 102278. [[CrossRef](#)]
4. Gao, T.; Yang, J.; Wang, W.; Fan, X. A Domain Feature Decoupling Network for Rotating Machinery Fault Diagnosis under Unseen Operating Conditions. *Reliab. Eng. Syst. Saf.* **2024**, *252*, 110449. [[CrossRef](#)]
5. Cheng, Z.; Han, C.; Liang, J.; Wang, Q.; Zhang, X.; Liu, D. Self-Supervised Adversarial Training of Monocular Depth Estimation against Physical-World Attacks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2024**, *46*, 1–17. [[CrossRef](#)] [[PubMed](#)]
6. Fang, Y.; Hu, J.; Liu, W.; Shao, Q.; Qi, J.; Peng, Y. Smooth and Time-Optimal S-Curve Trajectory Planning for Automated Robots and Machines. *Mech. Mach. Theory* **2019**, *137*, 127–153. [[CrossRef](#)]
7. Liu, Y.; Sun, Y.; Xiong, Z. An Approximate Maximum Likelihood Estimator for Instantaneous Frequency Estimation of Multicomponent Nonstationary Signals. *IEEE Trans. Instrum. Meas.* **2022**, *71*, 6501509. [[CrossRef](#)]
8. Fang, Y.; Qi, J.; Hu, J.; Wang, W.; Peng, Y. An Approach for Jerk-Continuous Trajectory Generation of Robotic Manipulators with Kinematical Constraints. *Mech. Mach. Theory* **2020**, *153*, 103957. [[CrossRef](#)]
9. Qin, C.; Sun, Y.; Tao, J.; Zeng, H.; Li, Y.; Liu, C. A Chatter Recognition Approach for Robotic Drilling System Based on Synchroextracting Chirplet Transform. *IEEE Sens. J.* **2023**, *23*, 27670–27683. [[CrossRef](#)]
10. Sun, Y.; Liu, C.; Sun, L.; Xiong, Z.; Zhu, X. Chatter Detection With Beat Effect Based on Beat Frequency Estimation. *IEEE Trans. Autom. Sci. Eng.* **2021**, *18*, 2166–2174. [[CrossRef](#)]
11. Salazar, A.; Vergara, L.; Safont, G. Generative Adversarial Networks and Markov Random Fields for Oversampling Very Small Training Sets. *Expert Syst. Appl.* **2021**, *163*, 113819. [[CrossRef](#)]
12. Pereira, L.M.; Salazar, A.; Vergara, L. A Comparative Analysis of Early and Late Fusion for the Multimodal Two-Class Problem. *IEEE Access* **2023**, *11*, 84283–84300. [[CrossRef](#)]
13. Salazar, A.; Safont, G.; Vergara, L. A New Application of Ultrasound Signal Processing for Archaeological Ceramic Classification. In Proceedings of the ICASSP 2020—2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, 4–8 May 2020; pp. 3082–3086.
14. Salazar, A.; Safont, G.; Vergara, L.; Vidal, E. Graph Regularization Methods in Soft Detector Fusion. *IEEE Access* **2023**, *11*, 144747–144759. [[CrossRef](#)]
15. Salazar, A.; Pereira, L.M.; Vergara, L. Experimental Study on Decision Fusion Parameters Using Alpha Integration. In Proceedings of the 2022 International Conference on Computational Science and Computational Intelligence (CSCI), Las Vegas, NV, USA, 14–16 December 2022; pp. 53–57.

16. Pereira, L.M.; Salazar, A.; Vergara, L. Simultaneous Analysis of FMRI and EEG Biosignals: A Multimodal Fusion Approach. In Proceedings of the 2021 International Conference on Computational Science and Computational Intelligence (CSCI), Las Vegas, NV, USA, 15–17 December 2021; pp. 1673–1677.
17. Sohail, S.S.; Himeur, Y.; Kheddar, H.; Amira, A.; Fadli, F.; Atalla, S.; Copiaco, A.; Mansoor, W. Advancing 3D Point Cloud Understanding through Deep Transfer Learning: A Comprehensive Survey. *Inf. Fusion* **2025**, *113*, 102601. [[CrossRef](#)]
18. Özince, N.; Mengüç, E.C.; Emlek, A. Sparsity-Aware Complex-Valued Least Mean Kurtosis Algorithms. *Signal Process.* **2025**, *226*, 109637. [[CrossRef](#)]
19. Lu, M.; Xing, L.; Chen, B. Measuring Generalized Divergence for Multiple Distributions with Application to Deep Clustering. *Pattern Recognit.* **2025**, *157*, 110864. [[CrossRef](#)]
20. Peng, Y.; Zhang, S.; Zhou, Z. Frequency-Domain Diffusion Adaptation over Networks with Missing Input Data. *Signal Process.* **2025**, *226*, 109661. [[CrossRef](#)]
21. Xian, Y.; Peng, Y.-X.; Sun, X.; Zheng, W.-S. Distilling Consistent Relations for Multi-Source Domain Adaptive Person Re-Identification. *Pattern Recognit.* **2025**, *157*, 110821. [[CrossRef](#)]
22. Fang, Y.; Wu, J.; Wang, Q.; Qiu, S.; Bozoki, A.; Liu, M. Source-Free Collaborative Domain Adaptation via Multi-Perspective Feature Enrichment for Functional MRI Analysis. *Pattern Recognit.* **2025**, *157*, 110912. [[CrossRef](#)] [[PubMed](#)]
23. Peng, D.; Wu, J.; Han, T.; Li, Y.; Wen, Y.; Yang, G.; Qu, L. Disentanglement-Inspired Single-Source Domain-Generalization Network for Cross-Scene Hyperspectral Image Classification. *Knowl.-Based Syst.* **2024**, *303*, 112413. [[CrossRef](#)]
24. Yang, M.; Yang, R.; Tao, S.; Zhang, X.; Wang, M. Unsupervised Domain Adaptive Building Semantic Segmentation Network by Edge-Enhanced Contrastive Learning. *Neural Netw.* **2024**, *179*, 106581. [[CrossRef](#)]
25. Tang, Y.; Lyu, T.; Jin, H.; Du, Q.; Wang, J.; Li, Y.; Li, M.; Chen, Y.; Zheng, J. Domain Adaptive Noise Reduction with Iterative Knowledge Transfer and Style Generalization Learning. *Med. Image Anal.* **2024**, *98*, 1033027. [[CrossRef](#)]
26. Jecklin, S.; Shen, Y.; Gout, A.; Suter, D.; Calvet, L.; Zingg, L.; Straub, J.; Cavalcanti, N.A.; Farshad, M.; Fünstahl, P.; et al. Domain Adaptation Strategies for 3D Reconstruction of the Lumbar Spine Using Real Fluoroscopy Data. *Med. Image Anal.* **2024**, *98*, 103322. [[CrossRef](#)]
27. Guo, X.; Yin, J.; Yang, J. Fine Classification of Crops Based on an Inductive Transfer Learning Method with Compact Polarimetric SAR Images. *GIScience Remote Sens.* **2024**, *61*, 2319939. [[CrossRef](#)]
28. Moraes, D.; Campagnolo, M.L.; Caetano, M. Training Data in Satellite Image Classification for Land Cover Mapping: A Review. *Eur. J. Remote Sens.* **2024**, *57*, 2341414. [[CrossRef](#)]
29. Chen, M.; Wu, J.; Mao, T.; Du, R.; Zhao, B.; Lin, J.; Zhang, J. An Improved Method for Rapid Un-Collapsed Building Extraction from Post-Disaster High-Resolution Remote Sensing Imagery Based on Multi-Scale Feature Alignment. *Int. J. Digit. Earth* **2024**, *17*, 2344599. [[CrossRef](#)]
30. Okafor, S.C.; Wei, L.; Boamah, S.; Zhang, L.; Diallo, M.B. Enhanced Wheat Head Detection in Images Using Fourier Domain Adaptation and Random Guided Filter: Détection Améliorée Des Têtes de Blé Dans Les Images à l’aide de l’adaptation Du Domaine Fourier et Du Filtre Guidé Aléatoire. *Can. J. Remote Sens.* **2024**, *50*, 2367479. [[CrossRef](#)]
31. Lowe, D.G. Object Recognition from Local Scale-Invariant Features. In Proceedings of the Seventh IEEE International Conference on Computer Vision, Corfu, Greece, 20–25 September 1999; Volume 2, pp. 1150–1157.
32. Chikkerur, S.; Serre, T.; Tan, C.; Poggio, T. What and Where: A Bayesian Inference Theory of Attention. *Vis. Res.* **2010**, *50*, 2233–2247. [[CrossRef](#)]
33. Woo, S.; Park, J.; Lee, J.-Y.; Kweon, I.S. CBAM: Convolutional Block Attention Module. In Proceedings of the Computer Vision—ECCV 2018, Munich, Germany, 8–14 September 2018; Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y., Eds.; Springer International Publishing: Cham, Switzerland, 2018; pp. 3–19.
34. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going Deeper with Convolutions. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 1–9.
35. Gao, S.-H.; Cheng, M.-M.; Zhao, K.; Zhang, X.-Y.; Yang, M.-H.; Torr, P. Res2Net: A New Multi-Scale Backbone Architecture. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *43*, 652–662. [[CrossRef](#)]
36. Zhang, Y.; Li, K.; Li, K.; Wang, L.; Zhong, B.; Fu, Y. Image Super-Resolution Using Very Deep Residual Channel Attention Networks. In Proceedings of the Computer Vision—ECCV 2018, Munich, Germany, 8–14 September 2018; Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y., Eds.; Springer International Publishing: Cham, Switzerland, 2018; pp. 294–310.
37. Ganin, Y.; Ustinova, E.; Ajakan, H.; Germain, P.; Larochelle, H.; Laviolette, F.; Marchand, M.; Lempitsky, V. Domain-Adversarial Training of Neural Networks. *arXiv* **2016**, arXiv:1505.07818.
38. Pei, Z.; Cao, Z.; Long, M.; Wang, J. Multi-Adversarial Domain Adaptation. In Proceedings of the AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018; Volume 32.
39. van Ginneken, B.; Stegmann, M.B.; Loog, M. Segmentation of Anatomical Structures in Chest Radiographs Using Supervised Methods: A Comparative Study on a Public Database. *Med. Image Anal.* **2006**, *10*, 19–40. [[CrossRef](#)]
40. Candemir, S.; Jaeger, S.; Palaniappan, K.; Musco, J.P.; Singh, R.K.; Xue, Z.; Karargyris, A.; Antani, S.; Thoma, G.; McDonald, C.J. Lung Segmentation in Chest Radiographs Using Anatomical Atlases With Nonrigid Registration. *IEEE Trans. Med. Imaging* **2014**, *33*, 577–590. [[CrossRef](#)]

41. Jaeger, S.; Candemir, S.; Antani, S.; Wang, Y.-X.J.; Lu, P.-X.; Thoma, G. Two Public Chest X-Ray Datasets for Computer-Aided Screening of Pulmonary Diseases. *Quant. Imaging Med. Surg.* **2014**, *4*, 475–477. [[CrossRef](#)] [[PubMed](#)]
42. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention Is All You Need. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–8 December 2017; Curran Associates, Inc.: Red Hook, NY, USA, 2017; Volume 30.
43. Li, J.; Jing, M.; Lu, K.; Zhu, L.; Shen, H.T. Locality Preserving Joint Transfer for Domain Adaptation. *IEEE Trans. Image Process.* **2019**, *28*, 6103–6115. [[CrossRef](#)]
44. Long, M.; Cao, Z.; Wang, J.; Jordan, M.I. Conditional Adversarial Domain Adaptation. In Proceedings of the 32nd International Conference on Neural Information Processing Systems, Montreal, QC, Canada, 3–8 December 2018; Curran Associates Inc.: Red Hook, NY, USA, 2018; pp. 1647–1657.
45. He, K.; Sun, J.; Tang, X. Guided Image Filtering. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 1397–1409. [[CrossRef](#)] [[PubMed](#)]
46. Farbman, Z.; Fattal, R.; Lischinski, D.; Szeliski, R. Edge-Preserving Decompositions for Multi-Scale Tone and Detail Manipulation. *ACM Trans. Graph.* **2008**, *27*, 1–10. [[CrossRef](#)]
47. Kou, F.; Chen, W.; Wen, C.; Li, Z. Gradient Domain Guided Image Filtering. *IEEE Trans. Image Process.* **2015**, *24*, 4528–4539. [[CrossRef](#)] [[PubMed](#)]
48. Cheng, J.; Li, Z.; Gu, Z.; Fu, H.; Wong, D.W.K.; Liu, J. Structure-Preserving Guided Retinal Image Filtering and Its Application for Optic Disk Analysis. *IEEE Trans. Med. Imaging* **2018**, *37*, 2536–2546. [[CrossRef](#)]
49. Liu, W.; Zhang, P.; Huang, X.; Yang, J.; Shen, C.; Reid, I. Real-Time Image Smoothing via Iterative Least Squares. *ACM Trans. Graph.* **2020**, *39*, 1–24. [[CrossRef](#)]
50. Tao, X.; Zhou, C.; Shen, X.; Wang, J.; Jia, J. Zero-Order Reverse Filtering. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 222–230.

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.