machine learning & knowledge extraction

MDPI

Article

ExShall-CNN: An Explainable Shallow Convolutional Neural Network for Medical Image Segmentation

Vahid Khalkhali [1], Sayed Mehedi Azim [1] and Iman Dehzangi [1,2,*]

[1] Center for Computational and Integrative Biology, Rutgers University, Camden, NJ 08102, USA; vahid.khalkhali@rutgers.edu (V.K.); sayedmehedi.azim@rutgers.edu (S.M.A.)
[2] Department of Computer Science, Rutgers University, Camden, NJ 08102, USA
* Correspondence: i.dehzangi@rutgers.edu

Abstract: Explainability is essential for AI models, especially in clinical settings where understanding the model's decisions is crucial. Despite their impressive performance, black-box AI models are unsuitable for clinical use if their operations cannot be explained to clinicians. While deep neural networks (DNNs) represent the forefront of model performance, their explanations are often not easily interpreted by humans. On the other hand, hand-crafted features extracted to represent different aspects of the input data and traditional machine learning models are generally more understandable. However, they often lack the effectiveness of advanced models due to human limitations in feature design. To address this, we propose ExShall-CNN, a novel explainable shallow convolutional neural network for medical image processing. This model improves upon hand-crafted features to maintain human interpretability, ensuring that its decisions are transparent and understandable. We introduce the explainable shallow convolutional neural network (ExShall-CNN), which combines the interpretability of hand-crafted features with the performance of advanced deep convolutional networks like U-Net for medical image segmentation. Built on recent advancements in machine learning, ExShall-CNN incorporates widely used kernels while ensuring transparency, making its decisions visually interpretable by physicians and clinicians. This balanced approach offers both the accuracy of deep learning models and the explainability needed for clinical applications.

Keywords: explainability; image segmentation; shallow convolutional neural network

check for updates

Academic Editor: Yoichi Hayashi

Received: 31 December 2024
Revised: 9 February 2025
Accepted: 11 February 2025
Published: 15 February 2025

Citation: Khalkhali, V.; Azim, S.M.; Dehzangi, I. ExShall-CNN: An Explainable Shallow Convolutional Neural Network for Medical Image Segmentation. *Mach. Learn. Knowl. Extr.* **2025**, *7*, 19. https://doi.org/10.3390/make7010019

1. Introduction

Machine learning models can be simply defined as a mapping from the input data to the output data. This mapping must be usable and robust for unseen samples. Thus, it cannot be a simple tabular mapping. Instead, outputs must be computed from input data. Hence, machine learning models are usually mathematical functions [1].

Two major steps in the development of machine learning models are model selection and training by optimization. The neural networks are well-known sets of models that have shown strong capability to facilitate both steps [2].

In theory, shallow neural networks, such as two-layer multilayer perceptron (MLP) with activation functions, should be able to approximate any input–output relationship or function if they have enough trainable parameters and non-linearity [3].

Deep learning models are shown to need a much smaller number of trainable parameters while preserving similar performance to large shallow networks [4]. A lower number of parameters means shorter training time and higher trainability by finding a better optimum in a fixed running time. Many different architectures and building blocks are deployed for deep learning models, such as convolutional neural networks (CNNs) [2], recurrent neural networks (RNNs) [5], and transformer neural networks (TNNs) [6]. Both CNNs and TNNs are used in image processing and analysis applications and, based on dataset and model size, one may be preferred over the other. TNNs are usually good for large datasets while CNNs have better performance on small datasets [7,8].

Medical image processing encompasses a wide range of applications and domains, including areas such as radiology and pathology. However, despite the vast potential of this field, only a limited number of these applications have access to large, well-curated datasets [9]. Hence, while transformers show slightly better performance on large cohorts, deep CNNs are more favorable in medical image processing because they can perform better on smaller cohorts [7,10].

Two well-known deep CNNs for image segmentations are fully convolutional neural networks (FCNNs) [11] and U-Nets [12]. However, both FCNNs and U-Nets are deep neural networks, and their functions cannot be explained clearly [13]. Although many methods discussed in [14–18] can be used for explainability, only a few of them are widely accepted in clinical interpretation, since direct visualization of their features is not human-interpretable [19]. This opacity in decision-making introduces challenges in clinical setups where the ability to audit a model's decision-making process is crucial. Explainable models provide transparency by making the model's reasoning accessible, which allows clinicians to validate AI outputs in a manner aligned with their expertise [20].

To summarize, the main contributions of this article are listed as follows:

1. We propose an explainable shallow convolutional neural network (ExShall-CNN) which delivers performance on par with leading models like U-Net for medical image segmentation.
2. ExShall-CNN is built upon progressive advancements in machine learning for image segmentation. Thus, it can cover the most well-known kernels.
3. The features of ExShall-CNN are explainable since they can be visually examined by physicians and clinicians.

In the next section, we will elaborate more on human-interpretable and manually engineered features. At the same time, we will develop a theoretical model chronologically from the ground up to include the most essential hand-crafted and machine learning-based features in an explainable shallow convolutional neural network. ExShall-CNN and its source code are publicly available at https://github.com/MLBC-lab/ExShall-CNN (Access date: 1 February 2025).

2. Theory and Calculation

Otsu [21] suggested a global thresholding by minimizing the intra-class variance, which yields to maximizing inter-class variance for the background/foreground segmentation of images.

$$T_{otsu} = \underset{t}{\operatorname{argmin}} \left\{ P(t)(P(L-1) - P(t)) \left(\frac{\sum_{i=0}^{t-1} iP(i)}{P(t)} - \frac{\sum_{i=t}^{L-1} iP(i)}{P(L-1) - P(t)} \right) \right\} \quad (1)$$

where L is the length of the pixel range, e.g., 256 for grayscale images, and p and P are probability density and probability cumulative functions, respectively. The t value which minimizes the equation in the brackets determines the global Otsu threshold, T_{otsu}.

This equation is usually solved iteratively, and its computation is often very fast, even for very large images, since it only needs the frequency of colors or shades of grays to calculate the threshold value. Although this method is highly efficient, it determines a single threshold for the entire image without taking into account the spatial distribution of pixel values. This means it may overlook important variations and contextual information present in different regions of the image, potentially affecting the accuracy of segmentation or analysis. By not considering how pixel values are distributed spatially, this method may not capture the nuances needed for more complex image processing tasks. To address the disparity between local and global pixel value distributions, Sauvola [22] proposed using a local threshold:

$$T_{sauvola} = \mu_n \left(1 + k\left(\frac{\sigma_n}{r} - 1\right)\right) \tag{2}$$

where n is the local neighborhood diameter, and μ_n and σ_n are the mean and standard deviation of local neighborhoods, respectively. k and r are two coefficients that are determined based on application. It is common to select the value of r as half of the pixels' maximum value. Hence, two values of n (local neighborhood diameter) and k are needed to be optimized to have an acceptable performance. If segmentation labels are available, these two values can be optimized adaptively based on the dataset.

It can be inferred that the mean (μ_n) and standard deviation (σ_n) serve as local features, with Equation (2) acting as a classification threshold. Thus, the advantages of the Sauvola method compared to Otsu highlight that local features effectively differentiate between background and foreground in classification tasks. Consequently, a general approach can focus on extracting local features followed by classification. The only raw data available are the pixel values and their spatial arrangement, and all other features must be derived from this foundational information. Mean and standard deviation are examples of features created from this raw data, which can be further expanded.

Raw data come from two sources: the pixel values themselves and the values of adjacent pixels. Features can be defined as any transformations of these raw values into a new domain that simplifies classification, such as linear mappings. Keeping this in mind, various mappings have been proposed, referred to as kernel methods [23]. Some common kernels are depicted in Table 1 [24].

Table 1. Common kernel mappings; x_i is an RGB vector of the current or neighbors' pixels.

Kernel Name	Equation
Linear	$x_1^T x_2$
Cosine	$\frac{x_1 x_2^T}{\|\|x_1\|\|\|\|x_2\|\|}$
Polynomial	$\left(\gamma x_1^T x_2 + c_0\right)^d$
Sigmoid	$\tanh\left(\gamma x_1^T x_2 + c_0\right)$
Radial Basis Function (RBF)	$\exp\left(-\gamma\|\|x_1 - x_2\|\|^2\right)$
Laplacian	$\exp\left(-\gamma\|\|x_1 - x_2\|\|_1\right)$
Chi-Squared	$\exp\left(-\gamma\sum_i \frac{(x_1[i] - x_2[i])^2}{x_1[i] + x_2[i]}\right)$

Kernel methods fully encompass the Otsu and Sauvola methods, as both rely on two key factors: the mean and the standard deviation. These factors represent the linear or second-order combinations of adjacent pixel values. Hence, if a more complicated model can have all the given kernel mappings, it can also cover the Otsu and Sauvola methods. All the kernels contain some fundamental mathematical operations, such as summation, subtraction, multiplication, division, power, and exponent.

The Otsu, Sauvola, and kernel methods are notable for their explainability, particularly in visual terms. With these methods, the classifications directly reflect the visual content.

Additionally, they are mathematically transparent, illustrating the relationships between neighboring pixels in the classification process.

We will show that all the Otsu, Sauvola, and kernel-based methods can be assimilated in a shallow deep CNN with appropriate activation functions. A CNN provides a weighted summation of raw data, and activation can allow for the kernel mappings. To assimilate the fundamental operations, i.e., summation/subtraction, multiplication/division, power, and exponent, we propose the following activation functions:

$$e(x) = \exp(k^T x) \tag{3}$$

$$l(x) = \log(k^T x) \tag{4}$$

where x is the input vector and k is the convolution kernel vector. Since spatial convolution can be viewed as a correlation, the two vectors are multiplied element-wise, and their sum is calculated. The exponential and logarithmic functions are computed element-wise in mini-batches. Thus, the length of both vectors is equal to the kernel size. To prove that all basic mathematical operations can be implemented using these activation functions, a simple two-dimensional input is assumed as an example. An additional dimension is only an extension of this simplification. The equations presented in Table 2 are implemented in a shallow convolutional neural network.

Table 2. Feasibility of required operations with proposed activation functions.

Weighted Operation	Induction	Implementation
Summation/Subtraction	$k_1x_1 + k_2x_2$	$k_1x_1 + k_2x_2$
Multiplication	$k_1x_1 * k_2x_2$	$\exp(\log(k_1x_1) + \log(k_2x_2))$
Division	k_1x_1 / k_2x_2	$\exp(\log(k_1x_1) - \log(k_2x_2))$
Power	$(k_1x_1)^{k_2x_2}$	$\exp\exp(\log(k_1x_1) + \log(\log(k_2x_2)))$

As shown in Table 2, all required operations are implementable with shallow convolutional neural networks with the given activation functions if there are enough residuals. Residuals that are a well-known contribution to CNNs were first proposed in [25]. Since then, they have been widely used in almost all successful CNNs. It is easy to envision that with the right combinations and configurations of these operations, all the kernels listed in Table 1 can be incorporated, along with many more complex combinations that have yet to be categorized into existing kernel types. The exponentials of logarithms are identity operations, but when combined with different coefficients, they can construct highly non-linear functions. An important consideration for these equations (especially taking logarithms) is that negative numbers do not have real-valued logarithms. Hence, we designed complex convolutional neural networks (CCNNs) instead of real-valued CNNs. Theoretically, the imaginary parts of the outputs from the equations presented in Table 2 should be zero when the input values are real numbers. Since we used normalized pixel values, which are real, we can discard the imaginary part after the final activation function of each module. Although the imaginary component will not be exactly zero due to computational errors, it will be very close to zero. We chose to work in the complex domain because intermediate calculations often involve significant imaginary components.

Moreover, it is known that the receptive fields of CNNs are enlarged through depth [26,27]. Since we are trying to use a shallow CNN for its higher explainability, we cannot rely on large receptive fields by depth, and we must compensate for it by the size of the kernels. In deep CNNs, kernel sizes of three, five, and sometimes seven are the most common. If we assume that the kernel is center-aligned, then by a large kernel of size seven, two neighbor pixels in every direction are included in the equation. After many layers in a deep CNN, this receptor field will be significantly increased proportionally to the kernel size, stride, and dilation of hidden layers. In a shallow CNN, there is no depth growth of the receptor field, and a wide range of sizes of kernels and dilations take the responsibility of achieving almost similar receptor field sizes in deep CNNs. Larger kernel sizes cause the non-linear equations to converge slowly through backpropagation. Hence, they usually need more time to be trained, while their number of parameters is usually significantly less than deep CNNs.

3. Material and Methods

In this study, we used the Retina Blood Vessel segmentation [28] and the International Skin Imaging Collaboration (ISIC) [29–31] datasets, two widely used benchmarks for medical image segmentation. The Retina Blood Vessel dataset contains 100 color images with their binary mask. The dataset is divided into two subsets: 80 images, along with their corresponding masks, are designated for training, while the remaining 20 images and their corresponding masks are reserved for testing. This dataset provides a well-curated collection of retinal fundus images, each with precise annotations for blood vessel segmentation. Accurate segmentation of blood vessels is essential in ophthalmology, as it assists in the early diagnosis and treatment of retinal diseases, including diabetic retinopathy and macular degeneration [32]. The ISIC 2016, 2017, and 2018 datasets consist of 7723 images along with their corresponding binary masks. Each dataset for the three years includes separate training, validation, and test subsets. We randomly chose 100 images across these years, with 60 selected from the training sets, 20 from the validation sets, and 20 from the test sets.

The structure of our proposed shallow CNN model is shown in Figure 1. As shown in this figure, there are three different module types, namely, Conv, Log-Conv-Exp (LCE), and Conv-Log-Conv-Exp (CLCE), each with multiple kernel sizes. The Conv layer finds a weighted summation of juxtaposed pixels and assimilates a linear kernel (Table 2, row 1). The LCE layer calculates multiplications and divisions of neighbor pixels and assimilates cosine, polynomial, and somehow sigmoid kernels (Table 2, rows 2 and 3). Finally, the CLCE layer approximates RBF, Laplacian, and Chi-Squared kernels (Table 2, row 4). Since the weights and biases of convolutional layers can be positive or negative, the expected logarithm values are complex numbers. Hence, the model is totally implemented in the complex number set except for the last layer, which aggregates real numbers. Unlike other layers, the Conv and Aggregate layers do not have non-linear activation functions. Here, we use kernel sizes 1, 3, 5, 9, 13, 17, 21, and 25. As a result, we have 8 * 3 = 24 modules.

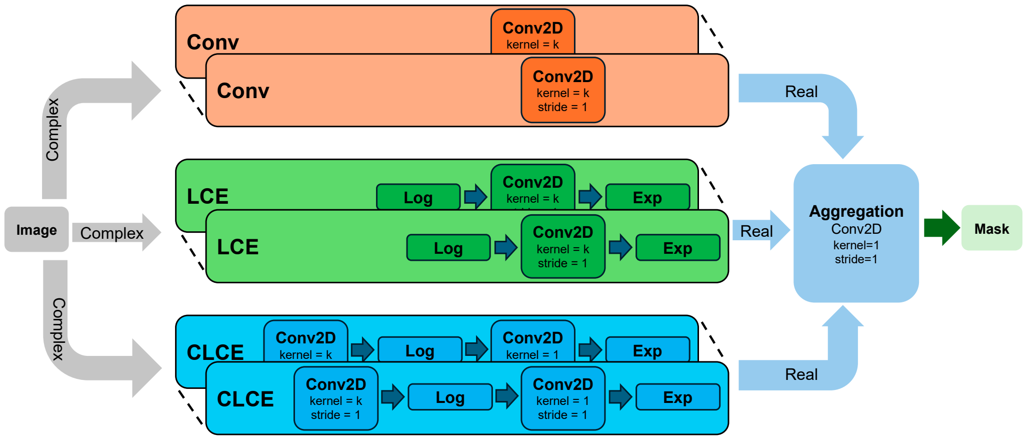

Figure 1. ExShall-CNN with 24 parallel complex modules, outputting real aggregation.

The developments are implemented in Python 3.10 and PyTorch 2.0 [33] environments. A batch size of 4 is used for all models and datasets. The Adam optimizer [34] is tested with different learning rates (1×10^{-2}, 1×10^{-3}, 1×10^{-4}, 1×10^{-5}) across all models. The best results for each model on each dataset are selected and reported.

We compare our explainable shallow CNN to two well-known models for image segmentation, namely FCNN [11] and U-Net [12]. The former architecture is similar to our proposed model, although much deeper, and the latter is the current state-of-the-art in medical image segmentation. Both models have two important differences from our proposed shallow CNN: Both are DNNs with several hidden layers and also use typical ReLU activation functions.

All three models are trained and validated on a similar training dataset and evaluated on the testing set. The training loss and optimizer are negative logarithms of the Jaccard score and Adam with a learning rate of 1×10^{-4}, respectively.

4. Results and Discussion

In this section, we present our achieved results and compare them to the state-of-the-art deep learning models. We then discuss the explainability of our model.

4.1. Comparison of Results

Here, we use Jaccard similarity and Sørensen–Dice metrics to evaluate the performance of our models. Since the segmentation is in a binary format (background and foreground), the Dice and F1 scores are expected to be similar. Tables 3 and 4 show the performance of the three models. In this study, we mainly compare our results with other state-of-the-art deep learning models. As is provided on the Kaggle website [28], these models significantly outperform conventional machine learning models. However, these models are usually very deep and not explainable. The aim of this study is to address these issues by proposing a shallow explainable deep learning model. On the training dataset, Shallow-CNN is weaker than both FCNN and U-Net. However, on the test dataset, Shallow-CNN operates better than FCNN while performing comparably to U-Net. The superior generalization of Shallow-CNN compared to FCNN is attributed to its smaller parameter space, which reduces the risk of overfitting. Another result from a smaller U-Net with 0.36M parameters (U-Net-G) on a similar dataset, as indicated in [35], achieved a DICE score of 0.88. While the U-Net performance is superior to that of ExShall-CNN, it is important to note that U-Net models have a significantly larger number of parameters—at least an order of magnitude more—compared to the proposed shallow model. By leveraging hand-crafted features and interpretable transformations, Shallow-CNN achieves a balance between model complexity and effective feature representation, leading to better performance on unseen data. These findings suggest that although Shallow-CNN does not surpass U-Net in performance, it demonstrates better generalization than FCNN. Additionally, while deep models like FCNN and U-Net pose challenges in terms of explainability, the shallow model is more transparent, as will be elaborated in the next section.

Table 3. Performance of the three models on Retina Blood Vessel.

Model	Num. Parameters	Retina Blood Vessel			
		Training Dataset		Testing Dataset	
		Jaccard (%)	DICE (%)	Jaccard (%)	DICE (%)
Fully CNN	54,304,086	63.3	77.6	56.8	72.4
U-Net	31,043,521	63.5	77.7	65.9	79.0
Shallow CNN	39,698	56.2	72.0	58.3	73.6

Table 4. Performance of the three models on International Skin Imaging Collaboration (ISIC).

Model	Num. Parameters	ISIC 2016, 2017, 2018			
		Training Dataset		Testing Dataset	
		Jaccard (%)	DICE (%)	Jaccard (%)	DICE (%)
Fully CNN	54,304,086	82.8	90.6	63.8	77.9
U-Net	31,043,521	56.0	71.2	69.4	81.9
Shallow CNN	39,698	45.5	62.5	68.5	81.3

A key distinction between shallow and deep convolutional neural networks is the size of the visual receptive field. It is commonly believed that increasing the depth of a neural network results in larger receptive fields compared to shallow models. To address this, we expanded the receptive field in our ExShall-CNN by using larger scales. However, the curse of dimensionality limited the model's ability to effectively learn the underlying patterns, resulting in poor performance. In practice, employing kernel sizes such as 21 and 25 enabled ExShall-CNN to achieve a sufficiently large receptive field, which mitigated the challenges posed by the network's depth and ultimately enhanced the model's performance.

Comparing these three models has several limitations. Here, we evaluated them using only two datasets with a relatively small sample size. A more extensive study with a larger dataset could provide clearer insights into the differences in model performance. Moreover, the Otsu, Sauvola, and kernel-based transformations are all hand-crafted feature extraction methods. While we acknowledge that features extracted by deep learning models are often the most effective, our goal was to explore the visual impact of these hand-crafted features. To this end, we developed ExShall-CNN, which can extract features similar to those of hand-crafted methods but with the performance benefits of deep learning feature sets. Therefore, we compared our model primarily with its main competitors—deep learning models—rather than with hand-crafted feature extraction methods.

4.2. Explainability

Unlike deep learning models, shallow networks have explainability capabilities [36]. Explainability helps to understand the reason behind the decisions that a model makes, which is very important in the clinical reliability of an AI model [16]. Given the complexity of deep learning models with multiple layers, it is often unclear which layer provides the most effective visual explanation. For example, the authors in [16] investigate several explainability approaches. Visual explanations offer several advantages, including ease of use and validity. Various methods can be used to generate visual explanations, and our approach is similar to Class Activation Mapping (CAM) [37]. However, while CAM provides an indirect indication of the impact of sequential layers (which can be numerous in architectures like VGG [38]), our method directly reflects the influence of each layer, thanks to the shallower nature of our model.

The proposed Shallow-CNN is explained by finding the most important modules in Figure 1 on the Retina Blood Vessel dataset. Each module is an optimized transformation (kernel) on the training dataset. We used model-based analysis to calculate the impact of each module on the performance of our model [39]. As shown in Figure 1, our model contains 24 modules. In each permutation, we remove that module from the model and try to compute the performance, such as the Dice score. If the module plays an important role, then the score should drop down significantly. We used the following scoring method to compute the impact of each module:

$$Impact_i = \frac{DICE_{max} - DICE_i}{DICE_{max}} = 1 - \frac{DICE_i}{DICE_{max}} \quad (5)$$

where $DICE_{max}$ is the maximum score that the model can achieve by including all modules, $DICE_i$ is the score by removing the module i from the model, and $Impact_i$ is the impact score of module i. $Impact_i$ is a score between 0 and 1, where 0 means no impact and 1 means the highest impact. To compute Equation (5), it is crucial that the model is not retrained after a layer is removed, and instead, it should be evaluated using precomputed parameters. Equation (5) makes meaningful sense of the significance of each module as long as the model remains shallow. Therefore, we can use this equation for shallow models and not for deep models. Equation (5) aims to deconstruct the Aggregation module (the final layer) and clarify its functionality. During training, the Aggregation module assigns greater weight to the most influential features from the middle modules. Consequently, removing these key modules will substantially degrade the performance of ExShall-CNN, as quantified by Equation (5). It is important to note that the Aggregation module makes its decision for each output pixel by performing a weighted sum of the corresponding pixels from all modules at the same location since the kernel size of the Aggregation module is one. The impact scores for Shallow-CNN are computed and depicted in Figure 2. As shown in Figure 2, some modules such as 1, 9, and 15 have the highest impact, while others such as 2, 5, and 12 have the lowest impact. This suggests that by retaining the most influential modules and eliminating the least impactful ones, we can attain performance comparable to that of the main Shallow-CNN.

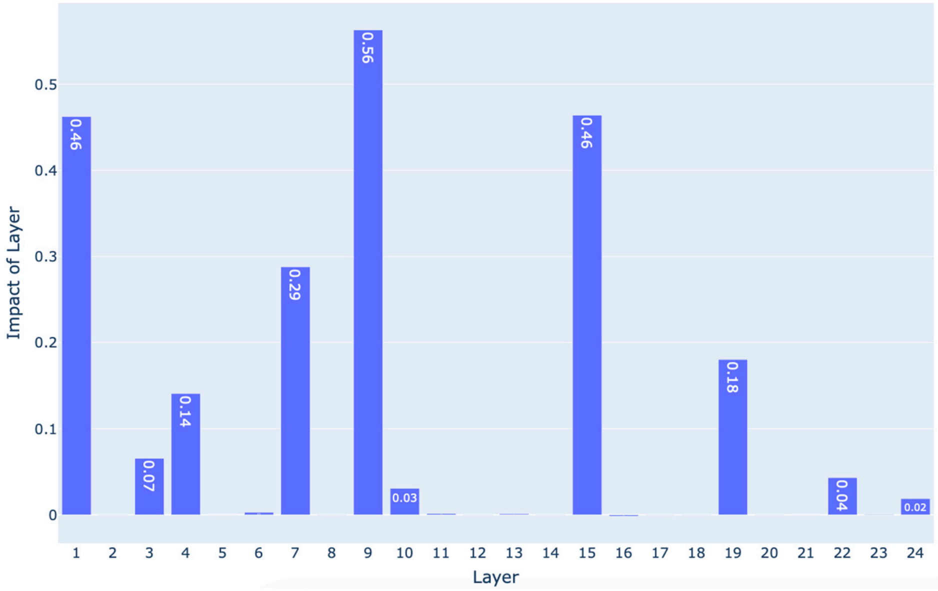

Figure 2. Comparison of layer impact on the performance of the Shallow-CNN. The higher values show more impact. Layers 9, 1, 15, 7, and 19 have the most impactful role in the decisions, respectively.

We identify modules with an impact score exceeding 80% of the interquartile range of the total 24 impact scores, selecting modules 1, 7, 9, 15, and 19, shown in Table 5. The transformations of the selected modules are shown in Figure 3. As Figure 3 shows, while the transformation of kernel 1 is linear, the other kernels have a non-linear response to the input color variation. The comparison of the background and foreground between the original and transformed images supports this observation.

The explainability of the model is primarily achieved through visual representation, as demonstrated in the example shown in Figure 3. Our main goal in introducing the innovative activation functions (Equations (3) and (4)) was to enhance explainability and also to integrate well-known kernel functions. Deep learning models often consist of numerous layers, making it unclear which specific layer provides the most effective visual explainability.

Table 5. Characteristics of the most impactful modules.

Kernel Index	Module Type	Kernel Size
1	Conv	1
7	Conv	21
9	LCE	1
15	LCE	17
19	CLCE	3

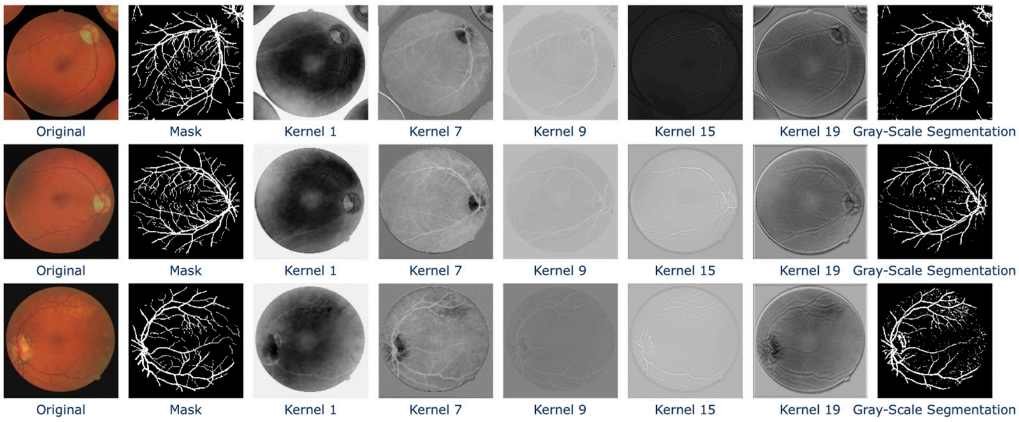

Figure 3. Transformations of the most important kernels on the original image. The mask serves as the ground truth, while the grayscale segmentation represents the model's output. Visual outputs from the most important layers—1, 7, 9, 15, and 19—are displayed. These outputs help to clarify the final decisions made by the ExShall-CNN model.

Visualizing transformed images greatly enhances the reliability of black-box AI models in clinical settings. This allows clinicians to understand the basis for decisions, enabling them to assess whether the transformations and resulting inferences are valid. For instance, clinicians and radiologists often rely on certain statistical features, such as variance, to identify specific patterns in organs like the liver or lungs. As a result, models that base their diagnoses on these features tend to gain their interest. The transformed image can highlight which aspects of the original image are most influential in the model's final decision.

5. Conclusions

Deploying an AI model in clinical applications requires careful consideration, particularly regarding its explainability. Although deep neural networks have demonstrated exceptional performance in comparison to human diagnosis, they remain black-box models that require transparency. In the traditional machine learning era, human-interpretable features were often developed, making the reasoning behind a model's decisions clear and eliminating the black-box nature. However, the performance of these hand-crafted features was inadequate because they did not fully leverage data-driven approaches.

In this study, we explored the evolution of hand-crafted features for image segmentation and demonstrated that a well-designed explainable shallow convolutional neural network (ExShall-CNN) can achieve performance comparable to deep CNN models while offering significantly better explainability. ExShall-CNN is developed as a data-oriented extension of the traditional kernel method capable of handling complex kernels and incorporating the most well-known kernels. With respect to applicability, we believe that our proposed model can be used as a replacement for well-known deep learning models in medical image processing applications when explainability and simplicity are important.

In conclusion, this study introduces the explainable shallow convolutional neural network (ExShall-CNN), which balances performance and explainability in medical image segmentation. While not outperforming U-Net, ExShall-CNN generalizes better than fully convolutional neural networks (FCNN), especially on unseen data, due to its smaller parameter space and reduced risk of overfitting. By leveraging hand-crafted features and explainable transformations, it offers transparency and interpretability, key for clinical applications. Although this study is limited by dataset size and reliance on hand-crafted features, the results highlight the potential of shallow models for explainable, high-performance medical image segmentation.

In the future, we aim to enhance ExShall-CNN for multi-resolution image analysis to address the dimensionality challenges associated with larger kernels. We also aim to explore both the theoretical capabilities and practical performance of the proposed model. This will involve standard theoretical analysis and an examination of practical factors such as pattern complexity and database size. Additionally, we plan to evaluate the performance of ExShall-CNN on a broader range of biomedical images, which will help us further refine and optimize the model.

While the initial results presented in this paper provide a strong foundation, we recognize that further validation on a more diverse set of datasets would enhance our findings. Due to the scope of this study, we have focused on preliminary experiments, while we are actively expanding our testing process. In our future work, we aim to conduct a more comprehensive evaluation of larger and more varied datasets.

ExShall-CNN and its source code are publicly available at https://github.com/MLBC-lab/ExShall-CNN (Access Date: 1 February 2025).

Author Contributions: Conceptualization, V.K., S.M.A., and I.D.; Methodology, V.K. and I.D.; Software, validation, writing—original draft, V.K.; Writing, reviewing, and editing, S.M.A. and I.D. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The source code and data are publicly available at https://github.com/MLBC-lab/ExShall-CNN (Access Date: 1 February 2025), and the data are publicly available at https://www.kaggle.com/datasets/abdallahwagih/retina-blood-vessel (Access Date: 1 February 2025), respectively.

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

Abbreviations

The following abbreviations are used in this manuscript:

DNN	Deep neural network
CNN	Convolutional neural network
RNN	Recurrent neural network
TNN	Transformer neural network
FCNN	Fully convolutional neural network
MLP	Multilayer perceptron

References

1. Murphy, K.P. *Machine Learning: A Probabilistic Perspective*; The MIT Press: Cambridge, MA, USA, 2012.
2. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [CrossRef] [PubMed]
3. Hornik, K.; Stinchcombe, M.; White, H. Multilayer feedforward networks are universal approximators. *Neural Netw.* **1989**, *2*, 359–366. [CrossRef]
4. Mhaskar, H.; Liao, Q.; Poggio, T. When and why are deep networks better than shallow ones? In Proceedings of the AAAI Conference On Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017; Volume 31.
5. Rumelhart, D.E.; Hinton, G.E.; Williams, R.J. Learning internal representations by error propagation, parallel distributed processing, explorations in the microstructure of cognition, ed. de rumelhart and j. mcclelland. *Biometrika* **1986**, *71*, 6.

6. Vaswani, A. Attention is all you need. In Proceedings of the Advances in Neural Information Processing Systems 30 (NIPS 2017), Long Beach, CA, USA, 4–9 December 2017.
7. Murphy, Z.R.; Venkatesh, K.; Sulam, J.; Yi, P.H. Visual transformers and convolutional neural networks for disease classification on radiographs: A comparison of performance, sample efficiency, and hidden stratification. *Radiol. Artif. Intell.* **2022**, *4*, e220012. [CrossRef] [PubMed]
8. Shamshirband, S.; Fathi, M.; Dehzangi, A.; Chronopoulos, A.T.; Alinejad-Rokny, H. A review on deep learning approaches in healthcare systems: Taxonomies, challenges, and open issues. *J. Biomed. Inform.* **2021**, *113*, 103627. [CrossRef] [PubMed]
9. Guan, H.; Yap, P.T.; Bozoki, A.; Liu, M. Federated learning for medical image analysis: A survey. *Pattern Recognit.* **2024**, *151*, 110424. [CrossRef] [PubMed]
10. Khan, M.S.I.; Rahman, A.; Debnath, T.; Karim, M.R.; Nasir, M.K.; Band, S.S.; Mosavi, A.; Dehzangi, I. Accurate brain tumor detection using deep convolutional neural network. *Comput. Struct. Biotechnol. J.* **2022**, *20*, 4733–4745. [CrossRef] [PubMed]
11. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3431–3440.
12. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Proceedings of the Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, 5–9 October 2015; Proceedings, Part III 18; Springer: Berlin/Heidelberg, Germany, 2015; pp. 234–241.
13. Ibrahim, R.; Shafiq, M.O. Explainable Convolutional Neural Networks: A Taxonomy, Review, and Future Directions. *ACM Comput. Surv.* **2023**, *55*, 1–37. [CrossRef]
14. Barredo Arrieta, A.; Díaz-Rodríguez, N.; Del Ser, J.; Bennetot, A.; Tabik, S.; Barbado, A.; Garcia, S.; Gil-Lopez, S.; Molina, D.; Benjamins, R.; et al. Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Inf. Fusion* **2020**, *58*, 82–115. [CrossRef]
15. Band, S.S.; Yarahmadi, A.; Hsu, C.-C.; Biyari, M.; Sookhak, M.; Ameri, R.; Dehzangi, I.; Chronopoulos, A.T.; Liang, H.-W. Application of explainable artificial intelligence in medical health: A systematic review of interpretability methods. *Inform. Med. Unlocked* **2023**, *40*, 101286. [CrossRef]
16. van der Velden, B.H.M.; Kuijf, H.J.; Gilhuijs, K.G.A.; Viergever, M.A. Explainable artificial intelligence (XAI) in deep learning-based medical image analysis. *Med. Image Anal.* **2022**, *79*, 102470. [CrossRef]
17. Zhang, Q.; Wu, Y.N.; Zhu, S.-C. Interpretable Convolutional Neural Networks. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 8827–8836.
18. Huang, S.; Xu, Z.; Tao, D.; Zhang, Y. Part-Stacked CNN for Fine-Grained Visual Categorization. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 1173–1182.
19. Zhang, Y.; Tino, P.; Leonardis, A.; Tang, K. A Survey on Neural Network Interpretability. *IEEE Trans. Emerg. Top. Comput. Intell.* **2021**, *5*, 726–742. [CrossRef]
20. Salahuddin, Z.; Woodruff, H.C.; Chatterjee, A.; Lambin, P. Transparency of deep neural networks for medical image analysis: A review of interpretability methods. *Comput. Biol. Med.* **2022**, *140*, 105111. [CrossRef] [PubMed]
21. Otsu, N. A threshold selection method from gray-level histograms. *Automatica* **1975**, *11*, 23–27. [CrossRef]
22. Sauvola, J.; Pietikäinen, M. Adaptive document image binarization. *Pattern Recognit.* **2000**, *33*, 225–236. [CrossRef]
23. Scholkopf, B.; Smola, A.J. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*; MIT Press: Cambridge, MA, USA, 2001.
24. Zhang, J.; Marszałek, M.; Lazebnik, S.; Schmid, C. Local Features and Kernels for Classification of Texture and Object Categories: A Comprehensive Study. *Int. J. Comput. Vis.* **2006**, *73*, 213–238. [CrossRef]
25. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
26. Araujo, A.; Norris, W.; Sim, J. Computing Receptive Fields of Convolutional Neural Networks. *Distill* **2019**, *4*, e21. [CrossRef]
27. Luo, W.; Li, Y.; Urtasun, R.; Zemel, R. Understanding the effective receptive field in deep convolutional neural networks. In Proceedings of the 30th International Conference on Neural Information Processing Systems, Barcelona, Spain, 5–10 December 2016; pp. 4905–4913.
28. Retina Blood Vessel. Available online: https://www.kaggle.com/datasets/abdallahwagih/retina-blood-vessel (accessed on 1 July 2024).
29. Gutman, D.; Codella, N.C.F.; Celebi, E.; Helba, B.; Marchetti, M.; Mishra, N.; Halpern, A. Skin Lesion Analysis toward Melanoma Detection: A Challenge at the International Symposium on Biomedical Imaging (ISBI) 2016, hosted by the International Skin Imaging Collaboration (ISIC). *arXiv* **2016**, arXiv:1605.01397. [CrossRef]
30. Codella, N.C.F.; Gutman, D.; Celebi, M.E.; Helba, B.; Marchetti, M.A.; Dusza, S.W.; Kalloo, A.; Liopyris, K.; Mishra, N.; Kittler, H.; et al. Skin lesion analysis toward melanoma detection: A challenge at the 2017 International symposium on biomedical imaging (ISBI), hosted by the international skin imaging collaboration (ISIC). In Proceedings of the 2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018), Washington, DC, USA, 4–7 April 2018; pp. 168–172. [CrossRef]

31. Codella, N.; Rotemberg, V.; Tschandl, P.; Emre Celebi, M.; Dusza, S.; Gutman, D.; Helba, B.; Kalloo, A.; Liopyris, K.; Marchetti, M.; et al. Skin Lesion Analysis Toward Melanoma Detection 2018: A Challenge Hosted by the International Skin Imaging Collaboration (ISIC). *arXiv* **2019**, arXiv:1902.03368.
32. Cervantes, J.; Cervantes, J.; García-Lamont, F.; Yee-Rendon, A.; Cabrera, J.E.; Jalili, L.D. A comprehensive survey on segmentation techniques for retinal vessel segmentation. *Neurocomputing* **2023**, *556*, 126626. [CrossRef]
33. Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. *PyTorch: An Imperative Style, High-Performance Deep Learning Library*; Curran Associates Inc.: Red Hook, NY, USA, 2019.
34. Kingma, D.P. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
35. Wang, K.Y.W.; Hugonot, J.; Fua, P.; Salzmann, M. Recurrent U-Net for Resource-Constrained Segmentation. Available online: https://ieeexplore.ieee.org/document/9010910/ (accessed on 8 January 2025).
36. Marques dos Santos, J.D.; Marques dos Santos, J.P. Towards XAI: Interpretable Shallow Neural Network Used to Model HCP's fMRI Motor Paradigm Data. In *Bioinformatics and Bio-Medical Engineering*; Springer International Publishing: Cham, Switzerland, 2022; pp. 260–274.
37. Zhou, B.; Khosla, A.; Lapedriza, A.; Oliva, A.; Torralba, A. Learning Deep Features for Discriminative Localization. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016.
38. Liu, S.; Deng, W. Very deep convolutional neural network based image classification using small training sample size. In Proceedings of the 2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR), Kuala Lumpur, Malaysia, 3–6 November 2015.
39. Fisher, A.; Rudin, C.; Dominici, F. All models are wrong, but many are useful: Learning a variable's importance by studying an entire class of prediction models simultaneously. *J. Mach. Learn. Res.* **2019**, *20*, 1–81.

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.