Proceedings of the International Conference on
Formal Methods and Foundations of Artificial Intelligence
Eszterházy Károly Catholic University
Eger, Hungary, June 5–7, 2025
pp. 148–160
DOI: 10.17048/fmfai.2025.148

Explainable image segmentation with wavelet-network*

Hanna-Georgina Lieb, Tamás Kaszta

Babeș-Bolyai University
{hanna.lieb,tamas.kaszta}@ubbcluj.ro

Abstract. Recent advances in artificial intelligence and its widespread adoption have imposed the necessity of research targeting the inner mechanisms of intelligent systems. We lack the exact mathematical tools needed to grasp what led to a certain output. The term *explainability* has recently emerged in the context of artificial intelligence (AI) as an area of development. An efficient way to introduce a checkpoint into decision-making systems is to incorporate prototype units. These bridge the difference between input image space and feature space, offering us a glimpse into an intermediary phase of decision-making. We created a new model – WaveProtoSeg – from the WaveProtoPNet classification model, combining image segmentation with the wavelet transform as a feature extractor. Our experiments were conducted on the Cityscapes dataset, which gathers real street scenes. Although we did not achieve the accuracy of the original paper, we explored various configurations of the system, and we managed to build a versatile system.

Keywords: image segmentation, wavelet, explainability, prototype

1. Introduction

Intelligent systems are experiencing significant advances at a rapid pace and are being progressively integrated into safety-critical fields, including healthcare, defense, and autonomous driving. Despite their accuracy, deep learning models often function as black boxes, offering little insight into their decision-making processes, posing risks when human lives or assets are at stake. This has led to growing interest in explainable artificial intelligence (XAI), which aims to improve model

*This work benefitted from the project "Romanian Hub for Artificial Intelligence - HRIA", Smart Growth, Digitization, and Financial Instruments Program, 2021-2027, MySMIS no. 334906.

transparency and trust [17, 19].

Prototype-based approaches have demonstrated significant potential in this field. Prototypes may be conceptualised as instances in the feature space that exemplify a certain class, encapsulating the core properties characteristic of that class, analogous to learning vector quantization [10]. They capture key features during training and serve as reference points for interpreting predictions from new data. They can reveal model errors, biases, and decision rationale, making them valuable tools for developers.

However, the implementation of prototypes poses a significant challenge, the need to connect the initial raw input with the more abstract high-level feature space. Many of the current methodologies to address this issue typically depend on complex, computationally demanding black-box architectures. In contrast, wavelet transforms [15] provide a lightweight alternative, effectively capturing spatial and frequency information from images viewed as 2D signals.

This study presents WaveProtoSeg (see Figure 2), a segmentation model that integrates wavelet-based feature extraction with interpretable prototype learning, starting from our previous work, WaveProtoPNet [12, 13]. Similarly to the ProtoSeg [18] framework, this approach focuses on achieving pixel-wise image segmentation, with the main difference in using wavelets as feature extractors. Its accuracy is assessed using the realistic Cityscapes dataset [3], which provides a rich and complex urban environment for evaluation, relevant to the needs of autonomous driving. WaveProtoSeg aims to deliver transparent and interpretable output.

The article is structured as follows. In Section 2 a review of the literature is conducted, after which the model is presented in Section 3, followed by the experiments and discussion in Section 4, ending with the conclusion in Section 5.

2. Literature review

2.1. Image segmentation and explainability

Image segmentation is a fundamental task in computer vision that assigns a class label to each pixel of an image, dividing it into semantically coherent regions. Unlike image classification, which outputs a single label per image, segmentation operates at a finer granularity, making it essential for applications like medical imaging, autonomous driving, and remote sensing [4, 22].

Although classification has historically been the entry point for XAI in research and industry, segmentation has received increasing attention since the late 2010s [25]. Early successful segmentation models include U-Net [16], designed for biomedical image analysis, and SegNet [1], which uses an encoder-decoder structure to map input images to prediction in pixels. More recently, transformer-based models such as Segmenter [21] have been introduced to capture long-range dependencies in segmentation tasks.

Segmentation poses unique challenges for interpretability. Decisions at the pixel level must account for local context and global consistency, often leading to com-

plex interactions between neighbouring pixels. This makes explanations harder to interpret and evaluate, especially when no clear ground truth is available for what constitutes a valid explanation.

Explainable image segmentation techniques can be broadly categorised into post-hoc and architecture-based methods [8]. For post-hoc methods, the model does not directly explain its predictions, and an independent model is employed to obtain this information. Architecture-based methods are inherently explainable, i.e. providing an explanation alongside the prediction. Among these inherently interpretable models, prototype-based explanations offer an intuitive and interpretable approach by associating predictions with representative examples from the training data. Each class is linked to a set of learnt prototypes, which provide visual justification for predictions based on similarity to prototypical regions [9]. Counterfactual explanations form another important branch, focusing on identifying the minimal changes to input that would alter the model output, thus helping to understand the decision boundaries and increase the robustness against adversarial perturbations. Perturbation-based methods systematically occlude or modify parts of the input image to analyse the resulting changes in output, offering insight into which regions are the most influential. Gradient-based approaches, such as saliency maps and Grad-CAM [20], utilise gradient information from subsequent network layers to produce heat maps that highlight the regions most responsible for a particular prediction [7, 24]. Finally, architecture-based methods are interpretable by design, embedding explainability directly into the model's structure rather than relying on post-hoc interpretation.

Our proposed model is similar to the previous ProtoSeg architecture introduced in [18], which demonstrated interpretable semantic segmentation through the use of prototypical image patches learnt from training data, a patch of an image being a smaller region of it. This model expands on ProtoPNet [2], which served as the basis for our earlier development of the WaveProtoPNet model. ProtoSeg incorporates a diversity loss to encourage the model to learn a broad and representative set of prototypes, thereby enhancing interpretability. In our work, we retain the core structure and interpretability framework of ProtoSeg, while extending its capabilities by integrating a wavelet-based feature extractor. This modification results in more transparent feature representations.

2.2. Wavelets in image segmentation

Wavelet transforms offer a powerful tool for analysing signals in both space and frequency domains simultaneously. Unlike the Fourier transform, which only provides frequency information and loses spatial localisation, wavelets allow multiresolution analysis, capturing both coarse structures and fine details [15].

The Discrete Wavelet Transform (DWT) decomposes a signal into approximation and detail coefficients through a series of low-pass and high-pass filters. In the context of image processing, this results in a set of sub-bands that highlight different spatial features, such as edges and textures. Wavelets are localised, meaning they are compact in both space and frequency, making them well suited for tasks

like denoising, compression, and feature extraction.

The two-dimensional wavelet transform applies these low- and high-pass filters to images by filtering along rows and columns, leading to four quarter-sized images at each decomposition level, focusing on: approximation coefficients (low+low) and detail coefficients in horizontal (low+high), vertical (high+low), and diagonal (high+high) directions. This hierarchical decomposition can be recursively applied to the approximation result for a finer scale analysis (i.e., higher decomposition levels); see Figure 1.

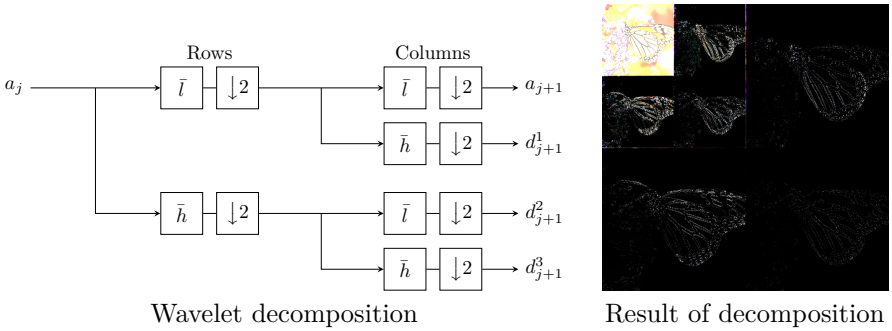

Figure 1. *(left)* Wavelet transform using low-pass and high-pass filters (l, h). *(right)* The result of the decomposition on an image. The average component (a_{j+1}, upper-left corner) underwent an additional decomposition ($\Rightarrow$ decomposition on 2 levels).

In wavelet segmentation methods, such as the Wavelet Segmentation Method (WSM) demonstrated improved performance in capturing background and small-scale features compared to classical threshold-based techniques [6]. In sonar imaging applications, wavelet filters have been used to reduce noise while enhancing feature localisation, showing the benefit of wavelet-based multiscale representations [23].

In deep learning, several models incorporate wavelet transforms into convolutional neural networks (CNNs) to replace conventional downsampling layers (e.g., max-pooling, strided convolution). For example, Haar wavelets have been used to decompose feature maps into low-frequency and high-frequency components during encoding. Integrating Haar wavelet downsampling improves segmentation accuracy, particularly in boundary regions. [26]

A notable advancement is XNet [27], a deep learning architecture that integrates DWT and Inverse Wavelet Transform into a U-Net-style encoder-decoder. XNet captures both global context and local detail by separating and recombining frequency information, leading to improved performance even under semi-supervised conditions. However, its success is limited when high-frequency features are not prominent in the input data.

In general, wavelets offer a lightweight and effective alternative for multiscale feature extraction in segmentation tasks, particularly where interpretability and

boundary precision are critical.

3. Methodology

To incorporate wavelets into the segmentation framework, we explore three distinct architectural configurations (see Figure 2), each offering different trade-offs in terms of the granularity of feature representation and output resolution.

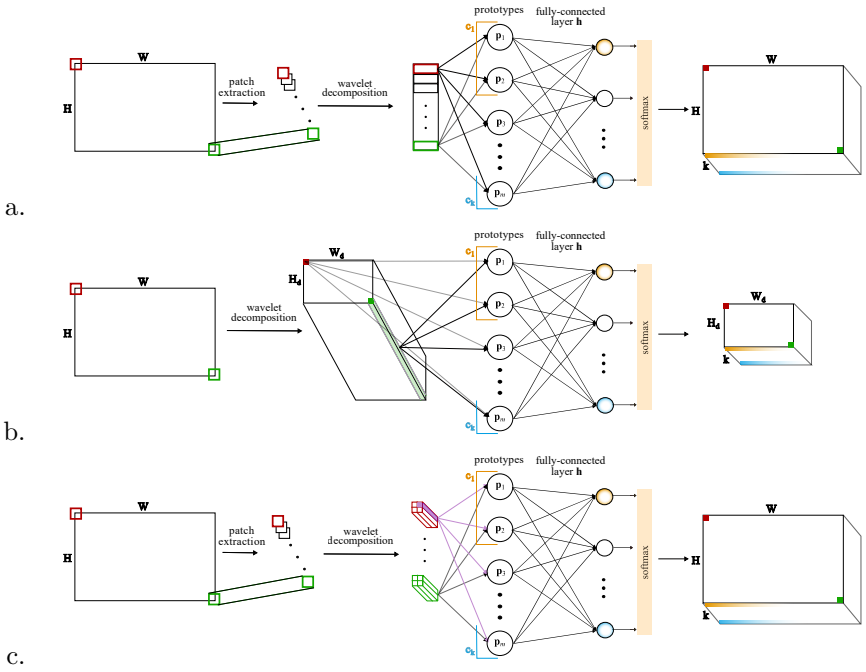

Figure 2. Different WaveProtoSeg builds: (a) patch extraction at the beginning and patch-sized prototypes, (b) patch extraction via wavelet decomposition; (c) pre-extracting patches and prototypes smaller than patch-size.

In WaveProtoSeg, image feature extraction is performed using wavelet decomposition – the different applications of it will be explained in detail in each three setups. This operation produces a set of feature maps from the input image x – denoted as $\phi(x) = z$ – which are then passed to the prototype layer. In case of *c.setup*, this will be split into smaller patches: $\tilde{z} \in \mathcal{P}(z) \equiv \text{patches}(z)$. The output produced by the j^{th} prototype unit (p_j) can be written as:

$$g_{p_j}(z) = \max_{\tilde{z} \in \mathcal{P}(z)} \log\big(\big(||\tilde{z} - p_j||_2^2 + 1\big)/\big(||\tilde{z} - p_j||_2^2 + \epsilon\big)\big).$$

Based on the similarity scores between the prototypes and the feature maps, the

classification will take place. After the classification of each pixel, the results will be gathered into one final result: the segmentation map of the original image. The prototype units themselves are also learnt from the wavelet-extracted features. The depth of wavelet decomposition, that is, the number of decomposition levels, is a tunable hyper-parameter.

In the first configuration, shown in Figure 2.a, each pixel in the input image is associated with a local patch. These patches are independently decomposed with wavelet transform until the average (low-frequency) component is reduced to a single pixel. The resulting feature map of each patch retains the same spatial dimensions as the original patch. The result is a vector of dimension $1 \times (H_p \cdot W_p)$, where $H_p \cdot W_p$ is the size of the patch. For an image x of size 32×32, this process results in 32×32 distinct patches – one for each pixel – that capture the local neighbourhood context. These feature vectors are then compared to the learnt prototypes in the prototype layer. Since this setup generates a prediction per input pixel, the output resolution matches the input. Conceptually, this is analogous to PrototypeDL [11], which learns entire images as prototypes; here, the prototype layer learns entire patches.

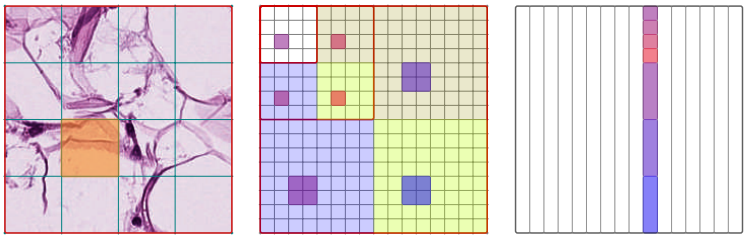

Figure 3. (left) Highlighted patch from the original image. (mid) Purple regions on the feature map indicate dispersed data of the patch. (right) Vector-form rearrangement of the feature map.

In the second configuration (*b.setup*), the entire image is wavelet-decomposed in one pass. Following decomposition, patches are extracted from the resulting feature map; see Figure 3. This significantly reduces the number of patches compared to *a.setup*, resulting in a smaller output resolution. For example, decomposing a 128×128 image may produce a 32×32 feature map, leading to 32×32 patches. Hence, the architecture in Figure 2.b provides fewer segmentation predictions, corresponding to the spatial dimensions of the decomposed map. This setup resembles the WaveProtoPNet approach, where images are decomposed into feature maps, and patches from these maps are used to learn and compare prototypes.

The third configuration (from Figure 2.c) shares structural similarities with *a.setup*, in that the patches are extracted before wavelet decomposition. However, the number and size of patches are user-defined hyper-parameters. Each patch is decomposed only up to a certain level, stopping before the approximation component reduces to a single pixel. This allows the average part to retain a spatial extent

of multiple pixels. For example, decomposing a 256×256 input into 16×16 patches and performing three levels of wavelet decomposition result in a 2×2 component for each 8×8 patch. These are flattened into $1 \times 1 \times 64$ feature vectors, which also define the prototype dimensions. This setup enables fine-grained semantic matching while maintaining computational efficiency. In this case, the prediction granularity of the model remains aligned with the input image. The inner part of this setup, after extracting patches is basically a WaveProtoPNet by structure – and its pixel-wise results will be gathered into one segmentation mask.

Despite differences in feature map construction and output resolution, all three configurations share the same prototype layer and fully connected classification layer. The size of prototype vectors varies depending on the setup, but the comparison logic remains consistent. The primary hyper-parameters include patch size, decomposition depth, and number of prototypes.

Loss functions

During training and evaluation, we adopted the loss formulation proposed in ProtoSeg [18], while also evaluating an alternative loss from our previous work, WaveProtoPNet. The training comprises two distinct phases: initially, it concentrates on prototype formation, and subsequently, it emphasizes classification accuracy while mitigating negative reasoning.

Let $D = \{(x_i, y_i)\} = [X, Y]$ denote a data set of images and labels.

In the first phase, when the focus is on prototype learning, the weights of the fully connected layer h are frozen, so that the edges that connect a prototype with the class for which they are responsible are set to 1, otherwise to -0.01:

$$w_h^{(k,j)} = \begin{cases} 1 & \text{if } p_j \in P_k \\ -0.01 & \text{otherwise,} \end{cases}$$

where $w_h^{(k,j)}$ denotes the weight of the edge connecting the j^{th} prtotoype to the class k, and P_k being the set of prototypes responsible for class k.

The loss responsible for prototype-formation from ProtoPNet includes:
Cluster Cost ($\mathcal{L}_{Clst}$), which encourages each prototype to be close to at least one patch of its corresponding class: $\mathcal{L}_{Clst} = \frac{1}{n} \sum_{i=1}^{n} \min_{p_j \in P_{y_i}} \min_{z \in \mathcal{P}(\phi(x_i))} ||z - p_j||_2^2$; Separation Cost (Sep), that promotes distance between prototypes and patches of different classes: $\mathcal{L}_{Sep} = -\frac{1}{n} \sum_{i=1}^{n} \min_{p_j \notin P_{y_i}} \min_{z \in \mathcal{P}(\phi(x_i))} ||z - p_j||_2^2$. The total loss, focusing on prototype formation, includes both the cross-entropy classification loss ($\mathcal{L}_{CE}$) and regularisation terms:

$$\mathcal{L}_1 = \mathcal{L}_{CE} + \lambda_1 \mathcal{L}_{Clst} + \lambda_2 \mathcal{L}_{Sep}$$

In contrast, the ProtoSeg loss function introduces a prototype diversity term based on Jeffrey's divergence. It will be written in a suitable form for *a.setup* and *b.setup*. This term ensures that prototypes of the same class are activated in different regions of the input image, promoting interpretability and coverage.

Jeffrey's divergence between the distributions U and V is defined as: $D_J(U,V) = \frac{1}{2}D_{KL}(U \parallel V) + \frac{1}{2}D_{KL}(V \parallel U)$, being the symmetrised version of the Kullback-Leibler divergence (D_{KL}). For multiple distributions $U_1, U_2, \ldots, U_n$, their similarity is computed as: $S_J(U_1, U_2, \ldots, U_n) = \frac{1}{C_2^n} \sum \exp(-D_J(U_i, U_j))$.

Given a prototype corresponding to class c: $p \in P_c$ and a feature map Z with the corresponding ground truth labels $Y_Z \in \mathbb{R}^{H_d \times W_d}$, the prototype-class-image activation vector is: $v(Z,p) = \text{softmax}(\parallel z_{ij} - p \parallel^2 | \; z_{ij} \in Z, Y_{ij} = c)$. The diversity loss for prototypes of class c is then: $L_J(Z, P_c) = S_J(v(Z,p_1), \ldots, v(Z,p_k))$.

The overall prototype diversity loss, averaged over all classes, is:

$$\mathcal{L}_J = \frac{1}{C} \sum_{c=1}^{C} L_J(Z, P_c),$$

Finally, the total prototype-formation loss used during training combines the cross-entropy loss with the diversity term:

$$\mathcal{L}_{1J} = \mathcal{L}_{CE} + \lambda_J \cdot \mathcal{L}_J,$$

where $\mathcal{L}_{CE}$ is the pixel-wise classification loss, and λ_J controls the weight of the diversity regularization.

The second loss in both cases is to focus on avoiding negative reasoning. They are mostly the same, with the difference that in the ProtoSeg loss, the prototype-formation loss is also included. In case of ProtoPNet loss, it looks as follows:

$$\mathcal{L}_2 = \mathcal{L}_{CE} + \lambda \sum_{k=1}^{K} \sum_{j, p_j \notin P_k} |w_h^{(k,j)}|.$$

In the ProtoSeg type, instead of the $\mathcal{L}_{CE}$ term, $\mathcal{L}_{1J}$ occurs.

4. Results and discussion

To evaluate the performance of the WaveProtoSeg model, we used a preprocessed version of the Cityscapes dataset [3], including 5000 images relevant to understanding the urban scene [14]. This data set provides high-resolution RGB street view images from various cities of size 128×256, annotated at the pixel level of the 20 classes. These fine annotations allow for a detailed analysis of semantic segmentation performance. The ProtoSeg model achieved 67% mIoU with this dataset.

We tested several variants of the model using different types of wavelets, decomposition levels, prototype numbers, and loss functions. Across all setups, we adopted the mean Intersection over Union (mIoU) metric to evaluate segmentation performance. The training included two phases. The first phase (responsible for prototype learning) has additional three learning components: first with a learning rate of 0.01 for 8 epochs, then a learning rate of 0.005 for 4 epochs, and finally with a learning rate of 0.001 for 4 epochs. The second phase (focusing on accuracy and

avoiding of negative reasoning) has also three sub-phases, with the same learning rates as in the first phase, just with different epoch numbers per each: it runs primary for 8 epochs, then for 6, then again for 8 epochs. As loss hyper-parameters the following values were used: $\lambda_1 = 1.25$, $\lambda_2 = 0.4$, $\lambda = 0.001$, $\lambda_j = 0.25$.

In the following, we present a summary of the most successful configurations of each, illustrated in Table 1. The best overall result was achieved with the *c.setup*, where dilation of convolutional filters was employed at the initial patch extraction. This led to a mIoU of 39.21% in the test set and 41.19% on the train set. Here, the loss from WaveProtoPNet was used. The second most successful setup was the *a.setup*, which is similar to the *c.setup*, with the difference that here the patches are not decomposed into smaller patches. More prototypes were needed to achieve the above 38% mIoU, using the loss of ProtoSeg, and as a consequence, training was much more time consuming. The model with the worst performance was *b.setup*, achieving the maximum test mIoU of 33.86%, applying the loss of ProtoPNet during training.

Table 1. Summary of the best results for each setup.

Setup	Receptive field of a pixel	Decomp	Wavelet	Proto/ Class	Test mIoU
a.setup	8×8	3	db2	20	38.2
b.setup	10×10	1	db4	20	33.86
c.setup	16×16	1	db4	5	**39.21**

Despite the architectural novelty, the WaveProtoSeg model underperformed compared to state-of-the-art semantic segmentation models. Figure 4 shows a visual comparison of input images, ground truth annotations, and predictions from the WaveProtoSeg model. It is evident that some classes such as *building*, *sky* or *vegetation* were learnt better, while others like *traffic light* or *human* suffered from inconsistent predictions and object-level confusion.

There are several potential causes that could contribute to the observed lower mIoU level. One potential explanation for this phenomenon is that certain classes are significantly under-represented when compared to others. The majority of images are covered with *road*, *building*, *sky* and *vegetation*, while the rest of the classes occur just occasionally, sometimes just in a really small part of the image. Another reason should be the low quality of the prototypes. As shown in Figure 5, the learnt prototypes lack semantic structure and often do not represent the meaningful features of the training data. This can be the result of the previously mentioned problem of class-imbalance. The features extracted by wavelet decomposition can also be a problem. The wavelets may not be suitable for extracting the most relevant information from this type of data. Finally, an essential drawback can be the small receptive field that is used to classify a pixel. By expanding the area considered around a single pixel, the amount of contextual semantic information increases, which can significantly enhance the precision of its classification process.

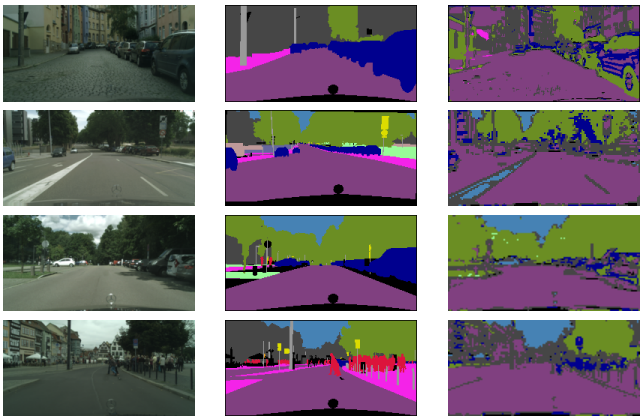

Figure 4. Top to bottom: original images, ground-truth annotations, predicted segmentation.

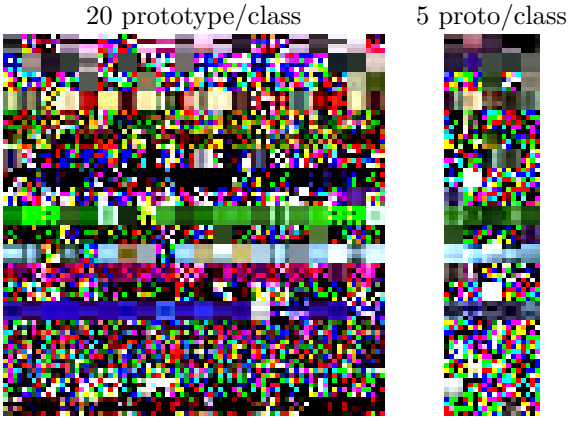

Figure 5. Visualization of learned prototypes: their low quality and poor discriminative power directly affect the model's accuracy.

5. Conclusion

In our previous research, we demonstrated that wavelets are as powerful in feature extraction as any classical backbone. Our experiments provided a thorough basis for including wavelet decomposition into prototype-based explainable systems. However, after several trials and experiments, we arrived at the conclusion that wavelet-based image segmentation remains backward compared to traditional backbone systems in terms of accuracy and interpretability.

Although the WaveProtoSeg model offers an explainable approach to semantic segmentation by combining prototype learning with wavelet-based feature ex-

traction, its performance fell significantly short of expectations on the Cityscapes dataset. The best test mIoU achieved was only 39.21%, suggesting that the current architecture and training methodology require refinement before it can be considered competitive.

The small receptive field of the model hinders its ability to make context-aware decisions. Since a pixel is classified based on features extracted from a relatively narrow patch, it lacks the broader contextual information that is often crucial in distinguishing between semantically similar regions (e.g., distinguishing a car from a bus, or a road from a sidewalk). Enlarging the receptive field may help mitigate this problem; however, this could prove to be prohibitive with respect to the number of prototypes that would be required, making the training process too expensive.

To thoroughly investigate if the model's architectural constraints are influenced by particular datasets, we intend to conduct experiments across various datasets. This includes testing on a more balanced dataset with a smaller number of distinct classes. Another direction would be to test it on a medical data set. The prior model we developed, WaveProtoPNet, demonstrated strong performance in the classification of human tissues. Inspired by these results, a worthwhile endeavour would be to investigate whether good segmentation capabilities can be achieved for medical data sets, such as the histology data set for nuclei segmentation [5]. We think that the "simpler" images – which can be converted to gray-scale without information loss – from the medical domain of the cellular level can be more convenient for prototype-based segmentation. Using these simpler data, one could experiment with the possibility of making the prototypes *rotation-invariant* by manually generating the rotated versions of the prototypes for every single learnt prototype.

In conclusion, while the model promotes explainability and novelty, substantial architectural and algorithmic improvements are necessary for it to be a viable tool for semantic segmentation in the natural or medical imaging domains.

References

[1] V. Badrinarayanan, A. Kendall, R. Cipolla: *SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation*, 2016, arXiv: 1511.00561 [cs.CV].

[2] C. Chen, O. Li, A. Barnett, J. Su, C. Rudin: *This looks like that: deep learning for interpretable image recognition*, in: NIPS 33, Curran Ass., 2018.

[3] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, B. Schiele: *The Cityscapes Dataset for Semantic Urban Scene Understanding*, in: Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

[4] D. Feng, C. Haase-Schütz, L. Rosenbaum, H. Hertlein, C. Gläser, F. Timm, W. Wiesbeck, K. Dietmayer: *Deep Multi-Modal Object Detection and Semantic Segmentation for Autonomous Driving: Datasets, Methods, and Challenges*, IEEE Transactions on Intelligent Transportation Systems 22.3 (2021), pp. 1341–1360, DOI: 10.1109/TITS.2020.2972974.

[5] J. Gamper, N. A. Koohbanani, K. Benes, A. Khuram, N. Rajpoot: *PanNuke: an open pan-cancer histology dataset for nuclei instance segmentation and classification*, in: European Congress on Digital Pathology, Springer, 2019, pp. 11–19.

[6] J. Gao, B. Wang, Z. Wang, Y. Wang, F. Kong: *A wavelet transform-based image segmentation method*, Optik 208 (2020), p. 164123, ISSN: 0030-4026, DOI: 10.1016/j.ijleo.2019.164123.

[7] R. Gipiškis, C.-W. Tsai, O. Kurasova: *Explainable AI (XAI) in image segmentation in medicine, industry, and beyond: A survey*, ICT Express 10.6 (2024), pp. 1331–1354, ISSN: 2405-9595.

[8] R. Gipiškis, C.-W. Tsai, O. Kurasova: *Explainable AI (xAI) in Image Segmentation in Medicine, Industry, and Beyond: A Survey*, 2024, arXiv: 2405.01636 [cs.CV].

[9] S. S. Y. Kim, N. Meister, V. V. Ramaswamy, R. Fong, O. Russakovsky: *HIVE: Evaluating the Human Interpretability of Visual Explanations*, 2022, arXiv: 2112.03184 [cs.CV].

[10] T. Kohonen: *Self-Organization and Associative Memory*, Third, New York, NY: Springer-Verlag, 1989.

[11] O. Li, Liu, et al.: *Deep Learning for Case-Based Reasoning Through Prototypes: A Neural Network That Explains Its Predictions*, in: Proc. of AAAI Conf. on Artificial Intelligence, vol. 32, 2018.

[12] H.-G. Lieb, T. Kaszta, L. Csató: *On Background Classes in Prototype-Based Architectures*, Acta Universitatis Sapientiae, Informatica 17.1 (2025), p. 6.

[13] H.-G. Lieb, T. Kaszta, L. Csató: *Wavelet-Based Prototype Learning for Medical Image Classification*, in: 2024 IEEE 22nd Jubilee International Symposium on Intelligent Systems and Informatics (SISY), 2024, pp. 000631–000636, DOI: 10.1109/SISY62279.2024.10737523.

[14] S. Liu, E. Johns, A. J. Davison: *End-to-End Multi-task Learning with Attention*, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 1871–1880.

[15] S. Mallat: *A Wavelet Tour of Signal Processing*, Elsevier, 2009.

[16] O. Ronneberger, P. Fischer, T. Brox: *U-Net: Convolutional Networks for Biomedical Image Segmentation*, 2015, arXiv: 1505.04597 [cs.CV].

[17] C. Rudin, C. Chen, Z. Chen, H. Huang, L. Semenova, C. Zhong: *Interpretable machine learning: Fundamental principles and 10 grand challenges*, Statistics Surveys 16 (2022), pp. 1–85, DOI: 10.1214/21-SS133.

[18] M. Sacha, D. Rymarczyk, Ł. Struski, J. Tabor, B. Zieliński: *ProtoSeg: Interpretable Semantic Segmentation with Prototypical Parts*, 2023, eprint: 2301.12276 (cs.CV).

[19] W. Samek, G. Montavon, S. Lapuschkin, C. Anders, K.-R. Muller: *Explaining Deep Neural Networks and Beyond: A Review of Methods and Applications*, Proceedings of the IEEE 109 (Mar. 2021), pp. 247–278, DOI: 10.1109/JPROC.2021.3060483.

[20] R. R. Selvaraju, A. Das, R. Vedantam, M. Cogswell, D. Parikh, D. Batra: *Grad-CAM: Why did you say that?*, 2017, arXiv: 1611.07450 [stat.ML].

[21] R. Strudel, R. Garcia, I. Laptev, C. Schmid: *Segmenter: Transformer for Semantic Segmentation*, 2021, arXiv: 2105.05633 [cs.CV].

[22] S. A. Taghanaki, K. Abhishek, J. P. Cohen, J. Cohen-Adad, G. Hamarneh: *Deep Semantic Segmentation of Natural and Medical Images: A Review*, 2024, arXiv: 1910.07655 [cs.CV].

[23] Y. Tian, L. Lan, H. Guo: *A review on the wavelet methods for sonar image segmentation*, International Journal of Advanced Robotic Systems 17 (Aug. 2020), p. 172988142093609, DOI: 10.1177/1729881420936091.

[24] K. Vinogradova, A. Dibrov, G. Myers: *Towards Interpretable Semantic Segmentation via Gradient-Weighted Class Activation Mapping (Student Abstract)*, Proceedings of the AAAI Conference on Artificial Intelligence 34.10 (Apr. 2020), pp. 13943–13944, ISSN: 2159-5399, DOI: 10.1609/aaai.v34i10.7244.

[25] K. Wickstrøm, M. Kampffmeyer, R. Jenssen: *Uncertainty and interpretability in convolutional neural networks for semantic segmentation of colorectal polyps*, Medical Image Analysis 60 (Feb. 2020), p. 101619, issn: 1361-8415, doi: 10.1016/j.media.2019.101619.

[26] G. Xu, W. Liao, X. Zhang, C. Li, X. He, X. Wu: *Haar wavelet downsampling: A simple but effective downsampling module for semantic segmentation*, Pattern Recognition 143 (2023), p. 109819, issn: 0031-3203.

[27] Y. Zhou, H. Jiaxing, C. Wang, L. Song, G. Yang: *XNet: Wavelet-Based Low and High Frequency Fusion Networks for Fully- and Semi-Supervised Semantic Segmentation of Biomedical Images*, in: 2023 IEEE/CVF International Conference on Computer Vision (ICCV), Oct. 2023, pp. 21028–21039, doi: 10.1109/ICCV51070.2023.01928.