future internet

MDPI

Article

Deep Learning Based Semantic Image Segmentation Methods for Classification of Web Page Imagery

Ramya Krishna Manugunta, Rytis Maskeliūnas and Robertas Damaševičius *

Faculty of Informatics, Kaunas University of Technology, 51368 Kaunas, Lithuania
* Correspondence: robertas.damasevicius@ktu.lt

Abstract: Semantic segmentation is the task of clustering together parts of an image that belong to the same object class. Semantic segmentation of webpages is important for inferring contextual information from the webpage. This study examines and compares deep learning methods for classifying webpages based on imagery that is obscured by semantic segmentation. Fully convolutional neural network architectures (UNet and FCN-8) with defined hyperparameters and loss functions are used to demonstrate how they can support an efficient method of this type of classification scenario in custom-prepared webpage imagery data that are labeled multi-class and semantically segmented masks using HTML elements such as paragraph text, images, logos, and menus. Using the proposed Seg-UNet model achieved the best accuracy of 95%. A comparison with various optimizer functions demonstrates the overall efficacy of the proposed semantic segmentation approach.

Keywords: semantic segmentation; webpage analysis; deep learning

Citation: Manugunta, R.K.; Maskeliūnas, R.; Damaševičius, R. Deep Learning Based Semantic Image Segmentation Methods for Classification of Web Page Imagery. *Future Internet* **2022**, *14*, 277. https://doi.org/10.3390/fi14100277

Academic Editor: Filipe Portela

Received: 15 September 2022
Accepted: 22 September 2022
Published: 27 September 2022

1. Introduction

Image segmentation is the process of separating an image into several segments while assigning an item class to each pixel in the image [1]. The two basic types of image segmentation are semantic and instance segmentations. Semantic image segmentation is defined as "the assignment of class labels for each pixel of an image dependent on the class it belongs to" [2]. The segmentation of semantic data has been done using traditional approaches such as k-means clustering, thresholding, and histogram-based methods. It has taken deep learning years to push the limits and build alternative architectures that may considerably improve segmentation accuracy.

In semantic segmentation, a single class label is used to identify all items of the same type; however, in instance segmentation, related objects are given their own labels. Even though it is a pixel-level classification problem in the pixels of an image, it is a far more challenging technique than traditional classification, which seeks to predict the labels of the complete image on which we trained. Deep learning's enormous success has had a tremendous impact on semantic segmentation algorithms, greatly improving their accuracy. This prospective breakthrough has piqued the interest of several technological and academic sectors that require advanced computer vision skills [3].

Various visual tasks have been brought to the cutting edge of technology using deep convolutional neural networks (DCNNs). Deep neural networks are highly efficient at semantic segmentation or labeling each region or pixel with an object or non-object class. Semantic segmentation is necessary for image analysis tasks and plays a key role in image comprehension. It has numerous applications in computer vision and artificial intelligence (AI), including autonomous driving [4,5], robot navigation [6], industrial inspection [7]; rehabilitation [8]; document analysis [9]; remote sensing [10,11]; cognitive and computational sciences—saliency object detection [12]; agriculture sciences [13]; fashion—classifying clothing items [14]; social sciences—marketing [15] and consumer preference analysis [16]; medical sciences—medical imaging analysis [17–19], etc. Earlier approaches

for semantic segmentation included slope difference distribution [20], support vector machine (SVM) [21], and random-forest-based classifiers [22]; however, deep learning techniques have enabled segmentation that is more exact and substantially faster [23,24].

Convolutional neural networks (CNNs) offer cutting-edge solutions for a variety of issues in machine learning and computer vision, including image recognition [25,26], semantic segmentation [27,28], and object identification [29]. CNNs were first utilized in 1998 [30]. LeNet-5, the first CNN, could read handwritten numbers from pictures. Layers of various types, including convolutional, pooling, and fully connected layers, are present in CNNs. By stacking these layers, CNNs may automatically learn feature representations by applying local neighborhood pooling operations and trainable filters to the resulting feature maps and raw input pictures. This is distinct from conventional pattern recognition models, where relevant data is manually extracted by a feature extractor from the input and then categorized by a trainable classifier.

Semantic segmentation methods can be used to categorize web pages, in which website snapshots must comprehend their complete layout structure (html tags), such as p tags for paragraphs, <img> tags for images, menu tags for website buttons, and logo tags for website image logos, among others.

The main contributions of this study are as follows: (1) creating a custom data set containing details about the annotations on website screenshots (using LabelMe), (2) providing a comparative understanding of the state-of-the-art of various backbone deep learning methods used in the UNet architecture and FCN-8, and (3) providing an entire collection of the most effective approaches for creating semantic segmentation models with low latency and high accuracy in a consistent manner.

We demonstrate that a fully convolutional network (FCN), trained end-to-end and pixel-to-pixel on semantic segmentation, outperforms the state-of-the-art without the use of additional hardware. This is, to our knowledge, the first study to train FCNs end-to-end (1) for pixelwise prediction and (2) from supervised pre-training. Existing networks with fully convolutional versions predict dense outputs from inputs of arbitrary size.

The paper is structured as follows. Section 2 discusses related works. Section 3 describes the theoretical backdrops of the deep learning architecture, including state estimation problems, accompanying our proposal for improvement. Section 4 presents our experimental protocols and reports the numerical result with a discussion. Section 5 presents the discussion of the results. Section 6 wraps up the result and suggests future works.

2. Related Works

Fully convolutional networks (FCNs) may be used for image segmentation and localization; however, they need a large amount of training data. Due to the time-consuming nature of manually labeling images with various categories, few datasets with pixel-level labels are available to the public. When segmenting images, FCNs [31–33] and UNets [34–38] are often used to address these difficulties.

Chen et al. [31] developed a novel image segmentation network (R-Net). Even with mixed noises present in the input photos, the suggested R-Net model still performed well on low-resolution images. Three modules make up the planned R-Net model: a module for data augmentation, a module for developing the CNN architecture, and a module for building the loss function. Our R-Net model can extract the discriminative feature for clean edge segmentation in comparison to the current networks.

Li et al. [32] introduced an attention mechanism at the end of the FCN that can extract image depth features and increase segmentation accuracy. The proposed model is based on an improved fully convolutional network that also combines data pre-processing and extracts texture information of the biomedical images acquired by biomedical sensors using Gabor filters to deepen image texture. The stability and segmentation quality of the segmentation network are further enhanced by pre-processing the input and including the attention mechanism in the FCN. Studies have revealed that combining the attention

mechanism with a fully convolutional network improves segmentation accuracy and stability and brings segmentation closer to artificial segmentation.

Wang et al. [33] suggested using the UNet model, which has the best performance in the real-world forest fire segmentation scenario, with a better description of semantic details; however, the inference speed for a single instance is relatively slow, making it more suitable for application scenarios requiring high accuracy.

Sun et al. [34] suggested a UNet model based on several attentions (MA-UNet) with a residual encoder based on a straightforward attention module to enhance the backbone's capacity to extract fine-grained features. The semantic representation of the feature map is rebuilt using multi-head self-attention for the lowest level feature while providing fine-grained segmentation for various pixel categories. Then, to address the issue of multiple scales in various categories, they add downsampling to subdivide the target's feature sizes at various scales and use channel attention and spatial attention in various feature fusion stages to better fuse the target's feature information at various scales.

Kadry et al. [36] proposed using pre-trained CNN segmentation techniques such as SegNet, UNet, and VGG-UNet to more accurately separate leukocytes. According to the experimental findings, VGG-UNet outperformed SegNet and UNet in terms of performance. The VGG-result UNets are also validated against various hybrid segmentation techniques that have been documented in the literature.

Maqsood et al. [37] suggested a tumor segmentation system based on categorization, edge detection, and picture enhancement using fuzzy logic. Dual tree–complex wavelet transform (DTCWT) is utilized at various scale levels to identify the edge in the source pictures after the input images have been pre-processed using contrast enhancement and fuzzy-logic-based edge identification. The features are determined from degraded sub-band pictures, and they are further classified using a UNet CNN.

Meraj et al. [38] suggested a quantization-assisted UNet technique for segmenting breast lesions. UNet and quantization are the first and second segmentation steps. When using UNet-based segmentation to separate precise lesion regions from sonography pictures, quantization is a helpful tool. The separated lesions are then employed in the independent component analysis (ICA) approach to extract features, which are subsequently merged with deep automated features.

In summary, due to the remarkable precision of deep neural networks in detection and multi-class recognition tasks, semantic segmentation based on deep learning is a viable option for reaching this objective. Consequently, it is essential to enhance the design of segmentation models to produce architectures that can be executed with the requisite precision in real-time and are efficient. In this study, we investigate the fastest and most precise semantic segmentation designs.

3. Methodology

In addition to providing background knowledge on the semantic segmentation job, this section offers a technical overview of the technique. In doing so, the usage of CNNs in this webpage image classification task, as well as a CNN variant dubbed the Seg-UNet, will be discussed. In addition, task-specific metrics and loss functions will be investigated, along with how they affect the segmentations compared to standard evaluation metrics for classification.

3.1. Background: Image Segmentation Architectures

An encoder and a decoder form the foundation of the image segmentation architecture. The encoder uses filters to extract features from the image. The decoder oversees the production of the final output, which is often a segmentation mask containing the object's shape. When it comes to semantic segmentation, what we are trying to predict is an output mask, so this mask will have the same width and height as our input image, and the depth of this mask is going to be equal to the number of classes. This architecture, or a version of it, can be found in almost all architectures.

Fully convolutional networks (FCNs) are a variation of CNNs that has made considerable strides in image segmentation. The last fully connected layer of the CNN is converted into a convolution layer for the FCN. Next, a nonlinear filter is applied to the output vectors of each layer. The network may accept inputs of any size and generate outputs with the same spatial dimensions as the inputs. In addition, the classification network may provide a heatmap of the item class in question. Adding network layers and a spatial loss produces a machine that is effective for end-to-end dense learning. FCNs can efficiently learn to make dense predictions for per-pixel tasks such as semantic segmentation.

Dense feedforward computation and backpropagation are used to conduct learning and inference in the whole image. In-network upsampling layers allow for pixel-by-pixel prediction and learning in sub-sampled pooling networks. This solution is both asymptotical and efficient and eliminates the need for the complexities found in earlier research. Patch-wise training is inefficient compared to fully convolutional training. Our method does not employ pre- or post-processing complexities, such as superpixels [39], suggestions [40], or post-hoc refining by random fields or local classifiers [41].

Our model uses dense prediction by reinterpreting classification nets as completely convolutional and fine-tuning from their learned representations. In contrast, previous research has employed small convnets without pre-training supervision. When it comes to learning dense predictions, FCNs are more versatile since they can take in pictures of varied sizes, but they are also more efficient because of the upsampling that takes place in the network. An important feature of the FCN is its ability to retain spatial information from the input, which is essential for semantic segmentation since this job involves both localization and categorization. Convolutions with no padding are used to lower the resolution of an FCN's output picture, even if the input image may be of any size. These were developed to reduce the size of filters and the associated computational burden. Consequently, the output is coarse and has been decreased in size by the pixel stride ratio of the output units. There is an inherent contradiction between semantics and location in semantic segmentation: global information resolves the 'what' question, while local information resolves the question of 'where'.

The UNet architecture is a variant of the FCN that is often used for semantic segmentation. UNets have the advantage over regular CNNs in that their output includes both localization and classification. In this instance, localization denotes the assignment of a class to each pixel in an image. UNets are superior to FCNs because they need fewer training images and give more precise segmentations. By producing upsampling layers with various feature channels, the layers with a greater resolution may get context information. When working with satellite data, it is crucial not to sacrifice localization and contextual information. UNet is similar to FCN-8 in that it is an encoder and decoder model. Its original purpose was for the semantic segmentation of biomedical images, which is for detecting whether a cell is cancerous or non-cancerous; a lot of the same concepts in the UNet model have been carried over from the FCN-8 model, and there are just a few minor differences. The first one is, instead of using a skip connection, where we add in previous layers to other layers, like we do in FCN-8, here, we do copying and concatenation; hence, these are concatenation operations where we take some layers from the encoder and we just concatenate them onto some output of a decoder. We have convolution pooling, where we eventually go into this 1024 dimension that is very dense; it is like the bottleneck of the encoder, and we try to preserve the dimensionality between this concatenation connection. The whole network ends up looking like a U shape, but concatenation adds to the depth of the layers here, and, instead of making a straight-up sampled prediction, there are some additional convolutional layers here, within the decoder itself, before the additional upsample convolution occurs; this is another difference between UNet and FCN-8. Because one has to go through two convolutional layers before you upsample, you are kind of making a further inference based on this concatenated information and not necessarily having the information added in. We pool and upsample it by a factor of two every single time; hence, in the final prediction mask, we are not using an 8×8 filter as we do with

FCN-8, so it has to be upsampled 8 times, which produces core segmentation, which is where we get those blocks from. With UNet, we can get much more fine segmentation because everything is just downsampled and upsampled by a factor of 2. In the final layer, we upsample and perform another concatenation and some convolution, and then we use that final 1×1 convolution to create the segmentation mask.

3.2. Proposed Methodology

3.2.1. Semantic Segmentation Architecture

In this paper, we describe a skip architecture to leverage this feature spectrum, which combines deep, coarse semantic information and shallow, fine appearance information.

We used a modified version of the Seg-UNet approach, as suggested by Abdollahi et al. [42]. The method, displayed in Figure 1, keeps efficient information while reducing data processing and enables convolutional features to have spatial homogeneity.

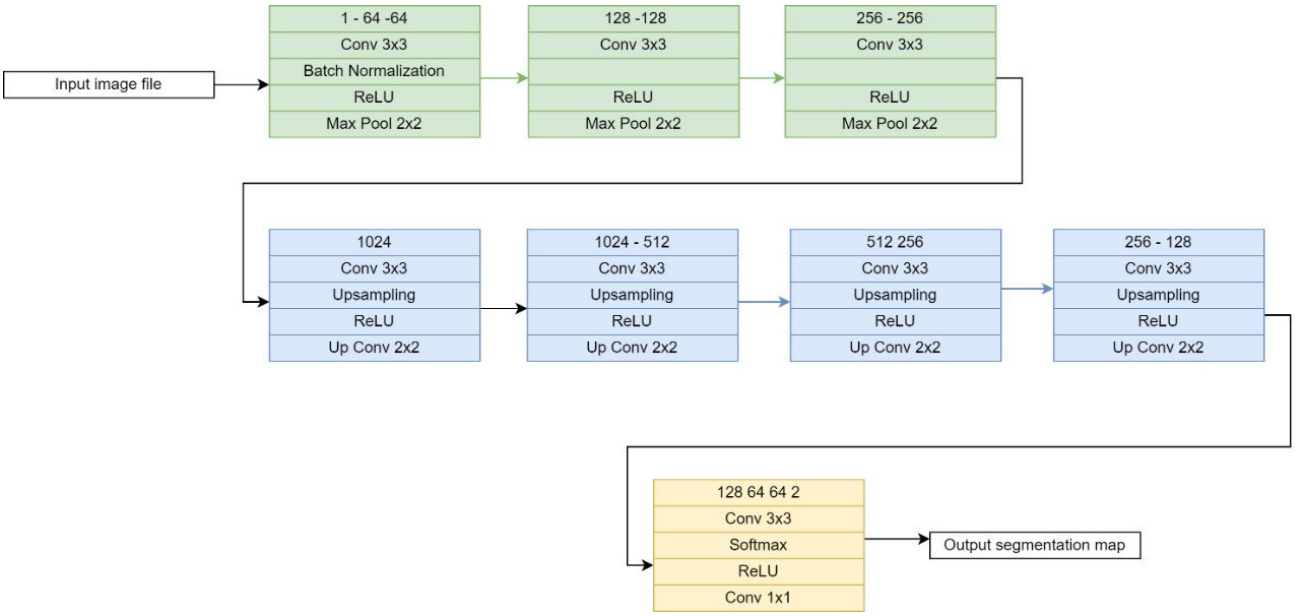


Figure 1. The proposed architecture for the semantic segmentation of webpage images.

Every encoder in the encoder network implements a filter bank with a convolution for creating and batch-normalizing a set of feature maps. This is followed by a contracting route consisting of three repeated convolution layers with 3×3 window sizes, followed by an upsampling layer with 2×2 window sizes. In the convolution process, the transformation function is employed. We empirically determined that ReLU works better in our circumstances. The resulting output is sub-sampled by a factor of two. Max-pooling layers are used to provide translation invariance over minor spatial changes in the input data. Although numerous map-pooling layers can provide extra translation invariance for strong classification, feature map spatial resolution suffers as a result.

Each encoder layer is linked to the matching decoder layer to send local contextual information into the decoder component. In contrast to the UNet approach, the same padding is used instead of valid padding. At the final decoder block, a dedicated convolution layer is used to categorize each pixel and generate the segmentation map. The cross-entropy loss function is also used to measure the difference between two possibility spreads, helping to avoid over-fitting since data normalization uses the batch normalization layer, which is placed after the convolutional layer.

3.2.2. Multi-Classification Masking

In terms of multi-classification, every single class, including the background, is defined. The labels are defined as triangle, pentagon, hexagon, and custom shapes for images in the dataset. This is the order followed for stacking up the number of channels, i.e., the label shapes, so now our target matrix will have four classes, as follows: triangle—logo; pentagon—paragraph text; hexagon—header menu; custom—images (Figure 2).

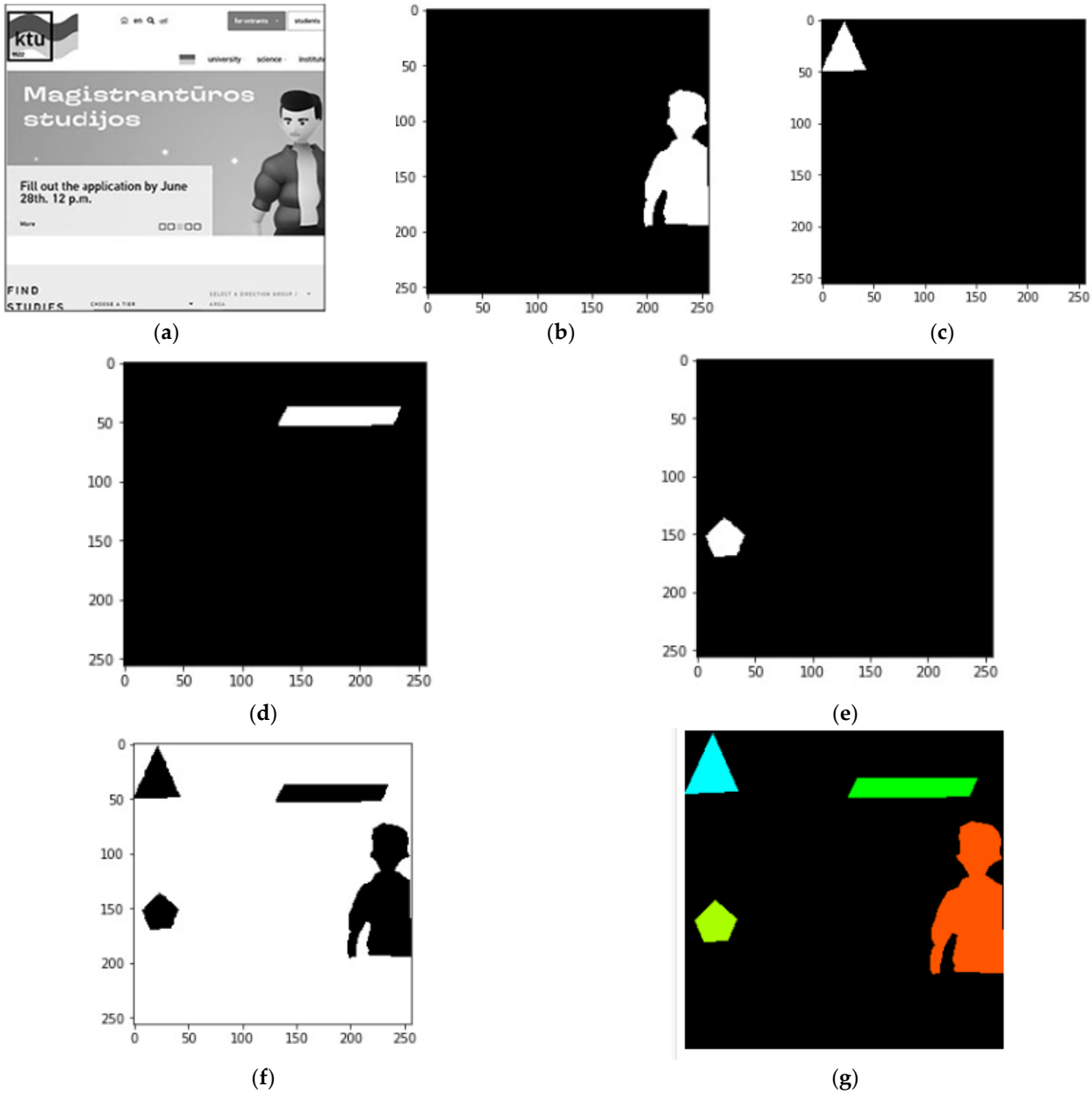


Figure 2. Multi-classification image with respect to labels: (**a**) webpage screenshot image, (**b**) segmented image, (**c**) segmented logo, (**d**) segmented header, (**e**) segmented paragraph text, and (**f**) all defined labels together, with background in white, and (**g**) all labels in the segmented RGB image.

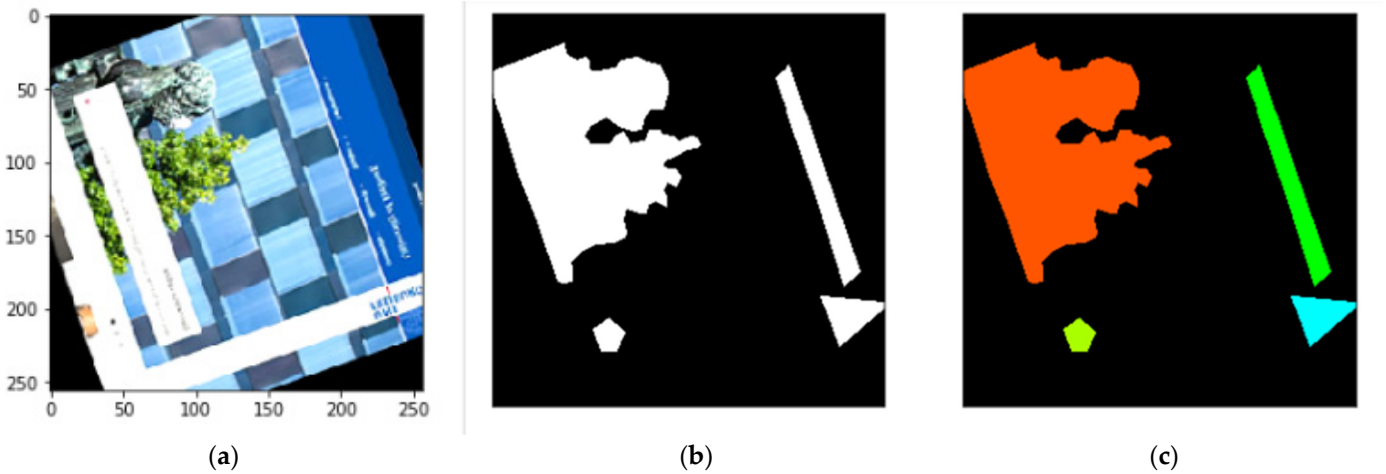
We take the final created target matrix, which is just all our label classifications in grayscale. In this part, we change the five grayscale channel images into channels that represent color so that we can draw in a single RGB image.

3.2.3. Image Augmentation

We augment the key points that are going to be creating the polygons to make the masks on our image. Because we are working with such a small number of images and because annotating the images is very expensive in terms of time, we would like to be able to augment the default dataset such that every single new batch that is sent over to the machine learning algorithm is seen as a slight variation of data every single time. The data is going to be continuously changing, but this will allow our algorithm to reach a more general solution and generalize to unseen data better than creating a fragile solution that will break because it has only been trained on 210 images.

There are a couple of ways to go about image augmentation; for example, keras has an image data generator class that can easily be built into a pipeline, and this is great for standard image classification tasks. However, when we start to deal with more complicated data, such as key points, bound boxes, and image masks, we use imgaug (an image augmentation library). The image augmentations we have used are brightening, blur, and rotation, which are generally encountered in the natural environment where we do a lot of machine learning.

We focus on key point augmentation, which we want to augment so that an image that is a little bit brighter and shrunk down can be seen; all the key points also move in correspondence with how the image is augmented. That is what we are really aiming for. We used augmentation to the key points instead of just drawing the mask (Figure 3) as storing an entire extra image that has a mask drawn over the original image increases the amount of disk space that is not scalable in terms of data collection.



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

Figure 3. (**a**) Augmented (rotated) image, (**b**) grayscale masked output, and (**c**) label-colored multi-color masked output.

We started with a created function called aug poly that takes in our original image path and the annotation path. Then, we created an index to keep track of the points that we are augmenting so that when we augment this, there are multiple label shapes that we do the augmentation to. The problem we see here is that when we receive the augmented key points, we really need to know we are going to get a whole bunch of different numbers because it will just index the key points further and further; we need to be able to recover the index for each label that we are interested in.

Hence, when we have the sequential, horizontal flip, and blur images in order, it will make sure that the augmentation happens on the source image and the annotation portion; it must do a random augmentation that randomly shuffles the sequential class so that different output images of blur, horizontal flip, and rotate will be obtained, as below.

4. Results and Discussion

In this section, the comparison and assessment results of segmentation models based on deep learning for multi-class labeled pictures are briefly described. For the experiments, we used the self-collected dataset of webpage images. The section is divided into three sections: the first section gives information on hyperparameters, the second section describes the visualization results of the segmentation models, and the third section provides an assessment of the segmentation models' accuracy-based performance.

4.1. Input Data and Hyperparameters

In deep-learning-based approaches, the hyperparameters (Table 1) are crucial. The learning rate is the most important parameter in the CNN; it is the number of steps done during training or while finding the local minimum of gradient descent. Using the stochastic gradient descent (SGD) approach, it is set to 0.0001 using the optimal experimental procedure of lowering it to its minimum and increasing it. The entropy is used for calculating the cost function or loss cross. Minibatch size is connected to the learning rate since setting the minibatch size to a low value causes gradient instability.

Table 1. Hyperparameters used for Seg-UNet.

Learning Rate	0.0001 Adam
Momentum	0.9
Regularization	L2norm
Cost/Loss	Cross-entropy
Batch size	5
Dropout	50%
Epoch	500 epochs
Decay Rate	0.9

Using the 210-image internet image dataset that is augmented with augmentation data, the proposed models are trained to recognize more than four unique item classes. This trained model will be evaluated concurrently to see how effectively it can learn the semantic segmentation of website images.

4.2. Analysis of UNet and FCN-8 Model Results

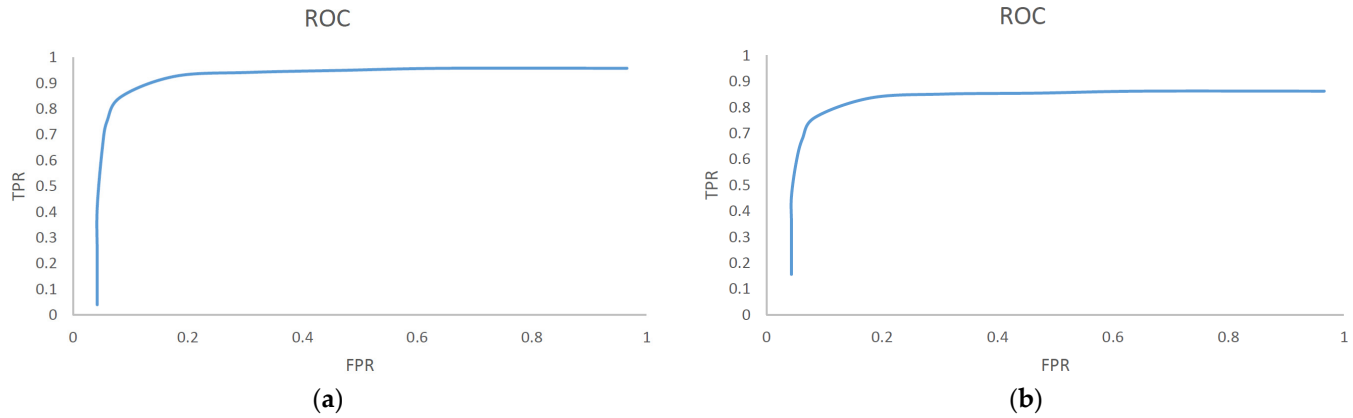
Utilizing L2norm and a momentum of 0.9, the stochastic gradient descent optimizers SGD, Adam, and Adagrad were able to avoid over-fitting. The whole network is trained with 500 epochs. A 50% dropout rate is utilized. The approach with this hyperparameter yielded 95% mean average accuracy and a 20% miss detection rate. The average mean accuracy is just a fraction of the true positive rates. Using L2norm and a momentum of 0.9, the stochastic gradient descent optimizers SGD, Adam, and Adagrad were able to avoid over-fitting. The whole network undergoes 500 iterations of training. We used a 50% dropout rate. The approach using this hyperparameter achieved an average mean accuracy of 95%.

To further improve the performance of segmentation models, training is done utilizing the data set. The above graph depicts the training accuracy and training loss curves for the segmentation models used in this study. Using cross-entropy and loss functions, as mentioned above, a result is produced (Table 2). It demonstrates that the UNet accuracy curves are marginally superior to those of the FCN model. Similarly, the UNet loss curves were reduced, indicating that it converges more readily.

Table 2. Evaluation of the results.

Model	Precision	Recall	F1-Score	IoU	mIoU
UNet	75%	93%	82%	85%	83%
FCN-8	63%	91%	77%	84%	81%

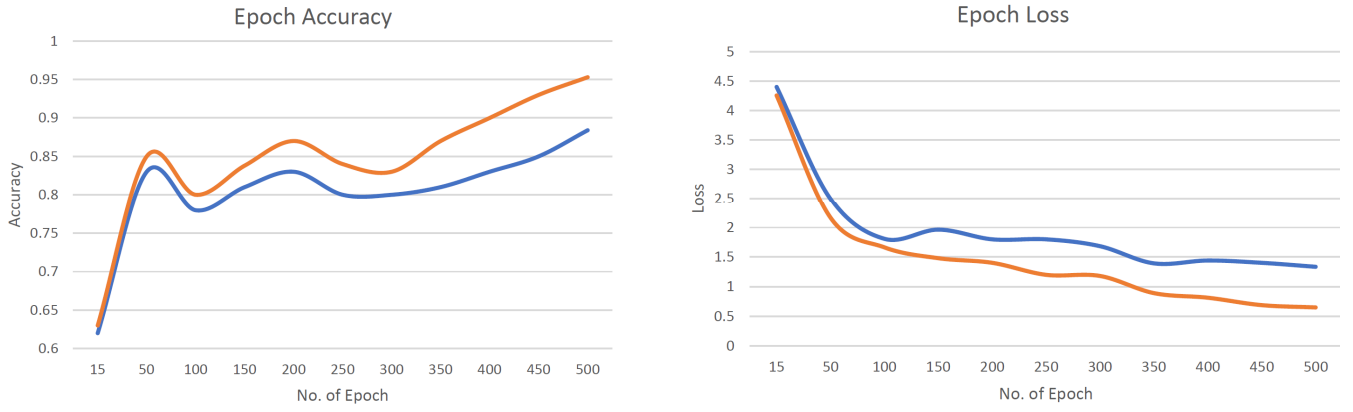
The performance of trained models is improved or increased in comparison to the performance of the pre-trained models, as demonstrated in the metrics (Figure 4a,b).



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

Figure 4. ROC graph of the (**a**) UNet model and (**b**) FCN-8 model.

The validation loss follows the training loss despite the noise, indicating that the model can learn the dataset. This is followed by the epoch accuracy, dice, and loss of the trained models. According to the results, the UNet model shows an accuracy of 95%, respectively, with different epoch range counts (Figure 5a,b).



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

Figure 5. Epoch accuracy graph (**a**) and loss graph (**b**) of UNet (red line) and FCN-8 (blue line).

The proposed model has been optimized in order to compare three different optimizers with different epoch range counts.

(1) Epoch count—300, Optimizer—Adam

From Figure 6, we can see that the training curve of the model gradually rises from epoch 10, though the given epoch count is in the 500 range. The accuracy of 95% is achieved by the 300 epoch count.

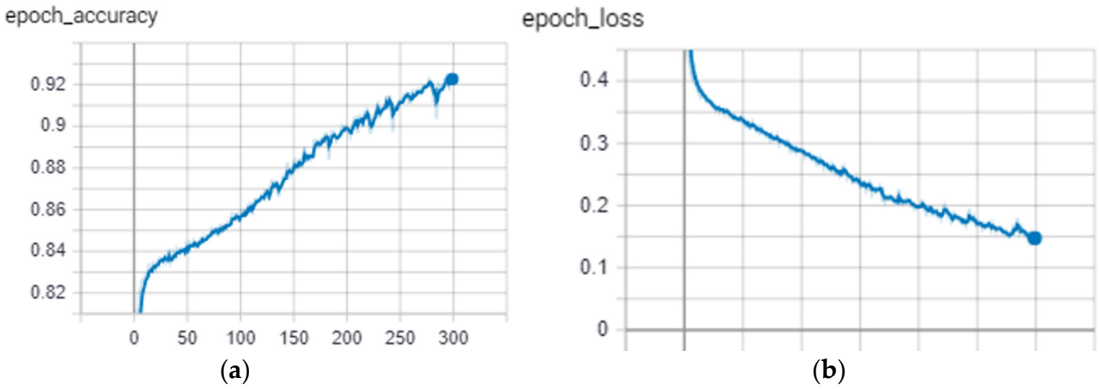


Figure 6. (**a**) Epoch accuracy (**a**) and loss (**b**) of the Adam optimizer with 300 epoch count.

(2) Epoch count—50, Optimizer—Adagrad

From Figure 7, we can see that the training curve of the model starts to get flat from the 30 range, similar to the loss curve.

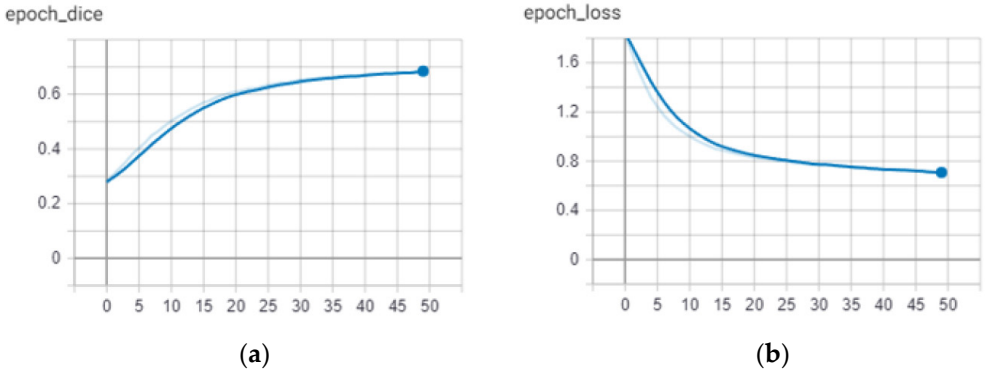


Figure 7. Dice score (**a**) and loss (**b**) of the Adagrad optimizer for the epoch count of 50.

(3) Epoch count—50, Optimizer—SGD

From Figure 8, the SGD optimizer shows that there are no gradual but only inconsistent levels in the dice and loss scalar graphs.

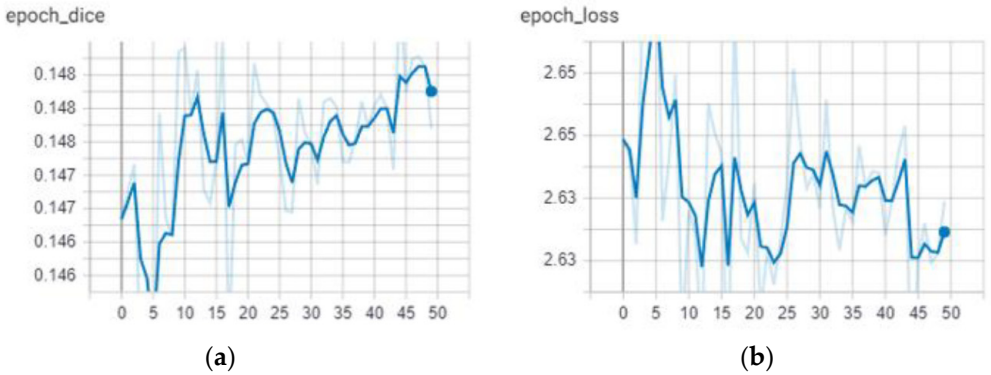


Figure 8. Dice score (**a**) and epoch loss (**b**) of the SGD optimizer for the epoch count of 50.

FCN-8 has shown worse results compared to UNet. We had to run for 500 epochs to achieve 88% accuracy in the training of the model for the desired output. The training curves show that there is a gradual increase from 100 epochs, with minimal loss (Figure 9).

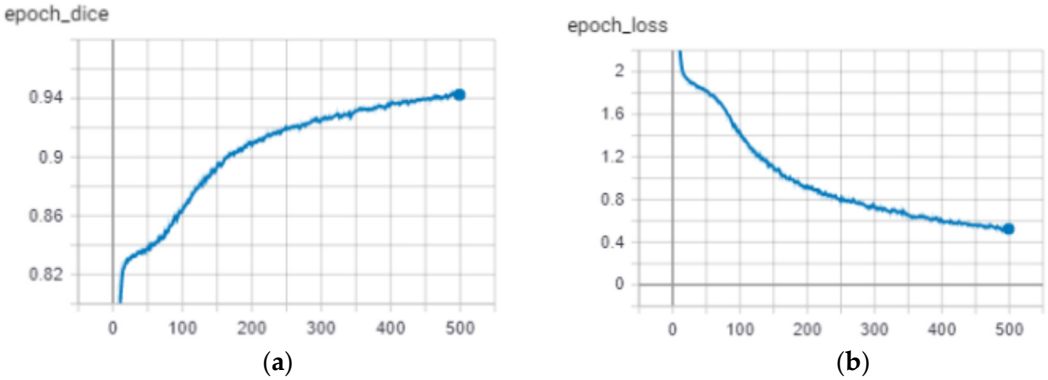


Figure 9. Dice score (**a**) and loss (**b**) for the proposed model with the Adam optimizer for the epoch count of 500.

The training curves explain that the SGD optimizer has continuous loss and gain in terms of training the model with the data (Figure 10).

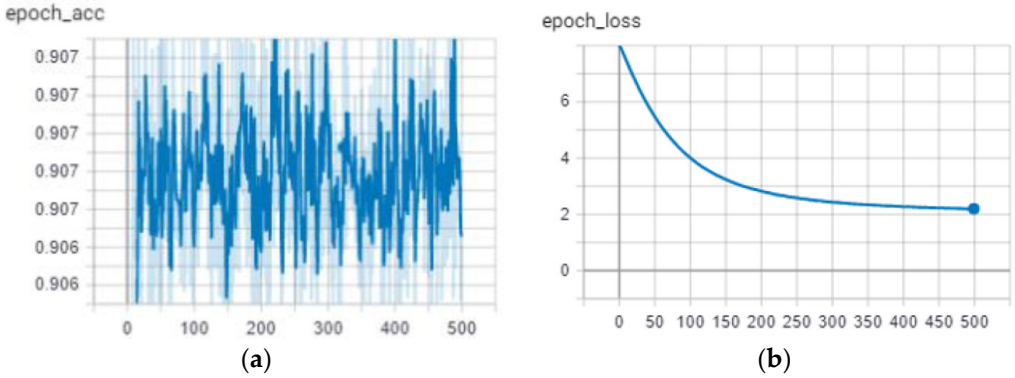


Figure 10. Accuracy (**a**) and loss (**b**) of the SGD optimizer for the proposed model for the epoch count of 500.

The training curves explain that the Adagrad optimizer had a flat accuracy gain from the epoch 300 to 500 range, similar to the loss (Figure 11).

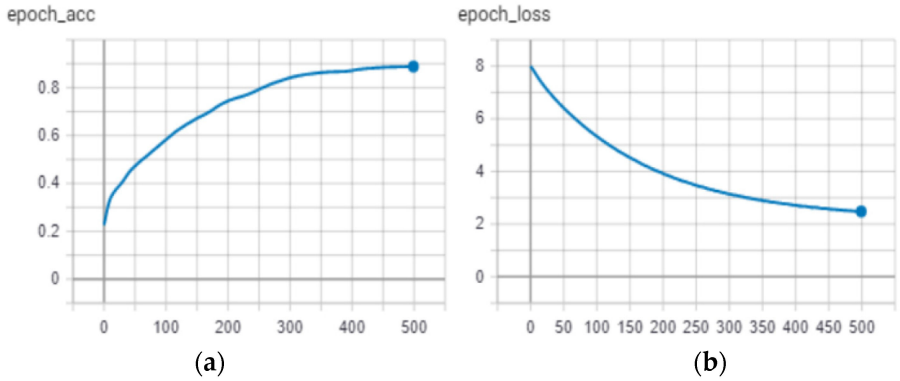


Figure 11. Accuracy (**a**) and loss (**b**) for the proposed model with the Adagrad optimizer for a 500 epoch count.

The masked image output of the proposed model with the Adam optimizer for 500 epochs is shown. Even for the model trained for 500 epochs, one can still see the pixel map resolution compared to the UNet model. After careful evaluation using three different optimizers, the best-predicted mask results were achieved with the Adam optimizer for the

300 epoch range for both UNet and FCN-8, whereas the other two optimizers produced results that were not even close enough to consider. Note that the Adam optimizer achieved accurate enough results.

5. Discussion

Our findings demonstrate that both UNet and FCN models offer high segmentation performance, particularly for webpage images that completely match the segmentation structural assumptions. The highest accuracy was attained using the proposed Seg-UNet model, at 95%. The suggested semantic segmentation method's overall effectiveness is demonstrated by a comparison with different optimizer functions (Figures 6–11).

There has been more effort made to improve the visual comprehension of online pages. In many instances, the document object model (DOM) tree is used to gain visual evidence rather than a picture of the displayed page [43]. This is difficult to generalize to other systems and does not have implementation independence. A system that uses images of webpages has been described by Sanoja Vargas et al. [44], who proposed a block-based conceptual model for segmentation that takes the content as well as the geometry of blocks into account to evaluate the correctness of a segmentation according to a predefined ground truth. Our suggested webpage segmentation mechanism is more complete since it is independent of the technology used for implementation. The results of our studies demonstrate that the method is effective in terms of expansibility and performance for segmenting web pages.

A high-quality segmentation of a web page can facilitate higher-level parsing of the page based on its visual features; this has significant applications, especially in the inspiring application field of assistive technology. Magnification and clearing are two uses for segmentation. Users who suffer from cognitive disorders such as attention deficiencies or older users may benefit from decluttering. Users with vision impairments may find benefits through magnification.

6. Conclusions

This study aims to train its own dataset variants using the proposed Seg-UNet model for semantic segmentation applications using a self-collected webpage imagery dataset and to compare them against UNet and FCN-8. This new dataset does not provide sufficient evidence for networks to learn the invariances between the two. Even though these networks were unable to attain state-of-the-art results and successfully classify the custom dataset, the performance of these models may serve as a benchmark for others to use when doing similar segmentations.

In terms of performance, when comparing the two models on a dataset that had been constructed, the segmentations produced by the UNet model performed better than those produced by the FCN model. The Seg-UNet model has also shown that it is an excellent tool for providing a comparison baseline, even if developing model architectures that are tailored specifically for a given dataset will result in higher dice and accuracy ratings. It is anticipated that in the future, the Seg-UNet architecture will be adopted as the standard for this work since it can provide appropriate prediction masks for classes that include a different range of characteristics.

In future work, retraining using labeled data may be beneficial. Annotated training data are expensive. The database should be expanded. Three-dimensional models and other computer-generated graphics may offer different views.

Author Contributions: Conceptualization, R.M.; methodology, R.M.; software, R.K.M.; validation, R.K.M., R.M. and R.D.; formal analysis, R.K.M., R.M. and R.D.; investigation, R.K.M. and R.M.; resources, R.M.; writing—original draft preparation, R.K.M.; writing—review and editing, R.M. and R.D.; visualization, R.K.M.; supervision, R.M.; funding acquisition, R.D. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data is available from the corresponding author upon reasonable request.

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

References

1. Garcia-Garcia, A.; Orts-Escolano, S.; Oprea, S.; Villena-Martinez, V.; Martinez-Gonzalez, P.; Garcia-Rodriguez, J. A survey on deep learning techniques for image and video semantic segmentation. *Appl. Soft Comput. J.* **2018**, *70*, 41–65. [CrossRef]
2. Wang, Z.; Cui, J.; Zha, H.; Kagesawa, M.; Ono, S.; Ikeuchi, K. Foreground object detection by motion-based grouping of object parts. *Int. J. Intell. Transp. Syst. Res.* **2014**, *12*, 70–82.
3. Bhatt, D.; Patel, C.; Talsania, H.; Patel, J.; Vaghela, R.; Pandya, S.; Ghayvat, H. Cnn variants for computer vision: History, architecture, application, challenges and future scope. *Electronics* **2021**, *10*, 2470. [CrossRef]
4. Butt, M.A.; Riaz, F. CARL-D: A vision benchmark suite and large scale dataset for vehicle detection and scene segmentation. *Signal Processing Image Commun.* **2022**, *104*, 116667. [CrossRef]
5. Li, X.; Yun, X.; Zhao, Z.; Zhang, K.; Wang, X. Lightweight deeplearning method for multi-vehicle object recognition. *Inf. Technol. Control.* **2022**, *51*, 294–312. [CrossRef]
6. Yao, Y.; Cai, Y.; Wei, W.; Farisi, Z. Semantic scene segmentation for indoor robot navigation via deep learning. In Proceedings of the 3rd International Conference on Robotics, Control and Automation, ICRCA, New York, NY, USA, 11–13 August 2018.
7. Zheng, Z.; Hu, Y.; Zhang, Y.; Yang, H.; Qiao, Y.; Qu, Z.; Huang, Y. CASPPNet: A chained atrous spatial pyramid pooling network for steel defect detection. *Meas. Sci. Technol.* **2022**, *33*, 085403. [CrossRef]
8. Ryselis, K.; Blažauskas, T.; Damaševičius, R.; Maskeliūnas, R. Computer-aided depth video stream masking framework for human body segmentation in depth sensor images. *Sensors* **2022**, *22*, 3531. [CrossRef] [PubMed]
9. Zaaboub, W.; Tlig, L.; Sayadi, M.; Solaiman, B. Neural network-based system for automatic passport stamp classification. *Inf. Technol. Control.* **2020**, *49*, 583–607. [CrossRef]
10. Tianhua, C.; Siqun, Z.; Junchuan, Y. Remote sensing image segmentation using improved deeplab network. *Meas. Control. Technol.* **2018**, *37*, 40–45.
11. Zhang, M.; Jing, W.; Lin, J.; Fang, N.; Wei, W.; Woźniak, M.; Damaševičius, R. NAS-HRIS: Automatic design and architecture search of neural network for semantic segmentation in remote sensing images. *Sensors* **2020**, *20*, 5292. [CrossRef]
12. Fu, K.; Fan, D.-P.; Ji, G.-P.; Zhao, Q.; Shen, J.; Zhu, C. Siamese network for RGB-D salient object detection and beyond. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, *44*, 5541–5559. [CrossRef]
13. Nasiri, A.; Omid, M.; Taheri-Garavand, A.; Jafari, A. Deep learning-based precision agriculture through weed recognition in sugar beet fields. *Sustain. Comput. Inform. Syst.* **2022**, *35*, 100759. [CrossRef]
14. Chen, L.; Yu, E.; Cong, H. Feature fusion network for clothing parsing. *Int. J. Mach. Learn. Cybern.* **2022**, *13*, 2229–2238. [CrossRef]
15. Gabryel, M.; Scherer, M.M.; Sułkowski, Ł.; Damaševičius, R. Decision making support system for managing advertisers by ad fraud detection. *J. Artif. Intell. Soft Comput. Res.* **2021**, *11*, 331–339. [CrossRef]
16. Chatterjee, S.; Majumdar, D.; Misra, S.; Damasevicius, R. The determinants of e-tourism websites for tourists while selecting a travel destination. *Int. J. Electron. Mark. Retail.* **2022**, *13*, 334–359. [CrossRef]
17. Irfan, R.; Almazroi, A.A.; Rauf, H.T.; Damaševičius, R.; Nasr, E.A.; Abdelgawad, A.E. Dilated semantic segmentation for breast ultrasonic lesion detection using parallel feature fusion. *Diagnostics* **2021**, *11*, 1212. [CrossRef]
18. Nawaz, M.; Nazir, T.; Masood, M.; Ali, F.; Khan, M.A.; Tariq, U.; Sahar, N.; Damaševičius, R. Melanoma segmentation: A framework of improved DenseNet77 and UNET convolutional neural network. *Int. J. Imaging Syst. Technol.* **2022**. [CrossRef]
19. Wang, Z.Z. Automatic localization and segmentation of the ventricle in magnetic resonance images. *IEEE Trans. Circuits Syst. Video Technol.* **2021**, *31*, 621–631. [CrossRef]
20. Wang, Z.Z. Robust segmentation of the colour image by fusing the SDD clustering results from different colour spaces. *IET Image Processing* **2020**, *14*, 3273–3281.
21. Vadhnani, S.; Singh, N. Brain tumor segmentation and classification in MRI using SVM and its variants: A survey. *Multimed. Tools Appl.* **2022**, *81*, 31631–31656. [CrossRef]
22. Thayumanavan, M.; Ramasamy, A. An efficient approach for brain tumor detection and segmentation in MR brain images using random forest classifier. *Concurr. Eng. Res. Appl.* **2021**, *29*, 266–274. [CrossRef]
23. Li, X.; Shi, B.; Nie, T.; Zhang, K.; Wang, W. Multi-object recognition method based on improved yolov2 model. *Inf. Technol. Control.* **2021**, *50*, 13–27. [CrossRef]
24. Sun, Z.; Zhao, M.; Jia, B. A gf-3 sar image dataset of road segmentation. *Inf. Technol. Control.* **2021**, *50*, 89–101. [CrossRef]
25. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
26. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.

27. Chen, L.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, fully connected crfs. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *40*, 834–848.
28. Noh, H.; Hong, S.; Han, B. Learning deconvolution network for semantic segmentation. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), ICCV'15, Santiago, Chile, 7–13 December 2015; pp. 1520–1528.
29. Girshick, R. Fast r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015.
30. LeCun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based learning applied to document recognition. *Proc. IEEE* **1998**, *86*, 2278–2324. [CrossRef]
31. Chen, S.; Xu, X.; Yang, N.; Chen, X.; Du, F.; Ding, S.; Gao, W. R-net: A novel fully convolutional network–based infrared image segmentation method for intelligent human behavior analysis. *Infrared Phys. Technol.* **2022**, *123*, 104164. [CrossRef]
32. Li, H.; Fan, J.; Hua, Q.; Li, X.; Wen, Z.; Yang, M. Biomedical sensor image segmentation algorithm based on improved fully convolutional network. *Meas. J. Int. Meas. Confed.* **2022**, *197*, 111307. [CrossRef]
33. Wang, Z.; Peng, T.; Lu, Z. Comparative research on forest fire image segmentation algorithms based on fully convolutional neural networks. *Forests* **2022**, *13*, 1133. [CrossRef]
34. Sun, Y.; Bi, F.; Gao, Y.; Chen, L.; Feng, S. A multi-attention UNet for semantic segmentation in remote sensing images. *Symmetry* **2022**, *14*, 906. [CrossRef]
35. Rajinikanth, V.; Kadry, S.; Nam, Y. Convolutional-neural-network assisted segmentation and svm classification of brain tumor in clinical mri slices. *Inf. Technol. Control.* **2021**, *50*, 342–356. [CrossRef]
36. Kadry, S.; Rajinikanth, V.; Taniar, D.; Damaševičius, R.; Valencia, X.P.B. Automated segmentation of leukocyte from hematological images—A study using various CNN schemes. *J. Supercomput.* **2022**, *78*, 6974–6994. [CrossRef]
37. Maqsood, S.; Damasevicius, R.; Shah, F.M. An efficient approach for the detection of brain tumor using fuzzy logic and U-NET CNN classification. In *Lecture Notes in Computer Science, Proceedings of the International Conference on Computational Science and Its Applications, Cagliari, Italy, 13–16 September 2021*; Springer: Berlin/Heidelberg, Germany, 2021; Volume 12953, pp. 105–118. [CrossRef]
38. Meraj, T.; Alosaimi, W.; Alouffi, B.; Rauf, H.T.; Kumar, S.A.; Damaševičius, R.; Alyami, H. A quantization assisted U-net study with ICA and deep features fusion for breast cancer identification using ultrasonic data. *PeerJ Comput. Sci.* **2021**, *7*, e805. [CrossRef] [PubMed]
39. Barbato, M.P.; Napoletano, P.; Piccoli, F.; Schettini, R. Unsupervised segmentation of hyperspectral remote sensing images with superpixels. *Remote Sens. Appl. Soc. Environ.* **2022**, *28*, 100823. [CrossRef]
40. Li, H.; Yin, Z. Attention, suggestion and annotation: A deep active learning framework for biomedical image segmentation. In *Lecture Notes in Computer Science, Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Lima, Peru, 4–8 October 2020*; Springer: Berlin/Heidelberg, Germany, 2020. [CrossRef]
41. Ashraf, H.; Waris, A.; Ghafoor, M.F.; Gilani, S.O.; Niazi, I.K. Melanoma segmentation using deep learning with test-time augmentations and conditional random fields. *Sci. Rep.* **2022**, *12*, 3948. [CrossRef]
42. Abdollahi, A.; Pradhan, B.; Alamri, A.M. An ensemble architecture of deep convolutional Segnet and Unet networks for building semantic segmentation from high-resolution aerial images. In *Geocarto International*; Informa UK Limited: London, UK, 2020; Volume 37, pp. 3355–3370. [CrossRef]
43. Fauzi, F.; Hong, J.-L.; Belkhatir, M. Webpage segmentation for extracting images and their surrounding contextual information. In Proceedings of the Seventeen ACM International Conference on Multimedia—MM '09, Beijing, China, 19–24 October 2009; ACM Press: New York, NY, USA, 2009. [CrossRef]
44. Sanoja Vargas, A. Web Page Segmentation, Evaluation and Applications. Ph.D. Thesis, Universite Pierre et Marie Curie—Paris VI, Jussieu, Paris, France, 2015.