Journal of
*Marine Science
and Engineering*

Article

A Multi-Strategy Framework for Coastal Waste Detection

Chengjuan Ren [1], Sukhoon Lee [2], Dae-Kyoo Kim [3], Guangnan Zhang [4,*] and Dongwon Jeong [2,*]

[1] Guangdong Atv Academy for Performing Arts, Zhaoqing 526000, China
[2] Software Convergence Engineering Department, Kunsan National University, Gunsan 54150, Korea
[3] Computer Science and Engineering Department, Oakland University, Rochester, MI 48309, USA
[4] Department of Computer Science, Baoji University of Arts and Science, Baoji 721000, China
[*] Correspondence: zgn_2003@163.com (G.Z.); djeong@kunsan.ac.kr (D.J.)

Abstract: In recent years, deep learning has been widely used in the field of coastal waste detection, with excellent results. However, there are difficulties in coastal waste detection such as, for example, detecting small objects and the low performance of the object detection model. To address these issues, we propose the Multi-Strategy Deconvolution Single Shot Multibox Detector (MS-DSSD) based on DSSD. The method combines feature fusion, dense blocks, and focal loss into a state-of-the-art feed-forward network with an end-to-end training style. In the network, we employ feature fusion to import contextual information to boost the accuracy of small object detection. The dense blocks are constructed by a complex function of three concurrent operations, which can yield better feature descriptions. Then, focal loss is applied to address the class imbalance. Due to the lack of coastal waste datasets, data augmentation is designed to increase the amount of data, prevent overfitting of the model, and speed up convergence. Experimental results show that MS-DSSD513 obtains a higher mAP, of 82.2% and 84.1%, compared to the state-of-the-art object detection algorithms on PASCAL VOC2007 and our coastal waste dataset. The proposed new model is shown to be effective for small object detection and can facilitate the automatic detection of coastal waste management.

Keywords: deep learning; Deconvolution Single Shot Multibox Detector; waste detection

check for updates

Citation: Ren, C.; Lee, S.; Kim, D.-K.; Zhang, G.; Jeong, D. A Multi-Strategy Framework for Coastal Waste Detection. *J. Mar. Sci. Eng.* **2022**, *10*, 1330. https://doi.org/10.3390/jmse10091330

Academic Editor: Fabio Bruno

Received: 28 July 2022
Accepted: 15 September 2022
Published: 19 September 2022

1. Introduction

Environmental pollution refers to natural or artificial damage caused by the addition of harmful substances that exceed the environment's self-purifying capacity and produce harm, which adversely affects the growth and reproduction of organisms and the everyday life of human beings. In recent years, with economic and social development and the acceleration of the pace towards tourism, the awareness of seaside coastal leisure has developed rapidly. Seaside coastal are becoming more and more prominent in people's tourism and holiday aspirations. While driving the development of related industries, they have also given rise to many coastal pollution problems where the chief source of waste on the coast is caused human activity. Coastal waste is composed mainly of plastic waste, occupying 90% of all debris, especially various beverage bottles. Other common types of coastal waste are paper, wood, cloth, metal, and food [1–3].

Despite the efforts to limit the production of coastal waste in various ways, this has become a serious ecological, aesthetic, and socially urgent task. Waste disposal is usually a tiresome and time-consuming task. Therefore, there is an urgent need to develop an efficient and automatic waste disposal process that can save significant workforce and resources. The automatic sorting and detection of waste are critical in business and life.

Object detection is one of the tasks of computer vision that addresses object instance detection and requires an answer to what object is located where. Most early object detection algorithms were established using hand-crafted features. Due to the lack of effective image representations at the time, there were no other options but to devise complex feature representations and various acceleration skills to refine the constrained computational resources in use. In 2005, the Histogram of Oriented Gradient (HOG) feature proposed by Dalal and Triggs [4] was proposed as a feature descriptor for computer vision and image processing for object detection. The HOG feature constitutes a feature of calculating and counting the histogram of the gradient direction of a local region of an image. The HOG operates on a local square cell of the image, which allows it to maintain good invariance to the image's geometric and optical deformations. Secondly, with coarse null sampling, fine directional sampling, strong local optical normalization, and some subtle body movements can be tolerated as long as the pedestrian in the image generally remains upright. These slight movements can be ignored without affecting the detection effect. The Deformable Part-based Model (DPM) proposed by Felzenszwalb et al. [5] can be seen as an extension of HOG, where the histogram of gradient directions is first computed and then trained with SVM to obtain a gradient model of the object. Such an algorithm can be used directly for classification, which can be understood as matching the model to the target. DPM is the highest achievement of conventional object detection algorithms. Although many of the current object detection algorithms far surpass these classical algorithms in terms of performance and speed, they have profoundly influenced the research and expansion of new algorithms.

Object detection has achieved a high point in the era of deep learning. Object detection can be divided into the following two groups: "two-stage detection", which positions the detection as a coarse-to-fine process, and "single-stage detection", which sets it as one-step detection. Most of the methods based on two-stage are derived from R-CNN. In response to the problems of computational redundancy and slow speed of R-CNN [6], Fast R-CNN [7] and Faster R-CNN [8] were proposed to simplify candidate region generation and improve the speed of object detection. They used selection search and RPN to replace sliding window search. Feature Pyramid Networks (FPN) [9] based on Faster R-CNN was proposed in 2017 to resolve the problem of multiscale variation in object detection with the technique of image pyramids. The method performs feature extraction for each scale of the image and can produce multiscale feature representation that enhances the semantic information about features. However, the technique occupies large amounts of memory space and substantially increases the inference time.

Although the Faster-RCNN method is many times faster than R-CNN, it still does not meet the needs of real-time applications, especially those that can no longer run on GPUs. Several CNN-based object detection methods have been developed to address the needs of real-time applications. The You Only Look Once (YOLO) [10] series is the most popular method, which belongs to one-stage detection algorithms. YOLO takes object detection as a single end-to-end network for solving regression problems. The input image is inferred once to obtain the positions of all objects in the image and the corresponding confidence probabilities for the classes in which the model falls. In contrast to YOLO, the Single Shot MultiBox Detector (SSD) [11] not only performs object prediction on the top layers but also on the first few layers of the network (i.e., multiscale fusion technology, which ensures accuracy of the recognition of small objects). The RetinaNet [12] aims to address the class imbalance between samples, making smaller loss contributions of easily classified samples and larger loss contributions of hard samples.

The highest precision object detector is based on two-stage detection methods, where the classifier is used to classify candidate regions produced in the first step. By contrast, a one-stage detector is applied to predict the possibility of objects with faster speed directly, but this trails the accuracy of two-stage detectors. The main reason is that a one-stage detector is prone to class imbalance during training. The imbalance in the ratio of extreme positive to negative samples tends to degrade the model's performance, so focal loss is employed to balance the weight of samples by substituting the standard loss function in the model. Moreover, context is essential for finding small object examples. Because of the large number of small objects (e.g., glass fragments and shredded paper) in coastal waste, it is not easy to accurately delineate the boundaries of these objects without zooming in, for most people. To address this issue, more contextual information about the object should be mined to facilitate the detection of small objects. In our model, the multiscale fusion technique is incorporated for accurate localization. This includes the following four points:

To consider the object detection of different scales, shallow features are used to detect small objects, as the perceptual field of the shallow layer is narrow and the size of the perceptual field exactly matches small objects. However, high-resolution features from the shallow layer lack semantic information, which can affect the detector's ability to determine whether the detection area is the target or background. Then, fusing the higher-level features with the lower-level features can result in features with the correct perceptual field and with semantic information preserved. Experimentally, feature fusion is effective in improving the performance of the model.

Dense blocks are introduced into the model to resolve some layers' contributions and obtain a more efficient feature description. In the blocks, the input of each layer is the output of all previous layers' mapping, and the result of this feature mapping is used as the input of later layers. With the addition of dense blocks, the reusing feature is carried forward. At the same time, the ability of feature propagation is also enhanced to achieve the implied deep monitoring. In experiments on this work, the blocks are shown to be feasible.

In two-stage algorithms, a large number of negative samples are filtered out by score and NMS in the candidate region stage. Then the proportion of positive and negative samples is fixed in the classification and regression stage. However, one-stage methods have the advantage of speed, but suffer from class imbalance during training. Therefore, the standard cross-entropy loss function is substituted by focal loss in the model, which reduces the weight of the large proportion of simple negative samples in training. Finally, it is shown that the improved model is as fast as the original model and also achieves accuracy comparable to that of two-stage algorithms.

Due to the shortage of coastal waste data, we use data augmentation techniques to improve the accuracy and generalization of the model. The essence of data augmentation is to increase the data by introducing prior knowledge. The augmented samples are strongly correlated with the original samples, allowing the model to learn more comprehensive image features and transformations.

To our knowledge, there is currently a severe lack of data on coastal waste. To bring this branch of research to the attention of more scholars, we have created a public dataset, "IST-Waste-V2" (an improvement on our first version of the dataset) in the field, which will be released later.

The rest of this paper is structured as follows. Section 2 reviews related work in waste classification and detection, multiscale fusion, dense blocks, and loss function. Section 3 describes YOLO, SSD, and DSSD in detail, laying the foundation for the proposed model. Section 4 gives the model's structure and a detailed presentation of the revised module. Experimental details and results analysis are listed in Section 5. Section 6 concludes the paper.

2. Related Works

The ability to automatically sort and detect waste is of broad concern in the fields of computer and robot vision. Data occupy a vital place in deep learning. Owing to the scarcity of publicly available waste data, there exists only a small amount of literature addressing waste detection and classification for this specific domain, compared to general object detection and classification. To resolve the waste sorting problem, which is time-consuming and labor-intensive, Ma et al. [13] proposed a framework called L-SSD to implement automatic waste sorting recognition. The main idea of L-SSD is to change the original SSD's backbone network, optimize the NMS algorithm, use the focal loss function, and employ feature fusion techniques. Their experiments demonstrated that the framework outperforms many state-of-the-art object detection approaches in accuracy and speed. Panwar et al. [14] developed a deep learning model called AquVision and organized a new sea body dataset named AquTrash to detect harmful litter floating on seashores and coasts. The model's accuracy reaches 0.81 mAP, which is adequate to clean the water body and purify the environment. To overcome the shortcomings of litter data, such as detection of small objects and difficult identification of stacked data, Shi et al. [15] rely on optimizing the network structure, building deep networks, and using short-circuit connections to improve the performance of previous models on the TrashNet dataset. They demonstrated that the M-bXception network achieves 94% classification accuracy on the TrashNet dataset. Toğaçar et al. [16] recomposed waste data using AutoEncoder and used a convolutional neural network to extract features from two datasets. Then, RR was used to reduce the number of features and filter the effective features. Finally, SVM was utilized for classification. Yi and Chellappan et al. [17] analyzed and designed a system for automatic detection and localization of street litter based on convolutional neural networks. The data were derived from real city street images containing 2500 images and 6474 objects. The evaluation results show that the system can be an essential part of the next generation of Intelligent Management Systems to achieve greener and healthier communities. Nazerdeylami et al. [18] presented a model for automatic waste surveying and regulation of human activities in coastal eco-cyber-physical systems. The model enables autonomous inspection of human activities and monitoring vehicles and boats for illegal entry into a coastal region. Kraft et al. [19] presented a low-cost scheme capable of locating litter objects in low-level images acquired by unmanned aerial vehicles (UAVs) during automated patrol operations. The core idea of this model is based on deep convolutional neural networks.

Multiscale techniques are often employed to improve the performance of models, especially for small objects. These can be divided into the following two main categories: (i) independent detection using features extracted from different layers of the network and (ii) fusion of features from different layers. The former is represented by SSD, which is proven to be more efficient than detecting objects on feature maps extracted from only the top layer of the coarser network. In 2018, Li et al. [20] achieved state-of-the-art performance in pedestrian detection by using a scale-aware technique to assign weightings and the combination of prediction results from large and small-size sub-networks based on input suggestions. A typical representative of the latter is the Spatial Pyramid Pooling Network (SPP) model proposed by He et al. [21]. SPP can produce a fixed size output regardless of the input size and use different sizes of the same image as input to obtain the same length of pooled features. A deep convolutional neural network is proposed to build a feature pyramid structure named FPN, a top-down structure with lateral connections to build high-level semantic feature maps at different scales. FPN can handle the multiscale variation of object detection with increased performance computational power.

Attention mechanisms are also widely used in deep learning to address detailed features, rather than considering entire images. Various kinds of attention are captured in different attention modules, including attention residual learning and bottom-up, top-down feed-forward attention. There is evidence from processes of human cognition [22] indicating the relevance of attentional mechanisms, which use top information to steer bottom-up feed-forward processes. Recently, initial attention has been applied to deep neural networks. The Deep Boltzmann Machine [23] incorporates top-down attention during the training phase through its reconstruction process. The soft attention exploited in more recent work [24,25] allows end-to-end training for deep convolutional networks.

Another area of related work is the study of loss functions and multiscale detection. The loss function, which is critical for object detection and classification, can be separated into three parts: the loss in classification, the loss in object position regression, and the combination of these two losses to form the total loss function. Object classification loss usually uses softmax cross-entropy [26–28] or sigmoid cross entropy [29]. In the work by Liu's et al. [11], the contribution of the sample in the loss is measured according to the difficulty of the training sample. For the location regression, smooth L2 or L1 loss is employed. Alternatively, their variant loss is used to evaluate the model. The union of different loss functions is necessary for multi-tasking networks, such as Multinent [30]. The loss of general target detection consists of classification loss and localization loss.

3. Background

Currently, many researchers have focused on improving object detection and classification accuracy to achieve impressive results. This section reviews several well-known first-stage object detection algorithms, including YOLO, SSD, and DSSD, that are relevant to this work.

3.1. YOLO

YOLO [10] utilizes a convolutional network to extract features. The fully connected layer is used to obtain predicted values. Figure 1 shows the network structure containing 24 convolutional layers and 2 fully connection layers. In the convolutional layers, 1 × 1 convolutions and 3 × 3 convolutions are used, where 1 × 1 convolutions are to change the number of channels, immediately followed by 3 × 3 convolutions. The network splits the input image into s × s cells. Each cell is responsible for detecting those targets whose center point falls within that cell and predicting the B bounding boxes and confidence score. The confidence score consists of two aspects. The first is the possibility of the bounding box containing the target. The second is the accuracy of this bounding box. YOLO views object detection as a regression problem. However, in the loss function, different weights can describe various components to balance the impact of loss. A larger weight is assigned to the positioning error (i.e., the bounding box coordinate error). The confidence of the bounding box outside the target is distinguished from that of the bounding box within the target. For the former, a smaller weight is assigned, and other weights can be set to 1. In addition, while each cell predicts multiple bounding boxes, its corresponding class is only one. If a target exists in that cell during training, only the bounding box with the largest IoU of ground truth is selected to be accountable for predicting. In contrast, the other bounding boxes are considered to have no target. YOLO is an end-to-end CNN network that employs a single-pipeline strategy. The model performs convolution on the whole image, so it has a larger field of view in detecting the target and meets the need for real-time detection in terms of speed. Its disadvantages include, for small objects, that the model does not perform as well as it should, and the low generalization rate in terms of an aspect ratio of objects makes it impossible to locate some objects.

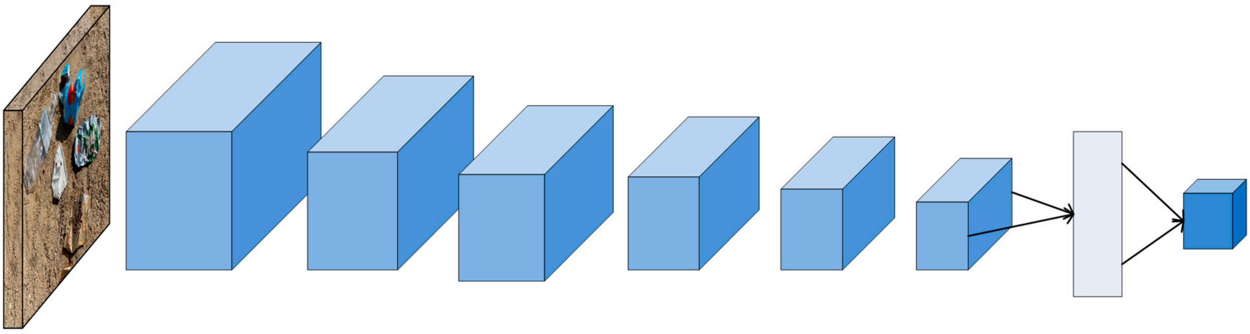

Figure 1. The basic structure of YOLO.

3.2. SSD

SSD performs better in terms of robustness and accuracy to various scales among many object detection algorithms. SSD is not like a two-stage object detection model to generate the region suggestion process. The localization is also done by convolution, which realizes end-to-end object detection. Figure 2 shows the structure of SSD, which uses feature maps of different scales to detect the same image and contains various semantic features and positional sensitivity for multiscale detection. The model usually detects images with six layers of different sizes of feature maps, including Conv7, Conv8_2, Conv9_2, Conv10_2, and Conv11_2. When the input image is 300 × 300 pixels, the size of each feature map of softmax classification and location regression varies from 38 × 38 to 1 × 1. At the same time, the SSD model filters positive and negative samples, mainly based on the IoU between ground-truth boxes and prior boxes. The choice of positive samples is carried out in two steps. The first step is to find the best matching prior boxes and put them into the positive sample set by the ground-truth boxes. The second step involves the prior boxes with IoU greater than 0.5 from the ground-truth boxes and putting them into the positive sample set, to increase the number of positive samples. Such a screening method causes a problem of unbalanced positive and negative samples. This problem is solved simply by controlling the number of negative samples through OHEM, forcing the number of positive samples to be 3:1. In the training process, SSD also adapts a data augmentation strategy to improve the generalization ability and accuracy of the model. Meanwhile, non-maximum suppression is also indispensable to remove prediction boxes with a large overlap. Finally, only one box with great confidence is retained as output.

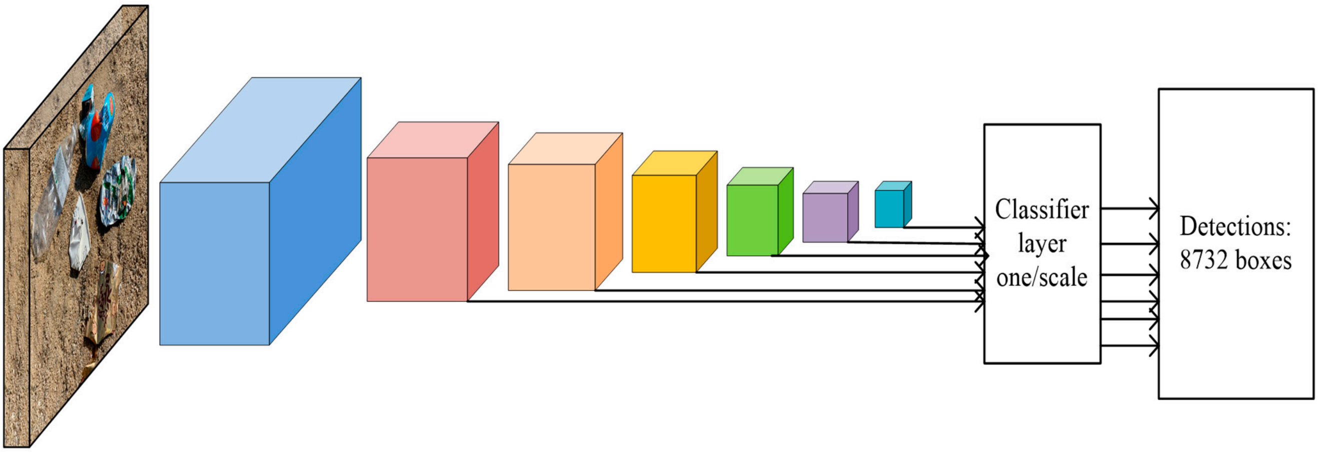

Figure 2. The basic structure of SSD.

3.3. DSSD

The original DSSD model based on the SSD model essentially makes the following two significant changes: (i) adopting ResNet101 instead of VGG16 as the feature extraction network, and (ii) introducing prediction modules to improve accuracy. In addition, it uses multiple deconvolution layers to expand the high-level feature information of the model. It performs an element-wise product with feature maps at the same scales of the front-end convolution layers to generate feature maps at the corresponding scales, which effectively improves the detection of small objects. The large depth of the original network framework and many fusion layers eventually lead to a decline in speed. The structure of the DSSD model is shown in Figure 3. The deconvolution module contributes to fusing the high-level and low-level feature mapping information. DSSD uses deconvolution layers instead of bilinear up-sampling and adds a normalization layer after each convolution layer. The residual unit is added to the SSD prediction module, and the original feature map is convolved in the residual bypass. Then the feature map of the network backbone is summed between channels to form a new prediction module.

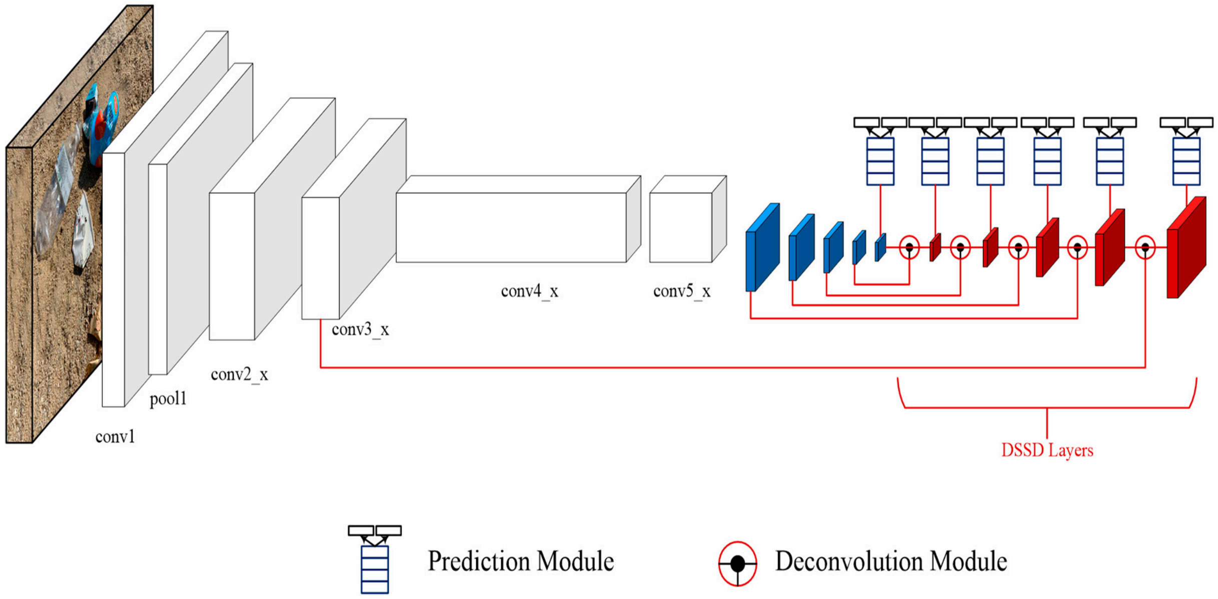

Figure 3. The basic structure of DSSD.

3.4. Limitations of YOLO, SSD, and DSSD

YOLO makes predictions based on whole image information, while other sliding-window detection frameworks can only perform inference based on partial image information. Because YOLO does not include sample regions, it performs well on global data, but poorly on small areas of information. The SSD object detection model is one of the best one-stage object detection algorithms. The model can not only run at the speed of real-time detection but also with high accuracy, compared to two-stage object detection. The training is relatively simple without the intermediate weight storage process of two-stage object detection, which can realize end-to-end object detection. However, the performance in small target detection is mediocre. On the one hand, the backbone network of the SSD model is mainly designed for recognition and classification tasks, so the representation capability is insufficient for object detection. On the other hand, the SSD model picks six feature maps generated by the backbone network VGG-16 and then classifies and regresses them separately, without considering that different feature maps have different representational capabilities for other targets. The deconvolution module of the DSSD model improves the utilization of information and increases the detection speed. However, the algorithm's computational complexity increases with the increase of network layers, thus reducing the detection speed of the model on an experimental data set.

4. Our Approach

In the section, we discuss the proposed model, named MS-DSSD in this work. The MS-DSSD model is composed of a series of convolution and deconvolution layers with Residual-101 as the backbone and a deconvolution module to introduce spatial contextual information. The structure schematic diagram of MS-DSSD is shown in Figure 4. Dense blocks are applied behind the Conv2_x and Conv5_x layers, which can enhance the transferability of features. It also allows more efficient use of image features. Although the network is becoming complicated compared to DSSD, the number of parameters is small. Subsequently, the union of multiscale features is added at the Conv2_x and Conv5_x layers, increasing the model's global and local feature information and extending the feature migration capability of dense blocks in object detection. In addition, focal loss functions and data augmentation are also introduced to improve the model's performance.

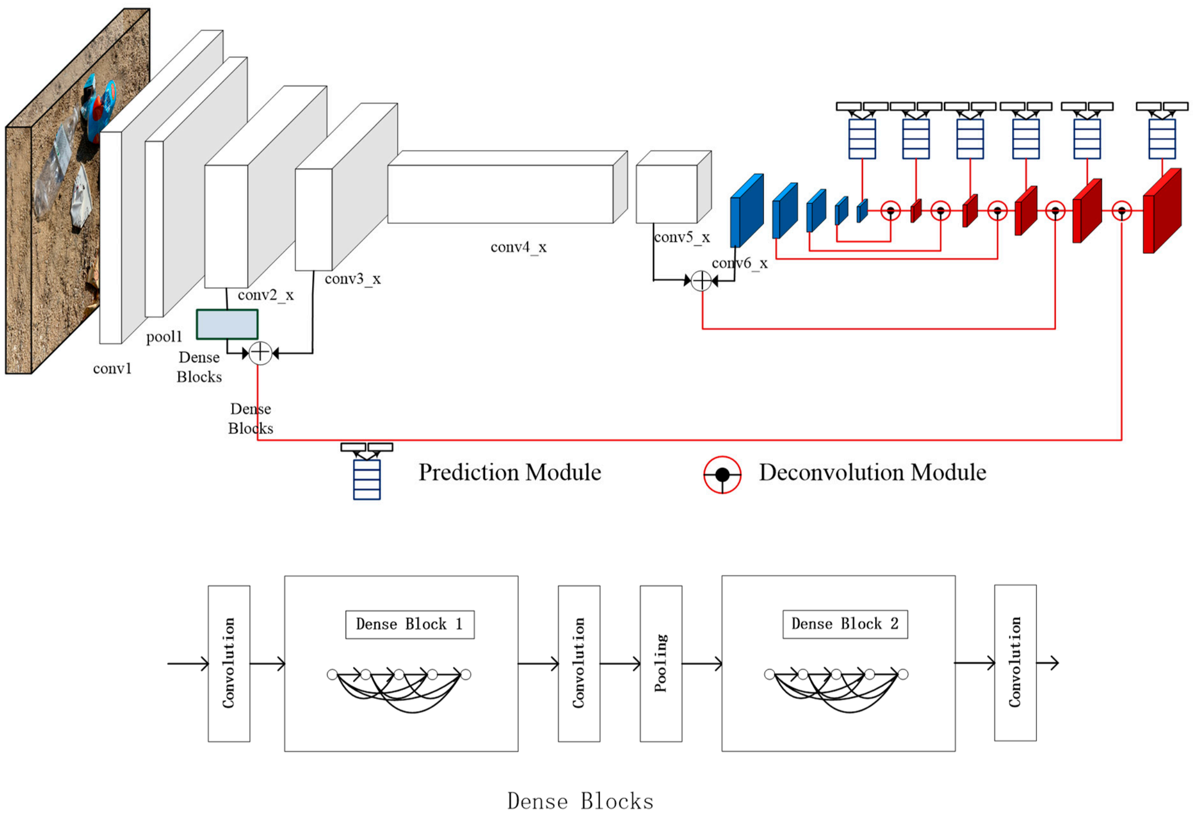

Figure 4. The structure of MS-DSSD.

4.1. Context Feature-Fused

Object detection consists of two subtasks in computer vision tasks, object identification and localization. Image invariance and equivalence transformation are two essential properties in image feature representation. Efficiently combining the two, to utilize their strengths while discarding their weaknesses, is the key to improving the model. In the model, low-level features have higher resolution and contain more location information, but they are less semantic and noisier because they undergo less convolution. However, the high-level features have stronger semantic information but very low resolution and poor perception of details. Usually, feature fusion consists of concatenating and adding, as shown in Figure 5. Connection increases the number of channels while adding is the summation of feature maps with the same number of channels. Adding increases the amount of information under the features describing the image, but the number of dimensions describing the image itself has not increased, which is beneficial for the classification of the final image. However, concatenating is the merging of the number of channels. The number of features (number of channels) describing the image itself is enlarged, while the information under each feature is not. So, in the latter, contextual feature fusion is applied. The concatenating algorithm is defined in (1). Suppose two input channels X1, X2, . . . , Xc and Y1, Y2, . . . , Yc. K stands for convolution kernel. Then, the individual output channels are combined. (* denotes convolution). Conv2_x goes through the dense blocks and Conv3_x is for feature fusion, Conv5_x and Conv6_x are for feature fusion, and the next is similar to DSSD in the model as illustrated in Figure 4. Experiments have shown that that the fusion of features at different scales is a significant means to improve detection performance.

$$\mathbf{Z}_{\mathbf{concat}} = \sum_{\mathrm{i}=1}^{\mathrm{c}} \mathbf{X}_{\mathrm{i}} * \mathbf{K}_{\mathrm{i}} + \sum_{\mathrm{i}=1}^{\mathrm{c}} \mathbf{Y}_{\mathrm{i}} * \mathbf{K}_{\mathrm{i+C}} \tag{1}$$

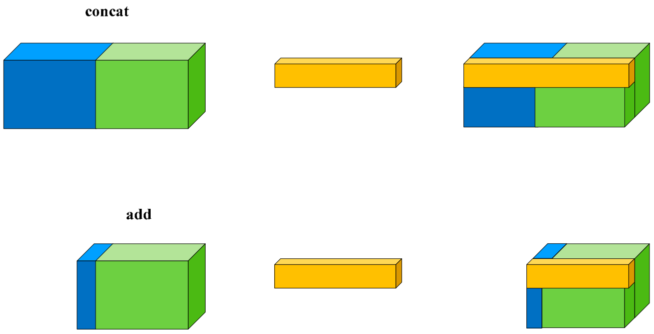

Figure 5. Feature fusion mode (All color represent features).

4.2. Dense Block

For more effective feature extraction, inspired by the DenseNets structure developed by Huang G et al. [31], we employ the dense structure that is used in the DSSD model to detect objects. For our MS-DSSD, we put two dense blocks after conv2_x in the network. This design also ensures maximum information transfer between layers in the network. The dense block structure is shown in Figure 6. When a network has L layers, the number of connections is **L(L + 1)/2**. This means that the input of each layer comes from the output of all the previous layers. Therefore, the feature results of the $\mathbf{l^{th}}$ the layer can be expressed as:

$$\mathbf{X_L} = \mathbf{H_L}([\,\mathbf{X}_0,\ \mathbf{X}_1, \cdots, \mathbf{X_{L-1}}]) \tag{2}$$

where $\mathbf{H_L}(.)$ is defined as a complex function of three concurrent actions, batch normalization rectified linear unit, and convolution. $[\,\mathbf{X}_0,\ \mathbf{X}_1, \cdots, \mathbf{X_{L-1}}]$ denotes the connection of the feature maps generated in layers $0, \cdots, \mathbf{L} - 1$. This approach can facilitate the building of deeper layers of network structures. Some features extracted from earlier layers may still be used directly by deeper layers. Each layer in the dense blocks receives supervision from the loss and has multiple bypasses and shortcuts. Therefore, the supervision of the blocks is diverse, which causes the network to obtain better features.

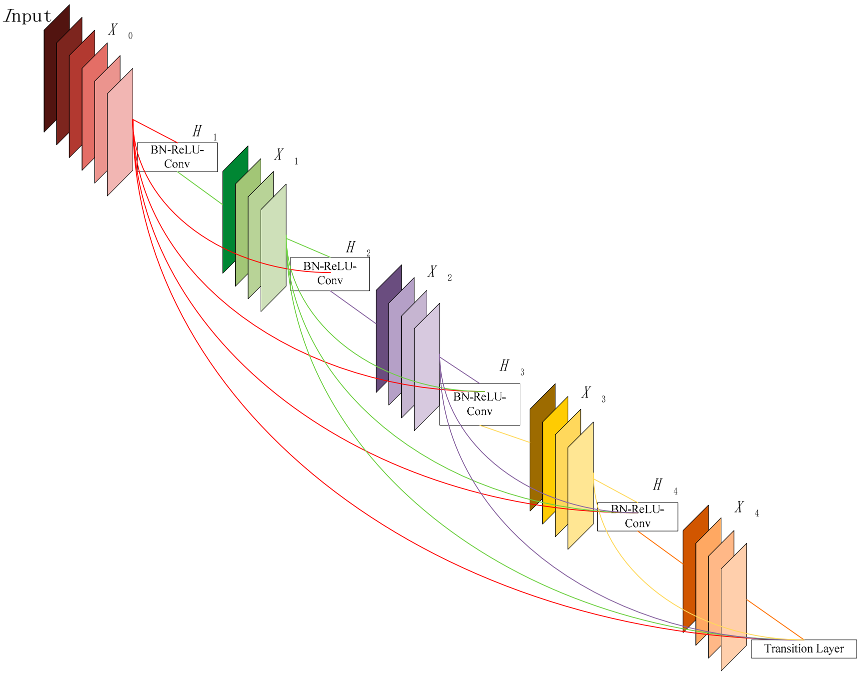

Figure 6. The basic structure of the dense block.

4.3. Focal Loss

Focal loss was proposed by Lin et al. [29] to answer the question of why one-stage accuracy is not as good as two-stage in the network. After extensive experimental analysis, it was concluded that the training phase of object detection is prone to sample class imbalance or hard and easy sample imbalance. Both imbalance issues may also exist. The process of focal loss formation is defined as follows:

The origin of focal loss is the common cross-entropy loss function for binary classification. It is defined as:

$$\mathbf{CE}(\mathbf{p},\ \mathbf{y}) = \begin{cases} -\mathbf{log}(\mathbf{p}), & \mathbf{if}\quad \mathbf{y{=}1} \\ -\mathbf{log(1{-}p)}, & \mathbf{otherwise} \end{cases} \tag{3}$$

The value of y is 1 for ground truth, positive samples, and −1 for negative samples. $0 <= \mathbf{p} <= 1$ denotes the model's output value for each sample with ground truth label = 1. To simplify the above equation, $\mathbf{p_t}$ is defined as:

$$\mathbf{p_t} = \begin{cases} \mathbf{p}, & \mathbf{if}\quad \mathbf{y{=}1} \\ \mathbf{1{-}p}, & \mathbf{otherwise} \end{cases} \tag{4}$$

Then, $\mathrm{CE}(\mathbf{p}, \mathbf{y}) = \mathrm{CE}(\mathbf{p_t}) = -\log(\mathbf{p_t})$

The number of easily classified samples is much larger than hard samples, so the loss of easy samples and the corresponding gradient dominates during the training process. To offset the relative importance of positive and negative samples, α is introduced. **n** is a regulating factor to decrease the loss percentage of easily classified samples as defined in (5).

$$\mathbf{n} = (1 - \mathbf{p_t})^{\gamma} \tag{5}$$

So, the focal loss function changes to

$$\mathbf{FL}(\mathbf{p_t}) = -\mathbf{a_t}(1 - \mathbf{p_t})^{\gamma}\mathbf{log}(\mathbf{p_t}) \tag{6}$$

It is noteworthy that γ and α are sensible to easily classified samples. FL is short for focal loss. The proposed model in this work uses the focal loss function to achieve better experimental performance.

4.4. Mosaic

As the neural network deepens, the number of parameters to be learned increases, which can easily lead to model overfitting. When the data set is small, the parameters can fit all the characteristics of the data sample instead of the commonality between the samples. To prevent the overfitting phenomenon, data augmentation enables limited data to produce value equivalent to more data. Data augmentation methods commonly include random rotation, random cropping, horizontal flipping, and scale. In this work, Mosaic is employed to increase the diversity of the sample and avoid model overfitting. Figure 7 describes the basic principle of Mosaic. The core idea of data augmentation is to take four images and stitch them together by random scaling, random cropping, and random lining up. The advantage of Mosaic is that it can significantly enrich the detection dataset, especially using random scaling, which can add many small targets. Thus, it makes the network more robust. Batch normalization can calculate four data images at a time, so that the mini-batch size does not need to be large, which helps the single GPU to achieve better results.

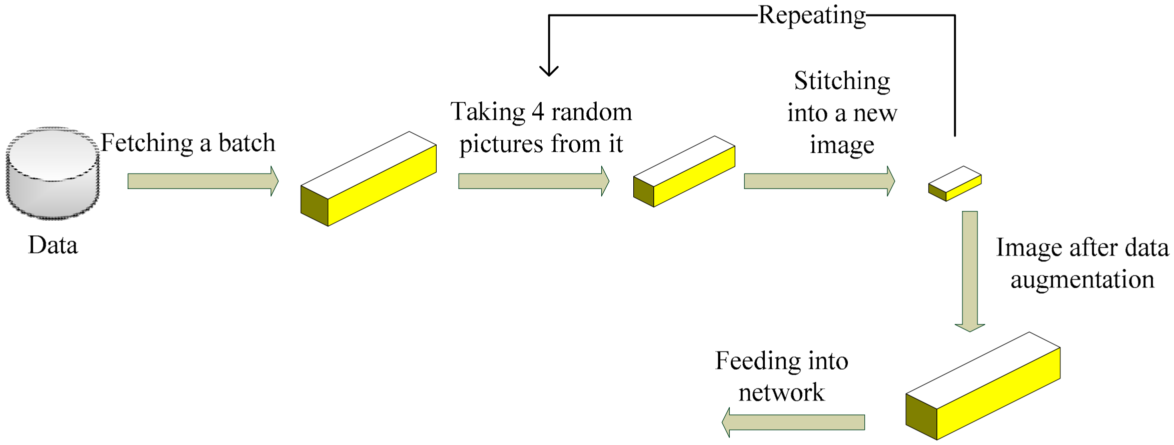
Repeating
Data
Fetching a batch
Taking 4 random pictures from it
Stitching into a new image
Image after data augmentation
Feeding into network

Figure 7. The principle of Mosaic.

5. Experiments

The performance of the proposed new model was measured on PASCAL VOC2007 and our coastal waste dataset (IST-Waste-V2) in experiments. The experimental hardware included mainly NVIDIA GeForce RTX 2080 and IntelXeon (R) CPU E5-2620 v3 2.40 GHz. The experimental operating system was Ubuntu 16.04. Python 3.8, and the Pytorch 1.2.0 deep learning framework were also used. All training and testing were conducted in the same setting. The model employed the Stochastic Gradient Descent (SGD) for end-to-end training. The momentum parameter was 0.9. The batch size was set to 32 for the model with 321 inputs and 16 for the model with 513 inputs. We used the well-trained DSSD model as the pre-trained model for the MS-DSSD.

5.1. PASCAL VOC2007

The learning rate was fixed to 1×10^{-3} at the beginning of iteration and decreased to 1×10^{-4} as the iterations increased to 40 K. The experiments were related to Residual-101, which is a fast convergence model. Table 1 shows the result of PASCAL VOC2007 test detection. SSD300* and SSD513* are the most recent SSD achievements with new extended data enhancement techniques that are already superior to many other state-of-the-art detectors [11]. MS-DSSD321 adds feature fusion and dense blocks and modifies loss function and the data enhancement method, compared to the original DSSD321 model [26]. The result shows 1.7% better performance than DSSD321. Interestingly, when we increased the input image size, MS-DSSD513 resulted in the best accuracy, outperforming DSSD513, DSSD321, and MS-DSSD321 by 0.7%, 3.6%, 1.9%, respectively. There are several reasons for this. Residual-101 needs more effective features for objects in the deeper layers to obtain spatial information. More importantly, because we incorporated feature fusion and dense blocks into DSSD, the model also addressed the class balance issue with focal loss function. These techniques allow the model to encompass more contextual information, focus on the tasks concerned more with this work, and mitigate the gradient disappearance problem, which prove the validity of our proposed method. The test results are shown in Figure 8. Due to space limitations, we show only a diagram of the detection results of the three models on the PACAL 2007 test set. The red test boxes on the left represent the test results of the SSD321, the blue boxes in the middle are the result of DSSD321, and the green boxes on the right are the result of MS-DSSD321. As shown in the results, MS-DSSD321 outperforms other methods in both detecting small objects and in object integrity.

Table 1. Detection results using PASCAL VOC2007.

Method	mAP	Bike	Aero	Bird	Boat	Bottle	Bus	Car	Cat	Chair	Cow
Faster R-CNN	76.4	80.7	79.8	76.2	68.3	55.9	85.1	85.3	89.8	56.7	87.8
SSD321	77.1	76.3	84.6	79.3	64.6	47.2	85.4	84.0	88.8	60.1	81.5
DSSD321	78.6	81.9	84.9	80.5	68.4	53.9	85.6	86.2	88.9	61.1	82.6
DSSD513	81.5	86.2	86.6	82.6	74.9	62.5	89.0	88.7	88.8	65.2	87.0
MS-DSSD 321	80.3	83.1	84.6	82.5	70.2	57.6	89.1	88.2	87.9	63.8	85.2
MS-DSSD 513	82.2	87.8	86.9	84.1	76.5	66.1	89.2	88.5	89.0	66.9	87.5
Method	mAP	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv
Faster R-CNN	76.4	69.4	87.2	88.9	80.9	78.4	41.7	78.6	79.8	85.3	72.0
SSD321	77.1	76.9	86.7	87.2	85.4	79.1	50.8	77.2	82.6	87.3	76.6
DSSD321	78.6	78.7	86.7	88.7	86.7	79.7	51.7	78.0	80.9	87.2	79.4
DSSD513	81.5	78.7	88.2	89.0	87.5	83.7	51.1	86.3	81.6	85.7	83.7
MS-DSSD 321	80.3	78.9	87.6	89.2	87.1	83.7	52.6	84.2	81.1	87.3	81.9
MS-DSSD 513	82.2	78.4	87.9	89.9	88.8	84.8	54.3	86.1	82.0	85.4	82.9

MS-DSSD shows excellent performance improvement for some classes with a particular background and includes small objects in the dataset. For example, the aircraft, train, cow, and boat classes have very concrete backgrounds, such as the horizontal plane for boats, the sky for birds and airplanes, and the road for cars. In addition, the examples of birds are difficult to detect because they are tiny in the sky. The introduction of a focal loss function allows the model to focus on such difficult categories during training, which can highlight the weights of the difficult samples and cause them to be noticed. We found that the addition of these techniques did not result in a significant increase in the number of parameters in the model. The experimental results in Table 1 show that the proposed MS-DSSD model improves the performance of small object detection of DSSD, and better accuracy was gained for classes in a specific context.

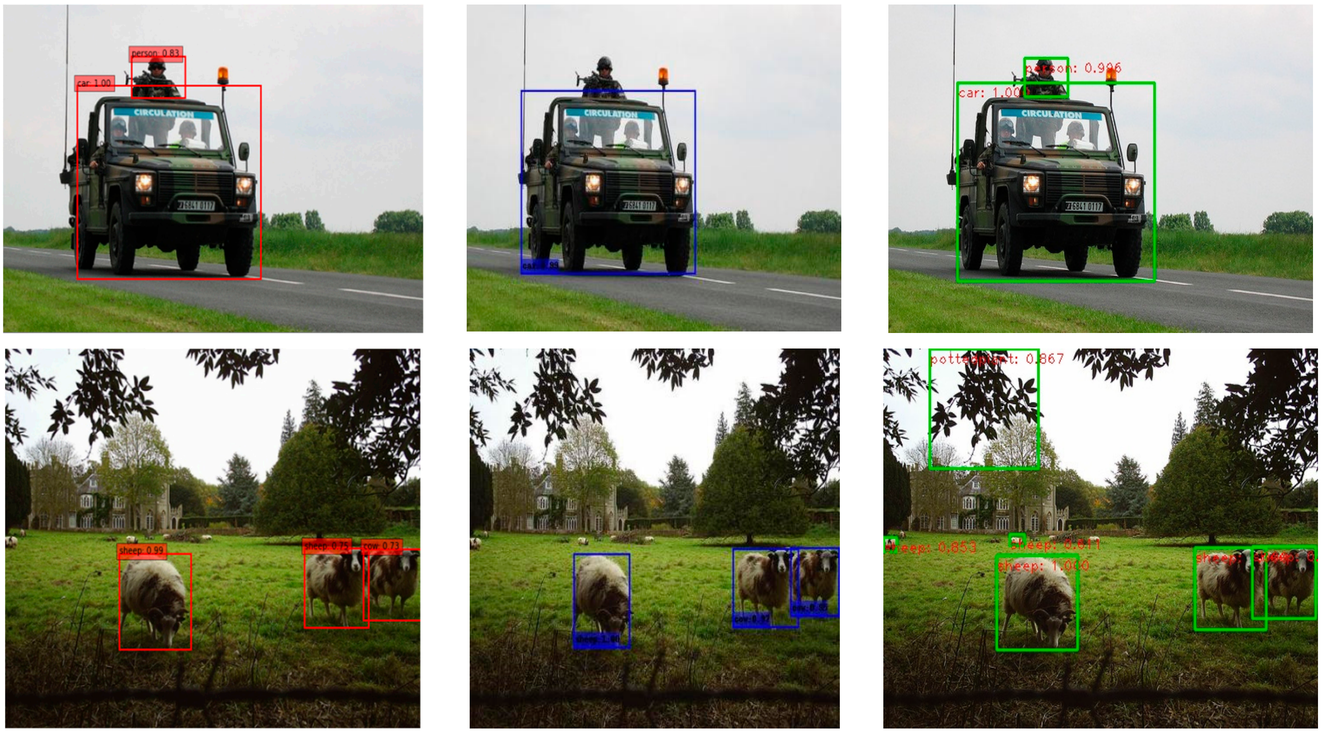

Figure 8. *Cont.*

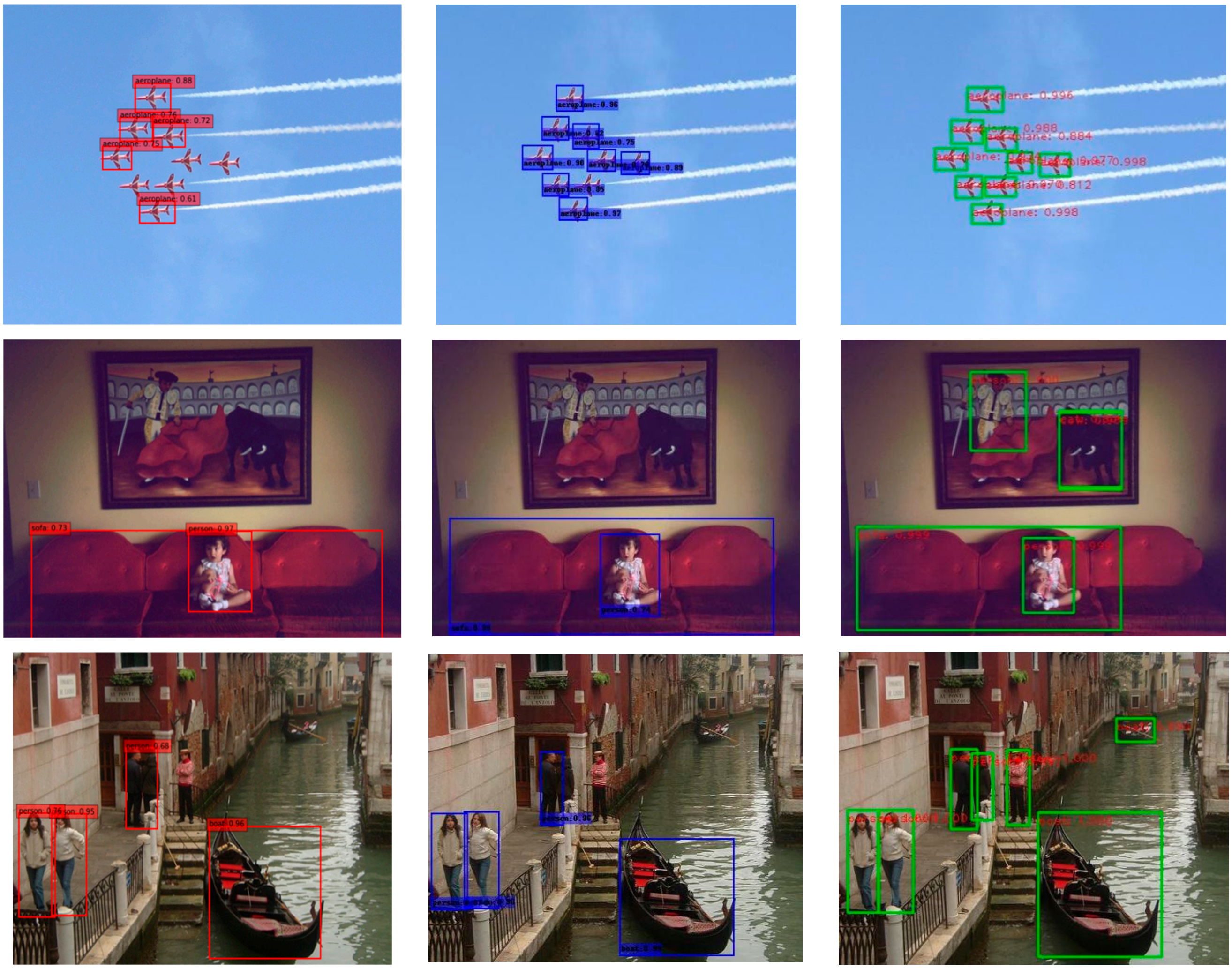

Figure 8. PASCAL 2007 test detection results (red detection boxes on the left show SSD321 detection results, blue boxes in the middle show DSSD321, and green detection boxes on the right show MS-DSSD321 detection results).

5.2. *IST-Waste-V2*

Since no publicly available dataset exists for coastal litter identification, the figures presented in this work were captured by CameraWare. Various images are contained in the IST-Waste dataset, comprising six different categories (plastic, glass, paper, cigarette butts, metal, and wood). The quantity of classes varies depending on conditions, such as weather (rainy, sunny), brightness (luminance, shadows), and blockage by other factors. One image can contain multiple classes with distinct shapes and sizes. Plastic and paper are more frequently seen in daily life than the other categories, so their targets are more numerous than any other category. The amount of objects in terms of each category in the dataset is shown in Table 2. During the training period, we separated the samples into training components (80%) and testing components (20%).

Table 2. Numbers of each sample.

No	Classes	Number of Images
1	Plastic	4789
2	Metal	405
3	Paper	1740
4	Cigarette Butts	389
5	Wood	429
6	Glass	254

We set the learning rate to 1×10^{-3} for the first 70 k iterations and kept training 1×10^{-4} for 20 k iterations and 1×10^{-5} for 20 k iterations. The batch size of MS-DSSD321 and MS-DSSD513 was set to 16 and 12, respectively. Our experiments show that a batch size of less than 10 is likely to give inconsistent results if trained on 4 GPUs.

Table 3 shows the results of the IST-Waste-V2 detection test. The table shows that SSD321 is more accurate than Faster R-CNN. This is because SDD uses a multiscale detection technique than can obtain more complete characteristics. DSSD321 is 0.3% lower than DSSD513, which indicates that the size of the input image affects the detection performance, i.e., the larger the input image size, the higher mAP. MS-DSSD321 and MS-DSSD513 also show this phenomenon when comparing the model performance. MS-DSSD513 demonstrates the best performance, with a significant improvement of 2.6% over DSSD513 in mAP. We think this is mainly due to improvements of the DSSD model in this work in focal loss, dense block, and feature fusion technique, which allow the detection performance of identifying small and hard-to-detect objects to be greatly increased. From the perspective of sample difficulty classification, focal loss makes loss focus on difficult samples and solves the problem of low classification accuracy for categories with few samples. Dense block enhances feature propagation, while reducing the number of parameters. Feature fusion can better utilize features with different characteristics to improve the performance of the model. For example, some objects in waste are sometimes difficult to identify with the naked eye because of their small size. However, the table shows that MS-DSSD513 performs 2.9%, 2.6%, 4.0%, and 4.2% better on these objects than DSSD321, DSSD513, SSD321, and Faster R-CNN, respectively. It is worth noting that the size of the input image is proportional to the training and inference time of the model.

Table 3. IST-Waste-V2 Test detection results.

Method	mAP	Plastic	Metal	Paper	Butt	Wood	Glass
Faster R-CNN (VGG)	79.8	88.1	85.4	88.3	67.6	59.8	89.5
SSD321	80.1	89.1	85.3	88.9	61.5	67.1	92.9
DSSD321	81.2	87.7	86.1	88.4	70.2	62.1	92.7
DSSD513	81.5	88.4	86.6	87.5	71.2	61.8	93.3
MS-DSSD 321	82.3	89.5	88.2	89.6	73.7	62.1	93.8
MS-DSSD 513	84.1	91.5	89.0	91.2	73.9	63.8	95.2

Figure 9 shows a graph of the test results for our database. MS-DSSD also performs better for coastal trash data detection. The detection results show that MS-DSSD outperforms SSD and DSSD in terms of detection accuracy for both small objects (e.g., cigarette butts and glass fragments of different sizes) and large objects (e.g., aircraft). In addition, all three models have a long way to go in detection when the objects in images show a dense and stacked pattern.

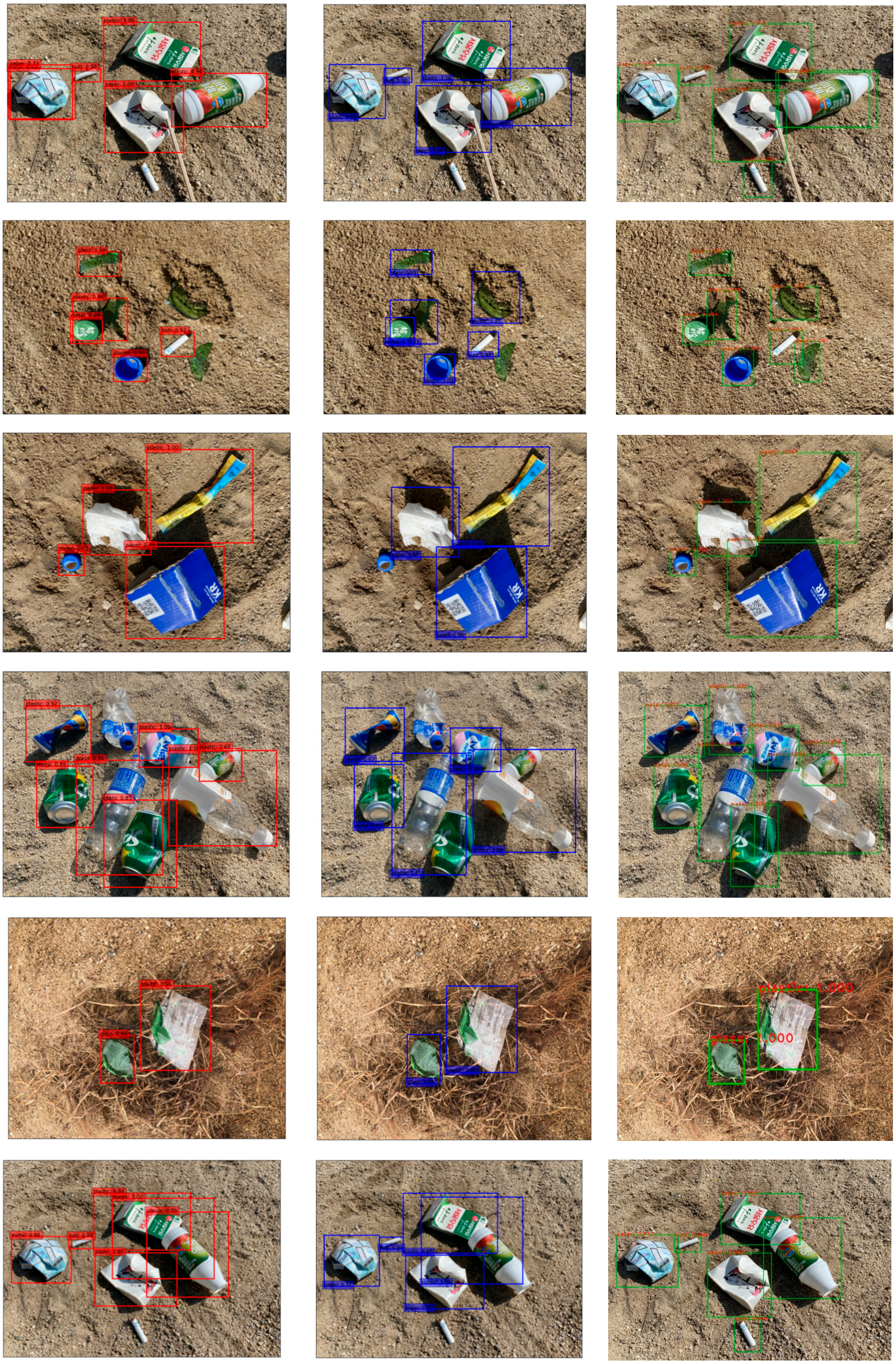

Figure 9. IST-Waste-V2 test detection results (red detection boxes on the left show SSD321 detection results, blue boxes in the middle show DSSD321 detection results, and green detection boxes on the right show MS-DSSD321 detection results).

6. Conclusions

This research sets new benchmarks for six categories of more than 3000 coastal images. Each image contains one or more types of waste. This can be used for various tasks in waste management, including multi-label waste classification and waste identification and localization in images, which has great potential to advance coastal waste management through deep learning approaches. In this work, we proposed the MS-DSSD model for coastal waste detection. The model consists of a sequence of convolution and deconvolution layers taking the famous Residual-101 as the backbone. It also includes feature fusion and dense blocks to bring spatial contextual information and robust features. In response to the difficulty of detecting small-scale objects, we fuse high-level features with low-level features to obtain more beneficial detection information. Moreover, the introduction of dense blocks increases the model's performance by reusing features and enhancing feature propagation. The primary purpose of using focal loss is to balance class weights so that hard samples can be found more easily. The experimental results show that the proposed model achieves a mAP of 82.2% and 84.1% on PASCAL VOC2007 and our dataset, demonstrating our proposed model's improved performance concerning DSSD. The model can also be applied to the automatic detection of coastal waste to manage the marine environment.

Nevertheless, there are many limitations to overcome in the proposed approach. Firstly, the existing datasets are inadequate, which makes it difficult to generalize them for use in engineering applications. We plan to collect various datasets, including common coastal waste such as plastic product waste and metal waste. Secondly, the question of which layer to use for the feature fusion method is left to be addressed. Next, we shall look into more effective image enhancement methods with stronger generalization ability. Addressing these limitations will help the model to be further optimized, thus improving the detection speed. The semantic segmentation of marine litter is another area of research to pursue. We expect that global sustainability and humanity will stand to benefit from this advancement in technology.

Author Contributions: Conceptualization, C.R. and G.Z.; methodology, S.L.; software, D.-K.K.; validation, C.R., D.J. and G.Z.; formal analysis, S.L.; investigation, D.-K.K.; resources, G.Z.; data curation, D.J.; writing—original draft preparation, C.R.; writing—review and editing, C.R.; visualization, D.J.; supervision, S.L.; project administration, D.J. and G.Z.; funding acquisition, S.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Research Foundation of Korea (NRF); grant funded by the Korean government (MSIP) (No. 2019R1I1A3A01060826).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data used to support the findings of this study are available from the corresponding author upon request.

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

References

1. Asensio-Montesinos, F.; Anfuso, G.; Williams, A. Beach litter distribution along the western Mediterranean coast of Spain. *Mar. Pollut. Bull.* **2019**, *141*, 119–126. [CrossRef] [PubMed]
2. Nachite, D.; Maziane, F.; Anfuso, G.; Williams, A.T. Spatial and temporal variations of litter at the Mediterranean beaches of Morocco mainly due to beach users. *Ocean Coast. Manag.* **2019**, *179*, 104846. [CrossRef]
3. Willis, K.; Hardesty, B.D.; Vince, J.; Wilcox, C. Local waste management successfully reduces coastal plastic pollution. *One Earth.* **2022**, *6*, 666–676. [CrossRef]
4. Dalal, N.; Triggs, B. Histograms of oriented gradients for human detection. In Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA, 20–25 June 2005; pp. 886–893.
5. Felzenszwalb, P.; McAllester, D.; Ramanan, D. A discriminatively trained, multiscale, deformable part model. In Proceedings of the 2008 IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, AK, USA, 23–28 June 2008; pp. 1–8.

6. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 580–587. [CrossRef]
7. Girshick, R. Fast r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 11–18 December 2015; pp. 1440–1448.
8. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. *Adv. Neural Inf. Processing Syst.* **2015**, *28*, 91–99. [CrossRef]
9. Lin, T.-Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature pyramid networks for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2117–2125.
10. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 779–788.
11. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. Ssd: Single shot multibox detector. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2016; pp. 21–37.
12. Wang, Y.; Wang, C.; Zhang, H.; Dong, Y.; Wei, S. Automatic Ship Detection Based on RetinaNet Using Multi-Resolution Gaofen-3 Imagery. *Remote Sens.* **2019**, *11*, 531. [CrossRef]
13. Ma, W.; Wang, X.; Yu, J. A Lightweight Feature Fusion Single Shot Multibox Detector for Garbage Detection. *IEEE Access* **2020**, *8*, 188577–188586. [CrossRef]
14. Panwar, H.; Gupta, P.; Siddiqui, M.K.; Morales-Menendez, R.; Bhardwaj, P.; Sharma, S.; Sarker, I.H. AquaVision: Automating the detection of waste in water bodies using deep transfer learning. *Case Stud. Chem. Environ. Eng.* **2020**, *2*, 100026. [CrossRef]
15. Shi, C.; Xia, R.; Wang, L. A Novel Multi-Branch Channel Expansion Network for Garbage Image Classification. *IEEE Access* **2020**, *8*, 154436–154452. [CrossRef]
16. Toğaçar, M.; Ergen, B.; Cömert, Z. Waste classification using AutoEncoder network with integrated feature selection method in convolutional neural network models. *Measurement* **2020**, *153*, 107459. [CrossRef]
17. Yi, H.S.; Chellappan, S. Computer Vision Assisted Approaches to Detect Street Garbage from Citizen Generated Imagery. In *International Summit Smart City 360°*; Springer: Cham, Switzerland, 2021; pp. 526–541.
18. Nazerdeylami, A.; Majidi, B.; Movaghar, A. Autonomous litter surveying and human activity monitoring for governance intelligence in coastal eco-cyber-physical systems. *Ocean Coast. Manag.* **2021**, *200*, 105478. [CrossRef]
19. Kraft, M.; Piechocki, M.; Ptak, B.; Walas, K. Autonomous, Onboard Vision-Based Trash and Litter Detection in Low Altitude Aerial Images Collected by an Unmanned Aerial Vehicle. *Remote Sens.* **2021**, *13*, 965. [CrossRef]
20. Li, J.; Liang, X.; Shen, S.; Xu, T.; Feng, J.; Yan, S. Scale-aware Fast R-CNN for Pedestrian Detection. *IEEE Trans. Multimed.* **2017**, *20*, 985–996. [CrossRef]
21. He, K.; Zhang, X.; Ren, S.; Sun, J. Spatial pyramid pooling in deep convolutional networks for visual recognition. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*; IEEE: Piscataway, NJ, USA, 2015; Volime 37, pp. 1904–1916.
22. Mnih, V.; Heess, N.; Graves, A. Recurrent models of visual attention. *Adv. Neural Inf. Processing Syst.* **2014**, *27*, 2204–2212.
23. Larochelle, H.; Hinton, G.E. Learning to combine foveal glimpses with a third-order Boltzmann machine. *Adv. Neural Inf. Processing Syst.* **2010**, *23*, 1243–1251.
24. Chen, L.-C.; Yang, Y.; Wang, J.; Xu, W.; Yuille, A.L. Attention to scale: Scale-aware semantic image segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 3640–3649.
25. Gregor, K.; Danihelka, I.; Graves, A.; Rezende, D.; Wierstra, D. Draw: A recurrent neural network for image generation. In Proceedings of the 32nd International Conference on Machine Learning, PMLR, Lille, France, 6–11 July 2015; pp. 1462–1471.
26. Fu, C.Y.; Liu, W.; Ranga, A.; Tyagi, A.; Berg, A.C. Dssd: Deconvolutional single shot detector. *arXiv* **2017**, arXiv:1701.06659.
27. Redmon, J.; Farhadi, A. YOLO9000: Better, faster, stronger. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7263–7271.
28. Sermanet, P.; Eigen, D.; Zhang, X.; Mathieu, M.; Fergus, R.; LeCun, Y. Overfeat: Integrated recognition, localization and detection using convolutional networks. *arXiv* **2013**, arXiv:1312.6229.
29. Lin, T.-Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal loss for dense object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2980–2988.
30. Teichmann, M.; Weber, M.; Zollner, M.; Cipolla, R.; Urtasun, R. MultiNet: Real-time Joint Semantic Reasoning for Autonomous Driving. In Proceedings of the 2018 IEEE Intelligent Vehicles Symposium (IV), Changshu, China, 26–30 June 2018; pp. 1013–1020.
31. Quan, T.M.; Hilderbrand, D.G.C.; Jeong, W. FusionNet: A deep fully residual convolutional neural network for image segmentation in connectomics. *arXiv* **2016**, arXiv:1612.05360. [CrossRef]