future internet

MDPI

Article

Dynamic Detection and Recognition of Objects Based on Sequential RGB Images

Shuai Dong [1], Zhihua Yang [1,2], Wensheng Li [1] and Kun Zou [1,*]

[1] Artificial Intelligence and Computer Vision Laboratory, Zhongshan Institute, University of Electronic Science and Technology of China, Zhongshan 528402, China; dongshuai@zsc.edu.cn (S.D.); 2112004214@mail2.gdut.edu.cn (Z.Y.); lws@zsc.edu.cn (W.L.)
[2] School of Automation, Guangdong University of Technology, Guangzhou 510006, China
* Correspondence: cszoukun@zsc.edu.cn

Abstract: Conveyors are used commonly in industrial production lines and automated sorting systems. Many applications require fast, reliable, and dynamic detection and recognition for the objects on conveyors. Aiming at this goal, we design a framework that involves three subtasks: one-class instance segmentation (OCIS), multiobject tracking (MOT), and zero-shot fine-grained recognition of 3D objects (ZSFGR3D). A new level set map network (LSMNet) and a multiview redundancy-free feature network (MVRFFNet) are proposed for the first and third subtasks, respectively. The level set map (LSM) is used to annotate instances instead of the traditional multichannel binary mask, and each peak of the LSM represents one instance. Based on the LSM, LSMNet can adopt a pix2pix architecture to segment instances. MVRFFNet is a generalized zero-shot learning (GZSL) framework based on the Wasserstein generative adversarial network for 3D object recognition. Multi-view features of an object are combined into a compact registered feature. By treating the registered features as the category attribution in the GZSL setting, MVRFFNet learns a mapping function that maps original retrieve features into a new redundancy-free feature space. To validate the performance of the proposed methods, a segmentation dataset and a fine-grained classification dataset about objects on a conveyor are established. Experimental results on these datasets show that LSMNet can achieve a recalling accuracy close to the light instance segmentation framework You Only Look At CoefficienTs (YOLACT), while its computing speed on an NVIDIA GTX1660TI GPU is 80 fps, which is much faster than YOLACT's 25 fps. Redundancy-free features generated by MVRFFNet perform much better than original features in the retrieval task.

Keywords: one-class instance segmentation; level set map; multiview feature; fine-grained recognition; generalized zero-shot learning

Citation: Dong, S.; Yang, Z.; Li, W.; Zou, K. Dynamic Detection and Recognition of Objects Based on Sequential RGB Images. *Future Internet* **2021**, *13*, 176. https://doi.org/10.3390/fi13070176

Academic Editors: Remus Brad and Arpad Gellert

Received: 29 May 2021
Accepted: 6 July 2021
Published: 7 July 2021

1. Introduction

Vision-based localization and recognition of objects on a conveyor (LROC) is an important type of application in the industry. In a typical scenario, such as vision-based automatic check-out in unmanned supermarkets or automatic sorting in factories, all categories are first registered in a database (saving features). Then, the features of objects on the conveyor are extracted and used to retrieve the categories from the database. It is a fine-grained recognition task in which we have to distinguish not only whether the object is Coca-Cola or milk, but also its flavor, volume, and packaging. The challenge of LROC consists of two large gaps between the registered images and the on-conveyor images. First, different views of a 3D object should be registered together, as they would be different; furthermore, only a group of close views of the object on a conveyor can be captured for prediction. Second, the registered images can be captured in an environment with stable illumination and background (source domain), whereas the on-conveyor images are often captured in open environments (target domain). Multi-view methods [1,2] can bridge the first gap, and domain-aligning methods can deal with the second gap. However, when

objects of new (unseen) categories occur and we cannot obtain their on-conveyor images for training, we need to train the network by learning a map from the source to target the domain that has heterogeneous views different from the seen categories. It is a generalized zero-shot learning (GZSL) problem [3,4].

In a usual detection-based multiobject tracking (MOT) [5,6] task, which mainly focuses on the final tracking performance of a few categories of objects (human, car, or some special categories), the detection module [7–10] is treated as an object detection task with coarse classification branches, and the reidentification module is treated as a fine-grained classification task for apparent features. The latter is an additional matching constraint to relieve ID switching, which is caused by the detection difference between frames. As object detection of unseen objects is one of the most challenging visual tasks and has not been exploited comprehensively, decomposition of localization and recognition would be a better choice. One advantage of decomposition is that some detection-free methods can be used to improve the computing speed of the localization subtask. Another advantage is that zero-shot classification architectures can be used to deal with new unseen categories. Considering the simple texture and the fixed motion direction of conveyors, LROC is a special MOT task with an easier localization subtask but a harder zero-shot learning fine-grained recognition subtask.

In this study, a framework of LROC involving three subtasks (one-class instance segmentation (OCIS), multiobject tracking (MOT), and zero-shot fine-grained recognition of 3D objects (ZSFGR3D)) is developed. First, the OCIS module segments all objects on the conveyor. Then, the MOT module tracks the motion trajectory of all objects with the masks obtained by the OCIS module. Finally, the ZSFGR3D module retrieves the segmented objects from the gallery of registered features. Because there are many effective methods for MOT subtasks, we mainly focus on the first and third subtasks in this study.

In the OCIS subtask, the conveyor is the background, and objects on it all belong to one same category named "Goods". OCIS is a special task different from semantic segmentation in which segmentation of instances is unnecessary. Because instance segmentation frameworks are designed for multiple categories and are extended from object detection frameworks with a lot of anchors, they often have a deep architecture and suffer from low speed. It hinders them from running on some high cost-effective devices [11]. In this study, a level set map neural network (LSMNet) is proposed for the OCIS subtask. LSMNet is inspired by the level set algorithm, which is a traditional active contour method. In the past three years, two other deep-learning-based active contour methods, level set loss [11] and deep snake [12], have been developed and applied in segmentation tasks. Unlike these methods, which comprise additional active contour constraints on the segmentation losses to improve the quality of masks indirectly, LSMNet utilizes the level set map (LSM) directly. The annotation of instance segmentation is often in the form of a multichannel binary image, in which each channel is the mask of one object. In the proposed method, we use a single level set map (LSM) to annotate the image instead of the multichannel binary image, where each peak of the LSM represents one object. LSMNet adopts a semantic segmentation architecture, for instance, a UNet or a pix2pix GAN, to predict LSMs. Once the LSM of an image is accurately predicted, the objects can be localized according to the peaks and their areas. An OCIS dataset about on-conveyor objects is established to validate the performance of LSMNet. An experiment on a subset of COCO2017 [13] is also conducted. The results show that LSMNet can achieve an accuracy close to You Only Look At CoefficienTs (YOLACT) [14], which is one of the fastest instance segmentation methods, on big and medium objects with a much higher speed. On the NVIDIA GTX1660TI GPU, the speed of LSMNet is 80 fps, and that of YOLACT is 25 fps.

For the FGR3D subtask, the registered multiview features are treated as the semantic attributes of every category, the features of old on-conveyor objects as the seen categories, and the features of new on-conveyor objects as the unseen categories. A multiview redundancy-free feature network (MVRFFNet) is proposed to learn the map from multiview features to single view on-conveyor features based on the GZSL framework proposed by Han et al. [3].

Multi-view features of an object are extracted with a pretrained network and combined into a compact registered feature. The registered features are the attributes for training. A Wasserstein generative adversarial network (WGAN) [15] is trained to learn a map function that maps the original retrieve features to a new redundancy-free feature space [1].

The proposed method has two advantages. First, LSMNet realizes instance segmentation with a semantic segmentation network by introducing the innovative LSM annotation. It has a much higher speed than YOLACT and a close performance on objects of big and medium size. Second, the redundancy-free feature is introduced into the multiview framework for fine-grained 3D object recognition. The redundancy-free features generated by MVRFFNet perform much better than the original features in the matching task.

This paper is organized as follows. Section 2 reviews the related work. Section 3 proposes the main algorithms. Experiments are presented in Section 4. At last, some conductions are drawn in Section 5.

2. Related Works

Instance segmentation is the most difficult task among the four classic visual tasks [16], which include classification, localization, detection, and segmentation. Its target is to obtain the pixel-level segmentation of individual objects, which combines the requirements of semantic segmentation and object detection. Over the past few years, deep learning has yielded a new generation of instance segmentation models with remarkable performance improvements and results in a paradigm shift in the field. There are two types of deep learning models for instance segmentation: two-stage models and one-stage models.

The most important two-stage method is Mask-RNN [17] proposed by He et al., which is extended from their earlier important work Faster RCNN [18]. Although two-stage methods have better performance, their computing burden is too heavy to run on some embedded devices. Thus, some researchers have proposed several excellent one-stage methods to reduce the computing burdens under the inspiration of the one-stage object detection frameworks [7–10]. Dai et al. [19] and Li et al. [20] designed special fully convolutional networks, together with positive-sensitive score maps, to segment instances. Daniel et al. proposed YOLACT [14] and improved it to YOLACT++ [21], which achieved the best balance between speed and accuracy. Xie et al. [22] used a polar mask to annotate the segmentation of an instance, which is a more precise bounding box. CenterMask [23,24] was developed from CenterNet [10] by inheriting the anchor-free ideas. Zhang et al. [25] represented the mask into a two-dimensional vector, which can be combined with the box detection branch. BlenderMask [26] combines the top-down and bottom-up methods based on an anchor-free framework.

Zero-shot learning (ZSL), which is one of the typical transfer learning methods, is perhaps the supreme goal of machine learning. For example, if machines could classify new classes accurately [27,28], we could collect labeled data as much as possible for free; if machines could reject samples of unknown classes [29,30], any recognition system would be shielded against outliers. Specifically, the goal of ZSL is to recognize objects of unseen classes, whose labels are not available, by learning high-level semantic information [26,27,30].

In the pre-deep-learning era, researchers focused on the conventional or standard ZSL, in which all test images come from the unseen classes only. Various semantic embedding methods have been developed based on traditional machine learning technologies [27,31,32]. A semantic embedding method learns to embed the original features into a new semantic descriptor space and then predict the classification of features via matching the most similar semantic descriptor.

In the past five years, the more challenging generalized zero-shot learning [3] (GZSL), in which the test set consists of data from both the seen and unseen classes and semantic embedding performs poorly, has attracted increasing attention. In a GZSL task, the training set only contains annotated objects of seen classes, but the test set contains objects from both seen and unseen classes. The extreme data imbalance of GZSL makes semantic embedding methods apt to be highly overfitted to seen classes and fail to classify the unseen classes [27].

Recently, some feature generation methods have been proposed to compensate for the lack of training images of unseen classes in GZSL. Bucher et al. [33] generated features of unseen classes with four different generative models, including generative moment matching network, auxiliary classifier GANs, denoising autoencoder, and adversarial autoencoder. The f-CLSWGAN has also been utilized to generate the unseen features conditioned on the class-level semantic descriptors [34]. Some methods [35,36] introduced reverse regressor networks into the generator network in the form of a cycle-consistent loss or to constrain the feature [37]. Verma et al. [38] designed a variational autoencoder to achieve the same function as f-CLSWGAN. Han et al. [3] proposed a redundancy-free feature generation framework that "limits the information dependence between the mapped features and the original features of the images to an upper bound". In the redundancy-free space, the overfitting problem can be restrained. Besides the generative models, some other innovative methods have also been proposed. Chen et al. [39] designed a semantic-preserving adversarial embedding network to avoid the loss of semantic information. Liu et al. [40] simultaneously calibrated the model confidence of seen classes and the model uncertainty of unseen classes with a special calibration network. Inspired by the information bottleneck method [40], an innovative counterfactual framework to balance seen and unseen classifications was proposed by Yue et al. [41].

3. Framework of the System

The LROC task consists of three subtasks: OSIC, MOT, and ZSFGR3D. As several mature methods can be used to accomplish the MOT subtask, we focused on the first and the third subtasks in this work. The hardware system and the overall architecture of the proposed method are introduced first in this section. Then, LSMNet and MVRFFNet are proposed to finish the first and third subtasks.

A simple visual recognition system for on-conveyor objects is illustrated in Figure 1. Serial images are captured by an RGB industrial camera mounted above the conveyor. The frame rate is about 30 fps, and the speed of the conveyor is about 10 cm/s.

(**a**) Conveyor (**b**) Camera in the black box

Figure 1. Conveyor system.

The overall architecture of the proposed method is depicted in Figure 2. It consists of three main modules: LSMNet, MOT tracker, and MVRFFNet. LSMNet generates instance segmentation results for the MOT tracker, and the tracker provides segmented retrieval images for the MVRFFNet. We can choose several frames for an object to improve its retrieval accuracy. In the inferring process, a matching network is used to retrieve the redundancy-free features of the on-conveyor objects from the gallery of registered multi-view features. Other metric learning methods, for instance, the cosine or Euclid distance, can be used to replace the matching network. When some new objects occur, we only need to register their multiview features without any additional work.

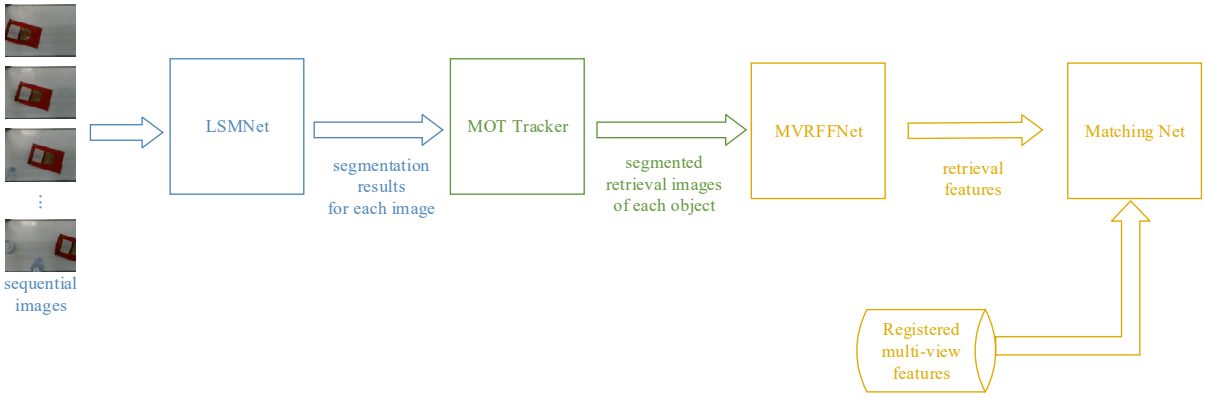


Figure 2. Overall architecture of the proposed method.

3.1. LSMNet

For an image X and its annotation set $\mathcal{M} = \{M_1^0, M_2^0, \cdots\}$, in which M_k^0 is the binary mask for the $k-th$ instance. The elements of M_k in the segmentation region are 1, and the remaining elements are 0. Let

$$M_k^{n+1} = erode(M_k^n, \Lambda) \tag{1}$$

where $erode(\cdot, \Lambda)$ is the eroding operation in morphology with the kernel Λ. The region of M_k^{n+1} lies in the contour of M_k^n. Then, the LSM of instance k can be obtained according to

$$M_k^{LSM} = 127 \cdot M_k^0 + h \cdot \sum_{k=1}^{K} M_i^n \tag{2}$$

where K is the max step of eroding operation, and h is the biggest interval of contour lines. We can normalize M_k^{LSM} to $[0, 255]$ in the following manner

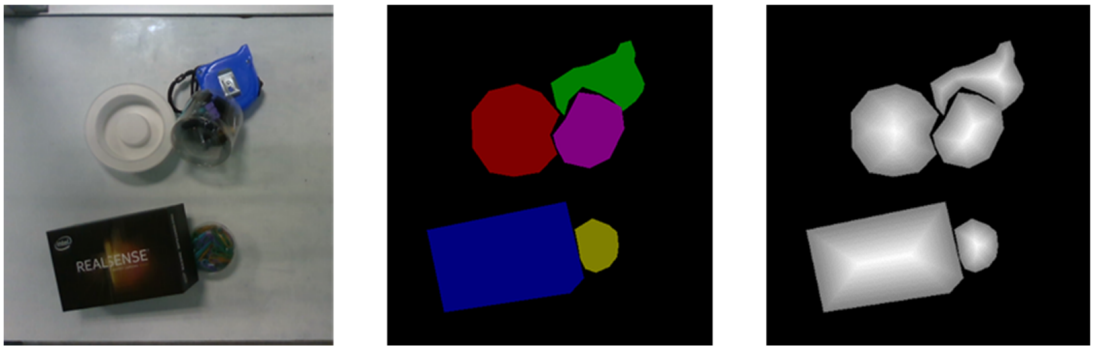
$$\overline{M}_k^{LSM} = \left(127 + \frac{128 \cdot (M_k^{LSM} - 127)}{\max\{M_k^{LSM}\} - 127}\right) / 255 \tag{3}$$

where $\max\{\cdot\}$ returns the maximum element of a 2D matrix.

Denote the final LSM of X as $\mathcal{M}^{LSM}$, then its element at (i, j) is calculated as below

$$\mathcal{M}_{i,j}^{LSM} = \max_k \left\{ (\overline{M}_k^{LSM})_{i,j} \right\} \tag{4}$$

A group of samples is depicted in Figure 3. In $\mathcal{M}^{LSM}$, the elements out of contours are 0, and elements on the contours are equal to 127. We can set $K = 1000$, and stop Iteration (2) when M_k^{n+1} has been eroded totally; that is $sum(M_k^{n+1}) = 0$. In an LSM, peaks represent objects and valleys represent the background. Once the LSM is predicted accurately, an appropriate threshold can be used to segment the peaks conveniently.



(**a**) Image X (**b**) Visualization of the annotation (**c**) $\mathcal{M}^{LSM}$

Figure 3. LSM with $K = 1000$ and $h = 2$.

It is a typical pixel-level image translation task to predict $\mathcal{M}^{LSM}$ with X, and any pix2pix architecture can be adopted. In this study, we chose an UNet as the basic architecture. The UNet contains 16 convolutional layers, where the number of convolutional kernels increases from 16 to 512 in the first 8 layers and decreases to 1 in the last 8 layers. The scale of downsampling and upsampling is 2, the input size is $256 \times 256 \times 3$, and the output size of each branch is $256 \times 256 \times 1$. To improve the quality of the LSM, we trained an LSM branch and a semantic segmentation branch simultaneously. Those two decoders have the same architecture.

The framework of LSMNet is depicted in Figure 4. In the training process, the LSM loss and mask loss, together with a consistent constraint loss, were calculated and used to update the backbone. In the inferring process, small noisy regions of $\mathcal{M}^{LSM}$ were filtered first, and the foreground regions were segmented with a threshold > 0.5. Each region corresponds to an instance, and the instance regions are dilated in a ratio proportional to its area.

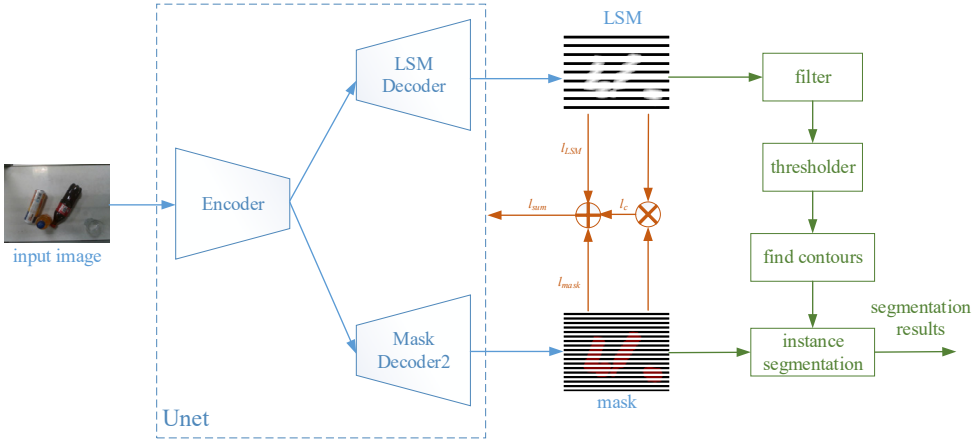


Figure 4. Framework of LSMNet.

Denote the ground truth of the semantic segmentation mask as $\mathcal{M}^{mask}$. Its elements are calculated according to

$$\mathcal{M}^{mask}_{i,j} = \max_k \left\{ \left(M^0_k \right)_{i,j} \right\} \tag{5}$$

The output LSM and mask of LSMNet are written as $\widetilde{\mathcal{M}}^{LSM}$ and $\widetilde{\mathcal{M}}^{mask}$. Then, the final lost function consists of three sublosses as below

$$\begin{aligned} l_{sum} &= l_{LSM} + l_{mask} + l_c \\ l_{LSM} &= \omega_1 \cdot \left| \mathcal{M}^{LSM} - \widetilde{\mathcal{M}}^{LSM} \right| \\ l_{mask} &= \omega_2 \cdot \left| \mathcal{M}^{mask} - \widetilde{\mathcal{M}}^{mask} \right| \\ l_c &= \omega_3 \cdot \left| \left(2\widetilde{\mathcal{M}}^{mask} - 1 \right) \otimes \left(\widetilde{\mathcal{M}}^{mask} - 0.5 \right) - \left(\mathcal{M}^{mask} - 0.5 \right) \right| \end{aligned} \tag{6}$$

in which $|\cdot|$ is the L1 norm, $\otimes$ is the element-wise multiplying operation, and ω_1, ω_2, and ω_3 are the weights for sublosses. l_{LSM} and l_{mask} are the L1-loss about LSM and mask, respectively, and l_c is an additional constraint to make $\widetilde{\mathcal{M}}^{LSM}$ and $\widetilde{\mathcal{M}}^{mask}$ consistent.

3.2. MVRFFNet

MVRFFNet consists of two modules: the registering module and the feature mapping module. The former extracts multiview features of an object and combines them into a compact registered feature. The latter uses a WGAN-based framework [3] to bridge the gap between registered features and on-conveyor features.

Denote the registering view set of N objects as $\mathcal{V} = \{v_{i,j}, i = 1, \cdots, N, j = 1, \cdots, M\}$, the seen conveyor view set as $\mathcal{U} = \{u_{i,k}, i = 1, 2, \cdots, N_1, j = 1, 2, \cdots, \infty\}$, and the unseen conveyor view set as $\mathcal{S} = \{s_{i,k}, i = 1, 2, \cdots, N_2, j = 1, 2, \cdots, \infty\}$, where M is the number of views, and N satisfies $N = N_1 + N_2$. In the registering module, features of different views are extracted separately with a feature extractor R first; that is $s_{i,j} = R(v_{i,j})$. $s_{i,j}$ are fused to a registered feature r_i with a fusing network F; that is $r_i = F(s_{i,1}, s_{i,2}, \cdots, s_{i,M})$. F could be a simple max-pooling layer [2] or a recurrent neural network (RNN) [1] that evolves along with the variation of view. A graphic neural network (GNN) is also a qualified candidate fuser. E_1 is trained separately, and F is trained together with the mapping module. The architecture is depicted in Figure 5.

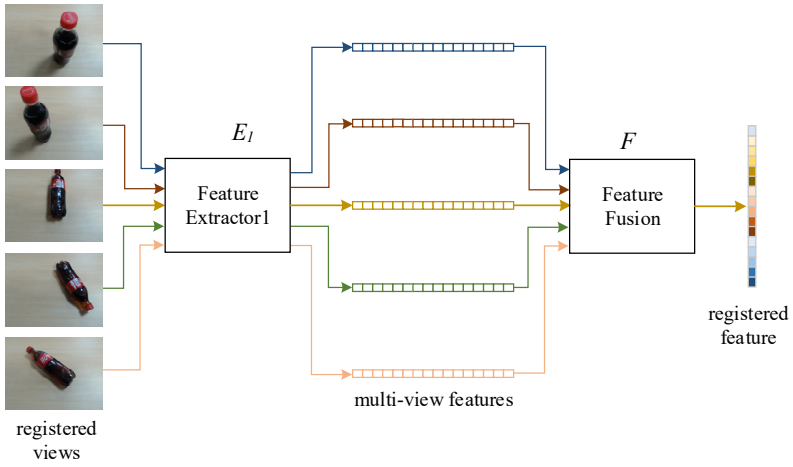


Figure 5. Architecture of the registering module of MVRFFNet.

In the mapping module, the redundancy-free feature mapping framework [3] is adopted. The structure is depicted in Figure 6. The generator G, with the concatenation of a registered feature and a noise vector as its input, generates a synthetic or fake on-conveyor feature. The noise represents the properties of difference between the registered and on-conveyor features, which includes environment light, background noise, the motion of the camera, and multiview modal to single-view modal. The mapping function M maps the on-conveyor features into a latent space, which is the redundancy-free feature space. D is the discriminator that is realized in the form of Wasserstein distance. Wasserstein distance is a symmetrical measurement for the difference between two random distributions. E_2 is

a feature extractor to obtain features from on-conveyor images, and C is a final classifier to predict the categories of latent features.

Besides the usual fake loss and Wasserstein distance in WGAN, the mutual information (MI) loss based on Kullback–Leibler divergence and the center loss [3] are also considered when training G and M.

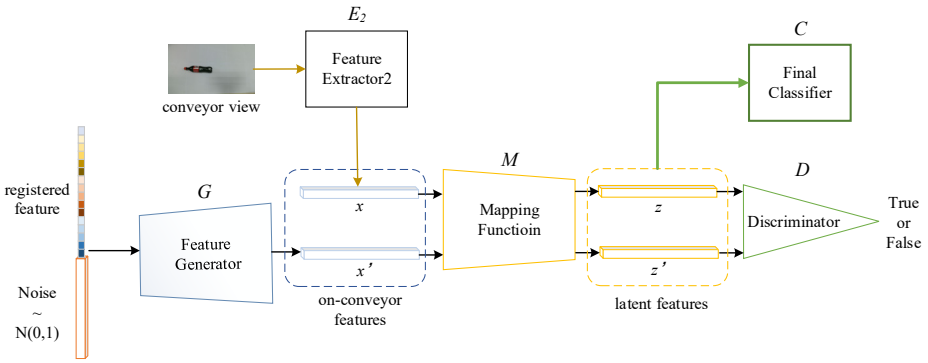


Figure 6. Structure of the feature mapping module.

The training configuration of MVRFFNet is similar to that of [3], except that the attribute vector is replaced with a registering module and F is updated together with G. E_2 is trained separately, and it is frozen when training other parts. In each training batch, F, G, and M are updated once or several times simultaneously first; and then, M is frozen while F and G are updated at the same time. At the last step of a training epoch, F, G, and M are frozen, and C is trained separately. The details about loss functions and training strategies can be referred to in [3].

4. Experiments and Analysis

In this section, we established an OCIS dataset and a fine-grained recognition dataset first, and experiments were conducted on them to validate the performance of LSMNet and MVRFFNet. LSMNet was also tested on the COCO-car dataset, which is a subset of the open dataset COCO2017 [13].

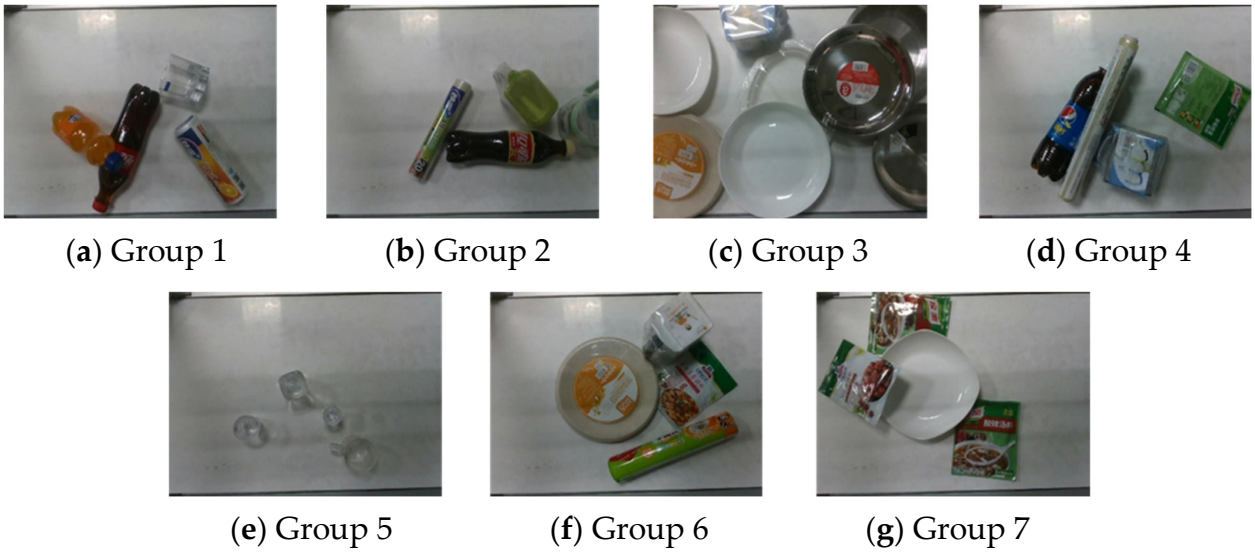
4.1. Datasets

4.1.1. OCIS Dataset and Fine-Grained Recognition Dataset

The distribution of the OCIS dataset is listed in Table 1. The objects in the test set are different from those in the training set. Some groups of samples are depicted in Figures 7 and 8. In addition, we collected 15 video clips of different levels of difficulty. Level 1 means that there is a distance between any two objects, Level 2 means that objects flock without any occlusion, and Level 3 means that there exist overlaps among objects. Due to the smooth surface of the conveyor, the reflection of illumination causes conspicuous noise.

Table 1. Distribution of the OCIS dataset.

	No. of Images/Videos	No. of Instances
Training Set	98	503
Test Set	18	105
Validation Set	Level 1: 5 clip	105
	Level 2: 5 clip	105
	Level 3: 5 clip	105



(**a**) Group 1 (**b**) Group 2 (**c**) Group 3 (**d**) Group 4

(**e**) Group 5 (**f**) Group 6 (**g**) Group 7

Figure 7. Objects in the training set of the OCIS dataset.

(**a**) Group 1 (**b**) Group 2 (**c**) Group 3 (**d**) Group 4

Figure 8. Objects in the test set of the OCIS dataset.

Table 2 presents the distribution of the fine-grained recognition dataset. The objects are the same as those in the training set of the OCIS dataset, as shown in Figure 5. Registered images were captured in a simpler environment, as shown in Figure 3. Each image only contains one object.

Table 2. Distribution of the fine-grained recognition dataset.

	No. of Categories	No. of On-Conveyor Images	No. of Registered Views
Training Set	34	2380	510
Seen Test Set	34	1020	-
Unseen Test Set	18	1800	70

4.1.2. COCO-Car Dataset

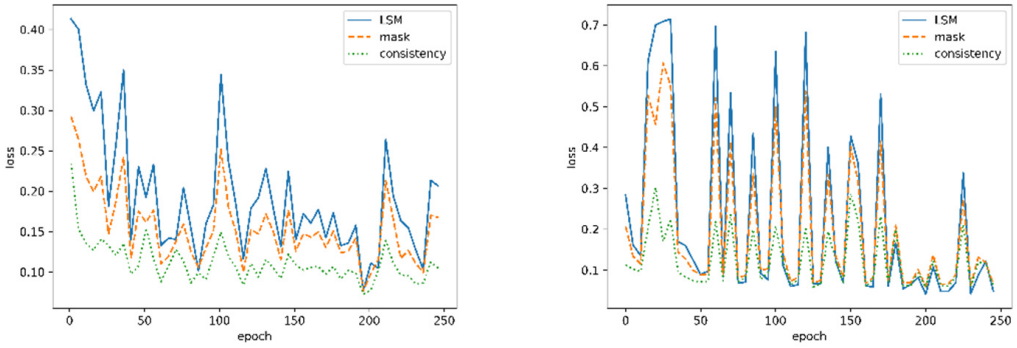
COCO dataset is a large-scale open dataset for object detection and instance segmentation tasks. It contains 80 classes in total. We collected all samples that contain car instances to establish the COCO-car dataset. The distribution of COCO-car is presented in Table 3. The average value of the number of instances in each image was 3.6.

Table 3. Distribution of the COCO-car dataset.

	No. of Images of COCO-Car (COCO2017)	No. of Instances of COCO-Car (COCO2017)
Training Set	12,251 (117,266)	43,867 (860,001)
Validation Set	535 (4952)	1932 (36,781)

4.2. Results of LSMNet

In this experiment, we used a fully convolutional UNet with skipping connection as the basic architecture of LSMNet. The learning rate was set as 0.0002, the batch size was 6, and the weights were $\omega_1 = \omega_2 = \omega_3 = 100$. Cropping, rotation, channel fusion, and color jitter were used to augment the dataset. The curves of the training loss are illustrated in Figure 9. The l_{LSM} curves are above that of l_{mask}, as LSMs are more difficult for the pix2pix network to learn than binary masks. The fluctuations of l_{LSM}, l_{mask}, and l_c are almost synchronous, and it means that the consistent constraint is violated more seriously when the predicted LSM and mask are poor. The fluctuations would be eliminated if a bigger training set is accessible. Because the objects in the test set were new for the model, the loss of the test set was unsurprisingly much larger than that of the training set. In Figure 10, a few samples of the predicted LSMs on the training set are shown. LSM output suffers from small noisy regions (Samples 1, 3, and 5), which are filtered in the consistency outputs. However, some tiny objects are missed. The performance of LSMNet on small objects would be discussed further in the experiment on the COCO-car dataset.



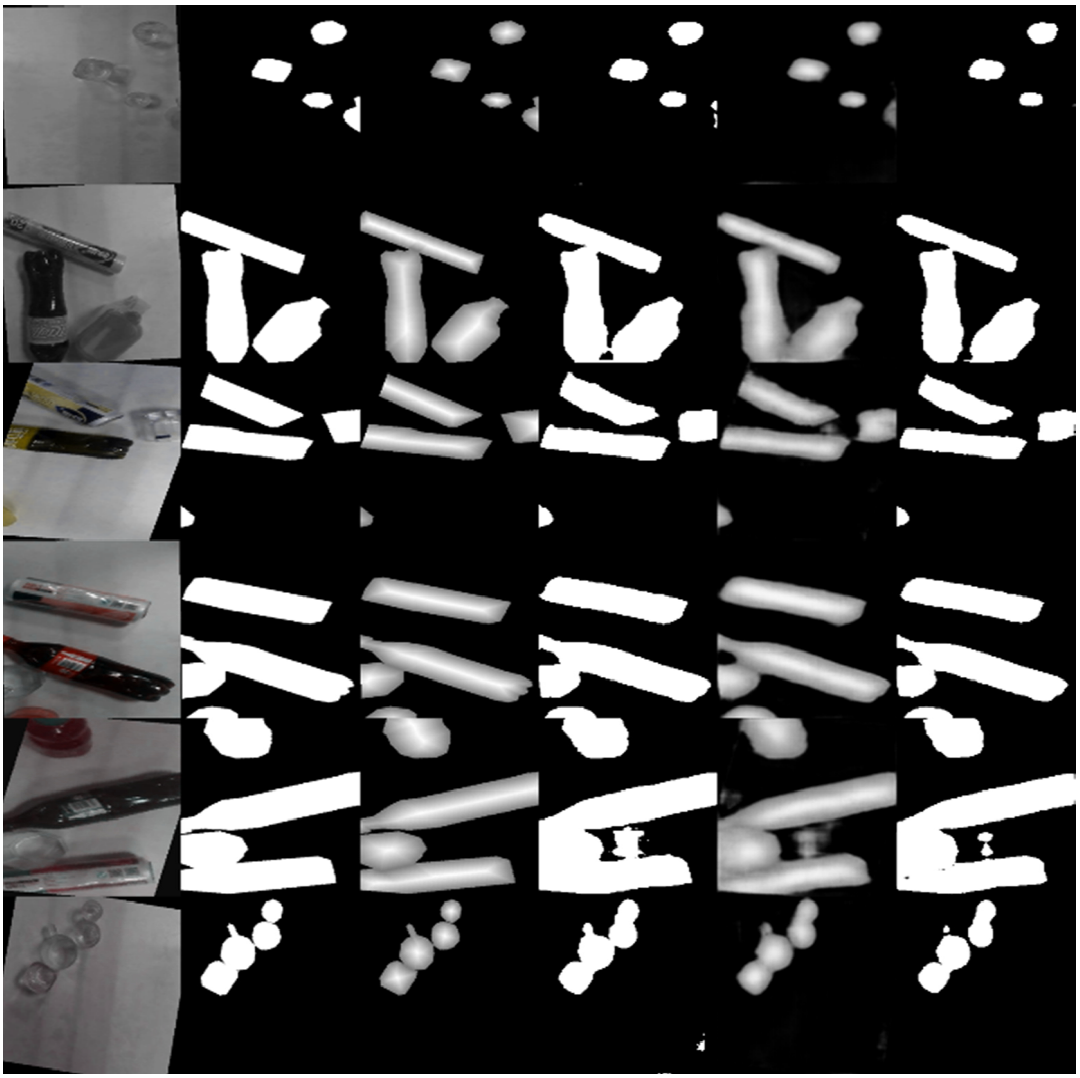
(**a**) Training set (**b**) Test set

Figure 9. Training loss curves of LSMNet on the OCIS dataset.

We compared LSMNet with YOLACT [14] to illustrate its performance. It should be noted that we fine-tuned the model of YOLACT, which is pretrained on COCO2017 [18]. The results of the videos are listed in Table 4. Because YOLACT is developed based on a one-stage object detection framework, the segmentation results are affected by the detection task. Moreover, YOLACT is apt to recall those objects ever seen in COCO2017 due to that the OCIS dataset is much smaller than COCO2017. As shown in Figure 11, some objects with large volumes are missed by YOLACT. LSMNet performs very well if there is no crowding. However, when there exists adjoin or occlusion among objects, it is hard to determine the threshold for extracting the peaks of LSM. Some poor results of LSMNet are presented in Figures 12 and 13.

Table 4. OCIS results on videos.

	Level 1		**Level 2**		**Level 3**	
	TPR	**FPR**	**TPR**	**FPR**	**TPR**	**FPR**
YOLACT	0.7532	0.0987	0.7398	0.1035	0.7794	0.2335
LSMNet	0.9437	0.0224	0.7098	0.1362	0.6277	0.3361



(**a**) (**b**) (**c**) (**d**) (**e**) (**f**)

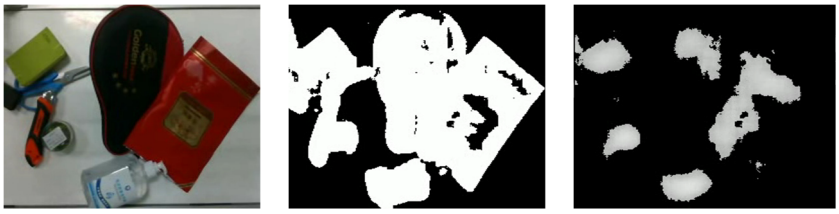
Figure 10. Results of LSMNet on the OSIC dataset. (**a**–**f**) Original image, mask ground truth, LSM ground truth, mask output, LSM output, and consistency output, respectively.

(**a**) Level 2 (**b**) Level 3

Figure 11. Poor results of YOLCACT on videos.

(**a**) Original Image (**b**) Mask (**c**) LSM

Figure 12. Poor results of LSMNet on the Level 2 video.



(**a**) Original Image (**b**) Mask (**c**) LSM

Figure 13. Poor results of LSMNet on the Level 3 video.

To analyze the performance of LSMNet, we compared it with SCNet [41] and YOLACT on the COCO-car dataset. They are trained with two NVIDIA RTX2080TI GPUs based on the open framework mmdetection [42], which is developed by SenseTime. The results on COCO-car are listed in Table 5. AP is the mean AP@IoU = 0.50:0.95, AP_L, AP_M, and AP_S are the mean AP@IoU on cars of the large, medium, and small size, respectively, and fps is observed on an NVIDIA GTX1660TI CPU. SCNet performed much better than the other two methods due to its introduction of the pregenerated stuffthingmaps. However, its cascade architecture made it much slower than the other two methods. LSMNet performed close to YOLACT on large and medium cars, but much poorer on small objects. The reason is that LSMNet extracts pixel-level information first and aggregates it to global semantic information. Noisy regions in the predicted LSM disturbed the segmentation of small objects. As the process of finding contours cannot be realized in the form of a differentiable module, the label information of the bounding boxes cannot be backpropagated to the pix2pix architecture. The performance of LSMNet would be improved if we can find a method to utilize the box information in the training process.

Table 5. OCIS results (AP@IoU = 0.50:0.95) on the COCO-car (COCO) dataset.

	AP	**AP_L**	**AP_M**	**AP_S**	**fps**
SCNet (R-50-FPN)	0.4060 (0.4020)	0.4640 (0.5340)	0.4830 (0.4280)	0.2500 (0.224)	5
Yolact (R-50-FPN)	0.2160 (0.2060)	0.4180 (0.3420)	0.3830 (0.2160)	0.1170 (0.0530)	25
LSMNet	0.1530 (-)	0.4150 (-)	0.3620 (-)	0.0450 (-)	80

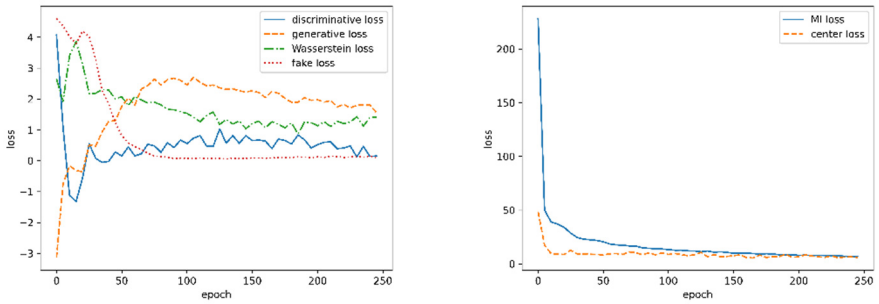
4.3. Results of MVRFFNet

With the same setting as [3], we trained MVRFFNet with four different fusing models. The results are listed in Table 6. The performance of RNN was close to GNN and better than a simple max-pooling or average-pooling layer. The training process is depicted in Figures 14 and 15. As seen classes can provide direct information for classification, the accuracy of seen classes achieves 0.3 in the first 10 epochs. The seen and unseen classes achieve a balance after 60 epochs and keep stable after 200 epochs. MI loss descends rapidly in the first several epochs and then smoothly till the end of the training process, while center

loss descends rapidly in the first several epochs and keeps stable in the following epochs. This is because center loss constrains the inner-class distance of redundancy-free features on themselves, while MI loss composes an upper bound on the conveyed information between the original and redundancy-free features.

Table 6. Classification accuracy of the MVRFFNet on the fine-grained recognition dataset.

	Accuracy		
	Seen Categories	**Unseen Categories**	**H**
Max Pooling	0.6329	0.3974	0.4883
Average Pooling	0.6405	0.3962	0.4895
RNN	0.6217	0.4348	0.5168
GNN	0.6203	0.4328	0.5099



(**a**) Basic losses (**b**) Redundancy-free losses

Figure 14. Training loss of MVRFFNet on the fine-grained recognition dataset.

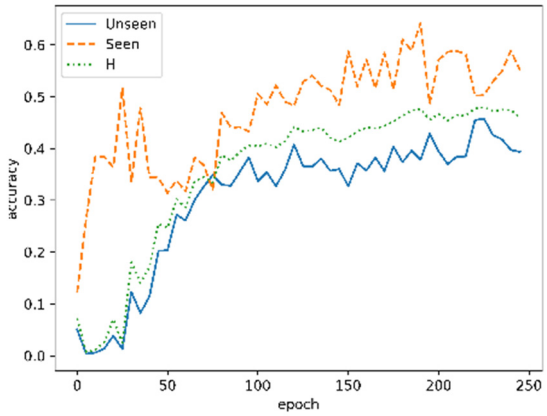


Figure 15. Accuracy of MVRFFNet on the fine-grained recognition dataset.

We also trained a binary matching network to predict whether a registered feature and an on-conveyor feature belong to the same object. The results are listed in Table 7. It can be found that mapping the feature into the redundancy-free space can improve the matching accuracy significantly.

Table 7. Matching accuracy on the test set of the fine-grained recognition dataset.

	Accuracy		
	Seen Categories	**Unseen Categories**	**H**
Original feature	0.5902	0.5781	0.5841
MVRFFNet feature	0.6870	0.6183	0.6508

5. Conclusions

LSMNet and MVRFFNet were proposed in this study for the OCIS and ZSFGR3D subtasks that are involved in the complex LROC task. Experiments were conducted on an OCIS dataset and a fine-grained recognition dataset about objects on a conveyor, respectively. LSMNet could achieve a recalling accuracy close to YOLACT on large and middle objects, while its computing speed on an NVIDIA GTX1660TI GPU was 25 fps, which is much faster than YOLACT's 80 fps. MVRFFNet performed much better than traditional metric learning methods in the retrieval task by mapping the features into a redundancy-free feature space.

Future works will concern three aspects. First, we are going to collect more samples that match the actual situation in Figures 9–11, each sample of which only contains one object that does not cover the case of crowdedness and occlusion. A more comprehensive analysis of the performance of MVRFFNet will be carried out. Second, we will extend the pix2pix architecture to multiple output branches and establish an LSM for each category, LSMNet would be suitable for multiple class instance segmentation tasks. The third is to find contours in a new manner that can transmit the supervised information of detection to improve the performance of LSMNet on small objects. The last task is to combine multiview features and semantic attribution together to improve the performance further. The motivation is that the multiview registered features serve as the attribution of GZSL setting in MVRFFNet, and high-level semantic attribution has not been utilized yet.

Author Contributions: Conceptualization, data curation, K.Z.; methodology, S.D.; project administration, W.L.; software, Z.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by National Natural Science Foundation of China (62002053), the Guangdong Basic and Applied Basic Research Projects (2019A1515111082, 2020A1515110504), Fund for High-Level Talents Afforded by University of Electronic Science and Technology of China, Zhongshan Institute (417YKQ12, 419YKQN15), Social Welfare Major Project of Zhongshan (2019B2010, 2019B2011, 420S36), Achievement Cultivation Project of Zhongshan Industrial Technology Research Institute (419N26), the Science and Technology Foundation of Guangdong Province (2021A0101180005), and Young Innovative Talents Project of Education Department of Guangdong Province (419YIY04).

Data Availability Statement: Not applicable, the study does not report any data.

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

References

1. Dong, S.; Zou, K.; Li, W. Fine-Grained Recognition of 3D Shapes Based on Multi-View Recurrent Neural Network. In Proceedings of the 12th International Conference on Machine Learning and Computing, Shenzhen, China, 15–17 February 2020; pp. 152–156.
2. Su, H.; Maji, S.; Kalogerakis, E.; Learned-Miller, E. Multi-View Convolutional Neural Networks for 3d Shape Recognition. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 945–953.
3. Han, Z.; Fu, Z.; Yang, J. Learning the Redundancy-Free Features for Generalized Zero-Shot Object Recognition. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 12862–12871.
4. Han, Z.; Fu, Z.; Chen, S.; Yang, J. Contrastive Embedding for Generalized Zero-Shot Learning. *arXiv* **2021**, arXiv:2103.16173.
5. Wojke, N.; Bewley, A.; Paulus, D. Simple Online and Realtime Tracking with a Deep Association Metric. In Proceedings of the IEEE International Conference on Image Processing (ICIP), Beijing, China, 17–20 September 2017; pp. 3645–3649.
6. Luo, W.; Xing, J.; Milan, A.; Zhang, X.; Liu, W.; Kim, T.-K. Multiple Object Tracking: A Literature Review. *Artif. Intell.* **2020**, *293*, 103448. [CrossRef]
7. Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement. *arXiv* **2018**, arXiv:1804.02767.
8. Bochkovskiy, A.; Wang, C.-Y.; Liao, H.-Y.M. YOLOv4: Optimal Speed and Accuracy of Object Detection. *arXiv* **2020**, arXiv:2004.10934.
9. Law, H.; Deng, J. CornerNet: Detecting Objects as Paired Keypoints. *arXiv* **2019**, arXiv:1808.01244v2.
10. Duan, K.; Bai, S.; Xie, L.; Qi, H.; Huang, Q.; Tian, Q. CenterNet: Object Detection with Keypoint Triplets. *arXiv* **2019**, arXiv:1904.08189v1.

11. Kim, Y.; Kim, S.; Kim, T.; Kim, C. CNN-Based Semantic Segmentation Using Level Set Loss. In Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV), Waikoloa Village, HI, USA, 7–11 January 2019; pp. 1752–1760.
12. Peng, S.; Jiang, W.; Pi, H.; Li, X.; Bao, H.; Zhou, X. Deep Snake for Real-Time Instance Segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 8533–8542.
13. Microsoft. Microsoft Common Objects in Context 2017. Available online: https://cocodataset.org/#detection-2017 (accessed on 1 September 2017).
14. Bolya, D.; Zhou, C.; Xiao, F.; Lee, Y.J. YOLACT: Real-Time Instance Segmentation. *arXiv* **2019**, arXiv:1912.06218v2.
15. Arjovsky, M.; Chintala, S.; Bottou, L. Wasserstein GAN. *arXiv* **2017**, arXiv:1701.07875v3.
16. Minaee, S.; Boykov, Y.Y.; Porikli, F.; Plaza, A.J.; Kehtarnavaz, N.; Terzopoulos, D. Image Segmentation Using Deep Learning: A Survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, 1–22. [CrossRef] [PubMed]
17. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask R-CNN. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2961–2969.
18. Ren, S.; He, K.; Girshick, R.; Jian, S. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *39*, 1137–1149. [CrossRef] [PubMed]
19. Dai, J.; He, K.; Li, Y.; Ren, S.; Sun, J. Instance-Sensitive Fully Convolutional Networks. In *Computer Vision–ECCV 2016*; Leibe, B., Matas, J., Sebe, N., Welling, M., Eds.; Springer: Cham, Switzerland, 2016.
20. Li, Y.; Qi, H.; Dai, J.; Ji, X.; Wei, Y. Fully Convolutional Instance-Aware Semantic Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 2359–2367.
21. Bolya, D.; Zhou, C.; Xiao, F.; Lee, Y.J. YOLACT++: Better Real-Time Instance Segmentation. *arXiv* **2019**, arXiv:1904.02689. [CrossRef] [PubMed]
22. Xie, E.; Sun, P.; Song, X.; Wang, W.; Liu, X.; Liang, D.; Shen, C.; Luo, P. Polarmask: Single Shot Instance Segmentation with Polar Representation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 16–18 June 2020; pp. 12193–12202.
23. Lee, Y.; Park, J. Centermask: Real-Time Anchor-Free Instance Segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 13906–13915.
24. Wang, Y.; Xu, Z.; Shen, H.; Cheng, B.; Yang, L. Centermask: Single Shot Instance Segmentation with Point Representation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 9313–9321.
25. Zhang, R.; Tian, Z.; Shen, C.; You, M.; Yan, Y. Mask Encoding for Single Shot Instance Segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 16–18 June 2020; pp. 10226–10235.
26. Chen, H.; Sun, K.; Tian, Z.; Shen, C.; Huang, Y.; Yan, Y. BlendMask: Top-down Meets Bottom-up for Instance Segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 16–18 June 2020; pp. 8573–8581.
27. Lampert, C.H.; Nickisch, H.; Harmeling, S. Learning to Detect Unseen Object Classes by Between-Class Attribute Transfer. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), Miami, FL, USA, 20–25 June 2009.
28. Xian, Y.; Lampert, C.H.; Bernt, S.; Zeynep, A. Zero-Shot Learning—A Comprehensive Evaluation of the Good, the Bad and the Ugly. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**, *41*, 2251–2265. [CrossRef] [PubMed]
29. Bendale, A.; Boult, T. Towards Open Set Deep Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016.
30. Xian, Y.; Sharma, S.; Schiele, B.; Akata, Z. F-VAEGAN-D2: A Feature Generating Framework for Any-Shot Learning. In Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–20 June 2019.
31. Palatucci, M.; Pomerleau, D.; Hinton, G.E.; Mitchell, T.M. Zero-Shot Learning with Semantic Output Codes. In Proceedings of the Advances in Neural Information Processing Systems 22: 23rd Annual Conference on Neural Information Processing Systems, Vancouver, BC, Canada, 7–10 December 2009.
32. Akata, Z.; Reed, S.; Walter, D.; Lee, H.; Schiele, B. Evaluation of Output Embeddings for Fine-Grained Image Classification. In Proceedings of the IEEE Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015.
33. Bucher, M.; Herbin, S.; Jurie, F. Generating Visual Representations for Zero-Shot Classification. In Proceedings of the IEEE International Conference on Computer Vision Workshops, Venice, Italy, 22–29 October 2017; pp. 2666–2673.
34. Xian, Y.; Lorenz, T.; Schiele, B.; Akata, Z. Feature Generating Networks for Zero-Shot Learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018.
35. Felix, R.; Kumar, B.; Reid, I.; Carneiro, G. Multi-Modal Cycle-Consistent Generalized Zero-Shot Learning. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018.
36. Huang, H.; Wang, C.; Yu, P.S.; Wang, C.-D. Generative Dual Adversarial Network for Generalized Zero-Shot Learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–21 June 2019; pp. 801–810.
37. Zhu, J.; Park, T.; Isola, P.; Efros, A.A. Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2242–2251.

38. Verma, V.K.; Arora, G.; Mishra, A.; Rai, P. Generalized Zero-Shot Learning via Synthesized Examples. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018.
39. Chen, L.; Zhang, H.; Xiao, J.; Liu, W.; Chang, S.-F. Zero-Shot Visual Recognition Using Semantics-Preserving Adversarial Embedding Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 1043–1052.
40. Liu, S.; Long, M.; Wang, J.; Jordan, M.I. Generalized Zero-Shot Learning with Deep Calibration Network. In *Advances in Neural Information Processing Systems*; Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., Garnett, R., Eds.; Curran Associates Inc.: New York, NY, USA, 2018; pp. 2005–2015.
41. Vu, T.; Kang, H.; Yoo, C.D. SCNet: Training Inference Sample Consistency for Instance Segmentation. *arXiv* **2021**, arXiv:2012.10150.
42. Chen, K.; Wang, J.; Pang, J.; Cao, Y.; Xiong, Y.; Li, X.; Sun, S.; Feng, W.; Liu, Z.; Xu, J.; et al. Open MMLab Detection Toolbox and Benchmark. *arXiv* **2019**, arXiv:1906.07155.