big data and cognitive computing

MDPI

Article

Deep Learning for Highly Accurate Hand Recognition Based on Yolov7 Model

Christine Dewi [1,2], Abbott Po Shun Chen [3,*] and Henoch Juli Christanto [4,*]

[1] Department of Information Technology, Satya Wacana Christian University, Salatiga 50711, Indonesia
[2] Artificial Intelligent Research Center, Satya Wacana Christian University, Salatiga 50711, Indonesia
[3] Department of Marketing and Logistics Management, Chaoyang University of Technology, Taichung 413310, Taiwan
[4] Department of Information System, Atma Jaya Catholic University of Indonesia, Jakarta 12930, Indonesia
* Correspondence: chprosen@gm.cyut.edu.tw (A.P.S.C.); henoch.christanto@atmajaya.ac.id (H.J.C.)

Abstract: Hand detection is a key step in the pre-processing stage of many computer vision tasks because human hands are involved in the activity. Some examples of such tasks are hand posture estimation, hand gesture recognition, human activity analysis, and other tasks such as these. Human hands have a wide range of motion and change their appearance in a lot of different ways. This makes it hard to identify some hands in a crowded place, and some hands can move in a lot of different ways. In this investigation, we provide a concise analysis of CNN-based object recognition algorithms, more specifically, the Yolov7 and Yolov7x models with 100 and 200 epochs. This study explores a vast array of object detectors, some of which are used to locate hand recognition applications. Further, we train and test our proposed method on the Oxford Hand Dataset with the Yolov7 and Yolov7x models. Important statistics, such as the quantity of GFLOPS, the mean average precision (mAP), and the detection time, are tracked and monitored via performance metrics. The results of our research indicate that Yolov7x with 200 epochs during the training stage is the most stable approach when compared to other methods. It achieved 84.7% precision, 79.9% recall, and 86.1% mAP when it was being trained. In addition, Yolov7x accomplished the highest possible average mAP score, which was 86.3%, during the testing stage.

Keywords: hand detection; Yolov7; Yolov7x; convolutional neural network; object detection

Citation: Dewi, C.; Chen, A.P.S.; Christanto, H.J. Deep Learning for Highly Accurate Hand Recognition Based on Yolov7 Model. *Big Data Cogn. Comput.* **2023**, *7*, 53. https://doi.org/10.3390/bdcc7010053

Academic Editor: Moulay A. Akhloufi

Received: 19 February 2023
Revised: 16 March 2023
Accepted: 21 March 2023
Published: 22 March 2023

1. Introduction

In everyday life, the hand is a very important part of how people talk to each other and interact with their surroundings. In order to identify hand movements and other human actions, the position and movement of a person's hands need to be meticulously tracked as they are being written down [1]. Being able to accurately recognize hands in pictures and videos will help with a wide range of visual processing tasks, such as understanding gestures and scenes. Because there are so many kinds of hands in pictures, it is harder to find the hand in uncontrolled situations [2,3]. Hands can have many different orientations, shapes, and sizes. Occlusion and motion blur make the different looks of hands even more noticeable [4,5]. Cluttered environments present a significant challenge for several applications of computer vision, including human–computer interaction [6,7], sign language recognition [8], hand action analysis [9,10], and complete hand gesture recognition systems [11,12].

In recent years, hand position estimation and gesture recognition in restricted situations have attained a level of maturity. However, hand-related applications in unconstrained environments will be an important trend in the next future. Under these circumstances, the identification of hands in an unrestricted environment constitutes a new bottleneck in the work that is related to hands. Therefore, the high-precision hand

recognition approach will be an essential stage in the process for hand-related applications that operate in an environment with few restrictions.

The complexity of the hand detection task is directly related to the variety of hand appearances, which can vary in terms of hand shape, skin color, orientation, scale, and partial occlusion, among several other features. This can make it challenging for the task to be performed [13,14]. Therefore, the shared information that was presented in the training signal for the hand appearance re-construction task can be utilized as an inductive bias in order to increase the performance of the hand detection job [15,16].

In July of 2022, the Yolov7 model was made available to the public [17]. Overall, Yolov7 provides a quicker and more resilient network architecture, as well as an improved method for feature integration, improved object recognition performance, a more robust loss function, and a higher label assignment and model training efficiency. In addition, Yolov7 offers an improved method for feature integration. Because of this, the computational hardware that Yolov7 needs to run is significantly less expensive than what is required by other deep learning models. It is possible to train it far more quickly on smaller datasets without using any pre-trained weights. The Yolov7 model pre-processing approach is combined with the Yolov5 model pre-processing method, and the usage of Mosaic data enhancement is appropriate for small object recognition [18,19]. In terms of its design, the proposal calls for an extended ELAN that is based on the original ELAN. To overcome the problem of automatic hand recognition, we incorporated Yolov7 in our experiment.

The following is the most important contribution that this research provides: (1) A brief description of the Yolov7 family of object identification algorithms, including Yolov7 and Yolov7x with 100 and 200 epochs, may be found in this research. (2) This study explores a wide range of object detectors. Performance metrics monitor critical data such as the average mean accuracy (mAP), Intersection over Union (IoU), and the quantity of GFLOPS.

This paper will continue with the following sections: Section 2 provides a summary of current research papers and an explanation of our methodology. The outcomes of the experiments are presented in Section 3. Section 4 discusses our findings, while Section 5 outlines our conclusions and directions for future study.

2. Materials and Methods

2.1. Hand Recognitions with Convolutional Neural Network (CNN)

The segmentation of skin tone was used by many older hand recognition algorithms to isolate hands from their backgrounds, which was a time-consuming and ineffective process, after removing other skin regions from the original image, such as the face. In the hue, saturation, and value (HSV) color space, Dardas et al. [20] suggested a thresholding method for fragmenting hands into individual colors. This was done after first removing other skin regions, such as the face. After trying out a number of different color spaces, Girondel and colleagues [21] found that the Cb and Cr channels in the YCbCr color space were particularly effective for the skin recognition job. Sigal et al. [22] proposed the Gaussian mixture model, which performed admirably under a variety of lighting condition.

Accurate hand detection is essential for many applications. Mittal et al. [23] came up with a technique that utilizes a collection of movable pieces. Karlinsky et al. [24] suggested a method for hand detection that makes use of sensing the hand's relative position in relation to other human body components [25]. Recognizing hand gestures and detecting fingertip locations can be broken down into three classes based on [26]. The first set of papers addresses the issue of gesture recognition. The second set of works is devoted to fingertip detection, while the third set tackles both gesture recognition and fingertip detection head-on. Moreover, Nunez et al. [27] combined a neural network with a long short-term memory (LSTM) network to recognize 3D hand gestures based on a skeleton's temporal properties [28].

Recently, there has been a surge in interest in CNN-based detection approaches as a research issue in the computer vision field. This is due to the fact that deeper and higher-level features can be learned from networked systems. By utilizing CNN, one is

able to effectively address both the multi-scale and varied rotation difficulties that were previously described. Recent research has concentrated on three primary avenues with the goal of producing improved object detection systems; these principals are also suited for CNN-based hand detection. The following is an explanation of each of the three primary directions: (1) Changing the fundamental architecture of these networks should be the first primary step in this approach. (2) The second primary aim is to exploit the data themselves by increasing the variety qualities of the training data. This is the second principal direction. (3) Thirdly, using proxy tasks for reasoning and other top-down processes to improve object detection representations is a promising avenue of research [29]. This third main direction guides our efforts. Through the use of hand appearance reconstruction, we are able to include universally available data into our detection system. Reconstruction can deal with significantly more complex information of the hand than was ever presented in earlier hand identification challenges, such as scales, contours, skin colors, and even partial occlusions of the hand [30,31].

2.2. Yolov7 Architecture

You Only Look Once version 7 (Yolov7) is a real-time object detector that only uses a single stage. In July 2022, it was presented to the Yolo family for the first time. The Yolov7 paper claims that it is the quickest and most accurate real-time object detector that has been developed to this day [17]. Through major improvements to its overall performance, Yolov7 has created a significant new benchmark.

Image frames are characterized by a backbone in a model known as Yolo [32]. Yolo predicts the positions and classes of objects around which bounding boxes should be created. These features are integrated and mixed in the neck, and then they are passed along to the head of the network. To arrive at its ultimate forecast, Yolo engages in a post-processing procedure known as non-maximum suppression (NMS) [33].

The authors of Yolov7 improve on previous research that has been conducted on this subject. They do so while keeping in mind the amount of memory that is required to keep layers stored in memory as well as the distance that a gradient must travel before it can back-propagate through the layers. If they make the gradient shorter, their network will be able to learn more effectively. The E-ELAN layer aggregation, which is an extended version of the efficient layer aggregation network (ELAN) computational block, is the one that they decide to go with as the final layer aggregation. Object detection models will typically consider the resolution that the network is trained on, as well as the depth and width of the network. The authors of Yolov7 scale the network depth and width in conjunction with one another while simultaneously concatenating layers. Studies on ablation demonstrate that this technique maintains an optimal model design even when scaled to different scales [19].

Yolov7 suggested a re-parameterized convolution that was intended. The creators of this model observed that there was a layer in this proposed planned re-parameterized model that included residual or concatenation connections; its RepConv should not have an identity connection. This was one of the findings of the model. RepConvN, which does not contain any identity links, can serve as a suitable replacement for it under these conditions. Within a single convolutional layer, RepConv utilizes a combination of 3×3 convolutions, 1×1 convolutions, and identity connections. To develop the architecture of the intended re-parameterized convolution, the authors used RepConv without identity connection (also known as RepConvN) after performing research on the combination of RepConv with various architectures and the resulting performance of those combinations. According to the findings of the research article, there should not be any identity connections when a convolutional layer that included residual or concatenation was replaced by a re-parameterized convolution.

In accordance with the structural diagram, the Yolov7 network may be broken down into three distinct components: the input network, the backbone network, and the head network [34]. The Yolov7 network, firstly, pre-processed the image, resized it to $640 \times 640 \times 3$, and input it into the backbone network. The length and width of the feature map were successively cut

in half by the CBS composite module, the ELAN module, and the MP module. At the same time, the number of output channels was raised to be equal to twice the number of input channels. As shown in Figure 1, the CBS composite module performed the *convolution + BN + activation function* on the input feature map. In Yolov7, the same as Yolov5, *Silu* was used as the activation function. It was suggested that we use the ELAN module. To continuously enhance the network's learning capabilities without ruining the initial gradient path, cardinality was expanded, shuffled, and merged. With the help of group convolution, we were able to increase the channel count and cardinality of the computational blocks while maintaining the same channel count in our feature map ensembles as we did in our initial design. Finally, the output from the ELAN module has twice as many channels as the input. Both the feature map's dimensions and the number of channels were cut in half by the max-pooling operation performed by the MP module's top branch. After the initial convolution, the length and breadth of the feature map were cut in half by the lower branch, while the kernel size and stride were increased by one and two, respectively. The two levels of the tree were joined into one. After all that work, we had a feature map with input and output channels that were the same size [35].

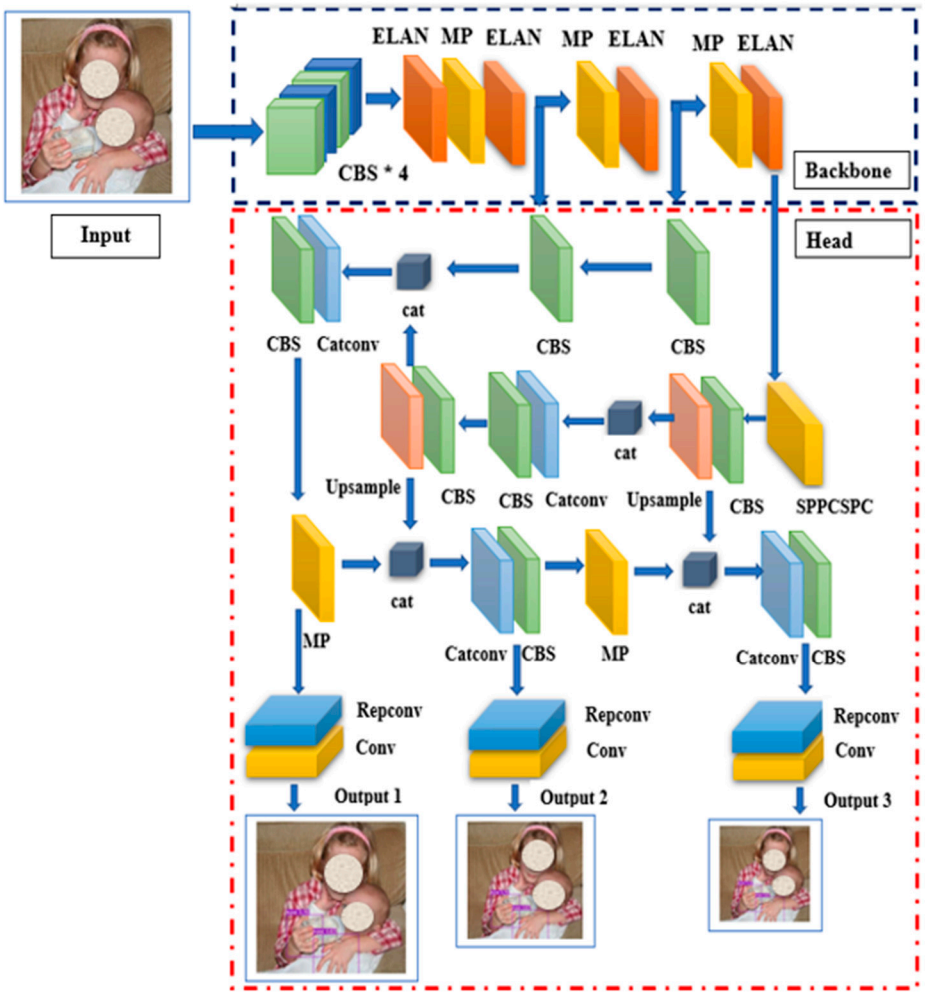


Figure 1. Yolov7 architecture. Source: The human faces in the figures are all from public datasets (Oxford Hand Dataset).

Figure 2 describes our research workflow. In our experiment, we use the hand detection process using images from the Oxford Hand Dataset [23] as input data. Next, we train our dataset with Yolov7 and Yolov7x, with 100 and 200 epochs for each model. We will then study and discuss the results of the training and testing phases of Yolov7, which involves calculating the bounding box with NMS.

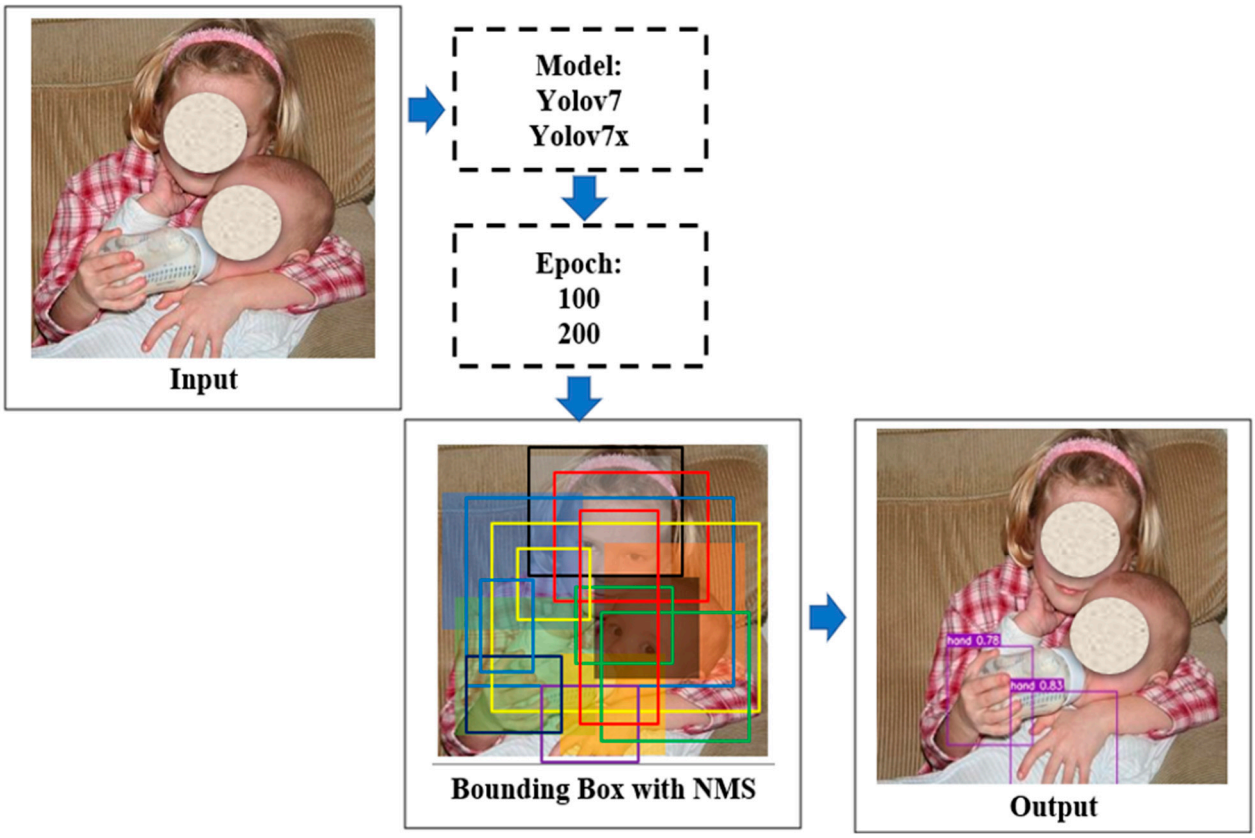


Figure 2. Research workflow. Source: The human faces in the figures are all from public datasets (Oxford Hand Dataset).

Most annotation programs output their findings in the Yolo labeling format, which creates a single text file with annotations for each image. Each text file has an annotation consisting of a bounding box, sometimes known as the abbreviation "BBox", for each of the graphical elements that are displayed in the image. The scale of the annotations has been adjusted so that it is proportional to the image, and their values vary from 0 all the way up to 1 [36]. The Equations (1)–(6) will serve as the foundation for the adjustment technique used in the calculation using the Yolo format.

$$dw = 1/W \tag{1}$$

$$x = \frac{(x_1 + x_2)}{2} \times dw \tag{2}$$

$$dh = 1/H \tag{3}$$

$$y = \frac{(y_1 + y_2)}{2} \times dh \tag{4}$$

$$w = (x_2 - x_1) \times dw \tag{5}$$

$$h = (y_2 - y_1) \times dh \tag{6}$$

H is used to denote the height of the image, dh is used to refer to the absolute height of the image, W is used to denote the width of the image, and dw is used to represent the absolute width of the picture.

3. Results

3.1. Oxford Hand Dataset

The Oxford Hand Dataset [23] is a free, extensive, and publicly available image dataset of hands that has been gathered from a wide range of public image dataset sources.

Annotations are included in each picture for all the many examples of hands that can be seen clearly by humans in that picture. Over the course of the entire dataset, there are a total of 13,050 hand instances. While there are 11,019 data points assigned to each hand instance in the training set, there are only 2031 data points assigned to each hand instance in the testing set. During the process of collecting data, there were no limitations imposed on the subject's attitude or visibility, and there were also no limitations imposed on the environment that was immediately surrounding the subject. Annotations are included in each picture for all the hands that can be readily discerned by humans in the picture. The annotations need to be aligned about the wrist, but the bounding rectangles do not have to be aligned along any axis. The files in the 'annotations' folder store the annotations for the four corners of the hand-bounding box in the normal MATLAB *".mat"* format.

The structure is composed of boxes, with hand-boxes standing in for the various indices that are associated with the cell array. On this dataset, we perform the data pre-processing and then convert the data into the Yolo format. The dataset is broken up into two sections: seventy percent is used for training, and thirty percent is used for testing. Both sections include pictures of a variety of things that can be held in the hand. The illustration of a representative image from the Oxford Hand Dataset collection may be found in Figure 3.

Figure 3. Oxford Hand Dataset sample images. Source: The human faces in the figures are all from public datasets (Oxford Hand Dataset).

3.2. Training Result

The training procedure and its outcome will be detailed at this point. The training for test batch 0 labels and test batch 0 predictions is depicted in Figure 4. Yolov7 uses a genetic

algorithm to generate the anchor boxes on its own. They call this procedure "auto anchor", since it automatically recalculates the anchor boxes to make them a better fit for the data if the default ones are inadequate. This information is then integrated with the k-means method to produce k-means evolved anchor boxes. An additional auxiliary head can be placed at any point in the network's middle layers to provide deep supervision, with the shallow network weights and the assistant loss serving as the guiding parameters. Even in circumstances in which the model weights would normally converge, this method can be valuable for making changes to the model. In Yolov7's design, the training head is called an auxiliary head, while the lead head is responsible for overseeing the production of the final output. The lead head prediction is used as guidance by Yolov7 to generate coarse-to-fine hierarchical labels, which are then used for auxiliary head learning and lead head learning, respectively.

Further, Yolov7's training phase involves the splicing together of four separate photos. After being subjected to a random processing step during the splicing phase, each of the four distinct images has dimensions and configurations that are different from the others. We will utilize the validation script to examine our model. The 'task' setting allows users to customize whether their model's performance is measured on the full training set, the validated test set, or the test set alone. The default location for results is the runs/train directory; for future training sessions, a new ex-experiment directory is created and given a unique name, such as *runs/train/exp1, runs/train/exp1,* etc. We look at the train and *val.jpg* to see the mosaics, labels, forecasts, and augmentation effects. It is worth noting that training requires an *Ultralytics* Mosaic Data loader, a device that combines four images into a single mosaic. After training our model for 100 and 200 epochs, we save our weights.

Fine-tuning, the final phase of training, is discretionary. In this stage, we will unfreeze the whole model we just built and retrain it at a much slower learning rate on our data. By gradually changing the previously trained features to accommodate the new data, significant gains may be possible. We can adjust the learning rate in the hyperparameters-configurations file. The learning rate with these hyperparameters is drastically reduced compared to the standard settings. The weights will initially be set to the last saved values from the previous step. As per the established practice in *PyTorch*, we have saved our trained model with the *.pt* file extension.

The mAP@0.5 will be monitored during the training phase to determine how well our detector is learning to detect on the validation set; a higher number indicates improved performance. One of the most crucial parts of the Yolov7 training is the dataset written in Yet Another Markup Language (YAML). Class names and the location of the data used for training and checking are listed in this file. For the training script to correctly identify the locations of the images, labels, and classes, this file path must be passed along as an argument. The dataset already includes these data. Table 1 describes the training process of the Yolov7 and Yolov7x models with 100 and 200 epochs. Yolov7x achieves the highest precision of 84.7%, recall of 79.9%, mAP of 86.1%, the training time needed of 8.616 h, and size of 142.1 MB when training with 200 epochs.

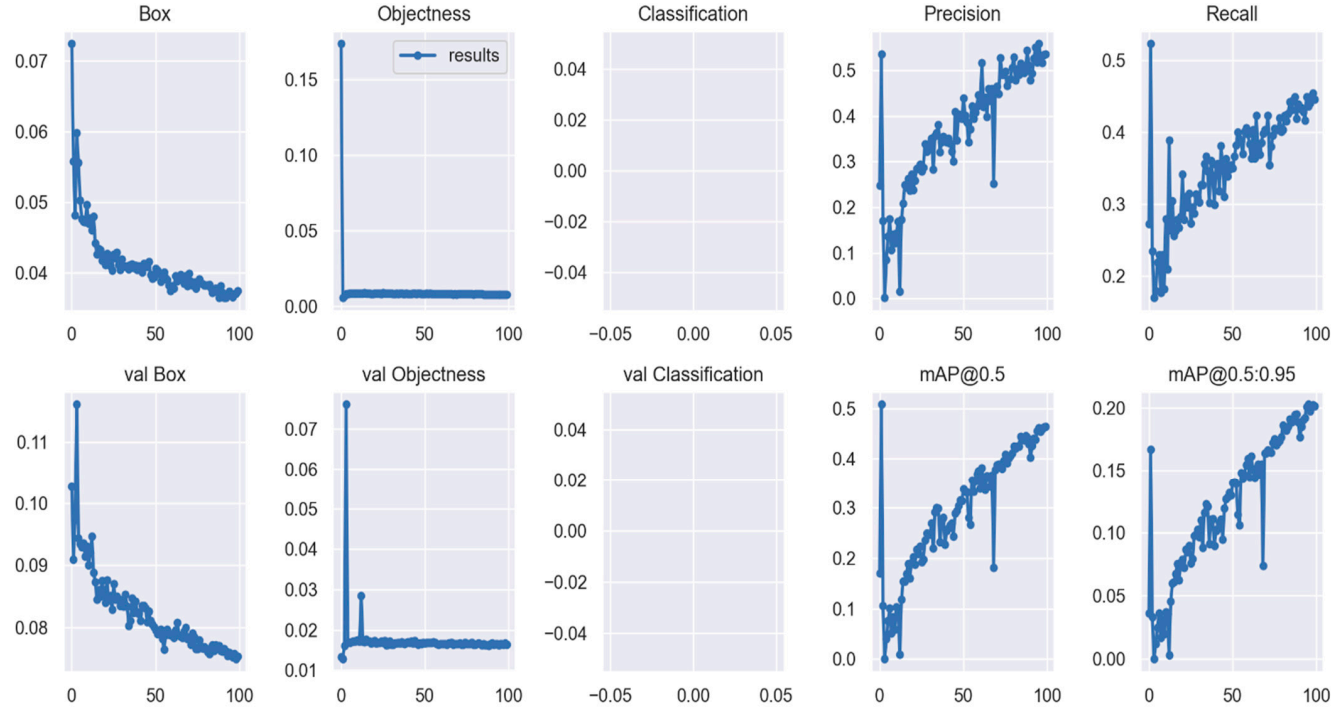
Table 1. Training performance of Yolov7 and Yolov7x.

Model	Epoch	Class	Images	Labels	P	R	mAP@0.5	Training Time (hours)	Size (MB)
Yolov7x	100	All	1205	2487	0.536	0.446	0.465	4.522	142.1
Yolov7x	200	All	1205	2487	0.847	0.799	0.861	8.616	142.1
Yolov7	100	All	1205	2487	0.591	0.509	0.539	2.599	74.8
Yolov7	200	All	1205	2487	0.774	0.663	0.742	5.313	74.8

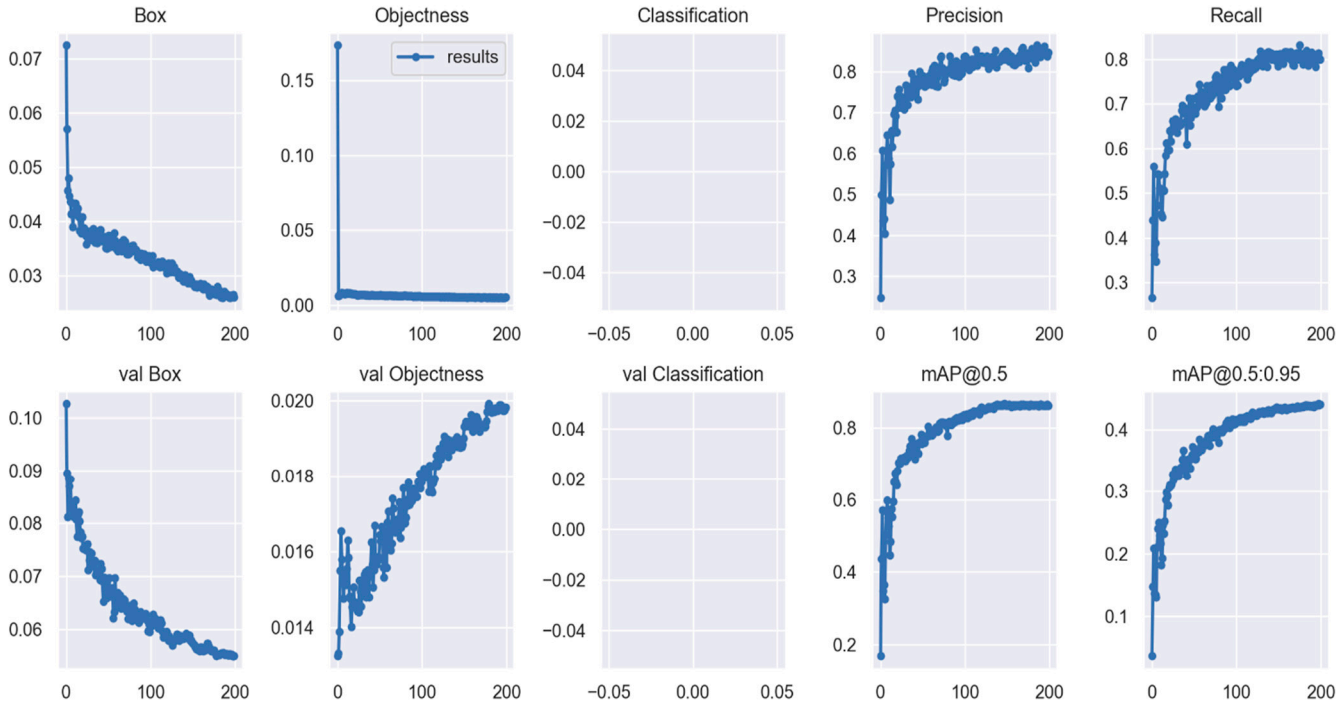
Figure 4. Test batch 0 labels and test batch 0 prediction. (**a**) Test batch 0 labels and (**b**) test batch 0 prediction. Source: The human faces in the figures are all from public datasets (Oxford Hand Dataset).

Further, Yolov7 exhibits a precision value of 77.4%, recall of 66.3%, mAP of 74.2%, the training time of 5.313 h, and size of 74.8 MB while training with 200 epochs. Based on the experiment result, the 200-epoch model achieves the highest performance of all models in

the experiment and the epochs affected the training result. The bigger the epoch, the better the performance, but the longer the processing time. Figure 5 depicts the Yolov7x training graph with (a) 100 epochs and (b) 200 epochs.



(**a**) 100 epochs

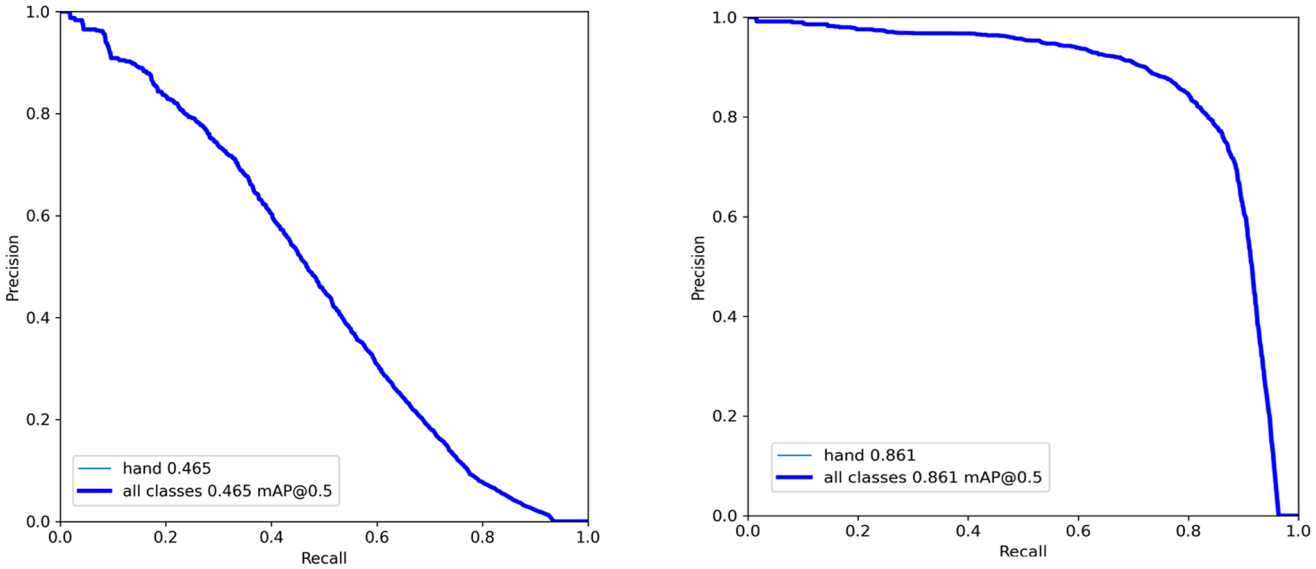


(**b**) 200 epochs

Figure 5. Yolov7x training graph with (**a**) 100 epochs and (**b**) 200 epochs.

In addition, we have the option of acquiring the precision–recall curve, which is stored in a persistent manner following each validation. Figure 6 shows Yolov7x's accuracy and

recall for both 100 and 200 epochs. These are the measurements we use to measure how well our Oxford Hand Dataset works in simulations using the Yolov7 and Yolov7x models. The measures include the F1 score, precision, recall, and accuracy. Among them, precision and recall are defined in Equations (7) and (8) [37], and then accuracy and F1 are defined in Equations (9) and (10) [38].



(**a**) 100 epochs (**b**) 200 epochs

Figure 6. Precision and recall of Yolov7x with (**a**) 100 and (**b**) 200 epochs.

Ultralytics has used the binary cross-entropy with the logits loss function that is offered in *PyTorch* for the purpose of calculating the amount of loss that has occurred in terms of both the class probability and the object score [39]. The true positive (TP) is the number of "yes"s in the real situation where the model evaluation is also a "yes", and the true negative (TN) is the number of "no"s in the real situation where the model evaluation is also a "no". The terms are abbreviated as "TP" and "TN", respectively. A false positive (FP) occurs when the observed data do not match the predicted values from the model, while a false negative (FN) occurs when the observed data do match the predicted values from the model [40].

$$Precision\ (P) = \frac{TP}{TP + FP} \tag{7}$$

$$Recall\ (R) = \frac{TP}{TP + FN} \tag{8}$$

$$Accuracy\ (Acc) = \frac{TP + TN}{TP + FN + FP + FN} \tag{9}$$

$$F1 = (2\ \times\ \text{Precision}\ \times\ \text{Recall})/(\text{Precision} + \text{Recall}) \tag{10}$$

The integral over the precision $p(0)$ is the average mean average precision (mAP) and IoU shown in Equation (11) and Equation (12), respectively:

$$mAP = \int_0^1 p(0)do \tag{11}$$

where $p(0)$ denotes the level of accuracy achieved by the object detection. *IoU* determines the percentage of overlap between the bounding box of the prediction (pred) and the ground-truth value (gt) [41].

$$IoU = \frac{Area_{pred} \cap Area_{gt}}{Area_{pred} \cup Area_{gt}} \tag{12}$$

Furthermore, FLOPS can be recorded in different measures of precision. In our experiment, we implement the GigaFLOPS/GFLOPS: 10^9 FLOPS, and this could be seen in Equation (13).

$$FLOPS = cores \times \frac{cycles}{second} \times \frac{FLOPS}{cycle} \tag{13}$$

Moreover, Equation (14) [42] shows the calculation of the Yolo loss functions.

$$\begin{aligned} Yolo\ Loss\ Function\ =\ & \lambda_{coord} \sum_{i=0}^{s^2} \sum_{j=0}^{B} \mathbb{I}_{ij}^{obj} \left[(x_i - \hat{x}_i)^2 + (\mathrm{y} - \hat{y}_i)^2 \right] \\ & + \lambda_{coord} \sum_{i=0}^{s^2} \sum_{j=0}^{B} \mathbb{I}_{ij}^{obj} \left[\left(\sqrt{w_i} - \sqrt{\hat{w}_i} \right)^2 + \left(\sqrt{h_i} - \sqrt{\hat{h}_i} \right)^2 \right] \\ & + \sum_{i=0}^{s^2} \sum_{j=0}^{B} \mathbb{I}_{ij}^{obj} \left(C_i - \hat{C}_i \right)^2 + \lambda_{noobj} \sum_{i=0}^{s^2} \sum_{j=0}^{B} \mathbb{I}_{ij}^{noobj} \left(C_i - \hat{C}_i \right)^2 \\ & + \sum_{i=0}^{s^2} \mathbb{I}_{i}^{obj} \sum_{c \in classes} (p_i(\mathrm{c}) - \hat{p}_i(c))^2 \end{aligned} \tag{14}$$

where S is the total number of grid cells in the image, B is the number of bounding boxes that are projected to exist within each grid cell, and c is the class that is predicted to exist between each grid cell. Moreover, $p_i(\mathrm{c})$ denotes the confidence probability score. For any box j of cell i, x_{ij} and y_{ij} represent the co-ordinates of the center of the anchor box, h_{ij} denotes the height, w_{ij} represents the width of the box, and C_{ij} denotes the confidence score. Finally, $\lambda coord$ and $\lambda noobj$ are the weights to decide the significance of localization.

4. Discussion

As shown in Table 2, we tested Yolov7 and Yolov7x with 100 and 200 epochs and found that both performed equally well. Our model is ready to move on to the inference phase after producing good results during training. The final forecast is an ensemble of all these enriched versions of the images. Test-time augmentations can be applied to the predictions after inference to further increase their accuracy (TTA). If we want to keep our frames-per-second (FPS) rate high, we will have to ditch the TTA because it produces an inference that is two to three times longer.

Table 2. Testing performance of Yolov7 and Yolov7x with Oxford Hand Dataset.

Model	Epoch	Class	Images	Labels	P	R	mAP@0.5
Yolov7x	100	All	1205	2487	0.532	0.46	0.459
Yolov7x	200	All	1205	2487	0.844	0.8	0.863
Yolov7	100	All	1205	2487	0.597	0.521	0.55
Yolov7	200	All	1205	2487	0.732	0.672	0.736

We used sample sets of images from each category to evaluate Yolov7 and Yolov7x. According to the findings of our experiment, Yolov7 and Yolov7x obtain the best performance when they are trained with a total of 200 epochs. Yolov7x exhibits 84.4% precision, 80% recall, and 86.3% mAP, followed by Yolov7 with the precision value of 73.2%, recall of 67.2%, and mAP of 73.6%. A collection of parameters called hyperparameters are determined before formal training begins in deep learning, which is still the case, even if the validation loss grew proportionally larger as the model's complexity increased, even

though the model's ability to spot outliers improved only somewhat. Two indicators of model complexity are the hefty size of its weight and the quantity of its parameters. These indices rise as the model complexity rises, and as a result, more memory (RAM) is needed by the GPU to hold the model while it is being trained.

Figure 7 shows the recognition result of the Oxford Hand Dataset with Yolov7x. The Yolov7x can detect all hands in Figure 7 very well with various accuracies such as 85%, 77%, 32%, 54%, 91%, 93%, 92%, and 77%.

Figure 7. Recognition result of Oxford Hand Dataset with Yolov7x. Source: The human faces in the figures are all from public datasets (Oxford Hand Dataset).

An overview of the Yolov7 models with the Oxford Hand Dataset is shown in Table 3. Yolov7x contains 14.1 inference, 1.2 NMS, total (inference + NMS) 15.30, 65.359 FPS, 326 layers, 70,782,444 parameters, and 188 GFLOPS while training with 200 epochs. On the other hand, Yolov7 contains of 314 layers, 36,481,772 parameters, 6,194,944 gradients, and 103.2 GFLOPS. Yolov7 is the foundational model, and it is designed to be as efficient as possible for general GPU processing.

Table 3. An overview of Yolov7 models with Oxford Hand Dataset.

Model	Epoch	Inference	NMS	Total	FPS	Layers	Parameters	Gradient	GFLOPS
Yolov7x	100	8.8	1.3	10.10	99.010	362	70,782,444	0	188
Yolov7x	200	14.1	1.2	15.30	65.359	362	70,782,444	0	188
Yolov7	100	8.9	1.2	10.10	99.010	314	36,481,772	6,194,944	103.2
Yolov7	200	8.8	1.3	10.10	99.010	314	36,481,772	6,194,944	103.2

The advantages of Yolov7 are numerous, and some of them are listed below: First, the improved network architecture in Yolov7 allows for a more efficient label assignment and model training, as well as an improved accuracy in object identification and a more robust loss function. Second, Yolov7 is faster than other object detectors that are state-of-the-art, and it is almost 120 times faster than Yolov5. Next, it demonstrates superior average precision on the COCO dataset compared to that of other object detectors. The design and loss function have been optimized. Instance segmentation, categorization, object identification, and posture estimation are all supported by the Yolov7 repository. Finally, we offer several distinct variations of the Yolov7 model to cater to customers with varying needs in terms of speed and accuracy. Table 4 provides an explanation of the comparison of the earlier study.

Table 4. Previous research comparison on the Oxford Hand Dataset.

Author	mAP (%)	Method
Mittal et al. [23]	48.2	Two-stage hypothesize-and-classify framework
Deng et al. [43]	57.7	Joint model
Le et al. [44]	75.1	Multiple-scale region-based fully convolutional networks (MS RFCN)
Li Yang et al. [45]	83.2	CNN, MobileNet
Our method	86.3	Yolov7x

The Yolov7x technique that we have proposed performs better than previous models in terms of mAP, with an accuracy of 86.3% when using the Oxford Hand Dataset. In a recent research study on hand detections, we were able to improve upon the study's overall performance. Le et al. [44] proposed the multiple-scale region-based fully convolutional networks (MS RFCN) which exhibit only 75.1% mAP. Another researcher [45] implements a CNN and MobileNet and achieves 83.2% mAP.

5. Conclusions

The goal of this research manuscript is to provide a thorough overview of CNN-based object identification algorithms. More specifically, the Yolov7 and Yolov7x algorithm with 100 and 200 epochs will serve as the key foci of examination in this study. Throughout our exploratory studies, we test and study a wide range of modern object detectors. Among the detectors we investigate are, for example, those that are designed to identify the Oxford Hand Dataset.

After putting all of the information from our investigation together, we have come to the following summary conclusion: First, while training with 200 epochs, Yolov7x has 14.1 inference, 1.2 NMS, a total of 15.30 (inference plus NMS), 65.359 FPS, 326 layers, 70,782,444 parameters, and 188 GFLOPS. On the other hand, Yolov7 has a total of 103.2 GFLOPS, 314 layers, 36,481,772 parameters, and 6,194,944 gradients. Next, according to the findings of the experiment, the model with 200 iterations had the best performance of all the models tested, and the number of iterations influenced the training result. The larger the epoch, the greater the performance; nevertheless, this will increase the amount of time it takes to process. We plan to combine hand detection with federated learning in our future research. Federated learning is just a type of machine learning that is not centralized.

Author Contributions: Conceptualization, C.D. and A.P.S.C.; data curation, H.J.C.; formal analysis, C.D. and A.P.S.C.; investigation, C.D. and H.J.C.; methodology, C.D.; project administration, C.D.; resources, H.J.C.; software, C.D. and H.J.C.; supervision, A.P.S.C.; validation, C.D., A.P.S.C. and H.J.C.; visualization, H.J.C.; writing—original draft, C.D. and A.P.S.C.; writing—review & editing, C.D. and A.P.S.C. All authors have read and agreed to the published version of the manuscript.

Funding: This paper is supported by the National Science and Technology Council, Taiwan (Grant number: MOST-111-2637-H-324-001-).

Institutional Review Board Statement: Ethical review and approval were waived for this study, due to the reason that we use the public and free Oxford Hand Dataset. The human faces in the figures are all from the public datasets.

Informed Consent Statement: Written informed consent was waived for this study due to the reason that we use the public and free Oxford Hand Dataset. The human faces in the figures are all from the public datasets.

Data Availability Statement: Oxford Hand Dataset (https://www.robots.ox.ac.uk/~vgg/data/hands/ (accessed on 10 January 2023) and https://drive.google.com/drive/folders/11zS5IYJAdyKrYD127-RPLNaeJgK1eRaP?usp=sharing (accessed on 13 January 2023).

Acknowledgments: The author would like to thank all colleagues from Chaoyang Technology University, Atma Jaya Catholic University of Indonesia, and Satya Wacana Christian University, Indonesia, and all involved in this research.

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

References

1. Xu, C.; Cai, W.; Li, Y.; Zhou, J.; Wei, L. Accurate Hand Detection from Single-Color Images by Reconstructing Hand Appearances. *Sensors* **2020**, *20*, 192. [CrossRef] [PubMed]
2. Narasimhaswamy, S.; Wei, Z.; Wang, Y.; Zhang, J.; Nguyen, M.H. Contextual Attention for Hand Detection in the Wild. In Proceedings of the IEEE International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019.
3. Dewi, C.; Chen, R.-C. Decision Making Based on IoT Data Collection for Precision Agriculture. In *Intelligent Information and Database Systems: Recent Developments*; Huk, M., Maleszka, M., Szczerbicki, E., Eds.; ACIIDS 2019. Studies in Computational Intelligence; Springer: Cham, Switzerland, 2020; Volume 830, pp. 31–42.
4. Dewi, C.; Christanto, J. Henoch Combination of Deep Cross-Stage Partial Network and Spatial Pyramid Pooling for Automatic Hand Detection. *Big Data Cogn. Comput.* **2022**, *6*, 85. [CrossRef]
5. Mohammed, A.A.Q.; Lv, J.; Islam, M.D.S. A Deep Learning-Based End-to-End Composite System for Hand Detection and Gesture Recognition. *Sensors* **2019**, *19*, 5282. [CrossRef]
6. Rapp, A.; Curti, L.; Boldi, A. The Human Side of Human-Chatbot Interaction: A Systematic Literature Review of Ten Years of Research on Text-Based Chatbots. *Int. J. Hum. Comput. Stud.* **2021**, *151*, 102630. [CrossRef]
7. Ashiquzzaman, A.; Lee, H.; Kim, K.; Kim, H.Y.; Park, J.; Kim, J. Compact Spatial Pyramid Pooling Deep Convolutional Neural Network Based Hand Gestures Decoder. *Appl. Sci.* **2020**, *10*, 7898. [CrossRef]
8. Shin, J.; Matsuoka, A.; Hasan, M.A.M.; Srizon, A.Y. American Sign Language Alphabet Recognition by Extracting Feature from Hand Pose Estimation. *Sensors* **2021**, *21*, 5856. [CrossRef] [PubMed]
9. Knights, E.; Mansfield, C.; Tonin, D.; Saada, J.; Smith, F.W.; Rossit, S. Hand-Selective Visual Regions Represent How to Grasp 3D Tools: Brain Decoding during Real Actions. *J. Neurosci.* **2021**, *41*, 5263–5273. [CrossRef]
10. Kang, P.; Li, J.; Fan, B.; Jiang, S.; Shull, P.B. Wrist-Worn Hand Gesture Recognition While Walking via Transfer Learning. *IEEE J. Biomed. Health Inform.* **2022**, *26*, 952–961. [CrossRef]
11. Qiang, B.; Zhai, Y.; Zhou, M.; Yang, X.; Peng, B.; Wang, Y.; Pang, Y. SqueezeNet and Fusion Network-Based Accurate Fast Fully Convolutional Network for Hand Detection and Gesture Recognition. *IEEE Access* **2021**, *9*, 77661–77674. [CrossRef]
12. Aamir, M.; Rahman, Z.; Ahmed Abro, W.; Tahir, M.; Mustajar Ahmed, S. An Optimized Architecture of Image Classification Using Convolutional Neural Network. *Int. J. Image Graph. Signal Process.* **2019**, *11*, 30–39. [CrossRef]
13. Ur Rehman, M.; Ahmed, F.; Khan, M.A.; Tariq, U.; Alfouzan, F.A.; Alzahrani, N.M.; Ahmad, J. Dynamic Hand Gesture Recognition Using 3D-CNN and LSTM Networks. *Comput. Mater. Contin.* **2022**, *70*, 4675–4690. [CrossRef]
14. Guan, Y.; Aamir, M.; Hu, Z.; Abro, W.A.; Rahman, Z.; Dayo, Z.A.; Akram, S. A Region-Based Efficient Network for Accurate Object Detection. *Trait. Signal* **2021**, *38*, 481–494. [CrossRef]
15. Chang, C.W.; Santra, S.; Hsieh, J.W.; Hendri, P.; Lin, C.F. Multi-Fusion Feature Pyramid for Real-Time Hand Detection. *Multimed. Tools Appl.* **2022**, *81*, 11917–11929. [CrossRef]
16. Alam, M.M.; Islam, M.T.; Rahman, S.M.M. Unified Learning Approach for Egocentric Hand Gesture Recognition and Fingertip Detection. *Pattern Recognit.* **2022**, *121*, 108200. [CrossRef]

17. Wang, C.Y.; Bochkovskiy, A.; Liao, H.Y.M. YOLOv7: Trainable Bag-of-Freebies Sets New State-of-the-Art for Real-Time Object Detectors. *arXiv* **2022**, arXiv:2207.02696.
18. Jiang, K.; Xie, T.; Yan, R.; Wen, X.; Li, D.; Jiang, H.; Jiang, N.; Feng, L.; Duan, X.; Wang, J. An Attention Mechanism-Improved YOLOv7 Object Detection Algorithm for Hemp Duck Count Estimation. *Agriculture* **2022**, *12*, 1659. [CrossRef]
19. Chen, J.; Liu, H.; Zhang, Y.; Zhang, D.; Ouyang, H.; Chen, X. A Multiscale Lightweight and Efficient Model Based on YOLOv7: Applied to Citrus Orchard. *Plants* **2022**, *11*, 3260. [CrossRef]
20. Dardas, N.H.; Georganas, N.D. Real-Time Hand Gesture Detection and Recognition Using Bag-of-Features and Support Vector Machine Techniques. *IEEE Trans. Instrum. Meas.* **2011**, *60*, 3592–3607. [CrossRef]
21. Girondel, V.; Bonnaud, L.; Caplier, A. A Human Body Analysis System. *EURASIP J. Adv. Signal Process.* **2006**, *2006*, 61927. [CrossRef]
22. Sigal, L.; Sclaroff, S.; Athitsos, V. Skin Color-Based Video Segmentation under Time-Varying Illumination. *IEEE Trans. Pattern Anal. Mach. Intell.* **2004**, *26*, 862–877. [CrossRef]
23. Mittal, A.; Zisserman, A.; Torr, P. Hand Detection Using Multiple Proposals. *Bmvc* **2011**, *2*, 5.
24. Karlinsky, L.; Dinerstein, M.; Harari, D.; Ullman, S. The Chains Model for Detecting Parts by Their Context. In Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Francisco, CA, USA, 13–18 June 2010.
25. Rastgoo, R.; Kiani, K.; Escalera, S. Real-Time Isolated Hand Sign Language Recognition Using Deep Networks and SVD. *J. Ambient Intell. Humaniz. Comput.* **2022**, *13*, 591–611. [CrossRef]
26. Bandini, A.; Zariffa, J. Analysis of the Hands in Egocentric Vision: A Survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**. [CrossRef] [PubMed]
27. Núñez, J.C.; Cabido, R.; Pantrigo, J.J.; Montemayor, A.S.; Vélez, J.F. Convolutional Neural Networks and Long Short-Term Memory for Skeleton-Based Human Activity and Hand Gesture Recognition. *Pattern Recognit.* **2018**, *76*, 80–94. [CrossRef]
28. Xia, Z.; Xu, F. Time-Space Dimension Reduction of Millimeter-Wave Radar Point-Clouds for Smart-Home Hand-Gesture Recognition. *IEEE Sens. J.* **2022**, *22*, 4425–4437. [CrossRef]
29. Dewi, C.; Chen, R.-C.; Zhuang, Y.-C.; Christanto, H.J. Yolov5 Series Algorithm for Road Marking Sign Identification. *Big Data Cogn. Comput.* **2022**, *6*, 149. [CrossRef]
30. Cheng, Y.T.; Patel, A.; Wen, C.; Bullock, D.; Habib, A. Intensity Thresholding and Deep Learning Based Lane Marking Extraction and Lanewidth Estimation from Mobile Light Detection and Ranging (LiDAR) Point Clouds. *Remote Sens.* **2020**, *12*, 1379. [CrossRef]
31. Chen, R.-C.; Manongga, W.E.; Dewi, C. Automatic Digit Hand Sign Detection With Hand Landmark. In Proceedings of the 2022 International Conference on Machine Learning and Cybernetics (ICMLC), Toyama, Japan, 9–11 September 2022.
32. Bose, S.R.; Kumar, V.S. In-Situ Recognition of Hand Gesture via Enhanced Xception Based Single-Stage Deep Convolutional Neural Network. *Expert Syst. Appl.* **2022**, *193*, 116427. [CrossRef]
33. Dewi, C.; Chen, R.C.; Yu, H. Weight Analysis for Various Prohibitory Sign Detection and Recognition Using Deep Learning. *Multimed. Tools Appl.* **2020**, *79*, 32897–32915. [CrossRef]
34. Dong, X.; Zhao, Z.; Wang, Y.; Zeng, T.; Wang, J.; Sui, Y. FMCW Radar-Based Hand Gesture Recognition Using Spatiotemporal Deformable and Context-Aware Convolutional 5-D Feature Representation. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 1–11. [CrossRef]
35. Ultralytics Yolo V5. Available online: https://github.com/ultralytics/yolov5 (accessed on 13 January 2021).
36. Long, J.W.; Yan, Z.R.; Peng, L.; Li, T. The Geometric Attention-Aware Network for Lane Detection in Complex Road Scenes. *PLoS ONE* **2021**, *16*, e0254521. [CrossRef] [PubMed]
37. Han, K.; Zeng, X. Deep Learning-Based Workers Safety Helmet Wearing Detection on Construction Sites Using Multi-Scale Features. *IEEE Access* **2022**, *10*, 718–729. [CrossRef]
38. Jiang, L.; Liu, H.; Zhu, H.; Zhang, G. Improved YOLO v5 with Balanced Feature Pyramid and Attention Module for Traffic Sign Detection. *MATEC Web Conf.* **2022**, *355*, 03023. [CrossRef]
39. Zhao, S.; Zhang, K. Online Predictive Connected and Automated Eco-Driving on Signalized Arterials Considering Traffic Control Devices and Road Geometry Constraints under Uncertain Traffic Conditions. *Transp. Res. Part B Methodol.* **2021**, *145*, 80–117. [CrossRef]
40. Dewi, C.; Chen, R.-C. Combination of Resnet and Spatial Pyramid Pooling for Musical Instrument Identification. *Cybern. Inf. Technol.* **2022**, *22*, 104. [CrossRef]
41. Arcos-García, Á.; Álvarez-García, J.A.; Soria-Morillo, L.M. Evaluation of Deep Neural Networks for Traffic Sign Detection Systems. *Neurocomputing* **2018**, *316*, 332–344. [CrossRef]
42. 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, 27–30 June 2016; pp. 779–788.
43. Deng, X.; Zhang, Y.; Yang, S.; Tan, P.; Chang, L.; Yuan, Y.; Wang, H. Joint Hand Detection and Rotation Estimation Using CNN. *IEEE Trans. Image Process.* **2018**, *27*, 1888–1900. [CrossRef]

44. Le, T.H.N.; Quach, K.G.; Zhu, C.; Duong, C.N.; Luu, K.; Savvides, M. Robust Hand Detection and Classification in Vehicles and in the Wild. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Honolulu, HI, USA, 21–26 July 2017.
45. Yang, L.; Qi, Z.; Liu, Z.; Liu, H.; Ling, M.; Shi, L.; Liu, X. An Embedded Implementation of CNN-Based Hand Detection and Orientation Estimation Algorithm. *Mach. Vis. Appl.* **2019**, *30*, 1071–1082. [CrossRef]