symmetry

MDPI

Article

A Novel Heteromorphic Ensemble Algorithm for Hand Pose Recognition

Shiruo Liu [1], Xiaoguang Yuan [1,*], Wei Feng [1], Aifeng Ren [1], Zhenyong Hu [1], Zuheng Ming [2], Adnan Zahid [3], Qammer H. Abbasi [3] and Shuo Wang [1]

[1] School of Electronic Engineering, Xidian University, Xi'an 710071, China
[2] Laboraory of Information Processing and Transmission, L2TI, Institut Galilée, University Paris XIII, 93079 Villetaneuse, France
[3] School of Engineering, University of Glasgow, Glasgow G12 8QQ, UK
* Correspondence: xgyuan@xidian.edu.cn; Tel.: +86-135-7211-2212

Abstract: Imagining recognition of behaviors from video sequences for a machine is full of challenges but meaningful. This work aims to predict students' behavior in an experimental class, which relies on the symmetry idea from reality to annotated reality centered on the feature space. A heteromorphic ensemble algorithm is proposed to make the obtained features more aggregated and reduce the computational burden. Namely, the deep learning models are improved to obtain feature vectors representing gestures from video frames and the classification algorithm is optimized for behavior recognition. So, the symmetric idea is realized by decomposing the task into three schemas including hand detection and cropping, hand joints feature extraction, and gesture classification. Firstly, a new detector method named YOLOv4-specific tiny detection (STD) is proposed by reconstituting the YOLOv4-tiny model, which could produce two outputs with some attention mechanism leveraging context information. Secondly, the efficient pyramid squeeze attention (EPSA) net is integrated into EvoNorm-S0 and the spatial pyramid pool (SPP) layer to obtain the hand joint position information. Lastly, the D–S theory is used to fuse two classifiers, support vector machine (SVM) and random forest (RF), to produce a mixed classifier named S–R. Eventually, the synergetic effects of our algorithm are shown by experiments on self-created datasets with a high average recognition accuracy of 89.6%.

Keywords: behavior feature extraction; deep learning; hand pose recognition; multi-classification

Citation: Liu, S.; Yuan, X.; Feng, W.; Ren, A.; Hu, Z.; Ming, Z.; Zahid, A.; Abbasi, Q.H.; Wang, S. A Novel Heteromorphic Ensemble Algorithm for Hand Pose Recognition. *Symmetry* **2023**, *15*, 769. https://doi.org/10.3390/sym15030769

Academic Editor: Sergei D. Odintsov

Received: 16 February 2023
Revised: 12 March 2023
Accepted: 15 March 2023
Published: 21 March 2023

1. Introduction

Hand pose recognition, a great subsidiary task of human behavior recognition, has been further studied to meet increasing demands in human–computer interaction areas such as medical treatment [1,2], robot control [3], and smart homes [4,5]. It has greatly attracted the attention of researchers, making it more practical to serve some related fields.

A whole gesture recognition process is roughly divided into data acquisition and preprocessing, and feature extraction and classification. Correspondingly, they all have their own specific research. For data acquisition and preprocessing, researchers achieve hand posture sequences in the real world in different modalities based on sensors, video images, and hand skeletons. Wearable sensors [6,7] would provide accurate measurements of hand posture and movement. However, such devices not only require precise calibration but also fail to capture the natural movement of human fingers due to their bulk, and are often very expensive [8]. Meanwhile, video images only depend on much cheaper cameras to obtain hand posture data, which is widely adopted and intensively studied. However, there still exist some drawbacks, including gesture image accounting for a small proportion of the whole picture, and background interference, which restricts the accuracy improvement of subsequent algorithms [9,10]. For skeleton-based gesture recognition, a relatively new modality has drawn some researchers' attentions due to its robustness to illumination variation and complex backgrounds [11,12].

For feature extraction and classification, there are two kinds of ways to design detectors to localize the human action of interest in data sequences. One is the traditional way [13], which tends to be time-consuming and costly due to handcrafted features by integral image. The other is feature extraction based on deep learning, which is more popular, accounting for its robustness in this academic field [14,15]. Although these methods have met demands in the past, in the face of the redundant, low-quality, mass data today [16], outstanding models [17–19] are often difficult to train and it is arduous to verify common features. In recent years, the research and development of the attention mechanism has greatly increased, which is a module that combines and emphasizes high-quality features [20–22].

Based on the above, some researchers have made great contributions to boosting these methods in some special applications. A gesture recognition framework under different illumination was proposed using symmetric patterns and a related luminosity-based filter with Microsoft Kinect sensor in [23]. Zaccagnino et al. [24] studied a set of touch-based gestures and used machine learning algorithms to determine whether it was possible to tell who was accessing a smartphone, a minor or an adult, to provide safeguards against threats online. Guarino et al. [25] introduced a method called touch gestures for soft biometrics (TGSB) to capture the age and gender traits of the users, which exploited images of touch gestures performed by users on mobile devices to train the pre-trained convolutional neural networks (CNNs). Hussain et al. used VGG16 [18] as the pre-training model and improved it into a classifier that could distinguish 11 categories [26]. The authors in [27] pioneered research on online karate action classification with an unsupervised learning algorithm.

Recently, the emergence of smart education [28,29] has called for more and more advanced technology to serve teachers and students. Based on early works, we propose our method to combine and improve some existing frameworks to classify hand poses for students' behavior recognition in electronic design automation (EDA) experimental courses at universities.

A novel heteromorphic ensemble algorithm for hand pose recognition is proposed. The contributions of this paper are as follows:

- For the hand detector YOLOv4-STD, the neck and prediction head of YOLOv4-tiny's network are improved to boost the directivity features and reduce the number of network layer parameters during model training.
- The hand joints feature extraction network, HandPose-PSA, is an improvement of the pipeline structure of the efficient pyramid squeezed attention network (EPSAnet), allowing the model to input pictures of any scale, not affected by the batch size. It retains more effective feature information and avoids information loss.
- A feature fusion method is proposed to aggregate the extracted feature information of hand joints into a one-dimensional vector for classification.
- For the classification of multiple actions, an S–R classifier is proposed, which combines the results of support vector machine (SVM) and random forest (RF) classifiers. The accuracy of gesture recognition is improved by the novel heteromorphic ensemble algorithm and a compromise video detection method is realized.

2. Related Work

2.1. Gesture Recognition Methods Based on Machine Learning

Dominio et al. [30] proposed a gesture recognition scheme based on depth data collected by depth cameras. Four different sets of feature descriptors were extracted from the data, considering the distance from fingertips to the palm and palm plane, the curvature of the hand contour, and the geometry of the palm area. Finally, a multi-class SVM classifier was used to recognize the performed gestures, which obtained a very high accuracy on standard datasets, and specialization ones collected for experimental evaluation. Their implementation without optimization was able to achieve about 10 fps. A light-intensity-invariant technique for gesture recognition [31], taking advantage of the principle that one skin tone looks different under changing light intensity but different skin tones may look

the same under changing light intensity, used directional histograms to identify unique features to recognize the features of gestures. ANN was used for gesture recognition with training images that came from a variety of sources, including online searches and manual collection.

2.2. Gesture Recognition Methods Based on Deep Learning

Aiming at the problem of abnormal gesture recognition in RGB-D video, a fine fusion model combining the Res-C3D network and long short-term memory (LSTM) was proposed [32]. The key to this design was to learn discriminative representations of abnormal gesture sequences by fusing multiple features from different models. Then, a fusion scheme was proposed to fuse the classification results through the weight fusion layer, which adaptively obtained the dominant weight of a class through training. Their experimental results showed that the proposed method can effectively distinguish abnormal gesture samples and achieve the best performance on the IsoGD dataset. However, the fusion strategy is not deep enough in extracting and representing abnormal gesture features. Zhang et al. [33] attempted to use vision-based sensors to sample the gestures of the target human body at first; then, part of the stacked hourglass network structure was improved into parallel modules, which introduced an attention mechanism to reduce the influence of the complex real environment. Zhang et al. [34] proposed the improved YOLOv3 algorithm [35], which was used by the Raspberry PI to connect the monocular camera for data acquisition and classification results display. In this method, IoU distance was introduced to replace Euclidean distance to improve the recognition accuracy of the system.

According to Table 1, researchers often design feature descriptors to construct features based on machine learning. Although good results can also be achieved, the results depend on the descriptors. In contrast, deep learning models are used to construct more general features from datasets. Therefore, training a good network model is necessary for hand detection and gesture recognition. The motivation of this paper is to identify gestures in videos better and faster based on the improved model, which reduces the number of training parameters in the model and reduces the computational burden.

Table 1. A summary of works for effective gesture recognition.

Authors	Method	Model	Highlights
Dominio et al. (2014) [30]	Machine learning (ML)	A multi-class SVM classifier	Four different sets of feature descriptors; achieved about 10 fps; based on depth data.
Chaudhary et al. (2018) [31]	ML	ANN	A light-intensity-invariant technique.
Lin et al. (2018) [32]	Deep learning (DL)	The Res-C3D network and LSTM	A weight fusion scheme; best performance on the large-scale isolated gesture dataset (IsoGD).
Zhang et al. (2019) [33]	DL	The stacked hourglass network	An attention mechanism to reduce the influence of a complex real environment.
Zhang et al. (2022) [34]	DL	YOLOv3 algorithm	IoU distance to improve the recognition accuracy to around 90%; to visualize the results by Raspberry PI.

3. Materials and Methods

3.1. Hand Pose Recognition from Video Frames

For deep learning to cope with the task, this paper opts for vertically positioned cameras to make data acquisition much easier; pays more attention to the improvement of the state-of-the-art model to enhance the common features; reduces the cost of trained parameters and computational force; and makes the combined effect of the processing of all parts optimal, which is more suitable for this problem, as illustrated in Figure 1.

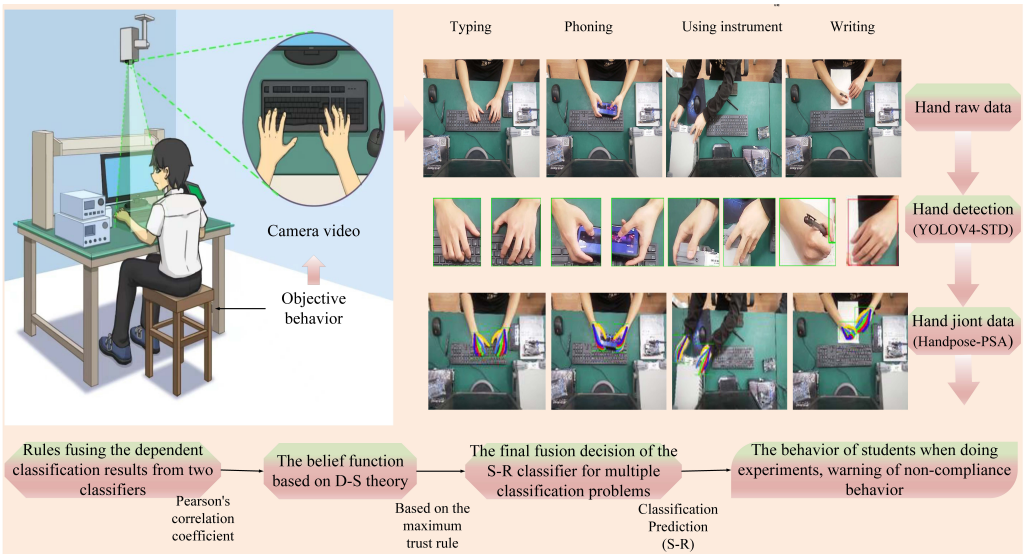

Figure 1. The overall method for hand pose recognition.

Currently, because there is no dataset on the hands of students in experimental classrooms, it is important to guarantee the performance of the model in this paper with the aid of mainstream datasets and homemade datasets—that is, standard datasets and part of homemade datasets were used when training the model. The collected data were used to verify a novel heteromorphic ensemble algorithm of this work. We collected 230 videos as a sample of the homemade dataset, of which 180 videos with a duration of about six seconds recorded content of a single behavior and the remaining 50 videos with a duration of about two minutes recorded all behaviors. The participants came from graduate students, doctors, and teachers of the subject, all of whom were aware of the data collection and agreed to it. The behaviors are mainly divided into using a computer, writing, experimental debugging, and playing with a mobile phone. Using a computer includes typing and using a mouse, which accounts for 30% and 25% of the size of the datasets, respectively. Experimental debugging includes using an instrument and using an experimental board (boarding), which account for 10% and 5% of the size of the video set, respectively. Using a mobile phone (phoning) and writing activities account for 18% and 12% of the size of the data set, respectively.

3.2. Model Framework

3.2.1. Detector: YOLOv4-STD

Since the publication of the first work LeNet [36], the deep learning models based on CNN have led researchers to continuously improve and optimize the models to pursue much stronger feature expression ability of models. There are two types of network construction methods emerging. One idea is to widen the width of the network, such as inceptionNet [37], which is more lightweight than MobileNetV2 [38]; the other is to increase the number of layers. For example, VGG16, VGG19, and ResNet-50 [18,19] are deep learning models with 16, 19, and 50 layers, respectively. The model improvements include their own depth and width. However, when researchers re-examined the construction of these networks, a construction method of the attention mechanism was proposed to consider the relationship between feature channels. Based on this, the squeeze-and-excitation network (SENet) [21] was proposed. Squeeze and excitation are two very critical structures in SENet. The motivation is the desire to explicitly model the interdependence between feature channels. Specifically, the importance of each feature channel is automatically obtained by learning; then, the useful features are promoted and the features that are not useful for the current task are suppressed according to their importance.

Once the one-stage detector, you only look once (YOLO) algorithm, was introduced, it was favored in the field of target detection for its excellent performance of rapid regression of

categories and locations. So, this work introduces the YOLOv4 algorithm for the detection of hands. The pipelines of YOLOv4 and YOLOv4-tiny [39] models consist of a backbone feature extraction network and a multi-scale prediction network including the feature fusion network and prediction head. Compared with YOLOv4, YOLOv4-tiny turns the backbone feature extraction network CSPDarkNet53 into lightweight CSPDarknet53-Tiny, and the prediction heads in the prediction network are changed from three to two.

Based on the above analysis, after the feature extraction network, the image contents are extracted from bottom to top and the feature images of different sizes are obtained. The shallow feature maps contain detailed information on graphics, while the deep feature maps summarize the high-level semantic information. It can be seen that the result of only using the latter to predict leads to higher prediction speed and low memory consumption. By contrast, the former with more abundant detailed information is more than enough to deal with simple goals. So, it is necessary to improve a multi-scale prediction network of detectors.

In this paper, because only one category of hand is detected, Yolov4-tiny is selected. At the same time, in combination with the purpose of more accurate recognition and less computational burden in this paper, we recombine the multi-scale prediction network of YOLOv4-tiny to capture and extract missing features through channel attention SENet so as to exact channel attention and boost the backbone's features. Additionally, different from the output of the original network, the shallow features are extracted and processed, and then fused into the original feature prediction channel to obtain more robust features to establish the model and improve the model's prediction ability. This is vividly demonstrated in Figure 2.

There are two major improvements for the detector. On the one hand, the outputs of the three residual convolutions are first processed by SEBlock and then sent to the feature prediction network. On the other hand, the output is increased by sampling from CSPBlock_body1. First, the feature map size converted to 26×26 after pooling is matched with the output feature map of CSPBlock_body2. Then, its output abbreviated as Y1 is obtained from two times DBL (DarkConv_BN_LeakReLU) convolution processing. Subsequently, Y1 is added to the feature vector with a size of $26 \times 26 \times 256$ to combine the number of channels. Finally, Out2 is obtained after three times of 3×3 convolution with a 1×1 convolution layer. Similarly, Y1 will go through one Max Pooling layer and two DBL layers to obtain the vector abbreviated as Y2. Then, Y2 is concatenated to the feature vector with a size of $13 \times 13 \times 512$. Out1 is obtained after 1×1 convolution. Finally, the YOLOV4-STD model is trained so that the output feature vectors Out1 and Out2 are predicted and decoded to achieve the goal of hand detection in video frames.

3.2.2. Feature Extraction: HandPose-PSA

In this paper, a method based on deep learning is applied to extract information from joint-based gestures. Namely, input data from the regression of YOLOv4-STD are reflected in feature maps and formalized as an objective function to obtain the signals. All this should be entirely due to the improved efficient pyramid squeezed attention network (EPSAnet) [22]. The authors of [22] proposed a lightweight and effective attention method called the pyramid extrusion attention (PSA) module; by replacing the 3×3 convolution in the ResNet bottleneck block with the PSA module, a new representation block was obtained, called EPSA.

Generally, the size of the training image is usually fixed to 224×224, and some important features may be lost in the process of image conversion. Therefore, the spatial pyramid pooling (SPP) [14] module is introduced at the end of the feature extraction network in EPSAnet, which could allow the model to input pictures of any scale, thus retaining more effective feature information and avoiding information loss. Moreover, SPP generates a fixed-length representation of any region and avoids the repeated calculation of convolution features. Importantly, the batch size becomes smaller due to a large image size, which could lead to a decrease in model accuracy. In order to solve this problem and improve the model detection effects, the EvoNorms-S0 [40] normalized activation layer

is applied to replace the batch normalization (BN) and rectified linear unit (ReLU) in the original ESPAnet. The improved EPSAnet is illustrated in Figure 3.

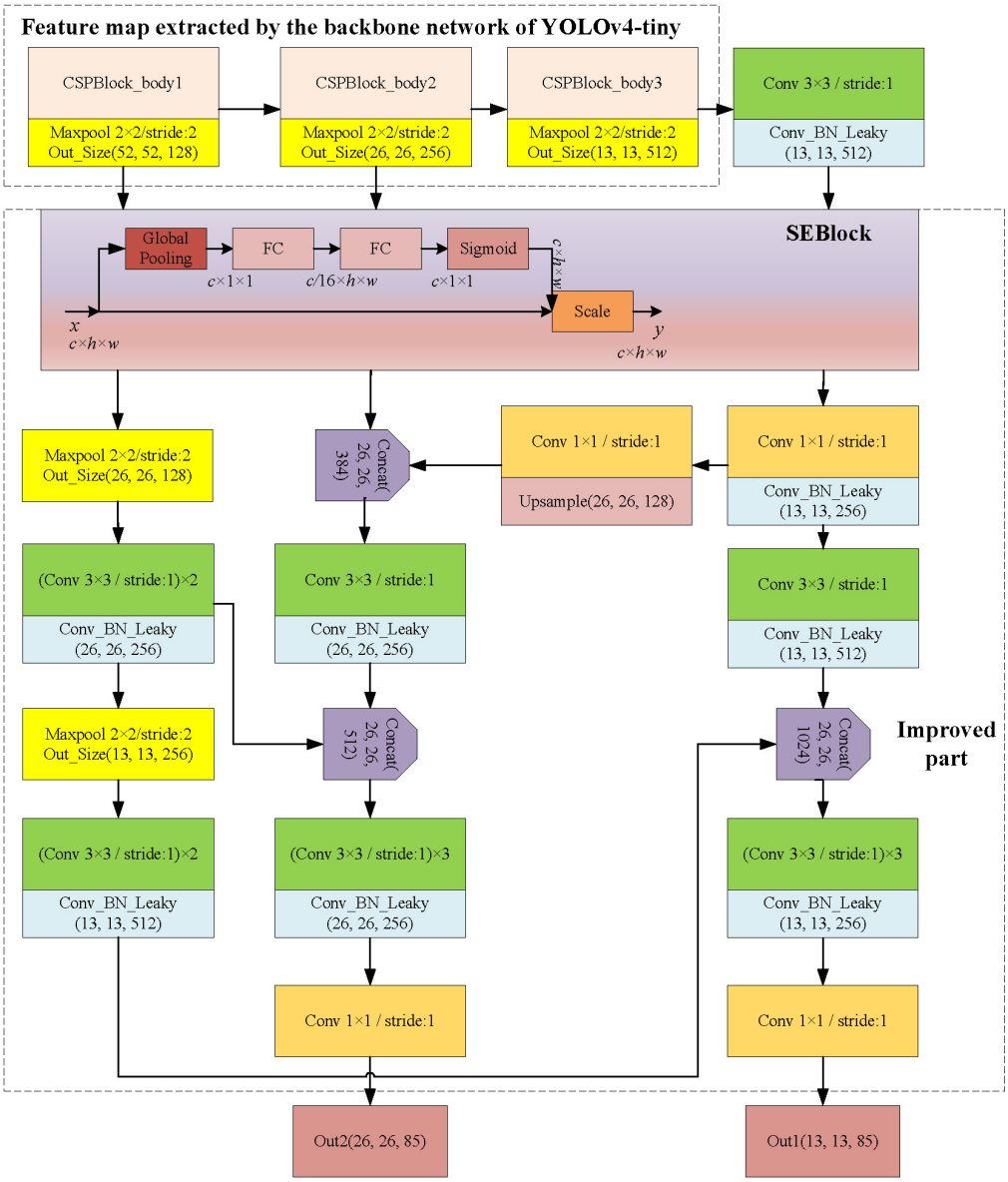

Figure 2. The improved YOLOv4-STD model.

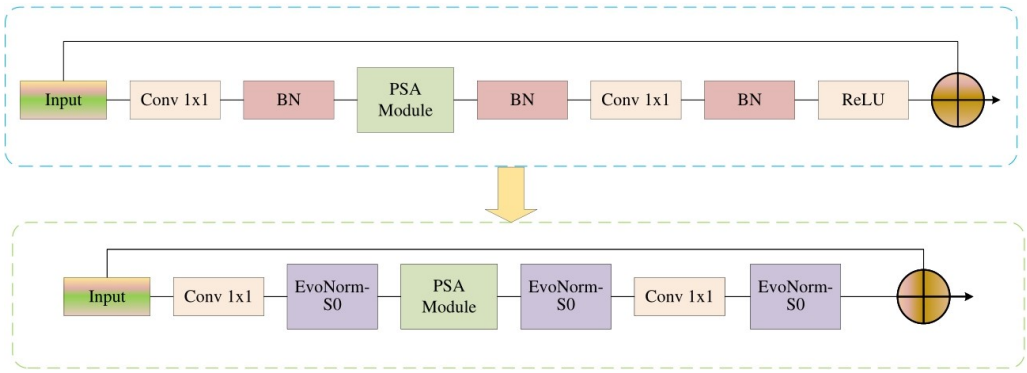

Figure 3. Some components have been added to improve the backbone EPSAnet. The blue dotted box is the original pipeline and the green one is the improved one.

3.3. Data Processing

3.3.1. Detector Result Optimization

Because the output calculated by the detector network is disabled to perform accurate object position, data processing that removes the prediction box with confidence less than the preset threshold is highly significant to obtain a high reliability and low redundancy box regarding a hand. This paper also filters out the box with the largest score of the same kind in a certain area of the picture through a non-maximum suppression ratio. Here, the loss function transforms errors into model predictability as a single degree quantity is formulated as three categories. Firstly, for a much more stable target regression box, this paper considers the coincidence degree, aspect ratio, and penalty factor of the box based on the ratio of the intersection and union of the predicted and true boxes of an object (IoU) to employ the regression loss function named CIoU,

$$loss_{CIoU} = 1 - IoU + (\rho^2(b, b^{gt}))/c^2 + \alpha v, \tag{1}$$

where $v = \frac{4}{\pi^2}\left(arctan\frac{w^{gt}}{h^{gt}} - arctan\frac{w}{h}\right)^2$, $\alpha = \frac{v}{(1-IoU)+v}$ and b^{gt}, h^{gt}, w, h represent the length and width of the prediction frame and the real frame, respectively. Moreover, $\rho^2(b, b^{gt})$ indicates the Euclidean distance between the center point of the prediction frame and the real frame.

Secondly, the confidence loss function containing the target in the box named *obj*, and not in the box named *noobj*, is calculated by cross-entropy

$$\begin{aligned} loss_c = &- \sum_{i=0}^{K\times K}\sum_{j=0}^{M} I_{ij}^{obj}\left[\hat{C}_i^j log\left(C_i{}^j\right) + (1-\hat{C}_i^j) log(1-C_i{}^j)\right] \\ &-\lambda_{noobj}\sum_{i=0}^{K\times K}\sum_{j=0}^{M} I_{ij}^{noobj}\left[\hat{C}_i^j log\left(C_i{}^j\right) + (1-\hat{C}_i^j) log(1-C_i{}^j)\right], \end{aligned} \tag{2}$$

where M is the number of anchors generated from grids totaling $K \times K$, $\hat{C}_i^j$ indicates parameter confidence, and I_{ij}^{obj} valued 0 or 1 is responsible for the target anchor j in the grid i. Thirdly, the classification loss function denoted $L_{classes}$ is also calculated by cross-entropy,

$$loss_{classes} = -\sum_{i=0}^{K\times K} I_{ij}^{obj} \times \sum_{C\in classes}[\hat{p}_i(c) log(p_i(c)) + (1-\hat{p}_i(c)) log(1-p_i(c))], \tag{3}$$

where $p_i(c)$ is probability of classification and I_{ij}^{obj} is the same as mentioned above.

Importantly, $loss_{obj}$ is shown as

$$loss_{obj} = loss_{CIoU} + loss_c + loss_{classes}. \tag{4}$$

3.3.2. Feature Merging Criterion

Having achieved hand-joint signals from the improved hand-joint detector, named EPSAnet, we utilize some formulas to induce the abstract interpretation of gestures from the one-dimensional signal vectors because the optimal fusion strategy can improve the performance of the trained network [41].

According to the existing theory [42], the hand joint points are numbered from 0 to 20, totaling 21 positions, where the relationship between them is defined as the absolute position, the relative position of adjacent finger joint points, and the combined position of finger internal relations nodes, as illustrated in Figure 4.

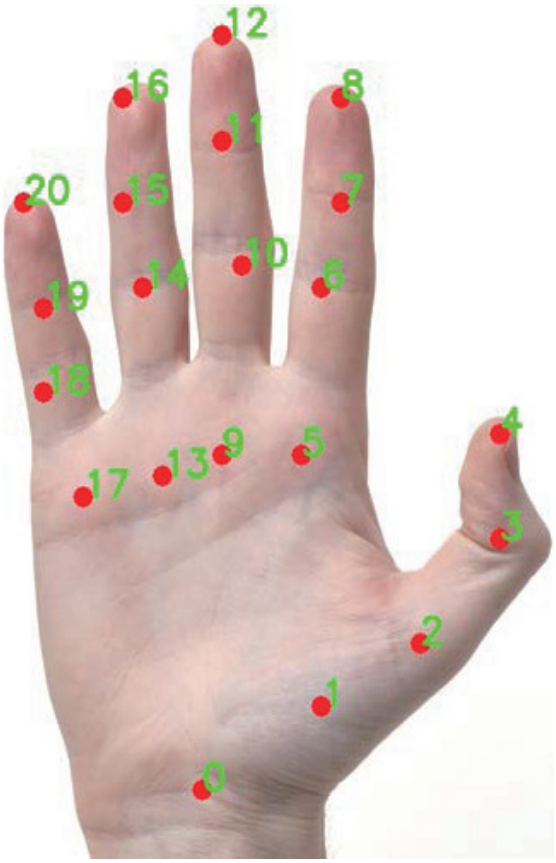

Figure 4. Hand joint nodes are calibrated, and the wrist is the first point. Then, starting from the thumb, a key point is assigned to each bone node of the palm positions to visualize hand key point numbering. Specifically, the red points represent the joint position, and the numbers are the count of the red points.

Correspondingly, there are also three types of feature vectors that depict at length semantic interpretation of hand behavior as follows. The length and width of the detection frame are denoted as l and h, respectively. Firstly, absolute position features directly portray the pose of the hand

$$F_a = [X_0, Y_0, X_1, Y_1, \ \ldots, \ X_{20}, Y_{20}], \tag{5}$$

where $X_i = \frac{x_i - x_c}{l}$ and $Y_i = \frac{y_i - y_c}{h}$ are noted as the normalized coordinate position based on the center of the detection frame. Here, $(x_c, \ y_c)$ is the center position of the detection frame and $(x_i, \ y_i)$ is the detection position labeled i.

Secondly, the 21 hand key points are divided into four groups—(1, 5, 9, 13, 17), (2, 6, 10, 14, 18), (3, 7, 11, 15, 19), (4, 8, 12, 16, 20)—to obtain the relative positions of adjacent points of each group.

$$X_{i,\,j} = \frac{x_{i,\,j} - x_{i,\,j+1}}{l}, \tag{6}$$

$$Y_{i,\,j} = \frac{y_{i,\,j} - y_{i,\,j+1}}{h}, \tag{7}$$

then, obtain the final vectors

$$F_b = [X_{11}, \ Y_{11}, X_{12}, \ Y_{12}, \ \ldots, \ X_{44}, \ Y_{44}], \tag{8}$$

where $(x_{i,\,j}, \ y_{i,\,j})$ locates hand joint points j from group i.

Thirdly, after key points are grouped as (i, j, k, v), i.e., (1, 2, 3, 4), (5, 6, 7, 8), (9, 10, 11, 12), (13, 14, 15, 16), and (17, 18, 19, 20), the vector is received as

$$F_c = [R_{1,\,2,\,3},\ R_{2,\,3,\,4},\ R_{5,\,6,\,7},\ \ldots,\ R_{18,\,19,\,20}], \tag{9}$$

where $d_{i,\,j} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$, $R_{i,\,j,\,k} = \frac{(x_i - x_j) \times (x_k - x_j) + (y_i - y_j) \times (y_k - y_j)}{d_{i,\,j} \times d_{j,\,k}}$.

Besides, given continuous gesture movement, we consider contrasting the same hand key point in different frames to acquire direction and range. In the end, the feature to illustrate it is shown as

$$F_d = [X_0^{n,\,n-m},\ Y_0^{n,\,n-m},\ \ldots,\ X_{20}^{n,\,n-m},\ Y_{20}^{n,\,n-m}, X_0^{n,\,n-2m},\ Y_0^{n,\,n-2m},\ \ldots,\ X_{20}^{n,\,n-mT},\ Y_{20}^{n,\,n-mT}]. \tag{10}$$

Here, we suppose the video frame window size to be $m \times T$; then,

$$X_i^{n,\,n-a} = \frac{x_i^n - x_i^{n-a}}{l}, \tag{11}$$

and

$$Y_i^{n,\,n-a} = \frac{y_i^n - y_i^{n-a}}{h}, \tag{12}$$

display the position difference between every frame and the current frame in a fixed time as the motion feature, where $(x_i^n,\ Y_i^n)$ and $(x_i^{n-a},\ Y_i^{n-a})$ represent hand joints i at frame n and $n - a$.

Ultimately, the above feature vectors are spliced to evaluate the optimal behavior as follows:

$$F = F_a + F_b + F_c + F_d. \tag{13}$$

3.3.3. Fusion Decision Classification Criteria

In machine learning, a classifier is a supervised learning method. Examples include SVM and RF. SVM is a widely used classifier in classification and recognition problems. Compared with neural networks and other classification algorithms, SVM has a faster classification speed and better performance in the case of small samples (thousands). For high feature latitude and a small number of samples, an SVM classifier with a linear kernel can be applied. The RF classifier is composed of multiple decision trees, and each decision tree selects part of the features as input. Due to the high dimension of the detected behavior action features, the random forest model is just as good at dealing with this and does not need to reduce the dimension of the feature selection. More importantly, RF can determine the importance of features and the interaction between features, reducing the impact of unimportant features on discrimination.

For the extracted student hand feature vector, a classifier needs to be designed to complete the last step of behavior recognition. At present, most behavior algorithms are based on a single classifier to complete the classification task; however, experiments have shown that a single classifier often has a good recognition effect for one or several behaviors. With the change of category, it needs to adjust parameters and retrain, which has certain limitations [43]. To solve this problem and improve the accuracy of behavior recognition, this paper uses the method of multi-classifier fusion to complete the classification of students' behavior characteristics based on the SVM and RF classifiers because we pay more attention to how to better fuse results of these two classifiers. Statistically good hyperparameters have been used in this work from reference [44] and related works. The parameter selection of SVM is mainly the penalty factor, and the parameter selection of RF includes two parts: RF framework and decision tree.

The decision classification here uses the fusion discrimination rule based on D–S evidence theory [45] and introduces the Pearson correlation coefficient to represent the evidence similarity of the two classifiers to obtain the fusion function based on the maximum trust rule.

Foremost, the probability assignment function of one classifier can be expressed as

$$m_{n,k} = \frac{p_k}{\sum_{k=1}^{N} p_k}, \tag{14}$$

where p_k represents the probability output of the kth category of classifier n valued 1 or 2, and N represents the total number of categories.

Then, this paper uses the fusion discrimination rule based on D–S evidence theory to formulate the probability assignment function,

$$\begin{aligned} m_k(k) &= (m_1 \oplus m_2)(k) \\ &= \frac{1}{1-K} \sum_{X_1 \cap X_2 = k} m_{1,k}(X_1) \cdot m_{2,k}(X_2), \end{aligned} \tag{15}$$

where $K = \sum_{X_1 \cap X_2 \neq \emptyset} m_{1,k}(X_1) \cdot m_{2,k}(X_2)$.

Meanwhile, we introduce the Pearson correlation coefficient to represent the evidence similarity of the two classifiers,

$$\rho_{m_i,m_j} = \frac{E\left[(m_i - \mu_{m_i})(m_j - \mu_{m_j})\right]}{\sigma_{m_i}\sigma_{m_j}} \quad (i, j = 1, 2, \ \ldots, \ n), \tag{16}$$

where μ_{m_i} and μ_{m_j} denote the mean values, σ_{m_i} and σ_{m_j} denote variance.

If the Pearson correlation coefficient, whose threshold in this experiment is set to 0.3 between evidences, is greater than the threshold, there is no conflict between the two classifiers. On the contrary, there is a conflict between the two classifiers. On this foundation, together with the combined focal elements meeting the Bayesian independence condition, the trust function $Bel_k(k)$ can be formulated as $Bel_k(k) = m_k(k)$. Ultimately, based on the maximum trust rule, the fusion function can be obtained as

$$d(x) = arg\ max Bel(1), Bel(2), \ \ldots, \ Bel(N), \tag{17}$$

where x is the number of categories.

4. Model Evaluation

4.1. YOLOv4-STD

There are two main sources of the hand target detection dataset. The first is the 32,417 pictures selected from the TV hand and coco hand dataset [46]. The second is the pictures intercepted from this paper's method of the video taken from a specific angle in the electronic design automation (EDA) experimental class, with a total of 12,600 pictures made into a VOC format dataset with Labelimg. Each labeled image in the dataset corresponds to an XML file, which contains the category and border position of the object to be detected in the labeled image. For the student hand detection dataset, 1/3 of the self-labeled data are used as the test set named Stu-hand-test and the rest are used as the training set named hand-train.

The enhanced YOLOv4-STD algorithm, which is built on a CSPDarknet-Tiny backbone feature network, improves the feature extraction of the output of the residual layers and incorporates a channel attention mechanism to obtain much more effective information. Consequently, more semantic data about the targets' information can be extracted. To assess the algorithm's ability to detect objects, the trained YOLOv4-tiny and YOLOv4-STD models are tested on the Stu-hand-test test set, as depicted in Figure 5.

As shown by the comparison of PR curves in Figure 5, when the number of correct categories is the same, the prediction accuracy of YOLOv4-STD is the highest, indicating that YOLOv4-STD can capture features outside the capture range of YOLOv4-tiny during target detection.

Table 2 compares the parameters of the improved model presented in this study with those of YOLOv4, and YOLOv4-tiny—the lightweight model from YOLOv4.

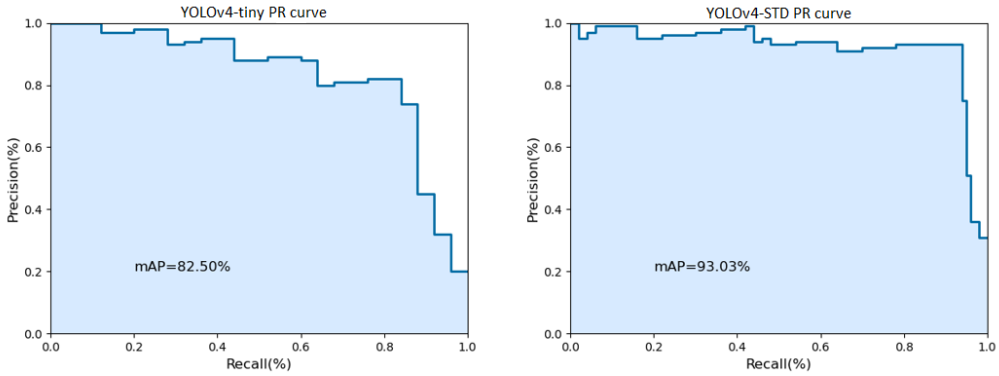

Figure 5. Plots of precision and recall for the original and improved models.

Table 2. Network performance comparison for hand detection.

Network	Backbone Layers	Parameters	FLOPs (G)	mAP (%)	FPS (f/s)
YOLOv4	53	63,056,606	70.90	92.97	52
YOLOv4-tiny	13	6,056,606	9.50	82.50	259
YOLOv4-STD	13	6,557,842	10.9	93.03	230

The testing results indicate that, on the one hand, YOLOv4-STD has only 13 backbone feature extraction network layers compared with YOLOv4, which reduces the number of parameters by 9.6 times while increasing the number of images processed per second by 5.6 times. The quantity of calculation drastically reduced and the accuracy on average marginally increased. Contrarily, the number of parameters and calculations in the YOLOv4-STD model grew in comparison to YOLOv4-tiny but the average accuracy still managed to reach 93.03%. As a result, YOLOv4-STD has a good detection effect for hand detection based on the background of the experimental classroom.

4.2. HandPose-PSA

Based on the enhanced HandPose-PSA network, we compare performance by replacing our feature extraction network with other famous backbones. The datasets meeting the experiment were mainly selected from the Large-scale Multiview 3D Hand Pose Dataset [47], totaling 49,560 pictures, and used to generate the handpose-test by labeling the outputs from our YOLOv4-STD detector, totaling 2096 pictures.

During the HandPose-PSA network model training, we set experimental conditions with 256 batch sizes and 100 epochs. Additionally, the origin learning rate is 1×10^{-3}, and the loss function is the cross-entropy loss function. Two indicators are emphasized, floating-point operations per second (FLOPs) and mean average precision (mAP), to evaluate the model performance as a result of several backbones. Besides, the number of parameters in the trained model such as bias and weight also reflect the model's performance indirectly. The outcomes are exhibited in Table 3.

Table 3. The different backbones of model performance for hand joints feature extraction.

Network	Parameters	FLOPs (G)	mAP (%)
VGG19-bn	143.67	19.78	75.33
MobileNetV2	3.40	0.41	64.32
EPSANet-50 (Small)	22.76	3.62	83.35
ResNet-50	26.18	4.12	81.66
This paper	22.77	3.77	89.21

We can quickly conclude from the table that the greatest parameters and FLOPs are found in the VGG19 model, which are 143.67 M and 19.78 G, respectively, with mobilenetv2 having the least FLOPs and parameters. This paper, however, has the highest mAP on the test set of 89.21%, demonstrating that including the SPP layer could improve the identification, expand the network, and effectively increase the accuracy of hand key point detection. Additionally, it is crucial to note that the PSA dramatically increased detection accuracy by comparing the mAP between ResNet-50 and EPSANet-50.

5. Experimental Results

The experiment of this paper is to complete the training and testing on the desktop operating system with Ubuntu Linux, distribution version 18.04, as the system kernel. The parallel computing engine adopts Nvidia's CUDA version 11.2.2.

The fusion classification of the two classifiers was realized based on the fusion decision rules mentioned above, which was conducted during training and testing of the dataset illustrated in part II. Then, the three classifiers were applied for classification. Furthermore, to classify the behaviors directly in the video captured by the camera, we tested the influence of different frame windows on the classification results.

Firstly, we conducted preliminary experiments to find some suitable values for the hyperparameters to set the Pearson correlation coefficient.

Table 4 shows the posterior probability output of the SVM classifier model and RF classifier model after training in each category. According to the expression of the Pearson correlation coefficient, the Pearson correlation coefficient between the two classifiers is calculated to be 0.49, which is greater than the threshold set in this paper by 0.3, indicating that the two classifiers do not conflict in behavior classification.

Table 4. Posterior probabilities for six gesture categories under two classifiers.

	Typing	Writing	Phoning	Boarding	Clicking	Using Instrument
SVM	0.303	0.225	0.257	0.056	0.138	0.021
RF	0.448	0.043	0.116	0.044	0.270	0.079

Moreover, we carried out the classification experiments of three classifiers on the dataset to verify the performance of the multi-class fusion classifier.

The S–R classifier's average recognition accuracy is 1.1% superior to that of the RF classifier and 2% superior to the SVM classifier, which is shown in Table 5.

Table 5. Recognition accuracy of gesture categories under three classifiers.

Classifier	SVM	RF	S–R
Accuracy (%)	87.6	88.5	89.6

Further, the videos have a frame rate of 30 fps in the student behavior dataset made from the experimental class, and each behavior action has a varied duration. The classifier model's classification abilities significantly increase if it learns all of the behavior features. When the video frame window size was 1, 5, 10, 15, 20, 25, and 30, respectively, hand behavior features were extracted to identify the ideal action window size. Finally, training and testing were carried out using data from the experimental classroom behavior dataset to contrast the SVM classifier's and the RF classifier's classification accuracies with various time windows.

It can be seen from Table 6 that each action has different optimal time window sizes, and different time window sizes also have an impact on the classification results. This is because the actions are continuous in time. Obtaining the hand key points of continuous characteristic changes with appropriate window sizes can effectively reduce behavior misjudgments. Finally, according to the same optimal time window size (25 frames) of the

two classifiers, the time window sizes of 1 frame and 25 frames were selected to test the S–R classifier and validate the discovery, as shown in Table 7.

Table 6. Recognition accuracy (%) of the two classifiers in different time windows for each action.

SVM / RF	Typing	Writing	Phoning	Boarding	Clicking	Using Instrument	Average
1	90.3	79.2	82.6	72.9	82.0	76.3	83.2
5	93.9	84.3	86.4	79.3	86.9	81.2	87.6
10	94.1	85.1	87.4	80.7	87.5	82.0	88.2
15	94.1	85.1	87.4	81.4	87.5	82.5	88.3
20	93.8	85.3	87.5	83.5	87.8	83.7	88.5
25	93.4	85.2	87.5	84.3	87.8	84.1	88.6
30	93.0	84.3	87.1	84.4	87.1	84.2	88.1
1	92.3	81.6	76.3	72.9	85.7	74.3	83.7
5	94.7	89.3	81.4	80.1	89.9	83.0	88.5
10	95.2	89.8	82.1	81.7	88.6	84.2	88.7
15	95.1	89.7	82.3	81.7	90.0	85.3	89.2
20	95.1	88.9	82.1	83.4	90.8	85.4	89.3
25	94.8	88.4	82.0	83.8	91.1	86.1	89.3
30	94.5	88.1	81.9	84.3	90.3	86.4	89.0

Table 7. S–R's probability of behavior recognition in the specific time window size.

S–R Prediction Probability	Typing	Writing	Phoning	Boarding	Clicking	Using Instrument
Video time window size: 1	0.91	0.81	0.82	0.73	0.85	0.76
Video time window size: 25	0.95	0.88	0.87	0.95	0.91	0.87

6. Conclusions

We have examined how to know what a person is doing in a sheltered environment such as a laboratory. Hands, as a flexible body organ, have been used to infer the behavior of students in the experiment, cleverly avoiding the occlusion problem caused by the messy experimental environments and experimental equipment. For the sake of eschewing infringing on facial privacy, we only capture hand behaviors to reflect the individual's action through a perpendicular camera. Firstly, the YOLOv4-STD detection network intercepts pictures only containing hand data, which reduces the number of parameters and the computational burden. Next, elaborate rules combine the joint-based gestures achieved by HandPose-PSA into enhanced features. Then, the S–R classifier, a combination fusion decision method, achieved greatly improved recognition of six behaviors and tested the influence of different-sized time windows on classification results. Our results show that coupled feature extraction and the fusion classifier decision aimed at multiple classification problems can further improve the accuracy of behavior recognition. Besides, the method of selecting an appropriate video frame window size to identify a video directly for feature classification carries a trade-off. However, the identified behavior categories are the frequent experimental classroom behavior categories from the electronic experimental class of college students; so, the category items cannot cover all the student behaviors in an experimental classroom. Therefore, in the actual detection, the behaviors with low frequency are inevitably classified into the prescribed categories, resulting in classification errors. Moreover, it remains a challenging task for real-time video detection to determine the start and stop of actions or to identify crossover actions.

By improving the famous network architecture, this paper realizes the symmetric transformation from the raw real world to the annotated real world by using video as the medium. In the future, according to the statistical data of students' inappropriate behaviors in the experimental class, we can give early warnings according to the students' behaviors and urge them to improve their learning initiative. Further, according to the obtained

statistical data analysis and evaluation of students' experimental classroom performance, we can reduce the traditional experimental teaching of teachers to give mechanical and heavy teaching workloads. To realize the purpose of teaching reform is to realize the common good in the development of teachers and students.

Author Contributions: Conceptualization, S.L., X.Y., and W.F.; methodology, S.L. and X.Y.; software, Z.H.; validation, S.L. and Z.H.; resources, A.R., A.Z., and Q.H.A.; data curation, A.R., A.Z., Q.H.A., and Z.M.; writing—original draft preparation, S.L.; writing—review and editing, S.L., X.Y., A.R., W.F., and S.W.; visualization, S.L. and Z.H.; supervision, X.Y. and A.R.; project administration, X.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the National Natural Science Foundation of China under grant 62201438.

Informed Consent Statement: Informed consent was obtained from all subjects who were not underage involved in the study.

Data Availability Statement: Not applicable.

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

References

1. Swindells, C.; Quinn, K.I.; Dill, J.; Tory, M.K. That one there! Pointing to establish device identity. In Proceedings of the ACM Symposium on User Interface Software and Technology, Paris, France, 27–30 October 2002.
2. Nickel, K.; Stiefelhagen, R. Pointing gesture recognition based on 3D-tracking of face, hands and head orientation. In Proceedings of the International Conference on Multimodal Interaction, Vancouver, BC, Canada, 5–7 November 2003.
3. Goza, S.M.; Ambrose, R.O.; Diftler, M.A.; Spain, I.M. Telepresence Control of the NASA/DARPA Robonaut on a Mobility Platform. In Proceedings of the CHI 2004 Conference on Human Factors in Computing Systems, Vienna, Austria, 24–29 April 2004; Association for Computing Machinery: New York, NY, USA, 2004; pp. 623–629. [CrossRef]
4. Nishikawa, A.; Hosoi, T.; Koara, K.; Negoro, D.; Hikita, A.; Asano, S.; Kakutani, H.; Miyazaki, F.; Sekimoto, M.; Yasui, M.; et al. FAce MOUSe: A novel human-machine interface for controlling the position of a laparoscope. *IEEE Trans. Robot. Autom.* **2003**, *19*, 825–841. [CrossRef]
5. Schultz, M.E.; Gill, J.; Zubairi, S.; Huber, R.; Gordin, F.M. Bacterial Contamination of Computer Keyboards in a Teaching Hospital. *Infect. Control Hosp. Epidemiol.* **2003**, *24*, 302–303. [CrossRef] [PubMed]
6. Dipietro, L.; Sabatini, A.; Dario, P. A Survey of Glove-Based Systems and Their Applications. *IEEE Trans. Syst. Man Cybern. Part C* **2008**, *38*, 461–482. [CrossRef]
7. Rashid, A.; Hasan, O. Wearable technologies for hand joints monitoring for rehabilitation: A survey. *Microelectron. J.* **2019**, *88*, 173–183. [CrossRef]
8. Chen, W.; Yu, C.; Tu, C.; Lyu, Z.; Tang, J.; Ou, S.; Fu, Y.; Xue, Z. A Survey on Hand Pose Estimation with Wearable Sensors and Computer-Vision-Based Methods. *Sensors* **2020**, *20*, 1074. [CrossRef]
9. Al-Shamayleh, A.S.; Ahmad, R.; Abushariah, M.A.M.; Alam, K.A.; Jomhari, N. A systematic literature review on vision based gesture recognition techniques. *Multimed. Tools. Appl.* **2018**, *77*, 28121–28184. [CrossRef]
10. Ohn-Bar, E.; Trivedi, M.M. Hand Gesture Recognition in Real Time for Automotive Interfaces: A Multimodal Vision-Based Approach and Evaluations. *IEEE Trans. Intell. Transp. Syst.* **2014**, *15*, 2368–2377. [CrossRef]
11. Devineau, G.; Moutarde, F.; Xi, W.; Yang, J. Deep Learning for Hand Gesture Recognition on Skeletal Data. In Proceedings of the 2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018), Xi'an, China, 15–19 May 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 106–113. [CrossRef]
12. Liu, J.; Liu, Y.; Wang, Y.; Prinet, V.; Xiang, S.; Pan, C. Decoupled Representation Learning for Skeleton-Based Gesture Recognition. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 5750–5759. [CrossRef]
13. Viola, P.; Jones, M.J. Robust real-time face detection. *Int. J. Comput. Vis.* **2004**, *57*, 137–154. [CrossRef]
14. He, K.; Zhang, X.; Ren, S.; Sun, J. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *37*, 1904–1916. [CrossRef] [PubMed]
15. Tang, X.; Yan, Z.; Peng, J.; Hao, B.; Wang, H.; Li, J. Selective spatiotemporal features learning for dynamic gesture recognition. *Expert Syst. Appl.* **2021**, *169*, 114499. [CrossRef]
16. Rajput, D.S.; Reddy, T.S.K.; Raju, D.N. Investigation on Deep Learning Approach for Big Data. In *Deep Learning and Neural Networks*; IGI Global: Harrisburg, PA, USA, 2018.
17. Bochkovskiy, A.; Wang, C.Y.; Liao, H.M. YOLOv4: Optimal Speed and Accuracy of Object Detection. *arXiv* **2020**, arXiv:2004.10934.
18. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv* **2014**, arXiv:1409.1556.

19. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 770–778. [CrossRef]
20. Vaswani, A.; Shazeer, N.M.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention is All you Need. In Proceedings of the NIPS, Long Beach, CA, USA, 4–9 December 2017.
21. Hu, J.; Shen, L.; Albanie, S.; Sun, G.; Wu, E. Squeeze-and-Excitation Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *42*, 2011–2023. [CrossRef]
22. Zhang, H.; Zu, K.; Lu, J.; Zou, Y.; Meng, D. EPSANet: An Efficient Pyramid Split Attention Block on Convolutional Neural Network. *arXiv* **2021**, arXiv:2105.14447.
23. Haroon, M.; Altaf, S.; Ahmad, S.; Zaindin, M.; Huda, S.; Iqbal, S. Hand Gesture Recognition with Symmetric Pattern under Diverse Illuminated Conditions Using Artificial Neural Network. *Symmetry* **2022**, *14*, 2045. [CrossRef]
24. Zaccagnino, R.; Capo, C.; Guarino, A.; Lettieri, N.; Malandrino, D. Techno-regulation and intelligent safeguards. *Multimed. Tools Appl.* **2021**, *80*, 15803–15824. [CrossRef]
25. Guarino, A.; Malandrino, D.; Zaccagnino, R.; Capo, C.; Lettieri, N. Touchscreen gestures as images. A transfer learning approach for soft biometric traits recognition. *Expert Syst. Appl.* **2023**, *219*, 119614. [CrossRef]
26. Hussain, S.; Saxena, R.; Han, X.; Khan, J.A.; Shin, H. Hand gesture recognition using deep learning. In Proceedings of the 2017 International SoC Design Conference (ISOCC), Seoul, Republic of Korea, 5–8 November 2017; pp. 48–49.
27. Hachaj, T.; Ogiela, M.R.; Koptyra, K. Application of Assistive Computer Vision Methods to Oyama Karate Techniques Recognition. *Symmetry* **2015**, *7*, 1670–1698. [CrossRef]
28. Khan, M.S.; Zualkernan, I.A. Using Convolutional Neural Networks for Smart Classroom Observation. In Proceedings of the 2020 International Conference on Artificial Intelligence in Information and Communication (ICAIIC), Fukuoka, Japan, 19–21 February 2020; pp. 608–612.
29. Ren, X.; Yang, D. Student Behavior Detection Based on YOLOv4-Bi. In Proceedings of the 2021 IEEE International Conference on Computer Science, Artificial Intelligence and Electronic Engineering (CSAIEE), Online, 20–22 August 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 288–291. [CrossRef]
30. Dominio, F.; Donadeo, M.; Zanuttigh, P. Combining multiple depth-based descriptors for hand gesture recognition. *Pattern Recognit. Lett.* **2014**, *50*, 101–111. [CrossRef]
31. Chaudhary, A.; Raheja, J. Light Invariant Real-Time Robust Hand Gesture Recognition. *Optik* **2018**, *159*, 283–294. [CrossRef]
32. Lin, C.; Lin, X.; Xie, Y.; Liang, Y. Abnormal gesture recognition based on multi-model fusion strategy. *Mach. Vis. Appl.* **2018**, *30*, 889–900. [CrossRef]
33. Zhang, Y.C. Gesture Recognition System Based on Improved Stacked Hourglass Structure. In Proceedings of the 2018 International Conference on Computer, Communications and Mechatronics Engineering (CCME 2018), Cuernavaca, Mexico, 26–29 November 2018.
34. Zhang, Z.; Wu, B.; Jiang, Y. Gesture Recognition System Based on Improved YOLO v3. In Proceedings of the 2022 7th International Conference on Intelligent Computing and Signal Processing (ICSP), Xi'an, China, 15–17 April 2022; IEEE: Piscataway, NJ, USA, 2022; pp. 1540–1543. [CrossRef]
35. Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement. *arXiv* **2018**, arXiv:1804.02767.
36. Lecun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based learning applied to document recognition. *Proc. IEEE* **1998**, *86*, 2278–2324. [CrossRef]
37. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.E.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going Deeper with Convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015.
38. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. *arXiv* **2017**, arXiv:1704.04861.
39. Wang, C.Y.; Bochkovskiy, A.; Liao, H.y. Scaled-YOLOv4: Scaling Cross Stage Partial Network. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021; pp. 13024–13033. [CrossRef]
40. Liu, H.; Brock, A.; Simonyan, K.; Le, Q.V. Evolving Normalization-Activation Layers. *arXiv* **2020**, arXiv:2004.02967.
41. Seeland, M.; Mäder, P. Multi-view classification with convolutional neural networks. *PLoS ONE* **2021**, *16*, e0245230. [CrossRef]
42. Simon, T.; Joo, H.; Matthews, I.A.; Sheikh, Y. Hand Keypoint Detection in Single Images using Multiview Bootstrapping. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 1145–1153.
43. Huang, J.; Lin, S.; Wang, N.; Dai, G.; Xie, Y.; Zhou, J. TSE-CNN: A Two-Stage End-to-End CNN for Human Activity Recognition. *IEEE J. Biomed. Health. Inf.* **2020**, *24*, 292–299. [CrossRef]
44. Fernández-Delgado, M.; Cernadas, E.; Barro, S.; Amorim, D. Do We Need Hundreds of Classifiers to Solve Real World Classification Problems? *J. Mach. Learn. Res.* **2014**, *15*, 3133–3181.
45. Feng, R.; Xu, X.; Zhou, X.; Wan, J. A Trust Evaluation Algorithm for Wireless Sensor Networks Based on Node Behaviors and D-S Evidence Theory. *Sensors* **2011**, *11*, 1345–1360. [CrossRef]

46. Narasimhaswamy, S.; Wei, Z.; Wang, Y.; Zhang, J.; Nguyen, M.H. Contextual Attention for Hand Detection in the Wild. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019; pp. 9566–9575. [CrossRef]
47. Gomez-Donoso, F.; Orts-Escolano, S.; Cazorla, M. Large-scale multiview 3D hand pose dataset. *Image Vis. Comput.* **2019**, *81*, 25–33. [CrossRef]